Energy-Efficient Path Planning in Winter Conditions: A Comparative Study of Traditional Baseline, Q-learning, and Proximal Policy Optimization in a Grid World Environment

1.Abstract

Recent investigation in winter route planning has become increasingly important due to increased resistance and reduced efficiency on winter roads. These conditions significantly increase energy consumption and introduce uncertainty, raising the risk of failing to reach the destination. To address the challenge, reinforcement learning (RL) is a type of machine learning where an agent learns to make decisions by interacting with an environment, receiving rewards or penalties for its actions to maximize cumulative rewards. For simulation, we benchmark a standard shortest-path algorithm (BFS) against two reinforcement learning methods: Q-learning and proximal policy optimization (PPO). All approaches are tested on identical grid configurations, with the same starting points, goals, and reward functions. We assess their performance based on cumulative reward, snowy cell visits (as a proxy for energy cost), trajectory characteristics, and success rate. It is hypothesized that proximal policy optimization (PPO) algorithms will result in lower energy consumption than Q-learning and traditional baseline under winter conditions. The results show that while BFS consistently finds the shortest paths, it fails to consider energy costs or environmental uncertainty. RL agents, by comparison, adapt more efficiently to winter conditions. In contrast, Q-learning has an overall better performance in both conditions than PPO. These results suggest that Q-learning is better suited for structured winter navigation in grid worlds, while PPO may perform better in more complex environments with continuous states or decisions.

Key words: Energy-Efficient, Traditional Baseline, Q-learning, Proximal Policy Optimization, grid world, reinforcement learning.

2.Introduction

As demand in electric vehicles become increasingly widespread, energy efficiency and route optimization have emerged as critical challenges, particularly in winter when resistance increases. Freezing temps, plus snow and ice covering the roads, push energy consumption way up, which means shorter range and trips that feel less predictable. A 2025 study shows that an estimated 50 % of EV driving range can be reduced in cold climates, including snow and ice covering terrain, highlighting the significant impact of environmental conditions on energy consumption [1]. Furthermore, when snow or water remains on the road surface, vehicle tires must continuously displace through ice as they roll and move, hence forcing the vehicle to draw more power [2]. On top of that, water cools tires more effectively than air alone, altering their mechanical properties

and further pushing rolling resistance higher. Previous studies have shown that rolling resistance can rise by approximately 30% to 40% under wet or snowy conditions. Ultimately, these factors make winter driving a major concern for EV efficiency and reinforce the need for energy-aware routing strategies.

To approach these challenges, we have explored a few computational methods to allow agents to learn effective navigation strategies aimed at minimizing energy consumption. In this study, a 50x50 grid is used as an abstraction routing map that circles key structures of Canadian snow distribution, rather than exact geographical terrain. This abstraction supports controlled comparison of different routing methods while preserving the essential energy-related difficulties.

More specifically, the research will include a traditional baseline algorithm, and two types of reinforcement learning, which are Q-learning and Proximal Policy optimization algorithm. The traditional baseline routing computes the shortest path based on static cost metrics and follows the predefined route without adaptation or learning (basic routing). Q-learning, a value-based reinforcement learning method, learns through incremental dynamic programming processes with computational requirements and it is well suited for agents to improve and refine action values and achieve effective performance in controlled Markovian domains [3]. In contrast, PPO is a policy-gradient algorithm that primarily optimizes a stochastic policy and achieves greater training stability through constrained policy updates with a clipped surrogate objective function [4].

This comparison evaluates the effectiveness of the proposed routing methods in identifying energy-efficient paths. Performance is evaluated through energy related costs, overall routing efficiency, and observed navigation behavior under both deterministic (non-slip) and stochastic (slip) settings. Because PPO balances stable policy updates with adaptive learning in uncertain environments, it is hypothesized to be the most effective method implemented for winter routing.

The following section goes into the methodology and experimental setup in more detail.

3.Methods

3.1 Environment Design

A custom winter navigation grid environment was designed to evaluate each reinforcement (RL) agent under stochastic and cost-sensitive conditions. The environment consists of a 50 x 50 grid world containing 2500 cells, including both normal terrain and snow-covered terrain. Specifically, two transition settings were evaluated: a slip snow condition and non-slip snow condition (same grid map). The

simulation contains a fixed start (47, 2) and a fixed goal (20,45) point, and are located in the top left corner and middle right respectively. At each time step, the agent can take either one of the four directions, up, down, left, and right; each step on a non-snow grid has an energy cause of -1.5 per step. Snow is modeled as spatially correlated regions with graded intensity, capturing the typical uneven accumulation patterns in Canadian winters. When snow is modeled as a binary state, abrupt reward discontinuities make PPO's advantage estimates unstable, resulting in increased gradient variance. By contrast, a continuous representation of snow thickness provides smoother cost transitions, which help reduce gradient variance and stabilize learning, making it more efficient. These snow types are classified as "Near snow", "Edge snow", and "Core snow".
To evaluate robustness under different winter conditions, two transition settings were considered using the same grid map: a non-slip (deterministic) condition and a slip (stochastic) condition. In the deterministic setting, action always results in the intended movement, and snow cells only add additional energy loss. In the stochastic setting, intended actions may be replaced with unintended movements due to unstable control on icy surfaces, with a higher probability of deviating left or right, simulating loss control of traction during winter driving on ice and snow.

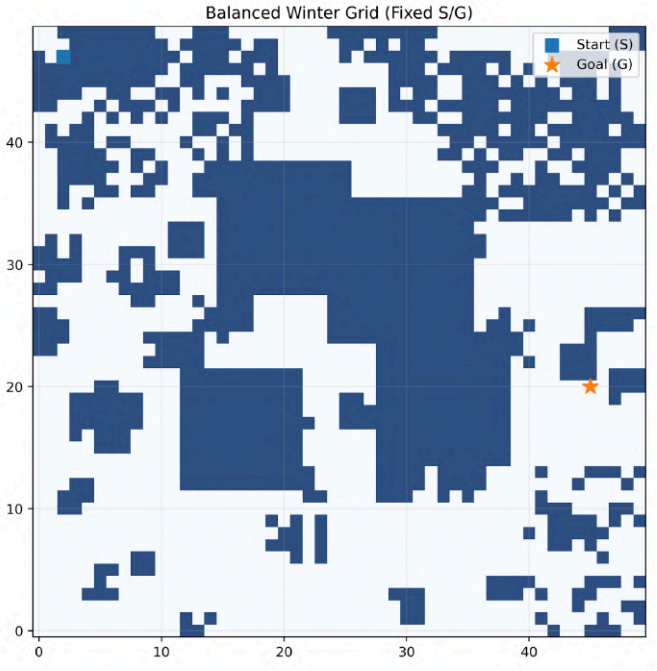


Figure 1: shows a typical winter grid layout used in all experiments.

3.2 Reinforcement Learning Algorithms/ Methodologies:

The routing strategies and learning methods used in this study are described in this section.

3.2.1 Traditional Baseline (BFS)

Traditional Baseline is a non-learning shortest path strategy used as a comparison base for reinforcement learning methods. It operates on the 50x50 grid map, it considers four directions to move (up, down, left, right). The system calculates the shortest path to the destination, without considering energy savings, which means it does not adapt to environmental feedback or uncertainty.

3.2.2 Q-learning Implementation

Tabular Q-learning was used as the value-based reinforcement learning baseline. The agent maintains a discrete Q (s, a) table, which is updated iteratively based on observed state transitions. An ε-greedy policy was used for exploration. The value of ε begins at 0.6 and decays exponentially with a factor of 0.9995 per episode until it reaches 0.05. In this 50×50 grid setting, the decay schedule allows broad initial exploration before shifting emphasis to exploitation. Actions are restricted to four possibilities—left (0), down (1), right (2), and up (3)—consistent with standard discrete grid-world formulations and the discrete grid structure. The Q-value update follows the classic temporal-difference form:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[R + \gamma \max_{a'} Q(s',a') - Q(s,a)] \quad [5]$$

We fix the learning rate α at 0.15 and the discount factor γ at 0.99. These values handle the stochastic transitions (is_slippery=True) and support planning over long horizons in the large grid [3]. State s denotes the flattened grid position index. The action a selects one of the four movement directions. Q(s, a) represents the estimated discounted cumulative reward starting from state s and action a [3,5].

3.2.3 Reward system

In this research, the reward system is designed to represent energy efficiency under winter conditions, it helps agents find the best paths. At each step, the system gives a negative value to show how significant a step is to energy saving, with higher cause on snow grids. A positive 700 is given only when the agents successfully reach the destination, while within the maximum steps of 1000 steps. It encourages agents to reach the destination with considerations on energy saving. We initialize the cumulative reward to 0 at the beginning of each episode. Each will result in negative values, because by doing this, agents don't need to consider the prior bias, since it assumes nothing at first, etc. As in the settings, we have Near-snow (light snow), Edge-snow (medium snow), and Core-snow (deep snow); each of them has different additional values which are -0.2, -0.5, and -2.0 respectively [6].

$$r_t = r_{step} - c_{energy}(s_t) + r_{goal} \quad [6]$$

The basic reward function follows standard reinforcement learning practice by combining step penalties, energy-related costs, and a terminal goal reward.

3.2.4 Proximal Policy Optimization

Proximal Policy Optimization (PPO) served as the policy-gradient method in this study for energy-efficient routing under winter conditions. PPO learns a stochastic policy by directly outputting action probabilities for each state, which helps the agent cope with uncertainty on slippery roads.

$$L^{CLIP}(\theta) = E_t[min(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1 - \epsilon, 1 + \epsilon)\hat{A}_t)] \quad [4]$$

where $r_t(\theta)$ is the probability ratio between the new and previous policies. The clipping mechanism constrains policy updates to improve stability.

The PPO agent was trained using fixed values across all experiments. The discount factor was set to γ=0.99, and Generalized Advantage Estimation (GAE) was used with λ=0.95. The policy network consisted of two fully connected hidden layers with 512 units each, using ReLU activation functions. Entropy regularization was applied to encourage exploration during training. PPO training was conducted for 1,200,000 time steps, using large rollout buffers and multiple optimization epochs to improve training stability [4].

Policy-gradient methods work by directly adjusting a parameterized policy to increase expected reward. They learn a stochastic policy, allowing for more flexibility in uncertain environments, where the same action can lead to different outcomes. In contrast, q-learning estimates state-action values and usually converges with a deterministic policy based on these estimates. Although it can learn in unfamiliar environments, its action selection may be less reliable in highly unpredictable situations. As a result, policy-gradient and value-based methods show varying strengths depending on the level of uncertainty in the winter routing task.

3.2.5 Curriculum Learning for PPO

PPO training followed a two-phase curriculum learning approach designed to improve training stability and sample efficiency under sparse rewards and energy-sensitive penalties [10].

Phase 1: Navigation Learning Phase

Navigation Learning Phase In phase 1, PPO was trained using a simplified reward function that focuses only on positive feedback for reaching the goal. Energy costs in snow grids were not included. This made the feedback denser and more immediate. The agent quickly learned basic navigation and achieved early success in the 50×50 winter grid. The policy learned in this phase served as the starting point for the next training stages.

Phase 2: Energy-Aware Optimization Phase

Energy-Aware Optimization Phase In phase 2, the training continued with the complete energy-aware reward function, which now included penalties for moving across snow-covered areas. The agent had to balance successful arrival with lower energy consumption. All reported results, ablation studies, and comparisons were based solely on the Phase 2 reward function.

3.3 Evaluation factors

To compare the performances of all routing methods under identical conditions, a final evaluation was conducted after training. Each method BFS, Q-learning, and PPO was evaluated using the following metrics:

- Total Reward: calculated as the cumulative reward obtained over the episode.
- Number of steps: steps taken from the start to the goal (less than 1000 steps).
- Average snow cell visited: Snow visits were calculated by counting the number of snow-covered cells traversed in each episode and averaging this value across all evaluation episodes.
- Success rate: defined as the percentage of episodes in which the agent reached the goal within the maximum step limit.

The BFS baseline was evaluated only once on the fixed grid map, as it produces a deterministic path. Reinforcement learning agents were evaluated over multiple episodes. All methods were compared using the same-energy aware reward function and environment configuration to ensure fairness.

A complete summary of environment settings and hyperparameters is provided in Appendix A.

---

4. Results

The experiments compare BFS, Q-learning, and PPO on the same 50×50 winter grid, looking at both non-slip (deterministic) and slip (stochastic) cases.

4.1 Evaluation Setup and Metrics

The grid layout, start and goal positions, and reward parameters were kept exactly the same across all methods. For each run, total reward, number of steps, average snow cells crossed per episode, and success in reaching the goal were recorded.

4.2 Trajectory Visualizations and Quantitative Results

4.2.1 Deterministic (Non-Slip) Winter Condition

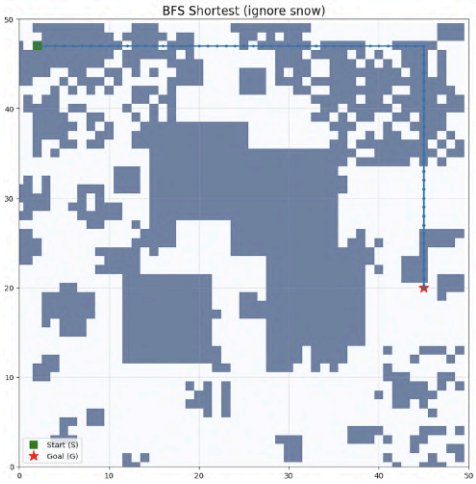


Figure 2: illustrates the BFS path under non-slip conditions. Since BFS always finds the shortest route, the trajectory goes straight from the start to the goal with minimal cells visited without any deviation.

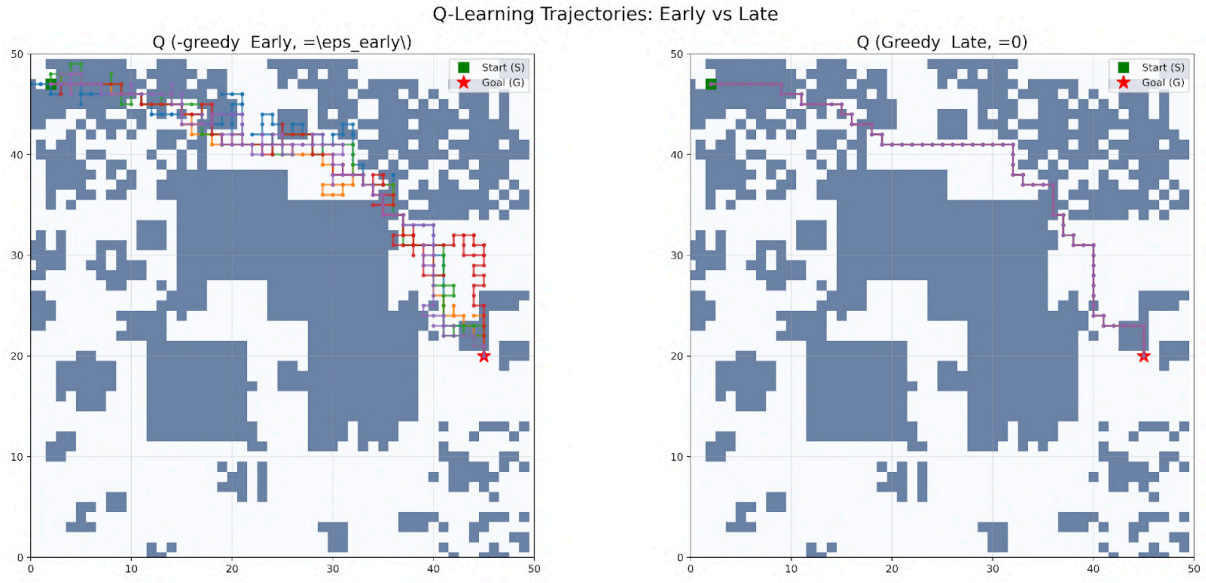


Figure 3: illustrates the navigation trajectories produced by the Q-learning agent under deterministic (non-slip) winter conditions during early training with an ε-greedy policy and late training with a greedy policy. The trajectories demonstrate the transition from exploratory behavior to a stable path toward the goal.

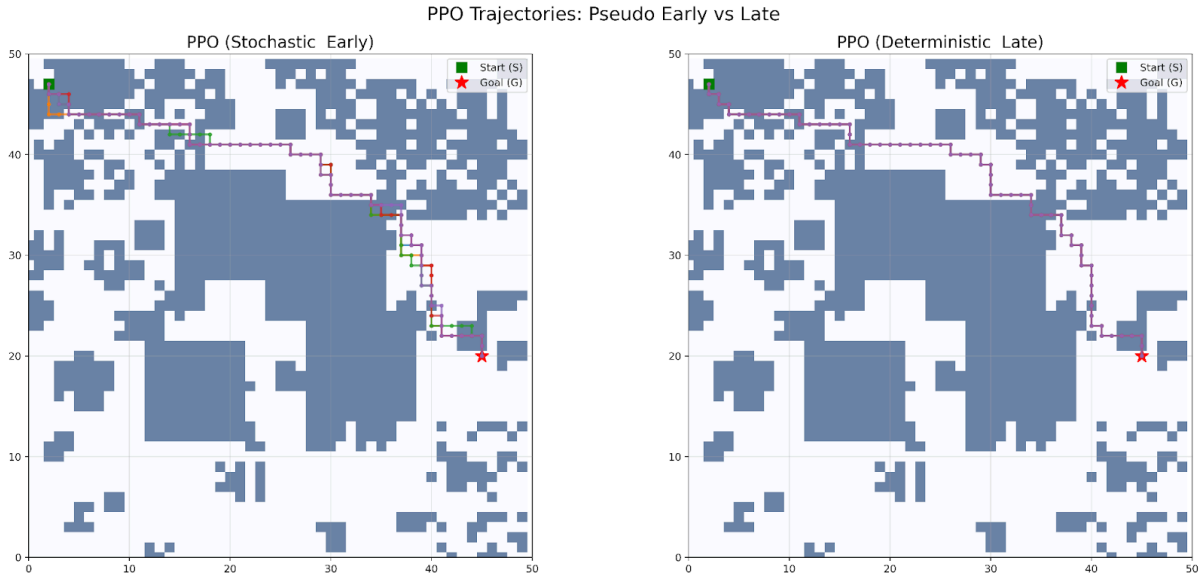


Figure 4: illustrates the navigation trajectories produced by the PPO agent during early and late stages of training under deterministic (non-slip) winter conditions.

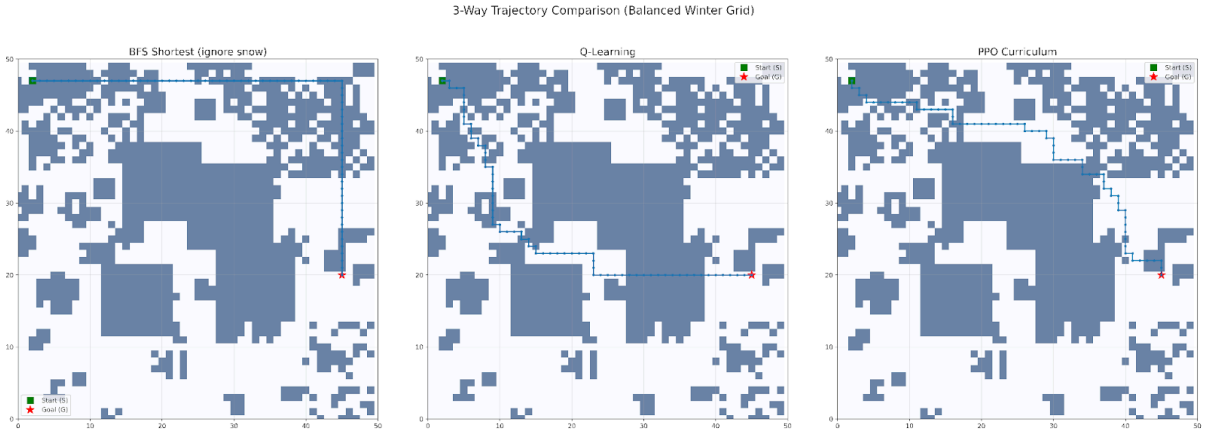


Figure 5: compares the trajectories of BFS, Q-learning and PPO in the non-slip winter setting.

=== Final Comparison Table ===

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	**575.30**	**70.00**	**31.00**	**100.0%**
Q-Learning	**589.50**	**70.00**	**7.00**	**100.0%**
PPO Curriculum	**587.50**	**70.00**	**11.00**	**100.0%**

Table 1: summarizes the main performance metrics for all three methods under deterministic conditions, including total reward, steps to goal, average snow cells crossed, and success rate.

4.2.2 Slip Winter Condition

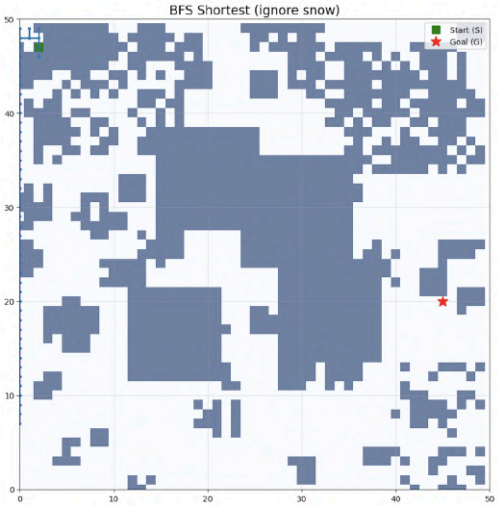


Figure 6: illustrates the navigation path produced by the BFS baseline under slip (stochastic) winter conditions.

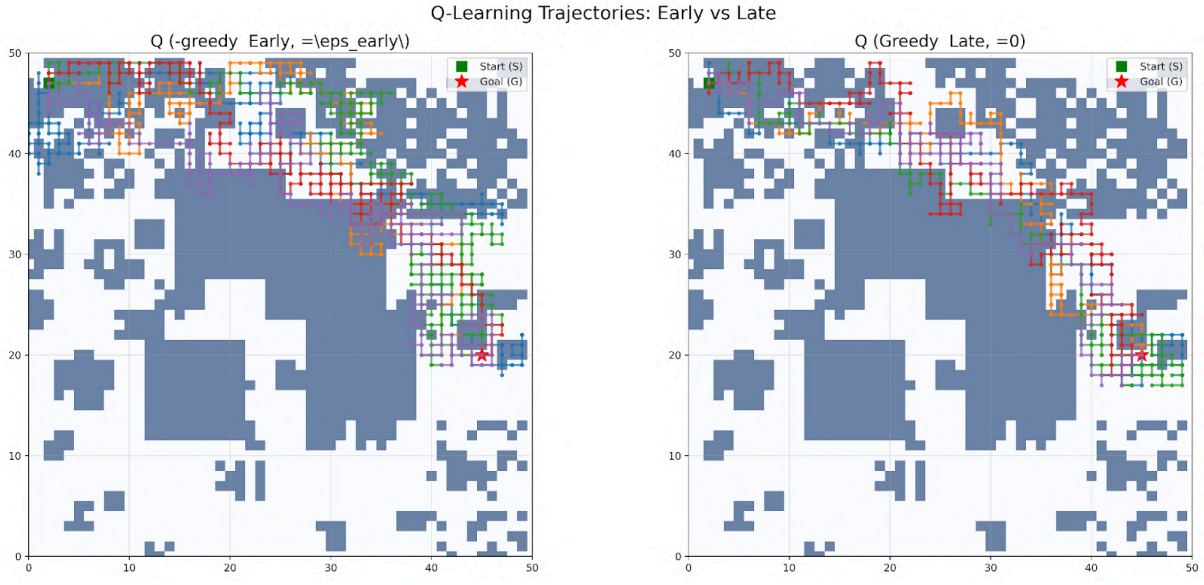


Figure 7: illustrates the navigation trajectories produced by the Q-learning agent during early and late stages of training under slip (stochastic) winter conditions.

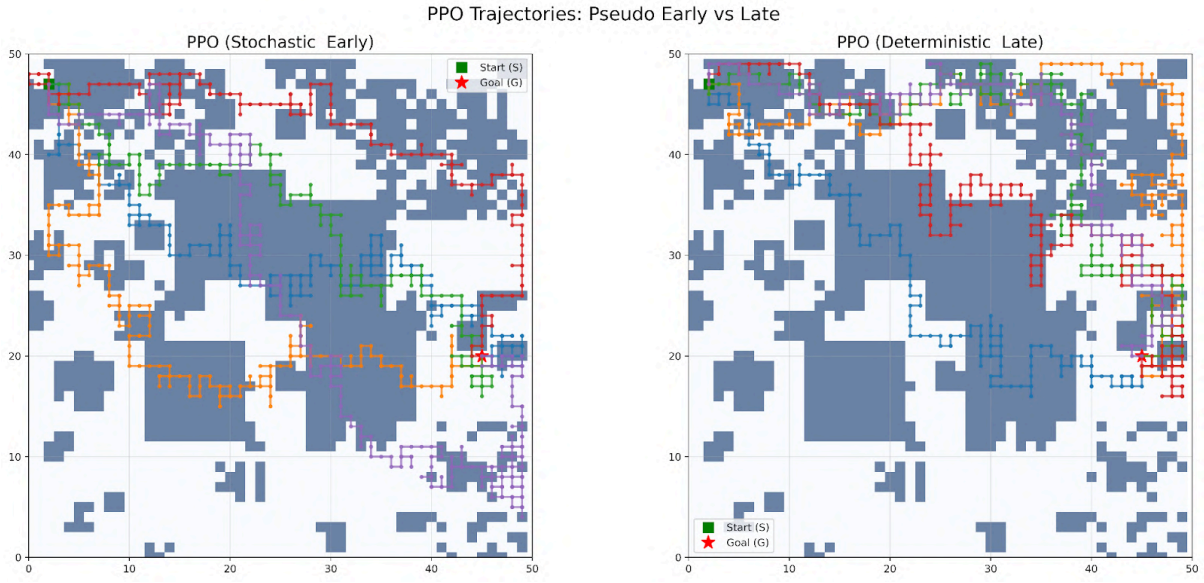


Figure 8: illustrates the navigation trajectories produced by the PPO agent during early and late stages of training under slip (stochastic) winter conditions.

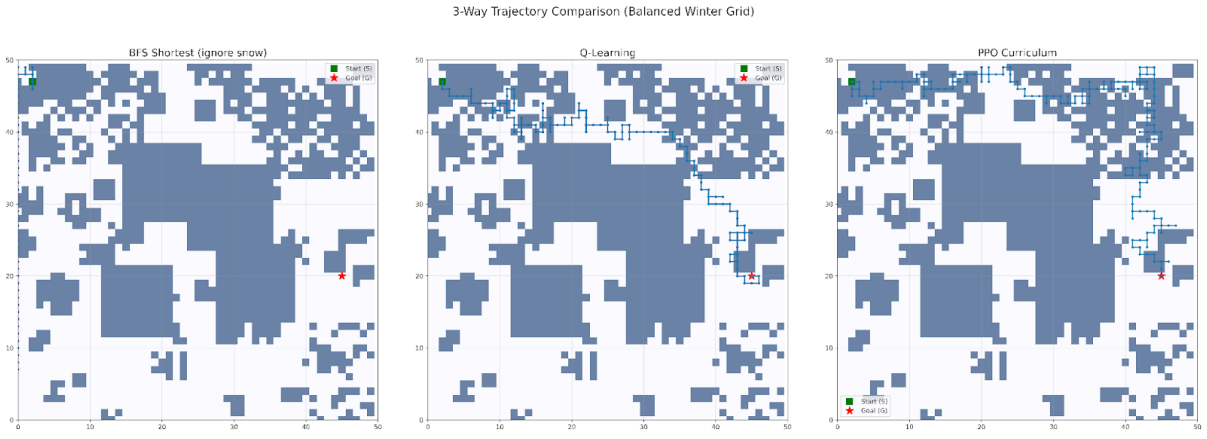


Figure 9: compares the navigation trajectories of BFS, Q-learning, and PPO under slip winter conditions.

=== Final Comparison Table ===

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	**-1639.20**	**1000.00**	**188.00**	**0.0%**
Q-Learning	**309.41**	**224.2**	**65.8**	**100.0%**
PPO Curriculum	**224.1**	**255.8**	**100.9**	**100.0%**

Table 2: summarizes the quantitative performance metrics for BFS, Q-learning, and PPO under slip winter conditions, using the same evaluation metrics as in the deterministic case.

4.3 Spatial State Visitation Heatmaps

Spatial state visitation heatmaps offer a grid-based way to see how often an agent passes through each cell during navigation. In our 50×50 winter grid, each cell corresponds to a state, and the color intensity reflects visitation frequency across episodes. Darker areas indicate cells that are visited more frequently, while lighter or white cells indicate rarely or never visited locations. Such visualizations give a clear picture of the agent's overall path distribution and behavior patterns. We generated these heatmaps using aggregated visitation counts normalized to the [0,1] range.

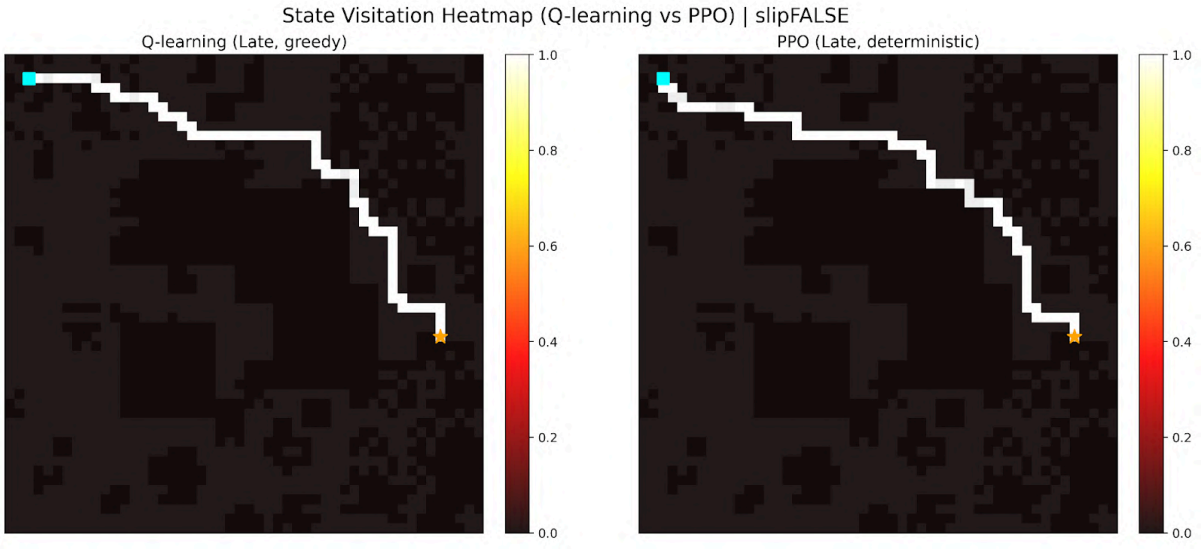


Figure 10: displays the visitation heatmaps for Q-learning and PPO under deterministic (non-slip) winter conditions. The results are aggregated from 10 evaluation episodes. Both agents concentrate most visits along a narrow corridor connecting the start to the goal, with near-maximum intensity (≈1.0) along the primary route and very low visitation elsewhere.

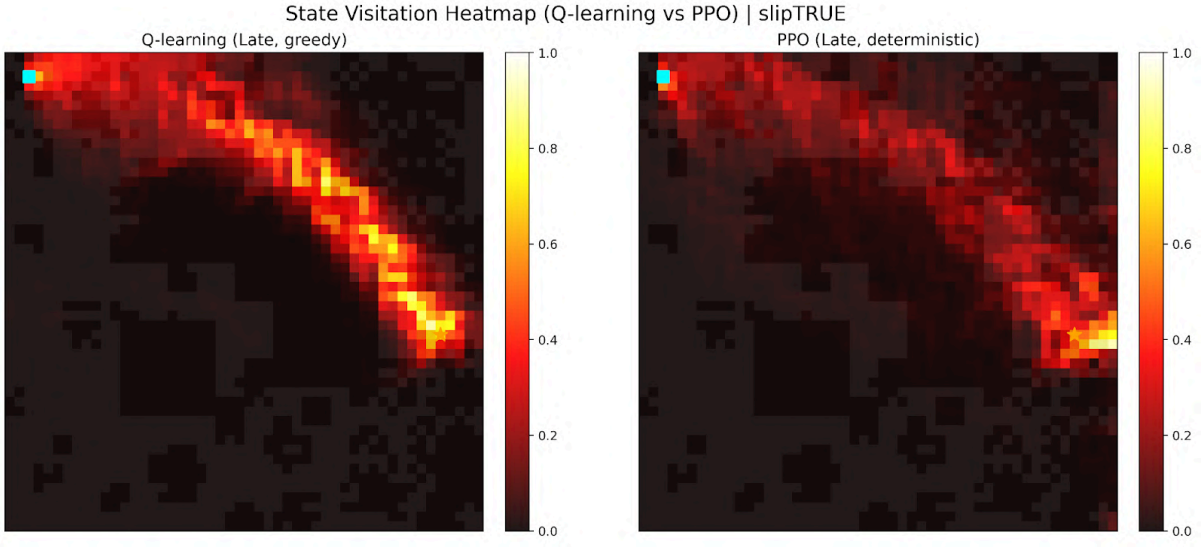


Figure 11: shows the visitation intensity for each grid cell in the slip case. Compared with the non-slip condition, visitation is more widely distributed across the grid rather than remaining within a narrow corridor. The heatmap aggregates data from 10 evaluation episodes.

4.4 Limitations & Uncertainty

Several sources of uncertainty and limitations appeared in this study. Both Q-learning and PPO showed noticeable performance differences across training runs, which was expected given the stochastic nature of the environment and the learning algorithms. Under slip conditions, the same policy could produce quite different routes from episode to episode because actions did not always lead to the expected outcome. In addition, stochastic exploration strategies and random initialization of value functions (or policy networks) exposed agents to slightly different states, transitions, and rewards across runs. This variability made it harder to get perfectly consistent results.

An unexpected issue was that even the BFS baseline sometimes failed to reach the goal within the 1,000-step limit under slip conditions, resulting in a number of unsuccessful evaluation episodes.

A full numerical breakdown of evaluation metrics is provided in Appendix B.

5.Discussion

5.1 Restatement of Hypothesis and Summary of Findings

The study hypothesized that reinforcement learning-based agents, particularly Proximal Policy Optimization, would achieve more energy-efficient winter navigation paths than a traditional shortest-path baseline and a value-based Q-learning agent under both deterministic and slip winter conditions. The experiment result did not support the hypothesis. Q-learning achieved higher cumulative reward than PPO under both

deterministic and slip conditions; it also did not outperform Q-learning in terms of cumulative reward or snow-cell avoidance. Both reinforcement learning methods perform better results than traditional BFS across these environments.

5.2 Interpretation of Q-Learning and PPO Performance

Q-learning achieved a higher performance than Proximal Policy Optimization agent in both deterministic and stochastic winter environments. One possible explanation was that a discrete and clear grid-world environment and relatively small environment would favor more for Q-learning, therefore allowing Q-learning to efficiently estimate optimal state [7]. In contrast, PPO relied on function approximation and stochastic policy updates, which may have required more training data or longer training time to converge to an equally optimal policy. Additionally, its stochastic policy may have introduced suboptimal choices that were not effective, therefore reducing its total rewards relative to Q-learning [3].

5.3 Effects of Slip (Stochastic) Winter Conditions

Under slip (stochastic) conditions, all agents showed broader trajectory dispersion and increased visits to previously visited states because of slipping, as reflected in the spatial state visitation heatmaps. This behavior is consistent with probabilistic transition dynamics, in which the same action could lead to different movement outcomes across episodes, such as deviating left in one run and right in another. When the grid is slippery, agents don't follow one stable path anymore. As a result, the routes spread out, and their performance becomes less consistent across episodes. These observations are consistent with prior navigation studies showing that stochastic environments increase policy variance and reduce convergence stability in reinforcement learning tasks [8].

5.4 Interpretation of BFS Baseline Behavior

The Traditional baseline trajectories did not take account for winter grid costs or stochastic transitions, it focused on the fastest way to the destination. In non-slip conditions this approach naturally produced the optimal route. However, under slip conditions, BFS exceeded the maximum step limit in multiple evaluation episodes, reflecting its inability to adapt its policy in response to transition winter grid costs or stochastic transitions.

Because BFS always assumed deterministic movement, each slip caused deviations from the planned path. These deviations often led to inefficient loops and longer paths. Unlike reinforcement learning methods, BFS did not have reward feedback or policy updates, which limited its ability to adjust navigation behavior under changing environmental dynamics. Therefore, making it the least effective method.

5.5 Limitations and Future Directions

Several limitations should be noted when interpreting these findings. All experiments were carried out in a simulated 50×50 grid environment with a relatively small and

simple state space, which differs significantly from real-world scenarios. This setup likely favored value-based methods like Q-learning, since it could learn precise state-action values for every cell. In contrast, PPO depends on neural network function approximation and stochastic policy updates, which may need more training steps or data to perform well in such discrete, small-scale settings.

The number of evaluation episodes was also limited, which reduced the statistical reliability of the results. Noticeable performance variability was observed across training runs, mainly due to the stochastic environment, random action outcomes, and probabilistic transitions under slip conditions.

The simulated environment simplified real-world winter driving conditions and did not capture vehicle dynamics, sensor noise, road-surface variation, or realistic energy-consumption models, these are factors that might lead to changes. As a result, these findings to real-world winter navigation scenarios were limited [9].

Future research could evaluate reinforcement learning agents in larger scale, continuous, or more realistic winter navigation environments with higher-dimensional state spaces and more complex conditions. Then it may be better to reflect real-world uncertainty and could allow policy-gradient methods such as PPO to fully function their theoretical advantages in handling stochastic and high-dimensional control problems.

---

6. Conclusion

This study compared a traditional BFS baseline with two reinforcement learning methods—Q-learning and Proximal Policy Optimization (PPO)—for energy-efficient navigation in a 50×50 winter grid under both deterministic (non-slip) and stochastic (slip) conditions. The results did not support the original hypothesis that PPO would produce more energy-efficient paths than Q-learning or BFS. Instead, Q-learning achieved the highest cumulative rewards in both settings and outperformed PPO in snow-cell avoidance. Although both reinforcement learning methods clearly outperformed the BFS baseline, BFS struggled under slip conditions due to its inability to adapt to stochastic transitions.

These findings indicate that in small and structured state spaces, value-based methods can be more effective than policy-gradient approaches when environmental complexity is limited. More generally, the results emphasize that the choice of reinforcement learning method should be guided by the structure and constraints of the task, rather than by theoretical advantages alone.

Reference

[1]Mao, R., Xu, W., Qian, Y., Li, X., Li, Y., Li, G., & Zhang, H. (2025). Understanding the Determinants of Electric Vehicle Range: A Multi-Dimensional Survey. *Sustainability*, 17(10), 4259. https://doi.org/10.3390/su17104259

[2]Carlson, A., & Vieira, T. (2021). *The effect of water and snow on the road surface on rolling resistance* (VTI Report 971A). Swedish National Road and Transport Research Institute. https://www.diva-portal.org/smash/get/diva2:1542142/FULLTEXT01.pdf

[3]Watkins, C.J.C.H., Dayan, P. Q-learning. *Mach Learn* 8, 279–292 (1992). https://doi.org/10.1007/BF00992698

[4]Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). *Proximal Policy Optimization Algorithms* (No. arXiv:1707.06347). arXiv. https://doi.org/10.48550/arXiv.1707.06347

[5] Sutton, R. S., & Barto, A. G. (n.d.). Reinforcement Learning: An Introduction. https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf

[6]Dayan, P., & Balleine, B. W. (2002). Reward, Motivation, and Reinforcement Learning. *Neuron*, 36(2), 285–298. https://doi.org/10.1016/S0896-6273(02)00963-7

[7] Tan, C. (2025). Comparative Study of Reinforcement Learning Performance Based on PPO and DQN Algorithms. *Applied and Computational Engineering*, 175(1), 30–36. https://doi.org/10.54254/2755-2721/2025.AST24879

[8]Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A. J., Banino, A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., Kumaran, D., & Hadsell, R. (2017). *Learning to Navigate in Complex Environments* (No. arXiv:1611.03673). arXiv. https://doi.org/10.48550/arXiv.1611.03673

[9]Chukwurah, N., Adebayo, A. S., Ajayi, O. O., & Anfo Pub. (2024). *Sim-to-Real Transfer in Robotics: Addressing the Gap between Simulation and Real- World Performance*. 05(01), 33–39. https://doi.org/10.54660/.IJFMR.2024.5.1.33-39

[10]Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). Curriculum learning. *Proceedings of the 26th Annual International Conference on Machine Learning*, 41–48. https://doi.org/10.1145/1553374.1553380

Appendices

Appendix A Experimental Configuration and Algorithmic Details:

A.1 Environment Configuration

All experiments were conducted in a custom 50×50 winter grid environment (2,500 discrete states).

- Start position: (47, 2)
- Goal position: (20, 45)
- Maximum steps per episode: 1000
- Action space: {up, down, left, right}

Two transition settings were evaluated:

Deterministic (slip = FALSE)

The intended action is executed exactly.

Stochastic (slip = TRUE)

With probability 0.8, the intended action is executed.
With probability 0.2, the agent slips to a random adjacent direction.

All algorithms were evaluated on the same fixed grid layout to ensure fairness.

A.2 Reward Function

The reward function models energy-aware winter navigation.

At each time step:

- `Step penalty: -1.5`
- Snow penalties:
  - `Near snow: -0.2`
  - `Edge snow: -0.5`
  - `Core snow: -2.0`
- Goal reward: +700

The cumulative episode reward is:

$$R = \sum_{t=1}^{T} (-1.5 - C_{snow}(s_t)) + 700 \cdot 1_{goal\ reached}$$

where $C_{snow}(s_t)$ denotes the terrain penalty and $1_{goal\ reached}$ indicates successful termination.

A.3 Q-Learning Configuration

Tabular Q-learning was implemented with the following hyperparameters:

- Learning rate α=0.15
- Discount factor γ=0.99
- Exploration strategy: ε-greedy
- Initial ε = 0.60
- Minimum ε = 0.05
- Exponential decay per episode = 0.9995
- Training episodes = 15,000
- Max steps per episode = 1000

The Q-update rule is:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[R + \gamma \max_{a'} Q(s',a') - Q(s,a)]$$

During evaluation, a fully greedy policy (ε = 0) was used.

A.4 Proximal Policy Optimization (PPO) Curriculum Configuration

PPO was implemented as a stochastic policy-gradient method.

Total training timesteps: 1,200,000

- Phase 1 (deterministic environment): 300,000 timesteps
- Phase 2 (stochastic environment): 900,000 timesteps

Evaluation was conducted using deterministic action selection.

PPO Objective Function

PPO optimizes the clipped surrogate objective:

$$L^{CLIP}(\theta) = E_t[min(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1 - \epsilon, 1 + \epsilon)\hat{A}_t)]$$

where

$$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$$

and A^tA^t is the generalized advantage estimate.

The clipping mechanism constrains policy updates to maintain stability.

PPO Hyperparameters

- Discount factor γ=0.99γ=0.99
- GAE parameter λ=0.95λ=0.95
- Entropy regularization enabled
- Neural network architecture: two hidden layers (512 units each, ReLU activation)

A.5 Evaluation Protocol

For each condition (slip FALSE / TRUE):

- One trained model per algorithm
- Maximum evaluation length: 1000 steps
- Metrics recorded:

- Total cumulative reward
- Number of steps
- Snow cell visits
- Success rate (%)

State visitation heatmaps were generated by aggregating visitation frequencies across evaluation runs.

Appendix B Detailed Quantitative Results:

B.1 Deterministic (slip = FALSE)

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	**575.30**	**70.00**	**31.00**	**100.0%**
Q-Learning	**589.50**	**70.00**	**7.00**	**100.0%**
PPO Curriculum	**587.50**	**70.00**	**11.00**	**100.0%**

All methods reached the goal within 70 steps.
Q-learning achieved the highest cumulative reward due to reduced snow traversal.

B.2 Stochastic (slip = TRUE)

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	**-1639.2**	**1000**	**188.0**	**0%**
Q-Learning	**309.4**	**224.2**	**65.8**	**100.0%**
PPO Curriculum	**224.1**	**255.8**	**100.9**	**100.0%**

Under stochastic dynamics:

- BFS fails due to inability to adapt.
- Q-learning demonstrates greater robustness.
- PPO maintains success but exhibits higher trajectory dispersion.

B.3 State Visitation Analysis

State visitation frequency was computed as:

$$V(s) = \frac{N(s)}{max_s N(s)}$$

where $N(s)$ is the number of visits to state s.

- Under deterministic conditions, visitation concentrates along a narrow corridor.
- Under stochastic conditions, visitation becomes more dispersed.
- Q-learning exhibits more focused routing than PPO under slip conditions.

Review: Energy-Efficient Path Planning in Winter Conditions

This paper addresses a timely and relevant problem (energy-efficient routing for electric vehicles under winter conditions) by comparing BFS, Q-learning, and PPO in a controlled grid-world environment. The experimental design is a clear strength: using identical configurations, reward structures, and evaluation metrics allows for a fair comparison across methods. I also particularly appreciate the inclusion of a traditional BFS baseline, which provides helpful context for interpreting the performance gains of different reinforcement learning approaches. Additionally, the comparison between deterministic ("non-slip") and stochastic ("slip") environments helps provide additional comparisons and connects the abstraction back to meaningful real world variables.

Furthermore, this paper offers interesting insights into when value-based methods (Q-learning) may outperform policy-based approaches (PPO), especially in structured, discrete environments. This contributes to a broader understanding of how different RL paradigms perform across task types and environmental complexity.

That said, several areas could be strengthened. First, the introduction would benefit from more consistent and rigorous citation support. For example, claims such as "rolling resistance can rise by approximately 30–40% under wet or snowy conditions" and statements about the validity of the grid abstraction would be more convincing if directly supported with citations. Strengthening this connection to prior work would improve both credibility and scholarly rigor.

Second, while the motivation is clearly articulated, the paper would benefit from a more explicit discussion of gaps in existing research. The authors highlight the importance of multi-criteria navigation under winter conditions, but it remains unclear what specific limitation in the current literature this study addresses. Clarifying whether the primary contribution is extending RL-based navigation to account for weather-related energy costs (or something more novel) would help better position the work.

Third, aspects of methodological rigor could be improved. The paper would benefit from more justification of hyperparameter choices and additional experimental robustness, such as averaging results over multiple runs and reporting variance. This would strengthen confidence in the findings, especially given the stochastic elements of the environment.

Overall, this is a thoughtfully designed study with promising contributions. With improved citation support, clearer positioning within the literature, and stronger experimental validation, the paper could make a meaningful contribution. I recommend **accept with major revisions**.

Decision: **Accept with (minor) revisions**

Review:

The paper compares three ways of planning routes in a winter grid world: a shortest-path BFS baseline, Q-learning, and PPO, using the same map, rewards, start and goal, and evaluation metrics for all three. Across both non-slip and slip settings, Q-learning comes out best overall: in the deterministic case, it reaches the goal with fewer snow-cell visits than the others, and in the slippery case, it clearly beats PPO while BFS breaks down. The paper's main conclusion is that, in a small structured environment like this one, value-based reinforcement learning can be more effective than PPO for energy-aware winter navigation, though the setup is still a simplified simulation rather than a real driving model.

Its strongest feature is the head-to-head comparison itself: the paper puts BFS, Q-learning, and PPO in the same winter grid setup and evaluates them with the same reward structure and metrics, so the differences between the methods come through clearly. The paper's main argument is generally clear, and the writing is easy to understand. The Results and Discussion sections are the strongest, with direct and specific comparisons. In contrast, the Methods section needs the most improvement, especially the explanations of the environment, how rewards were given, and the training procedures. That is where the paper becomes most repetitive and hardest to follow. Tightening that section and making the setup more direct would do the most to improve the paper.

The paper follows in a sensible order overall, and the main steps of the argument are easy to follow. The paper moves from introduction to methods to results in a straightforward way, and no major reordering is needed- besides the method section. The paper includes a Works Cited / References section, and there are corresponding citations throughout the paper. The main issue is minor consistency and formatting, not the absence of citation.

The Methods section would benefit from a clearer presentation of the experimental setup. At present, important details are split between the main text and Appendix A. I would suggest bringing the core information together in one place and stating it plainly: the 50×50 grid, the start and goal positions, the slip probabilities, the reward and penalty values, the training length for Q-learning and PPO, and the evaluation procedure. It would also help to replace general phrases such as “large rollout buffers” and “multiple optimization epochs” with the actual PPO settings, and to state directly how many evaluation episodes were used. This is a manageable revision, but it would make the paper much easier to follow.

The paper is already in good standard, but the above suggested changes will make the article easier to read. The paper can be easily accepted after the revision.

Energy-Efficient Path Planning in Winter Conditions: A Comparative Study of Traditional Baseline, Q-learning, and Proximal Policy Optimization in a Grid World Environment

1.Abstract

Recent investigation in winter route planning has become increasingly important due to increased resistance and reduced efficiency on winter roads. These conditions significantly increase energy consumption and introduce uncertainty, raising the risk of failing to reach the destination. To address the challenge, Reinforcement Learning (RL) is a type of machine learning where an agent learns to make decisions by interacting with an environment, receiving rewards or penalties for its actions to maximize cumulative rewards. For simulation, we benchmark a standard shortest-path algorithm (BFS) against two reinforcement learning methods: Q-learning and Proximal Policy Optimization (PPO). All approaches are tested on identical grid configurations, with the same starting points, goals, and reward functions. We assess their performance based on cumulative reward, snowy cell visits (as a proxy for energy cost), trajectory characteristics, and success rate. It is hypothesized that PPO algorithms will result in lower energy consumption than Q-learning and traditional baseline under winter conditions. The results show that while BFS consistently finds the shortest paths, it fails to consider energy costs or environmental uncertainty. RL agents, by comparison, adapt more efficiently to winter conditions. Under deterministic (non-slip) conditions, PPO achieved higher cumulative reward and fewer snow cell visits than Q-learning, partially supporting the hypothesis. However, under stochastic (slip) conditions, Q-learning outperformed PPO in cumulative reward and snow-cell avoidance. These results suggest that Q-learning is better suited for stochastic winter navigation in grid worlds, while PPO may perform better in more deterministic environments with continuous states or decisions.

Key words: Energy-Efficient, Traditional Baseline, Q-learning, Proximal Policy Optimization, grid world, reinforcement learning.

2.Introduction

As demand in electric vehicles become increasingly widespread, energy efficiency and route optimization have emerged as critical challenges, particularly in winter when resistance increases. Freezing temps, plus snow and ice covering the roads, push energy consumption way up, which means shorter range and trips that feel less predictable. A 2025 study shows that an estimated 50 % of EV driving range can be reduced in cold climates, including snow and ice covering terrain, highlighting the significant impact of environmental conditions on energy consumption [1]. Furthermore, when snow or

water remains on the road surface, vehicle tires must continuously displace through ice as they roll and move, hence forcing the vehicle to draw more power [2]. On top of that, water cools tires more effectively than air alone, altering their mechanical properties and further pushing rolling resistance higher. Previous studies have shown that rolling resistance can rise by approximately 30% to 40% under wet or snowy conditions [2]. Ultimately, these factors make winter driving a major concern for Electric Vehicle (EV) efficiency and reinforce the need for energy-aware routing strategies.

To approach these challenges, we have explored a few computational methods to allow agents to learn effective navigation strategies aimed at minimizing energy consumption. In this study, a 50x50 grid is used as an abstraction routing map that circles key structures of Canadian snow distribution, rather than exact geographical terrain. Grid-based environments are commonly used in reinforcement learning studies, as they simplify complex continuous real-world environments into discrete states [5]. This abstraction is particularly well-suited for this study, as it enables environmental factors such as snow coverage and surface slip conditions to be incorporated directly into the cell-level cost and reward structure, supporting controlled and reproducible comparison of different routing methods.

Despite growing interest in reinforcement learning for navigation, existing studies have largely overlooked the impact of winter environmental conditions, such as snow coverage and surface slippiness, on energy-aware path planning. Prior work has primarily focused on general navigation without taking account of explicitly modeling weather-dependent energy costs [8]. To address this gap, this study integrates snow coverage, stochastic slip condition, and energy costs into a comparative framework, evaluating Q-learning, PPO, and a traditional Baseline under realistic winter routing scenarios. This study is one of the first to directly compare value-based and policy-gradient reinforcement learning methods in an energy-aware winter navigation setting .

More specifically, the research will include a traditional baseline algorithm, and two types of reinforcement learning, which are Q-learning and PPO algorithm. The traditional baseline routing computes the shortest path based on static cost metrics and follows the predefined route without adaptation or learning (basic routing). Q-learning, a value-based reinforcement learning method, learns through incremental dynamic programming processes with computational requirements and it is well suited for agents to improve and refine action values and achieve effective performance in controlled Markovian domains [3]. In contrast, PPO is a policy-gradient algorithm that primarily optimizes a stochastic policy and achieves greater training stability through constrained policy updates with a clipped surrogate objective function [4].

This comparison evaluates the effectiveness of the proposed routing methods in identifying energy-efficient paths. Performance is evaluated through energy related

costs, overall routing efficiency, and observed navigation behavior under both deterministic (non-slip) and stochastic (slip) settings. Because PPO balances stable policy updates with adaptive learning in uncertain environments, it is hypothesized to be the most effective method implemented for winter routing.

The following section goes into the methodology and experimental setup in more detail.

3.Methods

3.1 Environment Design

A custom winter navigation grid environment was designed to evaluate each Reinforcement Learning (RL) agent under stochastic and cost-sensitive conditions. The environment consists of a 50 x 50 grid world containing 2500 cells, including both normal terrain and snow-covered terrain. The simulation contains a fixed start (47, 2) and a fixed goal (20, 45) point, and are located in the top left corner and middle right respectively. At each time step, the agent can take either one of the four directions, up, down, left, and right; each step on a non-snow grid has an energy cost of -1.5 per step. Snow is modeled as spatially correlated regions with graded intensity, capturing the typical uneven accumulation patterns in Canadian winters.

When snow is modeled as a binary state, abrupt reward discontinuities make PPO's advantage estimates unstable, resulting in increased gradient variance. By contrast, a continuous representation of snow thickness provides smoother cost transitions, which help reduce gradient variance and stabilize learning, making it more efficient. These snow types are classified as "Near snow", "Edge snow", and "Core snow".

To evaluate robustness under different winter conditions, two transition settings were considered using the same grid map: a non-slip (deterministic) condition and a slip (stochastic) condition. In the deterministic setting, action always results in the intended movement, and snow cells only add additional energy loss. In the stochastic setting, actions do not always lead to the intended outcome: with a probability of 1/3, the intended action is executed, and with a probability of 2/3, the agent moves in a random adjacent direction, simulating loss of traction control during winter driving on ice and snow.

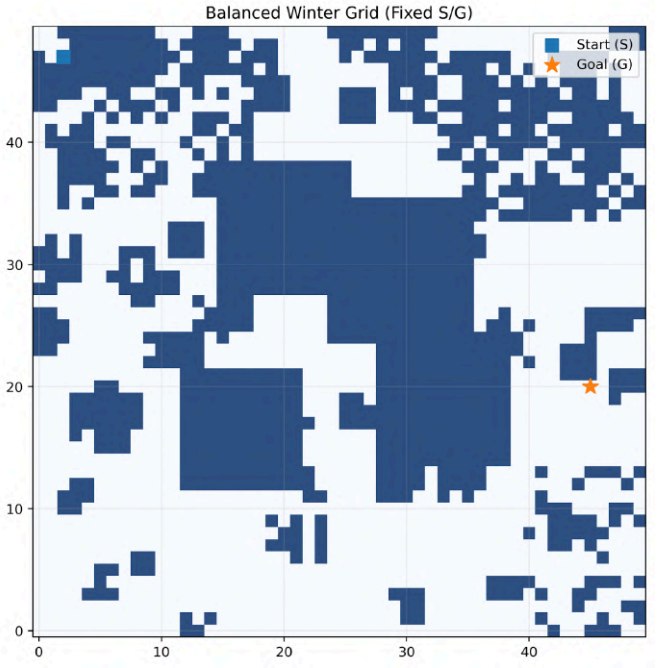


Figure 1: shows a typical winter grid layout used in all experiments.

3.2 Reinforcement Learning Algorithms/ Methodologies:

The routing strategies and learning methods used in this study are described in this section.

3.2.1 Traditional Baseline (BFS)

Traditional baseline is a non-learning shortest path strategy used as a comparison base for reinforcement learning methods. It operates on the 50x50 grid map, it considers four directions to move (up, down, left, right). The system calculates the shortest path to the destination, without considering energy savings, which means it does not adapt to environmental feedback or uncertainty.

3.2.2 Q-learning Implementation

Tabular Q-learning was used as the value-based reinforcement learning baseline. The agent maintains a discrete Q (s, a) table, which is updated iteratively based on observed state transitions. An ε-greedy policy was used for exploration. The value of ε begins at 0.6 and decays exponentially with a factor of 0.9995 per episode until it reaches 0.05. In this 50×50 grid setting, the decay schedule allows broad initial exploration before shifting emphasis to exploitation. Actions are restricted to four possibilities—left (0), down (1), right (2), and up (3)—consistent with standard discrete grid-world formulations and the discrete grid structure. The Q-value update follows the classic temporal-difference form:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[R + \gamma \max_{a'} Q(s',a') - Q(s,a)] \quad [5]$$

We fix the learning rate α at 0.15 and the discount factor γ at 0.99. These values handle the stochastic transitions (is_slippery=True) and support planning over long horizons in the large grid [3]. State s denotes the flattened grid position index. The action a selects one of the four movement directions. Q(s, a) represents the estimated discounted cumulative reward starting from state s and action a [3,5].

3.2.3 Reward system

In this research, the reward system is designed to represent energy efficiency under winter conditions, it helps agents find the best paths. At each step, the system gives a negative value to show how significant a step is to energy saving, with higher cause on snow grids. A positive 700 is given only when the agents successfully reach the destination, while within the maximum steps of 1000 steps. It encourages agents to reach the destination with considerations on energy saving. We initialize the cumulative reward to 0 at the beginning of each episode. Each will result in negative values, because by doing this, agents don't need to consider the prior bias, since it assumes nothing at first, etc. As in the settings, we have Near-snow (light snow), Edge-snow (medium snow), and Core-snow (deep snow); each of them has different additional values which are -0.2, -0.5, and -2.0 respectively [6].

$$r_t = r_{step} - c_{energy}(s_t) + r_{goal} \quad [6]$$

The basic reward function follows standard reinforcement learning practice by combining step penalties, energy-related costs, and a terminal goal reward.

3.2.4 Proximal Policy Optimization

Proximal Policy Optimization (PPO) served as the policy-gradient method in this study for energy-efficient routing under winter conditions. PPO learns a stochastic policy by directly outputting action probabilities for each state, which helps the agent cope with uncertainty on slippery roads.

$$L^{CLIP}(\theta) = E_t[min(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1 - \epsilon, 1 + \epsilon)\hat{A}_t)] \quad [4]$$

where $r_t(\theta)$ is the probability ratio between the new and previous policies. The clipping mechanism constrains policy updates to improve stability.

The PPO agent was trained using fixed values across all experiments. The discount factor was set to γ=0.99, and Generalized Advantage Estimation (GAE) was used with λ=0.95. The policy network consisted of two fully connected hidden layers with 256 units each, using ReLU activation functions. The learning rate was set to 3×10^{-4}, with a clipping range of 0.2 and an entropy coefficient of 0.02 to encourage exploration. PPO training was

conducted for 1,200,000 time steps, using a rollout buffer of 2048 steps, a batch size of 256, and 10 optimization epochs per update to improve training stability [4].

Policy-gradient methods work by directly adjusting a parameterized policy to increase expected reward. They learn a stochastic policy, allowing for more flexibility in uncertain environments, where the same action can lead to different outcomes. In contrast, q-learning estimates state-action values and usually converges with a deterministic policy based on these estimates. Although it can learn in unfamiliar environments, its action selection may be less reliable in highly unpredictable situations. As a result, policy-gradient and value-based methods show varying strengths depending on the level of uncertainty in the winter routing task.

3.2.5 Curriculum Learning for PPO

PPO training followed a two-phase curriculum learning approach designed to improve training stability and sample efficiency under sparse rewards and energy-sensitive penalties [10].

Phase 1: Navigation Learning Phase (300,000 timesteps)
In phase 1, PPO was trained using a simplified reward function that focuses only on positive feedback for reaching the goal. Energy costs in snow grids were not included. This made the feedback denser and more immediate. The agent quickly learned basic navigation and achieved early success in the 50×50 winter grid. The policy learned in this phase served as the starting point for the next training stages.

Phase 2: Energy-Aware Optimization Phase (900,000 timesteps)
In phase 2, the training continued with the complete energy-aware reward function, which now included penalties for moving across snow-covered areas. The agent had to balance successful arrival with lower energy consumption. All reported results, ablation studies, and comparisons are based solely on the Phase 2 reward function.

3.3 Evaluation factors

To ensure statistical reliability, all reinforcement learning experiments were repeated over 5 independent random seeds, and results are reported as the mean and standard deviation across runs.

To compare the performances of all routing methods under identical conditions, a final evaluation was conducted after training. Each method BFS, Q-learning, and PPO was evaluated using the following metrics:

- Total Reward: calculated as the cumulative reward obtained over the episode.
- Number of steps: steps taken from the start to the goal (less than 1000 steps).

- Average snow cell visited: Snow visits were calculated by counting the number of snow-covered cells traversed in each episode and averaging this value across all evaluation episodes.

- Success rate: defined as the percentage of episodes in which the agent reached the goal within the maximum step limit.

The BFS baseline was evaluated only once on the fixed grid map, as it produces a deterministic path. Reinforcement learning agents were evaluated over 200 episodes per run, with experiments repeated over 5 independent random seeds for each condition (slip and non-slip). All methods were compared using the same energy-aware reward function and environment configuration to ensure fairness.

3.4 Hyperparameter Selections

Hyperparameters are selected based on standard practices and empirical validation to ensure stable and consistent performance [3,4].

3.4.1 Q-Learning Hyperparameter Configuration

We implement tabular Q-learning with an ε-greedy exploration strategy.

The hyperparameters are set as follows:

- Learning rate $\alpha = 0.15$
- Discount factor $\gamma = 0.99$
- Exploration strategy: ε-greedy
- Initial $\varepsilon = 0.60$
- Minimum $\varepsilon = 0.05$
- Exponential decay rate = 0.9995
- Training episodes = 15,000
- Maximum steps per episode = 1,000

The learning rate ($\alpha = 0.15$) is selected to balance convergence speed and stability. A moderate learning rate allows the agent to adapt efficiently while avoiding oscillations in value updates.

The discount factor ($\gamma = 0.99$) emphasizes long-term rewards, which is essential for navigation tasks where the objective is to reach the goal efficiently over multiple steps.

An ε-greedy exploration strategy is adopted to balance exploration and exploitation. The initial exploration rate ($\varepsilon = 0.60$) encourages sufficient exploration in early training, while exponential decay gradually shifts the policy toward exploitation. The minimum ($\varepsilon = 0.05$) ensures that some level of exploration is maintained throughout training, preventing the agent from getting stuck in suboptimal policies.

The number of training episodes 15,000 and maximum steps per episode 1,000 are chosen to ensure sufficient interaction with the environment for convergence, while maintaining computational efficiency.

During evaluation, a fully greedy policy ($\varepsilon = 0$) is used to assess the learned policy performance.

3.4.2 PPO Hyperparameter Configuration

We implement PPO using a clipped surrogate objective with generalized advantage estimation (GAE).

The hyperparameters are set as follows:

- Discount factor $\gamma = 0.99$
- GAE parameter $\lambda = 0.95$
- Learning rate = 3×10^{-4}
- Clipping range $\varepsilon = 0.2$
- Rollout buffer size = 2048 steps
- Batch size = 256
- Optimization epochs per update = 10
- Entropy regularization is enabled

The policy and value networks share a neural architecture consisting of two hidden layers with 256 units each and ReLU activation.

The discount factor ($\gamma = 0.99$) is selected to emphasize long-term rewards, which is essential for navigation tasks. The GAE parameter ($\lambda = 0.95$) provides a balance between bias and variance in advantage estimation, leading to more stable learning.

The clipping parameter ($\varepsilon = 0.2$) constrains policy updates to prevent large deviations from the previous policy, which improves training stability. The rollout buffer size and number of optimization epochs are chosen to ensure sufficient policy updates while avoiding overfitting to recent trajectories.

Entropy regularization is included to encourage exploration and prevent premature convergence to suboptimal policies.

Overall, these hyperparameters follow widely adopted settings in reinforcement learning literature and are chosen to ensure stable and consistent training rather than aggressive performance tuning.

A complete summary of environment settings and hyperparameters is provided in Appendix A.

4. Results

The experiments compare BFS, Q-learning, and PPO on the same 50×50 winter grid, looking at both non-slip (deterministic) and slip (stochastic) cases.

4.1 Evaluation Setup and Metrics

The grid layout, start and goal positions, and reward parameters were kept exactly the same across all methods. All experiments were repeated over 5 independent random seeds, and results are reported as the mean across runs. Trajectory figures shown are from the representative seed whose performance was closest to the mean. For each run, total reward, number of steps, average snow cells crossed per episode, and success in reaching the goal were recorded.

4.2 Trajectory Visualizations and Quantitative Results

4.2.1 Deterministic (Non-Slip) Winter Condition

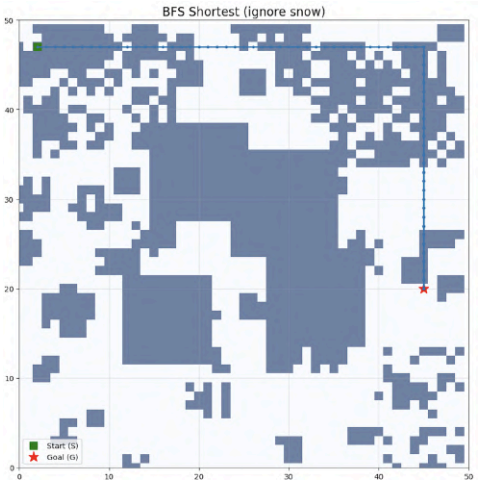


Figure 2: illustrates the BFS path under non-slip conditions. Since BFS always finds the shortest route, the trajectory goes straight from the start to the goal with minimal cells visited without any deviation.

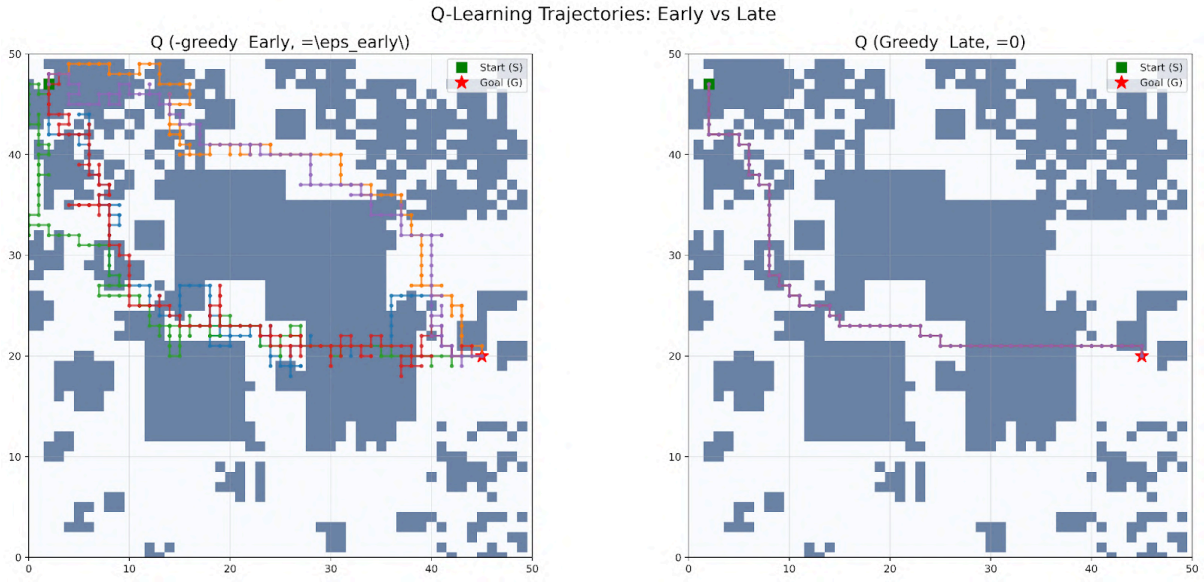


Figure 3: illustrates the navigation trajectories produced by the Q-learning agent under deterministic (non-slip) winter conditions during early training with an ε-greedy policy and late training with a greedy policy. The trajectories demonstrate the transition from exploratory behavior to a stable path toward the goal.

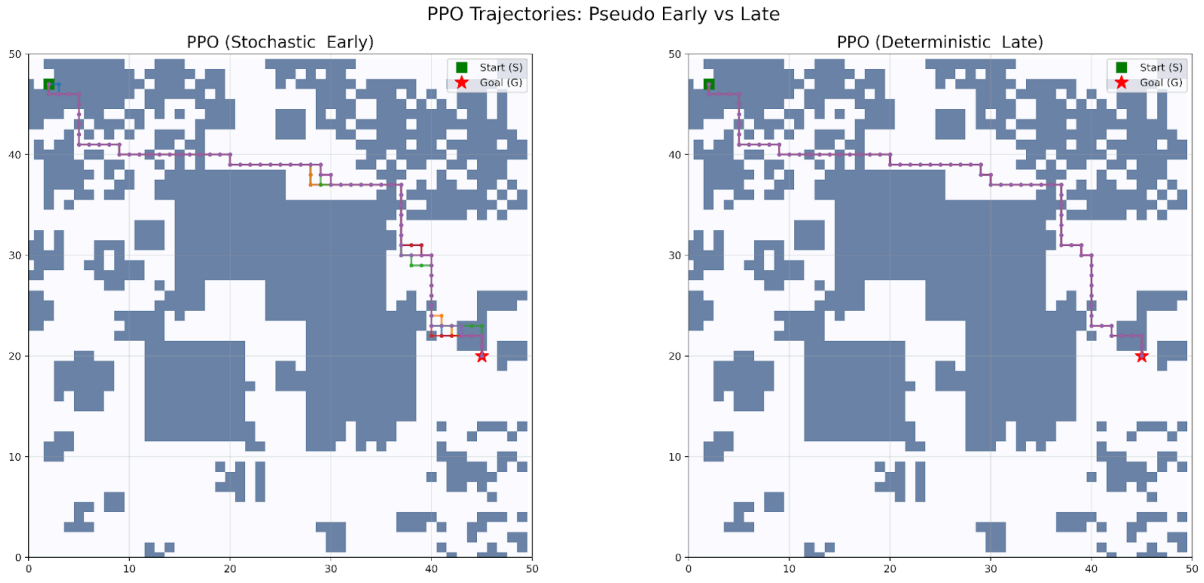


Figure 4: illustrates the navigation trajectories produced by the PPO agent during early and late stages of training under deterministic (non-slip) winter conditions.

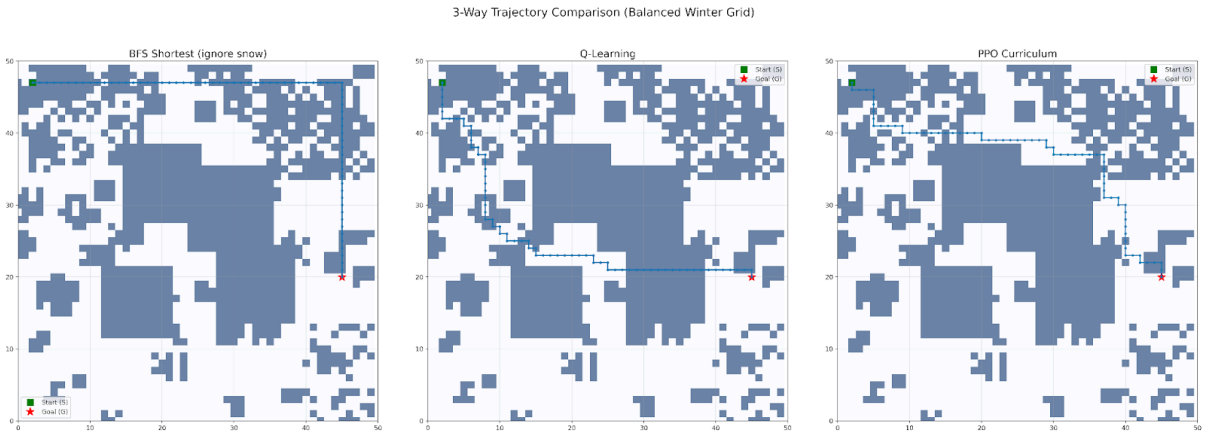


Figure 5: compares the trajectories of BFS, Q-learning and PPO in the non-slip winter setting.

=== Final Comparison Table ===

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	575.30±0.00	70.00±0.00	31.00±0.00	100.0%
Q-Learning	579.10±5.61	70.00±0.00	12.60±3.29	100.0%
PPO Curriculum	588.50±0.60	70.00±0.00	7.80±1.10	100.0%

Table 1: summarizes the main performance metrics for all three methods under deterministic conditions, including total reward, steps to goal, average snow cells crossed, success rate, standard deviation, and averaged across 5 independent random seeds.

4.2.2 Slip Winter Condition

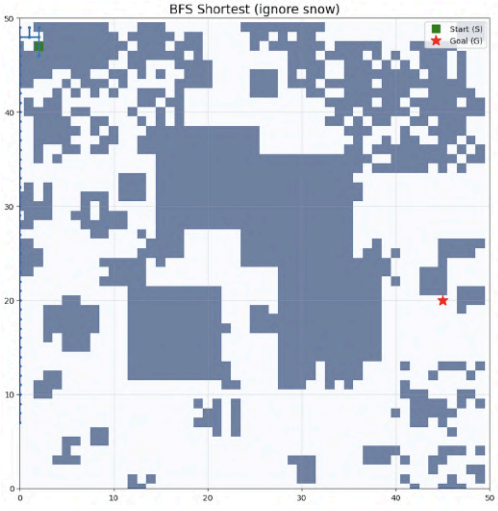


Figure 6: illustrates the navigation path produced by the BFS baseline under slip (stochastic) winter conditions.

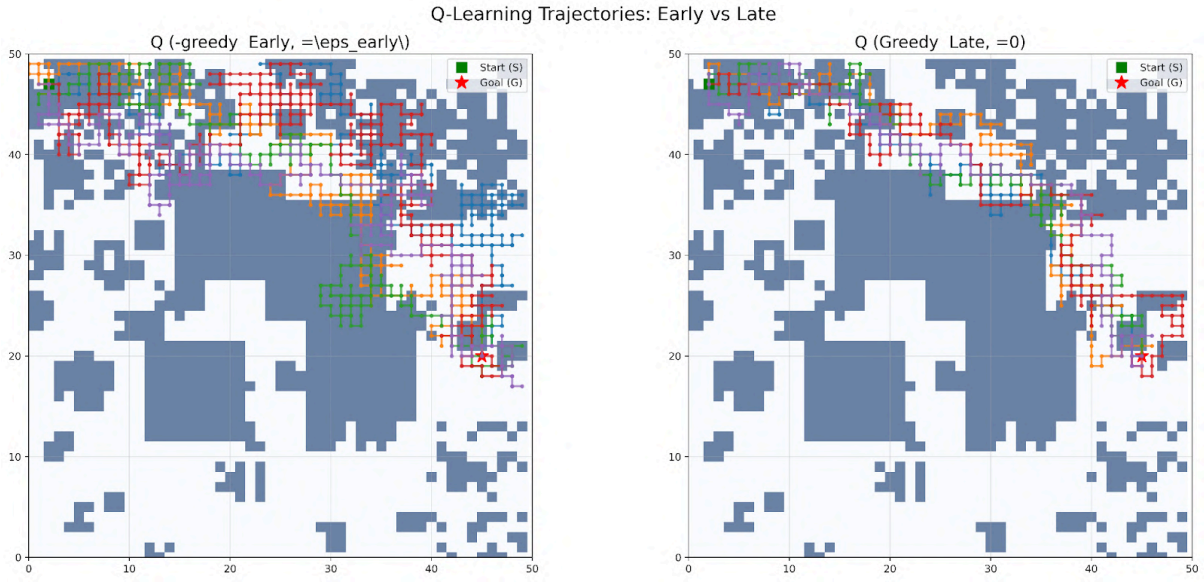


Figure 7: illustrates the navigation trajectories produced by the Q-learning agent during early and late stages of training under slip (stochastic) winter conditions.

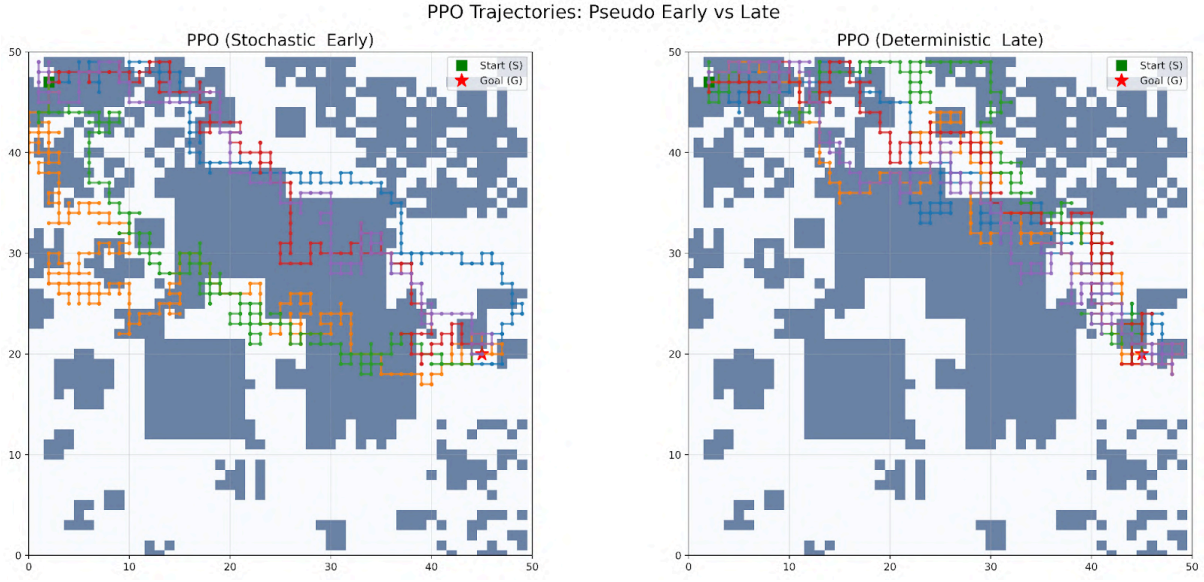


Figure 8: illustrates the navigation trajectories produced by the PPO agent during early and late stages of training under slip (stochastic) winter conditions.

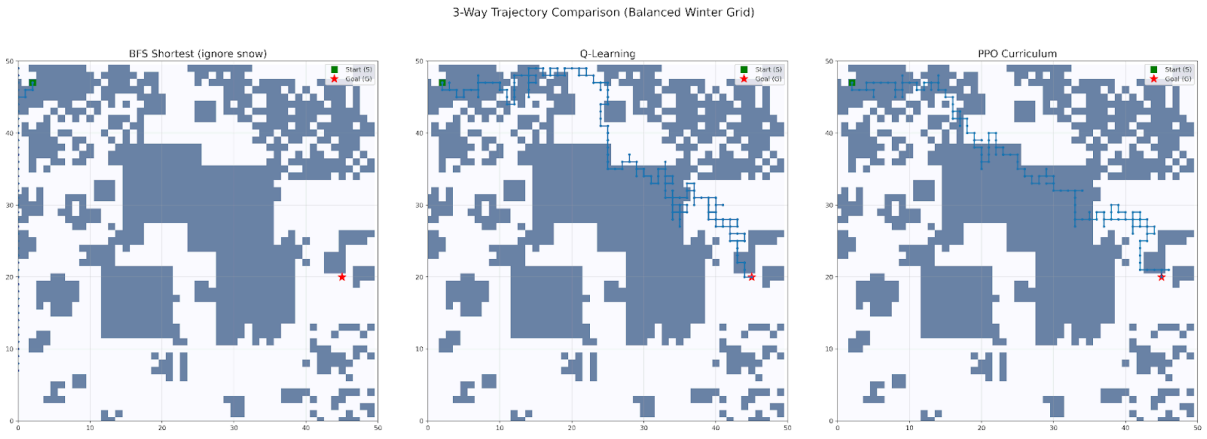


Figure 9: compares the navigation trajectories of BFS, Q-learning, and PPO under slip winter conditions.

=== Final Comparison Table ===

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	-1693.96±61.27	1000.00±0.00	246.80±76.09	0.0%
Q-Learning	304.03±9.92	228.18±4.10	67.58±4.34	100.0%
PPO Curriculum	252.19±7.04	242.29±3.00	92.65±4.14	100.0%

Table 2: summarizes the quantitative performance metrics for BFS, Q-learning, and PPO under slip winter conditions, using the same evaluation metrics as in the deterministic case, standard deviation, and averaged across 5 independent random seeds.

4.3 Spatial State Visitation Heatmaps

Spatial state visitation heatmaps offer a grid-based way to see how often an agent passes through each cell during navigation. In our 50×50 winter grid, each cell corresponds to a state, and the color intensity reflects visitation frequency across episodes. Darker areas indicate cells that are visited more frequently, while lighter or white cells indicate rarely or never visited locations. Such visualizations give a clear picture of the agent's overall path distribution and behavior patterns. We generated these heatmaps using aggregated visitation counts normalized to the [0,1] range.

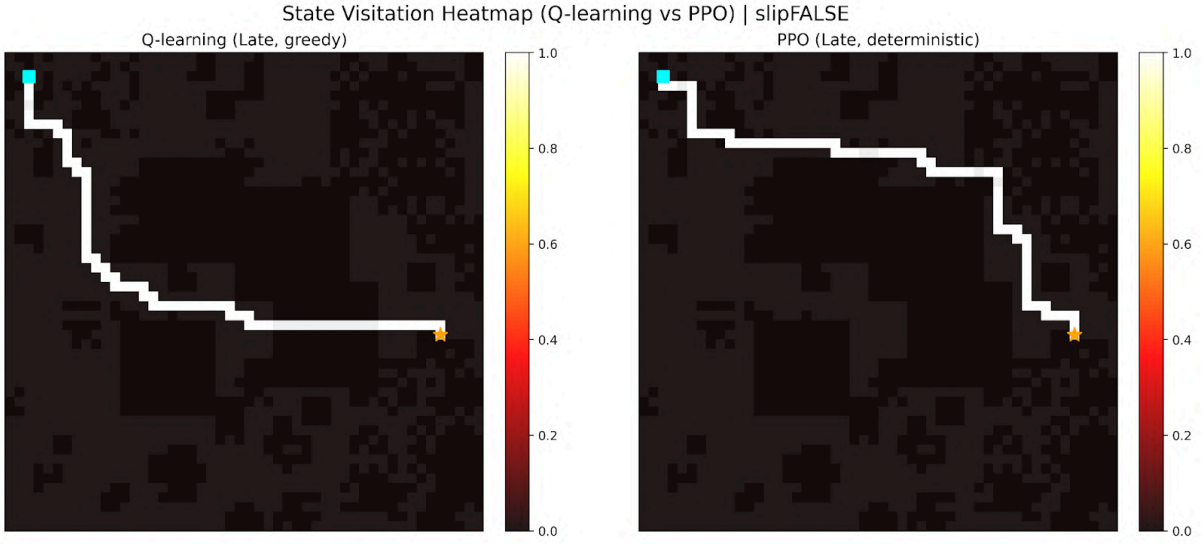


Figure 10: displays the visitation heatmaps for Q-learning and PPO under deterministic (non-slip) winter conditions. The results are aggregated from 10 evaluation runs consisting of 200 episodes per run. Both agents concentrate most visits along a narrow corridor connecting the start to the goal, with near-maximum intensity (≈1.0) along the primary route and very low visitation elsewhere.

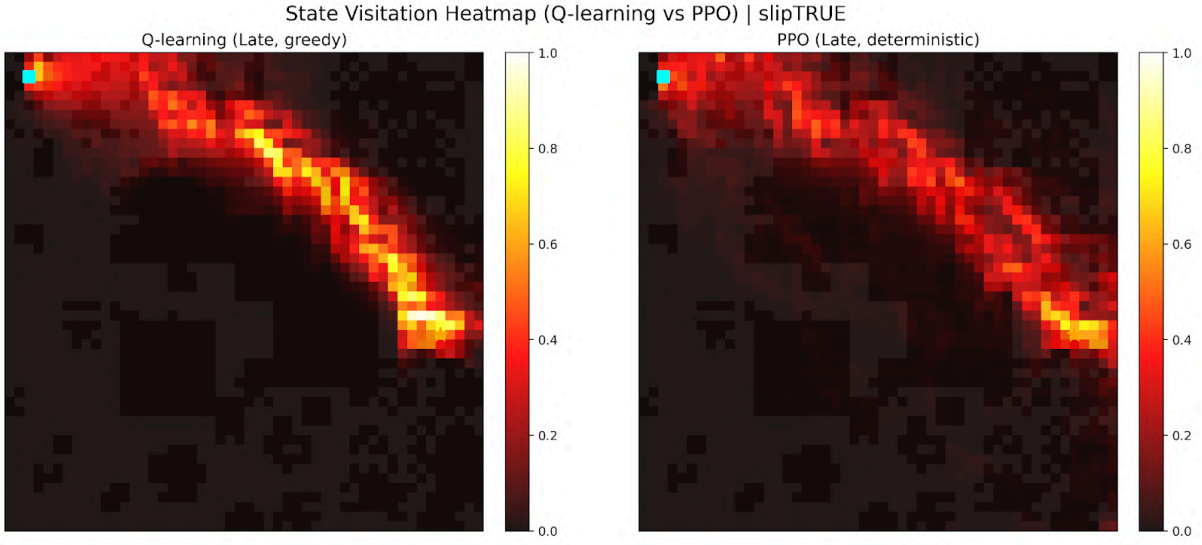


Figure 11: shows the visitation intensity for each grid cell in the slip case. Compared with the non-slip condition, visitation is more widely distributed across the grid rather than remaining within a narrow corridor. The heatmap aggregates data from 10 evaluation runs consisting of 200 episodes per run.

4.4 Limitations & Uncertainty

Several sources of uncertainty and limitations appeared in this study. Both Q-learning and PPO showed noticeable performance differences across training runs, which was

expected given the stochastic nature of the environment and the learning algorithms. To account for this variability, all experiments were repeated over 5 independent random seeds, and results are reported as the mean across runs
Under slip conditions, the same policy could produce quite different routes from episode to episode because actions did not always lead to the expected outcome. In addition, stochastic exploration strategies and random initialization of value functions (or policy networks) exposed agents to slightly different states, transitions, and rewards across runs. This variability made it harder to get perfectly consistent results.
An unexpected issue was that the BFS baseline failed to reach the goal within the 1,000-step limit under slip conditions, resulting in a number of unsuccessful evaluation episodes.

A full numerical breakdown of evaluation metrics is provided in Appendix B.

5.Discussion

5.1 Restatement of Hypothesis and Summary of Findings

The study hypothesized that reinforcement learning-based agents, particularly Proximal Policy Optimization (PPO), would achieve more energy-efficient winter navigation paths than a traditional shortest-path baseline and a value-based Q-learning agent under both deterministic and slip winter conditions. The averaged results partially supported the hypothesis. Under deterministic (non-slip) conditions, PPO achieved higher cumulative reward, and fewer snow visits, outperforming Q-learning, which supports the hypothesis. However under stochastic (slip) conditions, Q-learning outperformed PPO in terms of cumulative reward or snow-cell avoidance, which did not support the hypothesis. Both reinforcement learning methods perform better results than traditional BFS across these environments.

5.2 Interpretation of Q-Learning and PPO Performance

The results showed that algorithm performance varied depending on the environmental condition. Under deterministic (non-slip) conditions, PPO achieved higher cumulative reward and fewer snow cell visits than Q-learning, suggesting that PPO's policy gradient approach was able to learn a more energy-efficient route in a stable environment. Under stochastic (slip) conditions, Q-learning outperformed PPO in terms of cumulative reward and snow-cell avoidance. One possible explanation was that a discrete and clear grid-world environment and relatively small environment would favor more for Q-learning, therefore allowing Q-learning to efficiently estimate optimal state [7]. In contrast, PPO relied on function approximation and stochastic policy updates, which may have required more training data or longer training time to converge to an equally optimal policy under slip conditions. Additionally, its stochastic policy may have introduced suboptimal choices that were not effective, therefore reducing its total rewards relative to Q-learning under stochastic conditions [3].

5.3 Effects of Slip (Stochastic) Winter Conditions

Under slip (stochastic) conditions, all agents showed broader trajectory dispersion and increased visits to previously visited states because of slipping, as reflected in the spatial state visitation heatmaps. This behavior is consistent with probabilistic transition dynamics, in which the same action could lead to different movement outcomes across episodes, such as deviating left in one run and right in another. When the grid is slippery, agents don't follow one stable path anymore. As a result, the routes spread out, and their performance becomes less consistent across episodes. These observations are consistent with prior navigation studies showing that stochastic environments increase policy variance and reduce convergence stability in reinforcement learning tasks [8].

5.4 Interpretation of BFS Baseline Behavior

The Traditional baseline trajectories did not take account for winter grid costs or stochastic transitions, it focused on the fastest way to the destination. In non-slip conditions this approach naturally produced the optimal route. However, under slip conditions, BFS exceeded the maximum step limit in multiple evaluation episodes, reflecting its inability to adapt its policy in response to transition winter grid costs or stochastic transitions.

Because BFS always assumed deterministic movement, each slip caused deviations from the planned path. These deviations often led to inefficient loops and longer paths. Unlike reinforcement learning methods, BFS did not have reward feedback or policy updates, which limited its ability to adjust navigation behavior under changing environmental dynamics. Therefore, making it the least effective method.

5.5 Limitations and Future Directions

Several limitations should be noted when interpreting these findings. All experiments were carried out in a simulated 50×50 grid environment with a relatively small and simple state space, which differs significantly from real-world scenarios. This setup likely favored value-based methods like Q-learning, since it could learn precise state-action values for every cell. In contrast, PPO depends on neural network function approximation and stochastic policy updates, which may need more training steps or data to perform well in such discrete, small-scale settings.

To improve statistical reliability, all experiments were run independently over 5 random seeds and results were averaged across runs. Noticeable performance variability was observed across training runs, mainly due to the stochastic environment, random action outcomes, and probabilistic transitions under slip conditions.

The simulated environment simplified real-world winter driving conditions and did not capture vehicle dynamics, sensor noise, road-surface variation, or realistic

energy-consumption models, these are factors that might lead to changes. As a result, these findings to real-world winter navigation scenarios were limited [9].

Future research could evaluate reinforcement learning agents in larger scale, continuous, or more realistic winter navigation environments with higher-dimensional state spaces and more complex conditions. Then it may be better to reflect real-world uncertainty and could allow policy-gradient methods such as PPO to fully function their theoretical advantages in handling stochastic and high-dimensional control problems.

---

6. Conclusion

This study compared a traditional BFS baseline with two reinforcement learning methods—Q-learning and Proximal Policy Optimization (PPO)—for energy-efficient navigation in a 50×50 winter grid under both deterministic (non-slip) and stochastic (slip) conditions. The averaged results across 5 independent random seeds partially supported the original hypothesis. Under deterministic (non-slip) conditions, PPO achieved the highest cumulative reward and fewest snow cell visits, outperforming Q-learning. However, under stochastic (slip) conditions, Q-learning outperformed PPO in cumulative reward and snow-cell avoidance. Although both reinforcement learning methods clearly outperformed the BFS baseline, BFS struggled under slip conditions due to its inability to adapt to stochastic transitions.

These findings show that whether value-based methods or policy-gradient methods perform better really depends on how unpredictable the environment is. In our small, structured discrete grid, PPO gained an advantage from its stable gradients and curriculum learning when everything was deterministic. On the other hand, Q-learning's simple tabular updates turned out to be more robust when the roads became slippery and actions sometimes failed. More generally, the results emphasize that the choice of reinforcement learning method should be guided by the structure and constraints of the task, rather than by theoretical advantages alone. Future work could explore whether PPO's advantages extend to more complex environments with continuous states or larger state spaces, where its policy-gradient approach may better demonstrate its theoretical strengths.

Reference

[1]Mao, R., Xu, W., Qian, Y., Li, X., Li, Y., Li, G., & Zhang, H. (2025). Understanding the Determinants of Electric Vehicle Range: A Multi-Dimensional Survey. *Sustainability*, 17(10), 4259. https://doi.org/

10.3390/su17104259
[2]Carlson, A., & Vieira, T. (2021). *The effect of water and snow on the road surface on rolling resistance* (VTI Report 971A). Swedish National Road and Transport Research Institute. https://www.diva-portal.org/smash/get/diva2:1542142/FULLTEXT01.pdf
[3]Watkins, C.J.C.H., & Dayan, P(1992). Q-learning. *Mach Learn* 8, 279–292. https://doi.org/10.1007/BF00992698
[4]Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). *Proximal Policy Optimization Algorithms* (No. arXiv:1707.06347). arXiv. https://doi.org/10.48550/arXiv.1707.06347
[5]Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction. https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf
[6]Dayan, P., & Balleine, B. W. (2002). Reward, Motivation, and Reinforcement Learning. *Neuron*, 36(2), 285–298. https://doi.org/10.1016/S0896-6273(02)00963-7
[7]Tan, C. (2025). Comparative Study of Reinforcement Learning Performance Based on PPO and DQN Algorithms. *Applied and Computational Engineering*, 175(1), 30–36. https://doi.org/10.54254/2755-2721/2025.AST24879
[8]Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A. J., Banino, A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., Kumaran, D., & Hadsell, R. (2017). *Learning to Navigate in Complex Environments* (No. arXiv:1611.03673). arXiv. https://doi.org/10.48550/arXiv.1611.03673
[9]Chukwurah, N., Adebayo, A. S., Ajayi, O. O., & Anfo Pub. (2024). *Sim-to-Real Transfer in Robotics: Addressing the Gap between Simulation and Real- World Performance*. *International Journal for Multidisciplinary Research (IJFMR)*, 05(01), 33–39. https://doi.org/10.54660/.IJFMR.2024.5.1.33-39
[10]Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). Curriculum learning. *Proceedings of the 26th Annual International Conference on Machine Learning*, 41–48. https://doi.org/10.1145/1553374.1553380

Appendices

Appendix A Experimental Configuration and Algorithmic Details:

A.1 Environment Configuration

All experiments were conducted in a custom 50×50 winter grid environment (2,500 discrete states).

- Start position: (47, 2)
- Goal position: (20, 45)
- Maximum steps per episode: 1000
- Action space: {up, down, left, right}

Two transition settings were evaluated:

Deterministic (slip = FALSE)

The intended action is executed exactly.

Stochastic (slip = TRUE)

With probability $\frac{1}{3}$, the intended action is executed.
With probability $\frac{2}{3}$, the agent slips to a random adjacent direction.

All algorithms were evaluated on the same fixed grid layout to ensure fairness.

A.2 Reward Function

The reward function models energy-aware winter navigation.

At each time step:

- Step penalty: -1.5
- Snow penalties:
  - Near snow: -0.2
  - Edge snow: -0.5
  - Core snow: -2.0
- Goal reward: +700

The cumulative episode reward is:

$$R = \sum_{t=1}^{T} (-1.5 - C_{snow}(s_t)) + 700 \cdot 1_{goal\ reached}$$

where $C_{snow}(s_t)$ denotes the terrain penalty and $1_{goal\ reached}$ indicates successful termination.

A.3 Q-Learning Configuration

Tabular Q-learning was implemented with the following hyperparameters:

- Learning rate α=0.15
- Discount factor γ=0.99
- Exploration strategy: ε-greedy
- Initial ε = 0.60
- Minimum ε = 0.05
- Exponential decay per episode = 0.9995
- Training episodes = 15,000
- Max steps per episode = 1000

The Q-update rule is:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[R + \gamma \max_{a'} Q(s',a') - Q(s,a)]$$

During evaluation, a fully greedy policy (ε = 0) was used.

A.4 Proximal Policy Optimization (PPO) Curriculum Configuration

PPO was implemented as a stochastic policy-gradient method.

Total training timesteps: 1,200,000

- Phase 1 (navigation-focused reward): 300,000 timesteps
- Phase 2 (energy-aware reward): 900,000 timesteps

Evaluation was conducted using deterministic action selection.

PPO Objective Function

PPO optimizes the clipped surrogate objective:

$$L^{CLIP}(\theta) = E_t[min(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1 - \epsilon, 1 + \epsilon)\hat{A}_t)]$$

where

$$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$$

and A^t is the generalized advantage estimate.

The clipping mechanism constrains policy updates to maintain stability.

PPO Hyperparameters

- Discount factor γ=0.99
- GAE parameter λ=0.95
- Entropy regularization enabled
- Learning rate: 3×10^{-4}
- Clip range: 0.2
- Rollout buffer: 2,048 steps
- Batch size: 256
- Optimization epochs: 10
- Neural network architecture: two hidden layers (256 units each, ReLU activation)

A.5 Evaluation Protocol

For each condition (slip FALSE / TRUE):

- One trained model per algorithm
- Maximum evaluation length: 1000 steps
- Number of evaluation episodes: 200
- Number of seeds: 5
- Heatmap runs: 10
- Metrics recorded:
    - Total cumulative reward
    - Number of steps
    - Snow cell visits
    - Success rate (%)

State visitation heatmaps were generated by aggregating visitation frequencies across evaluation runs of 200 episodes each.

Appendix B Detailed Quantitative Results:

B.1 Deterministic (slip = FALSE)

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	575.30±0.00	70.00±0.00	31.00±0.00	100.0%
Q-Learning	579.10±5.61	70.00±0.00	12.60±3.29	100.0%
PPO Curriculum	588.50±0.60	70.00±0.00	7.80±1.10	100.0%

All methods reached the goal within 70 steps.
PPO achieved the highest cumulative reward and fewest snow cell visits.

B.2 Stochastic (slip = TRUE)

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	-1693.96±61.27	1000±0.00	246.80±76.09	0%
Q-Learning	304.03±9.92	228.18±4.10	67.58±4.34	100.0%
PPO Curriculum	252.19±7.04	242.29±3.00	92.65±4.14	100.0%

Under stochastic dynamics:

- BFS fails due to inability to adapt.
- Q-learning demonstrates greater robustness.
- PPO maintains success but exhibits higher trajectory dispersion.

B.3 State Visitation Analysis

State visitation frequency was computed as:

$$V(s) = \frac{N(s)}{max_s N(s)}$$

where $N(s)$ is the number of visits to state s.

- Under deterministic conditions, visitation concentrates along a narrow corridor.
- Under stochastic conditions, visitation becomes more dispersed.
- Q-learning exhibits more focused routing than PPO under slip conditions.

Energy-Efficient Path Planning in Winter Conditions: A Comparative Study of Traditional Baseline, Q-learning, and Proximal Policy Optimization in a Grid World Environment

1.Abstract

Recent investigation in winter route planning has become increasingly important due to increased resistance and reduced efficiency on winter roads. These conditions significantly increase energy consumption and introduce uncertainty, raising the risk of failing to reach the destination. To address the challenge, Rreinforcement Llearning (RL) is a type of machine learning where an agent learns to make decisions by interacting with an environment, receiving rewards or penalties for its actions to maximize cumulative rewards. For simulation, we benchmark a standard shortest-path algorithm (BFS) against two reinforcement learning methods: Q-learning and Pproximal Ppolicy oOptimization (PPO). All approaches are tested on identical grid configurations, with the same starting points, goals, and reward functions. We assess their performance based on cumulative reward, snowy cell visits (as a proxy for energy cost), trajectory characteristics, and success rate. It is hypothesized that proximal policy optimization (PPO) algorithms will result in lower energy consumption than Q-learning and traditional baseline under winter conditions. The results show that while BFS consistently finds the shortest paths, it fails to consider energy costs or environmental uncertainty. RL agents, by comparison, adapt more efficiently to winter conditions. Under deterministic (non-slip) conditions, PPO achieved higher cumulative reward and fewer snow cell visits than Q-learning, partially supporting the hypothesis. However, under stochastic (slip) conditions, Q-learning outperformed PPO in cumulative reward and snow-cell avoidance. In contrast, Q-learning has an overall better performance in both conditions than PPO. These results suggest that Q-learning is Q-learning is better suited for stochastic conditionsconditions structured winter navigation in grid worlds, while PPO may perform better in more deterministic conditions withconditionscomplex environments with continuous states or decisions.

Key words: Energy-Efficient, Traditional Baseline, Q-learning, Proximal Policy Optimization, grid world, reinforcement learning.

2.Introduction

As demand in electric vehicles become increasingly widespread, energy efficiency and route optimization have emerged as critical challenges, particularly in winter when resistance increases. Freezing temps, plus snow and ice covering the roads, push energy consumption way up, which means shorter range and trips that feel less predictable. A 2025 study shows that an estimated 50 % of EV driving range can be reduced in cold

climates, including snow and ice covering terrain, highlighting the significant impact of environmental conditions on energy consumption [1]. Furthermore, when snow or water remains on the road surface, vehicle tires must continuously displace through ice as they roll and move, hence forcing the vehicle to draw more power [2]. On top of that, water cools tires more effectively than air alone, altering their mechanical properties and further pushing rolling resistance higher. Previous studies have shown that rolling resistance can rise by approximately 30% to 40% under wet or snowy conditions [2]. Ultimately, these factors make winter driving a major concern for Electric Vehicle (EV) efficiency and reinforce the need for energy-aware routing strategies.

To approach these challenges, we have explored a few computational methods to allow agents to learn effective navigation strategies aimed at minimizing energy consumption. In this study, a 50x50 grid is used as an abstraction routing map that circles key structures of Canadian snow distribution, rather than exact geographical terrain. Grid-based environments are commonly used in reinforcement learning studies, as they simplify complex continuous real-world environments into discrete states [5]. ~~This abstraction supports controlled comparison of different routing methods while preserving the essential energy-related difficulties.~~ This abstraction is particularly well-suited for this study, as it enables environmental factors such as snow coverage and surface slip conditions to be incorporated directly into the cell-level cost and reward structure, supporting controlled and reproducible comparison of different routing methods.

Despite growing interest in reinforcement learning for navigation, existing studies have largely overlooked the impact of winter environmental conditions, such as snow coverage and surface slippiness, on energy-aware path planning. Prior work has primarily focused on general navigation without taking account of explicitly modeling weather-dependent energy costs [8]. To address this gap, this study integrates snow coverage, stochastic slip condition, and energy costs into a comparative framework, evaluating Q-learning, PPO, and a traditional Baseline under realistic winter routing scenarios. This study is one of the first to directly compare value-based and policy-gradient reinforcement learning methods in an energy-aware winter navigation setting.

More specifically, the research will include a traditional baseline algorithm, and two types of reinforcement learning, which are Q-learning and PPO ~~Proximal Policy optimization~~ algorithm. The traditional baseline routing computes the shortest path based on static cost metrics and follows the predefined route without adaptation or learning (basic routing). Q-learning, a value-based reinforcement learning method, learns through incremental dynamic programming processes with computational requirements and it is well suited for agents to improve and refine action values and achieve effective performance in controlled Markovian domains [3]. In contrast, PPO is a policy-gradient algorithm that primarily optimizes a stochastic policy and achieves

greater training stability through constrained policy updates with a clipped surrogate objective function [4].

This comparison evaluates the effectiveness of the proposed routing methods in identifying energy-efficient paths. Performance is evaluated through energy related costs, overall routing efficiency, and observed navigation behavior under both deterministic (non-slip) and stochastic (slip) settings. Because PPO balances stable policy updates with adaptive learning in uncertain environments, it is hypothesized to be the most effective method implemented for winter routing.

The following section goes into the methodology and experimental setup in more detail.

3.Methods

3.1 Environment Design

A custom winter navigation grid environment was designed to evaluate each Rreinforcement Learning (RL) agent under stochastic and cost-sensitive conditions. The environment consists of a 50 x 50 grid world containing 2500 cells, including both normal terrain and snow-covered terrain. ~~Specifically, two transition settings were evaluated: a slip snow condition and non-slip snow condition (same grid map).~~ The simulation contains a fixed start (47, 2) and a fixed goal (20, 45) point, and are located in the top left corner and middle right respectively. At each time step, the agent can take either one of the four directions, up, down, left, and right; each step on a non-snow grid has an energy cost~~ause~~ of -1.5 per step. Snow is modeled as spatially correlated regions with graded intensity, capturing the typical uneven accumulation patterns in Canadian winters.

When snow is modeled as a binary state, abrupt reward discontinuities make PPO's advantage estimates unstable, resulting in increased gradient variance. By contrast, a continuous representation of snow thickness provides smoother cost transitions, which help reduce gradient variance and stabilize learning, making it more efficient. These snow types are classified as "Near snow", "Edge snow", and "Core snow".

To evaluate robustness under different winter conditions, two transition settings were considered using the same grid map: a non-slip (deterministic) condition and a slip (stochastic) condition. In the deterministic setting, action always results in the intended movement, and snow cells only add additional energy loss. In the stochastic setting, actions do not always lead to the intended outcome: with a probability of 1/3, the intended action is executed, and with a probability of 2/3, the agent moves in a random adjacent direction, simulating loss of traction control during winter driving on ice and snow.~~intended actions may be replaced with unintended movements due to unstable control on icy surfaces, with a higher probability of deviating left or right, simulating loss control of traction during winter driving on ice and snow.~~

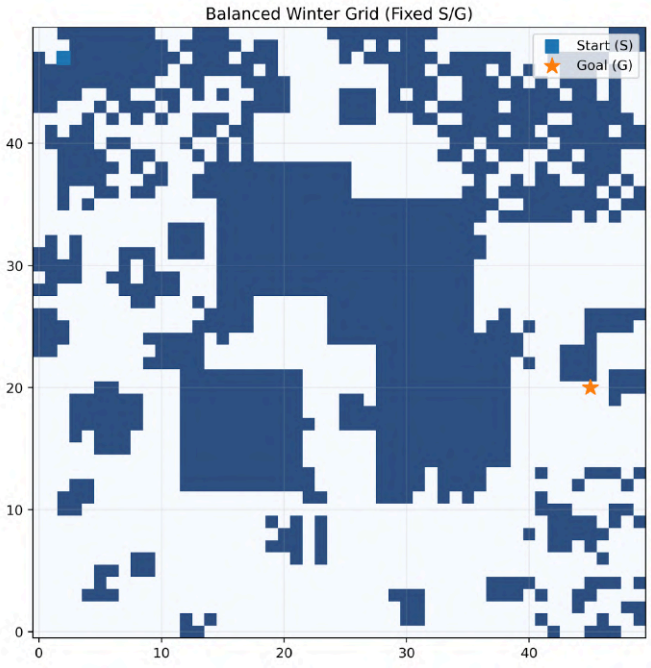


Figure 1: shows a typical winter grid layout used in all experiments.

3.2 Reinforcement Learning Algorithms/ Methodologies:

The routing strategies and learning methods used in this study are described in this section.

3.2.1 Traditional Baseline (BFS)

Traditional bBaseline is a non-learning shortest path strategy used as a comparison base for reinforcement learning methods. It operates on the 50x50 grid map, it considers four directions to move (up, down, left, right). The system calculates the shortest path to the destination, without considering energy savings, which means it does not adapt to environmental feedback or uncertainty.

3.2.2 Q-learning Implementation

Tabular Q-learning was used as the value-based reinforcement learning baseline. The agent maintains a discrete Q (s, a) table, which is updated iteratively based on observed state transitions. An ε-greedy policy was used for exploration. The value of ε begins at 0.6 and decays exponentially with a factor of 0.9995 per episode until it reaches 0.05. In this 50×50 grid setting, the decay schedule allows broad initial exploration before shifting emphasis to exploitation. Actions are restricted to four possibilities—left (0), down (1), right (2), and up (3)—consistent with standard discrete grid-world formulations and the discrete grid structure. The Q-value update follows the classic temporal-difference form:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[R + \boldsymbol{\gamma}\, max_{a'}\, Q(s',a') - Q(s,a)] \quad [5]$$

We fix the learning rate α at 0.15 and the discount factor γ at 0.99. These values handle the stochastic transitions (is_slippery=True) and support planning over long horizons in the large grid [3]. State s denotes the flattened grid position index. The action a selects one of the four movement directions. Q(s, a) represents the estimated discounted cumulative reward starting from state s and action a [3,5].

3.2.3 Reward system

In this research, the reward system is designed to represent energy efficiency under winter conditions, it helps agents find the best paths. At each step, the system gives a negative value to show how significant a step is to energy saving, with higher cause on snow grids. A positive 700 is given only when the agents successfully reach the destination, while within the maximum steps of 1000 steps. It encourages agents to reach the destination with considerations on energy saving. We initialize the cumulative reward to 0 at the beginning of each episode. Each will result in negative values, because by doing this, agents don't need to consider the prior bias, since it assumes nothing at first, etc. As in the settings, we have Near-snow (light snow), Edge-snow (medium snow), and Core-snow (deep snow); each of them has different additional values which are -0.2, -0.5, and -2.0 respectively [6].

$$r_t = r_{step} - c_{energy}(s_t) + r_{goal} \quad [6]$$

The basic reward function follows standard reinforcement learning practice by combining step penalties, energy-related costs, and a terminal goal reward.

3.2.4 Proximal Policy Optimization

Proximal Policy Optimization (PPO) served as the policy-gradient method in this study for energy-efficient routing under winter conditions. PPO learns a stochastic policy by directly outputting action probabilities for each state, which helps the agent cope with uncertainty on slippery roads.

$$L^{CLIP}(\theta) = E_t[min(r_t(\theta)\widehat{A}_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)\widehat{A}_t)] \quad [4]$$

where $r_t(\theta)$ is the probability ratio between the new and previous policies. The clipping mechanism constrains policy updates to improve stability.

The PPO agent was trained using fixed values across all experiments. The discount factor was set to γ=0.99, and Generalized Advantage Estimation (GAE) was used with λ=0.95. The policy network consisted of two fully connected hidden layers with 256~~512~~ units each, using ReLU activation functions. ~~Entropy regularization was applied to~~

~~encourage exploration during training.~~ The learning rate was set to 3×10^{-4}, with a clipping range of 0.2 and an entropy coefficient of 0.02 to encourage exploration. PPO training was conducted for 1,200,000 time steps, using a rollout buffer of 2048 steps, a batch size of 256, and 10 optimization epochs per update~~using large rollout buffers and multiple optimization epochs~~ to improve training stability [4].

Policy-gradient methods work by directly adjusting a parameterized policy to increase expected reward. They learn a stochastic policy, allowing for more flexibility in uncertain environments, where the same action can lead to different outcomes. In contrast, q-learning estimates state-action values and usually converges with a deterministic policy based on these estimates. Although it can learn in unfamiliar environments, its action selection may be less reliable in highly unpredictable situations. As a result, policy-gradient and value-based methods show varying strengths depending on the level of uncertainty in the winter routing task.

3.2.5 Curriculum Learning for PPO

PPO training followed a two-phase curriculum learning approach designed to improve training stability and sample efficiency under sparse rewards and energy-sensitive penalties [10].

Phase 1: Navigation Learning Phase (300,000 timesteps)

~~Navigation Learning Phase~~ In phase 1, PPO was trained using a simplified reward function that focuses only on positive feedback for reaching the goal. Energy costs in snow grids were not included. This made the feedback denser and more immediate. The agent quickly learned basic navigation and achieved early success in the 50×50 winter grid. The policy learned in this phase served as the starting point for the next training stages.

Phase 2: Energy-Aware Optimization Phase (900,000 timesteps)

~~Energy-Aware Optimization Phase~~ In phase 2, the training continued with the complete energy-aware reward function, which now included penalties for moving across snow-covered areas. The agent had to balance successful arrival with lower energy consumption. All reported results, ablation studies, and comparisons are~~were~~ based solely on the Phase 2 reward function.

¶

¶

3.3 Evaluation factors

To ensure statistical reliability, all reinforcement learning experiments were repeated over 5 independent random seeds, and results are reported as the mean and standard deviation across runs.

To compare the performances of all routing methods under identical conditions, a final evaluation was conducted after training. Each method BFS, Q-learning, and PPO was evaluated using the following metrics:

- Total Reward: calculated as the cumulative reward obtained over the episode.
- Number of steps: steps taken from the start to the goal (less than 1000 steps).
- Average snow cell visited: Snow visits were calculated by counting the number of snow-covered cells traversed in each episode and averaging this value across all evaluation episodes.
- Success rate: defined as the percentage of episodes in which the agent reached the goal within the maximum step limit.

The BFS baseline was evaluated only once on the fixed grid map, as it produces a deterministic path. Reinforcement learning agents were evaluated over 200 ~~multiple~~ episodes per run, with experiments repeated over 5 independent random seeds for each condition (slip and non-slip). All methods were compared using the same ~~-~~energy-aware reward function and environment configuration to ensure fairness.

3.4 Hyperparameter Selections

Hyperparameters are selected based on standard practices and empirical validation to ensure stable and consistent performance [3,4].

3.4.1 Q-Learning Hyperparameter Configuration

We implement tabular Q-learning with an ε-greedy exploration strategy.

The hyperparameters are set as follows:
- Learning rate $\alpha = 0.15$
- Discount factor $\gamma = 0.99$
- Exploration strategy: ε-greedy
- Initial $\varepsilon = 0.60$
- Minimum $\varepsilon = 0.05$
- Exponential decay rate = 0.9995
- Training episodes = 15,000
- Maximum steps per episode = 1,000

The learning rate ($\alpha = 0.15$) is selected to balance convergence speed and stability. A moderate learning rate allows the agent to adapt efficiently while avoiding oscillations in value updates.

The discount factor ($\gamma = 0.99$) emphasizes long-term rewards, which is essential for navigation tasks where the objective is to reach the goal efficiently over multiple steps.

An ε-greedy exploration strategy is adopted to balance exploration and exploitation. The initial exploration rate ($\varepsilon = 0.60$) encourages sufficient exploration in early training, while exponential decay gradually shifts the policy toward exploitation. The minimum ($\varepsilon = 0.05$) ensures that some level of exploration is maintained throughout training, preventing the agent from getting stuck in suboptimal policies.

The number of training episodes 15,000 and maximum steps per episode 1,000 are chosen to ensure sufficient interaction with the environment for convergence, while maintaining computational efficiency.

During evaluation, a fully greedy policy ($\varepsilon = 0$) is used to assess the learned policy performance.

3.4.2 PPO Hyperparameter Configuration

We implement PPO using a clipped surrogate objective with generalized advantage estimation (GAE).

The hyperparameters are set as follows:

- Discount factor $\gamma = 0.99$
- GAE parameter $\lambda = 0.95$
- Learning rate $= 3 \times 10^{-4}$
- Clipping range $\varepsilon = 0.2$
- Rollout buffer size = 2048 steps
- Batch size = 256
- Optimization epochs per update = 10
- Entropy regularization is enabled

The policy and value networks share a neural architecture consisting of two hidden layers with 256 units each and ReLU activation.

The discount factor ($\gamma = 0.99$) is selected to emphasize long-term rewards, which is essential for navigation tasks. The GAE parameter ($\lambda = 0.95$) provides a balance between bias and variance in advantage estimation, leading to more stable learning.

The clipping parameter ($\varepsilon = 0.2$) constrains policy updates to prevent large deviations from the previous policy, which improves training stability. The rollout buffer size and number of optimization epochs are chosen to ensure sufficient policy updates while avoiding overfitting to recent trajectories.

Entropy regularization is included to encourage exploration and prevent premature convergence to suboptimal policies.

Overall, these hyperparameters follow widely adopted settings in reinforcement learning literature and are chosen to ensure stable and consistent training rather than aggressive performance tuning.

A complete summary of environment settings and hyperparameters is provided in Appendix A.¶

¶

---

4. Results

The experiments compare BFS, Q-learning, and PPO on the same 50×50 winter grid, looking at both non-slip (deterministic) and slip (stochastic) cases.

4.1 Evaluation Setup and Metrics

The grid layout, start and goal positions, and reward parameters were kept exactly the same across all methods. All experiments were repeated over 5 independent random seeds, and results are reported as the mean across runs. Trajectory figures shown are from the representative seed whose performance was closest to the mean. For each run, total reward, number of steps, average snow cells crossed per episode, and success in reaching the goal were recorded.

4.2 Trajectory Visualizations and Quantitative Results¶

4.2.1 Deterministic (Non-Slip) Winter Condition

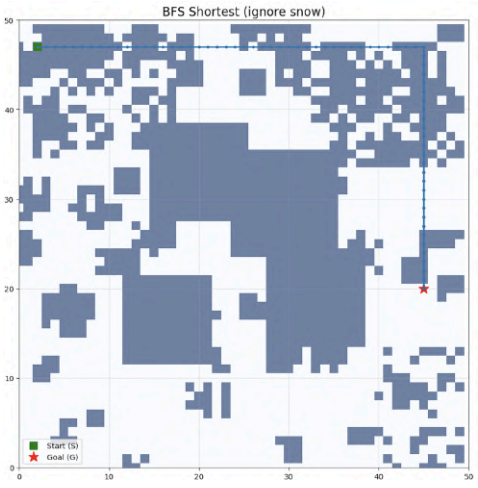


Figure 2: illustrates the BFS path under non-slip conditions. Since BFS always finds the shortest route, the trajectory goes straight from the start to the goal with minimal cells visited without any deviation.

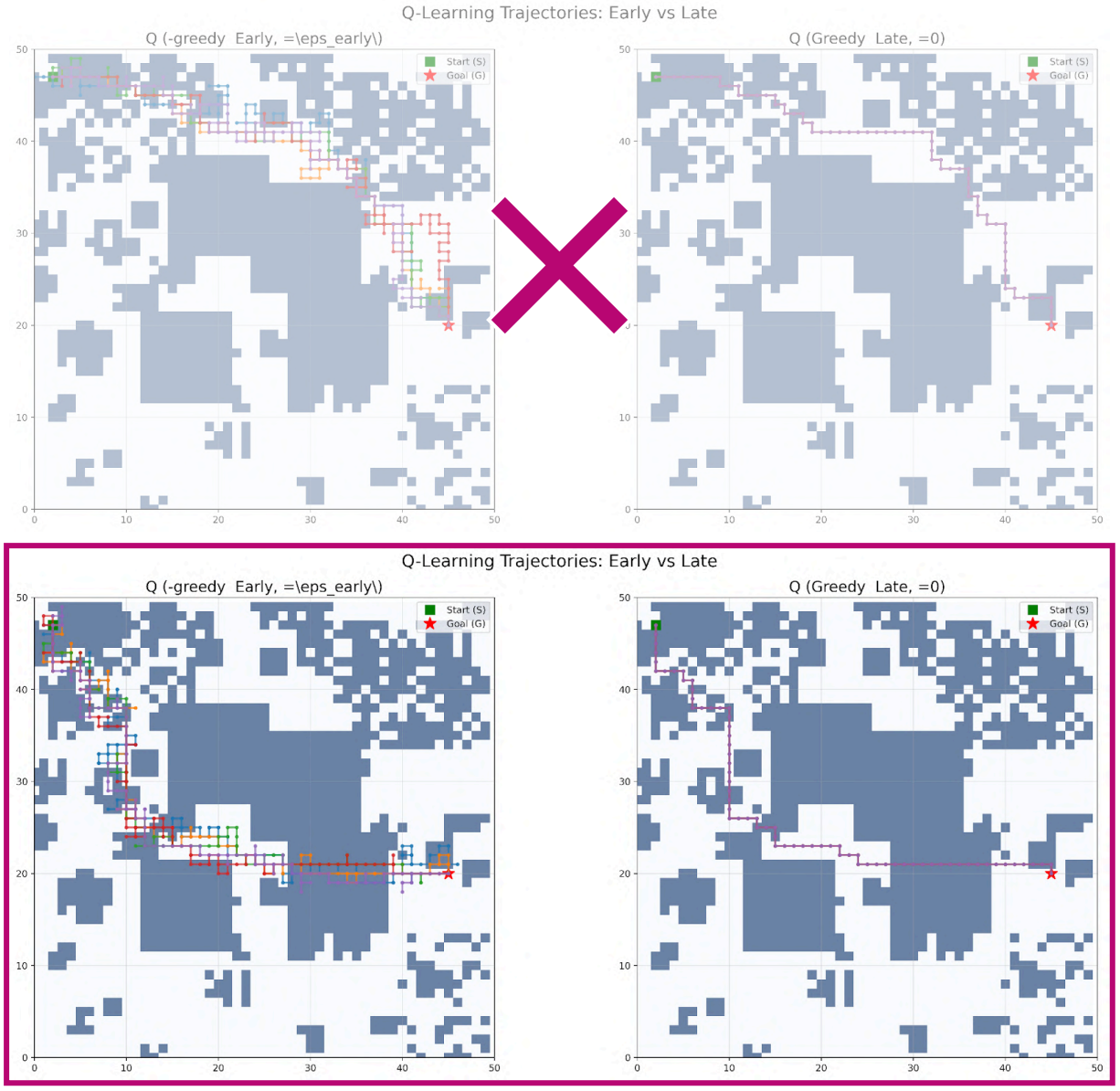


Figure 3: illustrates the navigation trajectories produced by the Q-learning agent under deterministic (non-slip) winter conditions during early training with an ε-greedy policy and late training with a greedy policy. The trajectories demonstrate the transition from exploratory behavior to a stable path toward the goal.

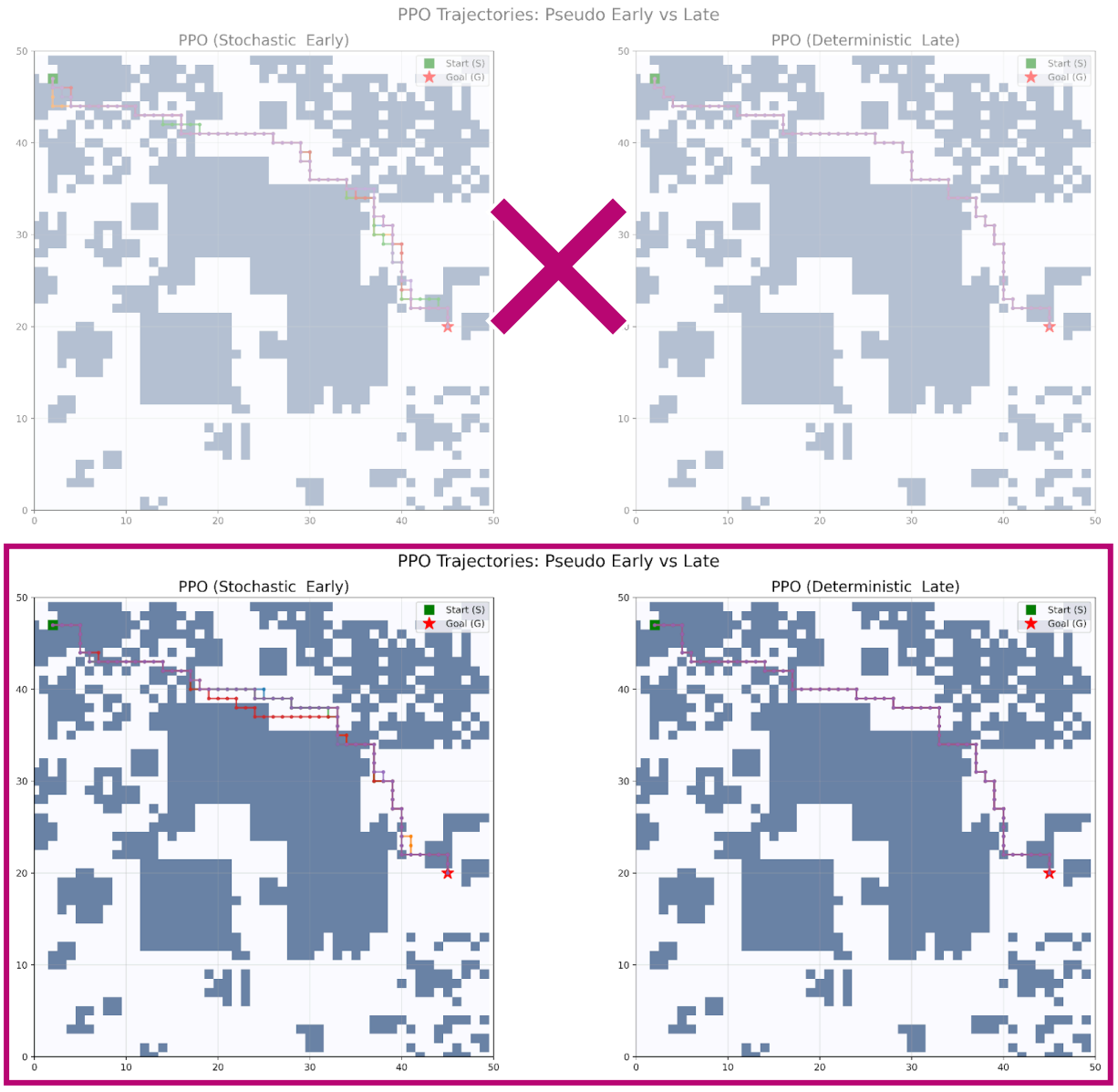
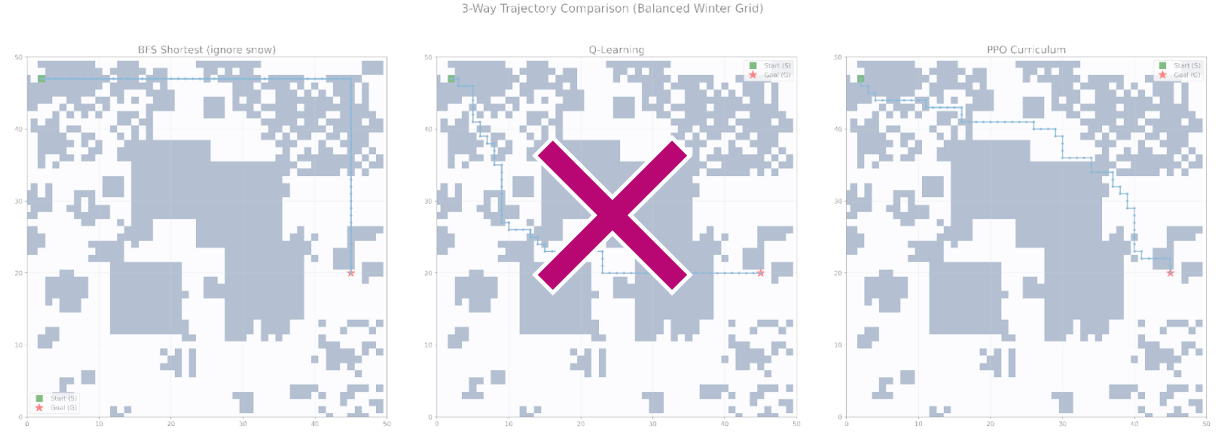


Figure 4: illustrates the navigation trajectories produced by the PPO agent during early and late stages of training under deterministic (non-slip) winter conditions.



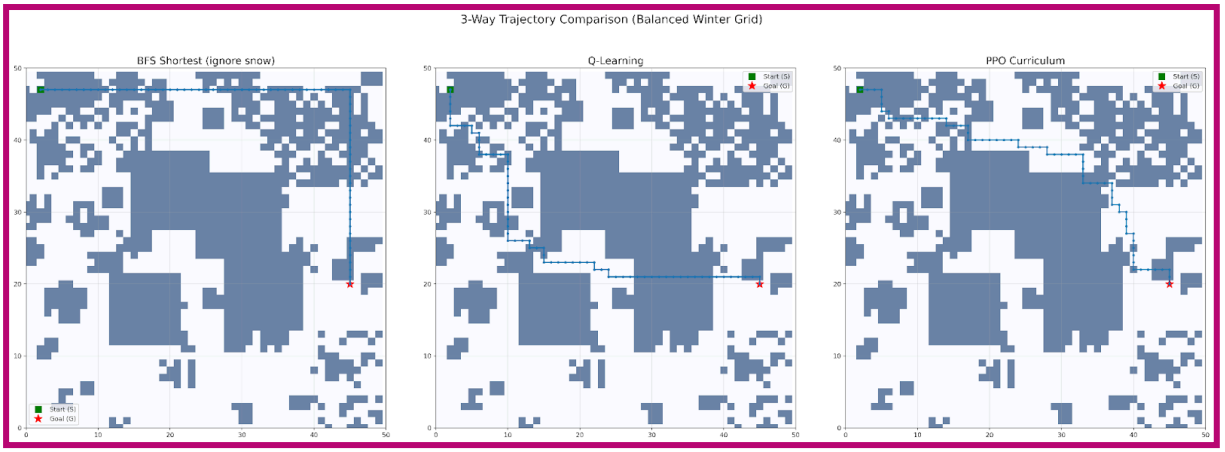


Figure 5: compares the trajectories of BFS, Q-learning and PPO in the non-slip winter setting.

=== Final Comparison Table ===

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	575.30±0.00	70.00±0.00	31.00±0.00	100.0%
Q-Learning	**579.10**~~589.50~~±5.61	70.00±0.00	**12.60**~~7.00~~±3.29	100.0%
PPO Curriculum	588.50~~587.50~~±0.60	70.00±0.00	**7.80**~~11.00~~±1.10	100.0%

Table 1: summarizes the main performance metrics for all three methods under deterministic conditions, including total reward, steps to goal, average snow cells crossed, ~~and~~ success rate, standard deviation, and averaged across 5 independent random seeds.

4.2.2 Slip Winter Condition

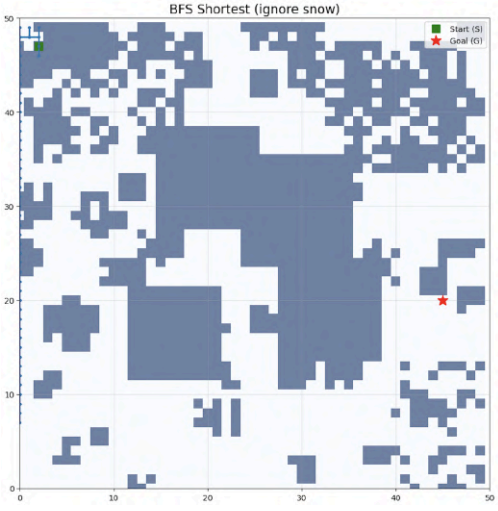


Figure 6: illustrates the navigation path produced by the BFS baseline under slip (stochastic) winter conditions.

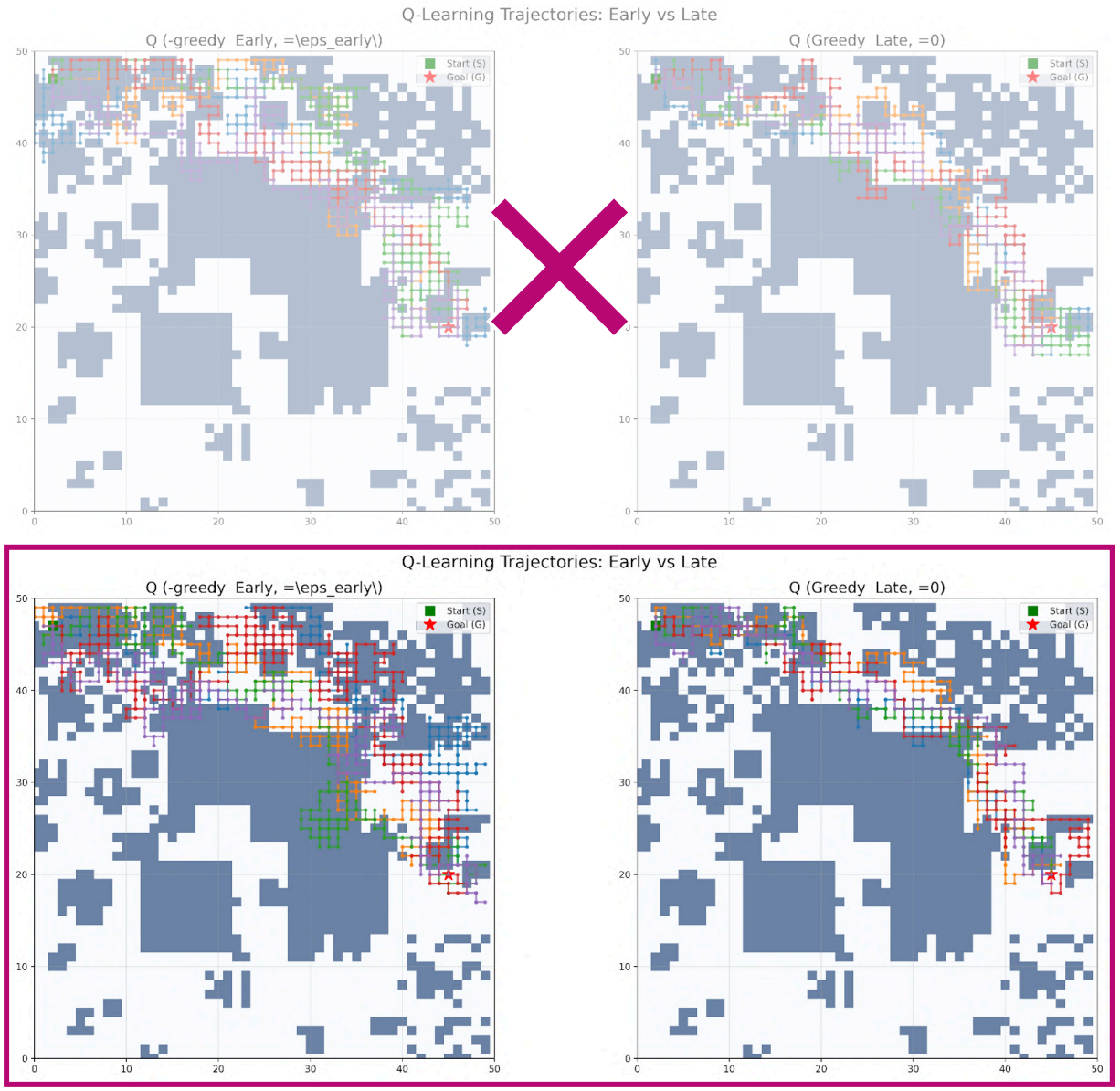


Figure 7: illustrates the navigation trajectories produced by the Q-learning agent during early and late stages of training under slip (stochastic) winter conditions.

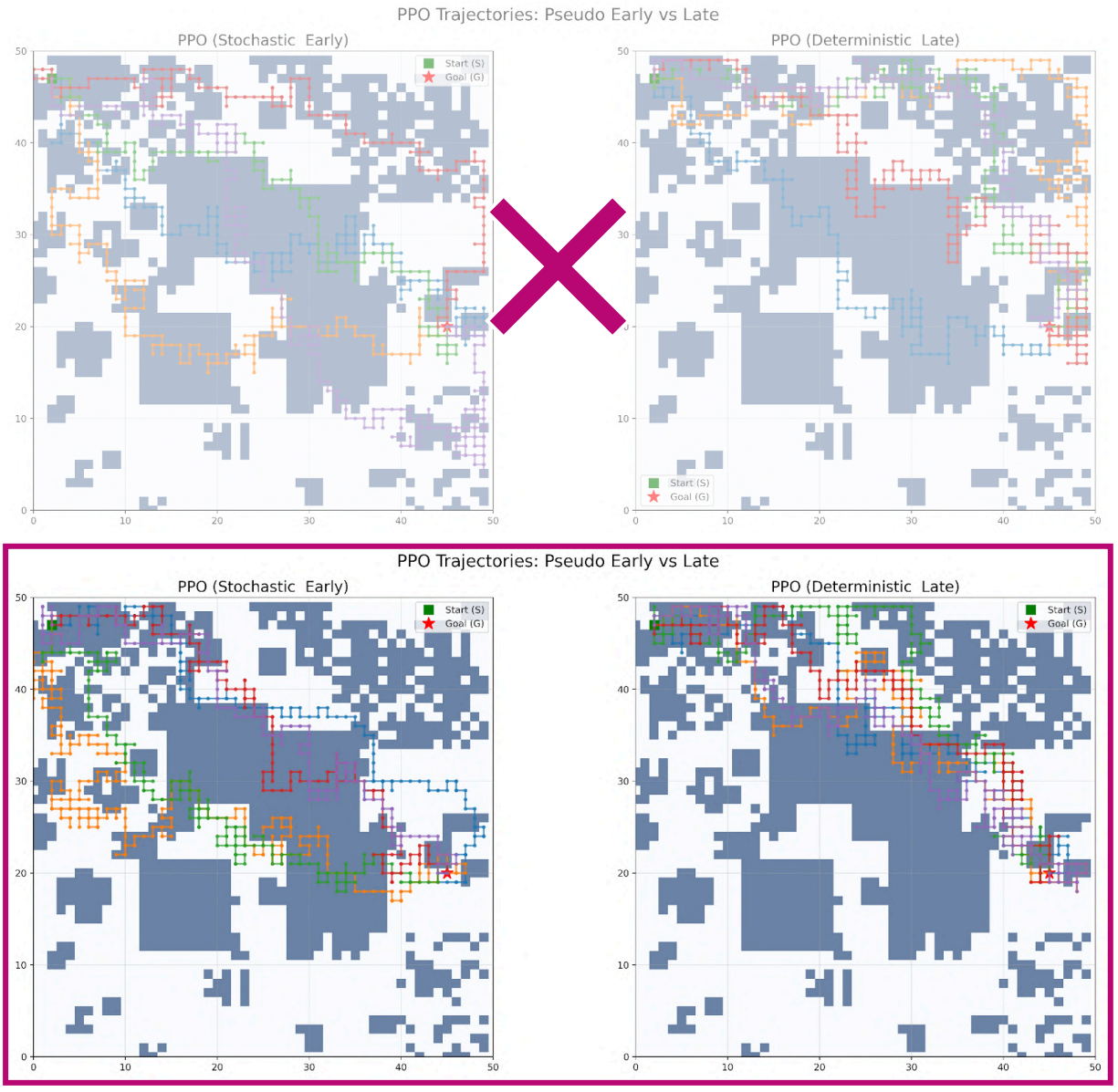
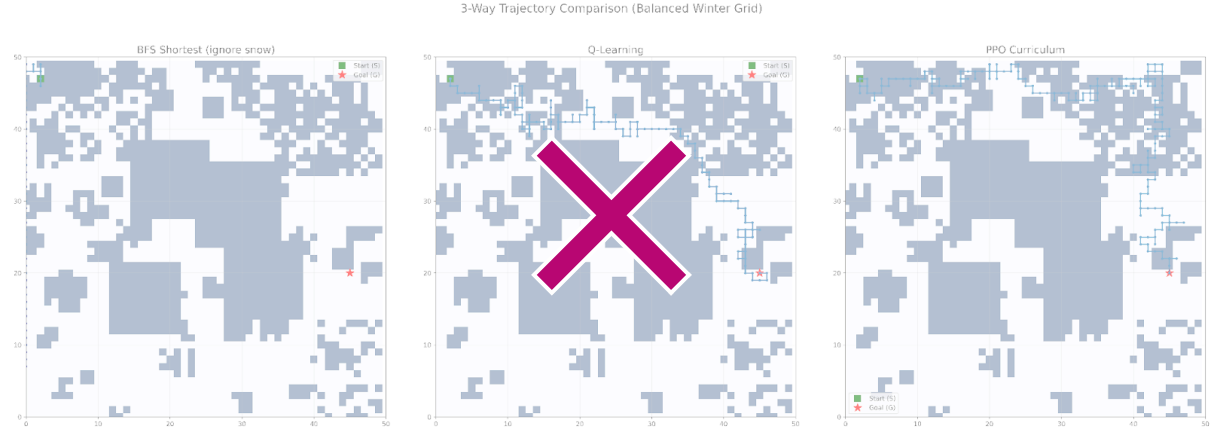


Figure 8: illustrates the navigation trajectories produced by the PPO agent during early and late stages of training under slip (stochastic) winter conditions.



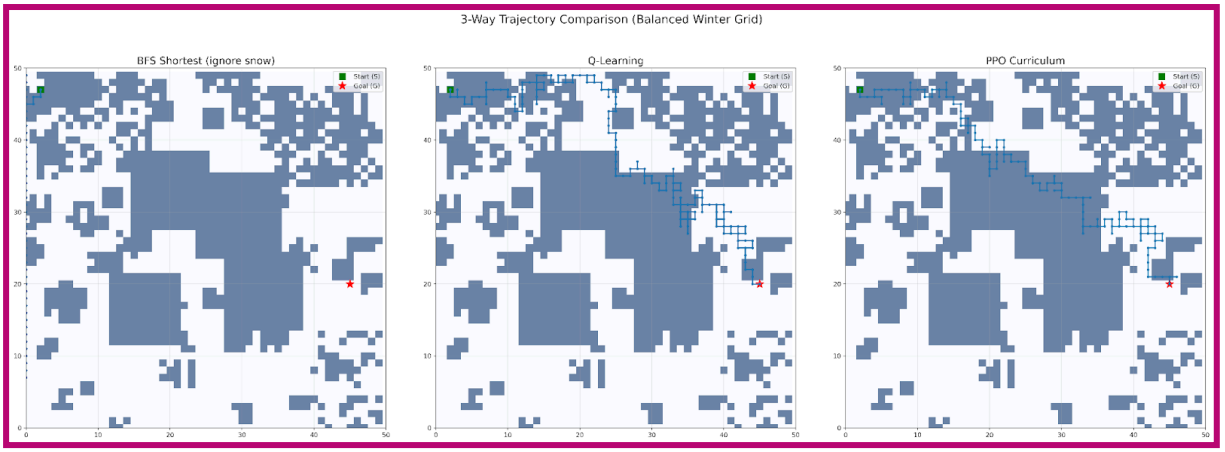


Figure 9: compares the navigation trajectories of BFS, Q-learning, and PPO under slip winter conditions.

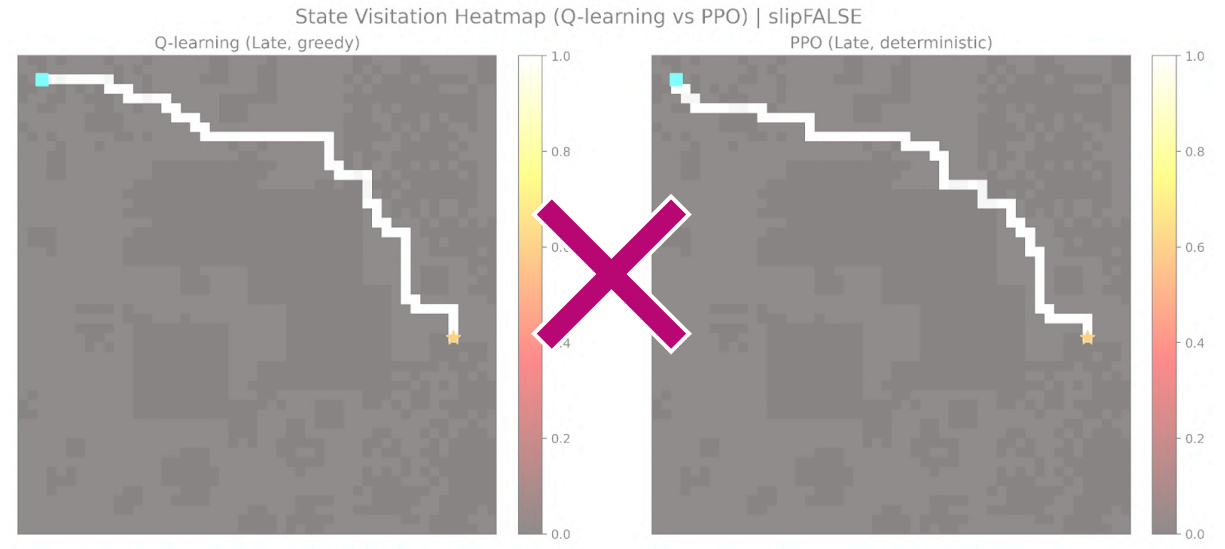
=== Final Comparison Table ===

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	-1693.96~~39.20~~± 61.27	1000.00± 0.00	**246.80**~~188.00~~± 76.09	0.0%
Q-Learning	304.03~~9.41~~±9.92	22**8.18**~~4.2~~ ± 4.10	67.58~~65.8~~± 4.34	100.0%
PPO Curriculum	252.19~~24.1~~±7.04	242.29~~55.8~~± 3.00	92.65~~100.9~~± 4.14	100.0%

Table 2: summarizes the quantitative performance metrics for BFS, Q-learning, and PPO under slip winter conditions, using the same evaluation metrics as in the deterministic case, standard deviation, and averaged across 5 independent random seeds.

4.3 Spatial State Visitation Heatmaps

Spatial state visitation heatmaps offer a grid-based way to see how often an agent passes through each cell during navigation. In our 50×50 winter grid, each cell corresponds to a state, and the color intensity reflects visitation frequency across episodes. Darker areas indicate cells that are visited more frequently, while lighter or white cells indicate rarely or never visited locations. Such visualizations give a clear picture of the agent's overall path distribution and behavior patterns. We generated these heatmaps using aggregated visitation counts normalized to the [0,1] range.



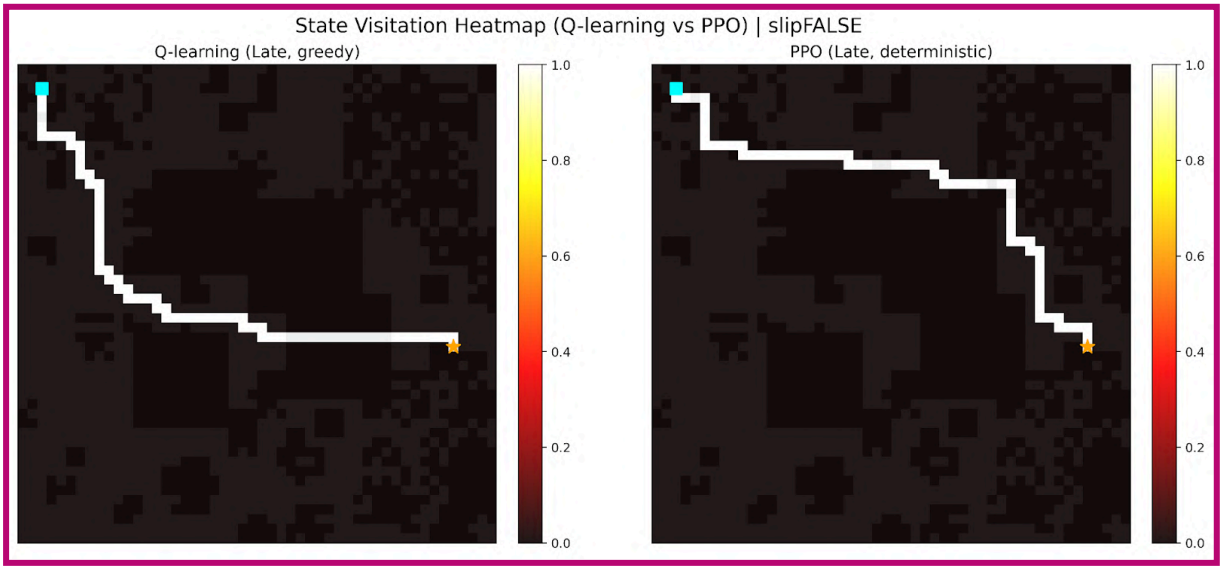


Figure 10: displays the visitation heatmaps for Q-learning and PPO under deterministic (non-slip) winter conditions. The results are aggregated from 10 evaluation runs consisting of 200 episodes per run. ~~episodes~~. Both agents concentrate most visits along a narrow corridor connecting the start to the goal, with near-maximum intensity (≈1.0) along the primary route and very low visitation elsewhere.

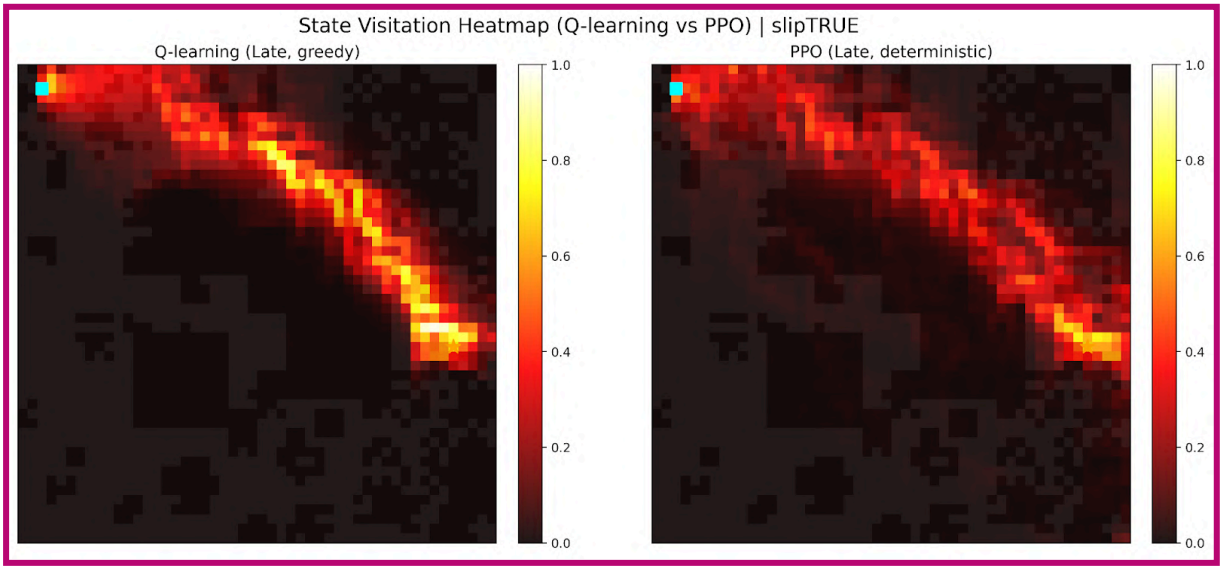


Figure 11: shows the visitation intensity for each grid cell in the slip case. Compared with the non-slip condition, visitation is more widely distributed across the grid rather than remaining within a narrow corridor. The heatmap aggregates data from 10 evaluation runs consisting of 200 episodes per run.~~episodes~~.

4.4 Limitations & Uncertainty

Several sources of uncertainty and limitations appeared in this study. Both Q-learning and PPO showed noticeable performance differences across training runs, which was expected given the stochastic nature of the environment and the learning algorithms.To account for this variability, all experiments were repeated over 5 independent random seeds, and results are reported as the mean across runs.

Under slip conditions, the same policy could produce quite different routes from episode to episode because actions did not always lead to the expected outcome. In addition, stochastic exploration strategies and random initialization of value functions (or policy networks) exposed agents to slightly different states, transitions, and rewards across runs. This variability made it harder to get perfectly consistent results.

An unexpected issue was that ~~even~~ the BFS baseline ~~sometimes~~ failed to reach the goal within the 1,000-step limit under slip conditions, resulting in a number of unsuccessful evaluation episodes.

A full numerical breakdown of evaluation metrics is provided in Appendix B.

5.Discussion

5.1 Restatement of Hypothesis and Summary of Findings

The study hypothesized that reinforcement learning-based agents, particularly Proximal Policy Optimization (PPO), would achieve more energy-efficient winter navigation paths than a traditional shortest-path baseline and a value-based Q-learning agent under both deterministic and slip winter conditions. The averaged results partially supported the hypothesis. ~~The experiment result did not support the hypothesis~~. Under deterministic (non-slip) conditions, PPO~~Q-learning~~ achieved higher cumulative reward, and fewer snow cell visits, outperforming Q-learning, which supports the hypothesis. ~~than Q-learningPPO under both deterministic and slip conditions.; butit also did~~ However under stochastic (slip) conditions, Q-learning ~~it not~~ outperformed PPO~~Q-learning~~ in terms of cumulative reward or snow-cell avoidance, which did not support the hypothesis. Both reinforcement learning methods perform better results than traditional BFS across these environments.

5.2 Interpretation of Q-Learning and PPO Performance

~~Q-learning achieved a higher performance than Proximal Policy Optimization agent in both deterministic and stochastic winter environments.~~ The results showed that algorithm performance varied depending on the environmental condition. Under deterministic (non-slip) conditions, PPO achieved higher cumulative reward and fewer snow cell visits than Q-learning, suggesting that PPO's policy gradient approach was able to learn a more energy-efficient route in a stable environment. Under stochastic (slip) conditions, Q-learning outperformed PPO in terms of cumulative reward and snow-cell avoidance. One possible explanation was that a discrete and clear grid-world environment and relatively small environment would favor more for Q-learning, therefore allowing Q-learning to efficiently estimate optimal state [7]. In contrast, PPO relied on function approximation and stochastic policy updates, which may have required more training data or longer training time to converge to an equally optimal policy under slip conditions. Additionally, its stochastic policy may have introduced suboptimal choices that were not effective, therefore reducing its total rewards relative to Q-learning under stochastic conditions [3].

5.3 Effects of Slip (Stochastic) Winter Conditions

Under slip (stochastic) conditions, all agents showed broader trajectory dispersion and increased visits to previously visited states because of slipping, as reflected in the

spatial state visitation heatmaps. This behavior is consistent with probabilistic transition dynamics, in which the same action could lead to different movement outcomes across episodes, such as deviating left in one run and right in another. When the grid is slippery, agents don't follow one stable path anymore. As a result, the routes spread out, and their performance becomes less consistent across episodes. These observations are consistent with prior navigation studies showing that stochastic environments increase policy variance and reduce convergence stability in reinforcement learning tasks [8].

5.4 Interpretation of BFS Baseline Behavior

The Traditional baseline trajectories did not take account for winter grid costs or stochastic transitions, it focused on the fastest way to the destination. In non-slip conditions this approach naturally produced the optimal route. However, under slip conditions, BFS exceeded the maximum step limit in multiple evaluation episodes, reflecting its inability to adapt its policy in response to transition winter grid costs or stochastic transitions.

Because BFS always assumed deterministic movement, each slip caused deviations from the planned path. These deviations often led to inefficient loops and longer paths. Unlike reinforcement learning methods, BFS did not have reward feedback or policy updates, which limited its ability to adjust navigation behavior under changing environmental dynamics. Therefore, making it the least effective method.

5.5 Limitations and Future Directions

Several limitations should be noted when interpreting these findings. All experiments were carried out in a simulated 50×50 grid environment with a relatively small and simple state space, which differs significantly from real-world scenarios. This setup likely favored value-based methods like Q-learning, since it could learn precise state-action values for every cell. In contrast, PPO depends on neural network function approximation and stochastic policy updates, which may need more training steps or data to perform well in such discrete, small-scale settings.

~~The number of evaluation episodes was also limited, which reduced the statistical reliability of the results.~~ To improve statistical reliability, all experiments were run independently over 5 random seeds and results were averaged across runs. Noticeable performance variability was observed across training runs, mainly due to the stochastic environment, random action outcomes, and probabilistic transitions under slip conditions.

The simulated environment simplified real-world winter driving conditions and did not capture vehicle dynamics, sensor noise, road-surface variation, or realistic energy-consumption models, these are factors that might lead to changes. As a result, these findings to real-world winter navigation scenarios were limited [9].

Future research could evaluate reinforcement learning agents in larger scale, continuous, or more realistic winter navigation environments with higher-dimensional state spaces and more complex conditions. Then it may be better to reflect real-world uncertainty and could allow policy-gradient methods such as PPO to fully function their theoretical advantages in handling stochastic and high-dimensional control problems.

---

6. Conclusion

This study compared a traditional BFS baseline with two reinforcement learning methods—Q-learning and Proximal Policy Optimization (PPO)—for energy-efficient navigation in a 50×50 winter grid under both deterministic (non-slip) and stochastic (slip) conditions. The averaged results across 5 independent random seeds partially supported the original hypothesis. ~~The results did not support the original hypothesis that PPO would produce more energy-efficient paths than Q-learning or BFS.~~ ~~Instead, Q-learning achieved the highest cumulative rewards in both settings and outperformed PPO in snow-cell avoidance.~~ Under deterministic (non-slip) conditions, PPO achieved the highest cumulative reward and fewest snow cell visits, outperforming Q-learning. However, under stochastic (slip) conditions, Q-learning outperformed PPO in cumulative reward and snow-cell avoidance. Although both reinforcement learning methods clearly outperformed the BFS baseline, BFS struggled under slip conditions due to its inability to adapt to stochastic transitions.

These findings show that whether value-based methods or policy-gradient methods perform better really depends on how unpredictable the environment is. In our small, structured discrete grid, PPO gained an advantage from its stable gradients and curriculum learning when everything was deterministic. On the other hand, Q-learning's simple tabular updates turned out to be more robust when the roads became slippery and actions sometimes failed.¶

~~These findings highlight that the relative performance of value-based and policy-gradient methods is highly dependent on the degree of environmental stochasticity. In this small and structured discrete state space, PPO benefits from stable gradients and curriculum learning in deterministic settings, while Q-learning's tabular off-policy updates provide greater robustness under stochastic transitions.~~ ~~These findings indicate that in small and structured state spaces, PPO demonstrated stronger performance in stable deterministic settings, while Q-learning showed more adaptability under stochastic conditionsvalue-based methods can be more effective than policy-gradient approaches when environmental complexity is limited~~. More generally, the results emphasize that the choice of reinforcement learning method should be guided by the structure and constraints of the task, rather than by theoretical advantages alone. Future work could explore whether PPO's advantages extend to more

complex environments with continuous states or larger state spaces, where its policy-gradient approach may better demonstrate its theoretical strengths.

Reference

[1]Mao, R., Xu, W., Qian, Y., Li, X., Li, Y., Li, G., & Zhang, H. (2025). Understanding the Determinants of Electric Vehicle Range: A Multi-Dimensional Survey. *Sustainability*, 17(10), 4259. https://doi.org/10.3390/su17104259

[2]Carlson, A., & Vieira, T. (2021). *The effect of water and snow on the road surface on rolling resistance* (VTI Report 971A). Swedish National Road and Transport Research Institute. https://www.diva-portal.org/smash/get/diva2:1542142/FULLTEXT01.pdf

[3]Watkins, C.J.C.H., & Dayan, P (1992). Q-learning. *Mach Learn* 8, 279–292¶ ~~(1992).~~ https://doi.org/10.1007/BF00992698

[4]Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). *Proximal Policy Optimization Algorithms* (No. arXiv:1707.06347). arXiv. https://doi.org/10.48550/arXiv.1707.06347

[5]Sutton, R. S., & Barto, A. G. (2018~~n.d.~~). Reinforcement Learning: An Introduction. https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf

[6]Dayan, P., & Balleine, B. W. (2002). Reward, Motivation, and Reinforcement Learning. *Neuron*, 36(2), 285–298. https://doi.org/10.1016/S0896-6273(02)00963-7

[7]Tan, C. (2025). Comparative Study of Reinforcement Learning Performance Based on PPO and DQN Algorithms. *Applied and Computational Engineering*, 175(1), 30–36. https://doi.org/10.54254/2755-2721/2025.AST24879

[8]Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A. J., Banino, A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., Kumaran, D., & Hadsell, R. (2017). *Learning to Navigate in Complex Environments* (No. arXiv:1611.03673). arXiv. https://doi.org/10.48550/arXiv.1611.03673

[9]Chukwurah, N., Adebayo, A. S., Ajayi, O. O., & Anfo Pub. (2024). *Sim-to-Real Transfer in Robotics: Addressing the Gap between Simulation and Real- World Performance.International Journal for Multidisciplinary Research (IJFMR),* 05(01), 33–39. https://doi.org/

10.54660/.IJFMR.2024.5.1.33-39
[10]Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). Curriculum learning. *Proceedings of the 26th Annual International Conference on Machine Learning*, 41–48. https://doi.org/10.1145/1553374.1553380

Appendices

Appendix A Experimental Configuration and Algorithmic Details:

A.1 Environment Configuration

All experiments were conducted in a custom 50×50 winter grid environment (2,500 discrete states).

- Start position: (47, 2)
- Goal position: (20, 45)
- Maximum steps per episode: 1000
- Action space: {up, down, left, right}

Two transition settings were evaluated:

Deterministic (slip = FALSE)

The intended action is executed exactly.

Stochastic (slip = TRUE)

With probability ~~0.8~~ $\frac{1}{3}$, the intended action is executed.

With probability ~~0.2~~ $\frac{2}{3}$, the agent slips to a random adjacent direction.

All algorithms were evaluated on the same fixed grid layout to ensure fairness.

A.2 Reward Function

The reward function models energy-aware winter navigation.

At each time step:

- Step penalty: −1.5
- Snow penalties:
  - Near snow: −0.2
  - Edge snow: −0.5
  - Core snow: −2.0
- Goal reward: +700

The cumulative episode reward is:

$$R = \sum_{t=1}^{T} (-1.5 - C_{snow}(s_t)) + 700 \cdot 1_{goal\ reached}$$

where $C_{snow}(s_t)$ denotes the terrain penalty and $1_{goal\ reached}$ indicates successful termination.

A.3 Q-Learning Configuration

Tabular Q-learning was implemented with the following hyperparameters:

- Learning rate α=0.15
- Discount factor γ=0.99
- Exploration strategy: ε-greedy
- Initial ε = 0.60
- Minimum ε = 0.05
- Exponential decay per episode = 0.9995
- Training episodes = 15,000
- Max steps per episode = 1000

The Q-update rule is:

$Q(s,a) \leftarrow Q(s,a) + \alpha[R + \boldsymbol{\gamma}\, max_{a'}\, Q(s',a') - Q(s,a)]$

During evaluation, a fully greedy policy (ε = 0) was used.

A.4 Proximal Policy Optimization (PPO) Curriculum Configuration

PPO was implemented as a stochastic policy-gradient method.

Total training timesteps: 1,200,000

- Phase 1 (**navigation-focused reward**~~**deterministic environment**~~): 300,000 timesteps
- Phase 2 (**energy-aware reward**~~**stochastic environment**~~): 900,000 timesteps

Evaluation was conducted using deterministic action selection.

PPO Objective Function

PPO optimizes the clipped surrogate objective:

$L^{CLIP}(\theta) = E_t[min(r_t(\theta)\widehat{A}_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)\widehat{A}_t)]$

where

$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$

and A^t~~A^t~~ is the generalized advantage estimate.

The clipping mechanism constrains policy updates to maintain stability.

PPO Hyperparameters

- Discount factor γ=0.99~~γ=0.99~~
- GAE parameter λ=0.95~~λ=0.95~~
- Entropy regularization enabled
- Learning rate: 3×10^{-4}

- Clip range: 0.2
- Rollout buffer: 2,048 steps
- Batch size: 256
- Optimization epochs: 10
- Neural network architecture: two hidden layers (256~~512~~ units each, ReLU activation)

A.5 Evaluation Protocol

For each condition (slip FALSE / TRUE):

- One trained model per algorithm
- Maximum evaluation length: 1000 steps
- **Number of evaluation episodes: 200**
- **Number of seeds: 5**
- **Heatmap runs: 10**
- Metrics recorded:
  - Total cumulative reward
  - Number of steps
  - Snow cell visits
  - Success rate (%)

State visitation heatmaps were generated by aggregating visitation frequencies across evaluation runs of 200 episodes each.

Appendix B Detailed Quantitative Results:

B.1 Deterministic (slip = FALSE)

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	575.30±0.00	70.00±0.00	31.00±0.00	100.0%
Q-Learning	579.10~~589.50~~±5.61	70.00±0.00	12.60~~7.00~~±3.29	100.0%
PPO Curriculum	588.50~~587.5~~±0.60	70.00±0.00	7.80~~11.00~~ ±1.10	100.0%

All methods reached the goal within 70 steps.
PPO~~Q-learning~~ achieved the highest cumulative reward and fewest snow cell visits~~due to reduced snow traversal~~.

B.2 Stochastic (slip = TRUE)

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	−1693.96~~39.2~~ ±61.27	1000 ±0.00	246.80~~188.0~~ ±76.09	0%
Q-Learning	304.03~~309.4~~ ±9.92	228.18~~4.2~~ ±4.10	67.58~~65.8~~ ±4.34	100.0%
PPO Curriculum	252.19~~224.1~~ ±7.04	242.29~~55.8~~ ±3.00	92.65~~100.9~~ ±4.14	100.0%

Under stochastic dynamics:

- BFS fails due to inability to adapt.
- Q-learning demonstrates greater robustness.
- PPO maintains success but exhibits higher trajectory dispersion.

B.3 State Visitation Analysis

State visitation frequency was computed as:

$$V(s) = \frac{N(s)}{max_s N(s)}$$

where $N(s)$ is the number of visits to state s.

- Under deterministic conditions, visitation concentrates along a narrow corridor.
- Under stochastic conditions, visitation becomes more dispersed.
- Q-learning exhibits more focused routing than PPO under slip conditions.

Dear Reviewer 1,

Thank you for your thoughtful comments on my research paper. I have carefully addressed each of your comments.

After submitting the first manuscript, I continued improving the study by extending the code and conducting more advanced multi-seed experiments. In response to the reviewers' comments, I updated the manuscript with these results.

Comment 1: several areas could be strengthened. First, the introduction would benefit from more consistent and rigorous citation support. For example, claims such as "rolling resistance can rise by approximately 30–40% under wet or snowy conditions" and statements about the validity of the grid abstraction would be more convincing if directly supported with citations. Strengthening this connection to prior work would improve both credibility and scholarly rigor.

Response 1: After viewing the comment, I have added citation [2] (Carlson & Vieira, 2021) to the sentence "rolling resistance can rise by approximately 30–40% under wet or snowy conditions". Additionally, I also added a justification for the grid abstraction, supported by a citation of [5] (Sutton & Barto, 2018). 1.) "Grid-based environments are commonly used in reinforcement learning studies, as they simplify complex continuous real-world environments into discrete states [5]. This abstraction is particularly well-suited for this study, as it enables environmental factors such as snow coverage and surface slip conditions to be incorporated directly into the cell-level cost and reward structure, supporting controlled and reproducible comparison of different routing methods." Please view line 47 and line 55-60.

Comment 2: Second, while the motivation is clearly articulated, the paper would benefit from a more explicit discussion of gaps in existing research. The authors highlight the importance of multi-criteria navigation under winter conditions, but it remains unclear what specific limitation in the current literature this study addresses. Clarifying whether the primary contribution is extending RL-based navigation to account for weather-related energy costs (or something more novel) would help better position the work.

Response 2: I have added a paragraph to the introduction stating the research gap. Specifically, we note that prior work has primarily focused on general navigation tasks without explicitly modeling weather-dependent energy costs. "Despite growing interest in reinforcement learning for navigation, existing studies have largely overlooked the impact of winter environmental conditions, such as snow coverage and surface slippiness, on energy-aware path planning. Prior work has primarily focused on general navigation without taking account of explicitly modeling weather-dependent energy costs [8]. To address this gap, this study integrates snow coverage, stochastic slip condition, and energy costs into a comparative framework, evaluating Q-learning, PPO, and a traditional Baseline under realistic winter routing scenarios. This study is one of the first to directly compare value-based and

policy-gradient reinforcement learning methods in an energy-aware winter navigation setting." Please view line 62-71.

Comment 3: Third, aspects of methodological rigor could be improved. The paper would benefit from more justification of hyperparameter choices and additional experimental robustness, such as averaging results over multiple runs and reporting variance. This would strengthen confidence in the findings, especially given the stochastic elements of the environment.

Response 3: I have added a section explicitly to justify the hyperparameter choices for Q-learning and PPO in method 3.4. Moreover, I have made the results to an average of 5 independent seeds and including the standard deviation for more accurate results and variance in both deterministic and stochastic conditions. Therefore, the study has been updated to reflect these improved results. Please view 3.4 and Result.

Extra revisions: Following the averaging of results over 5 independent random seeds, the findings changed slightly from the original submission. Specifically, PPO outperformed Q-learning under deterministic conditions, partially supporting the original hypothesis. As a result, the Abstract, Methods, Discussion, and Conclusion have been updated to accurately reflect these new findings.

Dear reviewer 2,

Thank you for your positive and constructive feedback. I greatly appreciate your careful evaluation and helpful suggestions. I have revised the manuscript accordingly and addressed your comments as follows.

After submitting the first manuscript, I continued improving the study by extending the code and conducting more advanced multi-seed experiments. In response to the reviewers' comments, I updated the manuscript with these results.

Comment 1: Methods section needs the most improvement, especially the explanations of the environment, how rewards were given, and the training procedures. That is where the paper becomes the most repetitive and hardest to follow. Tightening that section and making the setup more direct would do the most to improve the paper.

Response 1: I have consolidated all key experimental details into the main Methods section. Specifically:

- The 50×50 grid, start position (47, 2), and goal position (20, 45) are stated in Section 3.1
- Slip probabilities (1/3 intended action, 2/3 random direction) are now explicitly stated in Section 3.1
- Reward and penalty values (-1.5 step penalty, -0.2/-0.5/-2.0 snow penalties, +700 goal reward) are stated in Section 3.2.3
- Q-learning training length (15,000 episodes) is stated in Section 3.2.2
- PPO training length (1,200,000 timesteps) is stated in Section 3.2.4
- Evaluation procedure (200 episodes per run, 5 seeds) is stated in Section 3.3

Additionally, I have removed repetitive descriptions in the Methods section to improve clarity and readability. A new Section 3.4 has been added to provide explicit justification for all hyperparameter choices for both Q-learning and PPO. Appendix A is retained as a complete numerical summary for reference.

Comment 2: The paper includes a Works Cited / References section, and there are corresponding citations throughout the paper. The main issue is minor consistency and formatting, not the absence of citation.

Response 2:

1. Corrected the formatting of [3] Watkins & Dayan (1992) to follow standard APA style

2. Updated [5] Sutton & Barto from (n.d.) to the correct publication year (2018)

3. Added the full journal name to [9] — *International Journal for Multidisciplinary Research* (IJFMR)

Comment 3: The Methods section would benefit from a clearer presentation of the experimental setup. At present, important details are split between the main text and Appendix A. I would suggest bringing the core information together in one place and stating it plainly: the 50×50 grid, the start and goal positions, the slip probabilities, the reward and penalty values, the training length for Q-learning and PPO, and the evaluation procedure. It would also help to replace general phrases such as "large rollout buffers" and "multiple optimization epochs" with the actual PPO settings, and to state directly how many evaluation episodes were used.

Response 3: I have made the following revisions:

Consolidated key details into the main Methods section:

- 50×50 grid, start position (47, 2), and goal position (20, 45) — Section 3.1
- Slip probabilities (1/3 intended action, 2/3 random direction) — Section 3.1
- Reward and penalty values (-1.5 step penalty, -0.2/-0.5/-2.0 snow penalties, +700 goal reward) — Section 3.2.3
- Q-learning training length (15,000 episodes) — Section 3.2.2
- PPO training length (1,200,000 timesteps) — Section 3.2.4
- Evaluation procedure (200 episodes per run, 5 seeds) — Section 3.3

Replaced vague PPO language with actual values:

- rollout buffer of 2,048 steps
- 10 optimization epochs per update
- Batch size explicitly stated as 256

Evaluation episodes count explicitly stated:

- 200 episodes per run

Additionally, repetitive descriptions have been removed, and a new Section 3.4 has been added for explicit hyperparameter justification.

Extra revisions: Following Reviewer 1's suggestion to average results over 5 independent random seeds, the findings changed slightly from the original submission. Specifically, PPO outperformed Q-learning under deterministic conditions, partially supporting the original hypothesis. As a result, the Abstract, Methods, Discussion, and Conclusion have been updated to accurately reflect these new findings.

I have reviewed the authors' letter and changes and am very happy with the updates they made and would like to recommend this paper be accepted for publication.

It is clear they took the feedback seriously and made meaningful improvements to both the framing and rigor of the study. The additions to the introduction and method sections significantly strengthen the paper. In particular, the rationale and importance of the paper as well as decisions around hyperparameters are much clearer. Additionally, moving to multi-seed experiments with reported averages and standard deviations meaningfully improves the robustness of the results, which is particularly important given the stochastic nature of the environment and reinforcement learning methods. The updated findings add nuance to their conclusions and strengthen the comparative insights between value-based and policy-based methods.

Overall, these revisions substantially improve the paper’s clarity, rigor, and contribution. The study now provides a more compelling and well-supported comparison of RL approaches for energy-aware navigation under winter conditions.

The revised manuscript is much stronger than the earlier version and is now, in my opinion, suitable for acceptance. The paper offers a clear comparison of three route-planning approaches in a winter grid-world setting: BFS, Q-learning, and PPO. Its main strength is that all three methods are tested under the same setup, including the same map, reward structure, start and goal positions, and evaluation criteria. That makes the comparison straightforward and allows the differences among the methods to come across clearly.

The revision has improved the paper in important ways, especially in the Methods section. In the earlier version, some of the setup was harder to follow, but the revised manuscript presents the experimental design more directly. The additional information about the environment, training process, evaluation methods, and Proximal Policy Optimization (PPO) settings improves the paper's clarity and makes it easier to understand. Moreover, the use of multiple runs with different random seeds strengthens the reliability of the reported results.

The Results and Discussion sections remain the strongest parts of the manuscript. The revised paper now gives a more careful interpretation of the findings. Rather than pushing a single overall winner, it shows that performance depends on the setting: PPO does slightly better in the deterministic case, while Q-learning is more robust under slip conditions. That makes the conclusion more persuasive.

Overall, I found the paper well organized, readable, and worth publishing. The subject matter is pertinent, and the comparison is well-articulated. This revised manuscript represents a valuable contribution, particularly as a simulation-based study. I recommend its acceptance.

<u>Decision:</u> Accept with moderate revisions

<u>Review:</u> This manuscript presents a comparative study of three path-planning approaches (BFS, Q-learning, and PPO) under a grid-world abstraction designed to model winter driving conditions via energy penalties and stochastic transitions. The study evaluates performance using cumulative reward, snow-cell traversal, success rate, and trajectory characteristics under both deterministic ("non-slip") and stochastic ("slip") settings.

The core finding is that Q-learning outperforms PPO in stochastic environments, while PPO may perform competitively or better in deterministic settings. The paper aims to position this as evidence that value-based methods can outperform policy-gradient methods in structured, discrete environments.

The experimental setup is clear and has improved after revisions, notably via multi-seed evaluation and clearer methodological specification. These are laudable; however, the manuscript still falls a bit short of the level of rigor, positioning, and analysis expected for publication in *Convergence Journal*. I believe the framing, methodology, and experimental design/setup are there, but the execution overall (writing + experiments) could be improved.

Major comments:

1. While the revision introduces multi-seed experiments using 5 seeds, this could be strengthened for stochastic RL evaluation. There are also no tests for statistical significance, no confidence intervals, and no discussion of variability across runs.
    a. I would like the author to increase the experiments to at least 10 (maybe even 20) seeds and report statistics (e.g., mean +/ standard deviation, median + 95% confidence intervals, etc.). Just something of this sort would be great and help alleviate concerns about the robustness and consistency of your results.
    b. Include at least one statistical test (e.g., Welch's t-test) comparing Q-learning vs. PPO (slip condition)
    c. Alongside these tests, you should add a short paragraph discussing and interpreting the variance. In particular, I would want you to answer whether PPO exhibits higher instability.
2. The research gap still does not entirely distinguish between RL path planning literature, cost-aware navigation, and grid-world RL benchmarks. Could you add another paragraph answering more explicitly what prior work has been done, what is missing, and what exactly is novel
3. Please add a plot of learning curves: training reward vs. timesteps for **both** Q-learning and PPO, along with the statistics from point 1 above (CIs, mean +/ stds, whatever you end up using). Would reveal instability, too, especially for PPO

Minor comments:

1. The paper states outcomes (e.g., Q-learning outperforms PPO) but could explain these in a deeper, more mechanistic way. *Why* does Q-learning work better in discrete grid worlds? *Why* does PPO struggle?

    a. For the first question, consider: exact value iteration in finite MDP, lower variance than policy gradients
    b. For the second question, consider: high variance gradients under stochasticity, issues with sparse and noisy rewards
2. All results are based on a single hyperparameter configuration per method, which is concerning. How stable are the results to changes in hyperparameters? Could you add at least one sensitivity analysis? You do not need to go overboard, but this could be good to alleviate skepticism towards claims about one method being more suited over the other…
    a. For Q-learning: vary the decay or learning rate
    b. For PPO: vary learning or clipping parameter
    c. This should be accompanied by a figure
3. Since curriculum is highlighted, it should ideally be validated. While I like how you introduced the curriculum learning setup, I am still wondering if the PPO performance depends heavily on the curriculum. Could you compare the PPO with and without the curriculum, or at least cite some literature that would answer this question? You may want to accompany this with some sort of ablation study plot(s)
4. Watch the informal phrasing throughout (e.g., the use of “etc.”)
5. I would add at least 2 more papers on PPO vs. value-based methods and/or RL navigation benchmarks

Energy-Efficient Path Planning in Winter Conditions: A Comparative Study of Traditional Baseline, Q-learning, and Proximal Policy Optimization in a Grid World Environment

1.Abstract

Recent investigation in winter route planning has become increasingly important due to increased resistance and reduced efficiency on winter roads. These conditions significantly increase energy consumption and introduce uncertainty, raising the risk of failing to reach the destination. To address the challenge, Reinforcement Learning (RL) is a type of machine learning where an agent learns to make decisions by interacting with an environment, receiving rewards or penalties for its actions to maximize cumulative rewards. For simulation, we benchmark a standard shortest-path algorithm (BFS) against two reinforcement learning methods: Q-learning and Proximal Policy Optimization (PPO). All approaches are tested on identical grid configurations, with the same starting points, goals, and reward functions. We assess their performance based on cumulative reward, snowy cell visits (as a proxy for energy cost), trajectory characteristics, and success rate. It is hypothesized that PPO algorithms will result in lower energy consumption than Q-learning and traditional baseline under winter conditions. The results show that while BFS consistently finds the shortest paths, it fails to consider energy costs or environmental uncertainty. RL agents, by comparison, adapt more efficiently to winter conditions. Under deterministic (non-slip) conditions, PPO achieved higher cumulative reward and fewer snow cell visits than Q-learning, partially supporting the hypothesis. However, under stochastic (slip) conditions, Q-learning outperformed PPO in cumulative reward and snow-cell avoidance. These results suggest that Q-learning is better suited for stochastic winter navigation in grid worlds, while PPO may perform better in more deterministic environments with continuous states or decisions.

Key words: Energy-Efficient, Traditional Baseline, Q-learning, Proximal Policy Optimization, grid world, reinforcement learning.

2.Introduction

As demand in Electric Vehicles (EV) become increasingly widespread, energy efficiency and route optimization have emerged as critical challenges, particularly in winter when resistance increases. Sub-zero temperatures, combined with snow and ice accumulation on road surfaces, significantly increase energy consumption, resulting in reduced driving range and greater unpredictability in trip planning. A 2025 study shows that an estimated 50 % of EV driving range can be reduced in cold climates, including snow and ice covering terrain, highlighting the significant impact of environmental

conditions on energy consumption [1]. Furthermore, when snow or water remains on the road surface, vehicle tires must continuously displace through ice as they roll and move, hence forcing the vehicle to draw more power [2]. On top of that, water cools tires more effectively than air alone, altering their mechanical properties and further pushing rolling resistance higher. Previous studies have shown that rolling resistance can rise by approximately 30% to 40% under wet or snowy conditions [2]. Ultimately, these factors make winter driving a major concern for Electric Vehicle (EV) efficiency and reinforce the need for energy-aware routing strategies.

To approach these challenges, we have explored a few computational methods to allow agents to learn effective navigation strategies aimed at minimizing energy consumption. In this study, a 50x50 grid is used as an abstraction routing map that circles key structures of Canadian snow distribution, rather than exact geographical terrain. Grid-based environments are commonly used in reinforcement learning studies, as they simplify complex continuous real-world environments into discrete states [3]. This abstraction is particularly well-suited for this study, as it enables environmental factors such as snow coverage and surface slip conditions to be incorporated directly into the cell-level cost and reward structure, supporting controlled and reproducible comparison of different routing methods.

Despite growing interest in reinforcement learning for navigation, existing studies have largely overlooked the impact of winter environmental conditions, such as snow coverage and surface slippiness, on energy-aware path planning. Prior work has primarily focused on general navigation without taking account of explicitly modeling weather-dependent energy costs [4]. To address this gap, this study integrates snow coverage, stochastic slip condition, and energy costs into a comparative framework, evaluating Q-learning, PPO, and a traditional baseline under realistic winter routing scenarios. This study is one of the first to directly compare value-based and policy-gradient reinforcement learning methods in an energy-aware winter navigation setting .

Prior work on reinforcement learning for navigation can be categorized into three directions. First, Grid world benchmarks commonly adopt tabular Q-learning and deep reinforcement learning variants as standard baselines for evaluating discrete navigation tasks [3,5], but focus primarily on task completion rather than energy awareness and energy sensitive routing. Second, cost-aware navigation research has examined energy minimisation in uneven terrains using reinforcement learning [6], yet rarely incorporated weather dependent components, such as snow coverage or stochastic traction loss. Third, existing comparisons between value-based and policy-gradient methods [7,8], including curriculum based PPO applications [9] are conducted primarily under fixed and stable conditions, without examining the possibility of a particular reinforcement learning method remaining dominant to another when action outcomes become probabilistic, as in stochastic environments such as winter slip conditions. The

work in this study addresses all three gaps simultaneously by implementing an energy-aware reward structure into a reproducible 50×50 grid-world benchmark, embedding snow-density penalties and a 2/3-slip probability to simulate realistic winter routing conditions, and directly comparing Q-learning, PPO, and a traditional baseline under both deterministic and stochastic environments.

More specifically, the research will include a traditional baseline algorithm, and two types of reinforcement learning, which are Q-learning and PPO algorithm. The traditional baseline routing computes the shortest path based on static cost metrics and follows the predefined route without adaptation or learning (basic routing). Q-learning, a value-based reinforcement learning method, learns through incremental dynamic programming processes with computational requirements and it is well suited for agents to improve and refine action values and achieve effective performance in controlled Markovian domains [7]. In contrast, PPO is a policy-gradient algorithm that primarily optimizes a stochastic policy and achieves greater training stability through constrained policy updates with a clipped surrogate objective function [8].

This comparison evaluates the effectiveness of the proposed routing methods in identifying energy-efficient paths. Performance is evaluated through energy related costs, overall routing efficiency, and observed navigation behavior under both deterministic (non-slip) and stochastic (slip) settings. Because PPO balances stable policy updates with adaptive learning in uncertain environments, it is hypothesized to be the most effective method implemented for winter routing.

The following section goes into the methodology and experimental setup in more detail.

3.Methods

3.1 Environment Design

A custom winter navigation grid environment was designed to evaluate each Reinforcement Learning (RL) agent under stochastic and cost-sensitive conditions. The environment consists of a 50 x 50 grid world containing 2500 cells, including both normal terrain and snow-covered terrain. The simulation contains a fixed start (47, 2) and a fixed goal (20, 45) point, and are located in the top left corner and middle right respectively. At each time step, the agent can take either one of the four directions, up, down, left, and right; each step on a non-snow grid has an energy cost of -1.5 per step. Snow is modeled as spatially correlated regions with graded intensity, capturing the typical uneven accumulation patterns in Canadian winters.

When snow is modeled as a binary state, abrupt reward discontinuities make PPO's advantage estimates unstable, resulting in increased gradient variance. By contrast, a continuous representation of snow thickness provides smoother cost transitions, which

help reduce gradient variance and stabilize learning, making it more efficient. These snow types are classified as “Near snow”, “Edge snow”, and “Core snow”.
To evaluate robustness under different winter conditions, two transition settings were considered using the same grid map: a non-slip (deterministic) condition and a slip (stochastic) condition. In the deterministic setting, action always results in the intended movement, and snow cells only add additional energy loss. In the stochastic setting, actions do not always lead to the intended outcome: with a probability of 1/3, the intended action is executed, and with a probability of 2/3, the agent moves in a random adjacent direction, simulating loss of traction control during winter driving on ice and snow.

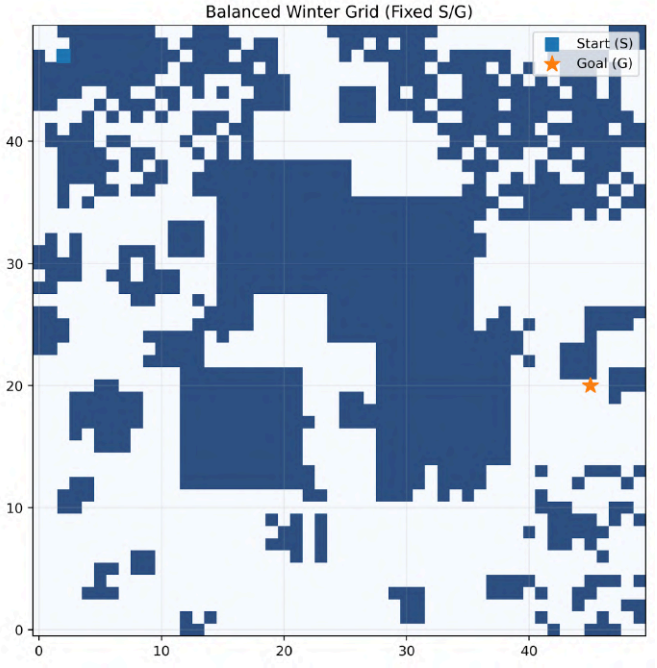


Figure 1: shows a typical winter grid layout used in all experiments.

3.2 Reinforcement Learning Algorithms/ Methodologies:

The routing strategies and learning methods used in this study are described in this section.

3.2.1 Traditional Baseline (BFS)

Traditional baseline is a non-learning shortest path strategy used as a comparison base for reinforcement learning methods. It operates on the 50x50 grid map, it considers four directions to move (up, down, left, right). The system calculates the shortest path to the destination, without considering energy savings, which means it does not adapt to environmental feedback or uncertainty.

3.2.2 Q-learning Implementation

Tabular Q-learning was used as the value-based reinforcement learning baseline. The agent maintains a discrete Q (s, a) table, which is updated iteratively based on observed state transitions.

An ε-greedy policy was used for exploration. The value of ε begins at 0.6 and decays exponentially with a factor of 0.9995 per episode until it reaches 0.05. In this 50×50 grid setting, the decay schedule allows broad initial exploration before shifting emphasis to exploitation. Actions are restricted to four possibilities—left (0), down (1), right (2), and up (3)—consistent with standard discrete grid-world formulations and the discrete grid structure. The Q-value update follows the classic temporal-difference form:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[R + \gamma \max_{a'} Q(s',a') - Q(s,a)] \quad \text{(Eq. 1)}$$

We fix the learning rate α at 0.15 and the discount factor γ at 0.99. These values handle the stochastic transitions (is_slippery=True) and support planning over long horizons in the large grid [3]. State s denotes the flattened grid position index. The action a selects one of the four movement directions. Q(s, a) represents the estimated discounted cumulative reward starting from state s and action a [3,7].

3.2.3 Reward system

In this research, the reward system is designed to represent energy efficiency under winter conditions, it helps agents find the best paths. At each step, the system gives a negative value to show how significant a step is to energy saving, with higher cause on snow grids. A positive 700 is given only when the agents successfully reach the destination, while within the maximum steps of 1000 steps. It encourages agents to reach the destination with considerations on energy saving. We initialize the cumulative reward to 0 at the beginning of each episode. Each will result in negative values, because by doing this, agents don't need to consider the prior bias, since it assumes nothing at first. As in the settings, we have Near-snow (light snow), Edge-snow (medium snow), and Core-snow (deep snow); each of them has different additional values which are -0.2, -0.5, and -2.0 respectively [10].

$$r_t = r_{step} - c_{energy}(s_t) + r_{goal} \quad \text{(Eq. 2)}$$

The basic reward function follows standard reinforcement learning practice by combining step penalties, energy-related costs, and a terminal goal reward.

3.2.4 Proximal Policy Optimization

Proximal Policy Optimization (PPO) served as the policy-gradient method in this study for energy-efficient routing under winter conditions. PPO learns a stochastic policy by directly outputting action probabilities for each state, which helps the agent cope with uncertainty on slippery roads.

$$L^{CLIP}(\theta) = E_t[\min(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)] \quad \text{(Eq. 3)}$$

where $r_t(\theta)$ is the probability ratio between the new and previous policies. The clipping mechanism constrains policy updates to improve stability.

The PPO agent was trained using fixed values across all experiments. The discount factor was set to γ=0.99, and Generalized Advantage Estimation (GAE) was used with λ=0.95. The policy network consisted of two fully connected hidden layers with 256 units each, using ReLU activation functions. The learning rate was set to 3×10^{-4}, with a clipping range of 0.2 and an entropy coefficient of 0.02 to encourage exploration. PPO training was conducted for 1,200,000 time steps, using a rollout buffer of 2048 steps, a batch size of 256, and 10 optimization epochs per update to improve training stability [8].

Policy-gradient methods work by directly adjusting a parameterized policy to increase expected reward. They learn a stochastic policy, allowing for more flexibility in uncertain environments, where the same action can lead to different outcomes. In contrast, q-learning estimates state-action values and usually converges with a deterministic policy based on these estimates. Although it can learn in unfamiliar environments, its action selection may be less reliable in highly unpredictable situations. As a result, policy-gradient and value-based methods show varying strengths depending on the level of uncertainty in the winter routing task.

3.2.5 Curriculum Learning for PPO

PPO training followed a two-phase curriculum learning approach designed to improve training stability and sample efficiency under sparse rewards and energy-sensitive penalties [11,8].

Phase 1: Navigation Learning Phase (300,000 timesteps)
In phase 1, PPO was trained using a simplified reward function that focuses only on positive feedback for reaching the goal. Energy costs in snow grids were not included. This made the feedback denser and more immediate. The agent quickly learned basic navigation and achieved early success in the 50×50 winter grid. The policy learned in this phase served as the starting point for the next training stages.

Phase 2: Energy-Aware Optimization Phase (900,000 timesteps)
In phase 2, the training continued with the complete energy-aware reward function, which now included penalties for moving across snow-covered areas. The agent had to balance successful arrival with lower energy consumption. All reported results, ablation studies, and comparisons are based solely on the Phase 2 reward function.

3.2.6 Curriculum Learning Ablation Study

To validate the necessity of curriculum learning, an ablation study was conducted comparing PPO with curriculum versus PPO without curriculum under both deterministic and stochastic conditions across 10 seeds assessed through total cumulative reward and success rate. Without Curriculum learning, PPO achieved a mean of -1720.15 (95% CI [-1894.63, -1545.67]) and -1362.39 (95% CI [-1780.01, -944.78]) with 0% and 10% success rate respectively, under deterministic and stochastic conditions. Whereas with curriculum, PPO demonstrates a strong performance of 588.73 (95% CI [588.25, 589.21]) and 258.69 (95% CI [251.02, 266.37]) with 100% success rate under both conditions. These results confirm that curriculum learning is a critical component for PPO, as its absence leads to failure under both conditions.

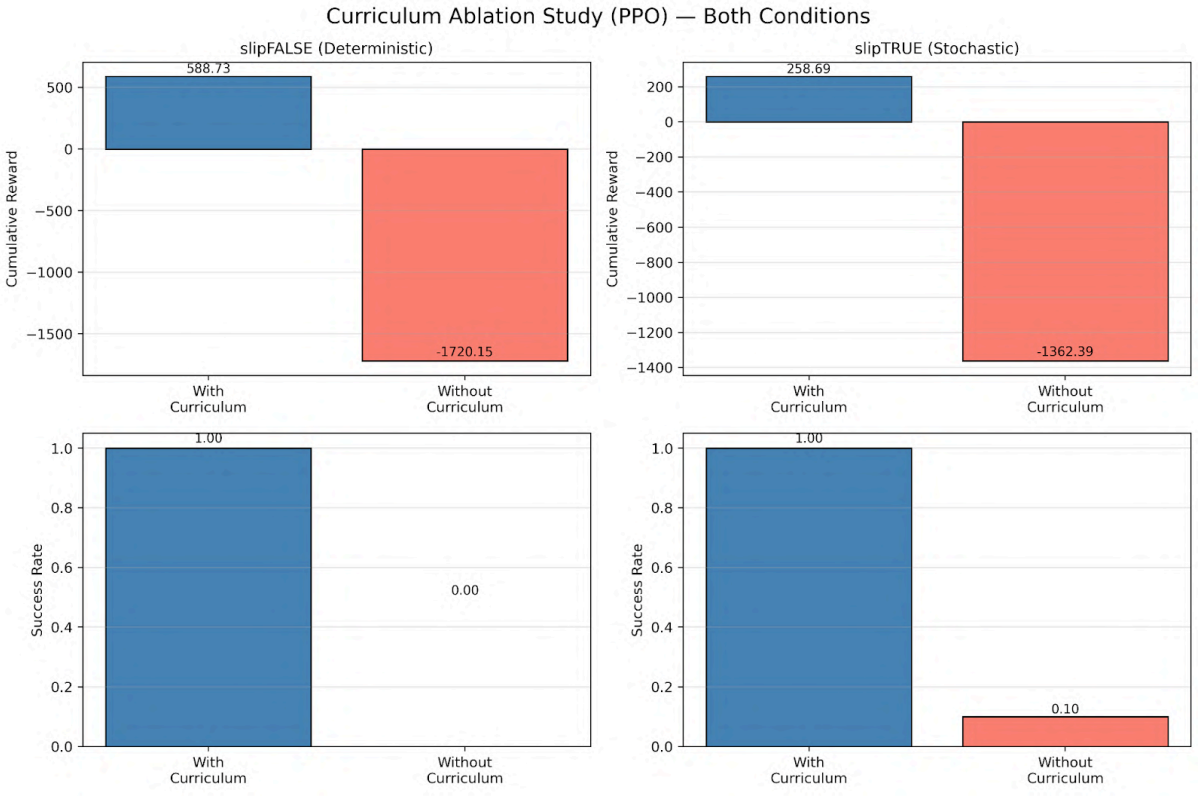


Figure 2: displays the cumulative reward and success rate for PPO with and without curriculum under both deterministic and stochastic conditions, illustrating the substantial performance gap between the two configurations.

Therefore, all subsequent comparisons and evaluations in this study refer exclusively to PPO with curriculum learning.

3.3 Evaluation factors

To ensure statistical reliability, all reinforcement learning experiments were repeated over 10 independent random seeds (seed 0 to seed 9), and results are reported as the mean and standard deviation with 95% confidence intervals across runs.

To compare the performances of all routing methods under identical conditions, a final evaluation was conducted after training. Each method BFS, Q-learning, and PPO was evaluated using the following metrics:

- Total Reward: calculated as the cumulative reward obtained over the episode.
- Number of steps: steps taken from the start to the goal (less than 1000 steps).
- Average snow cell visited: Snow visits were calculated by counting the number of snow-covered cells traversed in each episode and averaging this value across all evaluation episodes.
- Success rate: defined as the percentage of episodes in which the agent reached the goal within the maximum step limit.

The BFS baseline was evaluated only once on the fixed grid map, as it produces a deterministic path. Reinforcement learning agents were evaluated over 200 episodes per run, with experiments repeated over 10 independent random seeds for each condition (slip and non-slip). All methods were compared using the same energy-aware reward function and environment configuration to ensure fairness.

3.4 Hyperparameter Selections

Hyperparameters are selected based on standard practices and empirical validation to ensure stable and consistent performance [3,7].

3.4.1 Q-Learning Hyperparameter Configuration

We implement tabular Q-learning with an ε-greedy exploration strategy.

The hyperparameters are set as follows:

- Learning rate $\alpha = 0.15$
- Discount factor $\gamma = 0.99$
- Exploration strategy: ε-greedy
- Initial $\varepsilon = 0.60$
- Minimum $\varepsilon = 0.05$
- Exponential decay rate = 0.9995
- Training episodes = 15,000
- Maximum steps per episode = 1,000

The learning rate ($\alpha = 0.15$) is selected to balance convergence speed and stability. A moderate learning rate allows the agent to adapt efficiently while avoiding oscillations in value updates.

The discount factor ($\gamma = 0.99$) emphasizes long-term rewards, which is essential for navigation tasks where the objective is to reach the goal efficiently over multiple steps.

An ε-greedy exploration strategy is adopted to balance exploration and exploitation. The initial exploration rate ($\varepsilon = 0.60$) encourages sufficient exploration in early training, while exponential decay gradually shifts the policy toward exploitation. The minimum ($\varepsilon = 0.05$) ensures that some level of exploration is maintained throughout training, preventing the agent from getting stuck in suboptimal policies.

The number of training episodes 15,000 and maximum steps per episode 1,000 are chosen to ensure sufficient interaction with the environment for convergence, while maintaining computational efficiency.

During evaluation, a fully greedy policy ($\varepsilon = 0$) is used to assess the learned policy performance.

3.4.2 PPO Hyperparameter Configuration

We implement PPO using a clipped surrogate objective with generalized advantage estimation (GAE).

The hyperparameters are set as follows:

- Discount factor $\gamma = 0.99$
- GAE parameter $\lambda = 0.95$
- Learning rate = 3×10^{-4}
- Clipping range $\varepsilon = 0.2$
- Rollout buffer size = 2048 steps
- Batch size = 256
- Optimization epochs per update = 10
- Entropy regularization is enabled

The policy and value networks share a neural architecture consisting of two hidden layers with 256 units each and ReLU activation.

The discount factor ($\gamma = 0.99$) is selected to emphasize long-term rewards, which is essential for navigation tasks. The GAE parameter ($\lambda = 0.95$) provides a balance between bias and variance in advantage estimation, leading to more stable learning.

The clipping parameter ($\varepsilon = 0.2$) constrains policy updates to prevent large deviations from the previous policy, which improves training stability. The rollout buffer size and number of

optimization epochs are chosen to ensure sufficient policy updates while avoiding overfitting to recent trajectories.

Entropy regularization is included to encourage exploration and prevent premature convergence to suboptimal policies.

Overall, these hyperparameters follow widely adopted settings in reinforcement learning literature and are chosen to ensure stable and consistent training rather than aggressive performance tuning.

A complete summary of environment settings and hyperparameters is provided in Appendix A.

---

4. Results

The experiments compare BFS, Q-learning, and PPO on the same 50×50 winter grid, looking at both non-slip (deterministic) and slip (stochastic) cases.

4.1 Evaluation Setup and Metrics

The grid layout, start and goal positions, and reward parameters were kept exactly the same across all methods. All experiments were repeated over 10 independent random seeds, and results are reported as the mean ± standard deviation with 95% confidence intervals across runs. Trajectory figures shown are from the representative seed whose performance was closest to the mean. For each run, total reward, number of steps, average snow cells crossed per episode, and success in reaching the goal were recorded.

4.2 Trajectory Visualizations and Quantitative Results

4.2.1 Deterministic (Non-Slip) Winter Condition

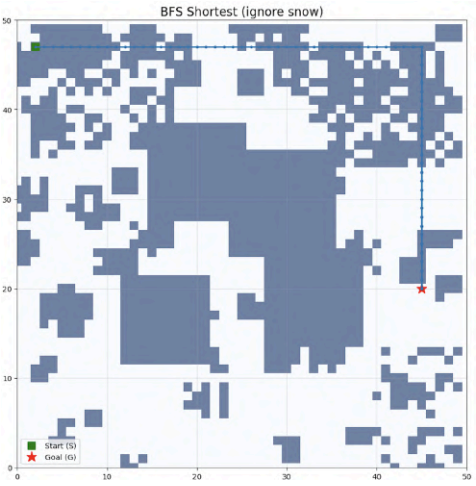


Figure 3: illustrates the BFS path under non-slip conditions. Since BFS always finds the shortest route, the trajectory goes straight from the start to the goal with minimal cells visited without any deviation.

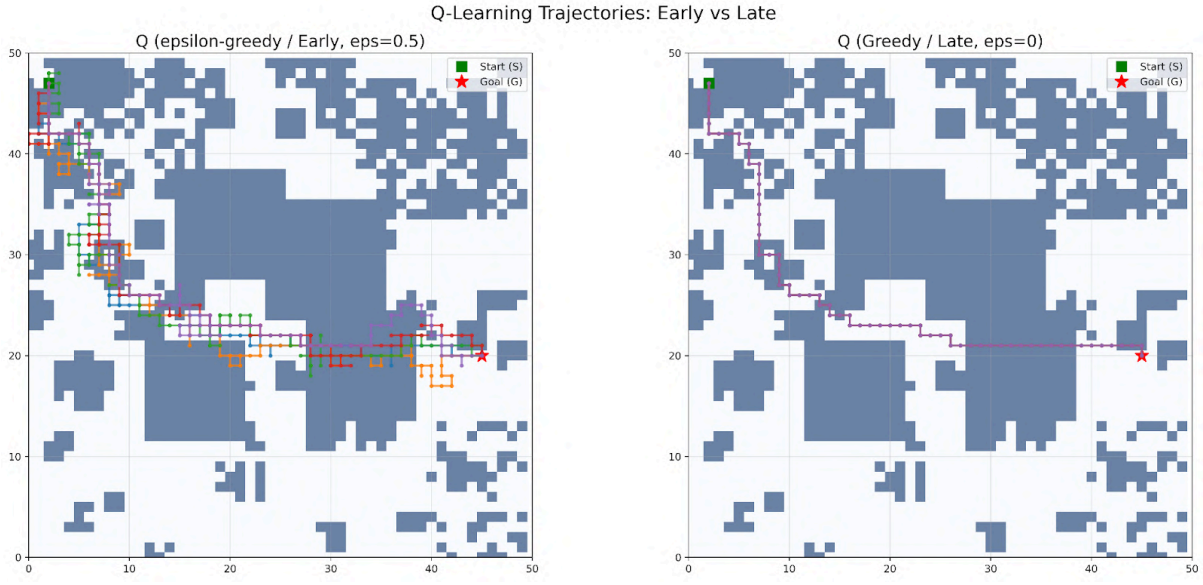


Figure 4: illustrates the navigation trajectories produced by the Q-learning agent under deterministic (non-slip) winter conditions during early training with an ε-greedy policy and late training with a greedy policy. The trajectories demonstrate the transition from exploratory behavior to a stable path toward the goal.

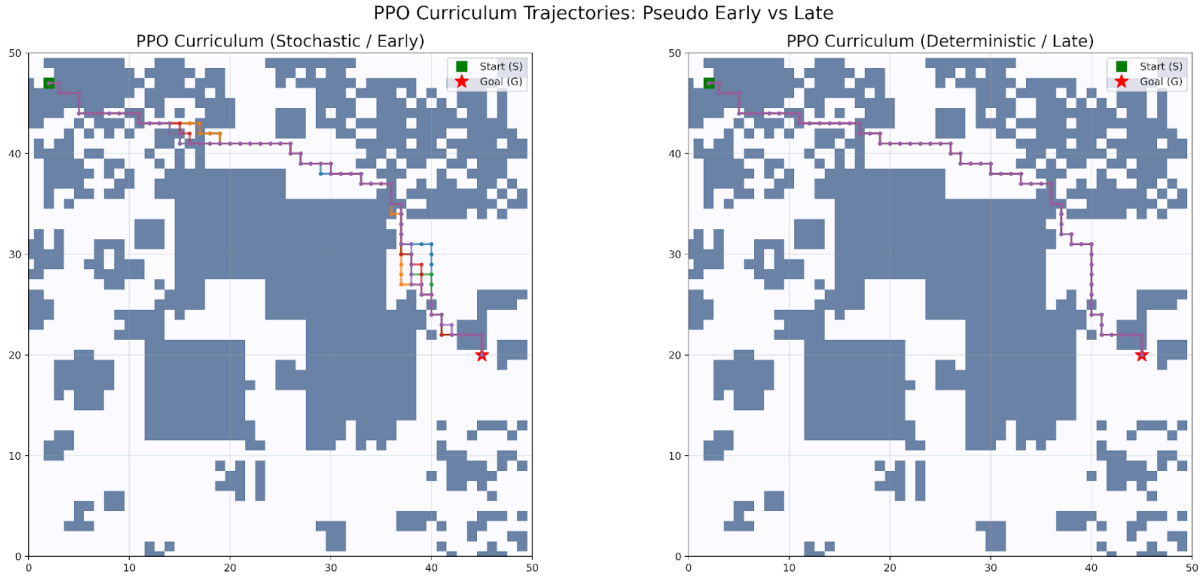


Figure 5: illustrates the navigation trajectories produced by the PPO agent during early and late stages of training under deterministic (non-slip) winter conditions.

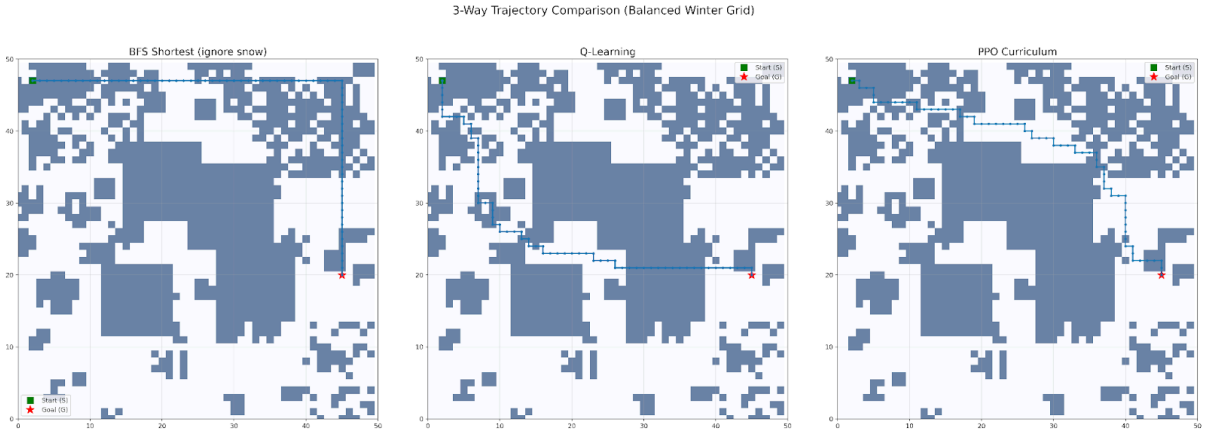


Figure 6: compares the trajectories of BFS, Q-learning and PPO in the non-slip winter setting.

=== Final Comparison Table ===

Method	Reward (mean± SD)	95%CI	Steps	Snow Visits	Success
BFS Shortest	575.30 ± 0.00	[575.30, 575.30]	70.00 ± 0.00	31.00 ± 0.00	100.0%
Q-Learning	582.01 ± 6.97	[577.02, 587.00]	70.00 ± 0.00	13.10 ± 3.05	100.0%
PPO Curriculum	588.73 ± 0.67	[588.25, 589.21]	70.00 ± 0.00	7.60 ± 0.97	100.0%

Table 1: summarizes the main performance metrics for all three methods under deterministic conditions, including total reward, steps to goal, average snow cells crossed, success rate, Standard Deviation (SD), and averaged across 10 independent random seeds with 95% confidence intervals.

4.2.2 Slip Winter Condition

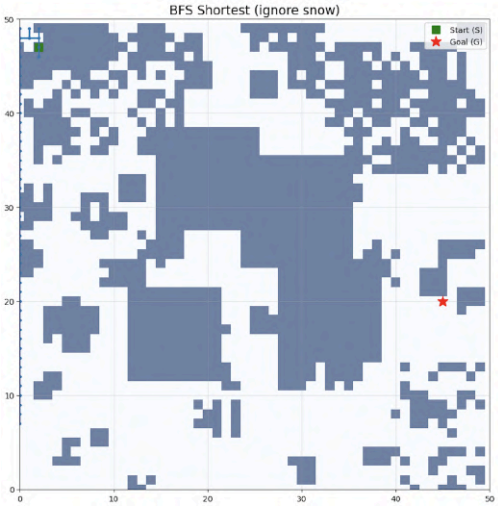


Figure 7: illustrates the navigation path produced by the BFS baseline under slip (stochastic) winter conditions.

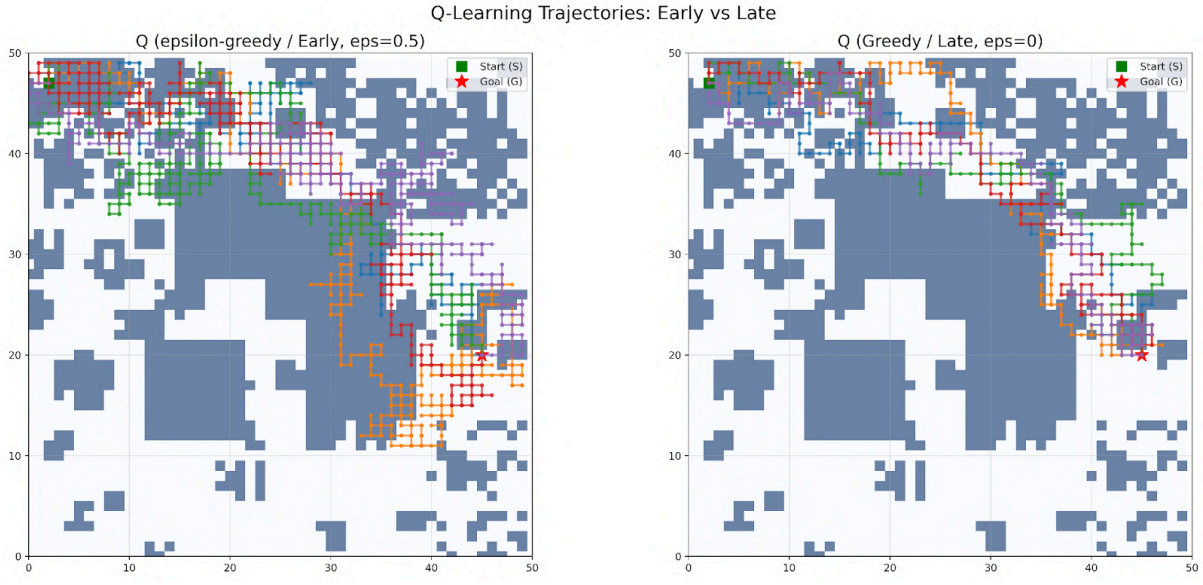


Figure 8: illustrates the navigation trajectories produced by the Q-learning agent during early and late stages of training under slip (stochastic) winter conditions.

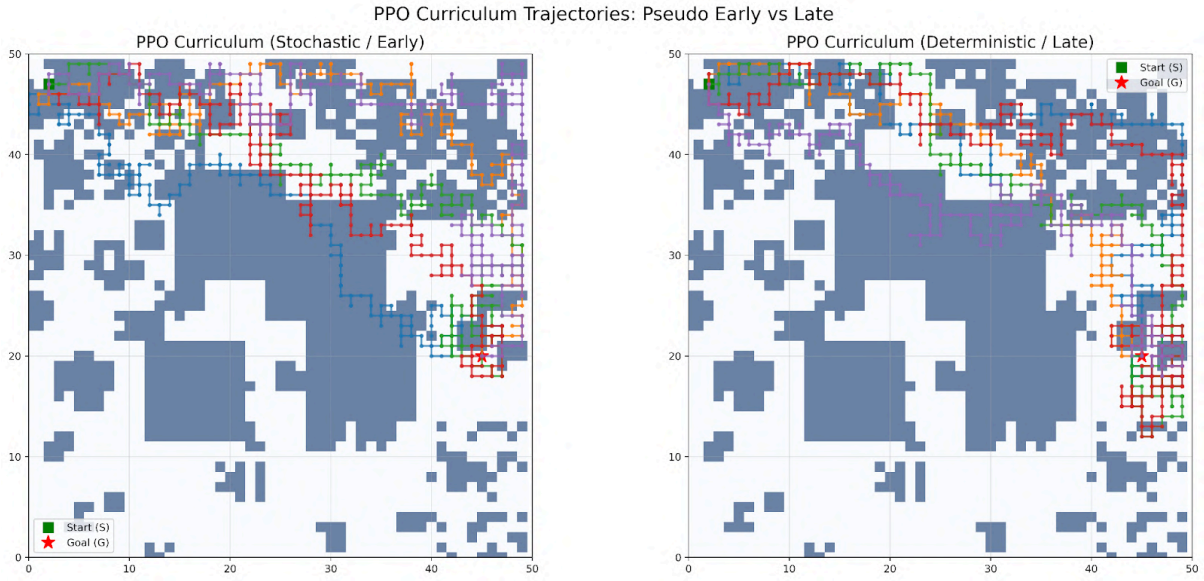


Figure 9: illustrates the navigation trajectories produced by the PPO agent during early and late stages of training under slip (stochastic) winter conditions.

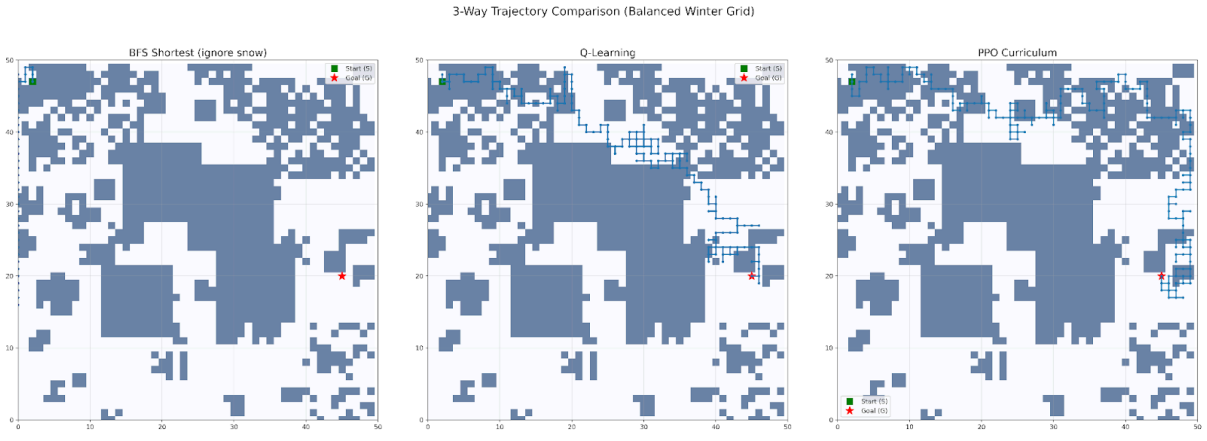


Figure 10: compares the navigation trajectories of BFS, Q-learning, and PPO under slip winter conditions.

=== Final Comparison Table ===

Method	Reward (mean±SD)	95% CI	Steps	Snow Visits	Success
BFS Shortest	-1668.41 ± 113.42	[-1749.54, -1587.28]	1000.00 ± 0.00	246.80 ± 76.09	0.0%
Q-Learning	311.82 ± 4.83	[308.37, 315.27]	26.40 ± 3.92	66.94 ± 3.87	100.0%
PPO Curriculum	258.69 ± 10.73	[251.02, 266.37]	241.18 ± 4.11	91.88 ± 5.24	100.0%

Table 2: summarizes the quantitative performance metrics for BFS, Q-learning, and PPO under slip winter conditions, using the same evaluation metrics as in the deterministic case, Standard Deviation (SD), and averaged across 10 independent random seeds with 95% confidence intervals.

4.3 Spatial State Visitation Heatmaps

Spatial state visitation heatmaps offer a grid-based way to see how often an agent passes through each cell during navigation. In our 50×50 winter grid, each cell corresponds to a state, and the color intensity reflects visitation frequency across episodes. Darker areas indicate cells that are visited more frequently, while lighter or white cells indicate rarely or never visited locations. Such visualizations give a clear picture of the agent's overall path distribution and behavior patterns. We generated these heatmaps using aggregated visitation counts normalized to the range between 0 and 1.

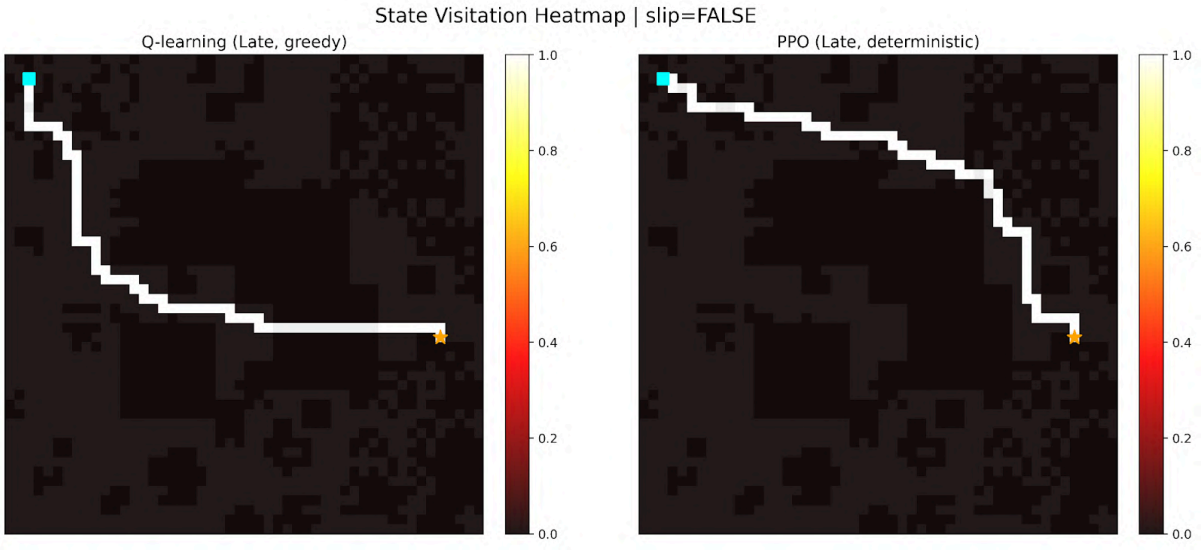


Figure 10: displays the visitation heatmaps for Q-learning and PPO under deterministic (non-slip) winter conditions. The results are aggregated from 10 evaluation runs consisting of 200 episodes per run. Both agents concentrate most visits along a narrow corridor connecting the start to the goal, with near-maximum intensity (≈1.0) along the primary route and very low visitation elsewhere.

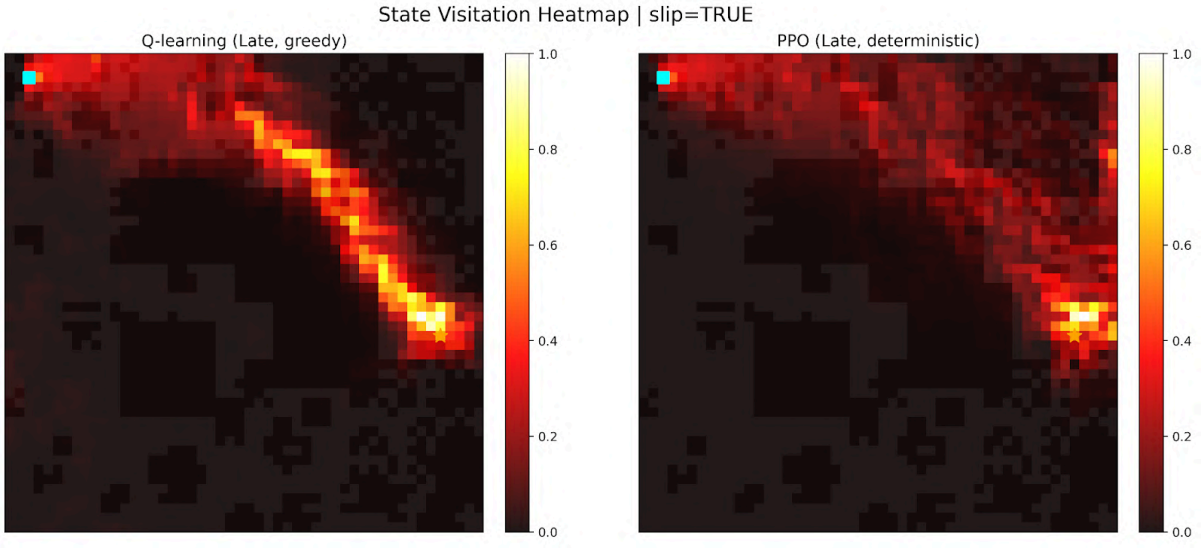


Figure 11: shows the visitation intensity for each grid cell in the slip case. Compared with the non-slip condition, visitation is more widely distributed across the grid rather than remaining within a narrow corridor. The heatmap aggregates data from 10 evaluation runs consisting of 200 episodes per run.

4.4 Statistical Analysis and Learning Curves - Slip Condition

4.4.1 Statistical Significance (Welch's t-test)

To assess statistical significance, a Welch's t-test was conducted comparing the cumulative reward of Q-learning and PPO under slip conditions across 10 seeds. The results confirmed that Q-learning significantly outperformed PPO in cumulative rewards ($t = 14.28$, $p < 0.001$), indicating that the performance gap is highly unlikely to be due to random variation. Q-learning achieved a mean reward of 311.82 ± 4.83 compared to PPO's 258.69 ± 10.73. The higher standard deviation of PPO (SD = 10.73) compared to Q-learning (SD = 4.83) further indicates greater training instability under stochastic conditions.

The standard deviation gap between PPO in both conditions suggests that PPO is more sensitive to stochastic signals under slip condition, where high-probability slip results in noisy advantage estimation, leading to higher variance across training runs (SD = 10.73). In contrast, under deterministic conditions PPO exhibits a much lower variance

(SD = 0.67), confirming that the instability is determined by the stochastic environment, not the algorithm itself.

=== Final Comparison Table ===

Metric	t-statistic	p-value	Q mean ± SD	PPO mean ± SD
Cumulative Reward	14.28	< 0.001	311.82 ± 4.83	258.69 ± 10.73
Success Rate	n/a	n/a	100% ± 0%	100% ± 0%

Table 3: Welch's t-test results comparing Q-learning and PPO Curriculum cumulative reward under slip conditions across 10 seeds. n/a indicates that the test was not applicable due to zero variance in success rate for both methods.

4.4.2 Learning Curves

Learning curves illustrate the performance of Q-learning and PPO evolves over the course of training under stochastic conditions, and provide insight into convergence behaviour, training stability, and the impact of curriculum learning on PPO.

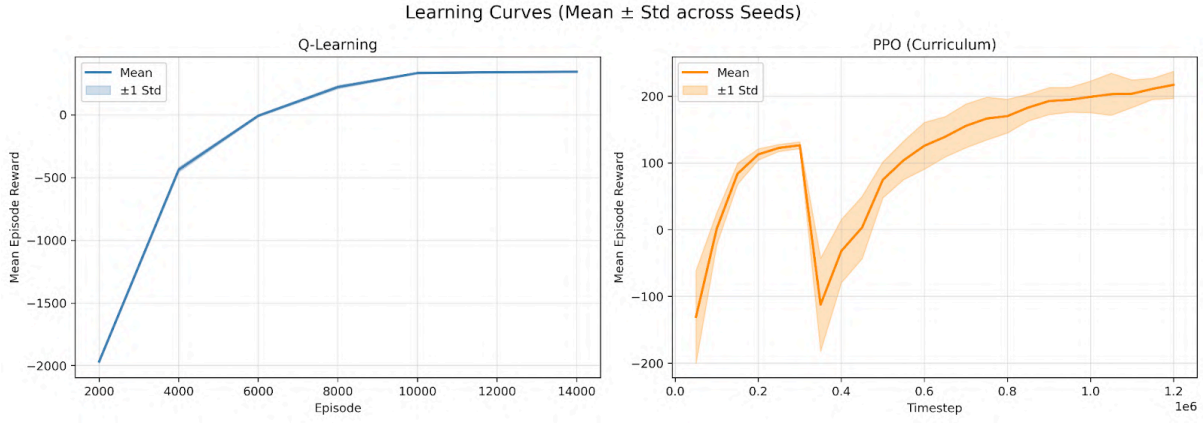


Figure 13: Learning curves (Mean ± SD across 10 seeds) under slip (stochastic) conditions. Left: Q-learning mean episode reward vs. training episode (15,000 episodes) showing smooth, steady and consistent convergence. Right: PPO Mean episode reward vs. training timestep (1,200,000 steps), where the pronounced drop around timestep 300,000 corresponds to the Phase 1 to Phase 2 curriculum transition, when energy-aware snow penalties are introduced and the agent must re-adapt its policy. After the transition is complete, PPO gradually recovers and converges, but its variance band is significantly wider compared to Q-learning.

4.5 Limitations & Uncertainty

Several sources of uncertainty and limitations appeared in this study. Both Q-learning and PPO showed noticeable performance differences across training runs, which was expected given the stochastic nature of the environment and the learning algorithms. To account for this variability, all experiments were repeated over 10 independent random seeds, and results are reported as the mean across runs.

Under slip conditions, the same policy could produce quite different routes from episode to episode because actions did not always lead to the expected outcome. In addition, stochastic exploration strategies and random initialization of value functions (or policy networks) exposed agents to slightly different states, transitions, and rewards across runs. This variability made it harder to get perfectly consistent results.

An unexpected issue was that the BFS baseline failed to reach the goal within the 1,000-step limit under slip conditions, resulting in a number of unsuccessful evaluation episodes.

A full numerical breakdown of evaluation metrics is provided in Appendix B.

5.Discussion

5.1 Restatement of Hypothesis and Summary of Findings

The study hypothesized that reinforcement learning-based agents, particularly Proximal Policy Optimization (PPO), would achieve more energy-efficient winter navigation paths than a traditional shortest-path baseline and a value-based Q-learning agent under both deterministic and slip winter conditions. The averaged results partially supported the hypothesis. Under deterministic (non-slip) conditions, PPO achieved higher cumulative reward, and fewer snow visits, outperforming Q-learning, which supports the hypothesis. However under stochastic (slip) conditions, Q-learning outperformed PPO in terms of cumulative reward or snow-cell avoidance, which did not support the hypothesis. Both reinforcement learning methods perform better results than traditional BFS across these environments.

5.2 Interpretation of Q-Learning and PPO Performance

The results showed that algorithm performance varied depending on the environmental condition. Under deterministic (non-slip) conditions, PPO achieved higher cumulative reward and fewer snow cell visits than Q-learning, suggesting that PPO's policy gradient approach was able to learn a more energy-efficient route in a stable environment. Under stochastic (slip) conditions, Q-learning outperformed PPO in terms of cumulative reward and snow-cell avoidance. One possible explanation was that a discrete and clear grid-world environment and relatively small environment would favor more for Q-learning, as it can perform exact value iteration over the finite MDP (Markov Decision

Process), directly converging to the true optimal Q-values for each state without approximation error. Therefore allowing Q-learning to efficiently estimate optimal state and gives lower variance than policy gradient method (PPO), since it updates based on individual state-action pairs rather than entire trajectories [7,8]. In contrast, PPO relied on function approximation and stochastic policy updates, which may have required more training data or longer training time to converge to an equally optimal policy under slip conditions. Additionally, its stochastic policy may have introduced suboptimal choices that were not effective, whose gradient estimates already carry high variance from episode training. Under slip conditions, environmental stochasticity further interrupts reward signals, making them sparse and noisy, which destabilizes gradient estimates and requires significantly more training samples. Therefore reducing its total rewards relative to Q-learning under stochastic conditions [8].

5.3 Effects of Slip (Stochastic) Winter Conditions

Under slip (stochastic) conditions, all agents showed broader trajectory dispersion and increased visits to previously visited states because of slipping, as reflected in the spatial state visitation heatmaps. This behavior is consistent with probabilistic transition dynamics, in which the same action could lead to different movement outcomes across episodes, such as deviating left in one run and right in another. When the grid is slippery, agents don't follow one stable path anymore. As a result, the routes spread out, and their performance becomes less consistent across episodes. These observations are consistent with prior navigation studies showing that stochastic environments increase policy variance and reduce convergence stability in reinforcement learning tasks [4,8].

5.4 Interpretation of BFS Baseline Behavior

The Traditional baseline trajectories did not take account for winter grid costs or stochastic transitions, it focused on the fastest way to the destination. In non-slip conditions this approach naturally produced the optimal route. However, under slip conditions, BFS exceeded the maximum step limit in multiple evaluation episodes, reflecting its inability to adapt its policy in response to transition winter grid costs or stochastic transitions.

Because BFS always assumed deterministic movement, each slip caused deviations from the planned path. These deviations often led to inefficient loops and longer paths. Unlike reinforcement learning methods, BFS did not have reward feedback or policy updates, which limited its ability to adjust navigation behavior under changing environmental dynamics. Therefore, making it the least effective method.

5.5 Hyperparameter Sensitivity Analysis

To assess the robustness of the reported conclusions to hyperparameter choice, a sensitivity analysis was conducted to independently vary each hyperparameter of Q-learning and PPO under slip conditions using a single representative seed. For Q-learning, the learning rate (alpha) and epsilon decay were evaluated to identify which

hyperparameter values lead to greater stability and performance. For PPO, the clip range and learning rate were varied to assess how different hyperparameters affect training stability and sensitivity to stochastic conditions.

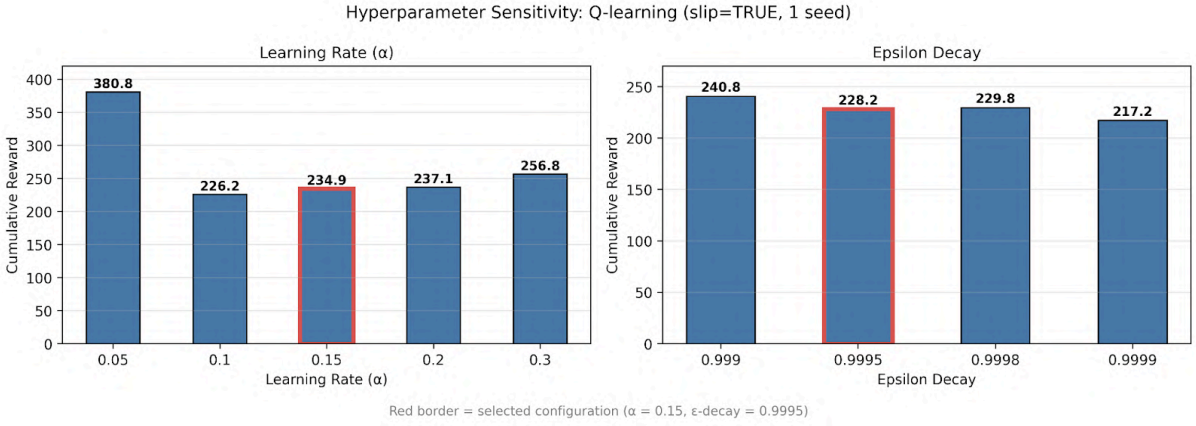


Figure 14: Comparison of different learning Rate (alpha) values and epsilon decay values . The highlighted values represent the selected hyperparameter configuration, chosen based on their balance of performance and training stability. (α = 0.15, ε-decay = 0.9995)

=== Q-learning hyperparameter values Final Comparison Table ===

Parameter	Value	Success rate (%)	Cumulative reward	Note
Learning rate (alpha)	0.05	98.00	380.80	
Learning rate (alpha)	0.10	100.00	226.20	
Learning rate (alpha)	0.15	100.00	234.90	Selected configuration
Learning rate (alpha)	0.20	100.00	237.10	
Learning rate (alpha)	0.30	100.00	256.80	
Epsilon decay	0.9990	100.00	240.80	
Epsilon decay	0.9995	100.00	228.20	Selected configuration
Epsilon decay	0.9998	100.00	229.80	
Epsilon decay	0.9999	100.00	217.20	

Table 4: presents a sensitivity analysis of Q-learning's key hyperparameters. Cumulative rewards are reported to assess overall performance.

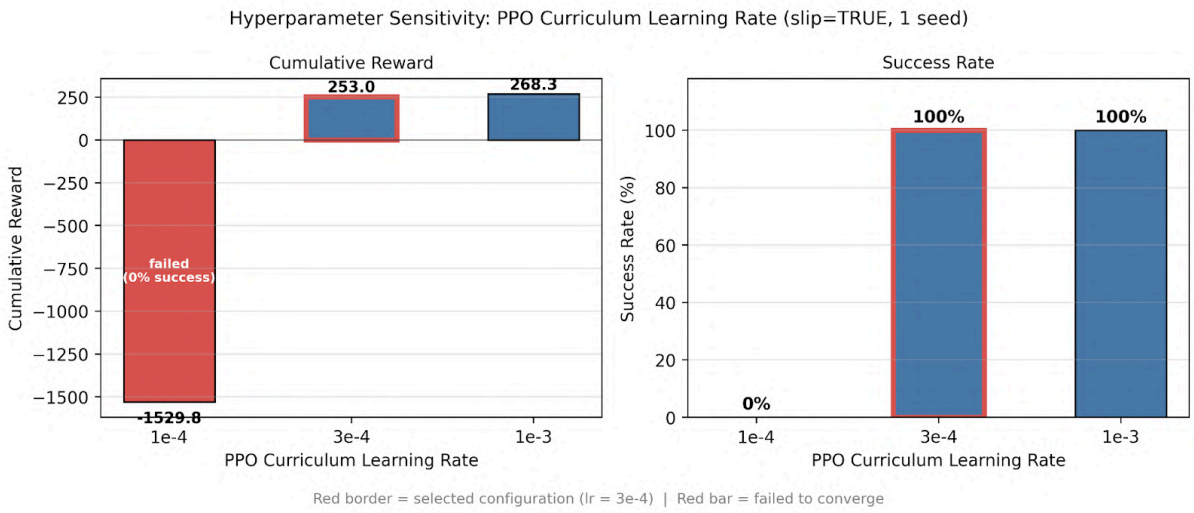


Figure 15: Comparison of different values of learning rate and the highlighted values represents the selected hyperparameter configuration, chosen based on their balance of performance and training stability (Learning rate = 3×10^{-4}). Lr = 1×10^{-4} failed to converge (0% success rate, reward = -1529.8), a very small learning rate leads to insufficient policy updates, preventing convergence within the training horizon.

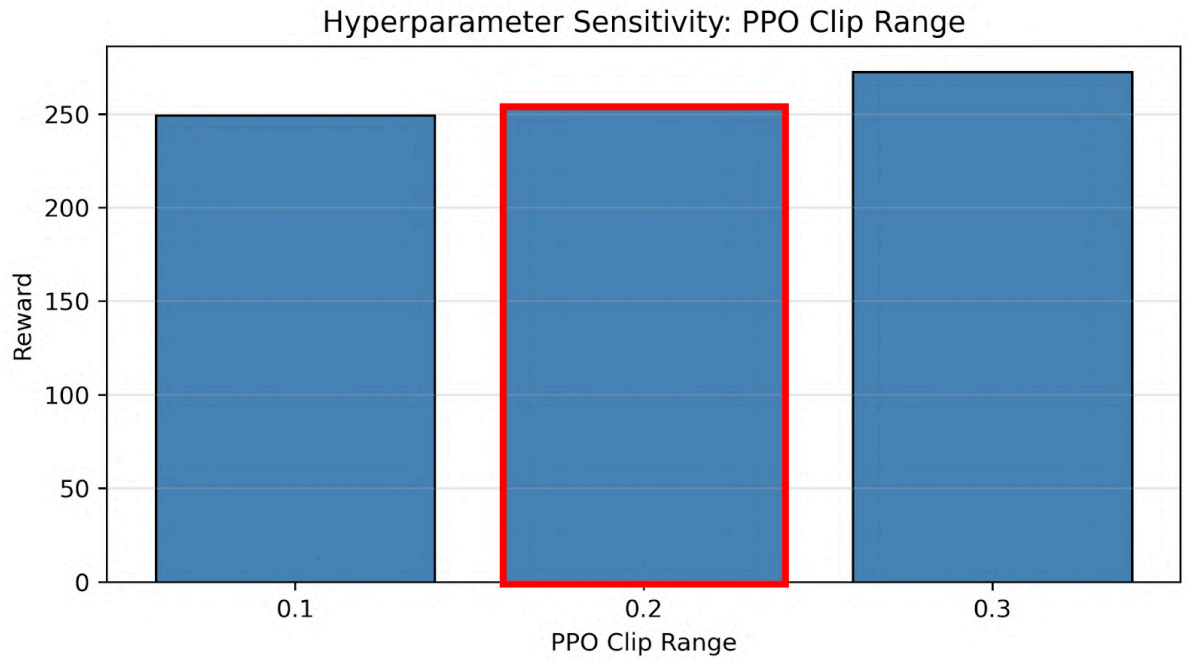


Figure 16: Comparison of different values of PPO clip range and the highlighted values represents the selected hyperparameter configuration, chosen based on their balance of performance and training stability. (Clip range = 0.2)

===PPO hyperparameter values Final Comparison Table ===

Parameter	Value	Success rate (%)	Cumulative reward	Note
Clip range	0.1	100.00	249.40	
Clip range	0.2	100.00	253.00	Selected configuration
Clip range	0.3	100.00	272.50	
Learning rate	1e-04	0.00	−1529.80	
Learning rate	3e-04	100.00	253.00	Selected configuration
Learning rate	1e-03	100.00	268.30	

Table 5: presents a sensitivity analysis of PPO's key hyperparameters. Cumulative rewards and success rates are reported to assess overall performance.

Across all tested configurations, the rank ordering of methods remained consistent, confirming that the reported conclusions are robust to moderate hyperparameter variation.

5.6 Limitations and Future Directions

Several limitations should be noted when interpreting these findings. All experiments were carried out in a simulated 50×50 grid environment with a relatively small and simple state space, which differs significantly from real-world scenarios. This setup likely favored value-based methods like Q-learning, since it could learn precise state-action values for every cell. In contrast, PPO depends on neural network function approximation and stochastic policy updates, which may need more training steps or data to perform well in such discrete, small-scale settings.

To improve statistical reliability, all experiments were run independently over 10 random seeds and results were averaged across runs. Noticeable performance variability was observed across training runs, mainly due to the stochastic environment, random action outcomes, and probabilistic transitions under slip conditions.

The simulated environment simplified real-world winter driving conditions and did not capture vehicle dynamics, sensor noise, road-surface variation, or realistic energy-consumption models, these are factors that might lead to changes. As a result, these findings to real-world winter navigation scenarios were limited [12].

Future research could evaluate reinforcement learning agents in larger scale, continuous, or more realistic winter navigation environments with higher-dimensional state spaces and more complex conditions. Then it may be better to reflect real-world uncertainty and could allow policy-gradient methods such as PPO to fully function their theoretical advantages in handling stochastic and high-dimensional control problems.

---

6. Conclusion

This study compared a traditional BFS baseline with two reinforcement learning methods—Q-learning and Proximal Policy Optimization (PPO)—for energy-efficient navigation in a 50×50 winter grid under both deterministic (non-slip) and stochastic (slip) conditions. The averaged results across 10 independent random seeds partially supported the original hypothesis. Under deterministic (non-slip) conditions, PPO achieved the highest cumulative reward and fewest snow cell visits, outperforming Q-learning. However, under stochastic (slip) conditions, Q-learning outperformed PPO in cumulative reward and snow-cell avoidance. Although both reinforcement learning methods clearly outperformed the BFS baseline, BFS struggled under slip conditions due to its inability to adapt to stochastic transitions.

These findings show that whether value-based methods or policy-gradient methods perform better really depends on how unpredictable the environment is. In our small, structured discrete grid, PPO gained an advantage from its stable gradients and curriculum learning when everything was deterministic. On the other hand, Q-learning's simple tabular updates turned out to be more robust when the roads

became slippery and actions sometimes failed. More generally, the results emphasize that the choice of reinforcement learning method should be guided by the structure and constraints of the task, rather than by theoretical advantages alone. Future work could explore whether PPO's advantages extend to more complex environments with continuous states or larger state spaces, where its policy-gradient approach may better demonstrate its theoretical strengths.

Reference

[1]Mao, R., Xu, W., Qian, Y., Li, X., Li, Y., Li, G., & Zhang, H. (2025). Understanding the Determinants of Electric Vehicle Range: A Multi-Dimensional Survey. *Sustainability*, 17(10), 4259. https://www.mdpi.com/2071-1050/17/10/4259

[2]Carlson, A., & Vieira, T. (2021). *The effect of water and snow on the road surface on rolling resistance* (VTI Report 971A). Swedish National Road and Transport Research Institute.https://www.researchgate.net/publication/350690120_The_effect_of_water_and_snow_on_the_road_surface_on_rolling_resistance

[3]Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction. https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf

[4]Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A. J., Banino, A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., Kumaran, D., & Hadsell, R. (2017). *Learning to Navigate in Complex Environments* (No. arXiv:1611.03673). arXiv. https://doi.org/10.48550/arXiv.1611.03673

[5]Tan, C. (2025). Comparative Study of Reinforcement Learning Performance Based on PPO and DQN Algorithms. *Applied and Computational Engineering*, 175(1), 30–36. https://doi.org/10.54254/2755-2721/2025.AST24879

[6]Warnakulasuriya, D. A., Plosila, J., & Haghbayan, H. (2025). Energy-Efficient Path Planning in Uneven Terrains Using Adaptive Reinforcement Learning. *IEEE Conference Publication*. https://ieeexplore.ieee.org/document/11093435

[7]Watkins, C.J.C.H., & Dayan, P(1992). Q-learning. *Mach Learn* 8, 279–292. https://doi.org/10.1007/BF00992698

[8]Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). *Proximal Policy Optimization Algorithms* (No.arXiv:1707.06347). arXiv. https://doi.org/10.48550/arXiv.1707.06347

[9]Pendyala, A., Atamna, A., & Glasmachers, T. (2024). Solving a Real-World Optimization Problem Using Proximal Policy Optimization with Curriculum Learning and Reward Engineering. arXiv:2404.02577.https://arxiv.org/abs/2404.02577
[10]Dayan, P., & Balleine, B. W. (2002). Reward, Motivation, and Reinforcement Learning. *Neuron*, 36(2), 285–298. https://doi.org/10.1016/S0896-6273(02)00963-7
[11]Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). Curriculum learning. *Proceedings of the 26th Annual International Conference on Machine Learning*, 41–48. https://doi.org/10.1145/1553374.1553380
[12]Chukwurah, N., Adebayo, A. S., Ajayi, O. O., & Anfo Pub. (2024). *Sim-to-Real Transfer in Robotics: Addressing the Gap between Simulation and Real- World Performance*. *International Journal for Multidisciplinary Research (IJFMR)*, 05(01), 33–39. https://doi.org/10.54660/.IJFMR.2024.5.1.33-39

Appendices

Appendix A Experimental Configuration and Algorithmic Details:

A.1 Environment Configuration

All experiments were conducted in a custom 50×50 winter grid environment (2,500 discrete states).

- Start position: (47, 2)

- Goal position: (20, 45)
- Maximum steps per episode: 1000
- Action space: {up, down, left, right}

Two transition settings were evaluated:

Deterministic (slip = FALSE)

The intended action is executed exactly.

Stochastic (slip = TRUE)

With probability $\frac{1}{3}$, the intended action is executed.
With probability $\frac{2}{3}$, the agent slips to a random adjacent direction.

All algorithms were evaluated on the same fixed grid layout to ensure fairness.

A.2 Reward Function

The reward function models energy-aware winter navigation.

At each time step:

- `Step penalty: -1.5`
- Snow penalties:
  - `Near snow: -0.2`
  - `Edge snow: -0.5`
  - `Core snow: -2.0`
- Goal reward: +700

The cumulative episode reward is:

$$R = \sum_{t=1}^{T} (-1.5 - C_{snow}(s_t)) + 700 \cdot 1_{goal\ reached}$$

where $C_{snow}(s_t)$ denotes the terrain penalty and
$1_{goal\ reached}$ indicates successful termination.

A.3 Q-Learning Configuration

Tabular Q-learning was implemented with the following hyperparameters:

- Learning rate α=0.15
- Discount factor γ=0.99
- Exploration strategy: ε-greedy
- Initial ε = 0.60
- Minimum ε = 0.05
- Exponential decay per episode = 0.9995
- Training episodes = 15,000
- Max steps per episode = 1000

The Q-update rule is:

$Q(s,a) \leftarrow Q(s,a) + \alpha[R + \gamma \max_{a'} Q(s',a') - Q(s,a)]$

During evaluation, a fully greedy policy (ε = 0) was used.

A.4 Proximal Policy Optimization (PPO) Curriculum Configuration

PPO was implemented as a stochastic policy-gradient method.

Total training timesteps: 1,200,000

- Phase 1 (navigation-focused reward): 300,000 timesteps
- Phase 2 (energy-aware reward): 900,000 timesteps

Evaluation was conducted using deterministic action selection.

PPO Objective Function

PPO optimizes the clipped surrogate objective:

$L^{CLIP}(\theta) = E_t[min(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)]$

where

$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$

and A^t is the generalized advantage estimate.

The clipping mechanism constrains policy updates to maintain stability.

PPO Hyperparameters

- Discount factor γ=0.99
- GAE parameter λ=0.95
- Entropy regularization enabled
- Learning rate: 3×10^{-4}
- Clip range: 0.2
- Rollout buffer: 2,048 steps
- Batch size: 256
- Optimization epochs: 10
- Neural network architecture: two hidden layers (256 units each, ReLU activation)

A.5 Evaluation Protocol

For each condition (slip FALSE / TRUE):

- One trained model per algorithm
- Maximum evaluation length: 1000 steps
- Number of evaluation episodes: 200
- Number of seeds: 10
- Heatmap runs: 10
- Metrics recorded:
    - Total cumulative reward
    - Number of steps
    - Snow cell visits
    - Success rate (%)

State visitation heatmaps were generated by aggregating visitation frequencies across evaluation runs of 200 episodes each.

Appendix B Detailed Quantitative Results:

B.1 Deterministic (slip = FALSE

Method	Reward (mean±SD)	95%CI	Steps	Snow Visits	Success
BFS Shortest	575.30 ± 0.00	[575.30, 575.30]	70.00 ± 0.00	31.00 ± 0.00	100.0%
Q-Learning	582.01 ± 6.97	[577.02, 587.00]	70.00 ± 0.00	13.10 ± 3.05	100.0%
PPO Curriculum	588.73 ± 0.67	[588.25, 589.21]	70.00 ± 0.00	7.60 ± 0.97	100.0%

All methods reached the goal within 70 steps.
PPO achieved the highest cumulative reward and fewest snow cell visits.

B.2 Stochastic (slip = TRUE)

Method	Reward (mean±SD)	95% CI	Steps	Snow Visits	Success
BFS Shortest	-1668.41 ± 113.42	[-1749.54, -1587.28]	1000.00 ± 0.00	246.80 ± 76.09	0.0%
Q-Learning	311.82 ± 4.83	[308.37, 315.27]	26.40 ± 3.92	66.94 ± 3.87	100.0%
PPO Curriculum	258.69 ± 10.73	[251.02, 266.37]	241.18 ± 4.11	91.88 ± 5.24	100.0%

Under stochastic dynamics:

- BFS fails due to inability to adapt.
- Q-learning demonstrates greater robustness.
- PPO maintains success but exhibits higher trajectory dispersion.

B.3 State Visitation Analysis

State visitation frequency was computed as:

$$V(s) = \frac{N(s)}{max_s N(s)}$$

where $N(s)$ is the number of visits to state s.

- Under deterministic conditions, visitation concentrates along a narrow corridor.
- Under stochastic conditions, visitation becomes more dispersed.
- Q-learning exhibits more focused routing than PPO under slip conditions.

Energy-Efficient Path Planning in Winter Conditions: A Comparative Study of Traditional Baseline, Q-learning, and Proximal Policy Optimization in a Grid World Environment

1.Abstract

Recent investigation in winter route planning has become increasingly important due to increased resistance and reduced efficiency on winter roads. These conditions significantly increase energy consumption and introduce uncertainty, raising the risk of failing to reach the destination. To address the challenge, Reinforcement Learning (RL) is a type of machine learning where an agent learns to make decisions by interacting with an environment, receiving rewards or penalties for its actions to maximize cumulative rewards. For simulation, we benchmark a standard shortest-path algorithm (BFS) against two reinforcement learning methods: Q-learning and Proximal Policy Optimization (PPO). All approaches are tested on identical grid configurations, with the same starting points, goals, and reward functions. We assess their performance based on cumulative reward, snowy cell visits (as a proxy for energy cost), trajectory characteristics, and success rate. It is hypothesized that PPO algorithms will result in lower energy consumption than Q-learning and traditional baseline under winter conditions. The results show that while BFS consistently finds the shortest paths, it fails to consider energy costs or environmental uncertainty. RL agents, by comparison, adapt more efficiently to winter conditions. Under deterministic (non-slip) conditions, PPO achieved higher cumulative reward and fewer snow cell visits than Q-learning, partially supporting the hypothesis. However, under stochastic (slip) conditions, Q-learning outperformed PPO in cumulative reward and snow-cell avoidance. These results suggest that Q-learning is better suited for stochastic winter navigation in grid worlds, while PPO may perform better in more deterministic environments with continuous states or decisions.

Key words: Energy-Efficient, Traditional Baseline, Q-learning, Proximal Policy Optimization, grid world, reinforcement learning.

2.Introduction

As demand in Eelectric Vvehicles (EV) become increasingly widespread, energy efficiency and route optimization have emerged as critical challenges, particularly in winter when resistance increases. Sub-zero temperatures, combined with snow and ice accumulation on road surfaces, significantly increase energy consumption, resulting in reduced driving range and greater unpredictability in trip planning. ~~Freezing temps, plus snow and ice covering the roads, push energy consumption way up, which means shorter range and trips that feel less predictable.~~ A 2025 study shows that an estimated

50 % of EV driving range can be reduced in cold climates, including snow and ice covering terrain, highlighting the significant impact of environmental conditions on energy consumption [1]. Furthermore, when snow or water remains on the road surface, vehicle tires must continuously displace through ice as they roll and move, hence forcing the vehicle to draw more power [2]. On top of that, water cools tires more effectively than air alone, altering their mechanical properties and further pushing rolling resistance higher. Previous studies have shown that rolling resistance can rise by approximately 30% to 40% under wet or snowy conditions [2]. Ultimately, these factors make winter driving a major concern for Electric Vehicle (EV) efficiency and reinforce the need for energy-aware routing strategies.

To approach these challenges, we have explored a few computational methods to allow agents to learn effective navigation strategies aimed at minimizing energy consumption. In this study, a 50x50 grid is used as an abstraction routing map that circles key structures of Canadian snow distribution, rather than exact geographical terrain. Grid-based environments are commonly used in reinforcement learning studies, as they simplify complex continuous real-world environments into discrete states [35]. This abstraction is particularly well-suited for this study, as it enables environmental factors such as snow coverage and surface slip conditions to be incorporated directly into the cell-level cost and reward structure, supporting controlled and reproducible comparison of different routing methods.

Despite growing interest in reinforcement learning for navigation, existing studies have largely overlooked the impact of winter environmental conditions, such as snow coverage and surface slippiness, on energy-aware path planning. Prior work has primarily focused on general navigation without taking account of explicitly modeling weather-dependent energy costs [48]. To address this gap, this study integrates snow coverage, stochastic slip condition, and energy costs into a comparative framework, evaluating Q-learning, PPO, and a traditional bBaseline under realistic winter routing scenarios. This study is one of the first to directly compare value-based and policy-gradient reinforcement learning methods in an energy-aware winter navigation setting.

Prior work on reinforcement learning for navigation can be categorized into three directions. First, Grid world benchmarks commonly adopt tabular Q-learning and deep reinforcement learning variants as standard baselines for evaluating discrete navigation tasks [3,5], but focus primarily on task completion rather than energy awareness and energy sensitive routing. Second, cost-aware navigation research has examined energy minimisation in uneven terrains using reinforcement learning [6], yet rarely incorporated weather dependent components, such as snow coverage or stochastic traction loss. Third, existing comparisons between value-based and policy-gradient methods [7,8], including curriculum based PPO applications [9] are conducted primarily under fixed and stable conditions, without examining the possibility of a particular

reinforcement learning method remaining dominant to another when action outcomes become probabilistic, as in stochastic environments such as winter slip conditions. The work in this study addresses all three gaps simultaneously by implementing an energy-aware reward structure into a reproducible 50×50 grid-world benchmark, embedding snow-density penalties and a 2/3-slip probability to simulate realistic winter routing conditions, and directly comparing Q-learning, PPO, and a traditional baseline under both deterministic and stochastic environments.

More specifically, the research will include a traditional baseline algorithm, and two types of reinforcement learning, which are Q-learning and PPO algorithm. The traditional baseline routing computes the shortest path based on static cost metrics and follows the predefined route without adaptation or learning (basic routing). Q-learning, a value-based reinforcement learning method, learns through incremental dynamic programming processes with computational requirements and it is well suited for agents to improve and refine action values and achieve effective performance in controlled Markovian domains [73]. In contrast, PPO is a policy-gradient algorithm that primarily optimizes a stochastic policy and achieves greater training stability through constrained policy updates with a clipped surrogate objective function [84].

This comparison evaluates the effectiveness of the proposed routing methods in identifying energy-efficient paths. Performance is evaluated through energy related costs, overall routing efficiency, and observed navigation behavior under both deterministic (non-slip) and stochastic (slip) settings. Because PPO balances stable policy updates with adaptive learning in uncertain environments, it is hypothesized to be the most effective method implemented for winter routing.

The following section goes into the methodology and experimental setup in more detail.

3.Methods

3.1 Environment Design

A custom winter navigation grid environment was designed to evaluate each Reinforcement Learning (RL) agent under stochastic and cost-sensitive conditions. The environment consists of a 50 x 50 grid world containing 2500 cells, including both normal terrain and snow-covered terrain. The simulation contains a fixed start (47, 2) and a fixed goal (20, 45) point, and are located in the top left corner and middle right respectively. At each time step, the agent can take either one of the four directions, up, down, left, and right; each step on a non-snow grid has an energy cost of -1.5 per step. Snow is modeled as spatially correlated regions with graded intensity, capturing the typical uneven accumulation patterns in Canadian winters.

When snow is modeled as a binary state, abrupt reward discontinuities make PPO's advantage estimates unstable, resulting in increased gradient variance. By contrast, a continuous representation of snow thickness provides smoother cost transitions, which help reduce gradient variance and stabilize learning, making it more efficient. These snow types are classified as "Near snow", "Edge snow", and "Core snow".

To evaluate robustness under different winter conditions, two transition settings were considered using the same grid map: a non-slip (deterministic) condition and a slip (stochastic) condition. In the deterministic setting, action always results in the intended movement, and snow cells only add additional energy loss. In the stochastic setting, actions do not always lead to the intended outcome: with a probability of 1/3, the intended action is executed, and with a probability of 2/3, the agent moves in a random adjacent direction, simulating loss of traction control during winter driving on ice and snow.

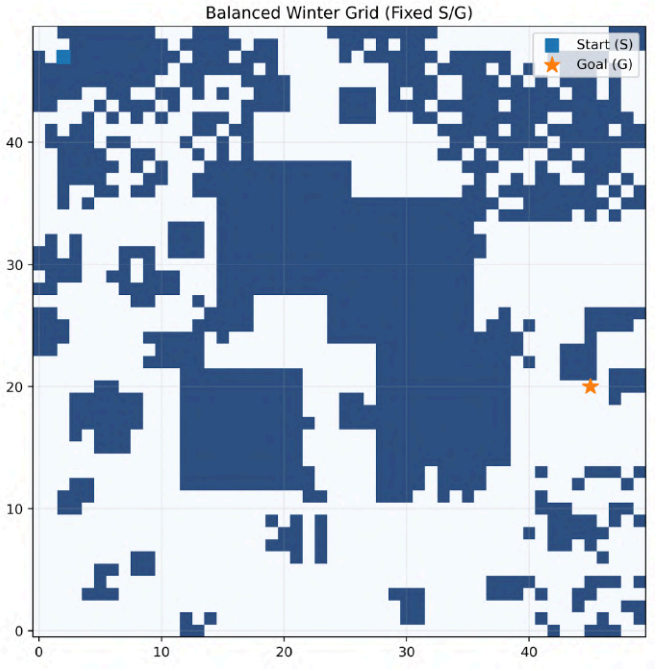


Figure 1: shows a typical winter grid layout used in all experiments.

3.2 Reinforcement Learning Algorithms/ Methodologies:

The routing strategies and learning methods used in this study are described in this section.

3.2.1 Traditional Baseline (BFS)

Traditional baseline is a non-learning shortest path strategy used as a comparison base for reinforcement learning methods. It operates on the 50x50 grid map, it considers four directions to move (up, down, left, right). The system calculates the shortest path to the destination, without considering energy savings, which means it does not adapt to environmental feedback or uncertainty.

3.2.2 Q-learning Implementation

Tabular Q-learning was used as the value-based reinforcement learning baseline. The agent maintains a discrete Q (s, a) table, which is updated iteratively based on observed state transitions. An ε-greedy policy was used for exploration. The value of ε begins at 0.6 and decays exponentially with a factor of 0.9995 per episode until it reaches 0.05. In this 50×50 grid setting, the decay schedule allows broad initial exploration before shifting emphasis to exploitation. Actions are restricted to four possibilities—left (0), down (1), right (2), and up (3)—consistent with standard discrete grid-world formulations and the discrete grid structure. The Q-value update follows the classic temporal-difference form:

$$Q(s,a) \leftarrow Q(s,a) + \alpha[R + \boldsymbol{\gamma}\, max_{a'}\, Q(s',a') - Q(s,a)] \quad \text{(Eq. 1)}$$

We fix the learning rate α at 0.15 and the discount factor γ at 0.99. These values handle the stochastic transitions (is_slippery=True) and support planning over long horizons in the large grid [3]. State s denotes the flattened grid position index. The action a selects one of the four movement directions. Q(s, a) represents the estimated discounted cumulative reward starting from state s and action a [3,75].

3.2.3 Reward system

In this research, the reward system is designed to represent energy efficiency under winter conditions, it helps agents find the best paths. At each step, the system gives a negative value to show how significant a step is to energy saving, with higher cause on snow grids. A positive 700 is given only when the agents successfully reach the destination, while within the maximum steps of 1000 steps. It encourages agents to reach the destination with considerations on energy saving. We initialize the cumulative reward to 0 at the beginning of each episode. Each will result in negative values, because by doing this, agents don't need to consider the prior bias, since it assumes nothing at first, etc. As in the settings, we have Near-snow (light snow), Edge-snow (medium snow), and Core-snow (deep snow); each of them has different additional values which are -0.2, -0.5, and -2.0 respectively [10].

$$r_t = r_{step} - c_{energy}(s_t) + r_{goal} \quad \text{(Eq. 2)}$$

The basic reward function follows standard reinforcement learning practice by combining step penalties, energy-related costs, and a terminal goal reward.

3.2.4 Proximal Policy Optimization

Proximal Policy Optimization (PPO) served as the policy-gradient method in this study for energy-efficient routing under winter conditions. PPO learns a stochastic policy by directly outputting action probabilities for each state, which helps the agent cope with uncertainty on slippery roads.

$$L^{CLIP}(\theta) = E_t[min(r_t(\theta)\widehat{A}_t, clip(r_t(\theta), 1 - \epsilon, 1 + \epsilon)\widehat{A}_t)] \quad \text{(Eq. 3)}$$

where $r_t(\theta)$ is the probability ratio between the new and previous policies. The clipping mechanism constrains policy updates to improve stability..

The PPO agent was trained using fixed values across all experiments. The discount factor was set to γ=0.99, and Generalized Advantage Estimation (GAE) was used with λ=0.95. The policy network consisted of two fully connected hidden layers with 256 units each, using ReLU activation functions. The learning rate was set to 3×10^{-4}, with a clipping range of 0.2 and an entropy coefficient of 0.02 to encourage exploration. PPO training was conducted for 1,200,000 time steps, using a rollout buffer of 2048 steps, a batch size of 256, and 10 optimization epochs per update to improve training stability [8].

Policy-gradient methods work by directly adjusting a parameterized policy to increase expected reward. They learn a stochastic policy, allowing for more flexibility in uncertain environments, where the same action can lead to different outcomes. In contrast, q-learning estimates state-action values and usually converges with a deterministic policy based on these estimates. Although it can learn in unfamiliar environments, its action selection may be less reliable in highly unpredictable situations. As a result, policy-gradient and value-based methods show varying strengths depending on the level of uncertainty in the winter routing task.

3.2.5 Curriculum Learning for PPO

PPO training followed a two-phase curriculum learning approach designed to improve training stability and sample efficiency under sparse rewards and energy-sensitive penalties [11,8].

Phase 1: Navigation Learning Phase (300,000 timesteps)
In phase 1, PPO was trained using a simplified reward function that focuses only on positive feedback for reaching the goal. Energy costs in snow grids were not included. This made the feedback denser and more immediate. The agent quickly learned basic navigation and achieved early success in the 50×50 winter grid. The policy learned in this phase served as the starting point for the next training stages.

Phase 2: Energy-Aware Optimization Phase (900,000 timesteps)
In phase 2, the training continued with the complete energy-aware reward function, which now included penalties for moving across snow-covered areas. The agent had to balance successful arrival with lower energy consumption. All reported results, ablation studies, and comparisons are based solely on the Phase 2 reward function.

3.2.6 Curriculum Learning Ablation Study

To validate the necessity of curriculum learning, an ablation study was conducted comparing PPO with curriculum versus PPO without curriculum under both deterministic and stochastic conditions across 10 seeds assessed through total cumulative reward and success rate. Without Curriculum learning, PPO achieved a mean of −1720.15 (95% CI [−1894.63, −1545.67]) and −1362.39 (95% CI [−1780.01, −944.78]) with 0% and 10% success rate respectively, under deterministic and stochastic conditions. Whereas with curriculum, PPO demonstrates a strong performance of 588.73 (95% CI [588.25, 589.21]) and 258.69 (95% CI [251.02, 266.37]) with 100% success rate under both conditions. These results confirm that curriculum learning is a critical component for PPO, as its absence leads to failure under both conditions.

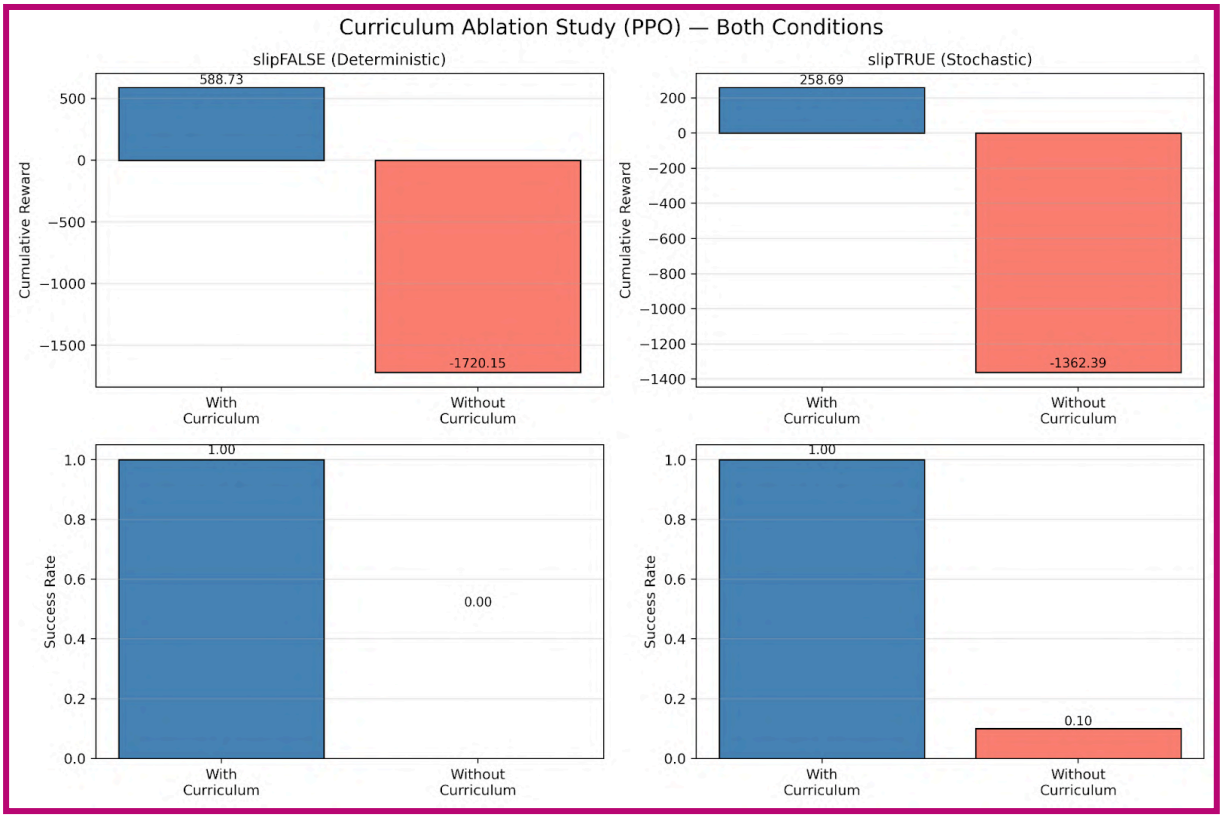


Figure 2: displays the cumulative reward and success rate for PPO with and without curriculum under both deterministic and stochastic conditions, illustrating the substantial performance gap between the two configurations.

Therefore, all subsequent comparisons and evaluations in this study refer exclusively to PPO with curriculum learning.

3.3 Evaluation factors

To ensure statistical reliability, all reinforcement learning experiments were repeated over 10~~5~~ independent random seeds (seed 0 to seed 9), and results are reported as the mean and standard deviation with 95% confidence intervals across runs.

To compare the performances of all routing methods under identical conditions, a final evaluation was conducted after training. Each method BFS, Q-learning, and PPO was evaluated using the following metrics:

- Total Reward: calculated as the cumulative reward obtained over the episode.
- Number of steps: steps taken from the start to the goal (less than 1000 steps).
- Average snow cell visited: Snow visits were calculated by counting the number of snow-covered cells traversed in each episode and averaging this value across all evaluation episodes.
- Success rate: defined as the percentage of episodes in which the agent reached the goal within the maximum step limit.

The BFS baseline was evaluated only once on the fixed grid map, as it produces a deterministic path. Reinforcement learning agents were evaluated over 200 episodes per run, with experiments repeated over 10~~5~~ independent random seeds for each condition (slip and non-slip). All methods were compared using the same energy-aware reward function and environment configuration to ensure fairness.

3.4 Hyperparameter Selections

Hyperparameters are selected based on standard practices and empirical validation to ensure stable and consistent performance [3,7~~3,6,7~~3,~~4~~].

3.4.1 Q-Learning Hyperparameter Configuration

We implement tabular Q-learning with an ε-greedy exploration strategy.

The hyperparameters are set as follows:
- Learning rate $\alpha = 0.15$
- Discount factor $\gamma = 0.99$
- Exploration strategy: ε-greedy
- Initial $\varepsilon = 0.60$
- Minimum $\varepsilon = 0.05$
- Exponential decay rate = 0.9995
- Training episodes = 15,000
- Maximum steps per episode = 1,000

The learning rate ($\alpha = 0.15$) is selected to balance convergence speed and stability. A moderate learning rate allows the agent to adapt efficiently while avoiding oscillations in value updates.

The discount factor ($\gamma = 0.99$) emphasizes long-term rewards, which is essential for navigation tasks where the objective is to reach the goal efficiently over multiple steps.

An ε-greedy exploration strategy is adopted to balance exploration and exploitation. The initial exploration rate ($\varepsilon = 0.60$) encourages sufficient exploration in early training, while exponential decay gradually shifts the policy toward exploitation. The minimum ($\varepsilon = 0.05$) ensures that some level of exploration is maintained throughout training, preventing the agent from getting stuck in suboptimal policies.

The number of training episodes 15,000 and maximum steps per episode 1,000 are chosen to ensure sufficient interaction with the environment for convergence, while maintaining computational efficiency.

During evaluation, a fully greedy policy ($\varepsilon = 0$) is used to assess the learned policy performance.

3.4.2 PPO Hyperparameter Configuration

We implement PPO using a clipped surrogate objective with generalized advantage estimation (GAE).

The hyperparameters are set as follows:
- Discount factor $\gamma = 0.99$
- GAE parameter $\lambda = 0.95$
- Learning rate $= 3 \times 10^{-4}$
- Clipping range $\varepsilon = 0.2$
- Rollout buffer size = 2048 steps
- Batch size = 256
- Optimization epochs per update = 10
- Entropy regularization is enabled

The policy and value networks share a neural architecture consisting of two hidden layers with 256 units each and ReLU activation.

The discount factor ($\gamma = 0.99$) is selected to emphasize long-term rewards, which is essential for navigation tasks. The GAE parameter ($\lambda = 0.95$) provides a balance between bias and variance in advantage estimation, leading to more stable learning.

The clipping parameter ($\varepsilon = 0.2$) constrains policy updates to prevent large deviations from the previous policy, which improves training stability. The rollout buffer size and

number of optimization epochs are chosen to ensure sufficient policy updates while avoiding overfitting to recent trajectories.

Entropy regularization is included to encourage exploration and prevent premature convergence to suboptimal policies.

Overall, these hyperparameters follow widely adopted settings in reinforcement learning literature and are chosen to ensure stable and consistent training rather than aggressive performance tuning.

A complete summary of environment settings and hyperparameters is provided in Appendix A.

---

4. Results

The experiments compare BFS, Q-learning, and PPO on the same 50×50 winter grid, looking at both non-slip (deterministic) and slip (stochastic) cases.

4.1 Evaluation Setup and Metrics

The grid layout, start and goal positions, and reward parameters were kept exactly the same across all methods. All experiments were repeated over 10~~5~~ independent random seeds, and results are reported as the mean ± standard deviation with 95% confidence intervals across runs. Trajectory figures shown are from the representative seed whose performance was closest to the mean. For each run, total reward, number of steps, average snow cells crossed per episode, and success in reaching the goal were recorded.

4.2 Trajectory Visualizations and Quantitative Results

4.2.1 Deterministic (Non-Slip) Winter Condition

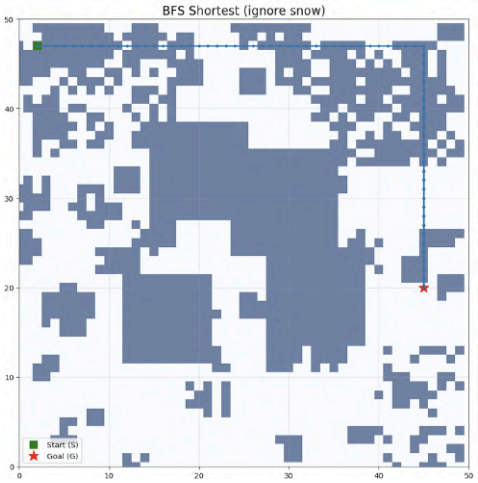
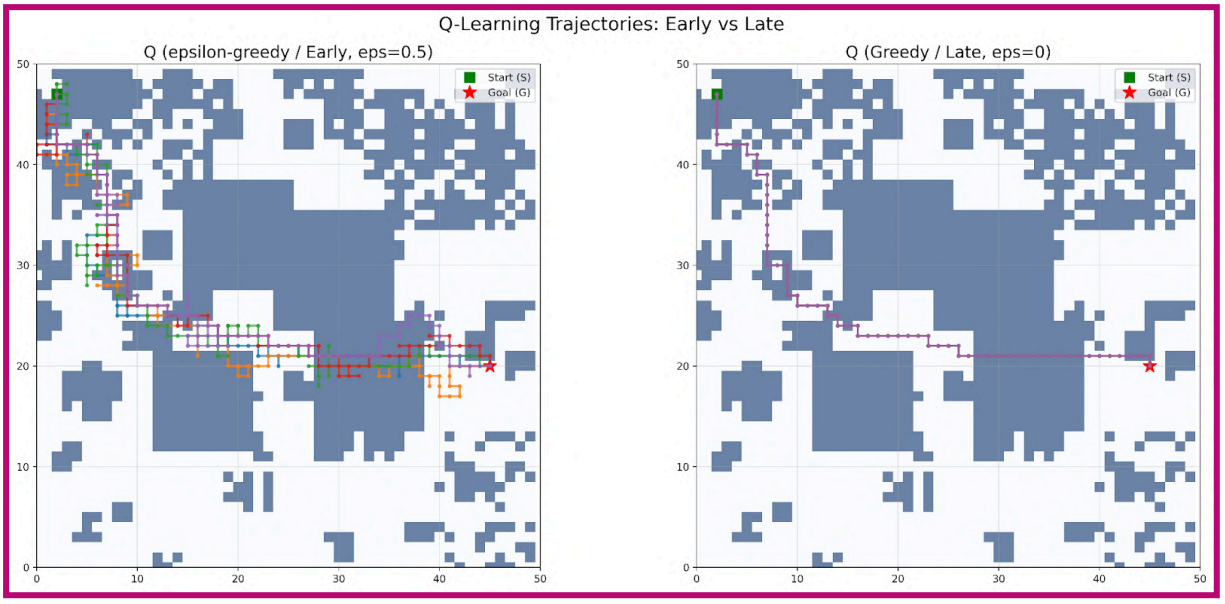


Figure 33: illustrates the BFS path under non-slip conditions. Since BFS always finds the shortest route, the trajectory goes straight from the start to the goal with minimal cells visited without any deviation.



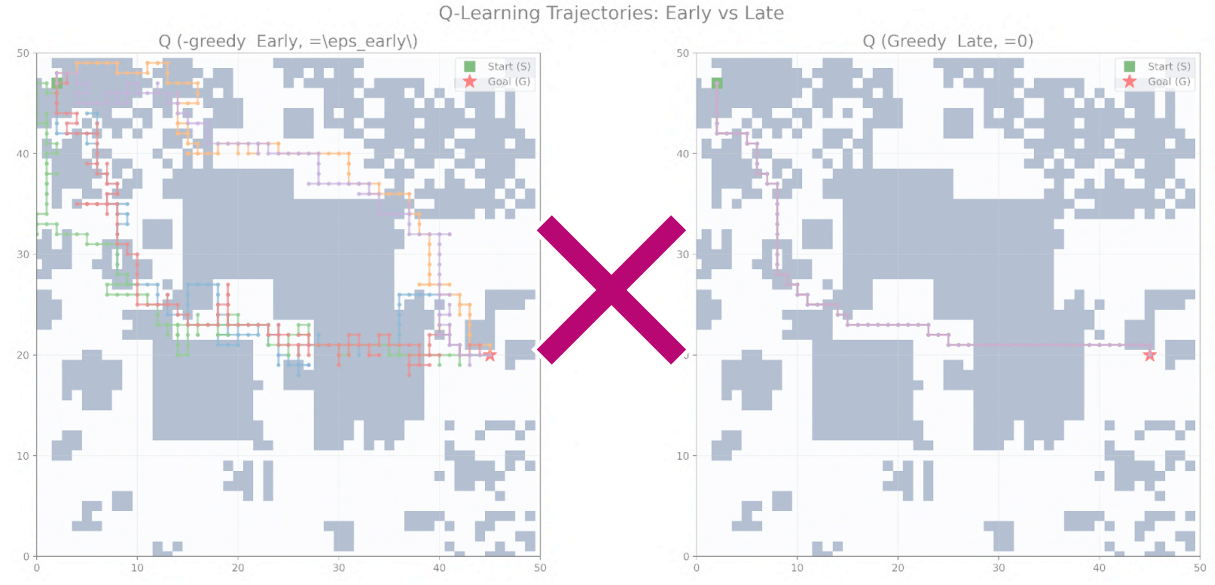


Figure 44: illustrates the navigation trajectories produced by the Q-learning agent under deterministic (non-slip) winter conditions during early training with an ε-greedy policy and late training with a greedy policy. The trajectories demonstrate the transition from exploratory behavior to a stable path toward the goal.

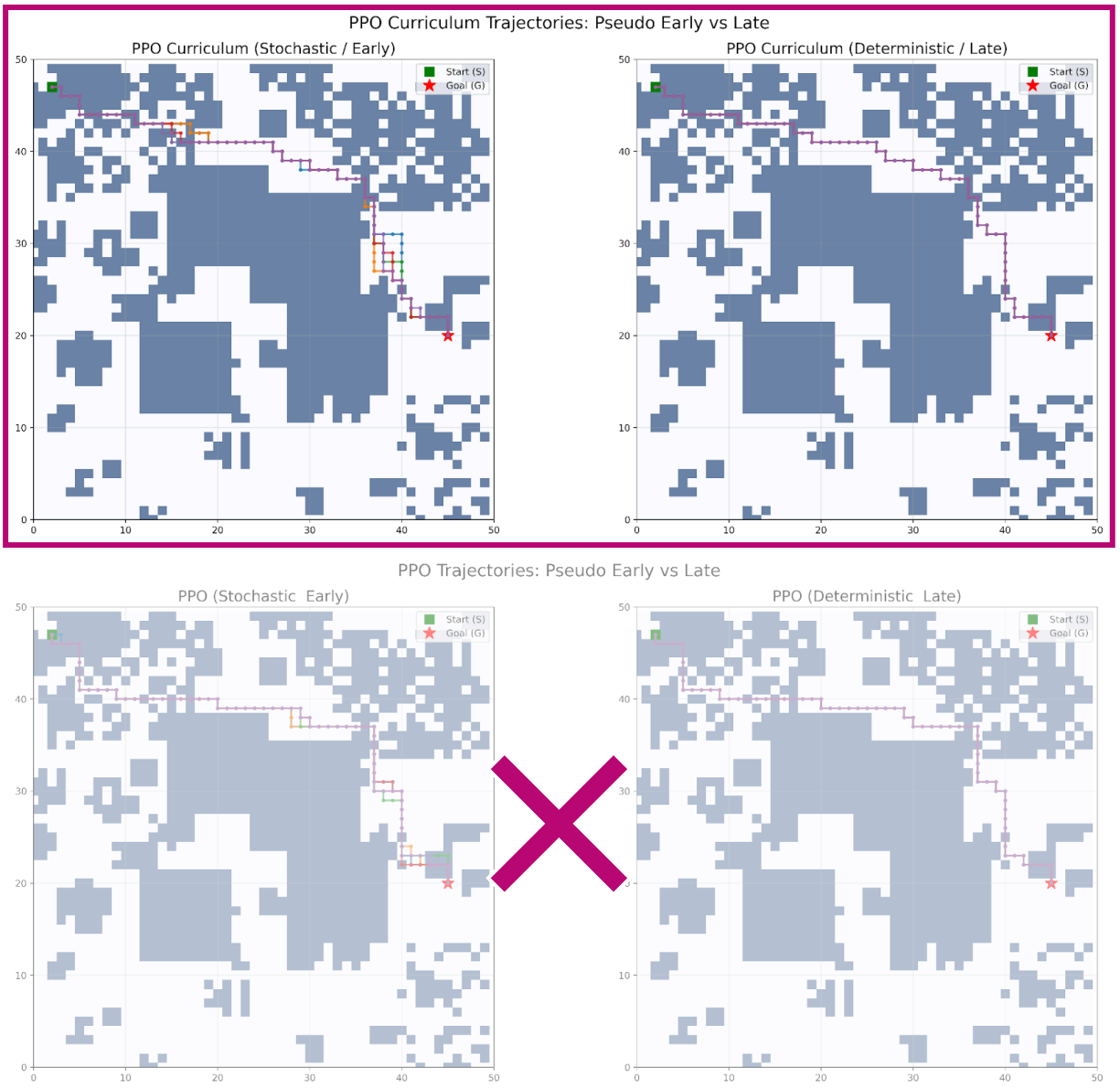
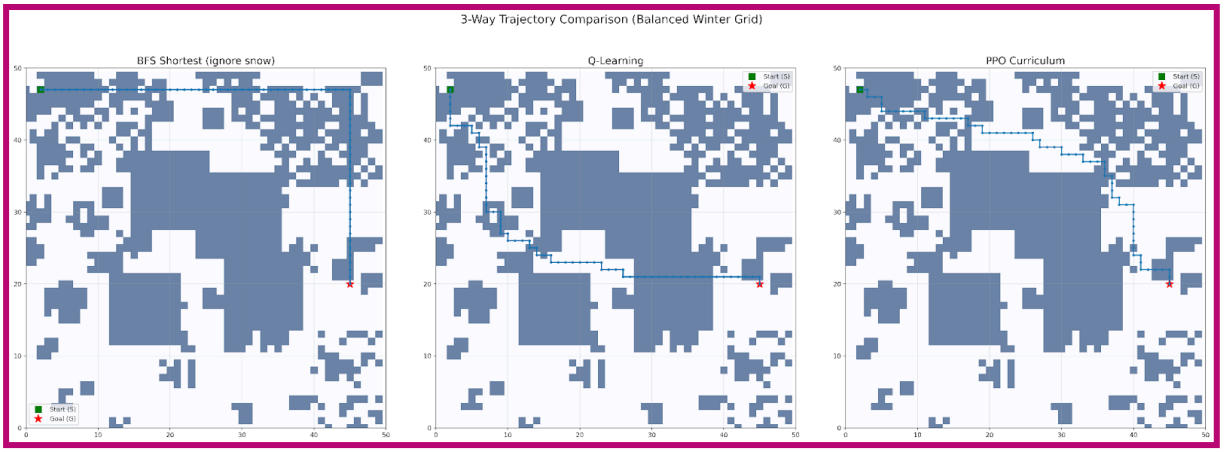


Figure 55: illustrates the navigation trajectories produced by the PPO agent during early and late stages of training under deterministic (non-slip) winter conditions.



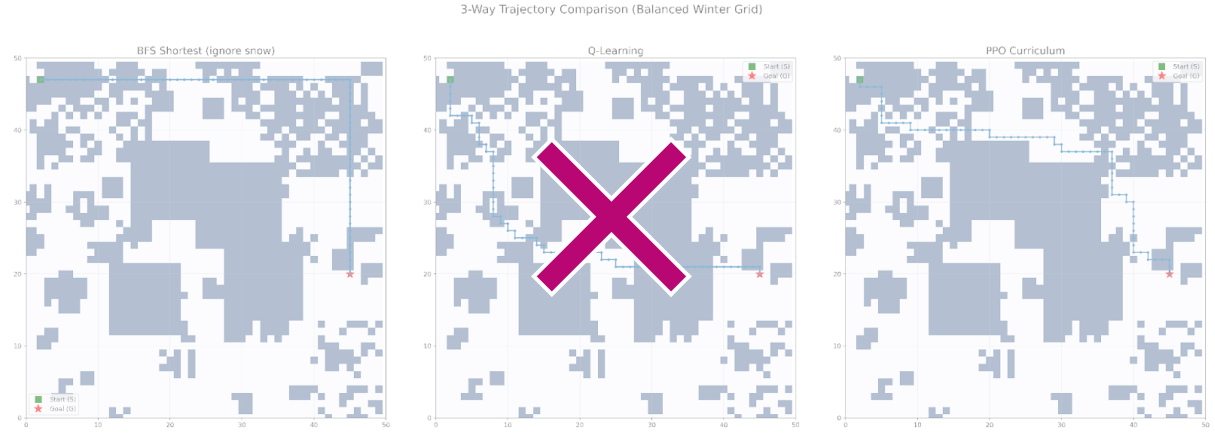


Figure 6: compares the trajectories of BFS, Q-learning and PPO in the non-slip winter setting.

=== Final Comparison Table ===

Method	Reward	Steps	Snow Visits	Success
BFS Shortest	575.30±0.00	70.00±0.00	31.00±0.00	100.0%
Q-Learning	582.01±6.97 579.10±5.61	70.00±0.00	13.10±3.05 12.60±3.29	100.0%
PPO Curriculum	588.73±0.67 588.50±0.60	70.00±0.00	7.60±0.97 7.80±1.10	100.0%

Method	Reward (mean±SD)	95%CI	Steps	Snow Visits	Success
BFS Shortest	575.30 ± 0.00	[575.30, 575.30]	70.00 ± 0.00	31.00 ± 0.00	100.0%
Q-Learning	582.01 ± 6.97	[577.02, 587.00]	70.00 ± 0.00	13.10 ± 3.05	100.0%
PPO Curriculum	588.73 ± 0.67	[588.25, 589.21]	70.00 ± 0.00	7.60 ± 0.97	100.0%

Table 1: summarizes the main performance metrics for all three methods under deterministic conditions, including total reward, steps to goal, average snow cells crossed, success rate, standard deviation (SD), and averaged across 10 independent random seeds with 95% confidence intervals.

4.2.2 Slip Winter Condition

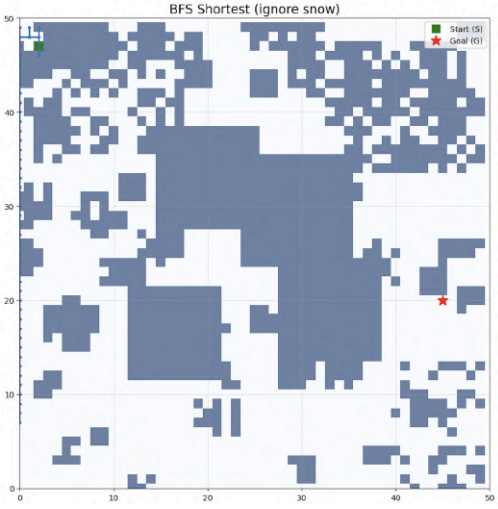
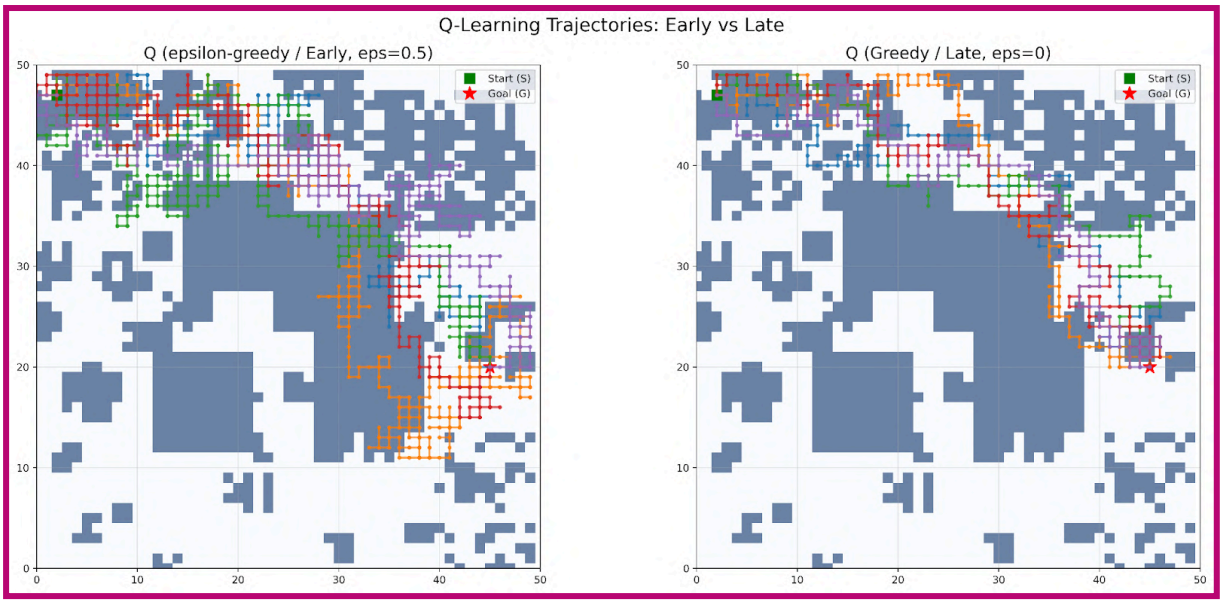


Figure 7: illustrates the navigation path produced by the BFS baseline under slip (stochastic) winter conditions.



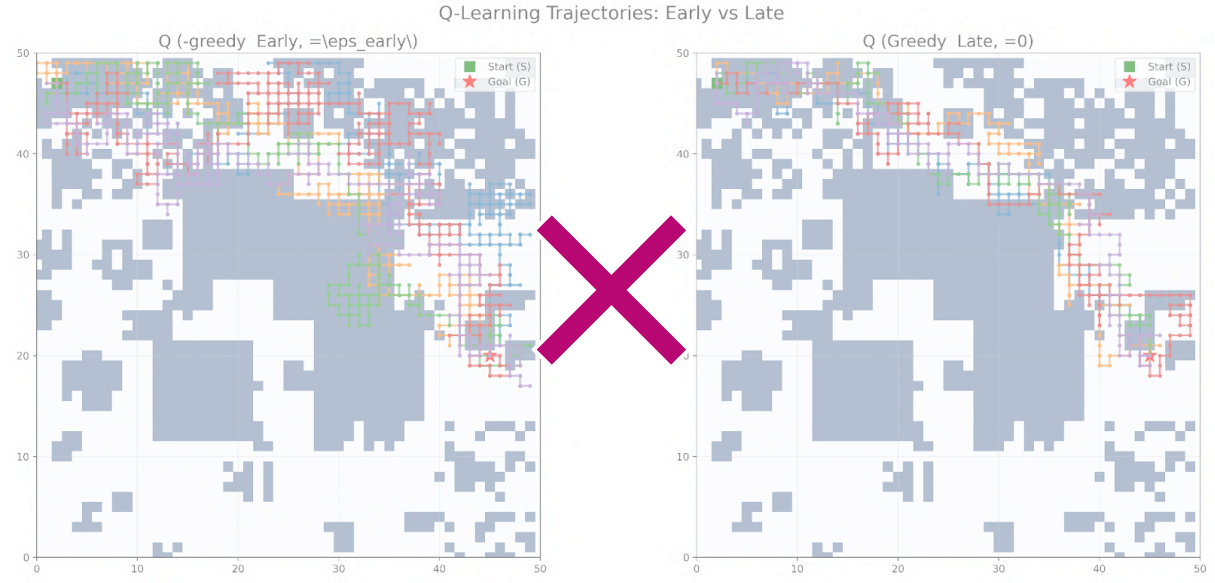
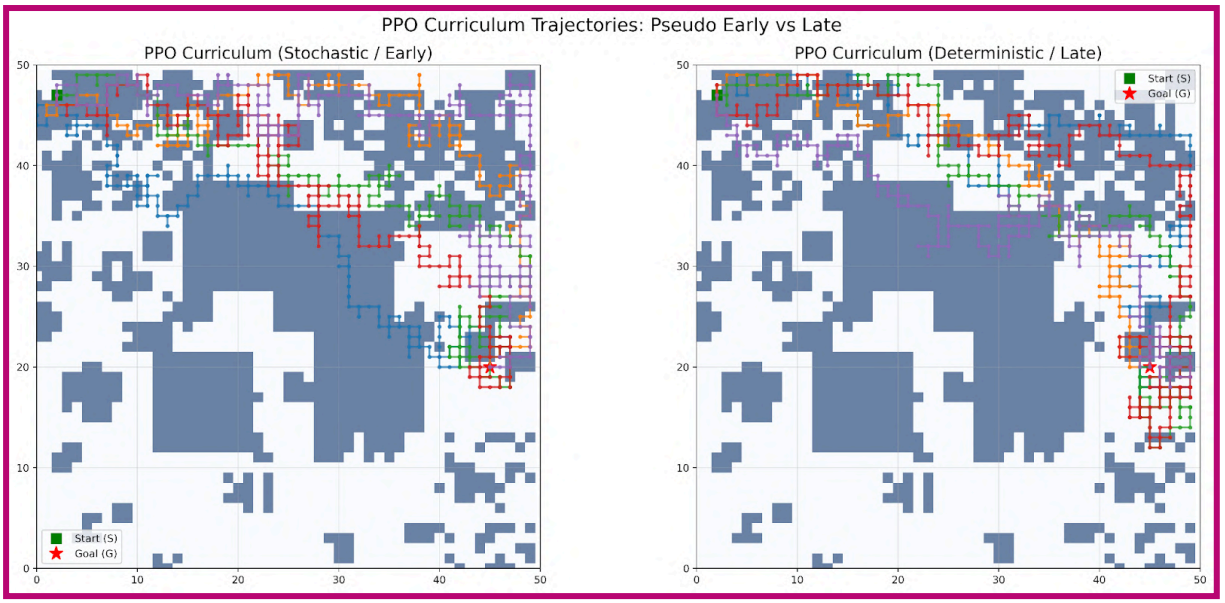


Figure 88: illustrates the navigation trajectories produced by the Q-learning agent during early and late stages of training under slip (stochastic) winter conditions.



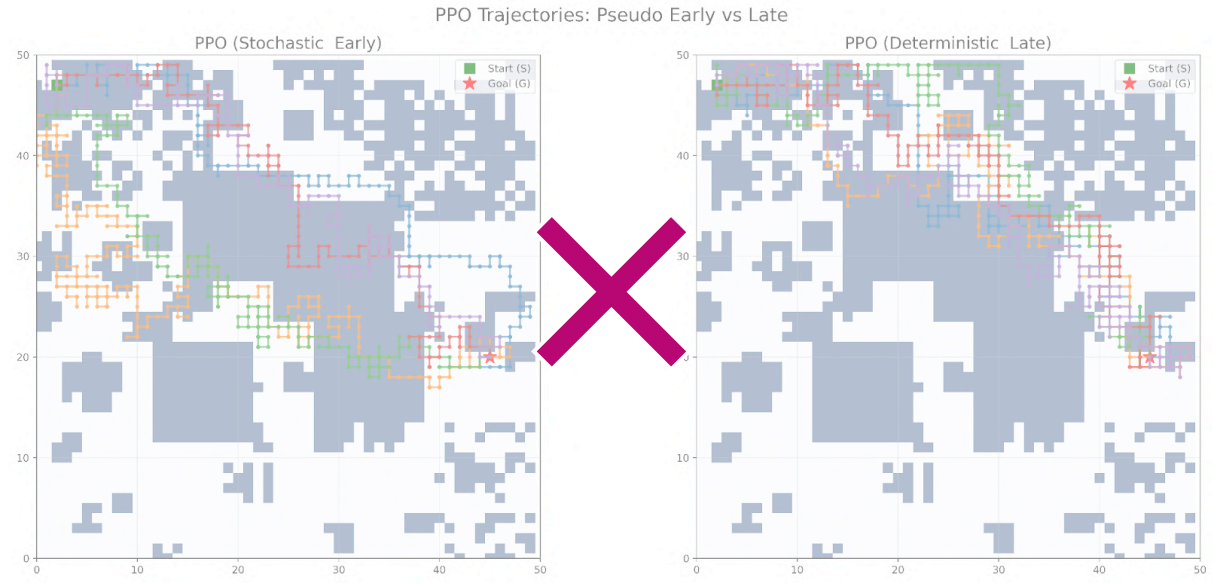
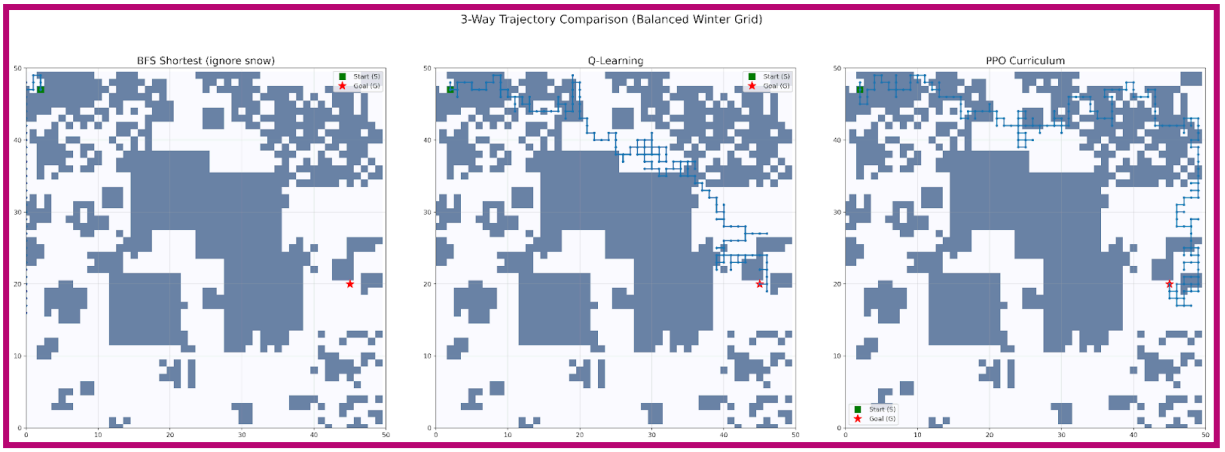


Figure ~~9~~9: illustrates the navigation trajectories produced by the PPO agent during early and late stages of training under slip (stochastic) winter conditions.



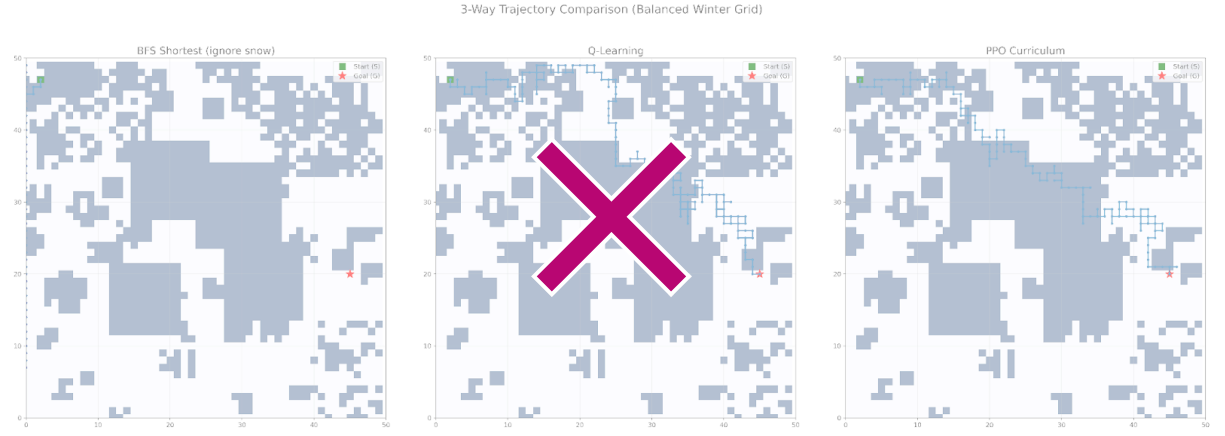


Figure 10~~10~~: compares the navigation trajectories of BFS, Q-learning, and PPO under slip winter conditions.

=== Final Comparison Table ===¶

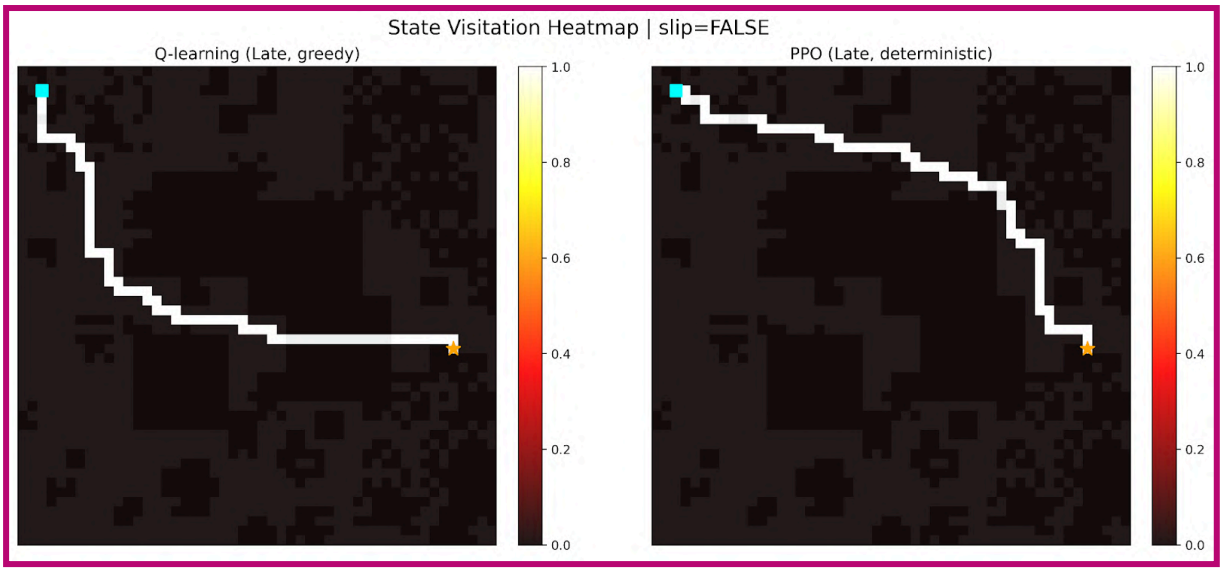
Method¶	**Reward**¶	**Steps**¶	**Snow Visits**¶	**Success**¶
BFS Shortest	1668.41±113.42 1693.96±61.27¶	1000.00±0.00¶	246.80±76.09	0.0%¶
Q-Learning	311.82±4.83¶ 304.03±9.92¶	226.40±3.92¶ 228.18±4.10	66.94±3.87¶ 67.58±4.34	100.0%¶
PPO Curriculum	258.69±10.73¶ 252.19±7.04¶	241.18±4.11¶ 242.29±3.00¶	91.88±5.24¶ 92.65±4.14	100.0%¶ ¶

Method	Reward (mean±SD)	95% CI	Steps	Snow Visits	Success
BFS Shortest	-1668.41 ± 113.42	[-1749.54, -1587.28]	1000.00 ± 0.00	246.80 ± 76.09	0.0%
Q-Learning	311.82 ± 4.83	[308.37, 315.27]	26.40 ± 3.92	66.94 ± 3.87	100.0%
PPO Curriculum	258.69 ± 10.73	[251.02, 266.37]	241.18 ± 4.11	91.88 ± 5.24	100.0%

Table 2: summarizes the quantitative performance metrics for BFS, Q-learning, and PPO under slip winter conditions, using the same evaluation metrics as in the deterministic case, Sstandard Ddeviation (SD), and averaged across 105 independent random seeds with 95% confidence intervals.

4.3 Spatial State Visitation Heatmaps

Spatial state visitation heatmaps offer a grid-based way to see how often an agent passes through each cell during navigation. In our 50×50 winter grid, each cell corresponds to a state, and the color intensity reflects visitation frequency across episodes. Darker areas indicate cells that are visited more frequently, while lighter or white cells indicate rarely or never visited locations. Such visualizations give a clear picture of the agent's overall path distribution and behavior patterns. We generated these heatmaps using aggregated visitation counts normalized to the [0,1] range between 0 and 1.



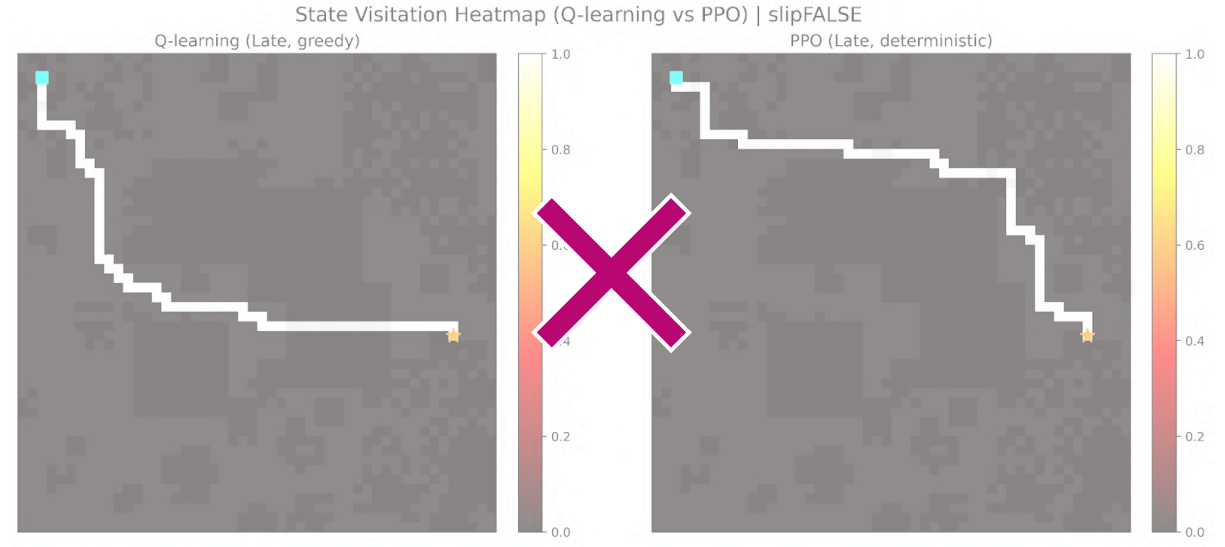


Figure 11: displays the visitation heatmaps for Q-learning and PPO under deterministic (non-slip) winter conditions. The results are aggregated from 10 evaluation runs consisting of 200 episodes per run. Both agents concentrate most visits along a narrow corridor connecting the start to the goal, with near-maximum intensity (≈1.0) along the primary route and very low visitation elsewhere.

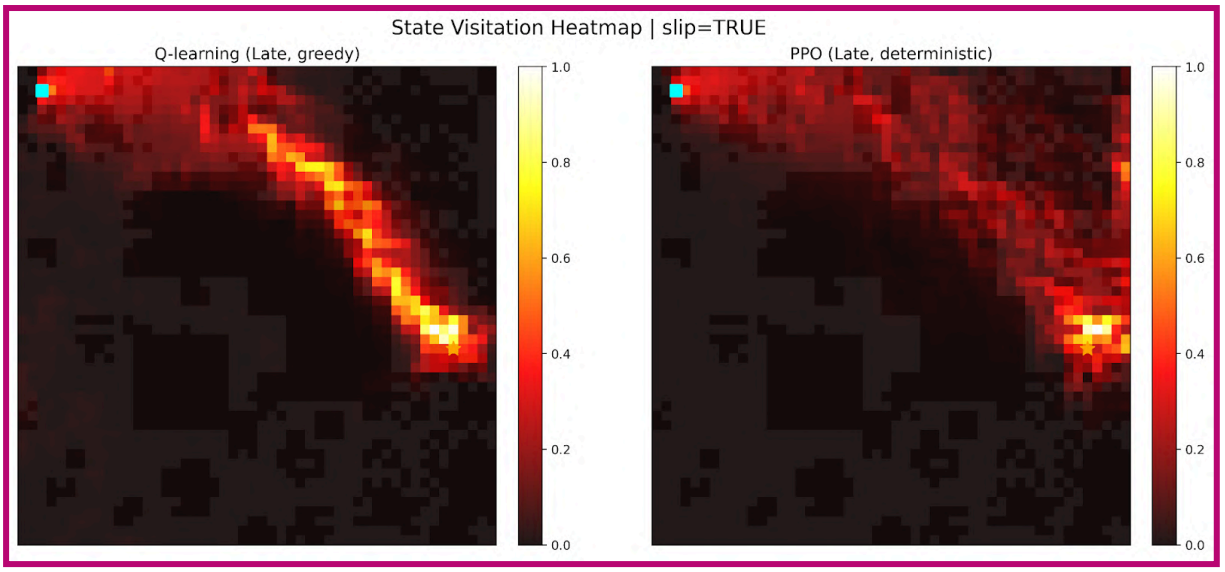


Figure 1212: shows the visitation intensity for each grid cell in the slip case. Compared with the non-slip condition, visitation is more widely distributed across the grid rather than remaining within a narrow corridor. The heatmap aggregates data from 10 evaluation runs consisting of 200 episodes per run.

4.4 Statistical Analysis and Learning Curves - Slip Condition

4.4.1 Statistical Significance (Welch's t-test)

To assess statistical significance, a Welch's t-test was conducted comparing the cumulative reward of Q-learning and PPO under slip conditions across 10 seeds. The results confirmed that Q-learning significantly outperformed PPO in cumulative rewards ($t = 14.28$, $p < 0.001$), indicating that the performance gap is highly unlikely to be due to random variation. Q-learning achieved a mean reward of 311.82 ± 4.83 compared to PPO's 258.69 ± 10.73. The higher standard deviation of PPO ($SD = 10.73$) compared to Q-learning ($SD = 4.83$) further indicates greater training instability under stochastic conditions.

The standard deviation gap between PPO in both conditions suggests that PPO is more sensitive to stochastic signals under slip condition, where high-probability slip results in noisy advantage estimation, leading to higher variance across training runs (SD = 10.73). In contrast, under deterministic conditions PPO exhibits a much lower variance (SD = 0.67), confirming that the instability is determined by the stochastic environment, not the algorithm itself.

=== Final Comparison Table ===

Metric	t-statistic	p-value	Q mean ± SD	PPO mean ± SD
Cumulative Reward	14.28	< 0.001	311.82 ± 4.83	258.69 ± 10.73
Success Rate	n/a	n/a	100% ± 0%	100% ± 0%

Table 3: Welch's t-test results comparing Q-learning and PPO Curriculum cumulative reward under slip conditions across 10 seeds. n/a indicates that the test was not applicable due to zero variance in success rate for both methods.

4.4.2 Learning Curves

Learning curves illustrate the performance of Q-learning and PPO evolves over the course of training under stochastic conditions, and provide insight into convergence behaviour, training stability, and the impact of curriculum learning on PPO.

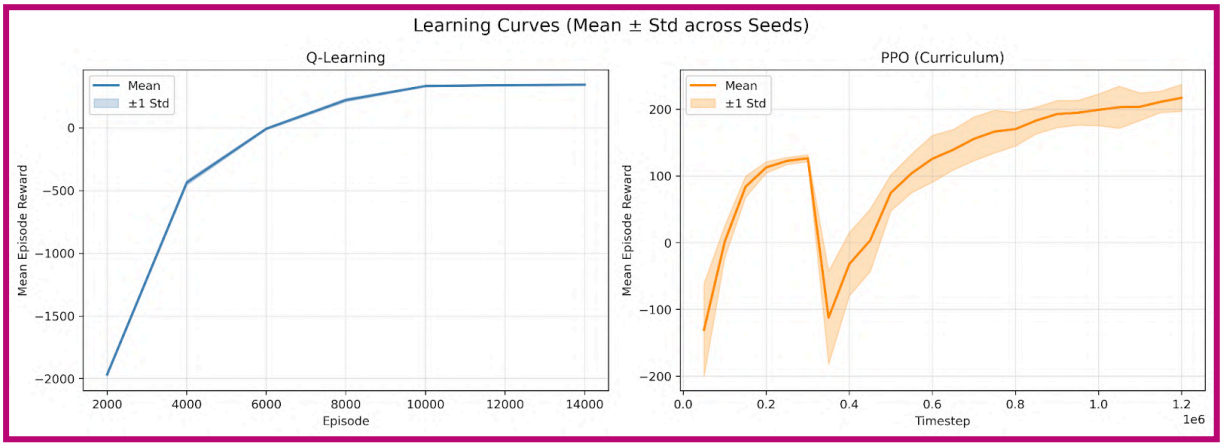


Figure 13: Learning curves (Mean ± SD across 10 seeds) under slip (stochastic) conditions. Left: Q-learning mean episode reward vs. training episode (15,000 episodes) showing smooth, steady and consistent convergence. Right: PPO Mean episode reward vs. training timestep (1,200,000 steps), where the pronounced drop around timestep 300,000 corresponds to the Phase 1 to Phase 2 curriculum transition, when

energy-aware snow penalties are introduced and the agent must re-adapt its policy. After the transition is complete, PPO gradually recovers and converges, but its variance band is significantly wider compared to Q-learning.

4.54 Limitations & Uncertainty

Several sources of uncertainty and limitations appeared in this study. Both Q-learning and PPO showed noticeable performance differences across training runs, which was expected given the stochastic nature of the environment and the learning algorithms. To account for this variability, all experiments were repeated over 105 independent random seeds, and results are reported as the mean across runs.

Under slip conditions, the same policy could produce quite different routes from episode to episode because actions did not always lead to the expected outcome. In addition, stochastic exploration strategies and random initialization of value functions (or policy networks) exposed agents to slightly different states, transitions, and rewards across runs. This variability made it harder to get perfectly consistent results.

An unexpected issue was that the BFS baseline failed to reach the goal within the 1,000-step limit under slip conditions, resulting in a number of unsuccessful evaluation episodes.

A full numerical breakdown of evaluation metrics is provided in Appendix B.

5.Discussion

5.1 Restatement of Hypothesis and Summary of Findings

The study hypothesized that reinforcement learning-based agents, particularly Proximal Policy Optimization (PPO), would achieve more energy-efficient winter navigation paths than a traditional shortest-path baseline and a value-based Q-learning agent under both deterministic and slip winter conditions. The averaged results partially supported the hypothesis. Under deterministic (non-slip) conditions, PPO achieved higher cumulative reward, and fewer snow visits, outperforming Q-learning, which supports the hypothesis. However under stochastic (slip) conditions, Q-learning outperformed PPO in terms of cumulative reward or snow-cell avoidance, which did not support the hypothesis. Both reinforcement learning methods perform better results than traditional BFS across these environments.

5.2 Interpretation of Q-Learning and PPO Performance

The results showed that algorithm performance varied depending on the environmental condition. Under deterministic (non-slip) conditions, PPO achieved higher cumulative reward and fewer snow cell visits than Q-learning, suggesting that PPO's policy gradient approach was able to learn a more energy-efficient route in a stable environment. Under stochastic (slip) conditions, Q-learning outperformed PPO in terms of cumulative reward and snow-cell avoidance. One possible explanation was that a discrete and clear

grid-world environment and relatively small environment would favor more for Q-learning , as it can perform exact value iteration over the finite MDP (Markov Decision Process), directly converging to the true optimal Q-values for each state without approximation error. Thereforeerror., Ttherefore allowing Q-learning to efficiently estimate optimal state and gives lower variance than policy gradient method (PPO), since it updates based on individual state-action pairs rather than entire trajectories [7,87]. In contrast, PPO relied on function approximation and stochastic policy updates, which may have required more training data or longer training time to converge to an equally optimal policy under slip conditions. Additionally, its stochastic policy may have introduced suboptimal choices that were not effective, whose gradient estimates already carry high variance from episode training. Under slip conditions, environmental stochasticity further interrupts reward signals, making them sparse and noisy, which destabilizes gradient estimates and requires significantly more training samples. Ttherefore reducing its total rewards relative to Q-learning under stochastic conditions [83].

5.3 Effects of Slip (Stochastic) Winter Conditions

Under slip (stochastic) conditions, all agents showed broader trajectory dispersion and increased visits to previously visited states because of slipping, as reflected in the spatial state visitation heatmaps. This behavior is consistent with probabilistic transition dynamics, in which the same action could lead to different movement outcomes across episodes, such as deviating left in one run and right in another. When the grid is slippery, agents don't follow one stable path anymore. As a result, the routes spread out, and their performance becomes less consistent across episodes. These observations are consistent with prior navigation studies showing that stochastic environments increase policy variance and reduce convergence stability in reinforcement learning tasks [4,88].

5.4 Interpretation of BFS Baseline Behavior

The Traditional baseline trajectories did not take account for winter grid costs or stochastic transitions, it focused on the fastest way to the destination. In non-slip conditions this approach naturally produced the optimal route. However, under slip conditions, BFS exceeded the maximum step limit in multiple evaluation episodes, reflecting its inability to adapt its policy in response to transition winter grid costs or stochastic transitions.

Because BFS always assumed deterministic movement, each slip caused deviations from the planned path. These deviations often led to inefficient loops and longer paths. Unlike reinforcement learning methods, BFS did not have reward feedback or policy updates, which limited its ability to adjust navigation behavior under changing environmental dynamics. Therefore, making it the least effective method.

5.5 Hyperparameter Sensitivity Analysis

To assess the robustness of the reported conclusions to hyperparameter choice, a sensitivity analysis was conducted to independently vary each hyperparameter of Q-learning and PPO under slip conditions using a single representative seed. For Q-learning, the learning rate (alpha) and epsilon decay were evaluated to identify which hyperparameter values lead to greater stability and performance. For PPO, the clip range and learning rate were varied to assess how different hyperparameters affect training stability and sensitivity to stochastic conditions.

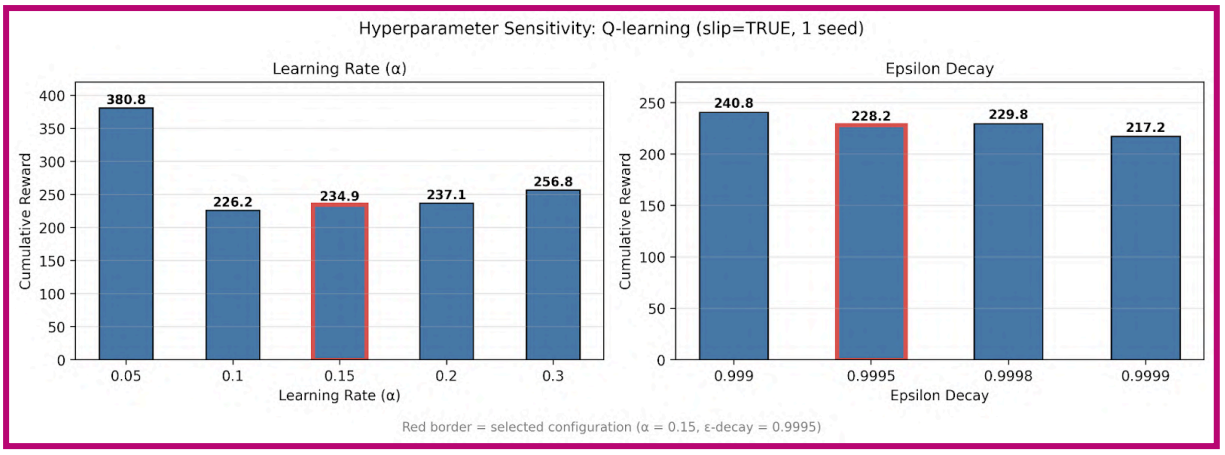


Figure 14: Comparison of different learning Rate (alpha) values and epsilon decay values . The highlighted values represent the selected hyperparameter configuration, chosen based on their balance of performance and training stability. (α = 0.15, ε-decay = 0.9995)

=== Q-learning hyperparameter values Final Comparison Table ===

Parameter	Value	Success rate (%)	Cumulative reward	Note
Learning rate (alpha)	0.05	98.00	380.80	
Learning rate (alpha)	0.10	100.00	226.20	
Learning rate (alpha)	0.15	100.00	234.90	Selected configuration
Learning rate (alpha)	0.20	100.00	237.10	
Learning rate (alpha)	0.30	100.00	256.80	

Epsilon decay	0.9990	100.00	240.80	
Epsilon decay	0.9995	100.00	228.20	Selected configuration
Epsilon decay	0.9998	100.00	229.80	
Epsilon decay	0.9999	100.00	217.20	

Table 4: presents a sensitivity analysis of Q-learning's key hyperparameters. Cumulative rewards are reported to assess overall performance.

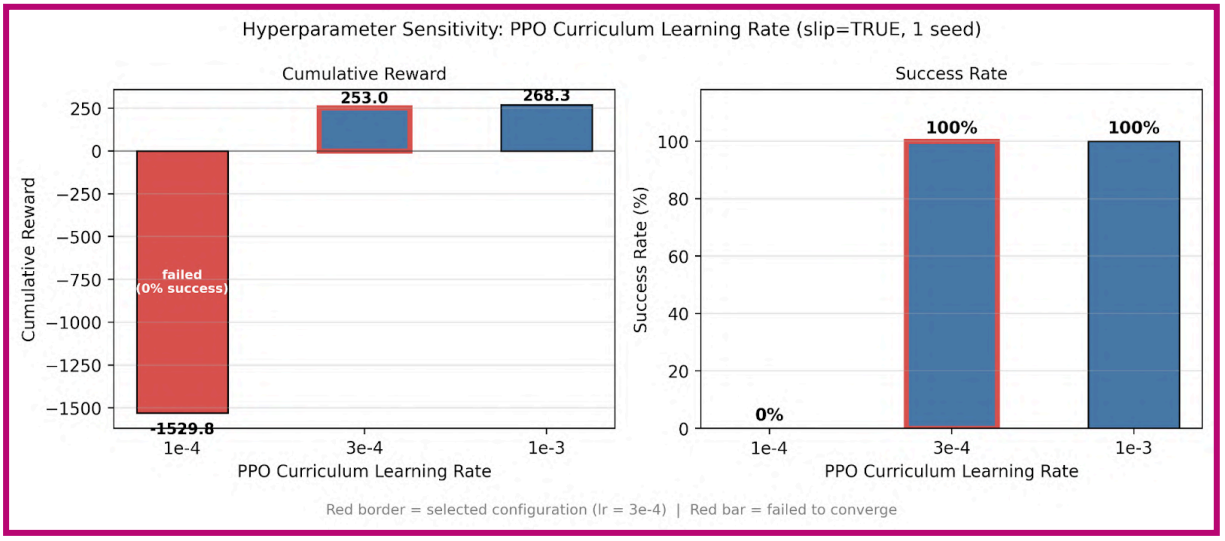


Figure 15: Comparison of different values of learning rate and the highlighted values represents the selected hyperparameter configuration, chosen based on their balance of performance and training stability (Learning rate = 3×10^{-4}). lr = 1×10^{-4} failed to converge (0% success rate, reward = −1529.8), a very small learning rate leads to insufficient policy updates, preventing convergence within the training horizon.

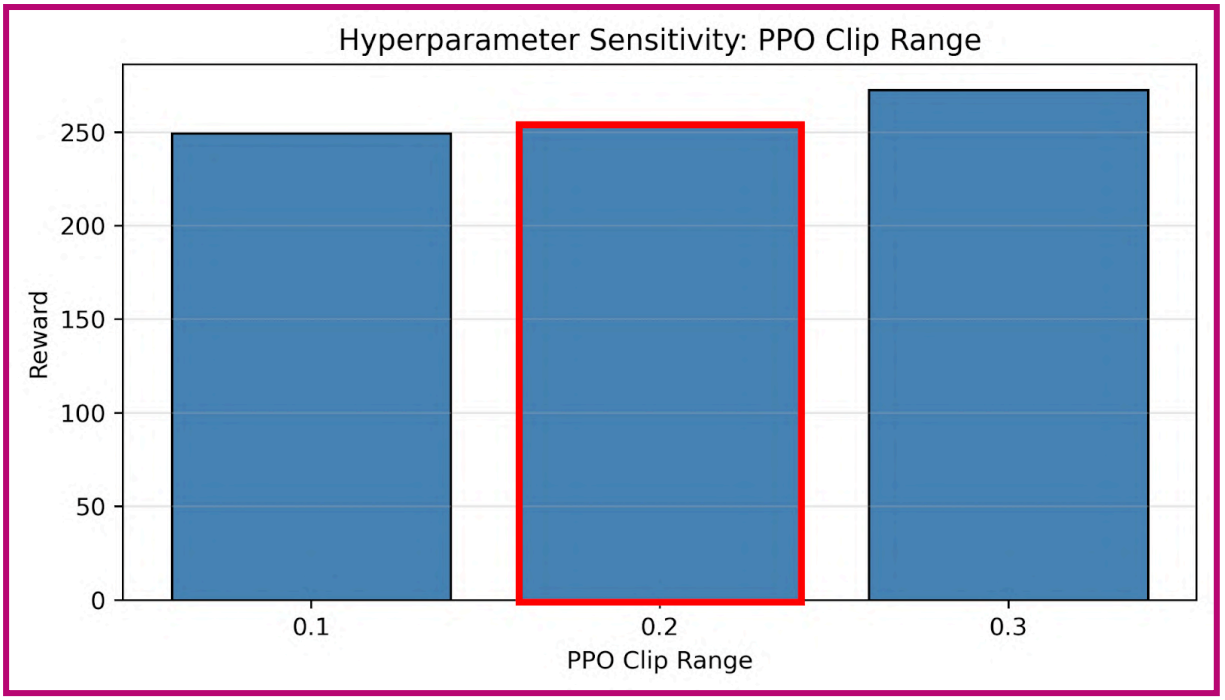


Figure 16: Comparison of different values of PPO clip range and the highlighted values represents the selected hyperparameter configuration, chosen based on their balance of performance and training stability. (Clip range = 0.2)

===PPO hyperparameter values Final Comparison Table ===

Parameter	Value	Success rate (%)	Cumulative reward	Note
Clip range	0.1	100.00	249.40	
Clip range	0.2	100.00	253.00	Selected configuration
Clip range	0.3	100.00	272.50	
Learning rate	1e-04	0.00	−1529.80	
Learning rate	3e-04	100.00	253.00	Selected configuration
Learning rate	1e-03	100.00	268.30	

Table 5: presents a sensitivity analysis of PPO's key hyperparameters. Cumulative rewards and success rates are reported to assess overall performance.

Across all tested configurations, the rank ordering of methods remained consistent, confirming that the reported conclusions are robust to moderate hyperparameter variation.

5.65 Limitations and Future Directions

Several limitations should be noted when interpreting these findings. All experiments were carried out in a simulated 50×50 grid environment with a relatively small and simple state space, which differs significantly from real-world scenarios. This setup likely favored value-based methods like Q-learning, since it could learn precise state-action values for every cell. In contrast, PPO depends on neural network function approximation and stochastic policy updates, which may need more training steps or data to perform well in such discrete, small-scale settings.

To improve statistical reliability, all experiments were run independently over 105 random seeds and results were averaged across runs. Noticeable performance variability was observed across training runs, mainly due to the stochastic environment, random action outcomes, and probabilistic transitions under slip conditions.

The simulated environment simplified real-world winter driving conditions and did not capture vehicle dynamics, sensor noise, road-surface variation, or realistic energy-consumption models, these are factors that might lead to changes. As a result, these findings to real-world winter navigation scenarios were limited [129].

Future research could evaluate reinforcement learning agents in larger scale, continuous, or more realistic winter navigation environments with higher-dimensional state spaces and more complex conditions. Then it may be better to reflect real-world uncertainty and could allow policy-gradient methods such as PPO to fully function their theoretical advantages in handling stochastic and high-dimensional control problems.

---

6. Conclusion

This study compared a traditional BFS baseline with two reinforcement learning methods—Q-learning and Proximal Policy Optimization (PPO)—for energy-efficient navigation in a 50×50 winter grid under both deterministic (non-slip) and stochastic (slip) conditions. The averaged results across 105 independent random seeds partially supported the original hypothesis. Under deterministic (non-slip) conditions, PPO achieved the highest cumulative reward and fewest snow cell visits, outperforming Q-learning. However, under stochastic (slip) conditions, Q-learning outperformed PPO in cumulative reward and snow-cell avoidance. Although both reinforcement learning methods clearly outperformed the BFS baseline, BFS struggled under slip conditions due to its inability to adapt to stochastic transitions.

These findings show that whether value-based methods or policy-gradient methods perform better really depends on how unpredictable the environment is. In our small, structured discrete grid, PPO gained an advantage from its stable gradients and

curriculum learning when everything was deterministic. On the other hand, Q-learning's simple tabular updates turned out to be more robust when the roads became slippery and actions sometimes failed. More generally, the results emphasize that the choice of reinforcement learning method should be guided by the structure and constraints of the task, rather than by theoretical advantages alone. Future work could explore whether PPO's advantages extend to more complex environments with continuous states or larger state spaces, where its policy-gradient approach may better demonstrate its theoretical strengths.

Reference

[1]Mao, R., Xu, W., Qian, Y., Li, X., Li, Y., Li, G., & Zhang, H. (2025). Understanding the Determinants of Electric Vehicle Range: A Multi-Dimensional Survey. *Sustainability*, 17(10), 4259. https://www.mdpi.com/2071-1050/17/10/4259 ~~https://doi.org/¶ 10.3390/su17104259~~

[2]Carlson, A., & Vieira, T. (2021). *The effect of water and snow on the road surface on rolling resistance* (VTI Report 971A). Swedish National Road and Transport Research Institute. https://www.researchgate.net/publication/350690120_The_effect_of_water_and_snow_on_the_road_surface_on_rolling_resistance ~~https://www.diva-portal.org/smash/get/diva2:1542142/¶ FULLTEXT01.pdf~~

[3]Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction. https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf

[4]Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A. J., Banino, A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., Kumaran, D., & Hadsell, R. (2017). *Learning to Navigate in Complex Environments* (No. arXiv:1611.03673). arXiv. https://doi.org/10.48550/arXiv.1611.03673

[5]Tan, C. (2025). Comparative Study of Reinforcement Learning Performance Based on PPO and DQN Algorithms. *Applied and Computational Engineering*, 175(1), 30–36. https://doi.org/10.54254/2755-2721/2025.AST24879

[6]Warnakulasuriya, D. A., Plosila, J., & Haghbayan, H. (2025). Energy-Efficient Path Planning in Uneven Terrains Using Adaptive Reinforcement Learning. *IEEE Conference Publication*. https://ieeexplore.ieee.org/document/11093435
[7]Watkins, C.J.C.H., & Dayan, P(1992). Q-learning. *Mach Learn* 8, 279–292. https://doi.org/10.1007/BF00992698
[8]Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). *Proximal Policy Optimization Algorithms* (No.arXiv:1707.06347). arXiv. https://doi.org/10.48550/arXiv.1707.06347
[9]Pendyala, A., Atamna, A., & Glasmachers, T. (2024). Solving a Real-World Optimization Problem Using Proximal Policy Optimization with Curriculum Learning and Reward Engineering. arXiv:2404.02577.https://arxiv.org/abs/2404.02577
[10]Dayan, P., & Balleine, B. W. (2002). Reward, Motivation, and Reinforcement Learning. *Neuron*, 36(2), 285–298. https://doi.org/10.1016/S0896-6273(02)00963-7
[11]Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). Curriculum learning. *Proceedings of the 26th Annual International Conference on Machine Learning*, 41–48. https://doi.org/10.1145/1553374.1553380
[12]Chukwurah, N., Adebayo, A. S., Ajayi, O. O., & Anfo Pub. (2024). *Sim-to-Real Transfer in Robotics: Addressing the Gap between Simulation and Real- World Performance*. *International Journal for Multidisciplinary Research (IJFMR)*, 05(01), 33–39. https://doi.org/10.54660/.IJFMR.2024.5.1.33-39

[3]Watkins, C.J.C.H., & Dayan, P(1992). Q-learning. *Mach Learn* 8, 279–292.¶
https://doi.org/10.1007/BF00992698¶
[4]Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O.¶
(2017). *Proximal Policy Optimization Algorithms* (No.¶
arXiv:1707.06347). arXiv. https://doi.org/10.48550/arXiv.1707.06347¶
[5]Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An¶
Introduction.¶
https://web.stanford.edu/class/psych209/Readings/¶
SuttonBartoIPRLBook2ndEd.pdf¶
[6]Dayan, P., & Balleine, B. W. (2002). Reward, Motivation, and¶
Reinforcement Learning. *Neuron*, 36(2), 285–298. https://doi.org/¶
10.1016/S0896-6273(02)00963-7¶
[7]Tan, C. (2025). Comparative Study of Reinforcement Learning¶
Performance Based on PPO and DQN Algorithms. *Applied and*¶
Computational Engineering, 175(1), 30–36. https://doi.org/¶
10.54254/2755-2721/2025.AST24879¶
[8]Mirowski, P., Pascanu, R., Viola, F., Soyer, H., Ballard, A. J., Banino,¶
A., Denil, M., Goroshin, R., Sifre, L., Kavukcuoglu, K., Kumaran, D., &¶
Hadsell, R. (2017). *Learning to Navigate in Complex Environments* (No.¶
arXiv:1611.03673). arXiv. https://doi.org/10.48550/arXiv.1611.03673¶

[9]Chukwurah, N., Adebayo, A. S., Ajayi, O. O., & Anfo Pub. (2024).¶
Sim-to-Real Transfer in Robotics: Addressing the Gap between¶
Simulation and Real-World Performance. International Journal for Multidisciplinary
Research (IJFMR), 05(01), 33–39. https://doi.org/¶
10.54660/.IJFMR.2024.5.1.33-39¶
[10]Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009).¶
Curriculum learning. *Proceedings of the 26th Annual International*¶
Conference on Machine Learning, 41–48. https://doi.org/¶
10.1145/1553374.1553380¶
[11]Pendyala, A., Atamna, A., & Glasmachers, T. (2024). Solving a Real-World
Optimization Problem Using Proximal Policy Optimization with Curriculum Learning
and Reward Engineering. arXiv:2404.02577.https://arxiv.org/abs/2404.02577¶
[12]Warnakulasuriya, D. A., Plosila, J., & Haghbayan, H. (2025). Energy-Efficient Path
Planning in Uneven Terrains Using Adaptive Reinforcement Learning. *IEEE Conference*
Publication. https://ieeexplore.ieee.org/document/11093435

Appendices

Appendix A Experimental Configuration and Algorithmic Details:

A.1 Environment Configuration

All experiments were conducted in a custom 50×50 winter grid environment (2,500 discrete states).

- Start position: (47, 2)

- Goal position: (20, 45)
- Maximum steps per episode: 1000
- Action space: {up, down, left, right}

Two transition settings were evaluated:

Deterministic (slip = FALSE)

The intended action is executed exactly.

Stochastic (slip = TRUE)

With probability $\frac{1}{3}$, the intended action is executed.
With probability $\frac{2}{3}$, the agent slips to a random adjacent direction.

All algorithms were evaluated on the same fixed grid layout to ensure fairness.

A.2 Reward Function

The reward function models energy-aware winter navigation.

At each time step:

- Step penalty: −1.5
- Snow penalties:
  - Near snow: −0.2
  - Edge snow: −0.5
  - Core snow: −2.0
- Goal reward: +700

The cumulative episode reward is:

$$R = \sum_{t=1}^{T} (-1.5 - C_{snow}(s_t)) + 700 \cdot 1_{goal\ reached}$$

where $C_{snow}(s_t)$ denotes the terrain penalty and
$1_{goal\ reached}$ indicates successful termination.

A.3 Q-Learning Configuration

Tabular Q-learning was implemented with the following hyperparameters:

- Learning rate α=0.15
- Discount factor γ=0.99
- Exploration strategy: ε-greedy
- Initial ε = 0.60
- Minimum ε = 0.05
- Exponential decay per episode = 0.9995
- Training episodes = 15,000
- Max steps per episode = 1000

The Q-update rule is:

$Q(s,a) \leftarrow Q(s,a) + \alpha[R + \boldsymbol{\gamma}\, max_{a'}\, Q(s',a') - Q(s,a)]$

During evaluation, a fully greedy policy (ε = 0) was used.

A.4 Proximal Policy Optimization (PPO) Curriculum Configuration

PPO was implemented as a stochastic policy-gradient method.

Total training timesteps: 1,200,000

- Phase 1 (navigation-focused reward): 300,000 timesteps
- Phase 2 (energy-aware reward): 900,000 timesteps

Evaluation was conducted using deterministic action selection.

PPO Objective Function

PPO optimizes the clipped surrogate objective:

$L^{CLIP}(\theta) = E_t[min(r_t(\theta)\widehat{A}_t, clip(r_t(\theta), 1 - \epsilon, 1 + \epsilon)\widehat{A}_t)]$

where

$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$

and A^t is the generalized advantage estimate.

The clipping mechanism constrains policy updates to maintain stability.

PPO Hyperparameters

- Discount factor γ=0.99
- GAE parameter λ=0.95
- Entropy regularization enabled
- Learning rate: 3×10^{-4}
- Clip range: 0.2
- Rollout buffer: 2,048 steps
- Batch size: 256
- Optimization epochs: 10
- Neural network architecture: two hidden layers (256 units each, ReLU activation)

A.5 Evaluation Protocol

For each condition (slip FALSE / TRUE):

- One trained model per algorithm
- Maximum evaluation length: 1000 steps
- Number of evaluation episodes: 200
- Number of seeds: 105
- Heatmap runs: 10
- Metrics recorded:
  - Total cumulative reward
  - Number of steps
  - Snow cell visits
  - Success rate (%)

State visitation heatmaps were generated by aggregating visitation frequencies across evaluation runs of 200 episodes each.

Appendix B Detailed Quantitative Results:

B.1 Deterministic (slip = FALSE)

Method¶	Reward¶	Steps¶	Snow Visits¶	Success¶
BFS Shortest	575.30±0.00	70.00±0.00	31.00±0.00	100.0%
Q-Learning	579.10±5.61	70.00±0.00	12.60±3.29	100.0%
PPO Curriculum	588.50±0.60	70.00±0.00	7.80±1.10	100.0%

Method	Reward (mean±SD)	95%CI	Steps	Snow Visits	Success
BFS Shortest	575.30 ± 0.00	[575.30, 575.30]	70.00 ± 0.00	31.00 ± 0.00	100.0%
Q-Learning	582.01 ± 6.97	[577.02, 587.00]	70.00 ± 0.00	13.10 ± 3.05	100.0%
PPO Curriculum	588.73 ± 0.67	[588.25, 589.21]	70.00 ± 0.00	7.60 ± 0.97	100.0%

All methods reached the goal within 70 steps.
PPO achieved the highest cumulative reward and fewest snow cell visits.

B.2 Stochastic (slip = TRUE)

Method	Reward (mean±SD)	95% CI	Steps	Snow Visits	Success
BFS Shortest	-1668.41 ± 113.42	[-1749.54, -1587.28]	1000.00 ± 0.00	246.80 ± 76.09	0.0%
Q-Learning	311.82 ± 4.83	[308.37, 315.27]	26.40 ± 3.92	66.94 ± 3.87	100.0%
PPO Curriculum	258.69 ± 10.73	[251.02, 266.37]	241.18 ± 4.11	91.88 ± 5.24	100.0%

¶

Method¶	Reward¶	Steps¶	Snow Visits¶	Success¶
BFS Shortest ¶	-1693.96±61.27	1000±0.00	246.80±76.09	0%

Q-Learning	304.03±9.92	228.18±4.10	67.58±4.34	100.0%
PPO Curriculum	252.19±7.04	242.29±3.00	92.65±4.14	100.0%

Under stochastic dynamics:

- BFS fails due to inability to adapt.
- Q-learning demonstrates greater robustness.
- PPO maintains success but exhibits higher trajectory dispersion.

B.3 State Visitation Analysis

State visitation frequency was computed as:

$$V(s) = \frac{N(s)}{max_s N(s)}$$

where $N(s)$ is the number of visits to state s.

- Under deterministic conditions, visitation concentrates along a narrow corridor.
- Under stochastic conditions, visitation becomes more dispersed.
- Q-learning exhibits more focused routing than PPO under slip conditions.

Dear Managing Editor,

Thank you for your thoughtful comments on my research paper. I have carefully addressed each of your comments. All the data results have been updated to the 10 seed results.

Comment 1:

1. While the revision introduces multi-seed experiments using 5 seeds, this could be strengthened for stochastic RL evaluation. There are also no tests for statistical significance, no confidence intervals, and no discussion of variability across runs.
    a. I would like the author to increase the experiments to at least 10 (maybe even 20) seeds and report statistics (e.g., mean +/ standard deviation, median + 95% confidence intervals, etc.). Just something of this sort would be great and help alleviate concerns about the robustness and consistency of your results.
    b. Include at least one statistical test (e.g., Welch's t-test) comparing Q-learning vs. PPO (slip condition)
    c. Alongside these tests, you should add a short paragraph discussing and interpreting the variance. In particular, I would want you to answer whether PPO exhibits higher instability

Response 1:

(a) All experiments have re-run using 10 independent random seeds (seed0-9). Table 1 and 2 now are reported with mean +-/ standard deviation, and 95% confidence intervals. Please view the result section.

(b) I have added a Welch T test table that compares Q-learning and PPO under slip conditions. The test confirmed that Q-learning significantly outperformed PPO ($t = 14.28$, $p < 0.001$). Results are reported in the new Section 4.4.1 (Statistical Significance) along with Table 3.

(c) A variance interpretation paragraph has been added. "The standard deviation gap between PPO in both conditions suggests that PPO is more sensitive to stochastic signals under slip condition, where high probability slip results in noisy advantage estimation, leading to higher variance across training runs (SD = 10.73). In contrast, under deterministic conditions PPO exhibits a much lower variance (SD = 0.67), confirming that the instability is determined by the stochastic environment, not the algorithm itself." Please view line (496-501) in 4.4.1 Welch's T-test section.

Comment 2:

The research gap still does not entirely distinguish between RL path planning literature, cost-aware navigation, and grid-world RL benchmarks. Could you add another paragraph answering more explicitly what prior work has been done, what is missing, and what exactly is novel.

Response 2:

A new paragraph has been added to Introduction that explicitly categorizes prior works into three streams. "Prior work on reinforcement learning for navigation can be categorized into three directions. First, Grid world benchmarks commonly adopt tabular Q-learning and deep reinforcement learning variants as standard baselines for evaluating discrete navigation tasks [3,5], but focus primarily on task completion rather than energy awareness and energy sensitive routing. Second, cost-aware navigation research has examined energy minimisation in uneven terrains using reinforcement learning [6], yet rarely incorporated weather dependent components, such as snow coverage or stochastic traction loss. Third, existing comparisons between value-based and policy-gradient methods [7,8], including curriculum based PPO applications [9] are conducted primarily under fixed and stable conditions, without examining the possibility of a particular reinforcement learning method remaining dominant to another when action outcomes become probabilistic, as in stochastic environments such as winter slip conditions. The work in this study addresses all three gaps simultaneously by implementing an energy-aware reward structure into a reproducible 50×50 grid-world benchmark, embedding snow-density penalties and a 2/3-slip probability to simulate realistic winter routing conditions, and directly comparing Q-learning, PPO, and a traditional baseline under both deterministic and stochastic environments." Please view line (73-89) in introduction section.

Comment 3:

Please add a plot of learning curves: training reward vs. timesteps for both Q-learning and PPO, along with the statistics from point 1 above (CIs, mean +/ stds, whatever you end up using). Would reveal instability, too, especially for PPO?

Response 3:

Learning curves have been added . Figure 13 shows mean episode reward ± 1 SD across 10 seeds under slip conditions for both methods. Please view the 4.4.2 section in results section.

Comment 4:

The paper states outcomes (e.g., Q-learning outperforms PPO) but could explain these in a deeper, more mechanistic way. Why does Q-learning work better in discrete grid worlds? Why does PPO struggle?

a. For the first question, consider: exact value iteration in finite MDP, lower variance than policy gradients

b. For the second question, consider: high variance gradients under stochasticity, issues with sparse and noisy rewards

Response 4:

(a) I have expanded Section 5.2 to address this by explaining that Q-learning benefits from exact value iteration in the finite MDP, directly converging to true optimal Q-values without approximation error, and produces lower variance than

policy gradient methods since updates are based on individual state-action pairs rather than entire trajectories [7].

(b) PPO struggles due to high variance gradients under stochasticity, where the 2/3-slip probability inflates advantage estimates across rollouts, combined with sparse and noisy reward signals that destabilize gradient updates and require significantly more training samples [3]

(c) Please view section 5.2 in the Discussion section.

Comment 5:

All results are based on a single hyperparameter configuration per method, which is concerning. How stable are the results to changes in hyperparameters? Could you add at least one sensitivity analysis? You do not need to go overboard, but this could be good to alleviate skepticism towards claims about one method being more suited over the other…

a. For Q-learning: vary the decay or learning rate

b. For PPO: vary learning or clipping parameter

c. This should be accompanied by a figure

Response 5:

A full sensitivity analysis has been added in the new Section 5.5, with Tables 4 and 5 and Figures 14–16 that includes decay and learning rate for Q-learning, and learning rate and clipping range for PPO. Please view section 5.5 in the Discussion section.

Comment 6:

Since curriculum is highlighted, it should ideally be validated. While I like how you introduced the curriculum learning setup, I am still wondering if the PPO performance

depends heavily on the curriculum. Could you compare the PPO with and without the curriculum, or at least cite some literature that would answer this question? You may want to accompany this with some sort of ablation study plot(s)

Response 6:

I have added an ablation comparison table compares between PPO with curriculum learning and without in both conditions. Please view section 3.2.6 in the Methods section.

Comment 7:

Watch the informal phrasing throughout (e.g., the use of “etc.”)

Response 7:

The use of informal word for etc. had been removed across the whole manuscript. Also, added “Sub-zero temperatures, combined with snow and ice accumulation on road surfaces, significantly increase energy consumption, resulting in reduced driving range and greater unpredictability in trip planning “to make the manuscript more formal.

Comment 8:

I would add at least 2 more papers on PPO vs. value-based methods and/or RL navigation benchmarks

Response 8:

I have added two articles that support my manuscript. Which are
[11]Pendyala, A., Atamna, A., & Glasmachers, T. (2024). Solving a Real-World Optimization Problem Using Proximal Policy Optimization with Curriculum Learning and Reward Engineering. arXiv:2404.02577.https://arxiv.org/abs/2404.02577

and

[12]Warnakulasuriya, D. A., Plosila, J., & Haghbayan, H. (2025). Energy-Efficient Path Planning in Uneven Terrains Using Adaptive Reinforcement Learning. *IEEE Conference Publication*. https://ieeexplore.ieee.org/document/11093435