WHEN DO OPTIONS HELP? POLICY NECROSIS AND REDUNDANT COVERAGE IN OPTION-CRITIC

Anonymous authors
Paper under double-blind review

ABSTRACT

Option-critic learns options: sub-policies together with a learned rule for when each one hands control back. Its headline result is that performance improves as options are added. We explain that result, with theory and experiment. First, the termination rule option-critic learns by maximising return contributes nothing. When the termination test and the policy that picks options read the same values, the test fires at every step, so the learned rule is identical to always terminating. When that policy explores and the test does not, as in option-critic itself, the rule can block the exploration; there are instances where it suffers $\Omega(T)$ regret while always terminating holds to $O(\log T)$. Forcing termination at every step leaves the option-count curve intact. Second, the policy inside an option barely explores at all, so a state locks onto the first action that looked good and never updates again. We name this policy necrosis, give a state-level test for it, and find three fifths of states necrotic in a typical option. Restoring exploration repairs those states, and one option then solves the task. Third, extra options improve no option; what falls is the chance that all of them fail in the same state, from 59% to 4%, and performance follows that joint quantity.

1 INTRODUCTION

The options framework gives reinforcement learning agents temporally extended actions: an option is an intra-option policy, an initiation set, and a termination condition (Sutton et al., 1999; Precup, 2000). Option-critic made the framework practical by differentiating the discounted return with respect to both the intra-option policy parameters and the termination parameters, so that the control objective itself shapes the options instead of a designer handing them to the agent (Bacon et al., 2017). Its tabular demonstration is FourRooms, and the demonstration is a scaling result: with more options the agent reaches the goal faster, and the effect grows with the difficulty of the dynamics. That result has been reproduced, extended to continuous control, and built upon many times over (Klissarov et al., 2017; Harb et al., 2018; Riemer et al., 2018; Zhang & Whiteson, 2019; Khetarpal et al., 2020).

What the scaling result licenses is less clear than it looks. Three things move together when the number of options K grows. Options are temporally extended, so a larger option set changes the temporal structure of behaviour. Options are separate function approximators, so a larger option set changes the capacity and the optimisation geometry of the lower level. And options are chosen by a learned router, so a larger option set changes how much of the decision burden sits above the primitive actions. A curve of performance against K cannot separate them, and reading it as evidence for temporal abstraction assumes the other two do nothing.

This paper separates them, and the answer is not temporal abstraction. We manipulate one factor at a time — the termination rule, the lower-level exploration rate, and K — across several transition kernels, two reward definitions, and a tabular and a neural implementation, and we read performance off frozen checkpoints under a protocol that gives every condition the same zero-exploration test policy. Figure 1 lays out the environment, the agent and the hypothesis; three results follow, and together they form a single account of what option-critic is doing.

The termination function that the termination gradient learns is not a source of performance. Our target is specific: the termination rule that option-critic obtains by maximising return, whose fixed point terminates exactly when continuing is worth less than handing control back. We prove that this rule is either vacuous or harmful. When the termination test and the option selector read the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

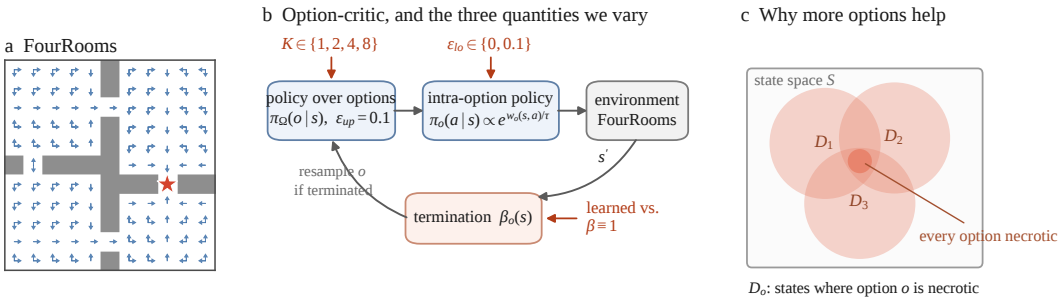

Figure 1: The benchmark, the agent, and the hypothesis. (a) FourRooms with the goal in the east doorway (star) and, for every non-goal cell, the complete set of actions that reduce the wall-respecting distance to the goal; 65 of the 103 cells admit two. (b) The option-critic loop and the three quantities we manipulate; everything else stays at the values of the original work. (c) An option is necrotic in a state when its policy has committed there to an action no optimal policy would take, so option multiplicity acts on $\bigcap_o D_o$ and not on any individual D_o.

same values, the test fires at every state, so the learned rule is pathwise identical to $\beta \equiv 1$ and adds nothing. When the selector explores while the critic feeding the test does not, the rule can veto the exploration the selector asks for; we exhibit instances on which the return-maximising rule suffers $\Omega(T)$ regret, while $\beta \equiv 1$ holds to $O(\log T)$ on every instance of the class. The experiments match the theory. Forcing $\beta \equiv 1$, which destroys temporal extension outright, leaves the whole dependence on K untouched, and the measured effect of learned termination reverses sign as K grows.

Option-critic's lower level suffers a state-local learning failure, and we can detect it. The original algorithm runs its intra-option policies as a softmax at temperature 10^{-3} with no lower-level exploration, and with learning rate 0.25 that multiplies every advantage by 250 before it reaches the logits. A state commits to whichever action was first sampled with a positive advantage; the score function then vanishes, and the state never updates again. We call this *policy necrosis*, and we give a state-level test for it that needs only the set of actions an optimal policy would accept. Roughly half of all states are necrotic in a typical option. Restarting lower-level exploration from a shared checkpoint repairs them: a single option goes from 306.7 to 55.4 expected steps under identical zero-exploration evaluation, and improves on all 350 seeds. Two different exploration mechanisms reproduce the repair in the neural implementation.

What extra options provide is redundant coverage. Adding options repairs no individual option — the mean single-option necrosis rate holds near 60% at every K — but the chance that *every* option is necrotic in the same state falls from 59% to 4%, and performance follows that joint quantity and not the individual one. Necrotic sets overlap more than independence predicts and far less than a common cause would, so a pool of K options is worth about $K/2$ independent attempts at covering the state space. Once joint coverage exists the residual error moves upstairs, to a router that picks a necrotic option while a viable one is available. And once exploration has repaired the lower level, further options stop returning anything and start costing.

The three results compose. Necrosis is the failure, option multiplicity is a workaround that hides it, lower-level exploration is a repair that removes it, and the termination gradient plays no part. Beyond the specific verdict on option-critic, the paper contributes a diagnostic that is cheap wherever an acceptable-action set is available, a theoretical account of why a return-maximising termination rule cannot be a source of gains, and an evaluation protocol — freeze, disable exploration everywhere, solve exactly — under which claims about what a hierarchical agent has learned can be settled instead of argued.

2 Setup

Option-critic. An option o consists of an intra-option policy $\pi_o(a \mid s)$ and a termination function $\beta_o(s)$; a policy over options $\pi_\Omega(o \mid s)$ picks the next option whenever the current one terminates (Sutton et al., 1999). Option-critic learns all three from the return (Bacon et al., 2017). In the tabular implementation we study, π_o is a softmax over preference weights, $\pi_o(a \mid s) \propto \exp(w_o(s, a)/\tau)$, and receives the update

$$w_o(s, \cdot) \leftarrow w_o(s, \cdot) + \alpha \big(Q_U(s, o, a) - Q_\Omega(s, o)\big) \big(e_a - \pi_o(\cdot \mid s)\big), \tag{1}$$

where $Q_U(s, o, a)$ is the critic's value for taking primitive action a under option o at s and $Q_\Omega(s, o)$ its value for running option o from s, so their difference is the intra-option advantage, and e_a is the one-hot vector on the executed action, making $e_a - \pi_o(\cdot \mid s)$ the softmax score function. The termination function β_o is a sigmoid updated along the option advantage $A_\Omega(s, o) = Q_\Omega(s, o) - \max_{o'} Q_\Omega(s, o')$, and π_Ω is ε_{up}-greedy on the tabular Q_Ω. The original hyperparameters are $\tau = 10^{-3}$, $\alpha = 0.25$ for both the intra-option and termination updates, 0.5 for the critic, $\varepsilon_{\text{up}} = 0.1$, and discount 0.99. Every reported configuration keeps them.

Environments. FourRooms is the benchmark option-critic was introduced on, and it is also an unusually good instrument. Its state space is finite and small enough to enumerate, so for any transition kernel we can solve the MDP exactly, read off the set of actions an optimal policy would accept in every state, and turn that set into a per-state test of whether an option is usable there; nothing in Section 3 is available without it. The grid is 13×13 with 104 free cells and the goal in the east doorway, leaving 103 non-goal states. Under *local slip* the agent moves to a uniformly random neighbouring free cell with probability p; under *global teleport* it jumps to a uniformly random free cell with probability p and slips locally with probability $(1 - p)p$, with $p \in \{0, \frac{1}{3}, \frac{1}{2}\}$. Two rewards are used: the original *sparse* reward of 1 on entering the goal, and a *constant cost* of -1 on every transition, which makes the optimal policy the shortest path and the acceptable action set purely combinatorial. Episodes are capped at 1000 steps and training runs for 1000 episodes.

Two continuous variants test whether the mechanism survives function approximation, where no state can be enumerated and the diagnostics do not apply. In *Direct* the actions are unit displacements with uniform noise and the observation is the position. *Inertia* changes both the state and the action spaces: actions are accelerations, the observation carries a velocity, and a collision zeroes it, so the agent controls a second-order system on the same floor plan. All four option-critic components become two-layer tanh networks trained with Adam, target networks and a replay buffer; Appendix A lists every hyperparameter.

Evaluation. Training curves conflate what a policy has learned with how much it randomises while acting, the exact confound at stake here, so we report performance from frozen checkpoints. Parameters are copied out and never updated; explicit lower-level exploration is set to zero in every condition, so a branch trained with $\varepsilon_{\text{lo}} = 0.1$ is evaluated with $\varepsilon_{\text{lo}} = 0$; $\varepsilon_{\text{up}} = 0.1$ is retained and marginalised exactly; and the expected number of capped transitions to the goal is computed by finite-horizon survival dynamic programming on the true kernel with horizon 1000, starting uniformly over the 103 non-goal states. This evaluation has no sampling error, so all reported uncertainty comes from the 350 matched training seeds, and every contrast is paired within seed. The neural experiments and the tabular termination grid instead use the mean number of steps over the last 100 training episodes, the metric the original scaling result was reported on, and every figure and table that uses it says so. Appendix B gives the resampling scheme behind every interval.

3 Detecting policy necrosis

The measurements below need a notion of what it means for one option to be usable in one state, and a way to attribute a failure either to the options or to the router that picks among them.

Necrotic options. Fix a frozen checkpoint and a state s, and let $A^\star(s)$ be the set of primitive actions an optimal policy for the true kernel would be willing to take at s. Option o is *necrotic* at s when its intra-option policy has committed there to something outside that set. Under the constant cost this needs no threshold: $A^\star(s)$ is exactly the set of actions that reduce the wall-respecting distance to the goal, all ties kept (65 of 103 states have two), and o is viable at s when $\arg\max_a w_o(s, a) \in A^\star(s)$. Under the sparse reward we take $A^\star(s)$ from value iteration on the exact kernel, keeping all numerically tied maximisers, and call o necrotic when $\sum_{a \in A^\star(s)} \pi_o(a \mid s) < 0.01$; the deterministic case agrees state by state with a breadth-first search. Both criteria weight all states equally and are evaluated with exploration switched off.

Where the failure sits. With K options, every (state, seed) pair falls in exactly one of three classes. It is **D** when all K options are necrotic, so no choice the router could make would help; **M** when at least one option is viable but the greedy $\arg\max_o Q_\Omega(s, o)$ selects a necrotic one, so usable

coverage exists and is missed; and **R** when the router selects a viable option. Multiplicity can act on $\Pr(\mathrm{D}) = \Pr(\bigcap_o D_o)$, while the mean single-option necrosis rate $\frac{1}{K}\sum_o \Pr(D_o)$ says whether individual options improved. Separating them is the point: if extra options make each option better, both fall together; if extra options supply redundancy, only the first falls.

4 THE LEARNED TERMINATION FUNCTION IS NOT A SOURCE OF PERFORMANCE

4.1 WHAT THE TERMINATION GRADIENT IS HEADING FOR

Option-critic updates β_o along the option advantage $A_\Omega(s,o) = Q_\Omega(s,o) - \max_{o'} Q_\Omega(s,o')$, which is never positive, so the update only ever raises $\beta_o(s)$, wherever the current option is not the greedy one. The rule it heads for is greedy interruption,

$$\beta(s,o) = \begin{cases} 0, & Q_\Omega(s,o) > V_\Omega(s), \\ 1, & \text{otherwise}, \end{cases} \tag{2}$$

and the collapse of learned β towards 1 that the literature reports (Harb et al., 2018; Harutyunyan et al., 2019) is that drift. We analyse (2) in a two-level setting where an upper level picks an option and a lower level picks an action inside it, both driven by value estimates. Appendix C gives the model, the algorithms and the proofs.

Proposition 1 (Vacuity under value consistency)**.** *If the selector and the termination test read the same estimates Q_t and the state value obeys $V_t(s) = \max_o Q_t(s,o)$, then $Q_t(s,C) \le V_t(s)$ for every current option C, so* (2) *terminates at every step and produces trajectories and regret identical to $\beta \equiv 1$.*

Theorem 2 (Worst-case separation under the standard definitions)**.** *If instead the selector uses optimistic indices while the test reads the critic's raw values, which is option-critic's own definitions, then for two-level UCB there is an instance with two options and two actions each on which, writing* gi *for rule* (2)*, for every $T \ge 4$,*

$$\mathcal{R}_T^{\mathrm{gi}} \;\ge\; \frac{T-2}{64}, \qquad \mathcal{R}_T^{\beta\equiv 1} \;\le\; 64\ln T + \tfrac{1}{2}\big(1 + \tfrac{\pi^2}{3}\big).$$

Over the class the worst case is $\Theta(T)$ for gi *against $O(\sqrt{MT\ln T})$ for $\beta \equiv 1$, with M the number of option–action pairs, and a two-step tabular MDP that reselects the option every episode still forces $\Omega(T)$.*

Theorem 3 (Removing termination restores a guarantee)**.** *Under $\beta \equiv 1$ the two-level agent is isomorphic to a flat agent on the MDP whose actions are the pairs (o,a), so UCBVI-BF gives, with probability at least $1-\delta$ and $L = \ln(5HSDT/\delta)$, $\mathrm{Reg}_N \le 30L\sqrt{HSDT} + 2500H^2S^2DL^2 + 4H\sqrt{TL} = \widetilde{O}(\sqrt{HSDT})$, where S counts states, D option–action pairs per state, H the horizon and $T = NH$.*

A termination function trained to maximise return therefore has no room in which to help. Consistent values make it an identity operation; inconsistent values make it an exploration veto with no uniform sublinear guarantee, and the veto costs most exactly when the option set grows and the selector has more to explore. The separation is worst case and not pointwise — the veto can also suppress exploration of a genuinely worse option, and Appendix C.6 gives an instance where it wins — but a component whose sign depends on the instance is not a source of reliable gains either way.

4.2 WHAT THE EXPERIMENTS MEASURE

Forcing $\beta \equiv 1$ removes temporal extension outright: the agent draws a new option after every primitive step, so an option is nothing more than a latent index refreshed at every state. Temporal extension is the only thing the control takes away, so a scaling result that rested on it should collapse here.

It does not. Table 1 gives the measured effect in the sparse-reward setting option-critic was introduced on, on the metric the original scaling result was reported on. The two conditions do differ, and they

Table 1: The measured effect of learned termination reverses as the option set grows. Sparse-reward FourRooms, the setting option-critic was introduced on. Each entry is $100 \times (\text{Term} - \beta{\equiv}1)/\beta{\equiv}1$ on the mean step count over the last 100 training episodes, so negative favours learned termination, with 95% paired bootstrap intervals. The tabular grid resamples 10 rank blocks of 35 seeds per cell, the neural grid 50 individual seeds. $K{=}1$ is omitted because a single option cannot change identity on termination.

	Relative effect (%) with 95% CI		
Environment	$K = 2$	$K = 4$	$K = 8$
Tabular			
No perturbation	-23.8 [-37.0, -7.6]	$+0.7$ [-12.0, 13.8]	$+11.1$ [7.6, 14.2]
Local slip $p = 1/3$	-11.6 [-21.3, -0.8]	-0.9 [-11.3, 10.1]	$+29.9$ [22.1, 36.3]
Local slip $p = 1/2$	-18.8 [-26.2, -10.3]	$+4.8$ [1.0, 9.0]	$+37.0$ [33.7, 40.7]
Teleport $p = 1/3$	-20.0 [-28.6, -10.6]	$+1.8$ [-4.1, 7.3]	$+29.9$ [24.5, 34.6]
Teleport $p = 1/2$	-14.6 [-25.0, -3.6]	$+5.0$ [0.3, 11.2]	$+20.6$ [17.0, 24.3]
Neural			
Direct	$+5.3$ [0.4, 11.3]	$+0.3$ [-21.4, 28.0]	-29.0 [-59.0, 20.2]
Inertia	-1.0 [-3.4, 1.3]	$+0.2$ [-13.1, 15.9]	$+16.4$ [-18.2, 65.0]

differ in the way Theorem 2 predicts: the sign tracks how much exploration the selector has to do. In the tabular grid learned termination wins all five kernels at $K{=}2$ by 11.6% to 23.8%, with every interval below zero; it is within $\pm 5\%$ in all five at $K{=}4$, with three intervals covering zero; and it loses all five at $K{=}8$ by 11.1% to 37.0%, with every interval above zero. With two options there is little for the veto to block; with eight, the router has a large candidate set and blocking it costs. The neural grid gives no consistent direction, with a 29.0% advantage in Direct at $K{=}8$ against a 16.4% disadvantage in Inertia, and intervals wide enough at $K{=}4$ and $K{=}8$ to admit either sign. A component whose measured contribution reverses with a hyperparameter it is meant to be orthogonal to is not a stable source of gains, and the option-count scaling it is supposed to explain survives its removal.

The shortest-path variant lets us make the comparison without relying on training curves at all, because both arms can be evaluated by exact frozen dynamic programming. There the paired difference between learned option-critic and its $\beta \equiv 1$ match contains zero at every option count, and the two agents trace the same eightfold improvement as K grows, $962.5 \to 821.4 \to 377.2 \to 117.4$ expected steps against $963.2 \to 822.3 \to 384.4 \to 122.5$. Whatever produces that improvement is present when options have no temporal extent at all. Table 4 in Appendix D gives those numbers, and Figure 5 there redraws Table 1 as a heat map.

5 Policy necrosis, and what repairs it

Why states stop learning. Look again at the intra-option update in Eq. 1. The quantity that enters the policy is $w_o(s,a)/\tau$, so with $\tau = 10^{-3}$ and $\alpha = 0.25$ an advantage is multiplied by $\alpha/\tau = 250$ before it reaches the logits. From a uniform policy over four actions, an advantage of 0.008 already leaves the sampled action with probability above 0.7, and 0.032 pushes it past 0.999. The commitment is self-sealing: the score function $e_a - \pi_o(\cdot \mid s)$ that multiplies the advantage goes to zero for the committed action, and the alternatives that carry a non-vanishing score are sampled with probability close to zero. Under a sparse reward the first advantage a state sees comes from an essentially uninformed critic, so what it commits to is close to arbitrary, and further training on the same option does not repair it. This is the softmax gravity well described by Mei et al. (2020a), with the temperature turning a slow drift into a one-update event.

A matched fork isolates the repair. To test the account we train a common 500-episode prefix with $\varepsilon_{\text{lo}} = 0$, snapshot the entire learner including the random number generator, and continue along two branches: a control that keeps $\varepsilon_{\text{lo}} = 0$ and a rescue branch that mixes in uniform random actions with probability $\varepsilon_{\text{lo}} = 0.1$. The rescue branch changes only which actions are executed, so nothing about the learning rule changes. Both branches are evaluated with $\varepsilon_{\text{lo}} = 0$, so any difference comes from parameters written during training and not from randomness at test time; the episode-500 checkpoints of the two branches are bit-for-bit identical.

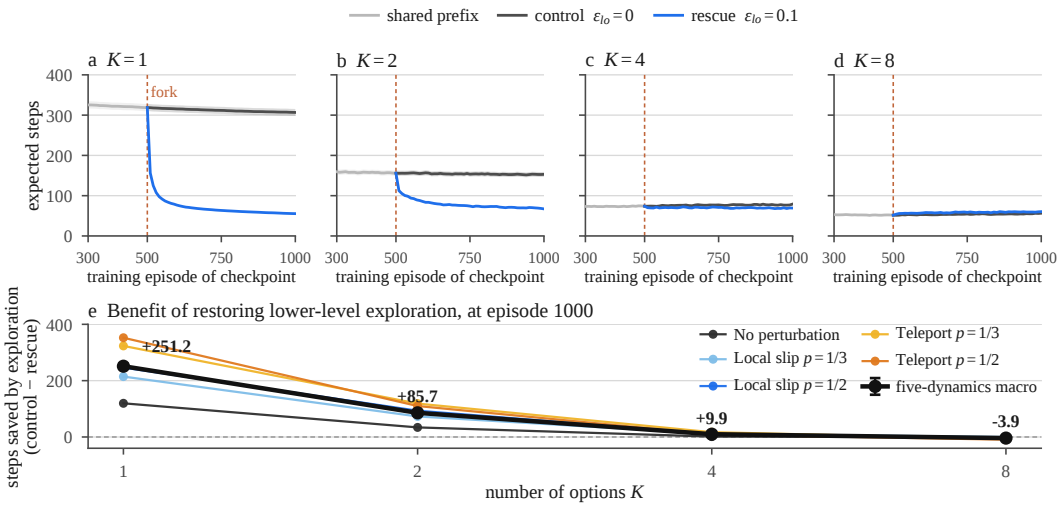

Figure 2: Restoring lower-level exploration replaces the entire benefit of extra options. (a–d) Frozen exact evaluation of every checkpoint, five-kernel average, per option count. Training shares a $\varepsilon_{\text{lo}} = 0$ prefix through episode 500 and then forks; both branches are evaluated with $\varepsilon_{\text{lo}} = 0$. Bands are 95% bootstrap intervals over 350 matched seeds. (e) Paired step reduction at episode 1000, per kernel and averaged.

Table 2: Repairing the lower level removes the work that extra options were doing. Left: sparse-reward tabular FourRooms, five-kernel average of expected capped transitions at the episode-1000 checkpoint, paired over 350 seeds. Right: constant-cost FourRooms, $\beta \equiv 1$ agents trained with and without lower-level exploration, evaluated identically. All evaluations use $\varepsilon_{\text{lo}} = 0$.

	Sparse reward: matched fork at episode 500					Constant cost	
K	control	rescue	reduction	95% CI	seeds helped	train $\varepsilon_{\text{lo}} = 0$	train $\varepsilon_{\text{lo}} = 0.1$
1	306.66	55.42	251.23	[242.59, 259.99]	350/350	963.21	**36.50**
2	152.93	67.18	85.75	[79.68, 91.98]	344/350	822.31	45.21
4	79.14	69.28	9.86	[6.74, 13.03]	211/350	384.37	55.09
8	56.42	60.35	−3.94	[−5.73, −2.14]	138/350	122.46	85.03

One option is enough. Figure 2a shows what happens with a single option. The control drifts from 318.7 at the fork to 306.7 expected steps by the end of training; the rescue branch is at 156.1 ten episodes later, 76.6 by episode 600, and 55.4 at the end. All 350 seeds improve, and the paired reduction is 251.2 steps [242.59, 259.99], running from 119.6 under no perturbation to 352.3 under teleport with $p = \frac{1}{2}$ (Figure 2e). Under the constant-cost reward the same intervention takes a single $\beta \equiv 1$ option from 963.2 to 36.5 expected steps and its frozen success probability from 0.037 to 0.999. One option, evaluated with no exploration at all, solves the task.

Why the benefit shrinks as K grows. Panels b–d of Figure 2 repeat the fork at $K = 2, 4, 8$, and the gap the rescue branch opens shrinks each time. The paired reduction falls from 251.2 steps at K=1 to 85.7, then 9.9, and at K=8 it turns negative at −3.9 [−5.73, −2.14], with only 138 of 350 seeds improving (Table 2). Under the constant-cost reward the exploring agent gets steadily worse as options are added, from 36.5 to 85.0 expected steps. The two interventions are not interchangeable and do not act on the same object. Exploration acts on the cause, restoring the sampling that lets a committed state recover, and Section 6 shows it lowers the single-option necrosis rate directly; extra options leave every individual policy where it was and act on the aggregate, by making it unlikely that all of them fail in the same place. The shrinking gap records the interaction: multiplicity hides most of the cost of necrosis, leaving a repair little headroom, and once the repair has removed the necrosis multiplicity has nothing to compensate for.

The neural implementation agrees, through two different mechanisms. Figure 3 repeats the comparison in continuous FourRooms with $\beta \equiv 1$ throughout, contrasting the base softmax against explicit ε-greedy behaviour and against entropy-regularised sampling. With one option the base softmax sits at the 1000-step cap in both environments (977 and 970), ε-greedy brings it to 299 and 250, and entropy regularisation to 207 and 156. Adding options helps the base softmax a

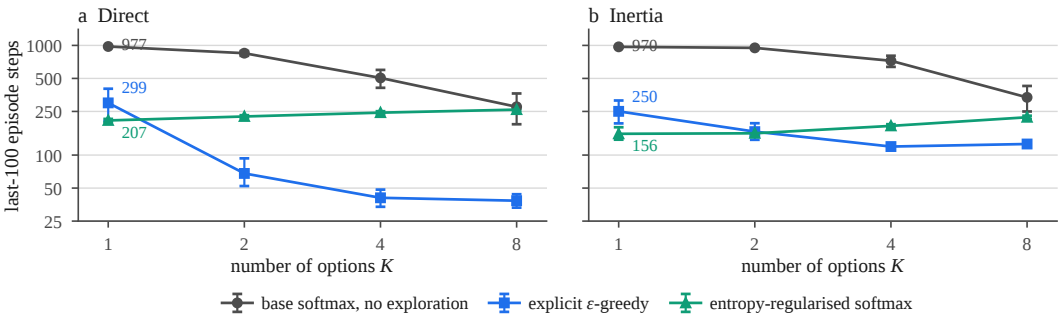

Figure 3: Two exploration mechanisms make one option sufficient in the neural implementation. Mean steps over the last 100 training episodes with $\beta \equiv 1$ throughout, 50 seeds per point, 95% bootstrap intervals, log scale. Numbers mark the one-option values.

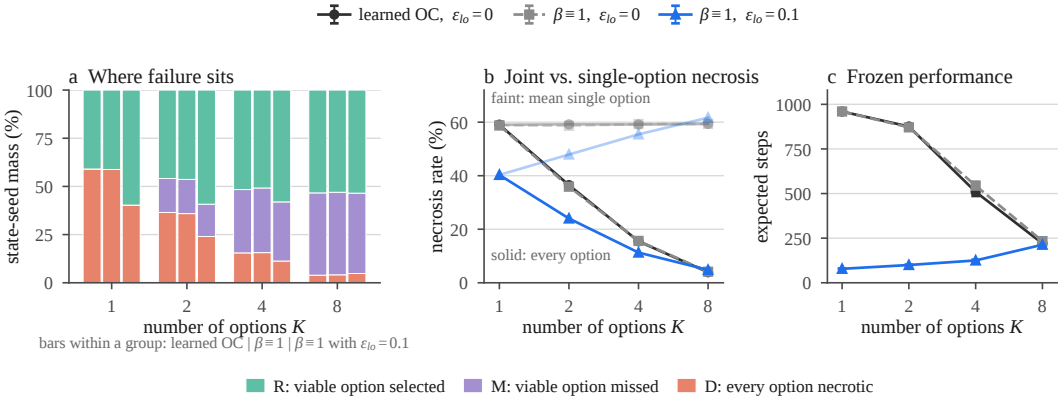

Figure 4: More options make simultaneous failure rare without making any option better. Constant-cost FourRooms under teleport $p = \frac{1}{2}$, episode-1000 checkpoints, 350 matched seeds, threshold-free necrosis test. (a) Every (state, seed) pair in exactly one class. (b) The rate at which every option is necrotic (solid) against the mean single-option rate (faint). (c) Exact frozen performance of the same checkpoints. Learned option-critic and its $\beta \equiv 1$ match coincide throughout.

great deal, helps ε-greedy much less, and makes the entropy configuration monotonically worse: $207 \to 225 \to 244 \to 260$ in Direct and $156 \to 158 \to 185 \to 221$ in Inertia. Inertia runs at $\tau = 0.5$, five hundred times the tabular temperature, and its one-option base softmax still fails, so the failure is no artefact of the extreme tabular setting.

6 What extra options provide is redundant coverage

If the lower level is necrotic and extra options carry the load instead, what exactly is the load they carry? Measuring that calls for the constant-cost reward, where the optimal policy is the shortest path, so the acceptable action set at a state is exactly the set of distance-reducing actions and the necrosis test is combinatorial, with no threshold and no dependence on a value scale. Under the sparse reward the value function encodes reachability and the numerical spread between distinct reachable routes is small, which makes a shortest-path yardstick a coarser instrument there. We therefore read the coverage story off the constant-cost grid, and Figure 6 in Appendix D reports the sparse-reward version, which agrees.

For an agent that never explored, the mean single-option necrosis rate is 59.04% at $K{=}1$ and 59.35% at $K{=}8$, and it never leaves that band. Close to three fifths of the state space is locked onto a wrong action in a typical option, and that is as true of the eighth option as of the first. The joint rate behaves completely differently: the probability that a state is necrotic for every option falls $59.0 \to 36.5 \to 15.5 \to 3.9$ percent, and the expected steps of those same frozen policies fall $959.5 \to 875.8 \to 509.4 \to 220.5$. Extra options do not produce better options; they produce a set whose members fail in different places. Table 3 says the same as a within-seed contrast, and the learned-termination agent and its $\beta \equiv 1$ match agree to within their intervals, which is Section 4 in coverage terms.

Table 3: Going from one option to eight, within seed. Constant-cost FourRooms, teleport $p = \frac{1}{2}$, episode-1000 checkpoints, 350 matched seeds. Necrosis rates are in percentage points and steps are expected capped transitions; negative means the eight-option agent is better. Joint necrosis collapses in every row while single-option necrosis does not improve in any of them.

Agent	Δ all necrotic	Δ single-option necrosis	Δ expected steps
learned OC, train $\varepsilon_{\text{lo}} = 0$	-55.11 [-55.65, -54.57]	$+0.31$ [-0.21, 0.83]	-739.0 [-749.6, -728.1]
$\beta \equiv 1$, train $\varepsilon_{\text{lo}} = 0$	-54.66 [-55.22, -54.10]	$+0.57$ [0.02, 1.12]	-726.6 [-739.7, -713.2]
$\beta \equiv 1$, train $\varepsilon_{\text{lo}} = 0.1$	-35.49 [-35.98, -34.98]	$+21.39$ [20.90, 21.88]	$+134.9$ [125.4, 144.8]

How independent are the failures? If the K necrotic sets were independent, the joint rate at $K{=}8$ would be $0.594^8 = 1.5\%$ against the measured 3.9%; if they were identical, it would be 59%. Solving $\Pr(\mathrm{D}) = \Pr(D_o)^{\kappa}$ for κ gives an effective number of independent options: $\kappa = 1.92$, 3.55 and 6.20 at $K = 2, 4, 8$. Options trained side by side, on shared trajectories with a shared critic, fail together somewhat more than chance predicts and far less than a common cause would, so a pool of K options is worth roughly $K/2$ independent attempts at covering the state space. That also says what limits the account: the return on adding options is set by how decorrelated their failures are, a property of the training procedure and not of the option count.

The bottleneck moves upstairs. Coverage that exists is not coverage that gets used. As D shrinks, the M class grows in step: the router picks a necrotic option while a viable one is available in 0%, 17.6%, 32.9% and 42.7% of states at $K = 1, 2, 4, 8$ (Figure 4a). By $K{=}8$ the dominant residual error is a routing error and not a coverage error, which is the failure mode that interest functions and attention over options were introduced to attack (Khetarpal et al., 2020; Chunduru & Precup, 2022).

Redundancy has a price once the lower level works. The third row of Table 3 is the informative boundary case. An agent trained with $\varepsilon_{\text{lo}} = 0.1$ still gets a coverage benefit from more options, with joint necrosis falling by 35.5 points, but its single-option rate rises by 21.4 points and its performance gets *worse* by 134.9 steps: a fixed budget spread over eight options degrades each one, and the router must distinguish eight policies instead of one. Redundancy earns its keep when there is a failure to be redundant against, and charges rent when there is not.

Figure 7 in Appendix D shows the transition on the map itself. For the agent that never explored, joint necrosis occupies 62 of 103 cells at $K{=}1$ and none at $K{=}8$, and four of every five cells it vacates turn into routing misses instead of correct routes. Its $\beta \equiv 1$ match is indistinguishable from it, cell by cell. The agent that explored starts at 37 necrotic cells with one option and ends with 4 at $K{=}8$, while its correctly routed cells fall from 66 to 50 along the way.

7 Related work

Options and option-critic. The options framework formalises temporally extended action (Sutton et al., 1999; Precup, 2000), alongside earlier hierarchical formulations (Parr & Russell, 1998; Dietterich, 2000). Option-critic learns intra-option policies and terminations from the task return (Bacon et al., 2017); extensions cover continuous control (Klissarov et al., 2017), deeper hierarchies (Riemer et al., 2018), off-the-shelf policy optimisation (Zhang & Whiteson, 2019) and learned initiation (Khetarpal et al., 2020; Chunduru & Precup, 2022), while other lines build options from environment structure or intrinsic objectives (Şimşek & Barto, 2009; Konidaris & Barto, 2009; Machado et al., 2017; 2018; Eysenbach et al., 2019; Jinnai et al., 2019; Bagaria & Konidaris, 2020) or hierarchy over goals (Kulkarni et al., 2016; Vezhnevets et al., 2017; Nachum et al., 2018). Regret guarantees for option-based learning exist under structural assumptions (Drappo et al., 2024); Section 4.1 asks instead what a return-trained termination rule can guarantee.

Termination collapse and its remedies. That option-critic terminations drift towards 1 is documented and has motivated direct remedies: a deliberation cost that charges for switching (Harb et al., 2018), and a termination objective built on option models instead of the control objective (Harutyunyan et al., 2019). Those works treat collapse as an empirical nuisance to engineer around. Section 4.1 shows the drift is a property of the objective itself, so those remedies act as stopgaps: they hold β away from the boundary without removing the reason it moves there.

Where hierarchy's benefits come from. Nachum et al. (2019) isolate the claimed benefits of goal-conditioned hierarchies on continuous control and attribute most of the gain to improved exploration in the high-level action space. Our result points the same way and localises the mechanism one level lower, inside the option's own policy parameterisation, where it is visible state by state.

Softmax policy gradient. Mei et al. (2020b) and Mei et al. (2020a) show that softmax-parameterised policy gradient is initialisation-sensitive and slows near the corners of the simplex, and Agarwal et al. (2021a) show that exploration or regularisation is needed for global convergence; entropy regularisation is the standard practical response (Williams, 1992; Mnih et al., 2016; Haarnoja et al., 2018). Policy necrosis is that phenomenon at $\tau = 10^{-3}$, where the committal step takes one update, and what we add is that option-critic covers for it with options.

Empirical practice. Reported gains often survive poorly once controls and seed counts are taken seriously (Henderson et al., 2018; Engstrom et al., 2020; Andrychowicz et al., 2021; Agarwal et al., 2021b; Patterson et al., 2024). This paper is an instance in the hierarchical setting.

8 Discussion

The three results compose into one account. A near-deterministic softmax with a large effective step size commits states to arbitrary actions and stops updating them, leaving most of the state space necrotic for any given option; the diagnostic of Section 3 locates that failure state by state, and restoring lower-level exploration repairs it, at which point one option solves the task. Extra options repair nothing individually. They supply policies whose necrotic sets only partly overlap, so joint failure falls while individual failure does not, performance follows the joint quantity, and the benefit goes with the failure it was covering. Learned termination plays no part in any of this, and Section 4.1 explains why it never could: the rule the termination gradient converges to does nothing at all when its values agree with the selector's, and blocks the selector's exploration when they do not. The option count measures redundancy against a repairable defect, and the case for learned temporal abstraction has to be made somewhere else.

A structural prior can be credited for a defect elsewhere. The broader reading is not about options. A structural component sits on a base learner, and if the base learner has a repairable failure, the component is credited with whatever that failure was costing. The option count is an unusually legible example: individual policies stay broken, their failures decorrelate, and the aggregate improves. The same accounting applies to any architectural prior evaluated by a curve over its own capacity, and it is settled the same way, by repairing the base learner first and asking whether the prior still earns its keep. Redundancy also deserves naming as a mechanism: ensembles obtain it deliberately (Osband et al., 2016) and option-critic stumbles into it, so the lever is not the number of options but how decorrelated their failures are.

Limitations. Two boundaries are worth stating. The diagnostic needs a notion of which actions are acceptable in a state, which an exactly solvable environment supplies and a large one does not, so elsewhere it requires a surrogate whose quality any conclusion inherits. And the exchange rate between options and exploration is set by κ, a property of how the options were trained together, so the point at which multiplicity stops helping moves with the training procedure even though the mechanism does not.

Future work. Make redundancy deliberate: diversity objectives should be judged by how much they raise κ at fixed K, a cheaper and sharper target than end-to-end return. Carry the necrosis test past exactly solvable environments, either against a reference policy or learned critic, or by dropping the oracle and tracking the two quantities that make necrosis self-sealing, policy saturation and the vanishing score-function norm. And ask which termination objectives escape Section 4.1: the obstruction applies to rules trained on return, while deliberation costs and predictability-based termination (Harb et al., 2018; Harutyunyan et al., 2019) change the objective, so the same analysis should say whether either can be shown to help.

Reproducibility statement

Section 2 states the evaluation contract and the hyperparameters, Appendix A lists the remaining settings for the tabular and neural experiments, and Appendix B gives the resampling scheme behind every interval. Section 3 gives both necrosis criteria in full, including the tie-handling rule and the audit of the value-iteration action sets against breadth-first search. Appendix C contains the model, the algorithms and complete proofs for Proposition 1, Theorem 2 and Theorem 3. The supplementary material contains the training, checkpoint-evaluation, and analysis code for all three experiment families, the derived per-seed and summary tables behind every figure and table, a single entry point that rebuilds every figure in this paper from those tables without touching a raw checkpoint, and a validator that checks file hashes and the figure contract.

Ethics statement

This work is a methodological study of a reinforcement learning algorithm. It involves no human subjects, no personal data, and no deployment. We see no ethical concerns beyond the general question of how empirical claims in the field are supported, which the paper addresses directly.

AI use statement

We used generative AI tools for readability editing of the manuscript. We have not used generative AI tools for proposing or refining the hypotheses, designing or critiquing the methodology or experiments, implementing the methods, interpreting the results, formulating the mathematical claims, or writing the proofs; generating synthetic datasets and qualitative data analysis are not applicable to this work. We have reviewed all AI-assisted work: every number in the paper was read back from the released derived data files by an author, the figure-generation code was checked against those files, and every theorem statement and proof was written and verified by the authors. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

References

Alekh Agarwal, Sham M. Kakade, Jason D. Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *Journal of Machine Learning Research*, 22(98):1–76, 2021a.

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, pp. 29304–29320, 2021b.

Marcin Andrychowicz, Anton Raichuk, Piotr Stańczyk, Manu Orsini, Sertan Girgin, Raphaël Marinier, Léonard Hussenot, Matthieu Geist, Olivier Pietquin, Marcin Michalski, Sylvain Gelly, and Olivier Bachem. What matters for on-policy deep actor-critic methods? a large-scale study. In *International Conference on Learning Representations (ICLR)*, 2021.

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2–3):235–256, 2002.

Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pp. 263–272, 2017.

Pierre-Luc Bacon, Jean Harb, and Doina Precup. The option-critic architecture. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 1726–1734, 2017.

Akhil Bagaria and George Konidaris. Option discovery using deep skill chaining. In *International Conference on Learning Representations (ICLR)*, 2020.

Raviteja Chunduru and Doina Precup. Attention option-critic. *arXiv preprint arXiv:2201.02628*, 2022.

Cédric Colas, Olivier Sigaud, and Pierre-Yves Oudeyer. How many random seeds? statistical power analysis in deep reinforcement learning experiments. *arXiv preprint arXiv:1806.08295*, 2018.

Thomas G. Dietterich. Hierarchical reinforcement learning with the MAXQ value function decomposition. *Journal of Artificial Intelligence Research*, 13:227–303, 2000.

Gianluca Drappo, Alberto Maria Metelli, and Marcello Restelli. A provably efficient option-based algorithm for both high-level and low-level learning. *Reinforcement Learning Journal*, 2:819–839, 2024.

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, and Aleksander Madry. Implementation matters in deep RL: A case study on PPO and TRPO. In *International Conference on Learning Representations (ICLR)*, 2020.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *International Conference on Learning Representations (ICLR)*, 2019.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1861–1870, 2018.

Jean Harb, Pierre-Luc Bacon, Martin Klissarov, and Doina Precup. When waiting is not an option: Learning options with a deliberation cost. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pp. 3165–3172, 2018.

Anna Harutyunyan, Will Dabney, Diana Borsa, Nicolas Heess, Rémi Munos, and Doina Precup. The termination critic. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 89 of *Proceedings of Machine Learning Research*, pp. 2231–2240, 2019.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pp. 3207–3214, 2018.

Yuu Jinnai, Jee Won Park, David Abel, and George Konidaris. Discovering options for exploration by minimizing cover time. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3130–3139, 2019.

Khimya Khetarpal, Martin Klissarov, Maxime Chevalier-Boisvert, Pierre-Luc Bacon, and Doina Precup. Options of interest: Temporal abstraction with interest functions. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, volume 34, pp. 4444–4451, 2020.

Martin Klissarov, Pierre-Luc Bacon, Jean Harb, and Doina Precup. Learnings options end-to-end for continuous action tasks. *arXiv preprint arXiv:1712.00004*, 2017. Hierarchical Reinforcement Learning Workshop, NIPS 2017.

George Konidaris and Andrew G. Barto. Skill discovery in continuous reinforcement learning domains using skill chaining. In *Advances in Neural Information Processing Systems 22 (NeurIPS)*, pp. 1015–1023, 2009.

Tejas D. Kulkarni, Karthik R. Narasimhan, Ardavan Saeedi, and Joshua B. Tenenbaum. Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. In *Advances in Neural Information Processing Systems 29 (NeurIPS)*, pp. 3675–3683, 2016.

Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985.

Marlos C. Machado, Marc G. Bellemare, and Michael Bowling. A Laplacian framework for option discovery in reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2295–2304, 2017.

Marlos C. Machado, Clemens Rosenbaum, Xiaoxiao Guo, Miao Liu, Gerald Tesauro, and Murray Campbell. Eigenoption discovery through the deep successor representation. In *International Conference on Learning Representations (ICLR)*, 2018.

Jincheng Mei, Chenjun Xiao, Bo Dai, Lihong Li, Csaba Szepesvári, and Dale Schuurmans. Escaping the gravitational pull of softmax. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, pp. 21130–21140, 2020a.

Jincheng Mei, Chenjun Xiao, Csaba Szepesvári, and Dale Schuurmans. On the global convergence rates of softmax policy gradient methods. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6820–6829, 2020b.

Volodymyr Mnih, Adrià Puigdomènech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1928–1937, 2016.

Ofir Nachum, Shixiang Gu, Honglak Lee, and Sergey Levine. Data-efficient hierarchical reinforcement learning. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, pp. 3303–3313, 2018.

Ofir Nachum, Haoran Tang, Xingyu Lu, Shixiang Gu, Honglak Lee, and Sergey Levine. Why does hierarchy (sometimes) work so well in reinforcement learning? *arXiv preprint arXiv:1909.10618*, 2019. NeurIPS 2019 Deep Reinforcement Learning Workshop.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped DQN. In *Advances in Neural Information Processing Systems 29 (NeurIPS)*, pp. 4026–4034, 2016.

Ronald Parr and Stuart Russell. Reinforcement learning with hierarchies of machines. In *Advances in Neural Information Processing Systems 10 (NeurIPS)*, pp. 1043–1049, 1998.

Andrew Patterson, Samuel Neumann, Martha White, and Adam White. Empirical design in reinforcement learning. *Journal of Machine Learning Research*, 25(318):1–63, 2024.

Doina Precup. *Temporal Abstraction in Reinforcement Learning*. PhD thesis, University of Massachusetts Amherst, 2000.

Matthew Riemer, Miao Liu, and Gerald Tesauro. Learning abstract options. In *Advances in Neural Information Processing Systems 31 (NeurIPS)*, pp. 10445–10455, 2018.

Özgür Şimşek and Andrew G. Barto. Skill characterization based on betweenness. In *Advances in Neural Information Processing Systems 21 (NeurIPS)*, pp. 1497–1504, 2009.

Richard S. Sutton, Doina Precup, and Satinder Singh. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1–2):181–211, 1999.

Alexander Sasha Vezhnevets, Simon Osindero, Tom Schaul, Nicolas Heess, Max Jaderberg, David Silver, and Koray Kavukcuoglu. FeUdal networks for hierarchical reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3540–3549, 2017.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3–4):229–256, 1992.

Shangtong Zhang and Shimon Whiteson. DAC: The double actor-critic architecture for learning options. In *Advances in Neural Information Processing Systems 32 (NeurIPS)*, pp. 2010–2020, 2019.

A EXPERIMENTAL SETTINGS

Tabular experiments. FourRooms is a 13×13 grid with 104 free cells; the goal is the east doorway, and the 103 remaining cells form both the initial-state distribution and the domain of the state-uniform diagnostics. Actions are the four cardinal moves; a move into a wall leaves the agent in place. The intra-option policy is a softmax over a 104×4 preference table with temperature $\tau = 10^{-3}$ and learning rate 0.25; the termination function is a sigmoid over a 104-vector with learning rate 0.25; the critic maintains tabular Q_Ω and Q_U over the same 104 states with learning rate 0.5; the policy over options is ε_{up}-greedy on Q_Ω with $\varepsilon_{\text{up}} = 0.1$. The discount is 0.99. Each cell of every grid runs seeds 0 through 349 for 1000 episodes with a 1000-step cap. Five distinct kernels are obtained from local slip and global teleport at $p \in \{0, \frac{1}{3}, \frac{1}{2}\}$, with $p = 0$ shared by the two families. The matched-fork experiment saves a checkpoint every 10 episodes, giving 51 prefix checkpoints and 51 per branch, and snapshots the Mersenne Twister state so that the fork is exact.

Neural experiments. The continuous environments place the agent at cell centres and use the same 13×13 layout. In Direct, an action displaces the agent by one unit in a cardinal direction with uniform noise in $[-0.1, 0.1]$ per coordinate, and the observation is (x, y). In Inertia, an action changes the corresponding velocity component by 0.1 ± 0.01, speed is clipped to 0.99 per axis, position integrates velocity, a collision zeroes it, and the observation is (x, y, v_x, v_y). The goal tolerance is half a cell. Q_Ω, Q_U, the policy, and the termination function are two-hidden-layer tanh networks of width 64, trained with Adam at learning rate 10^{-3}, Polyak-averaged target networks with coefficient 0.005, a replay buffer of $10{,}000$ transitions, batches of 64, and one update every 4 environment steps after 1000 steps of warm-up. Output biases are initialised so that the mean output over a grid of states is zero. Each configuration runs 50 seeds for 1000 episodes with a 1000-step cap. Direct uses $\tau = 10^{-3}$ for the base and ε-greedy configurations with $\varepsilon_{\text{lo}} = 0.1$, and $\tau = 1$ with entropy coefficient 0.5 for the entropy configuration; Inertia uses $\tau = 0.5$ throughout, with $\varepsilon_{\text{lo}} = 0.5$ and entropy coefficient 0.1.

B STATISTICS

The tabular grids use 350 training seeds per cell, matched across every condition being contrasted, so all contrasts are paired within seed. Confidence intervals are 95% percentile intervals from $100{,}000$ paired bootstrap resamples over seeds (Colas et al., 2018; Patterson et al., 2024), with fixed bootstrap seeds recorded alongside the derived tables. The termination grid retains block-level aggregates, so its resampling unit is a 35-seed rank block, ten per cell; the neural experiments resample 50 individual seeds. Where the five kernels are pooled we average within a seed first and bootstrap over seeds, counting the duplicated $p = 0$ cell once. The frozen evaluation is an exact dynamic program and contributes no sampling error, so every reported interval reflects training-seed variability alone.

C TERMINATION RULES AND REGRET

This appendix supplies the model, the algorithms and the proofs behind Section 4.1. Two remarks frame what is and is not claimed. First, "both levels use UCB" does not by itself pin down an algorithm, so the analysis fixes a minimal but non-trivial two-level model in which the interaction between termination and exploration is exposed without any other moving part. Second, the negative results are constructions inside that model, which is a special case of the option setting, so they suffice to rule out a general guarantee.

C.1 A TWO-LEVEL STOCHASTIC BANDIT

Let T be the number of atomic steps. The environment has a single recurring state s. Let $\mathcal{O} = \{1, \dots, K\}$ be the option set and $\mathcal{A}_o$ the actions available inside option o, and write $\mathcal{L} = \{(o, a) : o \in \mathcal{O},\ a \in \mathcal{A}_o\}$ for the set of option–action leaves, with $M = |\mathcal{L}|$. Pulling leaf $i \in \mathcal{L}$ returns an independent reward in $[0, 1]$ with mean μ_i. Put $\mu^\star_o = \max_a \mu_{o,a}$, $\mu^\star = \max_o \mu^\star_o$, and define the upper, lower and total gaps

$$\Delta^{\text{up}}_o = \mu^\star - \mu^\star_o, \qquad \Delta^{\text{lo}}_{o,a} = \mu^\star_o - \mu_{o,a}, \qquad \Delta_{o,a} = \mu^\star - \mu_{o,a} = \Delta^{\text{up}}_o + \Delta^{\text{lo}}_{o,a}.$$

With $I_t = (O_t, A_t)$ the leaf executed at step t and $N_i(T)$ its visit count, the pseudo-regret is $\mathcal{R}_T = T\mu^\star - \mathbb{E}[\sum_{t\le T} \mu_{I_t}] = \sum_i \Delta_i \, \mathbb{E}[N_i(T)]$. Splitting by level, with $N_o(T) = \sum_a N_{o,a}(T)$,

$$\mathcal{R}_T^{\mathrm{up}} = \sum_o \Delta_o^{\mathrm{up}} \mathbb{E}[N_o(T)], \qquad \mathcal{R}_T^{\mathrm{lo}} = \sum_{o,a} \Delta_{o,a}^{\mathrm{lo}} \mathbb{E}[N_{o,a}(T)].$$

Lemma 4 (Exact decomposition)**.** *For any selection algorithm and any T, $\mathcal{R}_T = \mathcal{R}_T^{\mathrm{up}} + \mathcal{R}_T^{\mathrm{lo}}$.*

Proof. Substitute $\Delta_{o,a} = \Delta_o^{\mathrm{up}} + \Delta_{o,a}^{\mathrm{lo}}$ into $\mathcal{R}_T = \sum_{o,a} \Delta_{o,a} \mathbb{E}[N_{o,a}(T)]$ and use $N_o(T) = \sum_a N_{o,a}(T)$ on the first summand. □

C.2 Compatible two-level UCB, and the two termination rules

Each leaf is pulled once during initialisation. Afterwards, write $b_i(t) = \widehat{\mu}_i(t-1) + \sqrt{2 \ln t / N_i(t-1)}$ for the UCB1 index of leaf i. The lower level proposes $a_o^+(t) \in \arg\max_{a \in \mathcal{A}_o} b_{o,a}(t)$ inside each option, the upper level scores options by $B_o(t) = b_{o,a_o^+(t)}(t) = \max_a b_{o,a}(t)$ and proposes $J_t \in \arg\max_o B_o(t)$. Ties are broken by a fixed deterministic rule. The two levels are compatible in the sense that the option score is the maximum of its own leaf indices, which is what makes UCB1's assumptions applicable.

Two algorithms are compared. $\mathsf{A}_{\mathrm{fix}}$ fixes $\beta \equiv 1$: at every step it sets $O_t = J_t$ and executes $A_t = a_{O_t}^+(t)$. A_{gi} carries a current option C_t into the step and applies the greedy-interruption rule (2): if $\beta_t = 1$ it sets $O_t = J_t$, and if $\beta_t = 0$ it sets $O_t = C_t$; either way it then executes $A_t = a_{O_t}^+(t)$ and passes $C_{t+1} = O_t$ to the next step. Write $\mathcal{R}_T^{\mathrm{fix}}$ and $\mathcal{R}_T^{\mathrm{gi}}$ for their pseudo-regrets; $\mathcal{R}_T^{\mathrm{fix}}$ is the $\mathcal{R}_T^{\beta \equiv 1}$ of the main text.

C.3 Fixed $\beta \equiv 1$: finite-time bounds

Theorem 5 (Stepwise equivalence to leaf-level UCB1)**.** *Under $\mathsf{A}_{\mathrm{fix}}$, $I_t \in \arg\max_{i \in \mathcal{L}} b_i(t)$ at every step, so $\mathsf{A}_{\mathrm{fix}}$ is UCB1 run directly on the M leaves.*

Proof. $b_{I_t}(t) = B_{J_t}(t) = \max_o B_o(t) = \max_o \max_a b_{o,a}(t) = \max_i b_i(t)$, and the fixed tie-breaking rule selects the same leaf in both formulations. □

Theorem 6 (Instance-dependent bound)**.** *Let $C_0 = 1 + \pi^2/3$. For rewards in $[0,1]$ and any $T \ge M$,*

$$\mathcal{R}_T^{\mathrm{fix}} \;\le\; \sum_{(o,a):\Delta_{o,a}>0} \Big(\frac{8 \ln T}{\Delta_{o,a}} + C_0 \Delta_{o,a} \Big),$$

and, level by level, $\mathcal{R}_T^{\mathrm{up,fix}} \le \sum_{(o,a):\Delta_{o,a}>0} \Delta_o^{\mathrm{up}} (8 \ln T / \Delta_{o,a}^2 + C_0)$ and $\mathcal{R}_T^{\mathrm{lo,fix}} \le \sum_{(o,a):\Delta_{o,a}>0} \Delta_{o,a}^{\mathrm{lo}} (8 \ln T / \Delta_{o,a}^2 + C_0)$.

Proof. UCB1 guarantees $\mathbb{E}[N_i(T)] \le 8 \ln T / \Delta_i^2 + C_0$ for every suboptimal arm (Auer et al., 2002). By Theorem 5 the arms are the leaves, so multiplying by Δ_i and summing gives the first display. Multiplying the same visit-count bound by Δ_o^{up} and by $\Delta_{o,a}^{\mathrm{lo}}$ and applying Lemma 4 gives the other two. □

Corollary 7 (Distribution-free bound)**.** *For $T \ge \max\{M, 2\}$, $\mathcal{R}_T^{\mathrm{fix}} \le \min\{T,\ 4\sqrt{2MT \ln T} + C_0 M\}$.*

Proof. Split the leaves at a threshold $\varepsilon \in (0, 1]$. Leaves with $\Delta_i \le \varepsilon$ contribute at most $T\varepsilon$; the rest contribute at most $8M \ln T / \varepsilon + C_0 M$ by Theorem 6. Choosing $\varepsilon = \sqrt{8M \ln T / T}$ makes the first two terms $4\sqrt{2MT \ln T}$. If that ε exceeds 1 the bound already exceeds the trivial bound T, which always holds because single-step gaps are at most one. □

A matching lower bound is classical: for Bernoulli leaves with a unique optimum and any consistent algorithm, $\liminf_T \mathcal{R}_T / \ln T \ge \sum_{i:\Delta_i>0} \Delta_i / \mathrm{kl}(\mu_i, \mu^\star)$ (Lai & Robbins, 1985), so the $O(\ln T)$ rate of Theorem 6 cannot be improved to $o(\ln T)$ on a fixed non-degenerate instance.

C.4 Two value settings for the return-maximising rule

Rule (2) compares the value of continuing with the value of handing control back, and everything turns on which estimates those two are.

Optimistic indices. Set $Q_t^+(s,o) = B_o(t)$ and let the upper policy be the point mass on J_t, so $V_t^+(s) = B_{J_t}(t) = \max_o B_o(t)$. Proposition 1 of the main text is then immediate.

Proof of Proposition 1. For any current option C_t we have $Q_t^+(s,C_t) = B_{C_t}(t) \le \max_o B_o(t) = V_t^+(s)$, so (2) selects $\beta_t = 1$ and the step executes J_t. Induct on t: the two algorithms start identically; if the histories before step t agree then all counts, sample means and indices agree, hence J_t agrees, and both algorithms execute J_t with the same lower-level action, so the histories after step t agree. Action sequences, and therefore regret, coincide on every sample path. □

Raw-value indices. Option-critic's state value is the upper policy's expectation of raw critic values, while a confidence bonus is a device the exploration rule adds on top. In the present model that setting is

$$\widehat{q}_o(t) = \widehat{\mu}_{o,a_o^+(t)}(t-1), \qquad Q_t(s,o) = \widehat{q}_o(t), \qquad V_t(s) = \widehat{q}_{J_t}(t), \tag{OC-UCB}$$

so the option the selector proposes need not be the one with the largest raw value. When $B_{J_t}(t) > B_{C_t}(t)$ and $Q_t(s,J_t) < Q_t(s,C_t)$ hold at once, the selector asks to explore J_t and the termination test refuses.

C.5 Proof of Theorem 2

Linear lower bound. Take two options. Both actions of the good option G return $\mathrm{Ber}(3/4)$; both actions of the bad option B return exactly $1/2$. Then $\mu_G^\star = 3/4$, $\mu_B^\star = 1/2$, $\Delta_B^{\mathrm{up}} = 1/4$, and every lower-level gap is zero, so any regret is caused by the option choice alone. Initialise in the order $(G,1),(G,2),(B,1),(B,2)$ without applying the termination test, so the current option after step four is B. Let E be the event that both initialisation draws from G return 0, which has probability $(1/4)^2 = 1/16$.

On E both sample means of G are 0 while both of B are $1/2$. We show by induction that the algorithm executes B forever. Assume the current option before step $t \ge 5$ is B and G has received no further samples. If $J_t = B$ then $Q_t(s,C_t) = 1/2 = V_t(s)$, and the convention that ties terminate gives $\beta_t = 1$, after which the reselection returns B. If $J_t = G$ then $Q_t(s,C_t) = 1/2 > 0 = Q_t(s,G) = V_t(s)$, so $\beta_t = 0$ and the step continues with B. Either way B executes and the hypothesis is preserved. On E the bad option therefore runs on all $T-2$ steps from step three onwards, each at pseudo-regret $1/4$, so $\mathcal{R}_T^{\mathrm{gi}} \ge \frac{1}{16}\cdot\frac{1}{4}(T-2) = (T-2)/64$. Since single-step gaps are at most one, $\mathcal{R}_T^{\mathrm{gi}} \le T$ always, and the worst case over the class is $\Theta(T)$.

Logarithmic upper bound on the same instance. That instance has two optimal leaves at $3/4$ and two suboptimal leaves at $1/2$, each with total gap $1/4$. Theorem 6 gives $\mathcal{R}_T^{\mathrm{fix}} \le 2(8\ln T/(1/4) + C_0/4) = 64\ln T + C_0/2$. Hence $\mathcal{R}_T^{\mathrm{fix}}/\mathcal{R}_T^{\mathrm{gi}} \to 0$. The class-level statement combines this with Corollary 7.

Episode resets do not repair it. The continuing counterexample is broken by an unconditional reset of the option at every episode boundary, but resets do not restore a sublinear rate in general. Take a two-step episodic MDP. At the first step, state s_0 admits only the bad option B; its single action returns 0 and moves deterministically to s_1, so the current option on arrival at s_1 is always B. At s_1 both options are available with two actions each; both actions of B return exactly $1/4$, both actions of G return $\mathrm{Ber}(3/4)$, and the episode ends. The optimal episode value is $3/4$ and choosing B at s_1 costs $1/2$. Initialise the four leaves at s_1 over the first four episodes without applying the test, which already costs 1, and let E be as before. On E the critic values at s_1 are 0 for both actions of G and $1/4$ for both actions of B; since the fixed prefix rebuilds $C = B$ at s_1 in every episode, the argument above applies unchanged and B executes at s_1 in all remaining $N-4$ episodes. Hence $\mathbb{E}\,\mathrm{Reg}_N^{\mathrm{gi}} \ge 1 + (N-4)/32 = 1 + (T-8)/64$ with $T = 2N$, while $\beta \equiv 1$ at s_1 is UCB1 over four leaves with gaps $1/2$ and satisfies $\mathbb{E}\,\mathrm{Reg}_N^{\beta\equiv 1} \le 32\ln N + C_0$. Removing this family of

counterexamples needs a condition stronger than resetting the option, for instance a guarantee that every available state–option pair is actually executed and not merely proposed.

C.6 No pointwise dominance

Theorem 2 is a worst-case separation, and the closing paragraph of Section 4.1 records that it cannot be upgraded. Consider a deterministic instance with $\mu_{G,1} = \mu_{G,2} = 1$ and $\mu_{B,1} = \mu_{B,2} = 0$, initialised in the order $(B,1),(B,2),(G,1),(G,2)$, so the current option after initialisation is G with $\widehat{q}_G = 1$ and $\widehat{q}_B = 0$. Under the raw-value indices, a proposal of G gives $Q_t(s, C_t) = V_t(s)$ and terminates into G again, and a proposal of B gives $Q_t(s, C_t) = 1 > 0 = V_t(s)$ and continues with G; so A_{gi} executes G for ever and $\mathcal{R}_T^{\mathrm{gi}} = 2$ for all $T \geq 4$. Under $\beta \equiv 1$ the leaf indices force a return to B: at $t = 11$, $B_B(11) = \sqrt{2\ln 11} > 1 + \sqrt{2\ln 11/4} = B_G(11)$, so $\mathcal{R}_{11}^{\mathrm{fix}} = 3 > 2 = \mathcal{R}_{11}^{\mathrm{gi}}$. The excursions recur, because if $\beta \equiv 1$ pulled B only finitely often its index would grow like $\sqrt{2\ln t/m} \to \infty$ while $B_G(t) \to 1$. The veto in (2) therefore has two faces: it locks in an error when the current option is suboptimal and the unexplored option is not, and it saves exploration cost when the current option is already optimal.

C.7 Relation to the interruption theorem

Sutton et al. (1999) show that interrupting an option whenever continuing is worth less than reselecting under a *fixed* Markov policy μ improves the value everywhere and strictly somewhere. Two hypotheses of that theorem fail for the online process analysed here. The quantities Q_t, V_t are finite-sample estimates and not exact values of a fixed policy, and the upper policy changes with the counts, so there is no fixed μ to improve upon. In the counterexample, G is genuinely worth $3/4$, an unlucky initialisation makes its estimate 0, and the termination rule reads that estimation error as permanent evidence for continuing with B while simultaneously preventing the samples that would correct it. The classical theorem concerns policy improvement once values are known; it makes no claim about this estimation-visitation feedback loop.

C.8 Episodic tabular RL

Let $\mathcal{S}$ be a finite state set with $S = |\mathcal{S}|$, let H be the episode length, N the number of episodes and $T = NH$. Write $\mathcal{U}(s) = \{(o,a) : o \in \mathcal{O}(s), a \in \mathcal{A}_o(s)\}$ for the augmented actions at s and $D = \max_s |\mathcal{U}(s)|$. A Bellman-consistent optimistic algorithm maintains $\overline{Q}_{k,h}(s,o,a)$ from the optimistic Bellman recursion of a single augmented MDP, and selects $a^+_{k,h}(s,o) \in \arg\max_a \overline{Q}_{k,h}(s,o,a)$ below and $o^+_{k,h}(s) \in \arg\max_o \overline{Q}_{k,h}(s,o,a^+_{k,h}(s,o))$ above.

Theorem 8 (Augmented isomorphism)**.** *Under $\beta \equiv 1$ the executed pair satisfies $u_{k,h} \in \arg\max_{u \in \mathcal{U}(s_{k,h})} \overline{Q}_{k,h}(s_{k,h}, u)$, and the hierarchical agent produces exactly the same states, actions, rewards and successor states as the same optimistic algorithm run on $\mathcal{M}^+ = (\mathcal{S}, \mathcal{U}, P, r, H)$.*

Proof. Nested maximisation is joint maximisation: $\overline{Q}(s, o^+, a^+(s,o^+)) = \max_o \max_a \overline{Q}(s,o,a) = \max_u \overline{Q}(s,u)$. With $\beta \equiv 1$ the option is resampled at every step, so no constraint carries over from the previous step and the pair u is free. Executing u yields reward $r(s,u)$ and successor $P(\cdot \mid s,u)$ by definition of $\mathcal{M}^+$, so the observation histories agree, and induction propagates the agreement to all counts and optimistic values. □

Theorem 3 follows by instantiating the optimistic algorithm as UCBVI-BF and quoting its guarantee.

Proof of Theorem 3. Assume known deterministic rewards in $[0,1]$, unknown transitions, a common augmented action set of size D, and failure probability δ. By Theorem 8 the hierarchical agent and UCBVI-BF on $\mathcal{M}^+$ have identical episode regret. Theorem 2 of Azar et al. (2017) gives, with probability at least $1-\delta$ and $L = \ln(5HSDT/\delta)$, $\mathrm{Reg}_N \leq 30HL\sqrt{SDN} + 2500H^2S^2DL^2 + 4H^{3/2}\sqrt{NL}$. Substituting $N = T/H$ turns the first term into $30L\sqrt{HSDT}$ and the third into $4H\sqrt{TL}$. □

Two caveats belong with this bound. If rewards and transitions depend only on the primitive action, distinct pairs (o, a) duplicate one another, and taking $D \leq |\mathcal{O}||\mathcal{A}|$ is correct but loose unless the algorithm pools statistics across options. And UCBVI is model-based: a bonus at each level is not enough to invoke the theorem, since the optimistic values must satisfy the corresponding Bellman recursion and confidence event.

C.9 Summary

Consider any tabular scheme whose two-level selection is equivalent to maximisation over augmented actions. If the behaviour selector and the termination test share one value function and the state value is the maximum over option values, then $\beta \equiv 1$ holds pathwise, the two algorithms coincide, and both inherit whichever guarantee the underlying optimistic algorithm carries. If the selector uses optimistic values while the termination test uses raw values, no uniform sublinear guarantee survives, even at $H = 2$ with per-episode option resets. The two rules are not comparable instance by instance. What can be asserted, and what Section 4.1 asserts, is that a termination rule trained to maximise return is redundant in the consistent case and unguaranteed in the standard case.

D Additional results

Table 4: Learned termination and forced per-step reselection are indistinguishable under exact frozen evaluation. Shortest-path FourRooms, episode-1000 checkpoints, expected capped transitions and success probability under horizon-1000 exact dynamic programming with $\varepsilon_{\text{lo}} = 0$, averaged over the five kernels within seed. Differences are paired over 350 matched seeds with 95% bootstrap intervals; negative favours learned termination. At K=1 termination cannot change which option runs, so that row measures the noise floor of the seed matching.

	Expected steps		Success probability		Paired difference in steps	
K	learned OC	$\beta \equiv 1$	learned OC	$\beta \equiv 1$	mean	95% CI
1	962.49	963.21	0.038	0.037	-0.71	$[-2.62,\ 1.19]$
2	821.44	822.31	0.195	0.194	-0.87	$[-7.93,\ 6.20]$
4	377.20	384.37	0.687	0.680	-7.17	$[-21.34,\ 6.95]$
8	117.35	122.46	0.964	0.959	-5.11	$[-12.77,\ 2.57]$

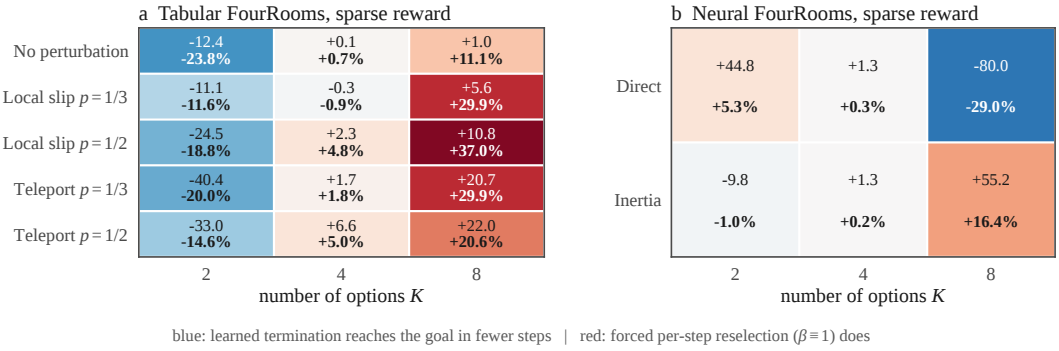

Figure 5: Table 1 as a heat map. Last-100-episode step count of learned termination minus that of a matched $\beta \equiv 1$ agent, in absolute steps (top of each cell) and relative to the control (bottom). Blue favours learned termination and red favours $\beta \equiv 1$. The tabular grid reverses sign completely between K=2 and K=8; the neural grid has no consistent direction.

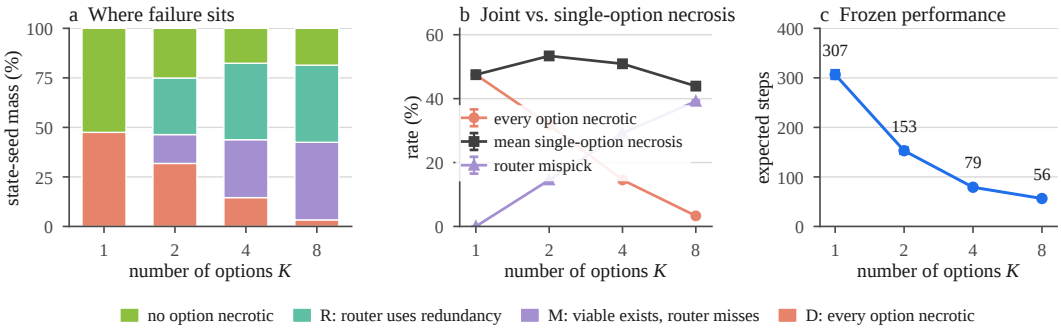

Figure 6: The sparse-reward replication of Figure 4. Episode-1000 control checkpoints of the matched-fork grid, five-kernel average over 350 matched seeds, with necrosis defined by the 1% optimal-action mass criterion of Section 3. Joint necrosis falls $47.5 \to 31.8 \to 14.6 \to 3.3$ percent while the mean single-option rate stays near 50%, and the router's miss rate rises to 39.2%; expected steps fall $306.7 \to 152.9 \to 79.1 \to 56.4$. The effective independent option count is $\kappa = 1.82$, 2.85 and 4.14 at $K = 2, 4, 8$.

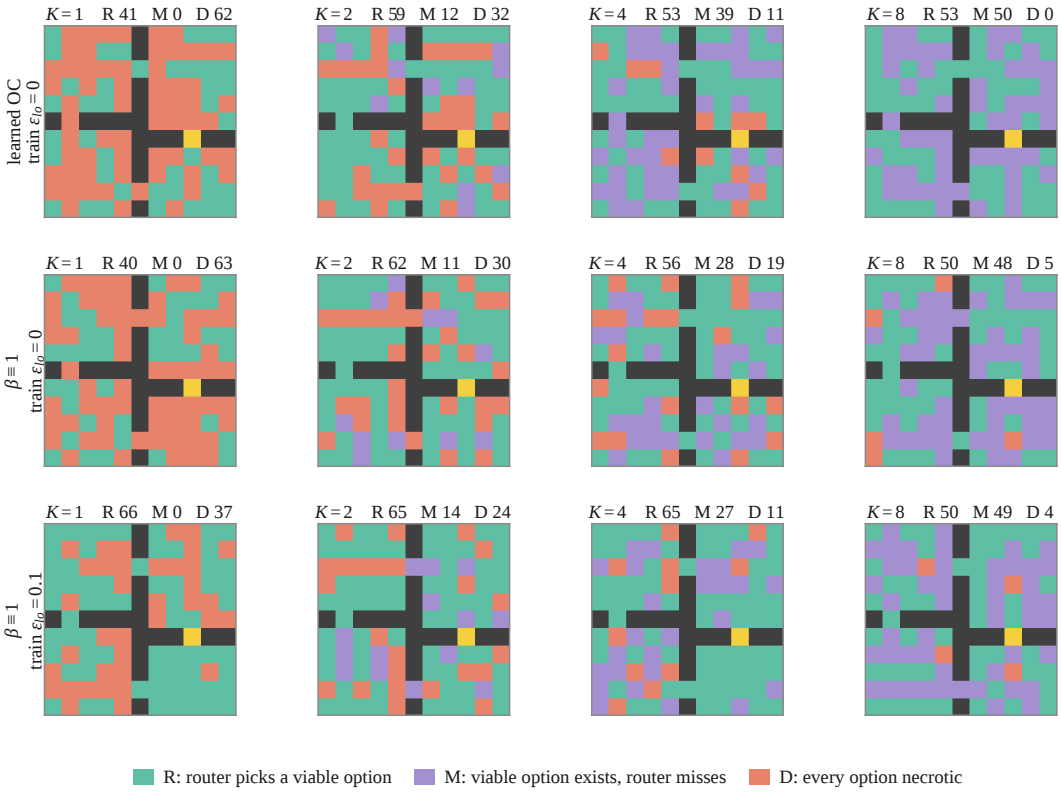

Figure 7: Where coverage appears, and where the router loses it. Every non-goal cell of constant-cost FourRooms under teleport $p = \frac{1}{2}$, at the episode-1000 checkpoint of the spatial-medoid seed, chosen by minimum Hamming distance across all twelve panels without reference to performance. Dark grey is wall and yellow is the goal. The middle row is a $\beta \equiv 1$ agent trained without lower-level exploration; its maps are visually and numerically close to the learned-termination agent above it at every option count, which is the coverage-level statement of Section 4.