A Quantitative Study of Hypothesis Selection

Philip W. L. Fong
Department of Computer Science
University of Waterloo, Ontario N2L 3G1, Canada
wlfong@logos.uwaterloo.ca

Abstract

The hypothesis selection problem (or the k-armed bandit problem) is central to the realization of many learning systems. This paper studies the minimization of sampling cost in hypothesis selection under a probably approximately optimal (PAO) learning framework. Hypothesis selection algorithms could be *exploration-oriented* or *exploitation-oriented*. Exploration-oriented algorithms tend to explore unfamiliar alternatives eagerly, while exploitation-oriented algorithms focus their sampling effort to the hypotheses which yield higher utility in the past. Both the exploration and exploitation element of a hypothesis selection algorithm could be useful in reducing sampling cost. We propose a novel family of learning algorithms, the γ-IE family, that explicitly trade off their exploration tendency with exploitation tendency. We establish the sample complexity for the entire γ-IE family. We empirically show that none of the algorithms in this family are cost-optimal for all problems. In addition, our novel parameterization of the family allows users to select the instantiation that best fits his or her application. Our results also imply that the PALO class of speed-up learners can retain their theoretical properties even when a more sophisticated sampling strategy is used.

1 Introduction

Competence in hypothesis selection is fundamental to all cognitive tasks, especially those involving learning. Various forms of speed-up learning deal with the selection of an optimal performance element based on performance history (Greiner & Jurisica 1992, Gratch & DeJong 1992). Inductive learning involves the selection of a hypothesis that best fits a given set of data (Musick et al 1993). Reinforcement learning selects a policy that returns best payoff (Kaelbling 1993). Automatic improvement of heuristic estimation can be viewed as the selection of a competent heuristic function among a pool of alternatives (Yakowitz & Lugosi 1990). Even the study of representational shifts could be formulated as the selection of an optimal representation scheme (Greiner & Elkan 1991, Greiner 1992). Recent works in diagnosis (Benjamins & Jansweijer 1994) and probably approximate planning (Yang & Fong 1995) are also concerned with hypothesis selection.

Here, we are not interested in how hypotheses are formed. Instead, we are interested in how a near optimal hypothesis can be selected from a pool of given alternatives. In addition, the only information that we can base our decision on is the actual experimentation of the hypotheses in an uncertain environment. This archetypal hypothesis selection task is called the *k-armed bandit problem* (Berry & Fristedt 1985). One is interested in knowing how the experimentation cost can be minimized while still guaranteeing the optimality of the selected hypothesis.

This paper presents theoretical and empirical studies of a particular family of hypothesis selection algorithms under a *probably approximately optimal (PAO)* learning framework (Greiner & Orponen 1991). Hypothesis selection algorithms differ by their choice of *sampling strategies.* Sampling strategies could be *exploration-oriented* or *exploitation-oriented.* Exploration-oriented strategies are very eager to explore unfamiliar hypotheses. Exploitation-oriented strategies are more biased towards to the hypotheses with high utility. We argue that both the exploration and exploitation element of a hypothesis selection algorithm are important in reducing the cost of experimen-

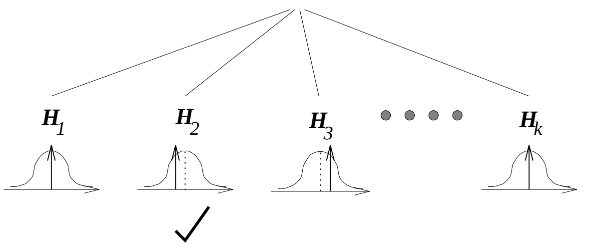

Figure 1: The k-Armed Bandit Problem

tation. We propose a novel family of sampling strategies called the γ-IE strategies, in which every unit of exploitation tendency is balanced out by γ unit of exploration tendency. The behaviour of a γ-IE strategy could be summarized by the formula:

$$(\text{exploitation}) + \gamma \cdot (\text{exploration})$$

We establish the worst case sample complexity for the whole family of γ-IE strategy. Experimental results also suggest that no single member of the γ-IE family is cost-optimal in all cases. In fact, the best tradeoff between exploration and exploitation is a function of the problem domain.

It is observed that, due to the use of an exclusively exploration-oriented sampling strategy, utility analysis in the class of PALO speed-up learners (Greiner 1992) could be unnecessarily costly. Our sampling complexity results imply that more sophisticated strategies like γ-IE can actually be integrated into the PALO framework without sacrificing its theoretical properties.

2 The Hypothesis Selection Problem

Suppose we are facing k alternative hypotheses. Each could be tested against a fixed environment. Every such test returns a numerical measure of how 'good' the hypothesis is. We call this numerical measure the *utility* of the hypothesis. Since the environment could be uncertain, the numerical utility of a hypothesis returned by the environment is in fact a random variable. A learning agent is free to sample the utility of each hypothesis in whatever order it prefers. Our goal is to decide which hypothesis has the highest average utility (figure 1). This so called *k-armed bandit problem* has fascinated researchers of statistics (Berry & Fristedt 1985) and reinforcement learning (Kaelbling 1993) for years. Readers will find no difficulty imagining the usefulness of k-armed bandit problem in factory quality control, in experimental design, and in clinical treatment research. In fact, this problem is intimately related to many learning problems in artificial intelligence, and any algorithmic solution to the problem could be an integral component of various learning systems.

Many hypothesis selection tasks involve a *training phase* and an *utilization phase*. A 'good' hypothesis is selected by experimentation in the training phase, and is then used for further problem solving in the utilization phase. Since the selection of hypothesis is based only on a finite number of samples, we could only expect a learning algorithm to return an approximately optimal hypothesis. Also, skew experimental observation could mislead the learner. At best we could only hope for a probably approximately optimal hypothesis. Following the practice of Greiner and Orponen (Greiner & Orponen 1991), we define our criterion of successful learning to be *probably approximate optimality.*

Given k hypotheses $H_1, H_2, \ldots, H_k$, let $\text{Query}(H_i)$ be a random variable that probabilistically returns the numerical utility of adopting H_i in a fixed environment. Let μ_i be the (unknown) mean of $\text{Query}(H_i)$, and call it the mean utility of H_i. Define $\mu^* = \max_i\{\mu_i\}$. A hypothesis H_i is *optimal* if $\mu_i = \mu^*$. A hypothesis H_i is *ϵ-optimal* if $\mu_i \geq \mu^* - \epsilon$. A learning algorithm L can observe multiple samples of each hypothesis H_i (i.e. sampling random variable $\text{Query}(H_i)$). If, for any $\epsilon > 0$ and $0 < \delta < 1$, after sampling each hypothesis H_i for only finitely many times, L is able to return, with probability at least $1 - \delta$, an ϵ-optimal hypothesis H_i, then L is said to be a *probably approximately optimal (PAO)* algorithm. We also abuse our terminology and say that the returned H_i is PAO. The number of samples that L collects before termination is called its *sample complexity.*

3 PAO Algorithms for Hypothesis Selection

3.1 An Algorithmic Skeleton

A generic approach to hypothesis selection is very straightforward:

> *A learning algorithm estimates the mean utility of each hypothesis by Monte Carlo sampling. According to some predefined fixed rule, the algorithm samples hypotheses one at a time. The precision of estimation increases with the number of samples. Eventually, we select the hypothesis with the highest estimated mean utility.*

To realize this procedure, we need to answer four questions: (1) How do we characterize the precision of the estimations? (2) When can we stop sampling and return a hypothesis guaranteed to be PAO? (3) How do we determine which hypothesis to sample next? (4) What is the sample complexity of such procedure? We answer these questions by deriving a series of theoretical results.

Proposition 1 (Hoeffding 1963)
Let $X_1, X_2, \ldots, X_n$ be n identically independently distributed random variables with finite mean μ. Suppose further that X_i is bounded above and below by U and L respectively. Denote $U - L$ by Δ. Let $\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} X_i$ be an estimation of μ. Then the following is a $100(1-\delta)\%$-confidence interval for μ:

$$\left[\hat{\mu} - \Delta \cdot \frac{z_\delta}{\sqrt{n}}\ ,\ \ \hat{\mu} + \Delta \cdot \frac{z_\delta}{\sqrt{n}}\right]$$

where $z_\delta \stackrel{def}{=} \sqrt{\frac{1}{2}\ln\frac{2}{\delta}}$.

Proposition 2 *Suppose we are given k hypotheses so that each hypothesis H_i has range of utility bounded by Δ_i. Suppose further that we have already collected n_i samples for each H_i, so that the estimated mean utility is $\hat{\mu}_i$. We define the following quantities:*

$$\begin{aligned} \delta_n &\stackrel{def}{=} \frac{6\delta}{\pi^2 n^2} \\ \epsilon_i &\stackrel{def}{=} \Delta_i \cdot \frac{z_{(\delta_{n_i}/k)}}{\sqrt{n_i}} \\ \epsilon_{ij} &\stackrel{def}{=} \epsilon_i + \epsilon_j - (\hat{\mu}_i - \hat{\mu}_j) \end{aligned}$$

If every μ_i lies within its $100(1-\delta_{n_i}/k)\%$ confidence interval as constructed in proposition 1, then any two hypotheses H_i, H_j with $\hat{\mu}_i \geq \hat{\mu}_j$ will have their actual mean utility μ_i, μ_j satisfying $\mu_i \geq \mu_j - \epsilon_{ij}$.

Proof: By proposition 1, $[\hat{\mu}_i - \epsilon_i\ ,\ \hat{\mu}_i + \epsilon_i]$ is a $100(1-\delta_{n_i}/k)\%$ confidence interval of μ_i. If every μ_i lies within its own confidence interval, then for any hypotheses H_i, H_j, we have

$$\mu_i \geq \hat{\mu}_i - \epsilon_i \tag{1}$$
$$\epsilon_j + \hat{\mu}_j \geq \mu_j \tag{2}$$

Combining (1) and (2), we get

$$\mu_i \geq \mu_j - (\epsilon_i + \epsilon_j - (\hat{\mu}_i - \hat{\mu}_j)) = \mu_j - \epsilon_{ij}$$

as required. □

Proposition 3 (Termination Condition) *At any stage of sampling, let $g = argmax_{i=1}^{k}\{\hat{\mu}_i\}$. If $\epsilon_{gj} \leq \epsilon$ for all $j \neq g$, then H_g is PAO.*

Algorithm GenHS(ϵ, δ, k, $\Delta_{1..k}$):
  repeat
    $i :=$ Pick-Next-Hypothesis(...);
    $P :=$ Query(H_i);
    $S_i := S_i + P$; $n_i := n_i + 1$; $\hat{\mu}_i := S_i / n_i$;
    $\epsilon_i := \Delta_i \cdot z_{(\delta_{n_i}/k)} / \sqrt{n_i}$;
    $g := \mathrm{argmax}_{i=1}^{k}\{\hat{\mu}_i\}$;
  until $(\forall j \neq g : \epsilon_g + \epsilon_j - (\hat{\mu}_g - \hat{\mu}_j) \leq \epsilon)$;
  return H_g;

Figure 2: A Generic Hypothesis Selection Algorithm

Proof: Suppose $H_j \neq H_g$ is an optimal hypothesis. Since $\hat{\mu}_g \geq \hat{\mu}_j$ and $\epsilon_{gj} \leq \epsilon$, by proposition 2, as long as every μ_i lies within its own $100(1-\delta_{n_i}/k)\%$ confidence interval, we have $\mu_g \geq \mu_j - \epsilon_{gj} \geq \mu_j - \epsilon$. Hypothesis g is thus ϵ-optimal.

The probability that the above decision can be incorrect is bounded from the above by the probability that some confidence interval constructed during the sampling process does not include its corresponding μ_i. When the n_i-th confidence interval for μ_i is constructed, there is a probability δ_{n_i}/k of excluding μ_i. In the worst case, the total exclusion probability cumulated over all iterations is at most

$$\begin{aligned} \sum_{i=1}^{k}\sum_{n_i=1}^{\infty}\frac{\delta_{n_i}}{k} &= k \cdot \sum_{n=1}^{\infty}\frac{\delta_n}{k} = \sum_{n=1}^{\infty}\delta \cdot \frac{6}{\pi^2 n^2} \\ &= \delta \cdot \frac{6}{\pi^2}\sum_{n=1}^{\infty}\frac{1}{n^2} = \delta \cdot \frac{6}{\pi^2}\cdot\frac{\pi^2}{6} = \delta \end{aligned}$$

□

Figure 2 shows the generic GenHS hypothesis selection algorithm. Accepting parameters ϵ, δ, k, and $\Delta_1, \ldots, \Delta_k$, the GenHS algorithm keeps estimating $\hat{\mu}_i$ and ϵ_i until the termination condition in proposition 3 is satisfied. A hypothesis selection algorithm is realized by instantiating GenHS with a *sampling strategy*, that is, by providing a Pick-Next-Hypothesis() procedure to tell the algorithm which hypothesis to sample next.

GenHS is a *distribution-free* learning algorithm. It does not make explicit assumption on the utility distributions of the hypotheses H_i except boundedness. Although this simple algorithm is guaranteed to return a PAO hypothesis if it terminates, nothing is said about whether the termination condition will ever be met. In the following, we show that there actually exist some sampling strategies which achieve the termination condition using only polynomially many samples.

This completes the solution to the four questions set out in the beginning of this section.

3.2 The Naive Strategy and its Sample Complexity

First, we look at a naive sampling strategy. This strategy eagerly attempts various alternatives, and does so evenly:

Naive Strategy: *Sample the hypotheses in a round-robin fashion — on round t, sample $H_{(t \bmod k)}$. Stop sampling a hypothesis H_i as soon as $\epsilon_i \leq \epsilon/2$*

Proposition 4
Let $m(i) \stackrel{def}{=} 4 \cdot \frac{\Delta_i^2}{\epsilon^2} \cdot \max\{\ln \frac{k\pi^2}{3\delta}, 4 \ln \frac{8\Delta_i^2}{\epsilon^2}\}$. *If* $n_i \geq m(i)$ *then* $\epsilon_i \leq \epsilon/2$.

Theorem 5 *The GenHS algorithm with naive round-robin sampling strategy has worst case sample complexity*[1] $\sum_{i=1}^{k} m(i)$.

Proof: Suppose the algorithm is able to complete $\sum_{i=1}^{k} m(i)$ iterations. By then every hypothesis H_i will satisfy $\epsilon_i \leq \epsilon/2$. So, for $g = \operatorname{argmax}_{i=1}^{k}\{\hat{\mu}_i\}$, and for $j \neq g$, we have

$$\epsilon_{gj} = \epsilon_g + \epsilon_j - (\hat{\mu}_g - \hat{\mu}_j) \leq \epsilon_g + \epsilon_j = \epsilon/2 + \epsilon/2 = \epsilon.$$

The termination condition of proposition 3 is thus satisfied. □

The GenHS algorithm, when appropriately instantiated, is able to figure out a PAO hypothesis by using number of samples polynomial in $\frac{1}{\epsilon}$, $\ln \frac{1}{\delta}$, k, and the total variance of the hypotheses ($\sum_{i=1}^{k} \Delta_i^2$).

3.3 The IE Strategy and its Sample Complexity

The naive strategy is a fair one. It eagerly tries out various alternatives, and thus reduces the size of the confidence intervals efficiently. Because of its eagerness to explore unfamiliar hypotheses, we say that the naive strategy is *exploration-oriented*. However, if sampling bad hypotheses incurs a cost to the learning agent, then one might want to allocate more samples to the better hypotheses. An *exploitation-oriented* strategy exploits its current knowledge about the mean utility of the hypotheses, and concentrates its sampling efforts to the high utility hypotheses. The interval estimation (IE) strategy proposed by Kaelbling (Kaelbling 1993) does exactly this:

Interval Estimation Strategy:
Sample H_i when[2] $i = argmax_{i=1}^{k}\{\hat{\mu}_i + \epsilon_i\}$.

Remember that the GenHS algorithm constructs confidence interval $[\hat{\mu}_i - \epsilon_i \ , \ \hat{\mu}_i + \epsilon_i]$ for the mean utility μ_i of every hypothesis H_i. The IE strategy selects the hypothesis whose confidence interval has the highest upper limit. The rationale behind this strategy is that, the upper limit of a confidence interval is high either because (a) the mean utility of the hypothesis is high, or (b) the estimation of the mean utility is too crude, and thus testing this hypothesis gives us more information for future use. In both cases, the hypothesis is worth trying.

Theorem 6 *The GenHS algorithm with interval estimation sampling strategy has worst case sample complexity* $\sum_{i=1}^{k} m(i)$.

Proof: We make an even stronger claim that hypothesis H_i will never be sampled again once $\epsilon_i \leq \epsilon/2$ is realized. By way of contradiction, assume H_i is the first hypothesis that is sampled after $\epsilon_i \leq \epsilon/2$ is already realized. Suppose this happens at iteration $t+1$. Consider what would happen at iteration t right before the termination condition of GenHS is checked. Let $g = \operatorname{argmax}_{j=1}^{k}\{\hat{\mu}_j\}$. Let H_j be any hypothesis.

By construction of H_i, we have

$$\epsilon_i \leq \epsilon/2. \tag{3}$$

By definition, H_g is the optimal hypothesis, we have

$$\hat{\mu}_g \geq \hat{\mu}_j. \tag{4}$$

Since H_i is going to be selected in the next iteration, it must be the case that

$$\hat{\mu}_j + \epsilon_j \leq \hat{\mu}_i + \epsilon_i. \tag{5}$$

Rewriting (5), we get

$$\begin{aligned} \epsilon_g &\leq \epsilon_i - (\hat{\mu}_g - \hat{\mu}_i) & \\ &\leq \epsilon_i & \text{by (4)} \\ &\leq \epsilon/2 & \text{by (3)} \end{aligned} \tag{6}$$

Therefore, we have

$$\begin{aligned} \epsilon_{gj} &= \epsilon_g + \epsilon_j - (\hat{\mu}_g - \hat{\mu}_j) & \\ &\leq \epsilon_g + \epsilon_i - (\hat{\mu}_g - \hat{\mu}_i) & \text{by (5)} \\ &\leq \epsilon_g + \epsilon_i & \text{by (4)} \\ &\leq \epsilon & \text{by (3) and (6)} \end{aligned}$$

[1]A matching (up to a constant factor) lowerbound can be established for the three sample complexity bounds in theorems 5, 6, and 7 (Fong 1995).

[2]Ties are broken arbitrarily.

As $g = \text{argmax}_{j=1}^{k}\{\hat{\mu}_j\}$ and $\epsilon_{gj} \leq \epsilon$ for all $j \neq g$, the termination condition is going to be satisfied. So the algorithm should have terminated before hypothesis i can actually be sampled again at iteration $t+1$, a contradiction. So H_i will not be sampled again once $\epsilon_i \leq \epsilon/2$. Since $m(i)$ samples are already enough to drive ϵ_i below $\epsilon/2$ (proposition 4). The worst case sample complexity is at most $\sum_{i=1}^{k} m(i)$. □

This is a surprising result to us. Although the IE strategy is highly biased towards hypotheses with higher utility, nevertheless it manages to gather enough information so that its worst case sample complexity is exactly the same as that of the naive strategy. In fact, no sampling strategy can be purely exploitation-oriented. Its exploitation element must be balanced by some exploration element so that enough information is gathered for each hypothesis. The question of how the tradeoff of exploration and exploitation can be settled in a profitable way is the topic of the next section.

4 Trading Off Exploitation and Exploration

4.1 The Issue

Samples are costly. Most applications require the minimization of sample complexity. More complicated than that, sampling bad hypotheses could be undesirable. For example, testing a bad drug on a patient could kill him or her; testing the performance of a slow problem solver could waste a lot of cpu time. (In fact, a major impasse to the practicality of speed-up learning is the computational cost of sampling while utility analysis is performed.) When utility is simply negative cost, we are interested in knowing how the total cost of the whole sampling process could be minimized, that is, in the case when utility is always negative, we want to maximize the total utility.

Intuitively, both exploration-oriented strategies and exploitation-oriented strategies could be useful in reducing cost of the sampling process:

Exploration: Because of its fair allocation and its willingness to explore unfamiliar hypothesis, exploration-oriented strategies reduce confidence intervals more quickly than exploitation-oriented strategies. Exploration is thus good for reducing sample complexity and, hopefully, experimentation cost. However, a fair strategy indifferently allocates the same number of samples to hypotheses with different costs. In some cases, such insensitivity to cost distribution may render an exploration-oriented sampling process more costly than an exploitation-oriented one.

Exploitation: Exploration-oriented strategies concentrate more on the better hypotheses, and thus they receive higher utility per sample. This might be useful in reducing the total cost of sampling. However, such focusing of sample allocation is very slow in gathering statistics for the rest of the hypotheses. Occasionally, a bias in sample allocation might result in an increase in sample complexity, and in turn imply a potential increase in sampling cost.

The above intuitions lead us to conjecture that both exploration and exploitation play a positive role in hypothesis selection. In fact, we suspect that **neither exploration-oriented algorithms nor exploitation-oriented algorithms are cost-optimal in all cases.** To achieve the best result, we believe, **a sampling strategy must carefully trade off its exploration tendency with its exploitation tendency.** In a coming section, we will set forth to verify the above series of conjectures by experiments. Before that, we will derive a mechanism that makes trading off exploration and exploitation possible.

4.2 The γ-IE Strategy and its Sample Complexity

The IE strategy selects a hypothesis that maximizes the expression $\hat{\mu}_i + \epsilon_i$. The term ϵ_i expresses the exploration element in the strategy — the larger ϵ_i is, the less information we know about H_i, the more we want to explore it. The term $\hat{\mu}_i$ expresses the exploitation element in the strategy — the higher the estimated mean utility is, the more we want to exploit it. In fact, IE is maximizing a particular linear combination of utility and error. It represents one particular way to specify the tradeoff between exploration (ϵ_i) and exploitation ($\hat{\mu}_i$). Here, one unit of information trades off with exactly one unit of utility. However, there is no reason to stop us from proposing a tradeoff scheme in which one unit of information is worth γ unit of utility, that is, a strategy that maximizes $\hat{\mu}_i + \gamma\epsilon_i$. We call this a *$\gamma$-IE strategy* with *tradeoff ratio* γ.

γ-IE Strategy:
Sample H_i when $i = argmax_{i=1}^{k}\{\hat{\mu}_i + \gamma\epsilon_i\}$.

We show that the γ-parameterization does not affect the worst case sample complexity.

Theorem 7 *The GenHS algorithm with γ-IE sampling strategy (for $\gamma \geq 1$) has sample complexity*[3] $\sum_{i=1}^{k} m(i)$.

Proof: The proof follows an argument almost identical to the proof of theorem 6. In the same way that we obtain inequalities (3), (4), and (5), we have the following inequalities:

$\epsilon_i \leq \epsilon/2$; $\quad \hat{\mu}_g \geq \hat{\mu}_j$; $\quad \hat{\mu}_j + \gamma\epsilon_j \leq \hat{\mu}_i + \gamma\epsilon_i$.

Repeating the argument in the proof of theorem 6, we obtain

$$\epsilon_g + \epsilon_j - \frac{1}{\gamma}\left(\hat{\mu}_g - \hat{\mu}_j\right) \leq \epsilon.$$

Since $\frac{1}{\gamma} \leq 1$, by transitivity, we conclude

$$\epsilon_{ij} = \epsilon_i + \epsilon_j - (\hat{\mu}_i - \hat{\mu}_j) \leq \epsilon_i + \epsilon_j - \frac{1}{\gamma}\left(\hat{\mu}_i - \hat{\mu}_j\right) \leq \epsilon$$

as we did in the proof of theorem 6. □

When the tradeoff ratio γ is 1, γ-IE degenerates to the original IE strategy. When $\gamma \to \infty$, we get back the naive strategy (assuming uniform variances Δ_i). The intermediate γ-IE will have behaviour lying somewhere between the two extremes. The larger γ is, the more even its sample allocation. The smaller γ is, the more it biases its sample allocation to the hypotheses with high utility (figure 4).

4.3 Empirical Evaluation

In this section, we will demonstrate empirically that the cost-optimal tradeoff between exploitation and exploration is a domain-dependent notion. We use the GenHS algorithm to select PAO hypotheses from four different instances of k-armed bandit problem. We in turn instantiate GenHS with the naive strategy and the γ-IE strategy, with γ being set to 1, 2, ..., 16. We fix $\delta = 0.05$ and $\epsilon = 0.1$. For each problem instance, and for each sampling strategy, we repeat the experiment 100 times, and then measure the average total cost of the hypothesis selection process. By cost we simply mean negative utility.

The utility of each hypothesis has normal distribution with tail probabilities being truncated. All probabilities beyond the 2nd standard deviation are now concentrated at the two ends. We define $\sigma = 1$. The four problems that we looked at are the following:

1. There are $k = 10$ hypotheses, in which $\mu_0 = -2\sigma$, $\mu_i = -2\sigma - \epsilon$ for $i \neq 0$, and $\sigma_i = \sigma$, for $i = 0, \ldots, k-1$. Basically, $H_1, \ldots, H_9$ have the same utility, and H_0 has a higher utility.

2. There are $k = 10$ hypotheses, in which $\mu_i = -2\sigma - i\epsilon$ and $\sigma_i = \sigma$ for $i = 0, \ldots, k-1$. Basically, the mean utility of each hypothesis decreases as one walks from H_0 to H_9.

3. There are $k = 3$ hypotheses, in which $\mu_i = -2\sigma - i\epsilon$ and $\sigma_i = \sigma - i\epsilon$ for $i = 0, 1, 2$. Basically, H_0 has a higher utility and larger variance than H_1 and H_2.

4. There are $k = 2$ hypotheses, in which $\mu_i = -2\sigma - i\epsilon$ and $\sigma_i = \sigma$ for $i = 0, 1$. Basically, H_0 has a higher utility than H_1 and their variances are the same.

In all cases, $\Delta_i = 4\sigma_i$.

The average total costs of solving the four problems are plotted against the various setting of the tradeoff ratio γ in figure 3. As we have hypothesized, neither the IE strategy nor the naive strategy is the best in all cases. The IE strategy ($\gamma = 1$) is near-optimal in problem 1, but the naive strategy ($\gamma = \infty$) is optimal only in problem 4. In fact, the results of problem 2 and problem 3 show us that intermediate tradeoff between exploration and exploitation could actually be cost-optimal. In these cases the optimal instantiation of γ are 2 and 5 respectively. This presents a very strong evidence that the cost-optimal tradeoff between exploration and exploitation is a function of the problem structure.

As seen from the results, some of the problem features that affect the optimal setting of γ are (a) the number of hypotheses (compare problem 1 and 4), (b) the relative distribution of mean utilities μ_i (compare problem 1 and 2), and (c) the variance of the distributions (compare problem 3 and 4). Experience suggests the following rules of thumb:

- More exploitation-oriented strategies are better for problems involving many hypotheses, and more exploration-oriented strategies are usually more appropriate for problems with only a small number of hypotheses (see figure 4).

- Setting γ to somewhere between 2 and 5 seems to work well for many typical problems in which the number of hypotheses is large, the mean utilities of the hypotheses are heterogeneous, and the variances are not uniform.

[3] The requirement that $\gamma \geq 1$ turns out to be important. Computational experience tells us that GenHS could fail to terminate within $\sum_{i=1}^{k} m(i)$ iterations when it is instantiated with a γ-IE strategy having $\gamma < 1$. Using an adversary argument, it can be shown analytically that GenHS may loop forever when γ-IE is used with $\gamma < 1$ (Fong 1995).

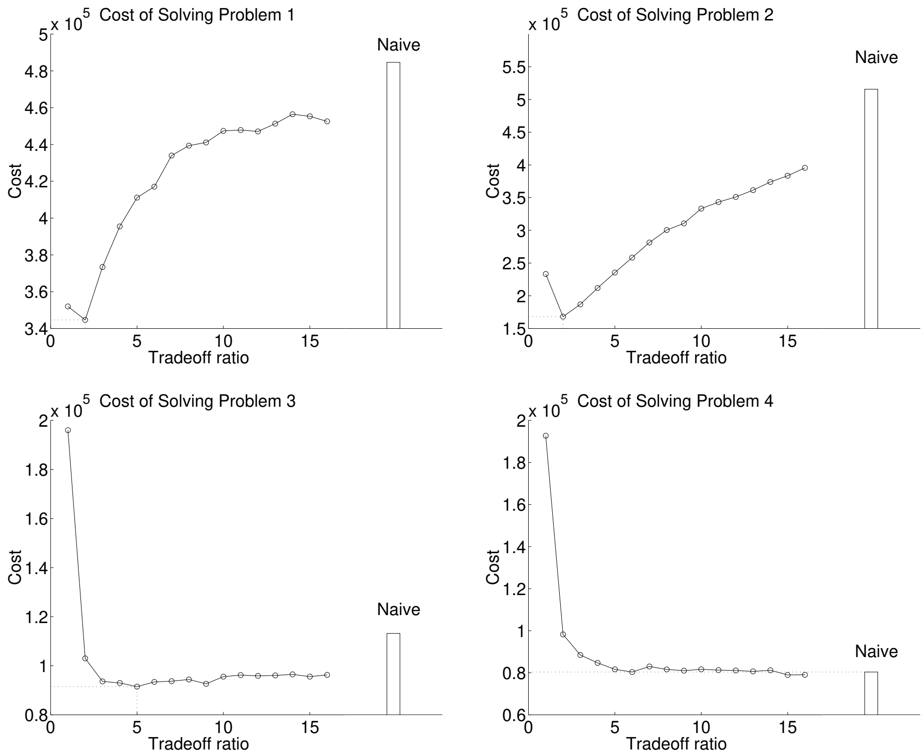

Figure 3: Cost of Problem Solving Using Various Sampling Strategies

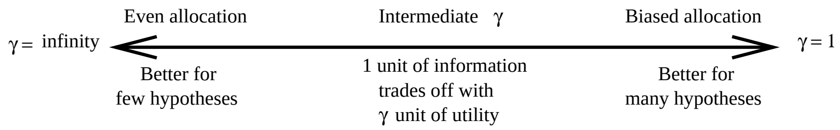

Figure 4: A Spectrum of Sampling Strategies

However, the lesson here is not so much that a particular algorithm in the γ-IE family is the best, but that **the problem defines which one is optimal**.

5 Implication to Probabilistic Hill-Climbing

The above results also have bearing on speedup learning. Greiner's PALO algorithm (Greiner 1992) performs probabilistic hill-climbing in a space of problem solvers. At each stage, a set of transformations are applied to a current problem solver to yield several alternative problem solvers. Then the alternatives are actually experimented with real training problem instances to determine their performance improvement over the original problem solver. Sampling continues until either (1) a problem solver with positive improvement is found, or, (2) the performance improvement of every alternative is upper-bounded by ϵ. In case (1), PALO hill climbs to the new problem solver with improved performance. In case (2), PALO terminates. Since PALO makes a mistake with probability no more than δ, the returned problem solver is thus *probably approximately locally optimal (PALO)*.

In fact, PALO is dealing with a hypothesis selection problem at each stage. The sampling strategy it uses is called *simultaneous extraction*, which is basically our naive strategy. Gratch et al (Gratch et al 1994) complains that such strategy overlooks the cost of sampling, and argues that more sophisticated strategy should be used[4]. However, if we do that, can we still reclaim the theoretical properties of PALO, especially its sample complexity per stage? We give a positive answer to this question.

[4]In some special cases, PALO addresses this problem by replacing actual sampling with a more tractable analysis that approximates the utility of a transformation (Greiner & Jurisica 1992).

Theorem 8 *Given $\epsilon > 0$ and $0 < \delta < 1$, when the GenHS algorithm terminates, we could either find H_i so that $\mu_i \geq 0$, or conclude that every H_i satisfies $\mu_i \leq \epsilon$. Moreover, such decisions could only be wrong with probability at most δ.*

Proof: Let H_i be the PAO hypothesis returned by the GenHS algorithm. If $\hat{\mu}_i - \epsilon_i \geq 0$ then we know that, with probability at least $1 - \delta$, $\mu_i \geq 0$. On the other hand, if $\hat{\mu}_i - \epsilon_i < 0$, then no H_j could have $\hat{\mu}_j + \epsilon_j > \epsilon$, or the termination condition $\epsilon_i + \epsilon_j - (\hat{\mu}_i - \hat{\mu}_j) \leq \epsilon$ will be violated. Thus, with probability at least $1 - \delta$, $\mu_j < \epsilon$ for all H_j. □

The theorem implies that determining positive performance improvement (i.e. whether some $\mu_i > 0$) takes a number of samples no more than determining a PAO hypothesis. This in turn implies that the sample complexity bounds that we established in the previous sections are also valid in the case of determining positive performance improvement. Thus, we have demonstrated that, even when PALO's simultaneous extraction is replaced by a more sophisticated sampling strategy like IE or γ-IE, one could still (1) guarantee the returned problem solver is PALO and, (2) establish a sample complexity per hill-climb.

On the other hand, our empirical results imply that no instantiation of γ is the best for PALO. The best tradeoff between exploration and exploitation is a problem-dependent notion. Nevertheless, our γ parameterization of IE provides a convenient way for the user to specify the instantiation that fits his or her speed-up learning task.

6 Related Work

The PAO learning framework was inspired by Valiant's PAC learning framework (Valiant 1984), and was first proposed by Greiner and Orponen (Greiner & Orponen 1991). Maron and Moore applied similar analysis to their Hoeffding Race algorithm (Maron & Moore 1993), but the algorithm is not provably PAO because it does not handle the indefinite cumulation of error probability over multiple iterations (see proposition 3). Obvious improvement to our framework would be the use of recently advocated sequential methods (Gratch & DeJong 1992, Schuurmans 1995, Wald 1947) instead of Chernoff-Hoeffding bound.

The notion of exploration and exploitation tradeoff is well known in the reinforcement learning literature (Kaelbling 1993). Our novel parameterization of γ-IE provides a generic, provably well-behaved way of specifying arbitrary tradeoff ratio between exploration and exploitation. Two future research problems beg for attention: (1) Is there an explicit characterization of when to use what tradeoff scheme? (2) Can we mechanically determine the best strategy for a given problem? The solution to these questions will definitely improve the performance of adaptive systems like speed-up learners.

Notice that the dimension on which γ-IE is parameterized is only one small space of heuristics for the k-armed bandit problem. For other possibilities, consult (Berry & Fristedt 1985, Gittins 1989).

The utility problem (Minton 1988) is a central issue in speed-up learning. Utility of a transformation has to be evaluated statistically before it is applied to a problem solver. PALO is a notable example of speed-up learner that does this. The application of our analytical results to speed-up learner like PALO demonstrates the possibility of integrating sophisticated sampling strategies into PALO without sacrificing its theoretical properties.

Our study gives empirical basis for the theoretical analysis of Gratch et al (Gratch et al 1994), which notices that optimal allocation of samples is dependent on the structure of the problem. While our analysis focuses on the characteristics of the sampling strategy, theirs characterizes a problem by its *disparity indices.* They also propose an algorithm which dynamically adapts its sampling strategy to the problem. This *rational resource allocation* algorithm selects the hypothesis H_i that maximizes $\Delta\delta_i / \hat{C}_i$, where $\Delta\delta_i$ is the increase of confidence when H_i is sampled, and $\hat{C}_i$ is the estimated cost of sampling H_i. In our framework, this can be naturally understood as another way of trading off exploration (represented by $\Delta\delta_i$) and exploitation (represented by $\hat{C}_i$). In fact, they are maximizing the linear combination $-\ln \hat{C}_i + \ln \Delta\delta_i$. One could always consider parameterizing the algorithm by maximizing $-\ln \hat{C}_i + \gamma \ln \Delta\delta_i$. We expect, again, different setting of γ will be good for different problem. Similar parameterization (Fong 1995) is also available for sampling strategies like the adaptive allocation rule (Lai & Robbins 1985) and the Z-heuristics (Rivest & Yin 1994).

7 Conclusion

We studied the hypothesis selection problem under the PAO learning framework. Worst case sample complexity was established for a family of sampling strategies called γ-IE. Members of this family represent different ways of trading off exploration tendency and exploitation tendency. The impact of such tradeoff on the average case behavior of hypothesis selection was examined empirically.

The following are the major contributions of this paper: (1) We demonstrated that sophisticated sampling strategies like γ-IE could be integrated into the PALO framework without sacrificing its theoretical properties. (2) We showed that no particular instantiation of γ-IE is the best for all problem. The optimal instantiation of γ is in fact a function of the problem structure. (3) The γ-IE parameterization provides the user a mechanism to specify a tradeoff ratio between exploration and exploitation that best fits his or her application.

Acknowledgements

The author is supported by a Postgraduate Scholarship from the National Sciences and Engineering Research Council of Canada.

Special thanks to E. Bach, J. Gratch, R. Greiner, M. Li, P. Ragde, D. Schuurmans, and J. Shallit for their valuable comments on an early draft of the paper. I am especially grateful to F. Ho and my supervisor Q. Yang for many stimulating discussions that led to the work reported here.

References

Benjamins, R. & Jansweijer, W. (1994). Toward a Competence Theory of Diagnosis. *IEEE Expert*, 9(5):43–52.

Berry, D. A. & Fristedt, B. (1985). *Bandit Problems: Sequential Allocation of Experiments.* Chapman and Hall.

Fong, P. W. L. (1995). *A Quantitative Study of Hypothesis Selection.* Master's thesis, Department of Computer Science, University of Waterloo, Canada.

Gittins, J. C. (1989). *Multi-Armed Bandit Allocation Indices.* New York: John Wiley & Sons.

Gratch, J., Chien, S., & DeJong, G. (1994). Improving Learning Performance Through Rational Resource Allocation. In *AAAI-94*, 576–581.

Gratch, J. & DeJong, G. (1992). COMPOSER: A Probabilistic Solution to the Utility Problem in Speedup Learning. In *AAAI-92*, 235–240.

Greiner, R. (1992). Probabilistic Hill-Climbing: Theory and Applications. In *CSCSI-92*, 60–67.

Greiner, R. & Elkan, C. (1991). Measuring and Improving the Effectiveness of Representations. In *IJCAI-91*, 518–524.

Greiner, R. & Jurisica, I. (1992). A Statistical Approach to Solving the EBL Utility Problem. In *AAAI-92*, 241–248.

Greiner, R. & Orponen, P. (1991). Probably Approximately Optimal Satisficing Strategies. In *KR-91.*

Hoeffding, W. (1963). Probability Inequalities for Sums of Bounded Random Variables. *American Statistical Association Journal*, pages 13–30, March 1963.

Kaelbling, L. P. (1993). *Learning in Embedded Systems.* Cambridge, MA: MIT Press.

Lai, T. L. & Robbins, H. (1985). Asymptotically Efficient Adaptive Allocation Rules. *Advances in Applied Mathematics*, 6:4–22.

Maron, O. & Moore, A. (1993). Hoeffding Races: Accelerating Model Selection Search for Classification and Function Approximation. In *Advances in Neural Information Processing Systems 6.*

Minton, S. (1988). *Learning Search Control Knowledge: An Explanation-Based Approach.* Boston: Kluwer Academic Publishers.

Musick, R., Catlett, J., & Russell S. (1993). Decision Theoretic Subsampling for Induction on Large Databases. In *ML-93*, 212–219.

Rivest, R. L. & Yin, Y. (1994). Simulation Results for a New Two-armed Bandit Heuristic. In Hanson, S. J., Drastal, G. A., & Rivest, R. L. (ed.), *Computational Learning Theory and Natural Learning Systems 1*, 477–486. Cambridge, MA: MIT Press.

Schuurmans, D. (1995). *Forthcoming.* PhD thesis, Department of Computer Science, University of Toronto, Canada.

Valiant, L. G. (1984). A Theory of the Learnable. *Communications of the ACM*, 27(11):1134–1142.

Wald, A. (1947). *Sequential Analysis.* New York:John Wiley & Sons.

Yakowitz, S. & Lugosi, E. (1990). Random Search in the Presence of Noise, with Application to Machine Learning. *SIAM Journal on Scientific and Statistical Computing*, 11(4):702–712.

Yang, Q. & Fong, P. W. L. (1995). A Formal Framework for Approximate Planning Under Uncertainty. In *IJCAI-95 Workshop on Agent Theories, Architectures, and Languages*, 1995. Submitted.