Previous Knowledge Utilization in Online Non-Parametric Belief Space Planning

Presented by: **Michael Novitsky**
Supervisor: **Assoc. Prof. Vadim Indelman**

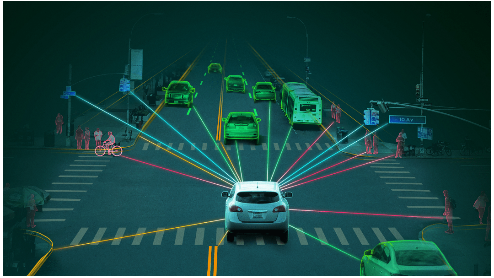
August 14th 2024

Outline

ANPL
Autonomous Navigation and Perception Lab

Motivation

Autonomous agents



Motivation

Autonomous agents

Introduction

Perception - environment representation and understanding usually deals with uncertainty

- Noisy measurements
- Partial information
- Multiple input sources

Introduction

Decision making - planning into the future and taking actions

- Uncertainty handling
- Dynamic environments
- Safety
- Predicting the future

ANPL Autonomous Navigation and Perception Lab

Introduction

Decision making usually starts from scratch at each planning session while discarding previous information

- Previous planning sessions might have valuable data
- Reusing previous information can be efficient and time saving

Partially Observable Markov Decision Process (POMDP)

A mathematical framework for modeling decision-making problems under uncertainty, with each individual problem being characterized by a 7-tuple: (S, A, O, T, Z, R, b_k).

- S - State space
- A - Action space
- O - Observation space
- $\mathbb{P}_T(s_{k+1}|s_k, a_k)$ - State transition density function
- $\mathbb{P}_Z(o_{k+1}|s_{k+1})$ - Observation density function
- $R(s_k, a, s_{k+1})$ - State reward function
- b_k - Current belief (probability over states)

ANPL Autonomous Navigation and Perception Lab

POMDP - Non-Parametric Distributions Estimation

- Statistical techniques are used to estimate probability distributions from samples
- No parametric assumptions regarding the functional form of a distribution
- Each belief is approximated with a finite number of state samples

ANPL Autonomous Navigation and Perception Lab

POMDP - Belief-Dependent Reward

Typical reward function of belief node is formulated as expected reward over states $r(b, a, b') = \mathbb{E}_{s' \in b}[r(s, a, s')]$, but it is not always enough and in many applications belief dependent reward is needed

- Quantify uncertainty using information-theoretic measures, such as information gain and differential entropy
- Needed in information gathering, active sensing and other tasks
- Typically more computationally demanding than the expected state reward.
  - expected state reward - $O(n)$
  - differential entropy (boers10) - $O(n^2)$

ANPL Autonomous Navigation and Perception Lab

POMDP - Belief-MDP

Every POMDP problem (S, A, O, T, Z, R, b_k) can be viewed as MDP over the belief space (B, A, τ, R, b_0)

- B - space of all possible beliefs over states
- A - same as in POMDP definition
- $\tau(b_{k+1}|b_k, a_k)$ - belief transition function
- R - same as in POMDP definition
- b_0 - same as in POMDP definition

POMDP - Policy, Value Function, Action Value Function, Return

Policy $\pi \in \Pi$ is a mapping from belief space B to action space A
$\pi : b \rightarrow a$.
$G_k = \sum_{i=k}^{k+L-1} \gamma^{i-k} r(b_i, \pi(b_i), b_{i+1})$ is the return.

- $V^\pi(b) = \underset{\pi}{\mathbb{E}}[G_k | b_k = b]$
- $Q^\pi(b, a) = \underset{\pi}{\mathbb{E}}[G_k | b_k = b, a_k = a]$

ANPL Autonomous Navigation and Perception Lab

POMDP - Autonomy Loop

True state of the agent is unknown, instead it maintains a belief (distribution over states)

- $H_k = (b_0, a_0, o_1, a_1, o_2, .., a_{k-1}, o_k) = \{o_{1:k}, a_{1:k-1}\}$
- $b_k = \mathbb{P}(s_k | H_k)$

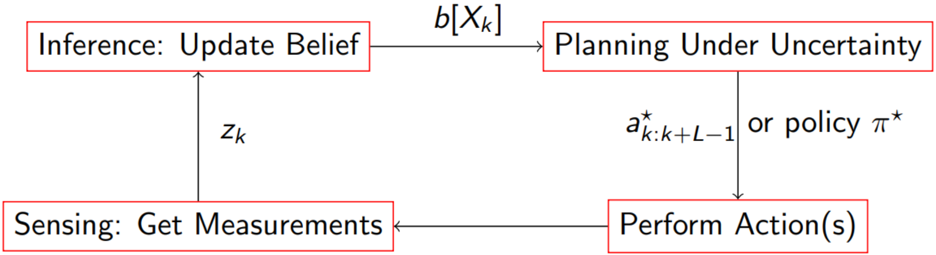


Figure: ANP - 086762

POMDP - Computational Complexity

Solving POMDPs is hard

- Curse of dimensionality
- Curse of history
- Continuous state space
- Continuous observation space
- Continuous action space

POMDP - Online Algorithms

Sample the belief space and build a partial belief tree/graph

- Operate within limited budget constraints
- Anytime property - adapt and improve solutions as more samples are generated
- Find optimal action/policy according to sampled tree

POMDP - Online Algorithms

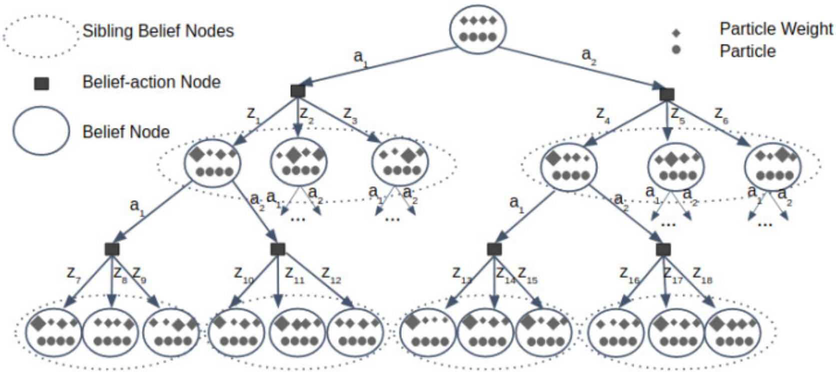


Figure: Garg 2019

Online Algorithms - MCTS

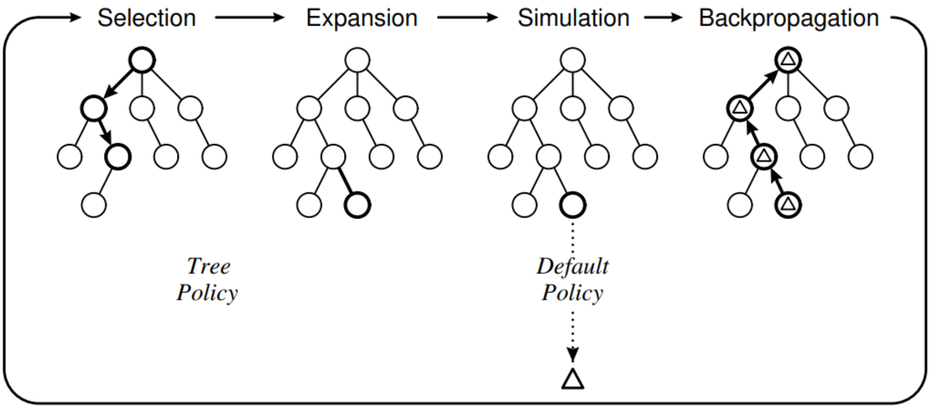


Figure: Browne et al. 2012

MCTS - Explanation

Four consecutive steps are applied in each iteration

- Selection - Starting from root node a child selection policy is applied recursively until the chosen expandable node is reached
- Expansion - A child node is added to expand the tree according to available actions
- Simulation - A pre-defined policy is applied to generate trajectory until horizon depth is reached
- Backpropagation - Simulation result is propagated back to the parents until root node is reached

ANPL Autonomous Navigation and Perception Lab

POMDP - Related Work Online Solvers

- Online solvers
  - POMCP (2010 Silver et al.)
  - Despot (2013 Somani et al.)
  - POMCPOW (2018 Sunberg et al.)
- Online solvers with belief dependent rewards
  - PFT-DPW (2018 Sunberg et al.)
  - IPFT (2020 Fischer et al.)
  - ρ-POMCP (2020 Thomas et al.)
- Calculation reuse
  - iX-BSP(2021 Farhi and Indelman)

ANPL Autonomous Navigation and Perception Lab

POMDP - Related Work Online Solvers

Algorithm	S	A	O	R	Reuse
POMCP	Continuous	Discrete	Discrete	State	Trivial
Despot	Continuous	Discrete	Discrete	State	Trivial
POMCPOW	Continuous	Continuous	Continuous	State	No
PFT-DPW	Continuous	Continuous	Continuous	Belief	No
IPFT	Continuous	Continuous	Continuous	Belief	No
ρ-POMCP	Continuous	Continuous	Continuous	Belief	No
iX-BSP	Continuous	Discrete	Continuous	Belief	Yes
IR-PFT	Continuous	Continuous	Continuous	Belief	Yes

ANPL
Autonomous Navigation and Perception Lab

Our Motivation

Solving POMDPs with continuous state, action and observation spaces

- The probability to sample same belief twice is zero.
- Each planning session starts only with root node and previous information is discarded
- Previously sampled beliefs can still provide useful information during current planning session
- We want to use previous trajectories to get efficient estimation of $Q^{\pi}(b,a) = \underset{\pi}{\mathbb{E}}[G_k | b_k = b, a_k = a]$

ANPL Autonomous Navigation and Perception Lab

Contributions

- Theoretical justification for information reuse in a non-parametric setting
- Novel MCTS-based algorithm that incorporates information reuse

Outline

ANPL
Autonomous Navigation and Perception Lab

Outline

ANPL
Autonomous Navigation and Perception Lab

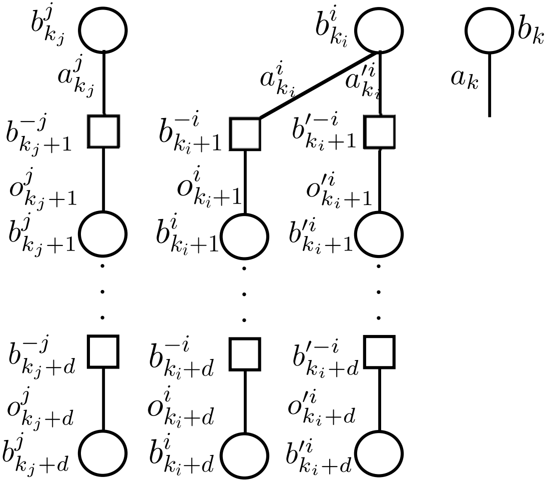
Approach - Experience-Based Value Function Estimation

We assume access to a dataset D containing trajectories with horizon d and returns of an expert agent that followed policy π, current belief b_k and action a_k

- $D \triangleq \{\tau^i, G^i\}$
- $\tau^i \triangleq (b^i_{k_i}, a^i_{k_i}) \rightarrow (b^{-i}_{k_i+1}, o^i_{k_i+1}, b^i_{k_i+1} a^i_{k_i+1}) \rightarrow \ldots \rightarrow (b^{-i}_{k_i+d}, o^i_{k_i+d}, b^i_{k_i+d})$

ANPL Autonomous Navigation and Perception Lab

Approach - Experience-Based Value Function Estimation



Approach - Experience-Based Value Function Estimation

How to estimate $Q^{\pi}(b_k, a_k)$?

- Continuous state, action and observation spaces
- The probability that we have trajectory that starts with b_k and a_k is 0

ANPL Autonomous Navigation and Perception Lab

Approach - Experience-Based Value Function Estimation

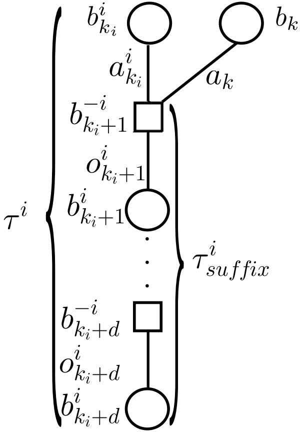


Figure: Illustration of τ'

Approach - Experience-Based Value Function Estimation

Given trajectory $\tau^i \in D$

- $\tau^i = (b^i_{k_i}, a^i_{k_i}) \to \tau^i_{suffix}$
- $\tau^i_{suffix} \triangleq (b^{-i}_{k_i+1}, o^i_{k_i+1}, b^i_{k_i+1} a^i_{k_i+1}) \to ... \to (b^{-i}_{k_i+d}, o^i_{k_i+d}, b^i_{k_i+d})$

and current belief b_k and action a_k we construct new trajectory τ'^i

- $\tau'^i = (b_k, a_k) \to \tau^i_{suffix}$

ANPL Autonomous Navigation and Perception Lab

Approach - Experience-Based Value Function Estimation

To estimate $Q^{\pi}(b_k, a_k)$ using trajectory τ'^{i} two adjustments are required

- $\tilde{G}^i \triangleq G^i - r(b^i_{k_i}, a^i_{k_i}, b^i_{k_{i+1}}) + r(b_k, a_k, b^i_{k_{i+1}})$
- Adjust the likelihood of $\tilde{G}^i$ since $\mathbb{P}(\tau^i_{suffix}|b^i_{k_i}, a^i_{k_i}, \pi) \neq \mathbb{P}(\tau^i_{suffix}|b_k, a_k, \pi)$

ANPL Autonomous Navigation and Perception Lab

Approach - Experience-Based Value Function Estimation

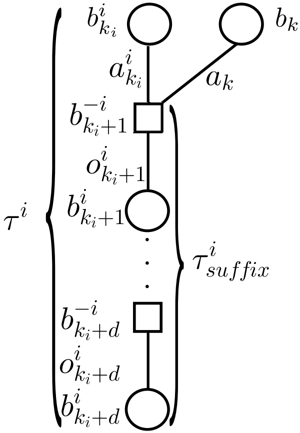


Figure: Illustration of τ'

Outline

Approach - Importance Sampling

Given target distribution $p(x)$ and proposal distribution $q(x)$

$\mathbb{E}_p[f(x)] \approx \frac{1}{N}\sum_{i=1}^{N} w_i \cdot f(x^i), w_i = \frac{p(x^i)}{q(x^i)}, x^i \sim q.$

q must satisfy $q(x^i) = 0 \Rightarrow p(x^i) = 0$

ANPL Autonomous Navigation and Perception Lab

Approach - Importance Sampling Action Value Function Estimator

Given N_{IS} partial trajectories sampled from the same proposal distribution $\mathbb{P}(\cdot|b^i_{k_i}, a^i_{k_i}, \pi)$ and target distribution $\mathbb{P}(\cdot|b_k, a_k, \pi)$ we define Importance Sampling estimator $\hat{Q}^{\pi}_{IS}(b_k, a_k) \triangleq \frac{1}{N_{IS}} \sum_{i=1}^{N_{IS}} w_i \cdot \tilde{G}^i$

- $w_i \triangleq \frac{\mathbb{P}(\tau^i_{suffix}|b_k,a_k,\pi)}{\mathbb{P}(\tau^i_{suffix}|b^i_{k_i},a^i_{k_i},\pi)}$
- Actually we have many different distributions so we want to use Multiple Importance Sampling

ANPL Autonomous Navigation and Perception Lab

Approach - Experience-Based Value Function Estimation

We designate by M the count of unique distributions $\{\mathbb{P}(\cdot|b^m_{k_m}, a^m_{k_m}, \pi)\}_{i=1}^{M}$ from which partial trajectories originate and we denote the sample count from each distribution as n_m.
Dataset D can be reformulated

- $D \triangleq \{\tau^{l,m}, G^{l,m}\}_{m=1,l=1}^{M,n_m}$.

Outline

- Introduction And Motivation
- Related Work
- Approach
  - Experience-Based Value Function Estimation
    - Importance Sampling
    - Multiple Importance Sampling
    - Action Value Function Estimator
  - IR-PFT Algorithm
    - IR-PFT Summary
- Evaluation
- Conclusions

ANPL
Autonomous Navigation and Perception Lab

Approach - Multiple Importance Sampling

$$\mathbb{E}_p[f(x)] \approx \sum_{m=1}^{M} \frac{1}{n_m} \sum_{i=1}^{n_m} w_m(x_{i,m}) f(x_{i,m}) \frac{p(x_{i,m})}{q_m(x_{i,m})}$$

- M proposal distributions $\{q_m\}_{m=1}^{M}$
- n_m - number of samples from distribution q_m
- $x_{i,m}$ - i'th sample from distribution q_m
- w_m is a weighting function that must satisfy
  - $q_m(x_{i,m}) = 0 \Rightarrow w_m(x_{i,m}) f(x_{i,m}) p(x_{i,m}) = 0$
  - $f(x_{i,m}) \neq 0 \Rightarrow \sum_{m=1}^{M} w(x_{i,m}) = 1$

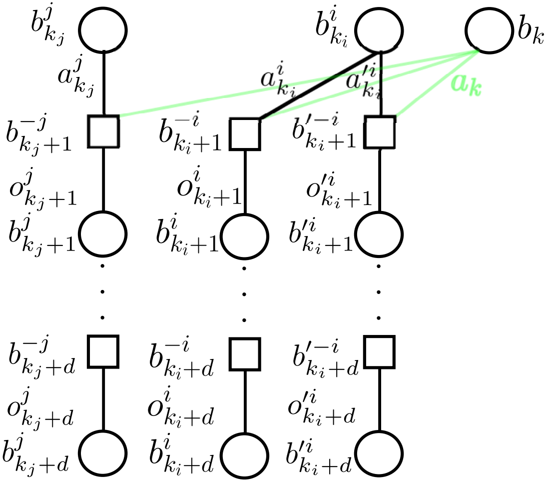
ANPL Autonomous Navigation and Perception Lab

Approach - Multiple Importance Sampling Action Value Function Estimation

Using Multiple Importance Sampling Estimator (Assuming Balance Heuristic)

$$\hat{Q}^{\pi}_{MIS}(b_k, a_k) \triangleq \sum_{m=1}^{M} \sum_{l=1}^{n_m} \frac{\mathbb{P}(\tau^{l,m}_{suffix} | b_k, a_k, \pi) \tilde{G}^{l,m}}{\sum_{j=1}^{M} n_j \cdot \mathbb{P}(\tau^{l,m}_{suffix} | b^j_{k_j}, a^j_{k_j}, \pi)}.$$

Approach - Experience-Based Value Function Estimation



ANPL
Autonomous Navigation and Perception Lab

Approach - Experience-Based Value Function Estimation

How to calculate a single term $\mathbb{P}(\tau^i_{suffix}|b^i_{k_i}, a^i_{k_i}, \pi)$?
Using Markov assumptions and chain rule!

$$\mathbb{P}(\tau^i_{suffix}|b^i_{k_i}, a^i_{k_i}, \pi) = \mathbb{P}(b^{-i}_{k_i+1}, o^i_{k_i+1}, \dots, b^i_{k_i+L}|b^i_{k_i}, a^i_{k_i}, \pi) =$$
$$\mathbb{P}(b^{-i}_{k_i+1}|b^i_{k_i}, a^i_{k_i}) \cdot \mathbb{P}(o^i_{k_i+1}, \dots, b^i_{k_i+L}|b^{-i}_{k_i+1}, \pi) =$$
$$\mathbb{P}(b^{-i}_{k_i+1}|b^i_{k_i}, a^i_{k_i}) \cdot \mathbb{P}(o^i_{k_i+1}|b^{-i}_{k_i+1}) \cdot \mathbb{P}(b^i_{k_i+2}, \dots, b^i_{k_i+L}|b^{-i}_{k_i+1}, o^i_{k_i+1}, \pi) =$$
$$\mathbb{P}(b^{-i}_{k_i+1}|b^i_{k_i}, a^i_{k_i}) \cdot \mathbb{P}(o^i_{k_i+1}|b^{-i}_{k_i+1}) \cdot \mathbb{P}(b^i_{k_i+2}|b^{-i}_{k_i+1}, o^i_{k_i+1}) \cdot$$
$$\mathbb{P}(a^i_{k_i+2}, \dots, b^i_{k_i+L}|b^i_{k_i+2}, \pi) = \mathbb{P}(b^{-i}_{k_i+1}|b^i_{k_i}, a^i_{k_i}) \cdot \mathbb{P}(o^i_{k_i+1}|b^{-i}_{k_i+1}) \cdot$$
$$\mathbb{P}(b^i_{k_i+2}|b^{-i}_{k_i+1}, o^i_{k_i+1}) \cdot \mathbb{P}(a^i_{k_i+2}|b^i_{k_i+2}, \pi) \cdot \mathbb{P}(b^{-i}_{k_i+2}, \dots, b^i_{k_i+L}|b^i_{k_i+1}, a^i_{k_i+1}, \pi) \dots$$

ANPL Autonomous Navigation and Perception Lab

Approach - Experience-Based Value Function Estimation

How to calculate a single term $\mathbb{P}(\tau^i_{suffix}|b^i_{k_i}, a^i_{k_i}, \pi)$?

- $\mathbb{P}(\tau^i_{suffix}|b^i_{k_i}, a^i_{k_i}, \pi) = \prod_{j=1}^{d} \mathbb{P}(b^{-i}_{k_i+j}|b^i_{k_i+j-1}, a^i_{k_i+j-1}) \prod_{l=1}^{d} \mathbb{P}(o^i_{k_i+l}|b^{-i}_{k_i+l}) \cdot \prod_{n=2}^{d} \mathbb{P}(b^i_{k_i+n}|b^{-i}_{k_i+n}, o^i_{k_i+n}) \cdot \prod_{m=2}^{d-1} \mathbb{P}(a^i_{k_i+m}|b^i_{k_i+m}, \pi)$

Luckily we don't have to make the full calculation!

Trajectory Likelihood Ratio

Lemma 1

$$\frac{\mathbb{P}(\tau^{i}_{suffix}|b_k,a_k,\pi)}{\mathbb{P}(\tau^{i}_{suffix}|b^{i}_{k_i},a^{i}_{k_i},\pi)} = \frac{\mathbb{P}(b^{-i}_{k_i+1}|b_k,a_k)}{\mathbb{P}(b^{-i}_{k_i+1}|b^{i}_{k_i},a^{i}_{k_i})}$$

Trajectory Likelihood Ratio

Theorem 1

$$\frac{\mathbb{P}(\tau^i_{suffix}|b_k,a_k,\pi)}{\mathbb{P}(\tau^i_{suffix}|b^i_{k_i},a^i_{k_i},\pi)} = \frac{\mathbb{P}(b^{-i}_{k_i+1}|b_k,a_k)}{\mathbb{P}(b^{-i}_{k_i+1}|b^i_{k_i},a^i_{k_i})}$$

Proof.

$$\frac{\mathbb{P}(\tau^i_{suffix}|b_k,a_k,\pi)}{\mathbb{P}(\tau^i_{suffix}|b^i_{k_i},a^i_{k_i},\pi)} = \frac{\mathbb{P}(b^{-i}_{k_i+1},o^i_{k_i+1},\ldots,b^i_{k_i+L}|b_k,a_k,\pi)}{\mathbb{P}(b^{-i}_{k_i+1},o^i_{k_i+1},\ldots,b^i_{k_i+L}|b^i_{k_i},a^i_{k_i},\pi)} =$$

$$\frac{\mathbb{P}(b^{-i}_{k_i+1}|b_k,a_k)}{\mathbb{P}(b^{-i}_{k_i+1}|b^i_{k_i},a^i_{k_i})} \cdot \cancel{\frac{\mathbb{P}(o^i_{k_i+1},\ldots,b^i_{k_i+L}|b^{-i}_{k_i+1},\pi)}{\mathbb{P}(o^i_{k_i+1},\ldots,b^i_{k_i+L}|b^{-i}_{k_i+1},\pi)}} = \frac{\mathbb{P}(b^{-i}_{k_i+1}|b_k,a_k)}{\mathbb{P}(b^{-i}_{k_i+1}|b^i_{k_i},a^i_{k_i})} \qquad \square$$

Outline

ANPL Autonomous Navigation and Perception Lab

Approach - Experience-Based Value Function Estimation

Using Theorem 1 we can rewrite Multiple Importance Sampling Estimator
$\hat{Q}^{\pi}_{MIS}(b_k, a_k) \triangleq \sum_{m=1}^{M} \sum_{l=1}^{n_m} \frac{\mathbb{P}(b^{-l,m}_{k_m+1}|b_k,a_k)\tilde{G}^{l,m}}{\sum_{j=1}^{M} n_j \cdot \mathbb{P}(b^{-l,m}_{k_m+1}|b^{j}_{k_j},a^{j}_{k_j})}$.

ANPL Autonomous Navigation and Perception Lab

Approach - Experience-Based Value Function Estimation

Given belief b_k, action a_k and dataset D

- Demonstrated estimation of action value function without planning
- Next we will show our algorithm IR-PFT

ANPL
Autonomous Navigation and Perception Lab

Outline

Approach - IR-PFT Algorithm

We name our algorithm Incremental Reuse Particle Filter Tree (IR-PFT). It is based on the PFT algorithm and incorporates trajectories from previous planning sessions for fast estimation of $Q(b_k, a_k)$.

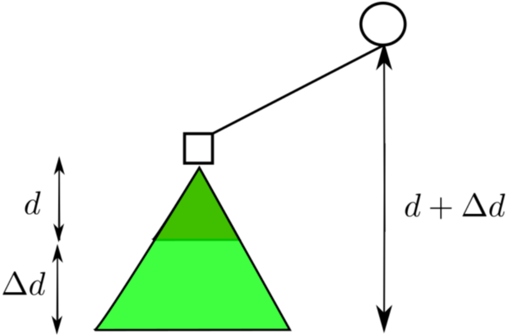
Approach - IR-PFT Algorithm

Two issues that must be addressed before reusing previous trajectories

- Horizon of previous trajectories is shorter than current Horizon
- Previous trajectories were sampled from different distributions

Approach - IR-PFT Algorithm

Align the horizon using Postorder traversal before reusing previous trajectories, as they have a shorter horizon



Incremental Multiple Importance Sampling Update

Lemma 2

Given an MCTS tree T with horizon d, number of simulations m and N nodes, extending its horizon by Δd will require adding at most $m \cdot \Delta d$ nodes.

Horizon Extension In MCTS

Theorem 2

Given a MCTS tree T with horizon d, number of simulations m and N nodes, extending its horizon by Δd will require adding at most $m \cdot \Delta d$ nodes.

Proof.

After m simulations, the MCTS tree T contains at most m leaves and we need to extend each leaf by Δd $\square$

Approach - IR-PFT Algorithm

To account for different distributions of trajectories, we use the same update as in the previous section
$\hat{Q}_{MIS}(b_k, a_k) \triangleq \sum_{m=1}^{M} \sum_{l=1}^{n_m} \frac{\mathbb{P}(b_{k_m+1}^{-l,m}|b_k,a_k)\tilde{G}^{l,m}}{\sum_{j=1}^{M} n_j \cdot \mathbb{P}(b_{k_m+1}^{-l,m}|b_{k_j}^{j},a_{k_j}^{j})}$.

- The tree policy varies between different simulations, and the trajectory distribution is non-stationary. Consequently, the update of $\hat{Q}_{MIS}(b_k, a_k)$ operates in a heuristic manner and its proof yet to be established.
- Regular calculation of $\hat{Q}_{MIS}(b_k, a_k)$ will take $O(M^2 \cdot n_{avg})$ time

ANPL
Autonomous Navigation and Perception Lab

Incremental Multiple Importance Sampling Update

Lemma 3

Given a batch of L samples from identical distribution m', $\hat{\mathbb{E}}_p^{MIS}[f(x)] = \sum_{m=1}^{M} \sum_{i=1}^{n_m} \frac{p(x_{i,m})}{\sum_{j=1}^{M} n_j \cdot q_j(x_{i,m})} f(x_{i,m})$ can be efficiently updated with $O(M \cdot n_{avg} + M \cdot L)$ time and $O(M \cdot n_{avg})$ memory complexity.

Incremental Multiple Importance Sampling Update

We look at $\sum_{i=1}^{n_m} \frac{p(x_{i,m})}{\sum_{j=1}^{M} n_j \cdot q_j(x_{i,m})} f(x_{i,m})$

- In case $m \neq m'$
  - $\sum_{j=1}^{M} n_j \cdot q_j(x_{i,m}) \leftarrow \sum_{j=1}^{M} n_j \cdot q_j(x_{i,m}) + L \cdot q_{m'}(x_{i,m})$
- Time complexity complexity $O(M \cdot n_{avg})$
- Space complexity complexity $O(M \cdot n_{avg})$

ANPL
Autonomous Navigation and Perception Lab

Incremental Multiple Importance Sampling Update

We look at $\sum_{i=1}^{n_m} \frac{p(x_{i,m})}{\sum_{j=1}^{M} n_j \cdot q_j(x_{i,m})} f(x_{i,m})$

- In case $m \triangleq m'$
  - $\sum_{j=1}^{M} n_j \cdot q_j(x_{i,m}) \leftarrow \sum_{j=1}^{M} n_j \cdot q_j(x_{i,m}) + L \cdot q_{m'}(x_{i,m})$
  - Calculate $\frac{p(x_{i,m})}{\sum_{j=1}^{M} n_j \cdot q_j(x_{i,m})} f(x_{i,m})$
- Time complexity complexity $O(M \cdot L)$
- Space complexity complexity $O(M \cdot L)$

ANPL
Autonomous Navigation and Perception Lab

Approach - IR-PFT Algorithm

Determining reuse candidate b^- for belief b and action a

- Large visitation count - $N(b^-) > N_{th}$
- $b^- = argmin_{b^-}\{f_D(b^-, b, a)\}$ where f_D is a function that measures how close is b^- to propagated beliefs sampled from $\mathbb{P}(\cdot|b, a)$
  - f_D is computed across the entire dataset, needs to be cheap for evaluation.
  - An example to f_D is $||\mathbb{E}[b^- - b^-_{MLE}]||_2^2$ where b^-_{MLE} is maximul likelihood propagated belief given belief b and action a
  - If the probability $\mathbb{P}(b^-|b, a)$ is low, the estimator will still be consistent

ANPL Autonomous Navigation and Perception Lab

Approach - IR-PFT Algorithm

Deciding on the balance between reusing previous and opening new trajectories

- Previous trajectories are cheaper to evaluate and we get a speedup in the processing time
- Previous trajectories might be less relevant to current belief b and action a so we still want to generate new trajectories

As a compromise we aim for the same ratio of reused and non-reused propagated beliefs

Outline

Approach - IR-PFT Algorithm

Algorithm summary in case reuse possible for b, a

- $b^- = argmin_{b^-}\{f_D(b^-, b, a)\}$
- $FillHorizon(b^-)$
- $N(b) \leftarrow N(b) + N(b^-)$
- $N(ba) \leftarrow N(ba) + N(b^-)$
- $Q(ba) \leftarrow IncrementalMISUpdate()$
- $C(ba) \leftarrow C(b, a) \cup \{b^-\}$

ANPL Autonomous Navigation and Perception Lab

Outline

ANPL Autonomous Navigation and Perception Lab

Evaluation

We assess IR-PFT by comparing it to the PFT algorithm (Sunberg18). Our evaluation focuses on two main aspects

- Runtime
- Accumulated reward

with statistics measured for each. Each algorithm was evaluated using different quantities of particles.

Evaluation

All experiments were conducted using the standard 2D Light Dark benchmark, wherein the agent is trying to reach goal while minimizing location uncertainty.

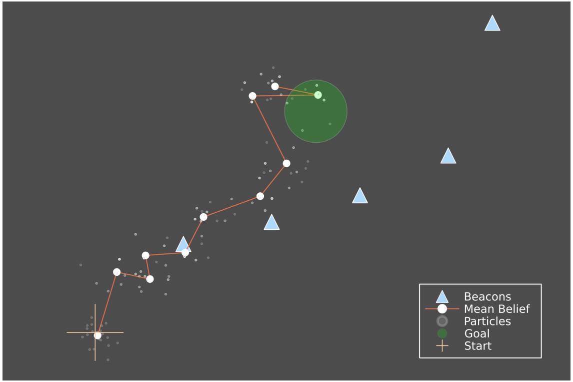


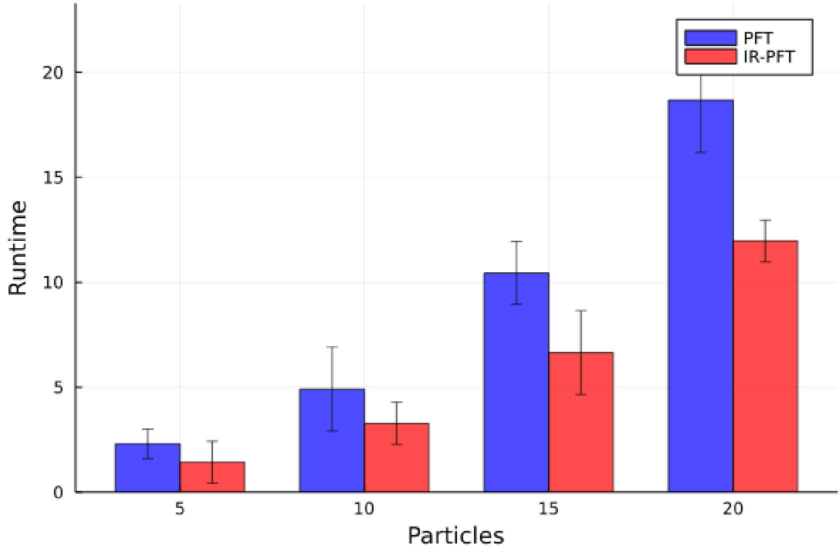
Figure: Illustration of light dark problem

Evaluation

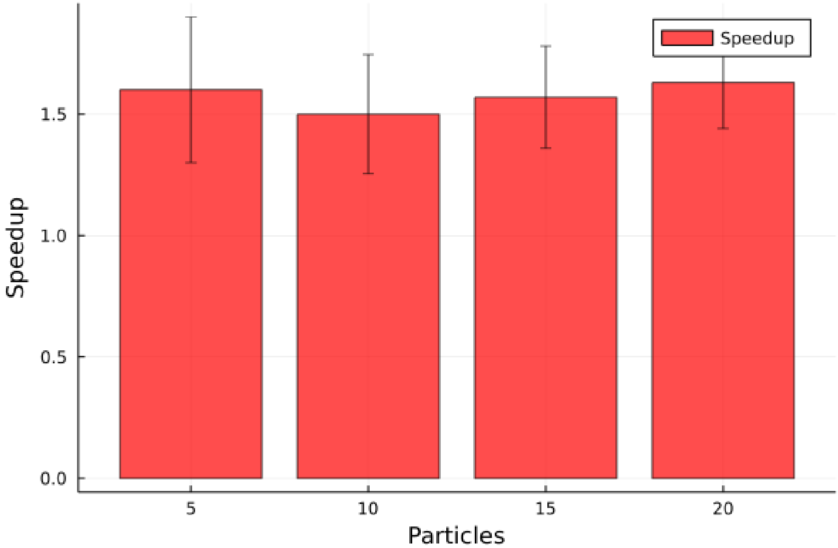
Parameters

- $d = 20$
- 1000 iterations per planning session
- The reward is a linear combination of expected state reward and differential entropy
- 100 random scenarios for each particle number

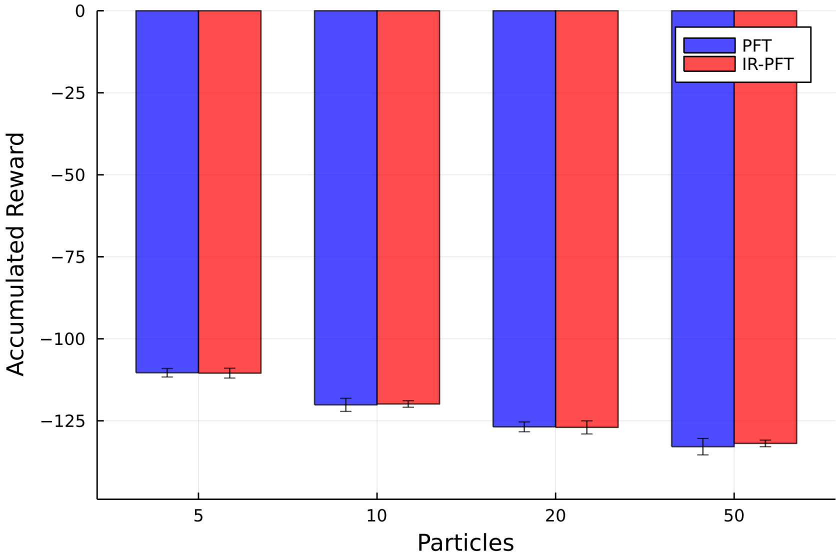
Evaluation - Runtime



Evaluation - Speedup



Evaluation - Accumulated Reward



Outline

ANPL
Autonomous Navigation and Perception Lab

Conclusions

- Theoretical justification for action value function estimation from data without planning
- We developed a novel MCTS-based algorithm that incorporates information reuse
- We presented empirical study that shows runtime performance gain without compramising on the accumulated reward
- Several future research directions

Questions?