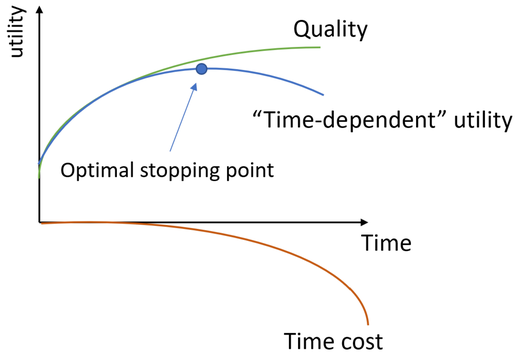
Tuning the Hyperparameters of Anytime Planning: A Metareasoning Approach with Deep Reinforcement Learning

Abhinav Bhatia[1], Justin Svegliato[2], Samer B. Nashed[1], Shlomo Zilberstein[1]

[1]College of Information and Computer Sciences, University of Massachusetts Amherst
[2]Department of Electrical Engineering and Computer Sciences, University of California Berkeley

Metareasoning for Anytime Algorithms

- **Anytime Algorithms** iteratively improve solutions
- **Time-dependent Utility (TDU):** solution quality + compution-time cost



- **Metareasoning with Deep RL**: **Control hyperparameters** *and* **optimal stopping** point **online**, **based on internal state features** of an execution.
- **Decision Theoretic**: Rewards improvement in TDU to **optimize final TDU**.

Anytime Heuristic Search

- Anytime Weighted A* inflates heuristic: $f_w(n) = g(n) + w \cdot h(n)$, $w \geq 1$.

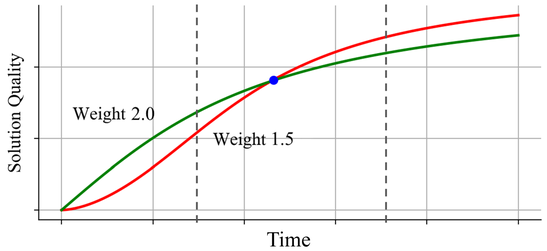


Figure: Quality-time tradeoff with AWA* weight w. Higher weights lead to better quality in short-term.

Metareasoning for Anytime Weighted A*

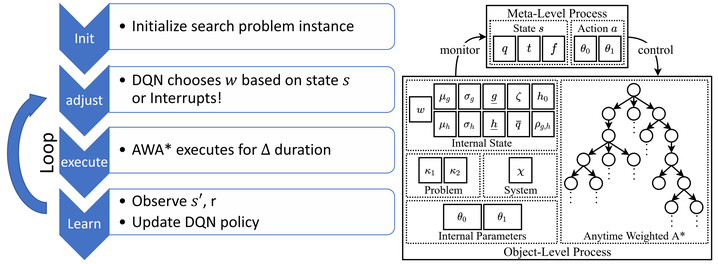


Figure: Metareasoning architecture. An episode of DQN corresponds to one execution of AWA*.

Metareasoning for Anytime Weighted A*: Experiments

- Time cost: exponential in time t until deadline $\tau = 1$ second
- Baslines: i) Static, ii) DEC: decreases w after each solution and executes until τ.
- Metareasoning DQN_τ: Adjusts w and executes until deadline τ
- Metareasoning $\text{DQN}(t)$: Adjusts w and can interrupt before deadline.

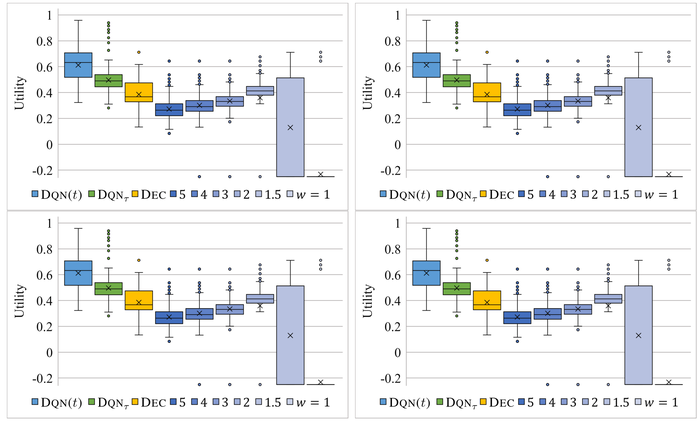


Figure: Box plots for final time-dependent utilities for each approach over all instances of *Sliding Puzzle*, *Inverse Sliding Puzzle*, *Travelling Salesman Problem*, *Grid Navigation Problem*

Metareasoning for Anytime Weighted A*: Analysis

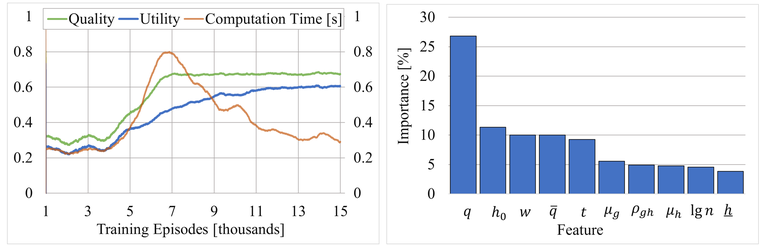


Figure: (a) Training curve for DQN on *Sliding Puzzle* domain. Initially, DQN appears to focus on learning to improve solution quality and later, focus on learning to reduce compution time.
(b) Sensitivity of the trained DQN network to various features, in decreasing order: quality q, instance starting state heuristic h_0, weight w, compution time t, and various statistics of the open list.

Metareasoning for Path Planning with RRT*

- Adjustables: Growth Factor, Focus Region
- Deadline: 1000 samples

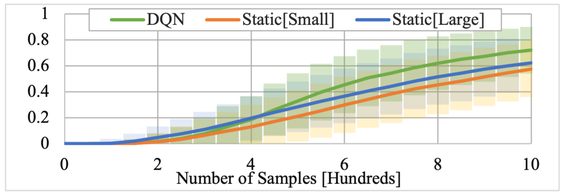


Figure: **Average quality vs compution time** for DQN and static growth-factor approaches

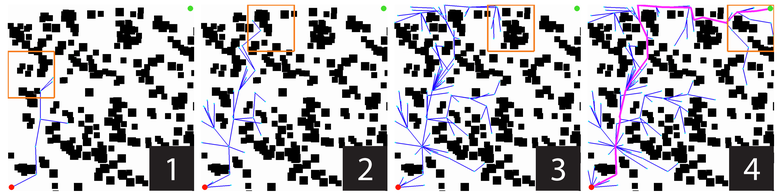


Figure: Snapshots from an instance of the **metareasoner biasing tree growth** by moving focus region to guide the tree to the goal.