Maze Shaping Experiment: Configuration Notes

February 13, 2026

1 Purpose

This document records the exact experiment settings used in `run_maze_shaping_experiment.py`, so the run logic is explicit and reproducible.

2 Environment (Maze MDP)

State space. Grid coordinates (r, c) over a binary maze map.

Action space. Four cardinal actions:

$$\mathcal{A} = \{\text{up}, \text{down}, \text{left}, \text{right}\}.$$

Start/goal.

- Start: $(1, 1)$
- Goal: $(h - 2, w - 2)$ where (h, w) is maze size

Transition stochasticity. With probability 0.2, intended action is replaced by a random action (slip).

Collision handling. If next cell is wall or out-of-bounds, the agent stays in place.

Base reward and termination.

- Step reward: $R(s, a, s') = -1$
- Episode ends when goal is reached, or after `max_steps`

3 Shaping Formulation

For potential scale $\kappa \in \{0, 0.5, 1.0\}$, define

$$\Phi_\kappa(s) = \kappa \Phi_0(s),$$

and shaped reward

$$R'_\kappa(s, a, s') = R(s, a, s') + \gamma \Phi_\kappa(s') - \Phi_\kappa(s).$$

Conditions.

- `no_shaping`: $\kappa = 0$
- `phi_half`: $\kappa = 0.5$
- `phi_full`: $\kappa = 1.0$

Potential distance type.

- `bfs`: $\Phi_0(s) = -d_{\text{BFS}}(s, g)$ (wall-aware shortest path)
- `manhattan`: $\Phi_0(s) = -(|r - r_g| + |c - c_g|)$ (wall-agnostic)

4 Learning Algorithm (Tabular SARSA)

Behavior policy. ϵ-greedy with tie-randomization.

Update rule (non-terminal):

$$Q(s,a) \leftarrow Q(s,a) + \alpha\Big(r' + \gamma Q(s',a') - Q(s,a)\Big).$$

Terminal update:

$$Q(s,a) \leftarrow Q(s,a) + \alpha\Big(r' - Q(s,a)\Big).$$

Q-table shape. $(h, w, 4)$.

5 Validation Protocol

Validation is run during training every `validation_interval` episodes.

- Policy for validation: greedy ($\arg\max_a Q(s,a)$)
- Validation rollouts per checkpoint: `validation_episodes`
- Metrics:
  - success rate
  - average steps-to-goal

6 Run Aggregation and Outputs

For each condition, training runs are repeated with different seeds and aggregated.

Saved artifacts per run directory:

- `learning_curve.csv`, `learning_curve.png`
- `validation_progress.csv`, `validation_progress.png`
- `run_summary.json`
- `gifs/policy_rollout_ep_*.gif` at 0%, 25%, 50%, 75%, 100%

7 Default CLI Configuration

8 Notes

Potential-based shaping is policy-invariant in theory (Ng et al., 1999), but finite-sample learning behavior can still differ substantially depending on how informative Φ is for the maze topology.

Argument	Default
`--maze-path`	`.../maze_00.npy`
`--episodes`	500
`--runs`	12
`--alpha`	0.02
`--epsilon`	0.10
`--gamma`	1.0
`--max-steps`	350
`--validation-interval`	25
`--validation-episodes`	30
`--seed`	7
`--potential-distance`	`bfs`

Table 1: Default settings in the experiment script.