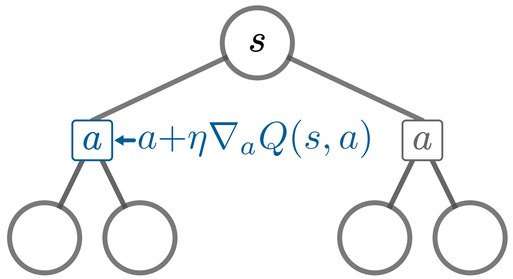
Action-Gradient Monte Carlo Tree Search for Non-Parametric Continuous (PO)MDPs

Idan Lev-Yehudi [1] Michael Novitsky [1] Moran Barenboim [1] Ron Benchetrit [2] Vadim Indelman [3,4]

[1]Technion Autonomous Systems Program [2]Faculty of Computer Science [3]Aerospace Engineering [4]Data and Decision Sciences
Technion – Israel Institute of Technology

AGMCTS (Action-Gradient MCTS) combines *global action sampling*, *local refinement* and *consistent value estimation*

1. Refine Actions Inside Tree Search



MCTS samples distinct action branches, each carrying a $Q(s,a)$ estimate. AGMCTS updates selected actions with gradient information $\nabla_a Q$.

Action Score Gradient: MDP Case

Samples may come from an earlier action through a proposal q:

$$\nabla_a Q_t^\pi(s,a) = \mathbb{E}_{s' \sim q}\Big[\rho_a(s')\Big(\nabla_a \log p_T(s' \mid s,a) \cdot \big(r(s,a,s') + \gamma V_{t+1}^\pi(s') - V(s)\big) + \nabla_a r(s,a,s')\Big)\Big],$$

$$\rho_a(s') = p_T(s' \mid s,a)/q(s').$$

Reuse correction	Transition score	Direct reward
Reweight earlier successors	Favor high-return successors	Immediate reward gradient

Extending to POMDPs via Particle-Belief MDPs

The exact density $p(b' \mid b,a)$ is intractable; we expand through the propagated belief b'^- and observation o':

$$p(b' \mid b,a) = \int p(b' \mid b'^-, o')\, p(o' \mid b'^-) p(b'^- \mid b,a)\, \mathrm{d}b'^-\, \mathrm{d}o'.$$

For an ordered particle belief $\bar{b} = ((s^j, \lambda^j))_{j=1}^J$, it is tractable:

$$p(\bar{b}'^- \mid \bar{b}, a) = \prod_{j=1}^{J} p_T(s'^{-,j} \mid s^j, a).$$

MDP: $p_T(s' \mid s,a) \implies$ POMDP: $\prod_{j=1}^J p_T(s'^{-,j} \mid s^j, a)$
The same action-score formula applies with adjusted likelihood

Scan for paper

2. Reuse the Tree After Action Optimization

Each successor s'^i remembers the proposal action a^i_{prop} under which it was sampled. Let $N_i := n(s'^i)+1$. The MIS Tree estimates future values by

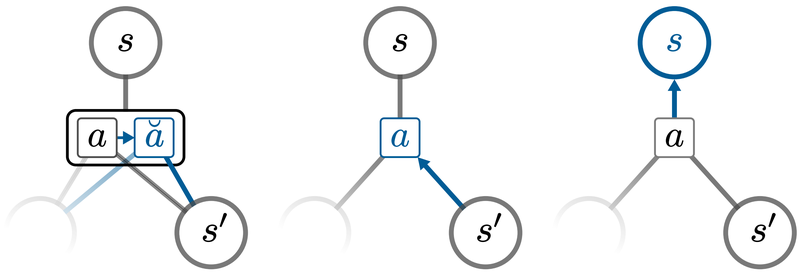
$$V_f(s,a) = \sum_i N_i\, \rho_a^i(s'^i)\, V(s'^i) \;/\; \eta(s,a), \qquad \eta(s,a) = \sum_i N_i \rho_a^i(s'^i).$$

MDP ratio

$$\rho_a^i(s') = p_T(s' \mid s,a)/p_T(s' \mid s, a^i_{\text{prop}}).$$

POMDP ratio

$$\rho_a^i(\bar{b}'^-) = p(\bar{b}'^- \mid \bar{b}, a)/p(\bar{b}'^- \mid \bar{b}, a^i_{\text{prop}}).$$



1. Action Update
$a \to \breve{a}$:
Update ratios in $O(|\mathcal{S}_{sa}|)$

2. Action Backprop
$V(s') \to \breve{V}(s')$:
Update $Q(s,a)$ in $O(1)$

3. State Backprop
$Q(s,a) \to \breve{Q}(s,a)$:
Update $V(s)$ in $O(1)$

Value estimates are defined through $V(s) = \sum_a n(s,a)Q(s,a) \;/\; \sum_a n(s,a)$.

Transition Densities from Smooth Generative Models

Suppose $s' = F(s,a,\xi)$ is locally Lipschitz, with $\operatorname{rank}(D_\xi F) = n_\xi$ almost everywhere. For noise $\xi \sim p_\xi(\cdot \mid s,a)$, the Area Formula gives the density induced on the successor manifold:

$$p_T(s' \mid s,a) = \sum_{\xi^* : F(s,a,\xi^*)=s'} p_\xi(\xi^* \mid s,a) \;/\; J_{n_\xi} F(s,a,\xi^*),$$

$$J_{n_\xi} F = \sqrt{\det((D_\xi F)^\intercal D_\xi F)}.$$

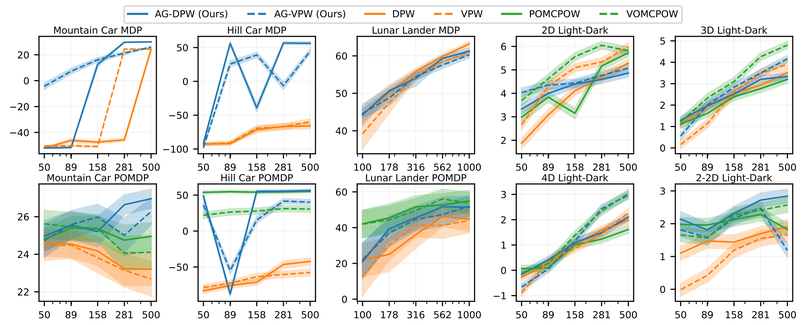
Noise before the simulator
$h(s,a,\xi) = (\widetilde{s}, \widetilde{a}), \quad s' = f(\widetilde{s}, \widetilde{a})$
$D_\xi F = D_{(\widetilde{s},\widetilde{a})} f \cdot \partial h / \partial \xi.$
Unique noise inverse: reuse the cached simulator Jacobian.

Noise after the simulator
$s' = g(h(s,a), \xi)$
$D_\xi F = \partial g / \partial \xi \big|_{h(s,a),\xi^*}.$
Solvable noise preimages: one forward simulator evaluation.

3. Does Refinement Improve Planning?

We compare Double Progressive Widening (DPW) and Voronoi Progressive Widening (VPW) to their Action-Gradient (AG-DPW/AG-VPW) variants. In POMDPs, we compare to POMCPOW and VOMCPOW as well.

We evaluate in 3 MDPs and 7 POMDPs with continuous states, actions and observations, exhibiting stochastic transition and observation models.



Average return over 1000 shared seeds; shading shows ±2 standard errors. The x-axis is the simulation budget per planning session. AG-DPW and AG-VPW are AGMCTS variants; PFT denotes the particle-filter-tree POMDP adaptation.

Empirical Bottom Lines

6/10	10 / 7 / 3	5/7
scenarios: at least one AG variant preferable to its matched baseline	improve / neither / underperform in matched AG comparisons	POMDP scenarios: AGMCTS best or statistically tied

- **AG-preferred scenarios:** all Mountain/Hill Car settings, 4D and two-agent Light-Dark
- **Largest gains:** Mountain/Hill Car; Light-Dark and Lunar Lander show more mixed outcomes

Runtime Trade-Off

$10\text{–}35\times$	$2\text{–}20\times$
single-threaded per-plan overhead in MDPs	single-threaded per-plan overhead in POMDPs

Practical trade-off: whether improved action quality per planner query is worth the additional density/gradient overhead within latency budget

Takeaway

Best fit: small action changes strongly affect states or rewards, like in long-horizon Mountain Car or continuous-reward Light-Dark

Requires: transition-density access and more hyperparameter tuning