Open-loop POMDP Simplification and Safe Skipping of Replanning with Formal Performance Guarantees Supplementary Material

I Proof of Theorem 1

Proof. We prove the upper and lower bounds separately by comparing the policy spaces and information available to each policy.

Upper Bound: $Q^{\pi^*}(b_0, a_0) \leq Q^{\pi^{AFO,\tau*}}(b_0, a_0)$ The Adaptive Fully-Observable (AFO) policy, $\pi^{\text{AFO},\tau}$, operates in an information space that is at least as rich as that of the standard POMDP policy, π^*. At any belief node where the topology indicator $\beta^{\text{AFO},\tau} = 1$, the AFO policy has access to the true state x_t, which is strictly more informative than the belief b_t available to the standard policy. A policy with access to more information can always achieve a value at least as high as a policy with less information, as it can simply choose to ignore the extra information and replicate the less-informed policy's strategy.

Consequently, the set of all AFO policies, $\mathit{\Pi}^{\text{AFO},\tau}$, can achieve any value that the set of standard POMDP policies, $\mathit{\Pi}^{\text{Full}}$, can. Maximizing over these policy spaces, the optimal value for the AFO policy must be greater than or equal to the optimal value for the standard POMDP:

$$Q^{\pi^*}(b_0, a_0) = \max_{\pi \in \mathit{\Pi}^{\text{Full}}} Q^{\pi}(b_0, a_0) \leq \max_{\pi' \in \mathit{\Pi}^{\text{AFO},\tau}} Q^{\pi'}(b_0, a_0) = Q^{\pi^{\text{AFO},\tau*}}(b_0, a_0).$$

Lower Bound: $Q^{\pi^{AOL,\tau*}}(b_0, a_0) \leq Q^{\pi^*}(b_0, a_0)$ The Adaptive Open-Loop (AOL) policy, $\pi^{\text{AOL},\tau}$, is more constrained than the standard POMDP policy. At any belief node where $\beta^{\text{AOL},\tau} = 1$, the AOL policy must select an action without conditioning on the most recent observation z_t. This means the policy space for AOL, $\mathit{\Pi}^{\text{AOL},\tau}$, is a subset of the full POMDP policy space, $\mathit{\Pi}^{\text{Full}}$. Since the optimization for the AOL policy is performed over a smaller set of policies, its optimal value cannot exceed that of the standard POMDP:

$$Q^{\pi^{\text{AOL},\tau*}}(b_0, a_0) = \max_{\pi \in \mathit{\Pi}^{\text{AOL},\tau}} Q^{\pi}(b_0, a_0) \leq \max_{\pi^* \in \mathit{\Pi}^{\text{Full}}} Q^{\pi^*}(b_0, a_0) = Q^{\pi^*}(b_0, a_0).$$

This relationship can be seen from the Bellman equations. The standard POMDP performs a maximization after observing z_t, while the AOL policy must maximize before the expectation over z_t when in an open-loop step. By Jensen's inequality, we have:

$$\begin{aligned} &\mathop{\mathbb{E}}_{x_t|h_t^-} \mathop{\mathbb{E}}_{z_t|x_t} \max_{a_t} \left[r(x_t, a_t) + \mathop{\mathbb{E}}_{x_{t+1}|h_t, a_t} \mathop{\mathbb{E}}_{z_{t+1}|x_{t+1}} V(b_{t+1}) \right] \\ &\geq \max_{a_t} \mathop{\mathbb{E}}_{x_t|h_t^-} \mathop{\mathbb{E}}_{z_t|x_t} \left[r(x_t, a_t) + \mathop{\mathbb{E}}_{x_{t+1}|h_t, a_t} \mathop{\mathbb{E}}_{z_{t+1}|x_{t+1}} V(b_{t+1}) \right]. \end{aligned} \tag{1}$$

The left side corresponds to the standard closed-loop update, and the right side corresponds to the open-loop update, confirming the sub-optimality.

This establishes the claimed upper and lower bounds.

II Proof of Theorem 2

Proof. We now provide the proof for the lower bound on the error of the sparse-sampling estimator for the adaptive open-loop policy, $\Delta\hat{lb}$. The proof proceeds by induction on the depth d of the belief tree, from $d = L$ down to $d = 0$. The value function $V_{\max}$ is defined as $V_{\max} = L \cdot R_{\max}$.

Base Case ($d = L$): At the maximum depth L, the Q-value is simply the immediate expected reward, as there are no future steps. $Q^{\pi^{\mathrm{AOL},\tau*}}(b_L, a_L) = \mathbb{E}_{x_L|b_L}[r(x_L, a_L)]$. The sparse-sampling estimator $\hat{lb}(b_L, a_L)$ for this value is an average over samples from the belief b_L. By Hoeffding's inequality, for any $\lambda' > 0$, the probability of the estimation error exceeding λ' is bounded:

$$\Pr\left[|\hat{lb}(b_L, a_L) - lb(b_L, a_L)| > \lambda'\right] \leq 2\exp\left(-\frac{2N(\lambda')^2}{R_{\max}^2}\right),$$

where N is the number of state samples from b_L.

Inductive Step: Assume that for any depth $d' > d$, any belief $b_{d'}$, and any action $a_{d'}$, the following holds with high probability: $|\hat{lb}(b_{d'}, a_{d'}, \tau) - lb(b_{d'}, a_{d'}, \tau)| \leq \frac{(L-d')(L-d'-1)}{2}\lambda$.

Now, consider depth d. The lower bound Q-value, $lb(b_d, a_d, \tau) = Q^{\pi^{\mathrm{AOL},\tau*}}(b_d, a_d)$, is given by the Bellman equation for the adaptive open-loop policy. We analyze the two cases based on the topology indicator $\beta^{\mathrm{AOL},\tau}$.

Case 1: Closed-loop step ($\beta^{AOL,\tau}(h_d) = 0$)

$$lb(b_d, a_d, \tau) = \mathbb{E}_{x_d|b_d}[r(x_d, a_d)] + \mathbb{E}_{z_{d+1}|b_d,a_d}\left[\max_{a_{d+1}} lb(b_{d+1}, a_{d+1}, \tau)\right]. \quad (2)$$

The estimator $\hat{lb}(b_d, a_d, \tau)$ is computed by sampling N^O observations $z_{d+1}^{(j)}$ and recursively calling the estimator on the resulting beliefs $b_{d+1}^{(j)}$:

$$\hat{lb}(b_d, a_d, \tau) = \mathbb{E}_{x_d|b_d}[r(x_d, a_d)] + \frac{1}{N^O}\sum_{j=1}^{N^O} \max_{a_{d+1}} \hat{lb}(b_{d+1}^{(j)}, a_{d+1}, \tau). \quad (3)$$

Let $Y(z_{d+1}) = \max_{a_{d+1}} lb(b_{d+1}, a_{d+1}, \tau)$ and $\hat{Y}(z_{d+1}) = \max_{a_{d+1}} \hat{lb}(b_{d+1}, a_{d+1}, \tau)$. The error can be bounded by the sum of a sampling error and the propagated error from the next level:

$$\Delta\hat{lb}(b_d, a_d, \tau) \leq \left|\mathbb{E}[\hat{Y}] - \frac{1}{N^O}\sum \hat{Y}_j\right| + \left|\mathbb{E}[Y] - \mathbb{E}[\hat{Y}]\right| \quad (4)$$

$$\leq \left|\mathbb{E}[\hat{Y}] - \frac{1}{N^O}\sum \hat{Y}_j\right| + \mathbb{E}[|Y - \hat{Y}|]. \quad (5)$$

The first term is a sampling error, bounded by λ with high probability using Hoeffding's inequality. The second term is bounded by the inductive hypothesis: $\mathbb{E}[|Y - \hat{Y}|] \leq \frac{(L-d-1)(L-d-2)}{2}\lambda$. Thus, the total error is bounded by $\lambda + \frac{(L-d-1)(L-d-2)}{2}\lambda = \frac{(L-d)(L-d-1)}{2}\lambda$.

Case 2: Open-loop step ($\beta^{AOL,\tau}(h_d) = 1$)

$$lb(b_d, a_d, \tau) = \mathbb{E}_{x_d|b_d}[r(x_d, a_d)] + \max_{a_{d+1}} \mathbb{E}_{x_{d+1}|b_d,a_d}\left[lb(b_{d+1}, a_{d+1}, \tau)\right]. \quad (6)$$

The estimator $\hat{lb}(b_d, a_d, \tau)$ samples N next states $x_{d+1}^{(j)}$:

$$\hat{lb}(b_d, a_d, \tau) = \mathbb{E}_{x_d|b_d}[r(x_d, a_d)] + \max_{a_{d+1}} \frac{1}{N}\sum_{j=1}^{N} \hat{lb}(b_{d+1}^{(j)}, a_{d+1}, \tau). \quad (7)$$

The logic is identical to Case 1, but the expectation is over next states x_{d+1} instead of observations z_{d+1}. The error is bounded similarly.

Since $C = \min\{N, N^O\}$, the error bound is largely dominated by the worst case. To complete the proof, we use a union bound over all possible histories and actions. Applying the union bound over all actions at depth d and all nodes in the subsequent subtree, the probability at depth d for a given action a_d is bounded by $2(|A|C)^{L-d}\exp(-\frac{C\lambda^2}{2V_{\max}^2})$. Taking another union bound for all $|A|$ actions at depth d gives the final probability in the theorem statement.

For the upper bound, we have a similar result as in the lower bound case.

III Bound Analysis of *ub* and *lb*

Complexity Analysis The computational complexity of solving $Q^{\pi^{\mathrm{AOL},\tau*}}$ and $Q^{\pi^{\mathrm{AFO},\tau*}}$ is primarily determined by the topology τ, where a larger number of belief nodes with simplification yields more substantial complexity reduction. For each belief node with simplification (i.e., $\beta^\tau = 1$), the complexity of a single-step Bellman update with state-dependent reward is reduced from $\mathcal{O}(|X||Z||A|)$ to $\mathcal{O}(|X||A|)$, consistent with the cancellation of expectation over observations.

Convergence. When the topology is switched, certain belief nodes are reverted to a closed-loop setting by appropriately introducing observations (as discussed in Section 4.1). In the extreme case, all belief nodes will be switched to a closed-loop setting, which corresponds to a standard POMDP belief tree. Consequently, both the upper and lower bounds converge to the optimal Q-function of the full POMDP.

Monotonicity. We establish monotonicity with respect to transitions toward topologies containing a greater number of closed-loop belief nodes. Let τ denote the original topology and τ' the modified topology. For the upper bound, we have $Q^{\pi^{AFO,\tau,*}}(b_0, a_0) \geq Q^{\pi^{AFO,\tau',*}}(b_0, a_0)$. For the lower bound, $Q^{\pi^{AOL,\tau*}}(b_0, a_0) \leq$

$Q^{\pi^{AOL,\tau'*}}(b_0, a_0)$. These inequalities demonstrate that the bounds monotonically tighten as the topology transitions to include more closed-loop nodes.

The proof of the monotonicity is based on the same method to prove Theorem 1. The key idea is that the AFO policy with more nodes using full observability can achieve at least as high a value as the AFO policy with fewer nodes with full observability, and similarly for the AOL policy.

Parallel Calculation of the Upper and Lower Bounds. The policy $\pi^{AFO,\tau*}$ can be computed in parallel with the policy $\pi^{AOL,\tau*}$, as the upper and lower bounds are independent of each other. This parallel computation can significantly reduce the overall computation time required to obtain both bounds.

IV Cross-topology Transition

When bounds defined in (1) under topology τ overlap, we have to explore an alternative topology τ' to achieve non-overlapping bounds through an iterative process, which continues until we identify the optimal action. We propose an incremental refinement method that ensures monotonic bound tightening and asymptotic convergence to the full POMDP solution. Since τ actually means a pair of τ_U and τ_L, we will transition both topologies simultaneously during the topology adaptation process from τ to τ', i.e., $\tau' = (\tau'_U, \tau'_L)$.

The transition process first selects some belief nodes in belief tree $\mathbb{T}^\tau$ to transition from open-loop step to closed-loop step, creating an updated topology τ' with a new belief tree $\mathbb{T}^\tau$. This transition typically introduces observations at the selected belief nodes and expands the belief tree structure with new nodes.

Then, the indicator function $\beta^{\tau'}$ is updated to: (1) enforce closed-loop planning for the chosen nodes, (2) preserve the original planning mode for the unchanged nodes, and (3) inherit the mode from the nodes in the previous topology for newly generated belief nodes. This approach ensures consistent topology evolution.

Belief nodes caching. To reduce the computational overhead, we cache belief computations during planning. During topology transitions, unchanged nodes are retrieved directly from the cache, while only selected nodes require recomputation.

V Algorithm of AT-POMCP

The proposed anytime solver, AT-POMCP, is given by Algorithm 1.

VI Convergence of AT-POMCP: Proof of Theorem 3

We provide the convergence guarantee of AT-POMCP in the following theorem.

Algorithm 1: Topology-based MCTS-style Anytime Solver

Input: state s, history h, planning horizon L, simulation index i; topology τ, topology adaptation index j, topology progressive adaptation parameter α and k.
Output: return estimate R and update belief tree.
Simulate(s, h, depth, i)
if depth $> L$ **then**
  return 0
end if
if $h \notin T$ **then**
  for $a \in \mathcal{A}$ **do**
    $T(ha) \leftarrow (N_{\text{init}}(ha), V_{\text{init}}(ha), \emptyset)$
  end for
  return Rollout(s, h, depth)
end if
$a \leftarrow \arg\max_b \left[V(hb) + c\sqrt{\frac{\log N(h)}{N(hb)}} \right]$
$(s', o, r) \sim \mathcal{G}(s, a)$
// — Progressive topology adaptation —
if $i \leq k \cdot j^{\alpha}$ **then**
  $\tau \leftarrow \tau$
else
  $j \leftarrow j + 1$
  // — Randomly set some belief nodes' indicator function β^{τ} to 0 —
  $\tau \leftarrow \text{RandomTopoTransition}(\tau)$
end if
Update History h' by (2) or (3) based on topology τ
$R \leftarrow r + \text{Simulate}(s', h', \text{depth} + 1, i)$
$B(h) \leftarrow B(h) \cup \{s\}$
$N(h) \leftarrow N(h) + 1$
$N(ha) \leftarrow N(ha) + 1$
$V(ha) \leftarrow V(ha) + \dfrac{R - V(ha)}{N(ha)}$
return R

Proof sketch. The proof starts from a fixed topology version of AT-POMCP. If the topology is fixed, the convergence of AT-POMCP directly follows from the convergence of POMCP [3]: $\hat{V}^{\tau *}(b_0) \xrightarrow{p} V^{\tau *}(b_0)$. Then, the proof will focus on the progressive adaptation of topology, which will affect the UCB action selection process and its convergence.

Consider the belief node with history $\tilde{h}_t$ [1], where it expands action branches, and leads to child nodes, among which we consider a specific history $\tilde{h}_{t+1}^{-}$. We assume after some iterations, the progressive adaptation of topology will change the topology from τ to τ' and modify the indicator function at this belief node from $\beta^{\tau}(\tilde{h}_{t+1}^{-}) = 1$ to $\beta^{\tau'}(\tilde{h}_{t+1}^{-}) = 0$. This means the belief node $\tilde{h}_{t+1}^{-}$ will expand observation branches after the topology adaptation. This adaptation will affect the UCB action selection process at the belief node $\tilde{h}_t$ and the value estimation

[1] For simplicity, we sometimes refer to belief node with history $\tilde{h}_t$ as belief node $\tilde{h}_t$.

at $\tilde{h}^{-}_{t+1}$. We will analyze the convergence of value estimation at these two belief nodes after the topology adaptation. If the convergence at these two nodes hold, the convergence of other nodes will follow.

1. We examine the average return at the node $\tilde{h}^{-}_{t+1}$. Denote $N(\tilde{h}^{-}_{t+1})$ to be the number of simulations at belief node with history $\tilde{h}^{-}_{t+1}$. Then, the average return at $\tilde{h}^{-}_{t+1}$ under the new topology will become:

$$\bar{G}^{\tau',N(\tilde{h}^{-}_{t+1})}(\tilde{h}^{-}_{t+1}) = \frac{G^{\tau,N_{\beta=1}(\tilde{h}^{-}_{t+1})}(\tilde{h}^{-}_{t+1}) + G^{\tau',N_{\beta=0}(\tilde{h}^{-}_{t+1})}(\tilde{h}^{-}_{t+1})}{N(\tilde{h}^{-}_{t+1})}, \tag{8}$$

where $N_{\beta=1}(\tilde{h}^{-}_{t+1})$ and $N_{\beta=0}(\tilde{h}^{-}_{t+1})$ are the number of simulations at $\tilde{h}^{-}_{t+1}$ before and after the topology adaptation, and $N(\tilde{h}^{-}_{t+1}) = N_{\beta=1}(\tilde{h}^{-}_{t+1}) + N_{\beta=0}(\tilde{h}^{-}_{t+1})$. The return will include some simulations under the old topology. Since the simulation number $N_{\beta=1}(\tilde{h}^{-}_{t+1})$ will not increase anymore and is finite, as we increase the number of simulations, the ratio of $N_{\beta=1}(\tilde{h}^{-}_{t+1})/N(\tilde{h}^{-}_{t+1})$ will converge to 0. Thus, the average return will converge:

$$\bar{G}^{\tau',N(\tilde{h}^{-}_{t+1})}(\tilde{h}^{-}_{t+1}) \to \bar{G}^{\tau',N_{\beta=0}(\tilde{h}^{-}_{t+1})}(\tilde{h}^{-}_{t+1}). \tag{9}$$

This indicates that the estimation at $\tilde{h}^{-}_{t+1}$ will converge to the new topology part.

2. We examine the UCB action selection at belief node $\tilde{h}_t$. At belief node $\tilde{h}_t$, we are interested in the estimated optimal Q-function under the new topology τ'. This part is based on the convergence of UCB derived by [2] and [1].

The analysis aims to show that the expected number of sub-optimal action selections, $\mathbb{E}(N_{\text{Sub}-\text{opt}})$, is bounded. Based on [1]'s proof of Theorem 1, we can split the choosing of the sub-optimal action into three events: **(A)** The optimal branch is not explored enough and has a much lower average return than the theoretical one, **(B)** The average return of the suboptimal branch is much higher than the theoretical one, **(C)** The theoretical value of the optimal branch is not large enough to distinguish the optimal action. (See Equation (7)-(9) in [1]). For event (C), our case follows the same theoretical level analysis as [1].

For our case, we will show that the topology adaptation will not affect event (A) and (B) in the estimation level, by considering the following three cases:

i. If the given history $\tilde{h}^{-}_{t+1}$ is not in the optimal action branch under the both old and new topologies, based on the proof in the previous part, the average return after the topology adaptation will converge to the new topology part as shown in Equation (9). Thus it will eventually converge to the theoretical value under the new topology, not breaking case (B).
ii. If the given history $\tilde{h}^{-}_{t+1}$ is in the optimal action branch under the old and new topologies, based on the proof of converge in the first part, the average return after the topology adaptation will converge to the new topology part, as shown in Equation (9). This will not break case (A).

iii. If the given history $\tilde{h}^-_{t+1}$ is in the optimal action branch under the new topology but not under the old topology, this will affect the UCB action selection at the time of topology adaptation. But if we keep increasing the number of simulations, due to the exploration bonus term, the UCB action selection will eventually explore the branch with $\tilde{h}^-_{t+1}$ under the new topology enough times. As long as the branch with $\tilde{h}^-_{t+1}$ under the new topology is explored enough times, based on the proof of converge in the first part, the average return after the topology adaptation will converge to the new topology part (see Equation (9)), which will become the optimal branch. So, this case will not break case (A).

iv. An additional case could be as follows: If the given history $\tilde{h}^-_{t+1}$ is in the optimal action branch under the old topology but not under the new topology. However, this will not happen because the topology transition will monotonically tighten the bounds toward the value function of the original POMDP, as shown in Appendix III.

So far, we have shown that all the cases will not affect the boundedness of $\mathbb{E}(N_{\text{Sub}-\text{opt}})$, then the rest of the proof can directly follow [2]. This means that the topology adaptation will not affect the convergence of AT-POMCP if adapting from topology τ to topology τ', as: $\hat{V}^{\tau'*}(b_0) \xrightarrow{p} V^{\tau'*}(b_0)$.

Using the progressive topology adaptation, eventually the topology of the belief tree will converge to the original POMDP topology τ_0, without any simplification. Repeating the process in the above proof, we can show that $\hat{V}^{\tau_0*}(b_0) \xrightarrow{p} V^{\tau_0*}(b_0) = V^*(b_0)$. Thus, the value function estimated by AT-POMCP, $\hat{V}^{\text{AT}*}(b_0)$ converges in probability to the optimal value function $V^*(b_0)$. This finishes the proof.

□

VII Algorithm of Safe Skipping Replanning

The algorithm of safe skipping replanning is given by Algorithm 2.

VIII Proof of Theorem 4

Proof. We will prove the lower bound first. The proof for the upper bound follows a similar approach. The core of the proof is to relate the posterior belief b_k at a future time step k to the belief propagated open-loop from the initial belief b_0.

Let $b_k(x_k) = P(x_k|h_k)$ be the posterior belief at time k, where the history is $h_k = \{a_{0:k-1}, z_{1:k}\}$. Let $b^-_k(x_k) = P(x_k|h^-_k)$ be the belief propagated up to time k before incorporating the observation z_k, where $h^-_k = \{a_{0:k-1}, z_{1:k-1}\}$. The relationship is given by Bayes' rule:

$$b_k(x_k) = \frac{P(z_k|x_k)b^-_k(x_k)}{P(z_k|h^-_k)}. \tag{10}$$

Algorithm 2: Safe Skipping Replanning in POMDPs

Input: Initial belief b_0, initial topology τ_0
while not in a terminal state **do**
  // — Planning Phase to find optimal action a_0^* —
  $\tau \leftarrow \tau_0$
  repeat
    For each action $a \in \mathcal{A}$, compute bounds $ub(\tau, b, a)$ and $lb(\tau, b, a)$.
    if an optimal action a_0^* is identified **then**
      Get corresponding policy $\pi^{AOL,\tau*}$
      break
    else
      Refine topology $\tau \leftarrow \tau'$ to tighten bounds.
    end if
  until optimal action a_0^* is found
  // — Execution Phase —
  for $k = 1, 2, \ldots, L$ **do**
    if the k-th step in $\pi^{AOL,\tau*}$ is open-loop **then**
      Let a_k be the k-th action in $\pi^{AOL,\tau*}$
      if $SRG(\tau, b_0, a_{0:k-1}^*, a_k, \bar{\mathcal{Z}}_{1:k}) =$ true **then**
        continue. //(Skip replanning)
      end if
    end if
    break. //(Trigger replanning)
  end for
end while

By recursively applying this, we can relate $b_k(x_k)$ to the initial belief $b_0(x_0)$ and the open-loop propagated belief $\phi^p(b_0, a_{0:k-1})(x_k)$:

$$b_k(x_k) = \frac{\prod_{j=1}^{k} P(z_j|x_j)}{\prod_{j=1}^{k} P(z_j|h_j^-)} \phi^p(b_0, a_{0:k-1})(x_k). \tag{11}$$

Let $c_j = \frac{\min_{z_j \in \mathcal{Z}, x_j \in \mathcal{X}} P(z_j|x_j)}{\max_{z_j \in \mathcal{Z}, x_j \in \mathcal{X}} P(z_j|x_j)}$, for $P(z_j|x_j) > 0$. We can bound the ratio of the belief distributions:

$$\prod_{j=1}^{k} c_j \cdot \phi^p(b_0, a_{0:k-1})(x_k) \leq b_k(x_k) \leq \frac{1}{\prod_{j=1}^{k} c_j} \cdot \phi^p(b_0, a_{0:k-1})(x_k). \tag{12}$$

This gives us

$$C_k(\mathcal{Z}, \mathcal{X}) \phi^p(b_0, a_{0:k-1}) \leq b_k \leq \frac{1}{C_k(\mathcal{Z}, \mathcal{X})} \phi^p(b_0, a_{0:k-1}). \tag{13}$$

Now, let's analyze the Q-value. Assume state-dependent rewards and positive Q-value. From Theorem 1, the original Q value is lower-bounded by the AOL Q-value, and we can have the following relationship by directly applying (13) :

$$Q^{\pi^*}(b_k, a_k) \geq Q^{\pi^{AOL,\tau,*}}(b_k, a_k) \geq C_k(\mathcal{Z}, \mathcal{X}) Q^{\pi^{\mathrm{AOL},\tau*}}(\phi^p(b_0, a_{0:k-1}), a_k). \tag{14}$$

The term $Q^{\pi^{\text{AOL},\tau*}}(\phi^p(b_0, a_{0:k-1}), a_k)$ is the expected value of a plan of length L starting from belief $\phi^p(b_0, a_{0:k-1})$. This can be rewritten in terms of a longer plan starting from b_0:

$$Q^{\pi^{\text{AOL},\tau*}}(\phi^p(b_0, a_{0:k-1}), a_k) = \tilde{Q}^{\pi^{\text{AOL},\tau*}}_{L+k}(b_0, a_{0:k-1}, a_k) - \sum_{i=0}^{k-1} \mathbb{E}[r(b_i, a_i)], \quad (15)$$

where $\tilde{Q}^{\pi^{\text{AOL},\tau*}}_{L+k}(b_0, a_{0:k-1}, a_k)$ is the value of a plan of horizon $L+k$ from b_0 where the first k actions are fixed to $a_{0:k-1}$. Combining these gives the lower bound:

$$\text{lb}^k(\tau, b_0, a_{0:k}) = C_k(\mathcal{Z}, \mathcal{X}) \left(\tilde{Q}^{\pi^{\text{AOL},\tau*}}_{L+k}(b_0, a_{0:k-1}, a_k) - \sum_{i=0}^{k-1} \mathbb{E}[r(b_i, a_i)] \right). \quad (16)$$

The proof for the upper bound, $\text{ub}^k(\tau, b_0, a_{0:k})$, follows symmetrically using the upper bound on the belief ratio and the upper bound from Theorem 1.

Finally, the state space $\mathcal{X}$ can be tighten to be the reachable space $\mathcal{X}^R$. This completes the proof.

IX Proof of Theorem 5

Proof. The proof follows the same structure as the proof of Theorem 4. The key difference lies in the bounding of the posterior belief b_k.

In the proof of Theorem 4, the belief ratio is bounded over the entire observation space $\mathcal{Z}$. For this theorem, we are given that the future observations z_k will fall within the allowed observation sets $\bar{\mathcal{Z}}_k$. Therefore, the belief ratio bound can be tightened by replacing the full observation space $\mathcal{Z}$ with the subset $\bar{\mathcal{Z}}_k$ when defining the constant factor. The ratio of belief distributions is now bounded by:

$$C_k(\bar{\mathcal{Z}}_k, \mathcal{X}^R)\phi^p(b_0, a_{0:k-1}) \leq b_k \leq \frac{1}{C_k(\bar{\mathcal{Z}}_k, \mathcal{X}^R)}\phi^p(b_0, a_{0:k-1}), \quad (17)$$

where $C_k(\bar{\mathcal{Z}}_k, \mathcal{X}^R)$ is defined as in Theorem 5 but with the 'min' and 'max' operations taken over the observation subset $\bar{\mathcal{Z}}_k$.

The remainder of the proof follows directly from the steps in the proof of Theorem 4, by substituting $C_k(\mathcal{Z}, \mathcal{X}^R)$ with the tighter factor $C_k(\bar{\mathcal{Z}}_k, \mathcal{X}^R)$. This yields the desired bounds $\bar{lb}^k$ and $\bar{ub}^k$.

X Experimental Details

Beacon Navigation Problem. The beacon navigation problem is a POMDP where an agent navigates a grid world to reach a target beacon while avoiding obstacles. The agent can move in four directions (up, down, left, right) and can observe the distance to the beacon. The goal is to reach the beacon while maximizing

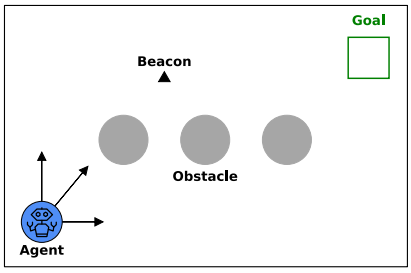

Fig. 1: Beacon navigation problem.

the cumulative reward. The observation is based on the beacons, where the observation noise is defined as the distance to the beacon plus some noise. Figure 1 illustrates the problem setup.

The transition model is defined as follows:

$$P(x'|x,a) = \begin{cases} p^T_{int} & \text{if } x' = x + a \text{ and } x' \text{ is within bounds,} \\ p^T_{adj} & \text{if } |x' - x - a| = 1, \\ p^T_{stay} & \text{if } x' = x, \\ 0 & \text{otherwise.} \end{cases} \tag{18}$$

Here, p^T_{int} is the probability of moving to the intended cell, p^T_{adj} is the probability of moving to an adjacent cell, and p^T_{stay} is the probability of staying in the same cell. And $p^T_{int} + p^T_{adj} + p^T_{stay} = 1$.

The observation model is defined as:

$$P(z|x) = \begin{cases} 1 - p^O_{error} & \text{if } z = x \text{ and within beacon range,} \\ \frac{p^O_{error}}{4} & \text{if } |z - x| = 1 \text{ and within beacon range,} \\ 0 & \text{otherwise.} \end{cases} \tag{19}$$

Here, the factor p^O_{error} represents the observation noise, which is based on the distance to the nearest beacon: $p^O_{error} = \min\{0.9, 1 - d_{beacon} * 0.15\}$, where d_{beacon} is the distance to the nearest beacon.

The reward function is defined as:

$$r(x,a) = 1_{x \in X^{goal}} \cdot r^{goal} + 1_{x \in X^{obstacle}} \cdot r^{obstacle} + r^{step} + r^{dis}, \tag{20}$$

where $1_{x \in X^{goal}}$ is an indicator function that returns 1 if the agent is in the goal state set (X^{goal}), r^{goal} is the reward for reaching the goal, $1_{x \in X^{obstacle}}$ is an indicator function that returns 1 if the agent is in an obstacle state set ($X^{obstacle}$), $r^{obstacle}$ is the penalty for hitting an obstacle, r^{step} is a small negative reward for each step taken, and r^{dis} is a reward based on the distance to the goal (d_{goal}) as: $r^{dis} = \frac{15}{1 + d_{goal}}$.

In the experiments, we set the parameters as follows: $p^T_{int} = 0.5$, $p^T_{adj} = 0.2$, $p^T_{stay} = 0.3$, $p^O_{error} = 0.1$, $r^{goal} = 200$, $r^{obstacle} = -30$, and $r^{step} = -0.5$. The

grid size is 20×20 with one beacon at $(3, 3)$ and obstacles at $(2, 3), (2, 4), (9, 3)$. The goal is at $(7, 5)$. The agent starts at $(1, 3)$.

The experiments are conducted on Ubuntu 22.04 with an Intel i9-9820X CPU and 64GB RAM. The code is implemented in Julia.

XI Open-loop Simplification Experiment

This section presents detailed results from the open-loop simplification experiment, including cumulative reward analysis and discussion of result standard deviation.

XI.A AT-SparsePFT Experiment Figure

This section provides additional visualization of the cumulative rewards at each step for both the baseline SparsePFT method and our proposed AT-SparsePFT method, as shown in Figure 2.

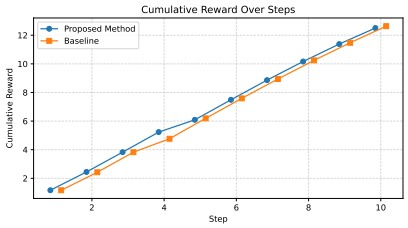

Fig. 2: Cumulative rewards at each step for the baseline, SparsePFT, and our proposed method, AT-SparsePFT. Results are averaged over 100 simulations.

Fig. 2 illustrates the cumulative rewards at each time-step for both the baseline sparsePFT and our proposed AT-SparsePFT. Results represent averages across 100 independent simulations. Our proposed method achieves performance nearly identical to the baseline, but shows a significant speedup while adapting topology online, empirically validating our theoretical performance guarantees.

XI.B Results with Standard Deviation Analysis

Table 1 presents the cumulative rewards and runtime performance for the baseline sparse sampling (SS) method and our proposed approach (including mean and std of the results). Our method demonstrates a speedup ratio of 16.7× compared to the baseline SS method while maintaining comparable cumulative rewards.

Both methods exhibit similar levels of standard deviation in cumulative rewards, with our proposed method demonstrating marginally lower variance.

Given the stochastic nature of the problem, which incorporates noise in both transition and observation models, elevated standard deviation values are expected. Notably, our method maintains performance consistency comparable to the baseline, validating the proposed performance guarantees. The observed standard deviation in both methods is primarily attributed to a single collision event among 100 simulation runs, arising from the inherent stochasticity in the motion and observation models.

Method	Returns	Runtime (s)	Speedup Ratio
Baseline (SparsePFT)	12.64 ± 6.02	345.9 ± 6.4	$1.0\times$
AT-SparsePFT	12.50 ± 5.42	$\mathbf{20.66 \pm 5.0}$	$\mathbf{16.7}\times$

Table 1: Cumulative rewards and runtime of the baseline SparsePFT method and our proposed method. The speedup ratio represents the ratio of baseline runtime to proposed method runtime. Results report mean values and standard deviations (std) computed over 100 independent 10-step simulation trials.

XI.C AT-POMCP Experiments

Figure 3 provides a visual summary of the experiment results for the baseline POMCP and our proposed AT-POMCP method.

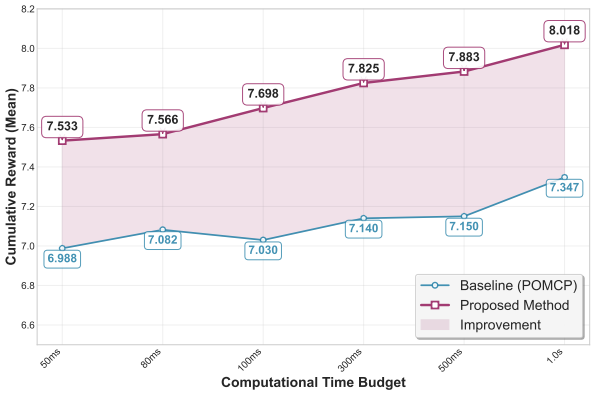

Fig. 3: cumulative reward comparison between the baseline POMCP and the proposed method. The plot shows mean cumulative rewards for budgets ranging from 50 ms to 1 s, highlighting consistent performance gains for the proposed method.

XII Replanning Skipping Experiments

This section presents a runtime analysis incorporating realistic execution and observation time.

XII.A Runtime analysis for skipping replanning check

Method	Planning Time (s)	Replanning Time (s)	Execution Time (s)	Total Time (s)
Skip Replanning (Ours)	220.64	80.70	150	**230.70**
Baseline (Always Replan)	90.58	90.58	150	240.58

Table 2: Runtime comparison between our replanning skip method and baseline approach over 100 trials. Execution and observation time per time-step: 1.5s.

We evaluate a scenario where action execution and observation acquisition require 1.5s per time-step. Table 2 presents the runtime analysis for our replanning skip approach. Our method achieves a total runtime of 230.70s, comprising 220.64s for planning (including 80.70s for replanning phases), compared to the baseline's 240.58s total runtime.

Our approach achieves a 24% replanning skip rate, indicating that replanning can be safely omitted in approximately one-quarter of all decision steps. While this theoretically corresponds to a 24% reduction in replanning overhead, the current implementation realizes a smaller improvement due to unoptimized code. Future optimization efforts are expected to enhance runtime performance.

References

1. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time analysis of the multiarmed bandit problem. Machine learning **47**(2), 235–256 (2002)
2. Kocsis, L., Szepesvári, C.: Bandit based monte-carlo planning. In: European conference on machine learning. pp. 282–293. Springer (2006)
3. Silver, D., Veness, J.: Monte-carlo planning in large pomdps. In: Advances in Neural Information Processing Systems (NeurIPS). pp. 2164–2172 (2010)