A OPTIMIZING VOI VIA SUBMODULAR SURROGATES

We have discussed three orthogonal aspects for the optimal VoI problem, namely, (1) the *sampling scheme* for hypothesis enumeration, (2) the *online learning* framework, and (3) the *choice of algorithms* for optimizing VoI. In this paper we focus on the first two aspects, and propose a general framework integrating the three components. Inevitably, the discussion of our algorithmic framework is grounded on existing submodular surrogate-based approaches for the VoI problem. We give more details of this class of algorithms in this section.

A.1 SUBMODULARITY AND ITS IMPLICATIONS.

The EC^2 objective function introduced in §2 is *adaptive submodular*, and *strongly adaptive monotone*. Formally, let $\mathbf{x}_{\mathcal{A}}$ and $\mathbf{x}_{\mathcal{B}}$ be two observation vectors. We call $\mathbf{x}_{\mathcal{A}}$ a *subrealization* of $\mathbf{x}_{\mathcal{B}}$, denoted as $\mathbf{x}_{\mathcal{A}} \preceq \mathbf{x}_{\mathcal{B}}$, if the index set $\mathcal{A} \subseteq \mathcal{B}$ and $\mathbb{P}\left[\mathbf{x}_{\mathcal{B}} \mid \mathbf{x}_{\mathcal{A}}\right] > 0$. A function $f : 2^{\mathcal{T}\times\{0,1\}} \rightarrow \mathbb{R}$ is called *adaptive submodular* w.r.t. a distribution $\mathbb{P}$, if for any $\mathbf{x}_{\mathcal{A}} \preceq \mathbf{x}_{\mathcal{B}}$ and any test t it holds that $\Delta(t \mid \mathbf{x}_{\mathcal{A}}) \geq \Delta(t \mid \mathbf{x}_{\mathcal{B}})$, where $\Delta(t \mid \mathbf{x}_{\mathcal{A}}) := \mathbb{E}_{x_t}\left[f(\mathbf{x}_{\mathcal{A}\cup\{t\}}) - f(\mathbf{x}_{\mathcal{A}}) \mid \mathbf{x}_{\mathcal{A}}\right]$ (i.e., "adding information earlier helps more"). Further, function f is called *strongly adaptively monotone* w.r.t. $\mathbb{P}$, if for all $\mathcal{A}$, $t \notin \mathcal{A}$, and $x_t \in \{0,1\}$, it holds that $f(\mathbf{x}_{\mathcal{A}}) \leq f(\mathbf{x}_{\mathcal{A}\cup\{t\}})$ (i.e., "adding information never hurts"). For adaptive optimization problems satisfying adaptive submodularity and strongly adaptive monotonicity, the policy that greedily, upon having observed $\mathbf{x}_{\mathcal{A}}$, selects the test $t^* \in \arg\max_t \Delta(t \mid \mathbf{x}_{\mathcal{A}})/c(t)$, is guaranteed to attain near-minimal cost [13].

A.2 GENERAL APPROACHES BASED ON SUBMODULAR SURROGATES.

It is noteworthy to mention that our results are not restricted to EC^2, and can be readily generalized to settings where regions are *overlapped*. In such cases, we can use the DiRECt algorithm [8], and prove something similar with Theorem 1: in the upperbound, we get $(r \cdot \log(1/\tilde{p}_{\min}) + 1)\,\mathrm{cost}_{wc}(\mathsf{OPT})$ (for the worst-case cost) and $(r \cdot \log(1/\tilde{p}_{\min}) + 1)\,\mathrm{cost}_{av}(\mathsf{OPT}) + \eta c(\mathcal{T})$ (for the average-case / expected cost), where r measures the amount of "overlap". The analysis follows closely from the proof of Theorem 1 in §C. More generally, Theorem 1 (with modified multiplicative constant) also applies to greedy algorithms whose objective function (1) is adaptive submodular, and (2) rely on a finite set of hypotheses. Other examples satisfying these conditions include GBS [12] and HEC [16].

Furthermore, since our framework is orthogonal to the choice of the submodular surrogate-based algorithms, we can also easily extend our analysis to handle the (more) practical setting where test outcomes are *noisy*. In such settings, one can no longer "cut-away" edges as suggested by the EC^2 algorithm, since with noisy observations one cannot "eliminate" any of the hypotheses (i.e., setting their probability mass to 0). In practice, after observing the outcome of a test, we can perform Bayesian updates on the posterior over H instead of eliminating those hypotheses that are "inconsistent" with the observation. Analogous to the analysis of Theorem 1, we can establish a bound on the *worst-case* cost of such greedy policy, based on the recent results of [9]. Since the theoretical question of handling noisy tests is beyond the scope of this paper, we omit the proof details for this setting.

B IMPLEMENTATION DETAILS OF ALGORITHM 1

In the main paper, we have stated the pseudo code of our dynamic hypothesis enumeration algorithms. Due to space limit we only provide a concise description of the main idea behind the framework. In this section, we elaborate Algorithm 1 by providing more intuitions and implementation details, as well as additional clarifications for better understanding of the algorithm.

The DAG. We use a DAG represents the hypotheses enumeration process, and is *not* a data structure which we actually adopted in Algorithm 1. Rather, algorithmically we are maintaining a "candidate frontier" F_y for each hidden state y, which corresponds to the set of "leaf" hypotheses (i.e., nodes of the DAG which have no outgoing edges), as a seed set to generate more hypotheses. Algorithm 1 enumerates hypotheses in decreasing order of probabilities $\mathbb{P}\left[h \mid y\right]$. The directed edges in the DAG indicate the relations between the conditional probabilities: if there is a directed edge from node h_1 to h_2 in the DAG, it indicates that $\mathbb{P}\left[h_1 \mid y\right] \geq \mathbb{P}\left[h_2 \mid y\right]$. Other than this, we do not use the edge for any other purposes.

Implementation details. In practice, the underlying distributions are often highly concentrated, such that a few number of hypotheses cover a significant part of the total mass. On the other hand, there are many configurations with very small (but non-null) probabilities. Algorithm 1 exploits such structural assumption, and generates the most likely hypotheses in the following four steps:

- *Step* 1 (line 2):
  Test definitions are possibly switched, in a way that $\mathbb{P}\left[X_i = 1 \mid y\right] \geq 0.5 \quad \forall i$ (i.e., when $\mathbb{P}\left[X_i = 1 \mid y\right] < 0.5$, we consider the complementary event $\bar{X}_i$ as the new test outcome so that $\mathbb{P}\left[\bar{X}_i = 1 \mid y\right] = 1 - \mathbb{P}\left[X_i = 1 \mid y\right] \geq 0.5$); test indices are re-ranked in decreasing order of $\mathbb{P}\left[X_i = 1 \mid y\right]$;
- *Step* 2 (line 3, 4):
  For $i = 1, \ldots, n$, compute $p_i \triangleq \log(\mathbb{P}\left[X_i = 1 \mid y\right])$, and $q_i \triangleq \log(\mathbb{P}\left[X_i = 0 \mid y\right])$;
- *Step* 3 (line 5, 6):
  If F_y is empty, initialize F_y with the configuration $h_1 = [1 \ldots 1]$ with log-weight $\lambda_y(h_1) = \sum_i p_i$; set $L_y^* = \emptyset$.
- *Step* 4 (line 7-10): while $\sum_{h \in L_y^*} \exp(\lambda_y(h)) < (1 - \eta)$
  - *Step* 4a: Choose the element h^* from F_y such that $\lambda_y(h^*)$ is maximum;
  - *Step* 4b: Remove h^* from F_y and push it into L_y^*;
  - *Step* 4c (line 9): Generate (at most) 2 children from h^* and add them to F_y if they were not already present in F_y.

In the main paper, we have given detailed description of how to generate the two children configurations (h_{c_1} and h_{c_2}) in Step 4c. We provide some additional insight to facilitate better understanding of the procedure:

- *Child* 1: Once we have re-ranked the tests in decreasing order of $\mathbb{P}\left[X = 1 \mid y\right]$ in Step 1, the last test in the ordered list will have the smallest probability (conditioning on y) of being realized to its more likely outcome, and hence is the most uncertain one. If follows that if we flip the outcome of such test, we will generate a new hypothesis h with the highest $\mathbb{P}\left[h \mid y\right]$ among the unseen hypotheses. The first child is generated exactly in this way: if the last (right-most) bit of h^* is 1, we then create h_{c_1} by switching the last bit to 0. For instance, the child hypothesis h_{c_1} of $h^* = [0, 1, 1, 0, 1]$ is $[0, 1, 1, 0, 0]$. Its log-probability is obtained by $\lambda_y(h_{c_1}) = \lambda_y(h^*) + q_n - p_n$.
- *Child* 2: Besides flipping the last bit, the next most-likely hypothesis can also be the one with two bit edits of an existing hypothesis: Find the right-most "$[1, 0]$" pair in h^* (if there exists any; otherwise we do nothing), and the create h_{c_2} by switching "$[1, 0]$" into "$[0, 1]$". For instance, the child hypothesis h_{c_2} of $h^* = [0, 1, 1, 0, 1]$ is $[0, 1, 0, 1, 1]$. Its associated log-probability is computed by $\lambda_y(h_{c_2}) = \lambda_y(h^*) + q_i - p_i + p_{i+1} - q_{i+1}$, where i is the bit index of the "1" in the right-most "$[1, 0]$" pair.

C PROOFS OF THE MAIN THEOREMS

C.1 PROOF OF THEOREM 1

In this section, we provide proofs for the upper bounds on the cost of Algorithm 2. In the analysis, we assume that we only sample the hypotheses *once* in the beginning of each experiment (i.e., we don't resample after each iteration).

Proof. The main idea of the proof is illustrated in Fig. 6.

Bound on the expected cost We first prove the upper bound on the expected cost of the algorithm. We use p to denote the true distribution over the hypotheses $h \in \mathcal{H}$, and $\tilde{p}$ be the sampled distribution. That is, $p(h) = \mathbb{P}\left[h\right]$, and

$$\tilde{p}(h) = \begin{cases} \mathbb{P}\left[h\right] / (1 - \eta), & \text{for } h \in \tilde{\mathcal{H}}; \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

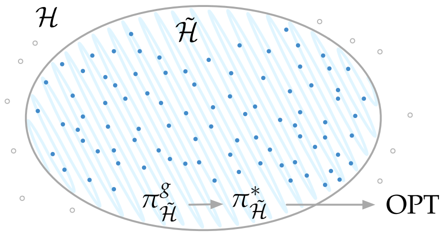

Figure 6: Depicting the main idea behind the proof. We introduce $\pi^*_{\tilde{\mathcal{H}}}$ (the optimal policy on the sampled distribution) as an auxiliary policy to connect $\pi^g_{\tilde{\mathcal{H}}}$ with OPT. If the realized hypothesis $h^* \in \tilde{\mathcal{H}}$, then $\pi^g_{\tilde{\mathcal{H}}}$ efficiently identifies the decision. Otherwise, (with probability at most η) $\pi^g_{\tilde{\mathcal{H}}}$ randomly chooses tests, and the cost can be at most $c(\mathcal{T})$.

For any policy π, let $\mathrm{cost}_{\tilde{p}}(\pi) \triangleq \mathbb{E}_{h\sim\tilde{p}(h)}[c(\mathcal{S}(\pi, h))]$ denote the expected cost of π w.r.t. $\tilde{p}$. Then, the expected cost of π w.r.t. the true distribution p satisfies

$$\begin{aligned}
\mathrm{cost}(\pi) &= \sum_{h\in\mathcal{H}} p(h)c(\mathcal{S}(\pi,h)) \\
&= \sum_{h\in\tilde{\mathcal{H}}} p(h)c(\mathcal{S}(\pi,h)) + \sum_{h\in\mathcal{H}\setminus\tilde{\mathcal{H}}} p(h)c(\mathcal{S}(\pi,h)) \\
&\overset{\text{Eq. (2)}}{=} (1-\eta)\sum_{h\in\tilde{\mathcal{H}}} \tilde{p}(h)c(\mathcal{S}(\pi,h)) + \sum_{h\in\mathcal{H}\setminus\tilde{\mathcal{H}}} p(h)c(\mathcal{S}(\pi,h)) \\
&= (1-\eta)\,\mathrm{cost}_{\tilde{p}}(\pi) + \sum_{h\in\mathcal{H}\setminus\tilde{\mathcal{H}}} p(h)\underbrace{c(\mathcal{S}(\pi,h))}_{\le c(\mathcal{T})} && (3)\\
&\le (1-\eta)\,\mathrm{cost}_{\tilde{p}}(\pi) + \eta\cdot c(\mathcal{T}). && (4)
\end{aligned}$$

The second term on the RHS of Eq. (3) is non-negative, which gives

$$\begin{aligned}
(1-\eta)\,\mathrm{cost}_{\tilde{p}}(\pi) &= \mathrm{cost}(\pi) - \sum_{h\in\mathcal{H}\setminus\tilde{\mathcal{H}}} p(h)c(\mathcal{S}(\pi,h)) \\
&\le \mathrm{cost}(\pi) && (5)
\end{aligned}$$

Let $\pi^*_{\tilde{p}}$ be the optimal policy w.r.t. the sampled distribution $\tilde{p}$. By Theorem 3 of [13] we get

$$\mathrm{cost}_{\tilde{p}}\left(\pi^g_{\tilde{\mathcal{H}}}\right) \le (2\ln(1/\tilde{p}_{\min})+1)\,\mathrm{cost}_{\tilde{p}}\left(\pi^*_{\tilde{\mathcal{H}}}\right). \quad (6)$$

Therefore,

$$\begin{aligned}
\mathrm{cost}(\pi^g_{\tilde{\mathcal{H}}}) &\overset{\text{Eq. (4)}}{\le} (1-\eta)\,\mathrm{cost}_{\tilde{p}}(\pi^g_{\tilde{\mathcal{H}}}) + \eta\cdot c(\mathcal{T}) \\
&\overset{\text{Eq. (6)}}{\le} (1-\eta)\,(2\ln(1/\tilde{p}_{\min})+1)\,\mathrm{cost}_{\tilde{p}}\left(\pi^*_{\tilde{\mathcal{H}}}\right) \\
&\quad + \eta\cdot c(\mathcal{T}).
\end{aligned}$$

By definition we know $\mathrm{cost}_{\tilde{p}}\left(\pi^*_{\tilde{\mathcal{H}}}\right) \le \mathrm{cost}_{\tilde{p}}(\mathsf{OPT})$. Hence

$$\begin{aligned}
\mathrm{cost}(\pi^g_{\tilde{\mathcal{H}}}) &\le (1-\eta)\,(2\ln(1/\tilde{p}_{\min})+1)\,\mathrm{cost}_{\tilde{p}}(\mathsf{OPT}) + \eta\cdot c(\mathcal{T}) \\
&\overset{\text{Eq. (5)}}{\le} (2\ln(1/\tilde{p}_{\min})+1)\,\mathrm{cost}(\mathsf{OPT}) + \eta\cdot c(\mathcal{T}),
\end{aligned}$$

which completes the first part of the proof.

Bound on the worst-case cost. Next, we provide the proof for bound on the worst-case cost. Analogous to the previous analysis, we consider two possible scenarios: (i) the realized hypotheses (i.e., the full realization vector) $h^* \in \tilde{\mathcal{H}}$; and (ii) $h^* \notin \tilde{\mathcal{H}}$.

For any policy π, the worst-case cost of π satisfies

$$\begin{aligned}\mathrm{cost}_{wc}(\pi) &= \max_{h \in \mathcal{H}} c(\mathcal{S}(\pi, h)) \\ &= \max\{\max_{h \in \tilde{\mathcal{H}}} c(\mathcal{S}(\pi, h)),\ \max_{h \in \mathcal{H} \setminus \tilde{\mathcal{H}}} c(\mathcal{S}(\pi, h))\}.\end{aligned}$$

Since policy $\pi^g_{\tilde{\mathcal{H}}}$ terminates if there is no edge left on $\tilde{\mathcal{H}}$, then $\max_{h \in \mathcal{H} \setminus \tilde{\mathcal{H}}} c(\mathcal{S}(\pi, h)) \le \max_{h \in \tilde{\mathcal{H}}} c(\mathcal{S}(\pi, h)$. Therefore,

$$\begin{aligned}\mathrm{cost}_{wc}(\pi^g_{\tilde{\mathcal{H}}}) &= \max_{h \in \tilde{\mathcal{H}}} c\left(\mathcal{S}\left(\pi^g_{\tilde{\mathcal{H}}}, h\right)\right) \\ &\overset{(a)}{\le} \left(2 \ln\left(1/\tilde{p}_{\min}\right) + 1\right) \max_{h \in \tilde{\mathcal{H}}} c\left(\mathcal{S}\left(\pi^*_{\tilde{\mathcal{H}}}, h\right)\right) \\ &\le \left(2 \ln\left(1/\tilde{p}_{\min}\right) + 1\right) \max_{h \in H} c\left(\mathcal{S}\left(\mathsf{OPT}, h\right)\right).\end{aligned}$$

Step (a) in the above equation follows from Theorem A.12 of [12].

Therefore, when $\pi^g_{\tilde{\mathcal{H}}}$ terminates, with probability at least $1 - \eta$, it succeeds to output the correct decision with cost $\left(2 \ln\left(1/\tilde{p}_{\min}\right) + 1\right) \mathrm{cost}_{wc}(\mathsf{OPT})$. □

C.2 PROOF OF THEOREM 2

In this section, we prove the bound on the expected regret of our online learning algorithm.

Proof of Theorem 2. One way to model the non-myopic value of information problem is to view it as a (finite horizon) Partially Observable Markov Decision Process (POMDP), where each (belief-) state represents the selected tests and observed outcome of each test. Formally, the POMDP can be written as

$$M \triangleq \left(\mathcal{B}, \mathcal{T}, R^M, P^M, \tau, \rho\right). \tag{7}$$

Here, $\mathcal{B}$ is the set of belief states, $\mathcal{T}$ is the set of actions (i.e., tests), $R^M_t(b)$ is the (expected) reward associated with action t while in belief state b, $P^M_t(b' \mid b)$ denotes the probabiliy of transitioning to state b' if action t is selectedwhile in state b, τ is the time horizon for each session, and ρ is the initial belief state distribution.

In our problem, the *transition probabilities* P^M can be fully specified by the conditional probabilities of the test outcomes given the hidden state $\mathbb{P}[x_t \mid y]$; the *prior* distribution ρ on belief states can be specified by the prior distribution on the hypotheses $\mathbb{P}[y]$, and $\mathbb{P}[x_t \mid y]$. The *reward* R^M for running a policy π on M is the utility achieved upon termination of the policy. More specifically, we can interpret the reward function R^M as follows: we get reward 0 as the policy keeps selecting new tests, but get (expected) reward $\mathrm{VoI}(\mathcal{S}(\pi, h)) \triangleq \max_{d \in \mathcal{D}} \mathbb{E}_h[u(h, d) \mid \mathcal{S}(\pi, h)]$ if the policy terminates upon observing $\mathcal{S}(\pi, h)$ and suggests a decision. The reward function measures the expected (total) utility one can get by making a decision after running policy π.

We now consider running Algorithm 3 over k sessions of fixed duration τ. Following the previous discussion, the problem is equivalent to learning to optimize a random finite horizon POMDP of length τ in k repeated episodes of interaction. To establish the regret bound of Theorem 2, we need the following result:

Theorem 3 (Theorem 1 of [24])**.** *Consider the problem of learning to optimize a random finite horizon (PO)MDP $M = \left(\mathcal{B}, \mathcal{T}, R^M, P^M, \tau, \rho\right)$ in k repeated episodes, and consider running the following algorithm: at the start of each episode it updates the prior distribution over the MDP and takes one sample from the posterior, and then follows the policy that is optimal for this sampled MDP. For any prior distribution on the MDPs, it holds that*

$$\mathbb{E}[\mathit{Regret}(k, \tau)] = O\left(\tau |\mathcal{B}| \sqrt{k\tau |\mathcal{T}| \log(k\tau |\mathcal{B}||\mathcal{T}|)}\right).$$

Theorem 3 implies that the posterior sampling strategy as employed in Algorithm 3 allows efficient learning of the MDP, given that one can find the *optimal* policy for the sampled MDP at each episode. However, since finding the optimal policy is NP-hard, in practice we can only *approximate* the optimal policy. In Algorithm 3, we consider running the greedy policy (i.e., Algorithm 2) in each episode to solve the sampled MDP:

Corollary 4. *Let M be a sampled MDP, and c^{wc}_{OPT} be the worst-case cost of the optimal algorithm on M. Consider running Algorithm 2 for $\tau = (2\ln(1/\delta)+1)\, c^{wc}_{\mathsf{OPT}}$ steps. Then, with probability at least $1-\eta$, it achieves the optimal VoI on M.*

Proof of Corollary 4. By Theorem 1, we know that the greedy policy finds the target decision region with probability at least $1-\eta$. Furthermore, by definition we know that each decision region $\mathcal{R}_d = \{h : U(d \mid h) = \mathrm{VoI}(h)\}$ represents an optimal action for any of its enclosed hypotheses. In other words, a policy that successfully outputs a decision region achieves the optimal VoI. □

Denote the optimal policy on the sampled MDP in episode i as OPT_i. From Corollary 4, we know that Algorithm 2 achieves optimal utility with probability at least $1-\eta$. Hence, the expected "regret" of Algorithm 2 over OPT_i is

$$\mathrm{Reg}(\text{Algorithm 2}) \overset{\text{(a)}}{\leq} (1-\eta)\cdot 0 + \eta \cdot 1 = \eta. \tag{8}$$

Here, Step (a) is due to the fact that the utility is normalized so that $U \in [0,1]$. Note that $\mathrm{Reg}(\text{Algorithm 2})$ in Equation (8) refers to the difference between the value of Algorithm 2 and the value of the optimal policy on the sampled MDP (not the optimal policy for the true MDP). In other words, the price of not following the optimal policy is at most η.

By Theorem 3, we know that following OPT_i for episode i achieves expected regret $O\left(\tau|\mathcal{B}|\sqrt{k\tau|\mathcal{T}|\log(k\tau|\mathcal{B}||\mathcal{T}|)}\right)$. Further, we know that the price of approximating the optimal policy at episode i is at most η. Combining these two results we get

$$\begin{aligned}\mathbb{E}[\text{Regret}(k,\tau)] &= O\left(\tau|\mathcal{B}|\sqrt{k\tau|\mathcal{T}|\log(k\tau|\mathcal{B}||\mathcal{T}|)}\right) + \sum_{i=1}^{k}\eta \\ &= O\left(\tau|\mathcal{B}|\sqrt{k\tau|\mathcal{T}|\log(k\tau|\mathcal{B}||\mathcal{T}|)} + \eta k\right),\end{aligned}$$

where $|\mathcal{B}| = S$ represents the number of the belief states, $|\mathcal{T}| = n$ represents the number of tests. Hence it completes the proof. □

D ADDITIONAL RESULTS

In §3.3 of the main paper (i.e., Upper Bounds on the Cost), we have provided upper bounds on the expected/worst-case cost of the greedy policy w.r.t. *non-adaptively* sampled prior. In this section, we provide preliminary results for the case with adaptive re-sampling, where we constantly maintain a $1-\eta$ coverage on posterior distribution over $\mathcal{H}$.

D.1 LOWER BOUND ON THE EXPECTED EC^2 UTILITY

Theorem 5. *Let k, ℓ be positive integers*[8]*, f be the* EC^2 *objective function, $\pi^{g}_{\tilde{\mathcal{H}},[\ell]}$ be the greedy policy with budget ℓ on $\tilde{\mathcal{H}}$, and $\pi^{*}_{\mathcal{H},[k]}$ be the optimal policy that achieves the maximal expected utility under budget k on $\mathcal{H}$, Then,*

$$f_{avg}\left(\pi^{g}_{\tilde{\mathcal{H}},[\ell]}\right) \geq \left(1 - e^{-\ell/k}\right) f_{avg}\left(\pi^{*}_{\mathcal{H},[k]}\right) - k\epsilon,$$

where $\epsilon = 2\eta\left(1-\left(\frac{1}{k}\right)^{\ell}\right)$, and $f_{avg}(\pi) \triangleq \mathbb{E}_h[f(\mathcal{S}(\pi,h))]$ denotes the expected utility of running policy π w.r.t. the original distribution.

[8] If we assume unit cost for all tests, then k, ℓ are the number of tests selected. Otherwise, with non-uniform test costs, k, ℓ are the budget on the cost of selected items.

Note that the above result applies to the EC^2 algorithm with *adaptive-resampled* posteriors at each iteration. The additive term $k\epsilon$ on the RHS is due to the incompleteness of the samples provided by the sampling algorithm. The main intuition behind the proof is that, due to the effect of resampling, the expected one-step gain of the greedy policy $\pi^g_{\tilde{\mathcal{H}},[\ell]}$ on the sampled distribution suffers a small loss at each iteration, comparing to the greedy algorithm on the true distribution. The loss will be accumulated after ℓ rounds, leading to a cumulative loss of up to $k\epsilon$ in the lower bound.

We defer the proof of Theorem 5 to the next subsection (§D.2). In the following we show that an additive term is necessary in the lower bound. That is, we cannot remove the additive term (for example, we cannot push it into the multiplicative term involving $1 - e^{-\ell/k}$).

Suppose the hidden state take two values y_1, y_2 and there are two test t_1, t_2. Let $\eta = 0.1$. The conditional probabilities for the test outcomes are as follows: $p(t_1 = 1 \mid y_1) = p(t_1 = 1 \mid y_2) = 1, p(t_2 = 1 \mid y_1) = 0.001, p(t_2 = 1 \mid y_2) = 0$. There are only two hypotheses with non-zero probability, i.e., $h_1 = (1, 0)$ and $h_2 = (1, 1)$. Further assume there are two distinct decisions d_1, d_2 that are optimal for hypotheses h_1 is h_2 respectively.

However, the sampler will output only one hypothesis $h_1 = (1, 0)$, since $p(h_1 \mid y_1) > 1 - \eta$ and $p(h_2 \mid y_2) > 1 - \eta$. Assume that we further add infinitely many "dummy tests", i.e., for all t in this set, $p(t = 1 \mid y) = 0$ for all y. Then the greedy algorithm will choose those tests with high probability, since the gain for all tests over $\tilde{\mathcal{H}}$ is 0; whereas a smarter algorithm will pick test t_2, because we can identify the target region (and hence obtain a positive gain) upon observing its outcome.

D.2 PROOF OF THEOREM 5

Assume that the cumulative probability of the enumerate hypotheses is at least $1-\eta$, i.e., using our sampling algorithm we enumerate $1 - \eta$ fraction of the total mass.

Denote the set of sampled hypotheses by $\tilde{\mathcal{H}}$, and the expected gain of test t on $\tilde{\mathcal{H}}$ by $\Delta_{\tilde{\mathcal{H}}}(t \mid \cdot)$. Suppose we run the greedy algorithm based on $\tilde{\mathcal{H}}$. We want to show that the following lemma holds:

Lemma 6. *Suppose $\tilde{\mathcal{H}} \subseteq \mathcal{H}$ and $p(\tilde{\mathcal{H}}, \mathbf{x}_\mathcal{A}) \geq (1 - \eta) p(\mathcal{H}, \mathbf{x}_\mathcal{A})$. Let $\tilde{t} \triangleq \arg\max_t \Delta_{\tilde{\mathcal{H}}}(t \mid \mathbf{x}_\mathcal{A})$ be the test with the maximal gain on $\tilde{\mathcal{H}}$ in the* EC^2 *objective function. Then for any test t, it holds that*

$$\Delta_\mathcal{H}(\tilde{t} \mid \mathbf{x}_\mathcal{A}) \geq \Delta_\mathcal{H}(t \mid \mathbf{x}_\mathcal{A}) - 2\eta p(\mathbf{x}_\mathcal{A})^2.$$

That is, the test $\tilde{t}$ which achieves the maximal gain on $\tilde{\mathcal{H}}$ will achieve a gain on $\mathcal{H}$ which is no less than $\varepsilon \triangleq 2\eta p(\mathbf{x}_\mathcal{A})^2$ below the maximal gain of any test. In the following we provide the proof of Lemma 6.

Proof. Clearly, if we can show that for any test t, the gain of t over $\tilde{\mathcal{H}}$ and the gain of t over $\mathcal{H}$ are at most ε apart, i.e.,

$$\Delta_\mathcal{H}(t \mid \mathbf{x}_\mathcal{A}) \leq \Delta_{\tilde{\mathcal{H}}}(t \mid \mathbf{x}_\mathcal{A}) + \varepsilon, \tag{9}$$

then we know that $\Delta_\mathcal{H}(t^* \mid \mathbf{x}_\mathcal{A}) \leq \Delta_{\tilde{\mathcal{H}}}(t^* \mid \mathbf{x}_\mathcal{A}) + \varepsilon \leq \Delta_{\tilde{\mathcal{H}}}(\tilde{t} \mid \mathbf{x}_\mathcal{A}) + \varepsilon$.

In the following, we show that inequality (9) holds.

The conditional expected gain of test t over observed tests $\mathbf{x}_\mathcal{A}$ is

$$\begin{aligned}\Delta_{\tilde{\mathcal{H}}}(t \mid \mathbf{x}_\mathcal{A}) &= \mathbb{E}[\delta_{\tilde{\mathcal{H}}}(x_t \mid \mathbf{x}_\mathcal{A})] \\ &= p(x_t = 1 \mid \mathbf{x}_\mathcal{A}) \delta_{\tilde{\mathcal{H}}}(x_t = 1 \mid \mathbf{x}_\mathcal{A}) + p(x_t = 0 \mid \mathbf{x}_\mathcal{A}) \delta_{\tilde{\mathcal{H}}}(x_t = 0 \mid \mathbf{x}_\mathcal{A}).\end{aligned}$$

Here $\delta_{\tilde{\mathcal{H}}}(x_t \mid \mathbf{x}_\mathcal{A})$ denotes the conditional benefit of test t if its outcome is realized as x_t. Note that we can compute the probability terms $p(x_t = 1 \mid \mathbf{x}_\mathcal{A})$ and $p(x_t = 0 \mid \mathbf{x}_\mathcal{A})$ *exactly* from the CPT $\{\theta_{ij}\}_{n\times m}$ via Bayesian update, i.e., $p(x_t \mid \mathbf{x}_\mathcal{A}) = \sum_y p(x_t, y \mid \mathbf{x}_\mathcal{A}) = \frac{\sum_y p(y) p(\mathbf{x}_\mathcal{A}|y) p(x_t|y)}{\sum_y p(y) p(\mathbf{x}_\mathcal{A}|y)}$. What remains to be approximated is the gain for each specific realization. For EC^2 object function, the gain of observing x_t over hypothesis set $\mathcal{H}$ after having observed $\mathbf{x}_\mathcal{A}$ is

$$\delta_\mathcal{H}(x_t \mid \mathbf{x}_\mathcal{A}) = \sum_{i>j} (p(\mathcal{R}_i, \mathbf{x}_\mathcal{A}) p(\mathcal{R}_j, \mathbf{x}_\mathcal{A}) - p(\mathcal{R}_i, \mathbf{x}_\mathcal{A}, x_t) p(\mathcal{R}_j, \mathbf{x}_\mathcal{A}, x_t)),$$

where $\mathcal{R}_i$ represent the set of hypotheses in *region / equivalence class* i.

We define short-hand notation $\gamma_i := p(\mathcal{R}_i \setminus \tilde{\mathcal{R}}_i, \mathbf{x}_{\mathcal{A}})$, where $\tilde{\mathcal{R}}_i$ denotes the sampled hypotheses of the i^{th} decision region. The difference in the gain of t over $\mathcal{H}$ and $\tilde{\mathcal{H}}$ can be expressed as

$$\begin{aligned}
&\delta_{\mathcal{H}}(x_t \mid \mathbf{x}_{\mathcal{A}}) - \delta_{\tilde{\mathcal{H}}}(x_t \mid \mathbf{x}_{\mathcal{A}}) \\
&= \sum_{i>j} \Big(p(\mathcal{R}_i, \mathbf{x}_{\mathcal{A}}) p(\mathcal{R}_j, \mathbf{x}_{\mathcal{A}}) - p(\tilde{\mathcal{R}}_i, \mathbf{x}_{\mathcal{A}}) p(\tilde{\mathcal{R}}_j, \mathbf{x}_{\mathcal{A}}) \Big) - \\
&\qquad \sum_{i>j} (p(\mathcal{R}_i, \mathbf{x}_{\mathcal{A}}, x_t) p(\mathcal{R}_j, \mathbf{x}_{\mathcal{A}}, x_t) - p(\tilde{\mathcal{R}}_i, \mathbf{x}_{\mathcal{A}}, x_t) p(\tilde{\mathcal{R}}_j, \mathbf{x}_{\mathcal{A}}, x_t) \Big) \\
&\leq \sum_{i>j} (p(\mathcal{R}_i, \mathbf{x}_{\mathcal{A}}) p(\mathcal{R}_j, \mathbf{x}_{\mathcal{A}}) - p(\tilde{\mathcal{R}}_i, \mathbf{x}_{\mathcal{A}}) p(\tilde{\mathcal{R}}_j, \mathbf{x}_{\mathcal{A}}) \Big) \\
&= \sum_{i>j} \left(\left(p(\tilde{\mathcal{R}}_i, \mathbf{x}_{\mathcal{A}}) + \gamma_i \right) \left(p(\tilde{\mathcal{R}}_j, \mathbf{x}_{\mathcal{A}}) + \gamma_j \right) - p(\tilde{\mathcal{R}}_i, \mathbf{x}_{\mathcal{A}}) p(\tilde{\mathcal{R}}_j, \mathbf{x}_{\mathcal{A}}) \right) \\
&= \sum_{i>j} \left(\gamma_i \left(\gamma_j + p(\tilde{\mathcal{R}}_j, \mathbf{x}_{\mathcal{A}}) \right) + \gamma_j p(\tilde{\mathcal{R}}_i, \mathbf{x}_{\mathcal{A}}) \right) \\
&= \sum_{i>j} \left(\gamma_i p(\mathcal{R}_j, \mathbf{x}_{\mathcal{A}}) + \gamma_j p(\tilde{\mathcal{R}}_i, \mathbf{x}_{\mathcal{A}}) \right) \\
&\leq \sum_i \gamma_i \sum_j p(\mathcal{R}_j, \mathbf{x}_{\mathcal{A}}) + \sum_j \gamma_j \sum_i p(\tilde{\mathcal{R}}_i, \mathbf{x}_{\mathcal{A}}).
\end{aligned}$$

By the definition of γ_i we know that

$$\sum_i \gamma_i = p\left(\bigcup_i \left(\mathcal{R}_i \setminus \tilde{\mathcal{R}}_i \right), \mathbf{x}_{\mathcal{A}} \right) \overset{(a)}{=} p\left(\left(\bigcup_i \mathcal{R}_i \setminus \bigcup_i \tilde{\mathcal{R}}_i \right), \mathbf{x}_{\mathcal{A}} \right) = p\left(\mathcal{H} \setminus \tilde{\mathcal{H}}, \mathbf{x}_{\mathcal{A}} \right) \leq \eta p(\mathbf{x}_{\mathcal{A}}).$$

Step (a) is because of the assumption that $\mathcal{R}_i$'s do not overlap. Hence,

$$\begin{aligned}
\Delta_{\mathcal{H}}(t \mid \mathbf{x}_{\mathcal{A}}) - \Delta_{\tilde{\mathcal{H}}}(t \mid \mathbf{x}_{\mathcal{A}}) &= \mathbb{E}[\delta_{\tilde{\mathcal{H}}}(x_t \mid \mathbf{x}_{\mathcal{A}})] \\
&\leq \eta p(\mathbf{x}_{\mathcal{A}}) \sum_j p(\mathcal{R}_j, \mathbf{x}_{\mathcal{A}}) + \eta p(\mathbf{x}_{\mathcal{A}}) \sum_i p(\tilde{\mathcal{R}}_i, \mathbf{x}_{\mathcal{A}}) \\
&\leq 2\eta p(\mathbf{x}_{\mathcal{A}})^2. \qquad (10)
\end{aligned}$$

Combining Equation (9) and (10) we finish the proof. □

Next, we provide the proof of Theorem 5 using the Lemma 6.

Proof of Theorem 5. The key of the proof is to bound the one-step gain of the policy $\pi^g_{\tilde{\mathcal{H}},[\ell]}$.

$$\begin{aligned}
&f_{avg}(\pi^g_{\tilde{\mathcal{H}},[i+1]}) - f_{avg}(\pi^g_{\tilde{\mathcal{H}},[i]}) \\
&\overset{\text{Lemma 6}}{\geq} \mathbb{E}\left[\max_t (\Delta(t \mid \mathbf{x}_{\mathcal{A}})) - 2\eta \right] \\
&\overset{\text{(a)}}{\geq} \mathbb{E}\left[\frac{\Delta(\pi^*_{\mathcal{H},[k]} \mid \mathbf{x}_{\mathcal{A}})}{k} - 2\eta \right] \\
&= \mathbb{E}\left[\frac{f_{avg}(\pi^*_{\mathcal{H},[k]} @ \pi^g_{\tilde{\mathcal{H}},[i]}) - f_{avg}(\pi^g_{\tilde{\mathcal{H}},[i]})}{k} - 2\eta \right] \\
&\overset{\text{(b)}}{\geq} \mathbb{E}\left[\frac{f_{avg}(\pi^*_{\mathcal{H},[k]}) - f_{avg}(\pi^g_{\tilde{\mathcal{H}},[i]})}{k} - 2\eta \right].
\end{aligned}$$

Here $\pi^*_{\mathcal{H},[k]} @ \pi^g_{\tilde{\mathcal{H}},[i]}$ denotes the concatenated policy of $\pi^*_{\mathcal{H},[k]}$ and $\pi^g_{\tilde{\mathcal{H}},[i]}$ (i.e., we first run $\pi^g_{\tilde{\mathcal{H}},[i]}$, and then run $\pi^*_{\mathcal{H},[k]}$ from scratch, ignoring the observations made by $\pi^g_{\tilde{\mathcal{H}},[i]}$).

The proof structure follows closely from the proof of Theorem A.10 in [12]: Step (a) follows from from the adaptive submodularity of f, and step (b) is due to monotonicity of f_{avg}. Define $\Delta_i := f_{avg}(\pi^*_{\mathcal{H},[k]}) - f_{avg}(\pi^g_{\tilde{\mathcal{H}},[i]})$, from the above equation we get $\Delta_\ell \leq \left(1 - \frac{1}{k}\right)^l \Delta_0 + \sum_{i=0}^{l} \left(1 - \frac{1}{k}\right)^i$. Hence, $f_{avg}\left(\pi^g_{\tilde{\mathcal{H}},[\ell]}\right) \geq \left(1 - e^{-\ell/k}\right) f_{avg}\left(\pi^*_{\mathcal{H},[k]}\right) - 2k\eta\left(1 - \left(\frac{1}{k}\right)^\ell\right)$. □