Predators and Prey: Reinforcement Learning in a GridWorld Environment

Valentina Alexeeva

December 5, 2025

1 Version Overview

1.1 Version 1: Single Agent

- One predator agent on mini-maps (20×20) with moving targets.
- Trained using DAgger: the network action is mixed with teacher advice (`ClosestTargetAgent`) using a decaying β schedule; the network is then evaluated in *greedy* mode.
- Compact feature representation: state "head" + local $P \times P$ patch around the agent + K nearest targets with toroidal normalization $(\Delta y, \Delta x, \Delta n)$.
- Reward is decomposed into components (base, capture, exploration, standing still, revisits) with careful shaping based on the change in shortest path distance (Δ via BFS), without overwhelming the base signals.

  Detailed logs and consistent heatmaps/renderings confirm correct geometry and pursuit dynamics. This version serves as a "sanity benchmark" for DAgger before scaling up to multi-agent scenarios. A bright spot on the heatmap reflects agent wandering between a few points.

1.2 Version 2: Multi-Agent Environment

- **Setup:** Multi-agent control of a team of five predators in maze-like maps with moving targets; discrete actions; toroidal patches used in input features.
- **Features:** For each agent: compact state "head", local $P \times P$ passability patch, K nearest targets $(\Delta y, \Delta x, \Delta n)$, and K nearest teammates (excluding self) with toroidal normalization; the feature vector length is fixed regardless of map size.
- **Model:** Lightweight MLP shared across all predators with two heads: policy (action logits) and value; inference modes include *greedy* and *sample*.

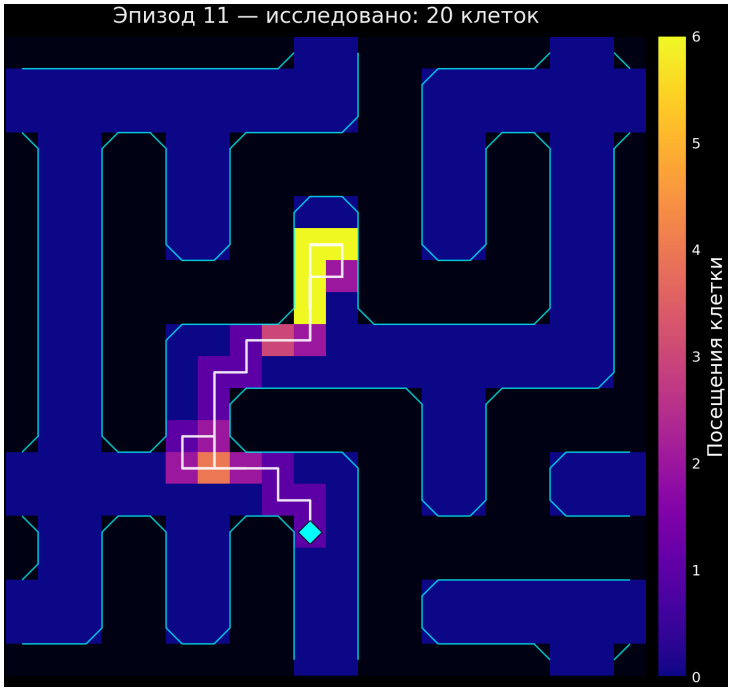

Figure 1: Visit heatmap, single predator.

- **Reward Decomposition:** Base step penalty, capture, exploration of new cells; soft penalties for standing still and revisits; capped shaping potential based on Δ of shortest path to nearest target (via BFS); behavioral regularizers to prevent crowding: `repulse` (anti-clumping, capped), `flip-flop`, and `same-dir-close`.
- **Training:** DAgger for multi-agent setup: each step uses a batch of behavioral cloning on teacher labels (`ClosestTargetAgent`) with scheduled action mixing ($\beta_{start} \rightarrow \beta_{end}$); followed by a brief RL stage with entropy reduction and early stopping based on median first-capture step. Best checkpoint is selected via *greedy* metrics.
- **Constraints:** Focus on stable behavior and coordination of the five-agent team on small to medium-sized maps.

1.3 Version 3: Large Maze and Sparse Targets

- **Environment Parameters:** Map size 40×40, number of targets: 100, step limit: 300.
- **Reward Decomposition:** Base penalty `r_base`, capture `r_capture`, exploration `r_explore`, soft penalties for standing still `r_standstill` and revisit `r_revisit`; limited-weight shaping `r_bfs` via Δ of shortest path to nearest target (BFS, no torus).
- **Training and Evaluation:** DAgger with teacher labels (`ClosestTargetAgent`) and scheduled action mixing; periodic rendering. Evaluation via `eval_network_only_multi` in *greedy* mode for reporting metrics: steps, captures, step of first capture.

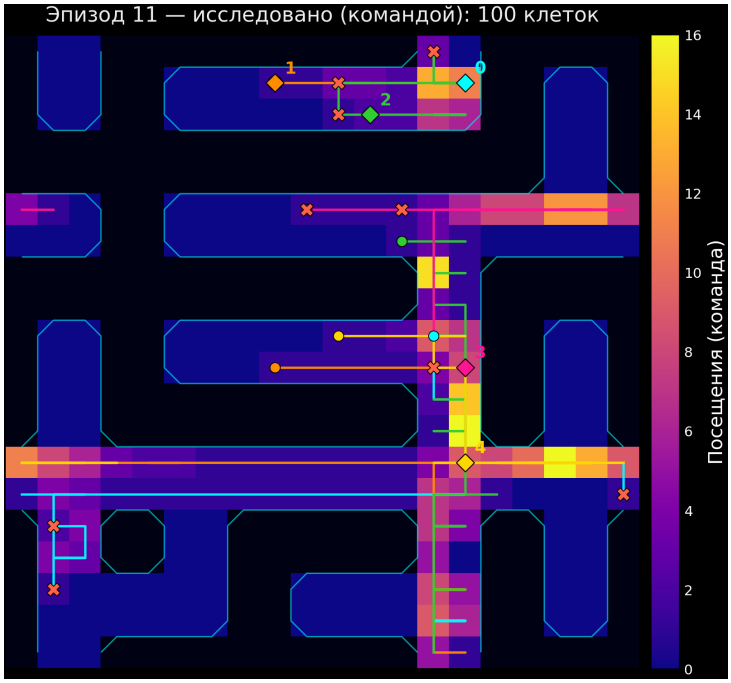

Figure 2: Visit heatmap, Version 2, 5 predators.

- **Limitations:** Main issue in large mazes is the maze layout itself. Teacher policy (`ClosestTarget`) is imperfect. In narrow corridors, predators tend to crowd and agents learn this behavior, sometimes remaining on the same cell for several steps. Visit maps show that agents frequently pursue the same targets instead of dispersing, reducing capture efficiency. Also, frequent target switching leads to oscillatory movement between adjacent cells ("flip-flop").

1.4 Version 4: New Teacher and Smoother Behavior

- **Features and Agent:** Updated `FeatureBuilder`: local $P \times P$ patch, K nearest targets $(\Delta y, \Delta x, \Delta n)$ *plus* K nearest teammates using the same triplet. Shared MLP agent `NetAgentShared` + small BC replay buffer for stability.
- **Hysteresis-based Teacher:** Replaced `ClosestTargetAgent` with `AssignedClosestTargetAgen` featuring *unique target assignment*, *target holding* for a fixed number of steps, and switching only with a distance margin improvement (`switch_margin`). Reduces oscillation and crowding.
- **Anti-clumping and Movement Smoothness:**
  - Introduced dynamic `repulse`: penalties for inter-agent distances of 0/1/2 scaled with map clearing; added `pair_stick_penalty` for agents sticking too close step after step.
  - Added `flipflop_penalty` (A→B→A) and `same_dir_close_penalty` (same movement vector at close distance).
  - Early exploration boosted via `early_steps_boost`, `early_explore_scale`.

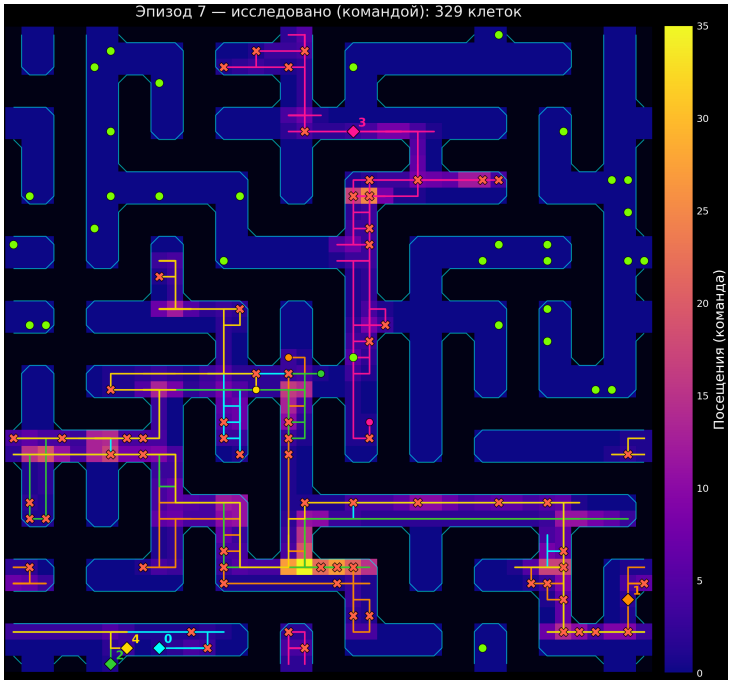

Figure 3: Visit heatmap, Version 3, 5 predators. Trajectory overlap, crowding, and unvisited areas.

- **DAgger and Replay:** Revised `train_dagger_multi`: geometric β schedule ($\beta_{start} \rightarrow \beta_{end}$), each step includes BC on teacher labels *plus* one replay step.
- **Evaluation and Best Checkpoint:** Added `eval_network_only_multi` with a best-agent criterion (captures↑, cluster↓, idle↓, first_cap↓) and auto-saving to `logs/checkpoints` (`agent_best_eval.pkl`, `agent.pkl`).

2 Final Agent Model

- **Combatting Idleness:** The original harsh penalty for inactivity was replaced by a soft, conditional one—applied only when there was no progress in team potential (sum of toroidal L1 distances to the nearest targets). This reduced useless stalling without harming interception behavior.
- **Flip-Flop Suppression:** A small penalty for oscillation (A→B→A) was introduced along with lower entropy during final optimization. Combined with target assignment hysteresis, this reduces repetitive moves between adjacent cells.
- **Clumping and Corridor Congestion:** A dynamic `repulse` mechanism with capped influence per agent and delayed activation was applied. Penalty added for close-aligned movement vectors (`same-dir-close`). Result: fewer "marching in sync" behaviors, better coverage.
- **Size-Invariant Features:** Local $P \times P$ patch and K nearest entities $(\Delta y, \Delta x, \Delta n)$ ensured the learned policy scaled from mini-maps to 40×40 without relying on global "flat" inputs.

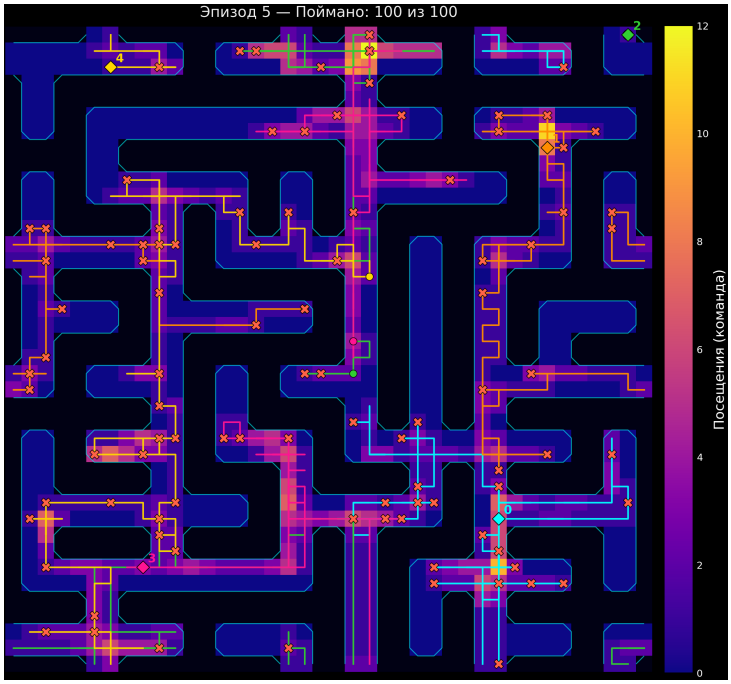

Figure 4: Visit heatmap, Version 4, 5 predators. All prey captured.

- **Improved Teacher and DAgger:** Replaced `ClosestTarget` with a teacher using target assignment, retention, and switching hysteresis. Sticky-DAgger with decaying β and small replay stabilized training across mixed static/dynamic map types.
- **Reward Shaping Constraints:** The Δ-BFS component remained weight-limited and functioned as a directional guide only. Core signals—capture and exploration—retained dominance to avoid proxy optimization.
- **"Best" Agent Criterion:** Greedy-mode checkpoint selection based on a compound metric: total captures ($\uparrow$), median step of first capture ($\downarrow$), clustering, idleness, and flip-flopping ($\downarrow$). This aligns directly with deployment-time behavior.
- **Final Policy:** Shared MLP on local features, with behavioral regularization (idle, flip-flop, repulse, same-dir-close) and target hysteresis. Demonstrated robustness on large mazes, winning 45% of PvP matches against `ClosestTarget` when distractor signals (hunter capture, bonuses) were removed.
- **Lite Version for Submission:** `NetAgentSharedLite` (reduced depth/width of MLP, same inputs and masks) was fine-tuned for runtime limits. While 5% weaker in PvP win rate, it remained behaviorally comparable.

3 PvP Observations and Submission Model

- **PvP vs. ClosestTarget:** In the VersusBotEnv PvP configuration, the learned policy wins **40%** of matches. Win rate improves when reward components for capturing other hunters and collecting bonuses are removed, as these distract from efficient pursuit.

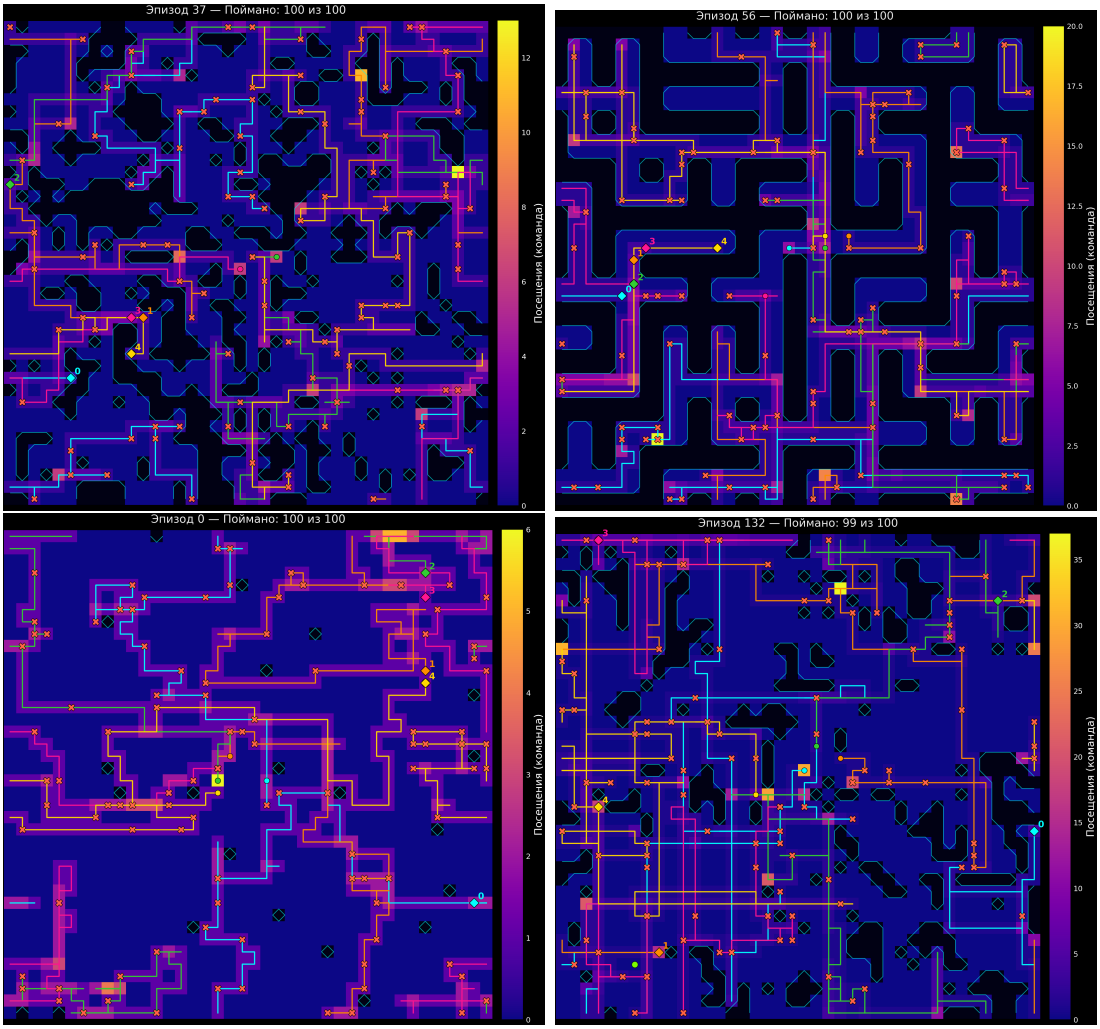

Figure 5: Agent visit heatmaps on different maps, final version.

- **Lite Version for Submission:** `NetAgentSharedLite` shares the same input structure ("head" + local $P \times P$ patch + K nearest targets/teammates, with action masks), but with a smaller MLP architecture for faster inference compatible with evaluation time limits.
- **Post-Tuning Comparison:** Final tuning applied only to the Lite version; it trails the full model by **5%** in PvP win rate, with negligible differences in other behavioral metrics.

4 Conclusions

- Major behavioral issues—idleness, flip-flop moves, corridor clumping—were addressed via soft conditional penalties, target hysteresis, dynamic `repulse`, and directional movement regularizers.
- The local feature design proved resilient to increased map sizes and mixed map types. No global flat representations were used.

- Δ-BFS shaping was controlled in weight to avoid proxy effects, maintaining healthy capture and exploration dynamics.
- The final model showed consistent performance in greedy mode and reached ˜45% win rate against scripted `ClosestTarget` in PvP, even without heavy exploit features.
- The Lite agent retained key behaviors after tuning and passed all system constraints, making it suitable for submission.

The resulting policy is robust for multi-agent pursuit on large maze maps with controlled coordination, minimized anti-patterns, and reproducible metrics in greedy mode.