Convergence
A Journal for Young Researchers

ORIGINAL RESEARCH

Received: 2-25-26; Accepted: 4-21-26;
Managing Editor: Jonas Katona;
Action Editor: Amy Nguyen

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

Sheng-Jui Yu

Moscrop Secondary School, Burnaby, British Columbia, Canada

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.

Keywords: reinforcement learning, energy-efficient path planning, grid-world navigation, Q-learning, proximal policy optimization (PPO), policy-gradient methods, Markov decision processes (MDP), autonomous navigation

1. Introduction

As demand for Electric Vehicles (EVs) becomes 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-covered 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 over 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.

Convergence
A Journal for Young Researchers

More specifically, the research will include a traditional baseline algorithm and two types of reinforcement learning, which are Q-learning and the 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.

2. Methods

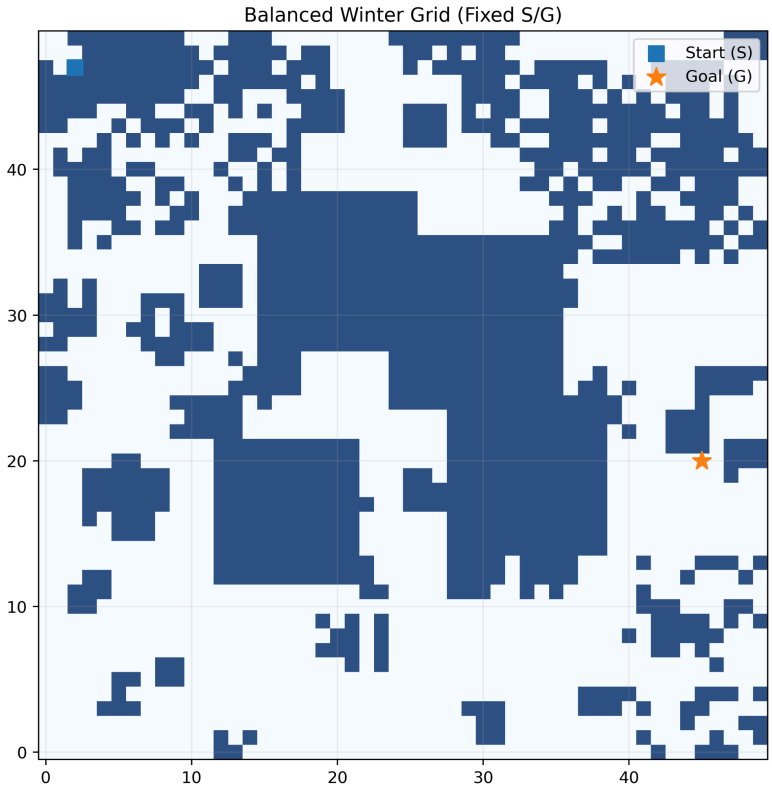

Figure 1: A typical winter grid layout used in all experiments.

A custom winter navigation grid environment was designed to evaluate each Reinforcement Learning (RL) agent under stochastic and cost-sensitive conditions, as shown in Figure 1. 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 is located in the top left corner and middle right, respectively. At each time step, the agent can take one of the four directions, up, down, left, or 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.

2.1. Environment Design

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

2.1.1. Traditional Baseline (BFS)

A 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, and 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.

2.1.2. 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 1.0 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 at 0.10 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].

2.1.3. Reward system

In this research, the reward system is designed to represent energy efficiency under winter conditions, which 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 values on snow grids. A positive 700 is given only when the agents successfully reach the destination, within a maximum of 1000 steps. It encourages agents to reach the destination with consideration for 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.

2.1.4. 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.

2.1.5. 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.

2.1.6. 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.

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

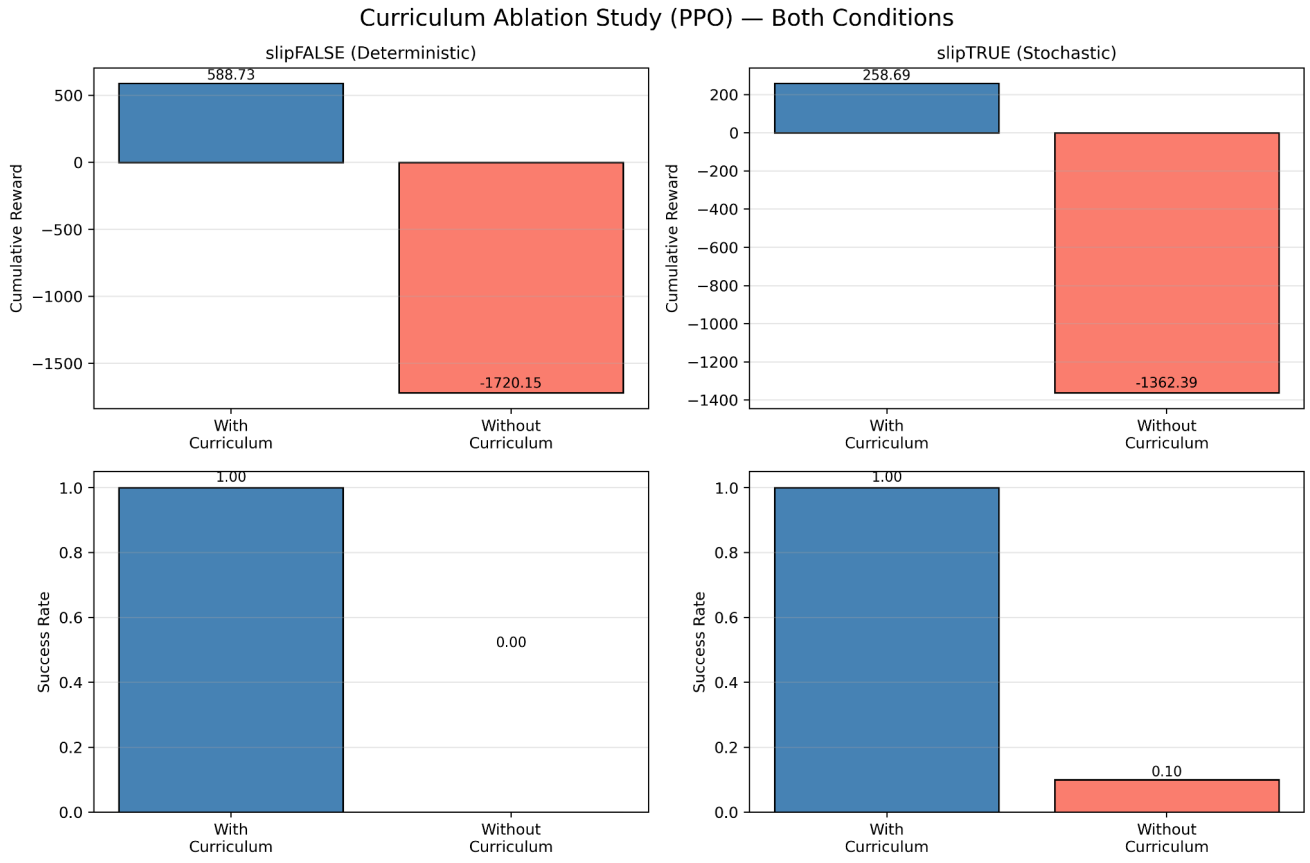

Figure 2: The cumulative reward and success rate for PPO with and without curriculum under both deterministic and stochastic conditions, illustrating the substantial performance gap between the two configurations.

2.2. 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.

2.3. Hyperparameter Selections

2.3.1. Q-Learning Hyperparameter Configuration

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

The hyperparameters are set as follows:

- Learning rate $\alpha = 0.10$

- Discount factor $\gamma = 0.99$

- Exploration strategy: ε-greedy

- Initial $\varepsilon = 1.0$

- Minimum $\varepsilon = 0.05$

- Exponential decay rate = 0.9995

- Training episodes = 15,000

- Maximum steps per episode = 1,000

The learning rate ($\alpha = 0.10$) 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 ($\varepsilon = 1.0$) 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.

2.3.2. 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.

3. 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.

3.1. 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.

3.2. Trajectory Visualizations and Quantitative Results

3.2.1. Deterministic (Non-Slip) Winter Condition

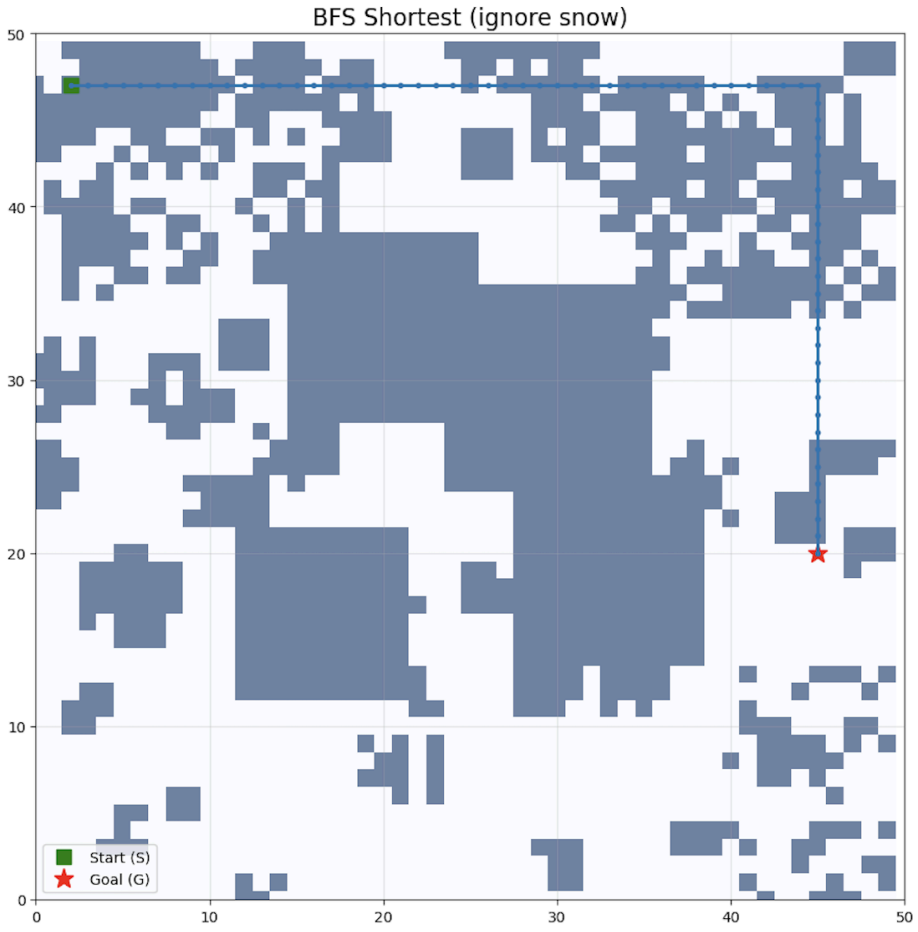

Figure 3: The BFS path under non-slip conditions.

Note: Since BFS always finds the shortest route, the trajectory goes straight from the start to the goal with minimal cells visited without any deviation.

Convergence
A Journal for Young Researchers

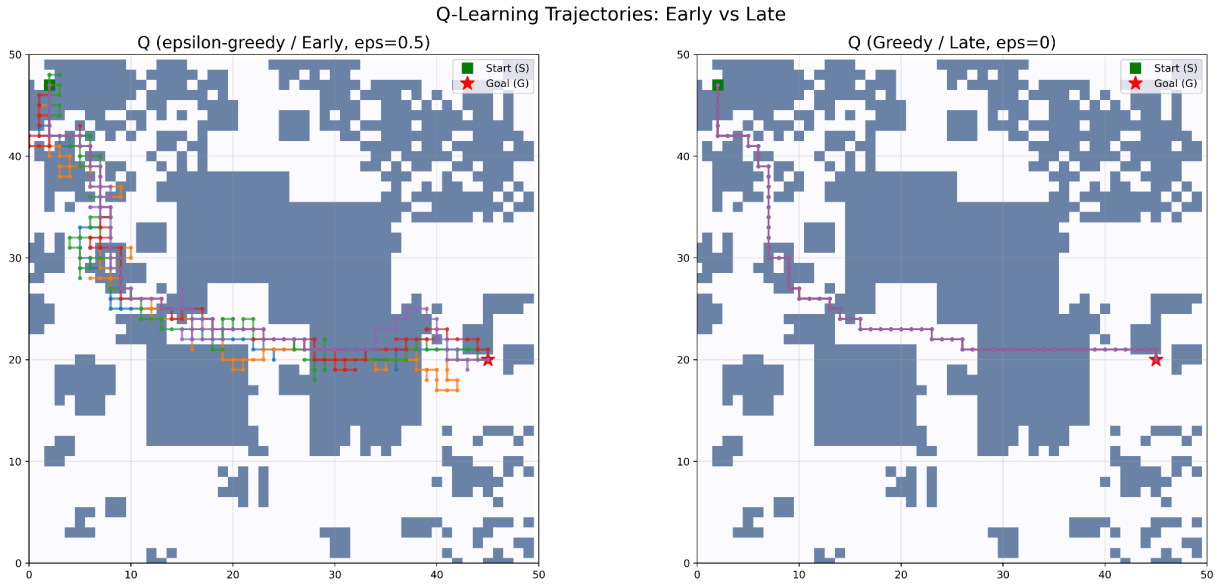

Figure 4: The navigation trajectories produced by the Q-learning agent under deterministic (non-slip) winter conditions during early training with an ε-greedy policy and late training with a greedy policy.

Note: The trajectories demonstrate the transition from exploratory behavior to a stable path toward the goal.

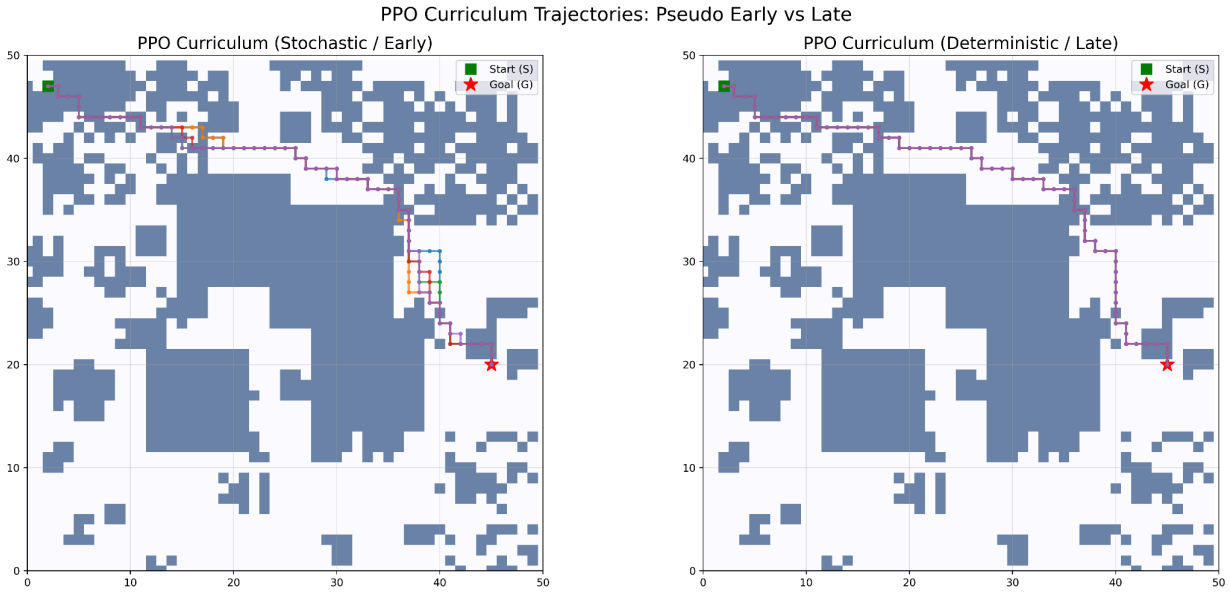

Figure 5: The navigation trajectories produced by the PPO agent during early and late stages of training under deterministic (non-slip) winter conditions.

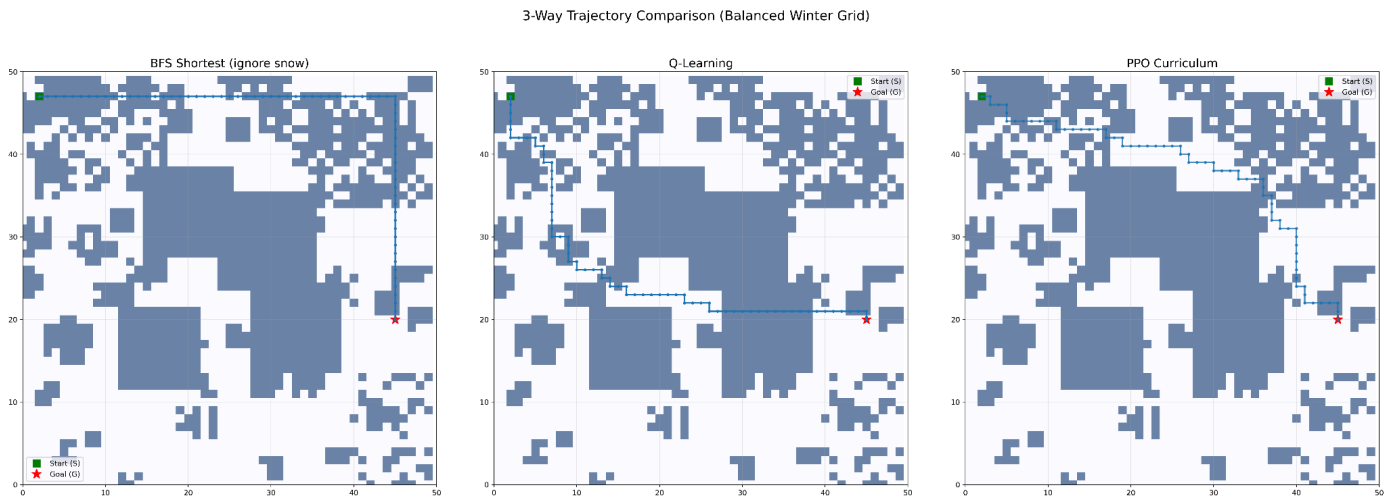

Figure 6: A comparison between the trajectories of BFS, Q-learning, and PPO in the non-slip winter setting.

Table 1: A summary of 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.

=== 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	11.60 ± 4.77	100.0%
PPO Curriculum	588.73 ± 0.67	[588.25, 589.21]	70.00 ± 0.00	7.60 ± 0.97	100.0%

Convergence
A Journal for Young Researchers

3.2.2. Slip Winter Condition

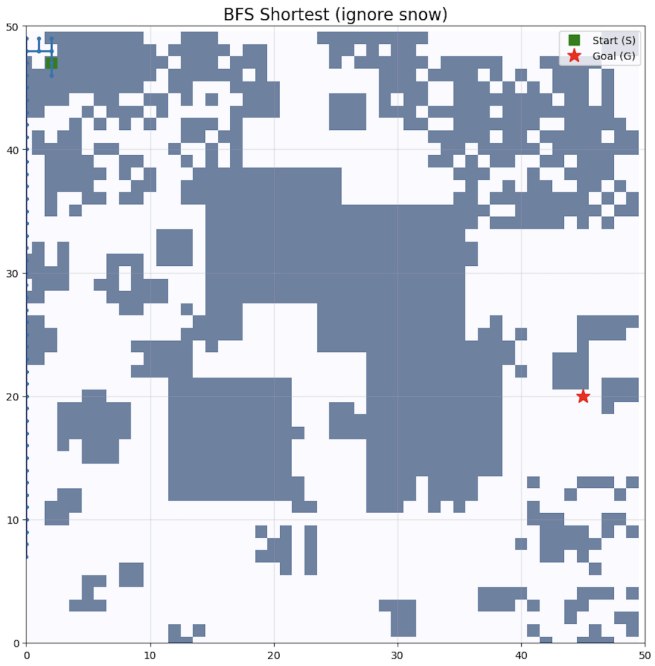

Figure 7: The navigation path produced by the BFS baseline under slip (stochastic) winter conditions.

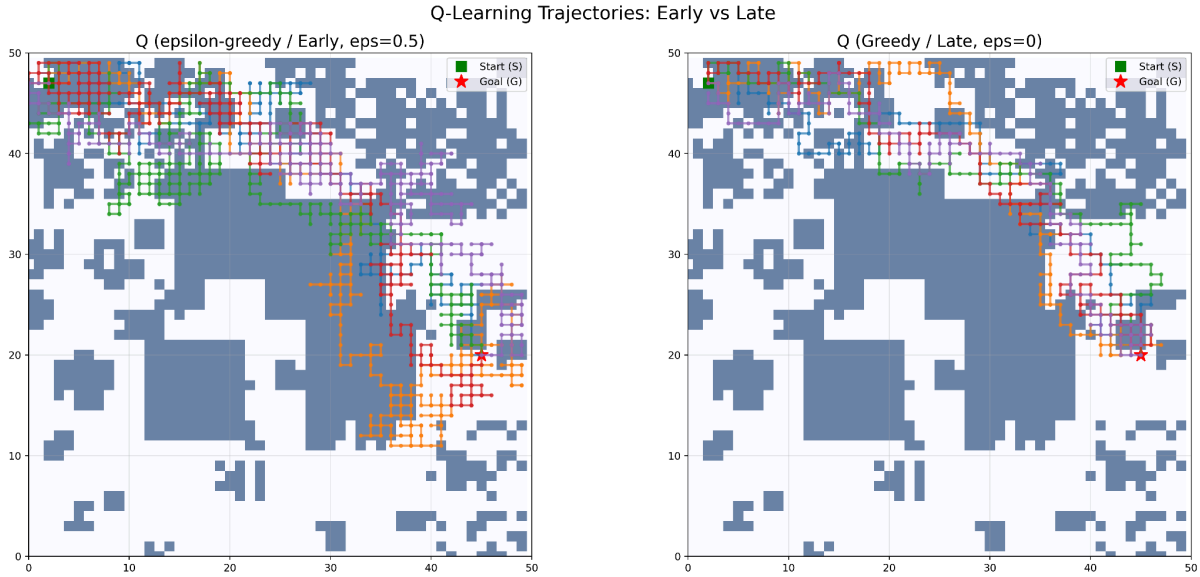

Figure 8: The navigation trajectories produced by the Q-learning agent during early and late stages of training under slip (stochastic) winter conditions.

Convergence
A Journal for Young Researchers

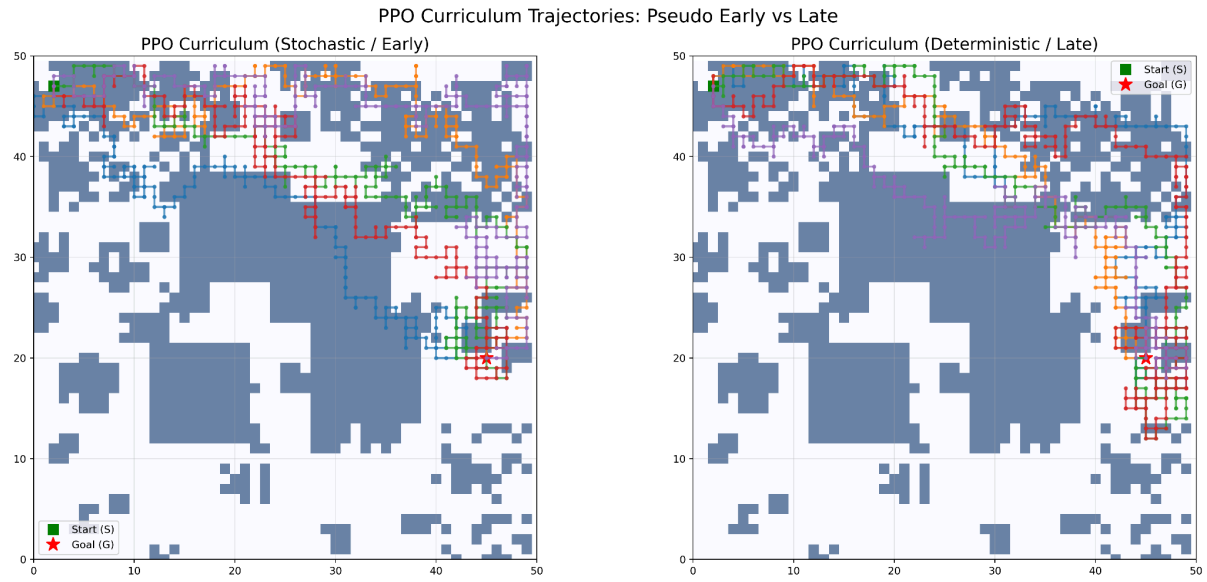

Figure 9: The navigation trajectories produced by the PPO agent during early and late stages of training under slip (stochastic) winter conditions.

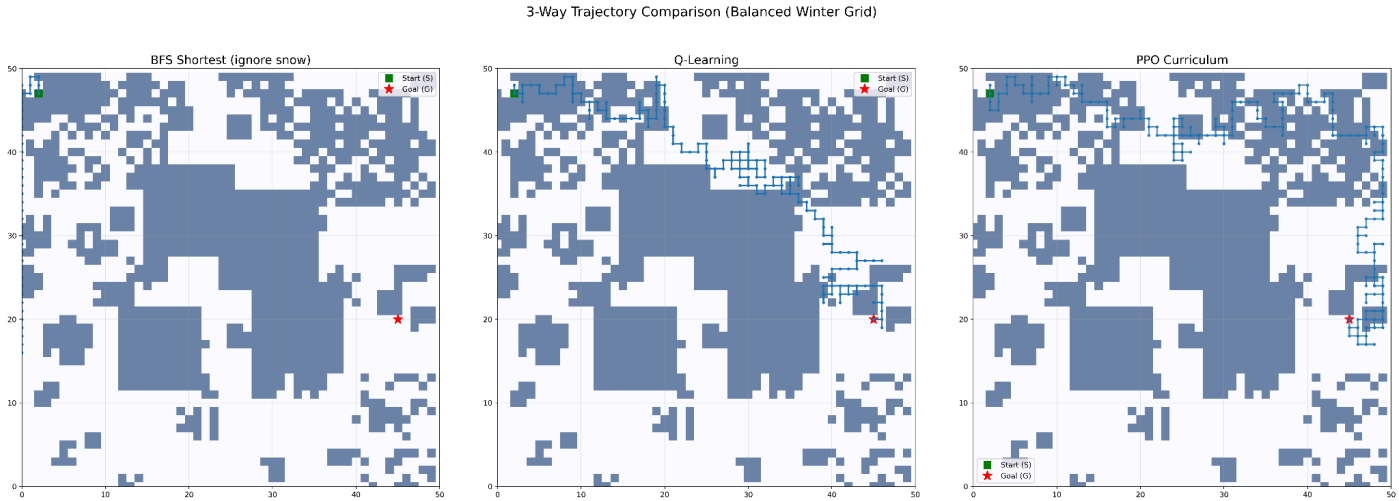

Figure 10: A comparison of the navigation trajectories of BFS, Q-learning, and PPO under slip winter conditions.

Table 2: 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.

=== 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	226.30 ± 170.62	0.0%
Q-Learning	311.82 ± 4.83	[308.37, 315.27]	225.62 ± 2.84	64.70 ± 3.03	100.0%
PPO Curriculum	258.69 ± 10.73	[251.02, 266.37]	241.18 ± 4.11	91.88 ± 5.24	100.0%

3.3. Spatial State Visitation Heatmaps

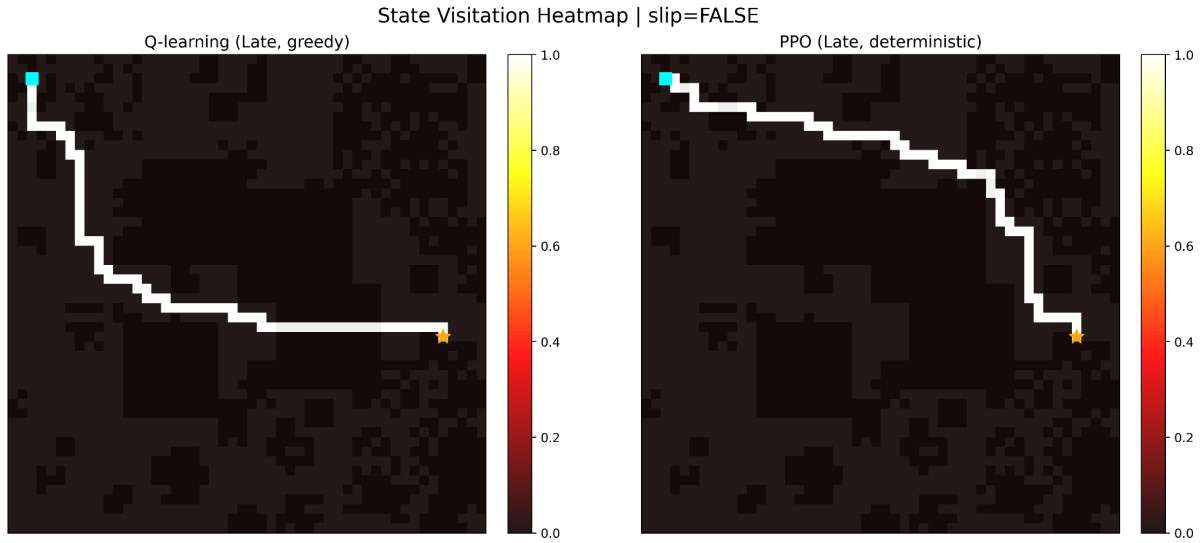

Figure 11: The visitation heatmaps for Q-learning and PPO under deterministic (non-slip) winter conditions.

Note: 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.

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.

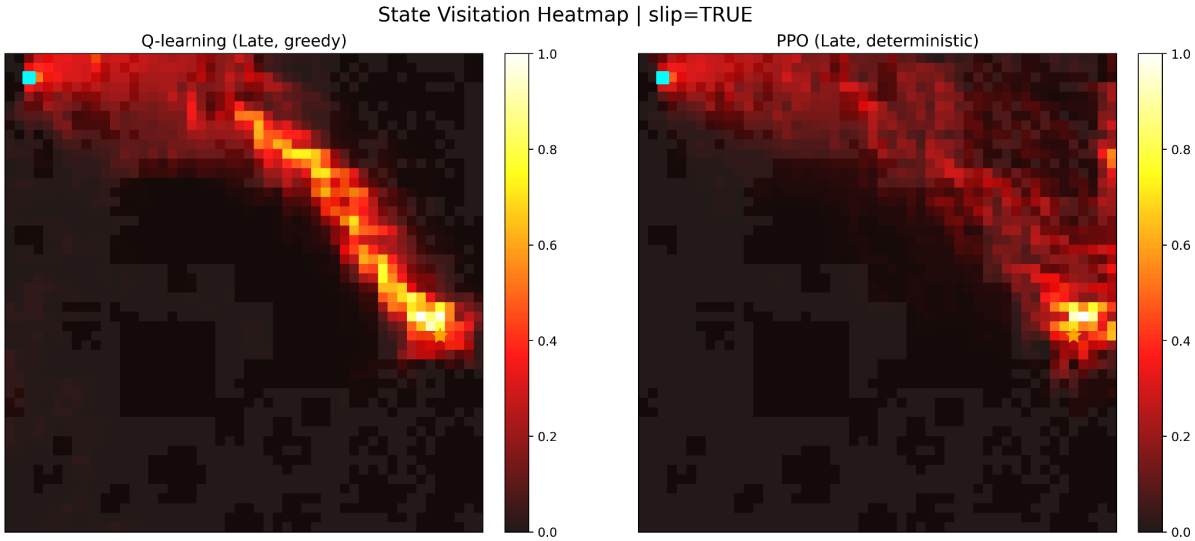

Figure 12: The visitation intensity for each grid cell in the slip case.

Note: 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.

3.4. Statistical Analysis and Learning Curves - Slip Condition

3.4.1. 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 the 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.

Table 3: Welch's t-test results comparing Q-learning and PPO Curriculum cumulative reward under slip conditions across 10 seeds.

=== 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%

Note: n/a indicates that the test was not applicable due to zero variance in success rate for both methods.

The standard deviation gap between PPO in both conditions suggests that PPO is more sensitive to stochastic signals under the 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.

3.4.2. 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.

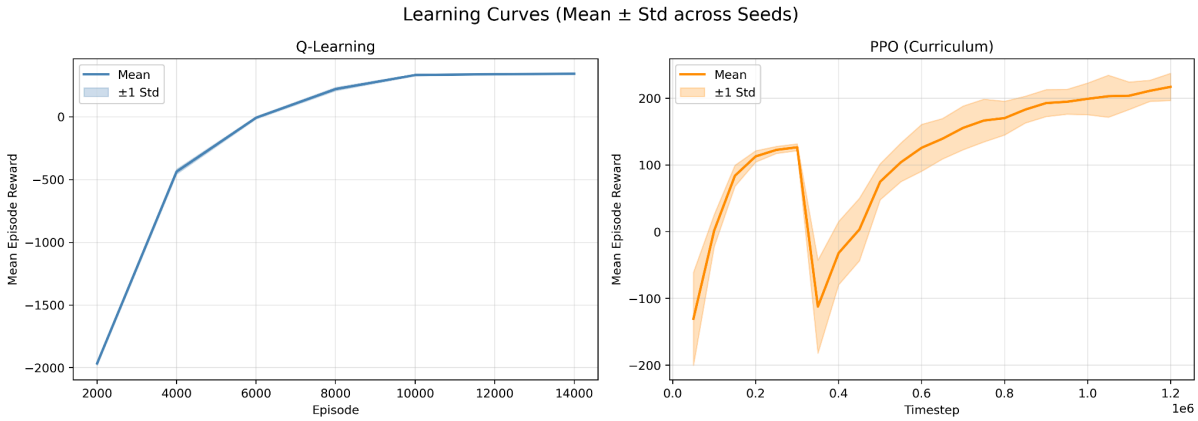

Figure 13: Learning curves (Mean ± SD across 10 seeds) under slip (stochastic) conditions.

Note: 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.

3.5. 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.

4. Discussion

4.1. 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 a 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 than traditional BFS across these environments.

4.2. Interpretation of Q-Learning and PPO Performance

The results showed that algorithm performance varied depending on the environmental conditions. 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 a relatively small environment would be more favorable 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 the 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].

4.3. 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].

4.4. Interpretation of BFS Baseline Behavior

The Traditional baseline trajectories did not take into account 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, it is the least effective method.

4.5. 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.

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

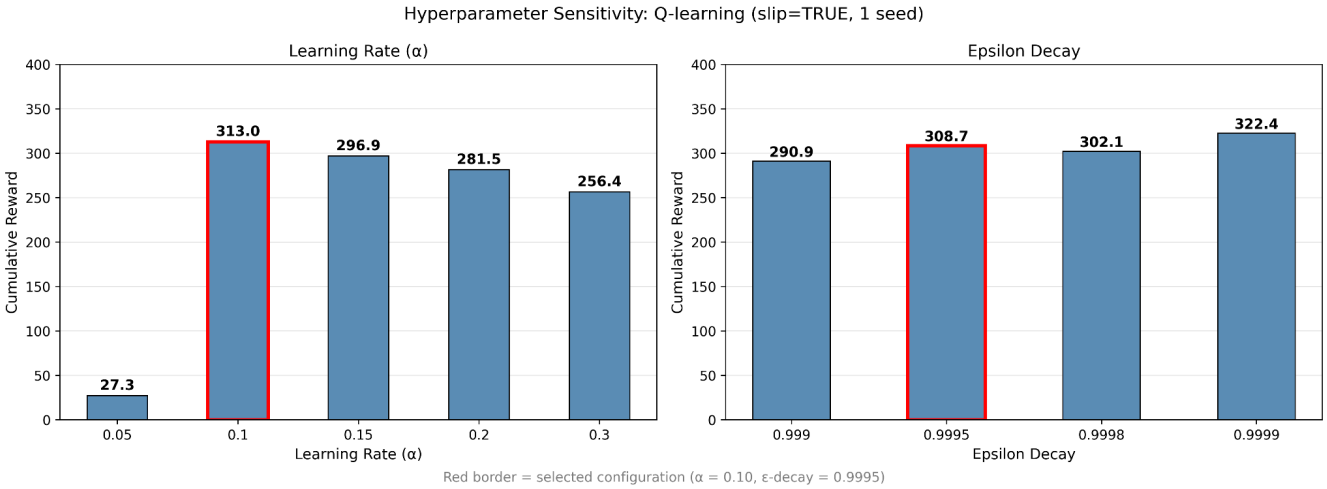

Figure 14: Comparison of different learning Rate (alpha) values and epsilon decay values.

Note: The highlighted values represent the selected hyperparameter configuration, chosen based on their balance of performance and training stability. (α = 0.10, ε-decay = 0.9995)

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

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

Parameter	Value	Success rate (%)	Cumulative reward	Note
Learning rate (alpha)	0.05	98.00	27.30	
Learning rate (alpha)	0.10	100.00	313.00	Selected configuration
Learning rate (alpha)	0.15	100.00	296.90	
Learning rate (alpha)	0.20	100.00	281.50	
Learning rate (alpha)	0.30	100.00	256.40	

Epsilon decay	0.9990	100.00	290.90	
Epsilon decay	0.9995	100.00	308.70	Selected configuration
Epsilon decay	0.9998	100.00	302.10	
Epsilon decay	0.9999	100.00	322.40	

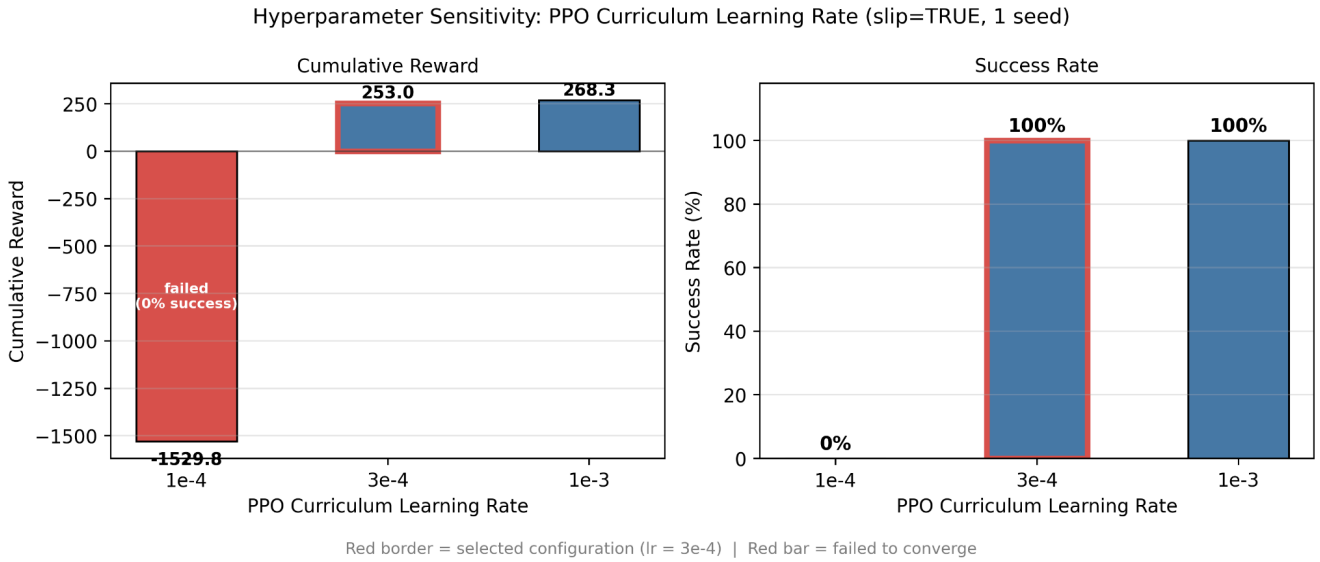

Figure 15: Comparison of different values of learning rate, and the highlighted values represent the selected hyperparameter configuration, chosen based on their balance of performance and training stability

Note: 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.

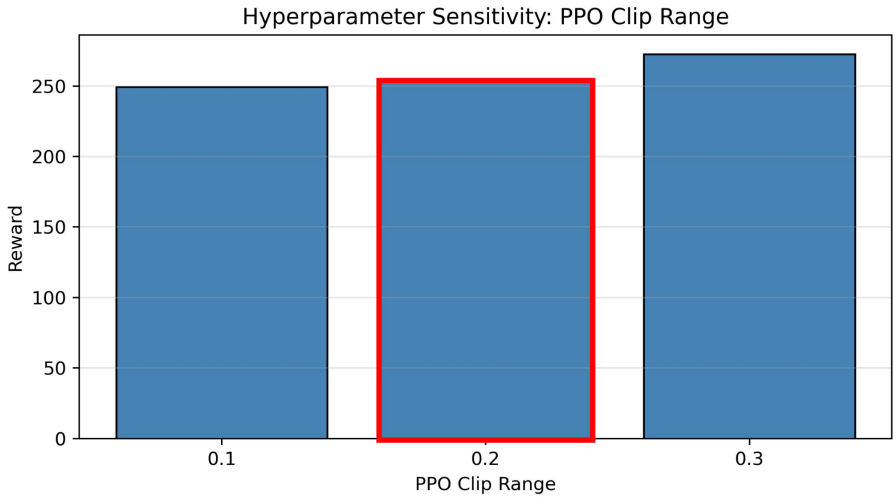

Figure 16: Comparison of different values of PPO clip range, and the highlighted values represent the selected hyperparameter configuration, chosen based on their balance of performance and training stability. (Clip range = 0.2)

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

===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	

4.6. 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 such as Q-learning, as 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, the generalization of findings to real-world winter navigation scenarios was 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 is better to reflect real-world uncertainty, which would allow policy-gradient methods such as PPO to fully function their theoretical advantages in handling stochastic and high-dimensional control problems.

5. 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.

6. References

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

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

Sutton, R. S., & Barto, A. G. (2018). Reinforcement learning: An introduction (2nd ed.). MIT Press. http://incompleteideas.net/book/the-book-2nd.html

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 (arXiv:1611.03673). arXiv. https://doi.org/10.48550/arXiv.1611.03673

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

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

Watkins, C. J. C. H., & Dayan, P. (1992). Q-learning. Machine Learning, 8, 279–292. https://doi.org/10.1007/BF00992698

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms(arXiv:1707.06347). arXiv. https://doi.org/10.48550/arXiv.1707.06347

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). arXiv. https://doi.org/10.48550/arXiv.2404.02577

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

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

Chukwurah, N., Adebayo, A. S., & Ajayi, O. O. (2024). Sim-to-real transfer in robotics: Addressing the gap between simulation and real-world performance. International Journal for Multidisciplinary Research, 5(1), 33–39. https://doi.org/10.54660/.IJFMR.2024.5.1.33-39

7. 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:

1. Deterministic (slip = FALSE):
    - The intended action is executed exactly.
2. 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=0.10
- Discount factor γ=0.99

- Exploration strategy: ε-greedy
- Initial ε = 1.0
- 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:

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

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):

8. One trained model per algorithm
9. Maximum evaluation length: 1000 steps
10. Number of evaluation episodes: 200
11. Number of seeds: 10
12. Heatmap runs: 10
13. Metrics recorded:
    a. Total cumulative reward
    b. Number of steps
    c. Snow cell visits
    d. 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	11.60 ± 4.77	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 the fewest snow cell visits.

B.2 Stochastic (slip = TRUE)Under stochastic dynamics:

Method	Reward (mean±SD)	95% CI	Steps	Snow Visits	Success
BFS Shortest	-1668.41 ± 113.42	[-1749.54, -1587.28]	1000.00 ± 0.00	226.30 ± 170.62	0.0%
Q-Learning	311.82 ± 4.83	[308.37, 315.27]	225.62 ± 2.84	64.70 ± 3.03	100.0%
PPO Curriculum	258.69 ± 10.73	[251.02, 266.37]	241.18 ± 4.11	91.88 ± 5.24	100.0%

- BFS fails due to the 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 state visits 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.

Code & Data Availability Statement

All source code is publicly available at:

https://github.com/Aiden-123-2/Energy-Efficient-Path-Planning-in-Winter-Conditions-

Acknowledgements

The author would like to express sincere gratitude to Dr. Eric Saak for his mentorship and guidance throughout this research. His willingness to share his expertise and provide continuous support was instrumental in the development of this study. The author acknowledges the teachers at Moscrop Secondary School for their instruction in computer science, mathematics, and robotics, which established the foundational knowledge that informed this work. Gratitude is also extended to the peer reviewers for their constructive feedback and thoughtful suggestions, which improved the clarity and quality of this manuscript, and to the managing editors of *Convergence Journal* for their guidance throughout the publication process. Finally, the author extends gratitude to his parents for their unwavering support and encouragement throughout this research journey.

Author Biography

Sheng-Jui Yu is a Grade 10 student researcher at Moscrop Secondary School in British Columbia, Canada. His research interests lie in reinforcement learning, with a current focus on energy-efficient path planning under environmental uncertainty. This work is motivated by real-world challenges in autonomous navigation, where mobile robots and unmanned systems must operate reliably under harsh environmental conditions and limited energy constraints.

He has developed practical experience in electronic circuit design, Arduino programming in C++, and digital hardware development using Verilog on FPGA platforms through independent experimentation and project work. He is the founder and president of his school's Hardware Innovation Club, where he organizes workshops and events and leads collaborative hardware system projects that bring together members across grade levels.

He participates actively in mathematics competitions organized by the University of Waterloo and mentors peers in mathematics. He plans to pursue undergraduate studies in a related field. He aims to contribute to the development of intelligent systems in artificial intelligence and robotics that address real-world challenges in automation, sustainability, and human well-being.

Mentor Contribution Statement

Dr. Eric Saak guided the research process, offering feedback on the study design and reinforcement learning methodologies. His mentorship contributed to improving the clarity and structure of the manuscript.