Proceedings of the 2007 IEEE Symposium on Approximate
Dynamic Programming and Reinforcement Learning (ADPRL 2007)

The Knowledge Gradient Policy for Offline Learning with Independent Normal Rewards

Peter Frazier
Department of Operations Research
and Financial Engineering
Princeton University
Engineering Quadrangle
Princeton, NJ 08544
Phone: +1 609 258 6239
Email: pfrazier@princeton.edu

Warren Powell
Department of Operations Research
and Financial Engineering
Princeton University
Engineering Quadrangle
Princeton, NJ 08544
Phone: +1 609 258 6239
Email: powell@princeton.edu

***Abstract*— We define a new type of policy, the knowledge gradient policy, in the context of an offline learning problem. We show how to compute the knowledge gradient policy efficiently and demonstrate through Monte Carlo simulations that it performs as well or better than a number of existing learning policies.**

I. Introduction

We consider a problem in which we are presented with a finite set of alternatives, allowed to sample their values a fixed number of times, and then asked to choose one alternative from among the set. We receive a reward equal to the value of the alternative chosen. Our goal is to distribute the sample measurements to maximize the expected reward.

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and let $\{1, \ldots M\}$ be the set of alternatives. For each $x \in \{1, \ldots M\}$ define a random variable Y_x to be the true value of alternative x. We will be allotted exactly N measurements, and at each time n, $0 \leq n < N$, we choose an alternative x^n to measure. Define $\hat{y}^{n+1}$ to be the measurement value observed, and define $\varepsilon^{n+1} := \hat{y}^{n+1} - Y_x$ to be the error in this measurement. We assume that the errors ε^{n+1} are unbiased and normally distributed with a known variance σ^2 that is the same across all alternatives. We also assume that errors are independent of each other and of the random vector Y. At time N, we choose an implementation decision x^N based on the measurements recorded and we receive a reward $\hat{y}^{N+1}$. We assume that the reward is unbiased, so that $\hat{y}^{N+1}$ satisfies $\mathbb{E}\left[\hat{y}^{N+1} | Y, x^N\right] = Y_{x^N}$. Define the filtration $(\mathcal{F}^n)_{n=0}^{n=N}$ by letting $\mathcal{F}^n$ be the sigma algebra generated by $x^0, \hat{y}^1, x^1, \ldots x^{n-1}, \hat{y}$. Measurement and implementation decisions x^n are restricted to be $\mathcal{F}^n$ measurable so that decisions may only depend on measurements observed in the past.

We assume a Bayesian setting for the problem in which we have a prior distribution on the random vector Y. Define $\mu^0 := \mathbb{E}[Y]$ to be the mean and $\Sigma^0 := \mathit{Var}[Y]$ the covariance under this prior distribution. We assume the prior is multivariate normal with independent components so that $Y \sim \mathcal{N}(\mu^0, \Sigma^0)$ under $\mathbb{P}$ with Σ^0 diagonal. We will use Bayes' rule to form a sequence of posterior distributions from this prior and the successive measurements. Define $\mu^n := \mathbb{E}_n[Y]$ to be the mean vector and $\Sigma^n := \mathit{Var}_n[Y]$ the covariance matrix under the posterior after n measurements have been made. Because the error terms ε^{n+1} are independent and normally distributed, Y will remain normally distributed with independent components, so that $Y \sim \mathcal{N}(\mu^n, \Sigma^n)$ under $\mathbb{P}$ conditioned on $\mathcal{F}^n$, with Σ^n almost surely diagonal.

Let us motivate this problem with an example. Suppose we are designing software that will transfer large blocks of data across the internet, e.g. for video on demand. The software system resides on the user's computer and the data to be transferred is mirrored at several different network locations. When the user requests a particular block of data the software must decide from which mirror to obtain it. Once the mirror is chosen, the path established through the network to the mirror is fixed for the duration of the transfer. Our goal is to choose the mirror that will provide the fastest transfer. The bandwidth to each mirror varies with network traffic intensity and cannot be forecast perfectly, but the software can estimate this bandwidth using historical data and geographic location, and can sample the current bandwidth by briefly transferring some data from the mirror before starting the full transfer. We assume that the sample transfers must be performed sequentially to avoid measurement errors due to congestion on local network links, and the number of bandwidth measurements the software may perform is limited by a design requirement to begin the transfer without undue delay. Our problem is to decide which mirrors the software should sample and which mirror it should choose afterward for the full transfer.

Similar problems have been investigated within the literature. Design of experiments [1] addresses the general question of how one should make measurements. Within this literature, algorithms from response surface methods [2] search for the maximum of some "response surface" using sequential experiments. This work is similar to our own but differs in that it assumes a continuous domain and a convex or concave response surface, and it does not employ the Bayesian, decision theoretic approach employed here.

Another area, ranking and selection, assumes as we do that the space of alternatives is discrete, and although the

1-4244-0706-0/07/$20.00 ©2007 IEEE

largest part of the ranking and selection literature focuses on non-sequential or partially sequential experimental designs [3], some investigations into fully sequential designs [5], [6] have also been made. Ranking and selection differs from our own work, however, in that it adopts a classical rather than Bayesian approach. It minimizes the number of measurements needed to guarantee that the probability of implementing a sub-optimal alternative is below some threshold, subject to the condition that the true values of the alternatives are not too close together. Our desire to penalize a policy more for implementing an alternative whose true value is very far from the best true value led us to use a different objective function than that used by ranking and selection.

The work within sequential design of experiments [7] adopts a Bayesian approach most similar to our own. Within this literature, our problem is most similar to the multi-armed bandit problem [8], [9], with the main difference being that our problem has a finite horizon and is concerned with offline learning while the multi-armed bandit problem has an infinite horizon and is concerned with online learning (although see generalizations like the restless bandit [10]). In offline learning there are distinct measurement and implementation phases, while in online learning, measurement and implementation occur simultaneously. We discuss these important differences further in section III-C. Applications of sequential design of experiments and multi-armed bandits to optimal learning, speed of convergence, and the issue of exploration vs. exploitation have also been discussed in the reinforcement learning [11], [16] and optimal control [12] literature.

In this paper we formulate our problem as a dynamic program, and then define a knowledge gradient policy which is optimal in some special cases and computationally tractable in all cases. We compare the knowledge gradient policy against existing policies with Monte Carlo simulation and demonstrate that the knowledge gradient policy performs as well or better than other policies across a broad class of problem settings.

II. Dynamic Programming

We will analyze the problem using dynamic programming. First we develop the transition function and an objective function which explicitly considers the choice of the implementation decision. We show that the optimal implementation decision is one of pure exploitation, and once the implementation decision is fixed to the optimal one, the objective function can be simplified. We formulate the problem as a dynamic program using this simplified objective function.

A. State Space and Transition Function

Our state space at time n is the space of all possible prior distributions for Y under $\mathbb{P}$ conditioned on $\mathcal{F}^n$. It can be shown by induction that all possible priors are multivariate normal, which allows us to parameterize the space by the mean vector μ^n and covariance matrix Σ^n.

Fix a time n. We use Bayes' rule to update the prior, $\mathbb{P}$ conditioned on $\mathcal{F}^n$, to reflect the observation $\hat{y}^{n+1} = Y_x + \varepsilon^{n+1}$, obtaining a posterior which is $\mathbb{P}$ conditioned on $\mathcal{F}^{n+1}$. Since ε^{n+1} is an independent normal random variable and the family of normal distributions is closed under sampling, the posterior distribution is also normal. Thus writing our posterior distribution as a function of the prior and the observation reduces to writing μ^{n+1} and Σ^{n+1} as functions of μ^n, Σ^n and $\hat{y}^{n+1}$. Bayes' rule tells us these functions are

$$\begin{aligned}\mu^{n+1} &= \Sigma^{n+1}\left((\Sigma^n)^{-1}\mu^n + \sigma^{-2}\hat{y}^{n+1}e_{x^n}\right)\\ \Sigma^{n+1} &= ((\Sigma^n)^{-1} + \sigma^{-2}e_{x^n}e_{x^n}^T)^{-1},\end{aligned}$$

where e_x is a column vector of zeros with a single 1 at index x.

Since Σ^0 is assumed diagonal, it follows via induction that Σ^n is diagonal for all n. This implies that the random vector Y has independent components under the probability measure conditioned on the filtration at any time n, and it allows discarding all but the diagonal elements of Σ^n from the state space. It also simplifies the transition function to

$$\mu_x^{n+1} = \begin{cases}\Sigma_{xx}^{n+1}\left((\Sigma_{xx}^n)^{-1}\mu_x^n + \sigma^{-2}\hat{y}^{n+1}\right) & \text{if } x^n = x,\\ \mu_x^n & \text{otherwise,}\end{cases} \tag{1}$$

$$\Sigma_x^{n+1} = \begin{cases}((\Sigma_{xx}^n)^{-1} + \sigma^{-2})^{-1} & \text{if } x^n = x,\\ \Sigma_{xx}^n & \text{otherwise.}\end{cases} \tag{2}$$

Under $\mathbb{P}$ conditioned on $\mathcal{F}^n$, $\hat{y}^{n+1}$ is a normal random variable with mean μ_x^n and variance $\sigma^2 + \Sigma_{xx}^n$. Note that Σ^{n+1} is $\mathcal{F}^n$ measurable rather than merely $\mathcal{F}^{n+1}$ measurable.

When considered as a random variable conditioned on $\mathcal{F}^n$, μ^{n+1} is a multivariate normal random variable whose mean and variance we can compute. First, we use the tower property of conditional expectation, and the definitions of μ^n and μ^{n+1} as the conditional means of Y with respect to $\mathcal{F}^n$ and $\mathcal{F}^{n+1}$ respectively, to write

$$\mathbb{E}_n\left[\mu^{n+1}\right] = \mathbb{E}_n\left[\mathbb{E}_{n+1}\left[Y\right]\right] = \mathbb{E}_n\left[Y\right] = \mu^n.$$

Then, we compute the variance of μ^{n+1} componentwise. For those alternatives $x \neq x^n$ which we do not measure at time n, our posterior is equal to our prior and $\mu^{n+1} = \mu^n$. This shows that $\mathit{Var}_n\left[\mu_x^{n+1}\right] = 0$ if $x \neq x^n$. Fixing $x = x^n$ momentarily, the variance of the component μ_x^{n+1} is computed using (1) as

$$\begin{aligned}\mathit{Var}_n\left[\mu_x^{n+1}\right] &= \mathit{Var}_n\left[\Sigma_{xx}^{n+1}\left((\Sigma_{xx}^n)^{-1}\mu_x^n + \sigma^{-2}\hat{y}^{n+1}\right)\right]\\ &= \sigma^{-4}(\Sigma_{xx}^{n+1})^2\mathit{Var}_n\left[Y_x + \varepsilon^{n+1}\right]\\ &= \sigma^{-4}(\Sigma_{xx}^{n+1})^2\left(\Sigma_{xx}^n + \sigma^2\right)\\ &= \sigma^{-4}\left((\Sigma_{xx}^n)^{-1} + \sigma^{-2}\right)^{-2}\left(\Sigma_{xx}^n + \sigma^2\right)\\ &= \left(\sigma^2(\Sigma_{xx}^n)^{-1} + 1\right)^{-2}\Sigma_{xx}^n\left(1 + \sigma^2(\Sigma_{xx}^n)^{-1}\right)\\ &= \Sigma_{xx}^n / \left(1 + \sigma^2(\Sigma_{xx}^n)^{-1}\right).\end{aligned} \tag{3}$$

Later, it will be more natural to parameterize the family of prior and posterior distributions by the mean and *inverse* variance of the distribution of Y under the current filtration, rather than by the mean and variance. Under this transformation, our state space at time n is $\mathbb{S} := \mathbb{R}^M \times \overline{\mathbb{R}}_+^M$, where $\overline{\mathbb{R}}_+ = [0, \infty]$. $\mathbb{S}$ is indexed by $\mu \in \mathbb{R}^M$ and $\beta \in \overline{\mathbb{R}}_+^M$, where $\beta_x = (\Sigma_{xx})^{-1}$.

We will often write $S = (\mu, \beta)$ for a generic state at any time, and $S^n = (\mu^n, \beta^n)$ for a state at time n.

Using the inverse variance, and fixing $x = x^n$, we define a function $\tilde{\sigma} : \overline{\mathbb{R}}_+ \to \overline{\mathbb{R}}_+$ that gives the standard deviation of μ_x^{n+1} with respect to $\mathcal{F}^n$ as a function of β_x^n. We already calculated this quantity as a function of the variance in (3), so we simply rewrite it as a function of the inverse variance,

$$\begin{aligned}\tilde{\sigma}(b) &:= \sqrt{Var\left[\mu_x^{n+1} \mid \beta_x^n = b\right]} = \sqrt{Var\left[\mu_x^{n+1} \mid \Sigma_{xx}^n = 1/b\right]} \\ &= \sqrt{b^{-1}/\left(1 + b\sigma^2\right)} = 1/\sqrt{b(1 + b\sigma^2)}. \qquad (4)\end{aligned}$$

We may also rewrite the state transition equations (1) and (2) in terms of the mean and inverse variance. When we write these transformed state transition equations, we replace dependence on $\hat{y}^{n+1}$ with dependence on a standard normal random variable Z^{n+1} that is determined by y^{n+1}. This makes the probability distribution of the state transition more apparent, and simplifies our calculations.

Since μ^{n+1} is a normal random variable with respect to $\mathbb{P}$ conditioned on $\mathcal{F}^n$ with the parameters computed above, there exists a random variable Z^{n+1} that is standard normal, also with respect to $\mathbb{P}$ conditioned on $\mathcal{F}^n$, such that

$$\mu^{n+1} = \mu^n + \tilde{\sigma}(\beta_{x^n}^n) Z^{n+1} e_{x^n} \qquad (5)$$

$$\beta^{n+1} = \beta^n + \sigma^{-2} e_{x^n}. \qquad (6)$$

We will also use the notation S^M with $S^{n+1} = S^M(S^n, x^n, Z^{n+1})$ to denote the state transition function, or "model" equation. This function S^M is defined as

$$S^M((\mu, \beta), x, z) := (\mu + \tilde{\sigma}(\beta_x) z e_x, \beta + \sigma^{-2} e_x). \qquad (7)$$

B. Objective Function

First we consider policies that include both measurement decisions and the single final implementation decision. We call these "combined" policies. Later we will specify the optimal implementation decision and restrict our focus from combined policies to measurement-only policies by specifying that, after all measurements have been made, the optimal implementation decision rule will be used. Ultimately we will see that the optimal implementation decision is one of pure exploitation and the value of this exploitive implementation decision will determine the objective function for our measurement problem.

Let the set of combined policies $\tilde{\Pi}$ be the class of policies $\tilde{\pi}$ that specify decision functions $X^{\tilde{\pi},n} : \mathbb{S}^{n+1} \to \{1 \dots M\}$ for n up to and *including* the terminal time N, as in

$$\tilde{\Pi} := \left\{ \tilde{\pi} = (X^{\tilde{\pi},0}, \dots X^{\tilde{\pi},N}) \mid X^{\tilde{\pi},n} : \mathbb{S}^{n+1} \to \{1 \dots M\} \right\}.$$

In this notation, $x^n = X^{\tilde{\pi},n}(S^0, \dots S^n)$ under policy $\tilde{\pi}$ and $\mathbb{S}^{n+1}$ is the cross-product of $n+1$ orthogonal state spaces, one for the state at each time from 0 up to and including n. Taking the domain of the decision function as this cross-product of state spaces allows for non-Markovian policies.

Any policy $\tilde{\pi} \in \tilde{\Pi}$ proceeds by making a measurement, receiving a measurement value, calculating the posterior distribution of Y using this measurement, making a new measurement, and repeating until all N measurements have been made. After the final measurement, the policy $\tilde{\pi}$ provides an implementation decision $x^N \in \mathcal{F}^N$. The alternative x^N is chosen and the policy receives a terminal reward $\hat{y}^{N+1} = Y_{x^N} + \varepsilon^{N+1}$. At this point we have the option of generalizing the problem by introducing a concave, increasing utility function and maximizing the expected value of the utility of the reward. This would induce risk aversion in the optimal policy. Instead, we simply seek to maximize over all policies the expected value of the reward itself. Our problem can be written,

$$\sup_{\tilde{\pi} \in \tilde{\Pi}} \mathbb{E}^{\tilde{\pi}} \left[\hat{y}^{N+1} \right], \qquad (8)$$

where the notation $\mathbb{E}^{\tilde{\pi}}$ means the expectation with the policy fixed to $\tilde{\pi}$. The unbiasedness of ε^{N+1} implies $\mathbb{E}^{\tilde{\pi}}\left[\hat{y}^{N+1}\right] = \mathbb{E}^{\tilde{\pi}}\left[Y_{x^N} + \varepsilon^{N+1}\right] = \mathbb{E}^{\tilde{\pi}}\left[Y_{x^N}\right]$, so (8) can be rewritten as

$$\sup_{\tilde{\pi} \in \tilde{\Pi}} \mathbb{E}^{\tilde{\pi}} \left[\hat{y}^{N+1} \right] = \sup_{\tilde{\pi} \in \tilde{\Pi}} \mathbb{E}^{\tilde{\pi}} \left[Y_{x^N} \right]. \qquad (9)$$

Then, let us consider any combined policy $\tilde{\pi} \in \tilde{\Pi}$ as the explicit combination of a measurement policy and an implementation decision. Define the set of measurement policies Π as the set of policies that specify measurement decision functions $X^{\pi,n} : \mathbb{S}^{n+1} \to \{1 \dots M\}$ for $0 \le n < N$ but leave X^N unspecified, as in

$$\Pi := \left\{ \pi = (X^{\pi,0}, \dots X^{\pi,N-1}) \mid X^{\pi,n} : \mathbb{S}^{n+1} \to \{1 \dots M\} \right\}.$$

From this definition, we see that any combined policy $\tilde{\pi} \in \tilde{\Pi}$ can be written as $\tilde{\pi} = (\pi, X^N) = (X^{\pi,0} \dots X^{\pi,N-1}, X^N)$ for some $\pi \in \Pi$ and some $X^N : \mathbb{S}^{N+1} \to \{1 \dots M\}$. Using this we rewrite (9) as

$$\begin{aligned}\sup_{\tilde{\pi} \in \tilde{\Pi}} \mathbb{E}\left[Y_{x^N}\right] &= \sup_{\pi \in \Pi} \; \sup_{X^N} \mathbb{E}^{\pi} \left[Y_{X^N(S^0 \dots S^N)} \right] \\ &= \sup_{\pi \in \Pi} \; \sup_{X^N} \mathbb{E}^{\pi} \left[\mathbb{E}_N \left[Y_{X^N(S^0 \dots S^N)} \right] \right] \\ &= \sup_{\pi \in \Pi} \; \sup_{X^N} \mathbb{E}^{\pi} \left[\mu^N_{X^N(S^0 \dots S^N)} \right],\end{aligned}$$

since Y_x has mean μ_x^N given $\mathcal{F}^N$. When the objective function is written in this way, we find the optimal choice for X^N,

$$X^{*N}(S^0, \dots S^N) = \arg\max_{x \in \{1 \dots M\}} \mu_x^N,$$

which constitutes a pure exploitation strategy at implementation time. This reduces our problem to finding the optimal measurement policy $\pi \in \Pi$ by solving

$$\sup_{\pi \in \Pi} \mathbb{E}^{\pi} \left[\max_x \mu_x^N \right]. \qquad (10)$$

Then, if we can find an optimal solution π^* to the measurement problem (10), the combined policy $\tilde{\pi}^* = (\pi^*, X^{N*})$ is optimal for the combined problem (8).

C. Dynamic Programming

As just shown, solving the simpler measurement-only problem (10) provides an immediate solution to the combined problem (8). We therefore focus our effort on the measurement-only problem. We apply a dynamic programming approach. In this approach, the value function is defined as the value of the optimal policy given a particular state S^n, and may also be determined recursively through Bellman's equation. If the value function can be computed efficiently, the optimal policy may then also be computed from it. Although in this problem the "curse of dimensionality" makes direct computation of the value function difficult even for M as small as 3, the dynamic programming principle still provides a valuable method for studying the problem.

The terminal value function, $V^N : \mathbb{S} \to \mathbb{R}$, is given by (10),

$$V^N(S^N) := \max_{x\in\{1\dots M\}} \mu_x^N.$$

The dynamic programming principle tells us that the value function at any other time $0 \le n < N$ is given recursively by

$$V^n(S^n) := \max_x \mathbb{E}_n \left[V^{n+1}(S^M(S^n, x, Z^{n+1}))\right].$$

We define the Q-factors, $Q^n : \mathbb{S} \times \{1 \dots M\} \to \mathbb{R}$, as

$$Q^n(S^n, x) := \mathbb{E}_n \left[V^{n+1}(S^M(S^n, x, Z^{n+1}))\right], \qquad (11)$$

and the dynamic programming principle tells us that the policy choosing its measurement decisions via

$$X^{*n}(S^n) := \operatorname*{arg\,max}_{x\in\{1\dots M\}} Q^n(S^n, x) \qquad (12)$$

is optimal. Finally, we define the value of a measurement policy $\pi \in \Pi$ as

$$V^{n,\pi}(S^n) := \mathbb{E}_n^\pi \left[V^N(S^N)\right].$$

III. The Knowledge Gradient Policy

In the problem discussed so far, we supposed that the entire reward was received after the final measurement. Instead, we may formulate an equivalent problem in which the reward is given in pieces over time. We define the knowledge gradient policy as that policy which maximizes the single period reward under this alternate formulation.

A. Definition

The problem given by (10) has a terminal reward $V^N(S^N) := \max_x \mu_x^N$, but no rewards at any other times. We restructure these rewards by writing $V^N(S^N)$ as a telescoping sequence, $V^N(S^N) = \left(V^N(S^N) - V^N(S^{N-1})\right) + \ldots + \left(V^N(S^{n+1}) - V^N(S^n)\right) + V^N(S^n)$. Thus the problem that provides single period reward $V^N(S^k) - V^N(S^{k-1})1_{\{k>n\}}$ at times $k = n, n+1, \ldots N$ is equivalent to problem (10) because the total reward provided is the same. The knowledge gradient policy π^{KG} is defined as the policy that chooses its measurements to maximize the expectation of the single period reward provided under this restructured formulation. The knowledge gradient policy has decision function

$$X^{KG,n}(S^n) := \operatorname*{arg\,max}_{x\in\{1\dots M\}} \mathbb{E}_n \left[V^N(S^M(S^n, x, Z^{n+1}) - V^N(S^n)\right]. \qquad (13)$$

Since $V^N(S^n)$ is measurable with respect to $\mathcal{F}^n$ and does not depend on the quantity x which the $\arg\max$ varies, the knowledge gradient policy's decision function may be rewritten as

$$\begin{aligned} X^{KG,n}(S^n) &= \operatorname*{arg\,max}_{x\in\{1\dots M\}} \mathbb{E}_n \left[V^N(S^M(S^n, x, Z^{n+1})\right] \\ &= \operatorname*{arg\,max}_{x\in\{1\dots M\}} Q^{N-1}(S^n, x). \qquad (14) \end{aligned}$$

Note that the knowledge gradient policy is optimal when $N = 1$ by (12) and (14).

Only the decision function's argument, S^n, depends on n while the decision function itself, $\arg\max_x Q^{N-1}(\cdot, x)$, does not. Thus the knowledge gradient policy is stationary in time, and we drop the time index n when we write X^{KG}.

If we think of $V^N(S^n)$ as a measure of the amount of "knowledge" contained in the state S^n, we see from (13) that the knowledge gradient policy chooses its decisions in the direction of steepest expected ascent of this metric. This is the reason for the name *knowledge gradient*.

B. Computation

We can compute an analytical and computationally tractable expression for X^{KG}. For each $x \in \{1, \ldots M\}$ define a function $\zeta_x : \mathbb{S} \to \overline{\mathbb{R}}_+$ by,

$$\zeta_x(\mu, \beta) := -\left|\frac{\mu_x - \max_{x' \ne x} \mu_{x'}}{\tilde{\sigma}(\beta_x)}\right|, \qquad (15)$$

and define $\zeta_x^n := \zeta_x(S^n)$. Except for the sign, ζ_x^n is the variance adjusted minimum distance that a measurement of alternative x must alter μ_x^{n+1} from its pre-measurement value of μ_x^n to make $\arg\max_{x'} \mu_{x'}^{n+1} \ne \arg\max_{x'} \mu_{x'}^n$ — that is, to make $\mathbb{P}$ conditioned on $\mathcal{F}^n$ disagree with $\mathbb{P}$ conditioned on $\mathcal{F}^{n+1}$ about which alternative has the largest expected value.

Theorem:

$$X^{KG}(S^n) = \operatorname*{arg\,max}_{x\in\{1,\ldots M\}} \tilde{\sigma}(\beta_x^n) \left[\zeta_x^n \Phi(\zeta_x^n) + \varphi(\zeta_x^n)\right], \qquad (16)$$

where Φ is the normal cdf and φ is the normal pdf.

Proof: From (14) we see that we may compute $X^{KG}(S^n)$ by computing the Q-factors $Q^{N-1}(S^n, x)$ for each action x. Using the definition of the Q-factors (11), we have for a fixed state S and a generic standard normal random variable Z,

$$\begin{aligned} Q^{N-1}(S, x) &:= \mathbb{E}\left[V^N(S^M(S, x, Z)\right] \\ &= \mathbb{E}\left[(\mu_x + \tilde{\sigma}(\beta_x) Z) \vee \max_{x' \ne x} \mu_{x'}\right]. \qquad (17) \end{aligned}$$

This expectation is the expectation of the maximum of a constant and a normal random variable. Let $a \in \mathbb{R}$ be an arbitrary constant and $W \sim \mathcal{N}(b, c^2)$ an arbitrary normal random variable. Then, [13] tells us

$$\mathbb{E}\left[W \vee a\right] = a\Phi\left(\frac{a-b}{c}\right) + b\Phi\left(\frac{b-a}{c}\right) + c\varphi\left(\frac{a-b}{c}\right). \tag{18}$$

Fix x and consider two cases. First, consider the case that $\mu_x > \max_{x'} \mu_{x'}$. This is the case in which we measure the alternative that is uniquely best according to the prior. We rewrite (18) as

$$\begin{aligned}
&\mathbb{E}\left[W \vee a\right] \\
&= a\Phi\left(\frac{a-b}{c}\right) + b\left(1 - \Phi\left(\frac{a-b}{c}\right)\right) + c\varphi\left(\frac{a-b}{c}\right) \\
&= b + (a-b)\Phi\left(\frac{a-b}{c}\right) + c\varphi\left(\frac{a-b}{c}\right) \\
&= b + c\left[\left(\frac{a-b}{c}\right)\Phi\left(\frac{a-b}{c}\right) + \varphi\left(\frac{a-b}{c}\right)\right].
\end{aligned}$$

In the case we are considering, $\mu_x - \max_{x' \neq x} \mu_{x'}$ is positive and $(\max_{x' \neq x} - \mu_x)/\tilde{\sigma}(\beta_x) = \zeta_x$. Compare this expression with $(a-b)/c$ and write (17) as

$$Q^{N-1}(S, x) = \mu_x + \tilde{\sigma}(\beta_x)\left[\zeta_x \Phi(\zeta_x) + \varphi(\zeta_x)\right],$$

which can be rewritten in this case using $\mu_x = \max_{x'} \mu_{x'}$ as

$$Q^{N-1}(S, x) = \max_{x'} \mu_{x'} + \tilde{\sigma}(\beta_x)\left[\zeta_x \Phi(\zeta_x) + \varphi(\zeta_x)\right].$$

Now consider the case that $\mu_x \leq \max_{x'} \mu_{x'}$. We rewrite (18) again using the substitution $\Phi(-z) = 1 - \Phi(z)$, and also using the symmetric property of the normal pdf, $\varphi(-z) = \varphi(z)$, as

$$\mathbb{E}\left[W \vee a\right] = a + c\left[\left(\frac{b-a}{c}\right)\Phi\left(\frac{b-a}{c}\right) + \varphi\left(\frac{b-a}{c}\right)\right].$$

In the case we are considering, $\mu_x - \max_{x' \neq x} \mu_{x'} \leq 0$ and $(\mu_x - \max_{x' \neq x})/\tilde{\sigma}(\beta_x) = \zeta_x$. Compare this expression with $(b-a)/c$ and write (17) as

$$Q^{N-1}(S, x) = \max_{x' \neq x} \mu_{x'} + \tilde{\sigma}(\beta_x)\left[\zeta_x \Phi(\zeta_x) + \varphi(\zeta_x)\right],$$

which can be rewritten in our case using $\max_{x' \neq x} \mu_{x'} = \max_{x'} \mu_{x'}$ as

$$Q^{N-1}(S, x) = \max_{x'} \mu_{x'} + \tilde{\sigma}(\beta_x)\left[\zeta_x \Phi(\zeta_x) + \varphi(\zeta_x)\right].$$

In both cases the expression for $Q^{N-1}(S, x)$ is the same, and we rewrite (14) as

$$\begin{aligned}
X^{KG}(S^n) &= \operatorname*{arg\,max}_{x \in \{1, \ldots M\}} \max_{x'} \mu^n_{x'} + \tilde{\sigma}(\beta^n_x)\left[\zeta^n_x \Phi(\zeta^n_x) + \varphi(\zeta^n_x)\right] \\
&= \operatorname*{arg\,max}_{x \in \{1, \ldots M\}} \tilde{\sigma}(\beta^n_x)\left[\zeta^n_x \Phi(\zeta^n_x) + \varphi(\zeta^n_x)\right],
\end{aligned}$$

since $\max_{x'} \mu^n_{x'}$ does not depend on x. ■

Computation of the knowledge gradient policy via (16) scales linearly with the number of alternatives M. This compares well with other offline learning policies. To compute the knowledge gradient policy's decision at time n, we must first find the largest and second largest μ^n_x across all alternatives x, which will be used to compute ζ^n_x. This may be implemented either by an initial pass through the alternatives at each time period, or by storing and updating the two values across time periods. Once we have the largest and second largest μ^n_x, we iterate through the alternatives, calculating $\tilde{\sigma}(\beta^n_x)\left[\zeta^n_x \Phi(\zeta^n_x) + \varphi(\zeta^n_x)\right]$ for each one, and return the alternative with the largest value for this expression. This iteration may be streamlined by recomputing the expression only for those alternatives that changed ζ^n_x or β^n_x from the previous iteration.

C. Exploration vs. Exploitation

The knowledge gradient policy balances two considerations when it chooses its measurement decisions. First, it prefers to measure those alternatives about which comparatively little is known. These alternatives are the ones with large variance Σ^n_{xx} or equivalently with small inverse variance β^n_x. Second, the knowledge gradient policy prefers to measure alternatives x with $|\mu^n_x - \max_{x' \neq x} \mu^n_{x'}|$ close to 0. We call $-|\mu^n_x - \max_{x' \neq x} \mu^n_{x'}|$ the *influence* of alternative x. Similarly, we call ζ^n_x the *normalized influence* of alternative x because it is the influence normalized by $\tilde{\sigma}(\beta^n_x)$. Measurements of alternatives with large influence are more likely to cause a change in the optimal implementation decision; that is, to cause $\arg\max_{x'} \mu^n_{x'} \neq \arg\max_{x'} \mu^{n+1}_{x'}$.

We can explicitly see these effects by computing derivatives of the two terms in (16). First consider the effect of increasing the influence, or, equivalently, decreasing $|\mu^n_x - \max_{x' \neq x} \mu^n_{x'}|$. This affects only the second term, $\zeta^n_x \Phi(\zeta^n_x) + \varphi(\zeta^n_x)$. By (15), ζ^n_x increases as we decrease $|\mu^n_x - \max_{x' \neq x} \mu^n_{x'}|$, and as ζ^n_x increases so does the entire second term because,

$$\frac{d}{dz}\left[z\Phi(z) + \varphi(z)\right] = \Phi(z) + z\varphi(z) - z\varphi(z) = \Phi(z) \geq 0.$$

Now consider the effect of decreasing β^n_x. The first term, $\tilde{\sigma}(\beta_x)$, increases because, from (4),

$$\frac{d}{d\beta^n_x}\tilde{\sigma}(\beta^n_x) = -\frac{1}{2}(1 + 2\beta^n_x \sigma^2)\left[\beta^n_x(1 + \beta^n_x \sigma^2)\right]^{-3/2} \leq 0.$$

Also, we see from (15) that as $\tilde{\sigma}(\beta^n_x)$ increases so does ζ^n_x, and we just saw that as ζ^n_x increases so does the entire second term.

Compare the classic tradeoff of exploration against exploitation with the knowledge gradient policy's tradeoff of variance against influence. The benefits of variance are exactly the benefits of exploration. Exploration pushes us to learn about things which we do not already know; if we already know something perfectly, there is little point in measuring it further.

The parallel between the exploitation strategy and the knowledge gradient policy's desire to measure influential alternatives is more subtle. Often, considerations of exploitation and maximizing influence agree on which measurement is best. For example, if alternative x is neither the best nor the second best alternative at time n, that is if there are two distinct alternatives i and j such that $\mu^n_x < \mu^n_i$ and $\mu^n_x < \mu^n_j$, then increasing μ^n_x by a small amount while holding all other parameters constant increases the influence of alternative x but does not the change the influence of any other alternative. Thus increasing μ_x^n makes the knowledge gradient policy more willing to measure alternative x. This is the same behavior advocated by the exploitation strategy. However, if $\mu_x^n = \max_{x'} \mu_{x'}^n$, then increasing μ_x^n decreases the influence of all alternatives, which, depending on the relative values of the inverse variances, may make the knowledge gradient policy *less* likely to measure alternative x.

The difference between exploitation and influence maximization stems from the different problem settings. Exploitation is appropriate for online learning while influence maximization is appropriate for offline learning. In online learning, a policy is given an immediate reward at each stage based on the true value of the alternative chosen. This explicitly discourages the policy from choosing poor alternatives. In offline learning, the only reward is at the end, so there is no explicit penalty for measuring poor alternatives. Instead, choosing poor alternatives is discouraged by an opportunity cost. If an alternative's estimated value is so poor that its true value is almost certainly suboptimal, then there is little value in measuring it. If we know that we will not be implementing a particular alternative, then we already know everything we need to know about it. Offline learning only has value if it can change the implementation decision.

D. *Summary of Theoretical Results*

It can be shown that the knowledge gradient policy is optimal in the following special cases [14]. First, the knowledge gradient policy is optimal by construction when we only have one measurement to make, i.e. when $N = 1$. Second, the knowledge gradient policy is asymptotically optimal in the limit as the number of measurements N becomes arbitrarily large. This property is due to the fact that, as N goes to infinity, the knowledge gradient policy samples every alternative infinitely often. Third, the knowledge gradient policy is optimal when there are only two alternatives to measure, i.e. when $M = 2$. Fourth, the knowledge gradient policy is optimal when the measurements are free from noise and the components of the time 0 prior are ordered by $\mu_1^0 \geq \mu_2^0 \geq \ldots \mu_M^0$ and $\Sigma_{11}^0 \geq \Sigma_{22}^0 \geq \ldots \Sigma_{MM}^0$.

A knowledge gradient policy is also optimal in other offline learning problems. For example, the game of twenty questions may be formulated as an offline learning problem [15] in which the terminal payoff is 1 if the final guess is correct and 0 otherwise. In this game we assume that the answers to our questions are correct (i.e. no measurement noise), and that the prior distribution over the space of objects to be guessed is uniform. The optimal policy is bisection, in which each question eliminates half of the possible objects. This bisection policy is also the knowledge gradient policy according to a reformulation using a telescoping sequence similar to the one we use in section III-A.

These theoretical results demonstrate that the knowledge gradient policy performs well in at least some problem settings, and they lead us to hypothesize that the knowledge gradient policy may perform well across a broader range of problem settings than just those for which optimality may be proven theoretically.

IV. Experimental Results

We compare the knowledge gradient policy using Monte Carlo simulation against the following policies:

Interval Estimation: Interval estimation [16] chooses its measurements by computing for each x the upper bound of a symmetric confidence interval for the true value of alternative x. It then measures the alternative with the largest upper bound according to $X(S^n) = \arg\max_x \mu_x^n + (\beta_x^n)^{-1/2} z_{\alpha/2}$, where $z_{\alpha/2}$ is the solution to $\Phi(z) = \alpha/2$. Interval estimation is parameterized by the confidence level α, or equivalently by $z_{\alpha/2}$.

Boltzmann Exploration: Under Boltzmann exploration, the probability of measuring an alternative x is proportional to a function of the expected value of alternative x and the current "temperature", $\mathbb{P}_n \{X(S^n) = x\} \propto \exp(\mu_x^n / T^n)$, where the policy is parameterized by the choice of a decreasing sequence of temperatures, $T = (T^n)_{n=0}^{N-1}$.

Gittins Index: The Gittins index policy chooses decisions by $X(S^n) = \arg\max_x \mu_x^n + (\beta_x^n)^{-1/2} \nu(0, \sigma^2 \beta_x^n, 1, \alpha)$, where values for ν may be found in [9]. This policy is provably optimal for an online discounted infinite horizon version of our problem with discount factor α. In the online problem, the discount factor α is specified by the objective function, but in our offline undiscounted problem α is a free parameter of the policy.

Pure Exploration: The pure exploration strategy chooses its measurement randomly among the alternatives according to a uniform distribution: $\mathbb{P}_n \{X(S^n) = x\} = 1/M$.

Pure Exploitation: The pure exploitation strategy always chooses the alternative with the largest expected value: $X(S^n) = \arg\max_x \mu_x^n$.

Simulations were performed in which true function values were generated according to the prior, a policy was simulated, and the contribution achieved by the policy was collected. The policy in question determined the measurement decisions, but the optimal implementation decision was always used at the final time. Many samples were collected and averaged to estimate the value of the policy in each problem setting. In each sample simulation, the same true function values were used to simulate each policy to reduce variance. Sample variances were estimated for each data point and used to estimate the standard deviation of our estimate of the expected value of the policy. These standard deviations are pictured as error bars.

The space of problem settings has many dimensions: Number of measurements N; number of alternatives M; initial mean $\mu^0 \in \mathbb{R}^M$; initial inverse variance $\beta^0 \in \overline{\mathbb{R}}_+^M$; and measurement noise $\sigma^2 \in \mathbb{R}_+$. This space is too large to allow numerical investigation of every scenario, so we restrict our investigations by focusing on the following scenarios: fixing N constant while varying M (figure 1); fixing M constant while varying N (figure 2); and holding the ratio of N/M

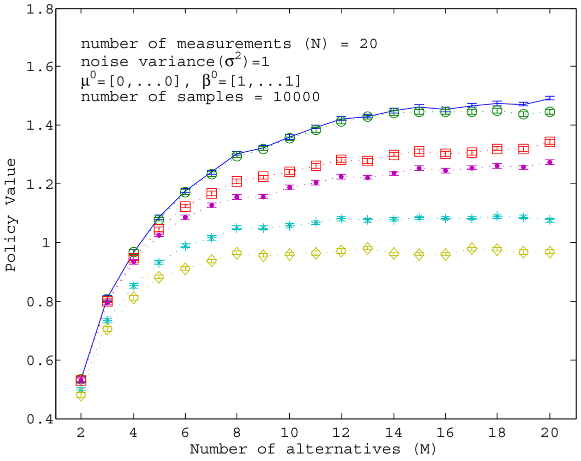

Fig. 1. Varying M with N constant and a homegenous prior.

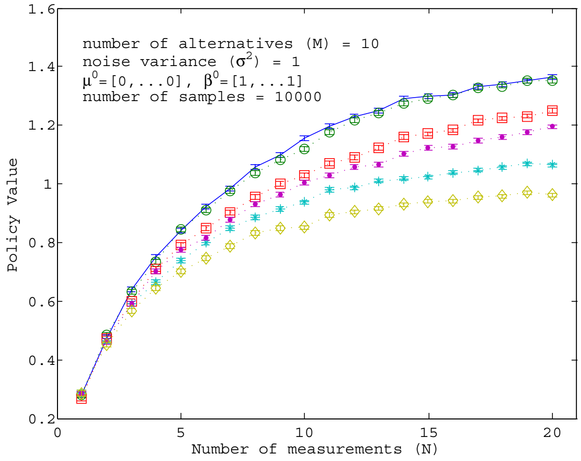

Fig. 2. Varying N with M constant and a homegenous prior.

constant while varying both of them (figure 3). In all cases we use a measurement noise (σ^2) equal to 1. In the first two scenarios the prior is homogenous with $\mu^0 = [0, \dots 0]$ and $\beta^0 = [1, \dots 1]$. In the third case, the prior is nonhomogenous with μ^0_1 somewhat larger than μ^0_x and β^0_1 much larger than β^0_x for $x \neq 1$. This nonhomegenous prior corresponds to a case in which we have one existing alternative which has been tried often and found to perform well, and several untried alternatives of which none are expected to perform as well.

Interval estimation, Boltzmann exploration, and Gittins index all have tuneable parameters. No single parameter was best across all problem settings sampled, but a few parameter choices were consistently better than others. For each policy with a parameter to tune, we simulated a representative set of problems with several parameter choices and chose the parameter that performed the best across this set of problems. We subsequently used this parameter whenever simulating the policy. We ultimately chose a constant temperature of 1 for Boltzmann exploration, a discount factor of 0.7 for the Gittins index policy, and $z_{\alpha/2} = 2.5$ for interval estimation. This choice of $z_{\alpha/2}$ is consistent with [16], which ran a different set of calibrating experiments and found that "the interval estimation algorithm performs best in all of these problems with a $z_{\alpha/2}$ value between 2 and 3". Although we only picture each policy with one choice of parameter, and in some cases another parameter choice made a policy perform better in that problem setting, in no case did any policy with any choice of parameter outperform the knowledge gradient policy.

In the experiment pictured in figure 1 we hold N constant at 20 while varying M from 2 to 20. Knowledge gradient and interval estimation perform best, and identically so, followed by Boltzmann exploration and uniform exploration. Uniform exploration performs as well as it does because in the early iterations the homogeneous prior does not distinguish between alternatives. In later iterations, however, some alternatives do distinguish themselves as better, reducing the effectiveness of uniform exploration. The Gittins index policy performs poorly because it places too large an emphasis on exploitation.

Note that once the number of alternatives M grows larger than the number of measurements N, the values of the knowledge gradient, interval estimation, and Gittins index policies remain constant due to the homogeneous prior. No matter how a policy distributes the N measurements, the at least $N - M$ alternatives that remain unmeasured at implementation time will retain their initial priors. Under the homogenous prior, these are all normal with mean 0 and variance 1 so that $\mu^N_x = 0$ for all $x \in U$, where U is this set of unmeasured alternatives, and the terminal value function may be written as $V^N(\mu^N, \beta^N) = \max_x \mu^N_x = \left(\max_{x \in U} \mu^N_x\right) \vee \left(\max_{x \notin U} \mu^N_x\right) = 0 \vee \max_{x \notin U} \mu^N_x$, which is distributed identically for all $M > N$ under these policies. The value of the Boltzmann exploration and pure exploration policies will change, however, because they will be less likely to sample an alternative twice.

In the experiment pictured in figure 2 we hold M constant at 10 while varying N. We see a similar situation to figure 1, in which the knowledge gradient policy and interval estimation perform best, followed by Boltzmann exploration, with uniform exploration again performing better than expected due to the homogeneous prior.

As N grows much larger than M, all policies that do at least some exploration, including the knowledge gradient, Boltzmann exploration, and pure exploration policies but excluding pure exploitation and interval estimation, will sample every alternative often enough to obtain accurate estimates of their true values. The value of any such policy grows toward the value of learning the value of every alternative exactly before making an implementation decision. That is for any policy π sampling every alternative infinitely often, $\lim_{N \to \infty} V^{0,\pi}(\mu^0, \beta^0) = \mathbb{E}\left[\max_x Y_x\right]$, where $Y_x \sim \mathcal{N}(\mu^0_x, (\beta^0_x)^{-1})$.

In the experiment pictured in figure 3, we vary both N and M while holding their ratio constant. In addition, the true value of alternative 1 is known almost exactly by the prior, and this true value is larger than the expected value under the prior of

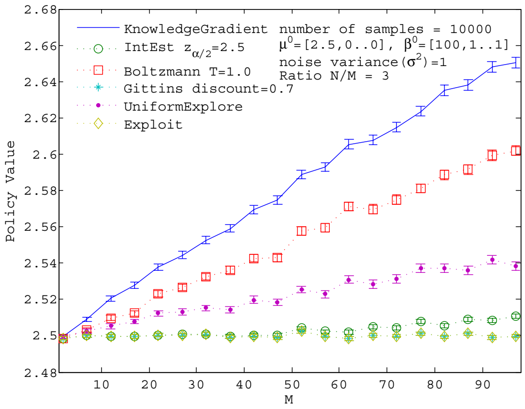

Fig. 3. Varying both M and N with a constant ratio between them, and with a non-homegenous prior.

each of the other alternatives. This special case illustrates one important aspect of the comparison between the knowledge gradient and interval estimation policies. The knowledge gradient policy measures every alternative infinitely often given enough measurements [14] while interval estimation may become stuck measuring one alternative repeatedly. In this experiment, interval estimation chooses to measure alternative 1 initially because its mean under the prior is large enough to offset its low variance. Almost no additional information is gained by this measurement because the variance of alternative 1 was so initially so low, and the posterior is nearly identical to the prior. Thus, on subsequent measurements the interval estimation algorithm will continue measuring alternative 1, and as a result learns very little.

In contrast, the knowledge gradient policy penalizes low variance more heavily and it chooses not to measure alternative 1, measuring another alternative with larger variance instead. Frequently, because the mean of alternative 1 is largest under the prior, the measurement reveals that the other alternative is worse than alternative 1, and nothing is gained from its measurement. On occasion, however, the measurement reveals that another alternative has larger true value, and the knowledge gradient realizes this additional reward. Indeed, if the prior is fixed to $\mu^0 = [z_{\alpha/2}, 0, \dots 0]$, $\beta^0 = [\infty, 1, \dots 1]$, the additional value that the knowledge gradient policy achieves over the interval estimation policy becomes arbitrarily large in the limit as both N and M become large.

V. Conclusion

We formulated an offline learning problem and defined a new type of policy for this problem, the knowledge gradient policy. We showed how to compute the knowledge gradient policy efficiently and compared its decision making process to the classic exploration vs. exploitation tradeoff. Using Monte Carlo simulation we compared the knowledge gradient policy to interval estimation, Boltzmann exploration, Gittins index, pure exploration and pure exploitation policies. The knowledge gradient policy performed as well or better than these other policies in all problem situations simulated, and it should be considered for use in offline learning applications because of its ease of use and rapid learning rate.

References

[1] G. Box, W. Hunter, and J. Hunter, *Statistics for Experimenters: An Introduction to Design, Data Analysis, and Model Building*. New York: John Wiley & Sons, 1978.

[2] R. Myers and D. Montgomery, *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. New York: John Wiley & Sons, 2002.

[3] R. Bechhofer, J. Kiefer, and M. Sobel, *Sequential Identification and Ranking Procedures*. Chicago: University of Chicago Press, 1968.

[4] R. Bechhofer, T. Santner, and D. Goldsman, *Design and Analysis of Experiments for Statistical Selection, Screening and Multiple Comparisons*. New York: J.Wiley & Sons, 1995.

[5] E. Paulson, "A sequential procedure for selecting the population with the largest mean from k normal populations," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 174–180, March 1964.

[6] Hartmann, "An improvement on paulsons procedure for selecting the population with the largest mean from k normal populations with a common unknown variance," *Sequential Analysis*, vol. 10, no. 1-2, pp. 1–16, 1991.

[7] H. Robbins, "Some aspects of the sequential design of experiments," *Bulletin of the American Mathematical Society*, vol. 58, pp. 527–535, 1952.

[8] J. C. Gittins and D. M. Jones, "A dynamic allocation index for the sequential design of experiments," in *Progress in Statistics*, J. Gani, Ed., 1974, pp. 241–266.

[9] J. Gittins, *Multi-Armed Bandit Allocation Indices*. New York: John Wiley and Sons, 1989.

[10] J. Nino-Mora, "Restless bandits, partial conservation laws and indexability," *Advances in Applied Probability*, vol. 33, no. 1, pp. 76–98, Mar. 2001.

[11] R. Sutton and A. Barto, *Reinforcement Learning*. Cambridge, Massachusetts: The MIT Press, 1998.

[12] I. Witten, "The apparent conflict between estimation and control-a survey of the two-armed bandit problem," *Journal of the Franklin Institute*, vol. 301, pp. 161–189, 1976.

[13] C. Clark, "The greatest of a finite set of random variables," *Operations Research*, vol. 9, pp. 145–163, 1971.

[14] P. Frazier and W. Powell, "A knowledge gradient policy for sequential information collection," 2007, working paper.

[15] T. Cover and J. Thomas, *Elements of Information Theory*. New York: John Wiley, 1991.

[16] L. Kaelbling, *Learning in embedded systems*. Cambridge, MA: MIT Press, 1993.