On Greedy Adaptive Measurements

Entao Liu, Member, IEEE and Edwin K. P. Chong, Fellow, IEEE

Abstract—The purpose of this article is to examine the greedy adaptive measurement policy in the context of a linear Gaussian measurement model with an optimization criterion based on information gain. In the special case of sequential scalar measurements, we provide sufficient conditions under which the greedy policy actually is optimal in the sense of maximizing the net information gain. In the general setting, we also discuss cases where the greedy policy is not optimal.

Index Terms—entropy, information gain, compressive sensing, compressed sensing, greedy policy, optimal policy.

I. INTRODUCTION

Consider a signal of interest \( x \), which is a random vector taking values in \( \mathbb{R}^N \) with (prior) distribution \( N(\mu, \mathbf{P}_0) \) (i.e., Gaussian distribution with mean \( \mu \) and \( N \times N \) covariance matrix \( \mathbf{P}_0 \)). We wish to estimate \( x \) based on \( M \) measurements of it (where \( M \) is specified upfront). The \( k \)th measurement \((k \in \{1, \ldots, M\})\) is given by

\[
y_k = \mathbf{A}_k x + \mathbf{w}_k,
\]

where \( y_k \) takes values in \( \mathbb{R}^L \), and the noise \( \mathbf{w}_k \) has distribution \( N(0, \mathbf{R}) \), independent over \( k \). The measurement matrix \( \mathbf{A}_k \) is \( L \times N \).

Consider the following adaptive (sequential) measurement problem. For each \( k \in \{1, \ldots, M\} \), we are allowed to choose the measurement matrix \( \mathbf{A}_k \) from a prespecified set \( \mathcal{A} \). Moreover, our choice is allowed to depend on the entire history of measurements up to that point: \( \mathcal{I}_{k-1} = \{y_1, \ldots, y_{k-1}\} \).

Let the posterior distribution of \( x \) given \( \mathcal{I}_k \) be \( \mathcal{N}(x_k, \mathbf{P}_k) \). More specifically, \( \mathbf{P}_k \) can be written recursively for \( k = 1, \ldots, M \) as

\[
\mathbf{P}_k = \left( I - \mathbf{P}_{k-1} \mathbf{A}_k^T (\mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^T + \mathbf{R})^{-1} \mathbf{A}_k \right) \mathbf{P}_{k-1}.
\]

If this expression seems a little unwieldy, a simpler version is as follows:

\[
\mathbf{P}_k = \left( \mathbf{P}_{k-1}^{-1} + \mathbf{A}_k^T \mathbf{R}^{-1} \mathbf{A}_k \right)^{-1},
\]

assuming that \( \mathbf{P}_{k-1} \) and \( \mathbf{R} \) are nonsingular. Also define the entropy of the posterior distribution of \( x \) given \( \mathcal{I}_k \):

\[
H_k = \frac{1}{2} \log \det(\mathbf{P}_k) + \frac{1}{2} N(1 + \log(2\pi)).
\]

We focus on a common information-theoretic criterion for choosing the measurement matrices: For the \( k \)th decision, we pick \( \mathbf{A}_k \) to maximize the per-stage information gain, defined as \( H_{k-1} - H_k \). For reasons that will be made clear later, we refer to this strategy as the greedy policy. The term policy simply refers to a rule for picking \( \mathbf{A}_k \) for each \( k \) based on \( \mathcal{I}_{k-1} \).

Suppose that the overall goal is to maximize the net information gain, defined as \( H_0 - H_M \). We ask the following questions: Does the greedy policy achieve this goal? If not, then what policy achieves it? How much better is such a policy than the greedy one? Are there cases where the greedy policy does achieve this goal? In Section II we analyze the greedy policy and compute its net information gain. In Section III to find the net information gain of the optimal policy, we introduce a relaxed optimization problem, which can be solved as a water-filling problem. In Section IV we derive two sufficient conditions under which the greedy policy is optimal. In Section V we give examples for which the greedy policy is not optimal. We also show that the greedy policy can be arbitrarily worse than the optimal policy.

II. GREEDY POLICY

A. Preliminaries

We now explore how the greedy policy performs in the adaptive measurement problem. Before proceeding, we first make some remarks on the information gain criterion:

- Information gain as defined in this paper also goes by the name mutual information (of \( x \) and \( y_k \) in the case of per-stage information gain, and of \( x \) and \( (y_1, \ldots, y_M) \) in the case of net information gain).
- The net information gain can be written as the cumulative sum of the per-stage information gains:

\[
H_0 - H_M = \sum_{k=1}^{M}(H_{k-1} - H_k).
\]

This is why the greedy policy is named as such; at each stage \( k \), the greedy policy simply maximizes the immediate (short-term) contribution \( H_{k-1} - H_k \) to the overall cumulative sum.

- Using the formulas (2) and (4) for \( H_k \) and \( \mathbf{P}_k \), we can write

\[
H_{k-1} - H_k = -\frac{1}{2} \log \det(I - \mathbf{P}_{k-1} \mathbf{A}_k^T (\mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^T + \mathbf{R})^{-1} \mathbf{A}_k).
\]

In other words, at the \( k \)th stage, the greedy policy minimizes (with respect to \( \mathbf{A}_k \))

\[
\log \det \left( I - \mathbf{P}_{k-1} \mathbf{A}_k^T (\mathbf{A}_k \mathbf{P}_{k-1} \mathbf{A}_k^T + \mathbf{R})^{-1} \mathbf{A}_k \right).
\]

- Equivalently, using the other formula (3) for \( \mathbf{P}_k \), the greedy policy maximizes

\[
\log \det \left( \mathbf{P}_{k-1}^{-1} + \mathbf{A}_k^T \mathbf{R}^{-1} \mathbf{A}_k \right)
\]
in every stage. For the purpose of optimization, the log function in the objective functions above can be dropped.

Notice that the greedy policy does not use the values of \( y_1, \ldots, y_{k-1} \); its choice of \( A_k \) depends only on \( P_{k-1} \) and \( R \). In fact, the formulas above show that information gain is a deterministic function of the of measurement matrices (in our particular setup). This implies that the optimal policy can be computed by deterministic dynamic programming. In general, we would not expect the greedy policy to solve such a dynamic programming problem. However, as we will see in following sections, there are cases where it does.

### B. Sequential Scalar Measurements

In this subsection we consider the special case where \( L = 1 \) (i.e., each measurement is a scalar). In this case, we can write \( A_k = a_k^T \), where \( a_k \in \mathbb{R}^N \), \( R = \sigma^2 I \in \mathbb{R}^{n \times n} \). This scalar measurements case is of special interest because it corresponds to the problem of choosing the rows of a \( M \times N \) measurement matrix \( \Phi \) sequentially, one at a time, common in discussions of adaptive compressive measurements. In this interesting special scenario,

\[
y = \Phi x + w, \tag{8}
\]

where \( y \in \mathbb{R}^M \) is called the measurements vector, and \( w \) is a white Gaussian noise vector. In this context, the construction of a “good” measurement matrix \( \Phi \) (which would convey more information about \( x \)) is also a topic of interest. The concept of sequential scalar measurements in a closed-loop fashion has been discussed in a number of recent papers; e.g., \([3], [6], [8], [9], [11], [12]\). The objective function for the optimization here can take a number of possible forms, besides the net information gain. For example, in \([12]\), the objective is to maximize the posterior variance of the expected measurement.

In this paper we investigate applying the greedy policy in designing the measurement matrix. If the \( a_k \) can only be chosen from a prescribed finite set, the optimal design of \( \Phi \) is essentially a sensor selection problem (see \([13], [15]\)), where the greedy policy has been shown to perform well. For example, in the problem of sensor selection under a submodular objective function subject to a matroid constraint \([14], [2], [16]\), the suboptimality of the greedy policy has a provable bound.

Consider a constraint of the form \( ||a_k|| \leq 1 \) for \( k = 1, \ldots, M \) (\( ||\cdot|| \) is the Euclidean norm in \( \mathbb{R}^N \)), which is much more relaxed than a finite prescribed set. The expression in (6) simplifies to

\[
\log \det \left( I - \frac{P_{k-1}a_k a_k^T}{a_k^T P_{k-1} a_k + \sigma^2} \right), \tag{9}
\]

which further reduces (see \([4]\) Lemma 1.1)) to

\[
\log \det \left( 1 - \frac{a_k^T P_{k-1} a_k}{a_k^T P_{k-1} a_k + \sigma^2} \right). \tag{10}
\]

Combining (5), (9), and (10), the information gain in the \( k \)th step is

\[
H_{k-1} - H_k = -\frac{1}{2} \log \det \left( I - \frac{P_{k-1} a_k a_k^T}{a_k^T P_{k-1} a_k + \sigma^2} \right) = -\frac{1}{2} \log \left( \frac{1}{1 + \sigma^2/a_k^T P_{k-1} a_k} \right). \tag{11}
\]

Apparently, we should maximize \( a_k^T P_{k-1} a_k \) to obtain the maximal information gain in the \( k \)th step. We denote the eigenvalues of \( P_0 \) by \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n > 0 \). Since \( P_0 \) is a covariance matrix, which is symmetric, there exist corresponding orthonormal eigenvectors \( v_1, v_2, \ldots, v_n \), respectively. Clearly, we should set \( a_1 \) equal to \( v_1 \), which is the eigenvector of \( P_0 \) corresponding to its largest eigenvalue \( \lambda_1^0 : = \lambda_1 \). Then \( P_1 = (P_0^{-1} + \sigma^2 v_1 v_1^T)^{-1} \), and we can verify the following:

\[
v_i = P_1 P_i^{-1} v_i = P_1 (P_0^{-1} + \sigma^2 v_1 v_1^T) v_i = \frac{1}{\lambda_i} P_1 v_i, \quad \text{for } i \neq 1, \tag{12}
\]

\[
v_1 = P_1 P_i^{-1} v_1 = P_1 (P_0^{-1} v_1 + \sigma^2 v_1) = \frac{1}{\lambda_1 + \sigma^2} P_1 v_1. \tag{13}
\]

So we claim that \( P_1 \) has the same collection of eigenvectors as \( P_0 \), and the eigenvalues of \( P_1 \) are \( \lambda_1/(1 + \sigma^2 \lambda_1), \lambda_2, \ldots, \lambda_n \). By induction, we conclude that, when applying the greedy policy, all the \( P_k \)’s for \( k = 0, \ldots, M \) have the same collection of eigenvalues and the greedy policy always picks vectors from the set of eigenvectors \( \{v_1, \ldots, v_n\} \).

Denote the eigenvalues of \( P_k \) by \( \lambda_k^1 \geq \lambda_k^2 \geq \cdots \geq \lambda_k^M \). After applying \( M \) iterations of the greedy policy, the net information gain is

\[
H_0 - H_M = \sum_{k=1}^{M} \max_{||a_k|| \leq 1} (H_{k-1} - H_k) = \frac{1}{2} \sum_{k=1}^{M} \log \left( \frac{\sigma^2}{\lambda_k^1 + \sigma^2} \right) = \frac{1}{2} \log \prod_{k=1}^{M} (1 + \sigma^{-2} \lambda_k^1). \tag{14}
\]

### III. Relaxed Optimal Policy

In this section we consider the problem of maximizing the net information gain

\[
\sum_{k=1}^{M} (H_{k-1} - H_k) = -\frac{1}{2} \sum_{k=1}^{M} \log \frac{\det(P_k)}{\det(P_{k-1})} = \frac{1}{2} \log \frac{\det(P_0)}{\det(P_M)} = \frac{1}{2} \log \det(P_0) \det \left( P_0^{-1} + \sigma^2 \sum_{k=1}^{M} a_k a_k^T \right). \tag{15}
\]
subject to \( \|a_k\| \leq 1, \; k = 1, \ldots, M \). This maximization problem is actually equivalent to the maximum a posteriori probability (MAP) problem (see [1] and [16]). Note that \( \sum_{k=1}^{M} a_k a_k^T \) is positive semidefinite. Let \( G := \sigma^{-1}[a_1, a_2, \ldots, a_M] \), which is \( n \times M \). The constraint can be written as \( (G^T G)_{ij} \leq \sigma^{-2} \) for \( i = 1, \ldots, M \). Again by [4, Lemma 1.1], we obtain

\[
\log \det(P_0) \det \left( P_0^{-1} + \sigma^{-2} \sum_{k=1}^{M} a_k a_k^T \right) = \log \det(P_0) \det(P_0^{-1} + G G^T) = \log \det(I_M + G^T P_0 G),
\]

where \( I_M \) is the \( M \times M \) identity matrix.

Now consider the relaxed constraint \( \text{tr} G^T G = \sigma^{-2} \sum_{k=1}^{M} \|a_k\|^2 \leq \sigma^{-2} M \). There exists a unitary matrix \( U := [v_1, \ldots, v_n] \), such that \( \sigma^{-2} M P_0 = U^T \Lambda U \), where \( \Lambda = \sigma^{-2} M \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) \) and \( \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \) are the eigenvalues of \( P_0 \). The notation \( \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) \) represents the diagonal matrix with entries \( \lambda_1, \lambda_2, \ldots, \lambda_n \). As the matrix \( M^{-1} U \) is nonsingular, \( G \mapsto G = \sigma M^{-1} U G \) maps the set of \( M \times n \) matrices one-to-one onto itself. The constraint \( \text{tr} G^T G \leq \sigma^{-2} M \) becomes \( \text{tr} (\tilde{G})^T \tilde{G} \leq 1 \). We continue with [16] to write

\[
\log \det(I_M + G^T P_0 G) = \log \det(I_M + \tilde{G}^T \tilde{\Lambda} \tilde{G}).
\]

Hence, the relaxed optimization problem is equivalent to the following maximization problem:

\[
\text{Maximize } \frac{1}{2} \log \det(I_M + \tilde{G}^T \tilde{\Lambda} \tilde{G}),
\]

subject to \( \text{tr} (\tilde{G})^T \tilde{G} \leq 1 \).

Recall the following known results from [18].

**Lemma 1:** Given \( \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_M > 0 \), there exists a unique integer \( r \), with \( 1 \leq r \leq M \), such that for \( 1 \leq k \leq r \) we have

\[
\frac{1}{\lambda_k} < \frac{1}{k} \left( 1 + \sum_{j=1}^{k-1} \frac{1}{\lambda_j} \right),
\]

while for indices \( k \), if any, satisfying \( r < k \leq M \) we have

\[
\frac{1}{\lambda_k} \geq \frac{1}{k} \left( 1 + \sum_{j=1}^{k-1} \frac{1}{\lambda_j} \right).
\]

**Lemma 2:** For \( \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_M > 0 \) and \( r \) as in Lemma [1] the sequence

\[
m_k = k^{-k} \left( 1 + \sum_{j=1}^{k} \lambda_j^{-1} \right)^k \prod_{j=1}^{k} \lambda_j, \quad k = 1, \ldots, M,
\]

is strictly increasing.

By applying [18] Theorem 2, the optimal value of the relaxed maximization problem [18] is

\[
\frac{1}{2} \log \left( r^{-r} \left( \sigma^{-2} M + \sum_{j=1}^{r} \lambda_j^{-1} \right)^r \prod_{j=1}^{r} \lambda_j \right),
\]

where \( r \) is defined by the \( M \) biggest eigenvalues of \( P_0, \lambda_1, \lambda_2, \ldots, \lambda_M \), as in Lemma [1].

In fact, (22) is obtained from the solution of the well known water-filling problem (see [7] for details). It is known that

\[
r^{-r} \left( \sigma^{-2} M + \sum_{j=1}^{r} \lambda_j^{-1} \right)^r \prod_{j=1}^{r} \lambda_j
\]

is the optimal value of the following maximization problem:

\[
\text{Maximize } \prod_{i=1}^{M} (1 + \sigma^{-2} M \lambda_i p_i)
\]

subject to \( \sum_{i=1}^{M} p_i \leq 1 \).

The maximal value is only obtained when

\[
p_i = (\mu - \sigma^2 M^{-1} \lambda_i^{-1})^+, \quad i = 1, 2, \ldots, M,
\]

where

\[
\mu := r^{-1} \left( 1 + \sum_{i=1}^{r} \sigma^2 M^{-1} \lambda_i^{-1} \right)
\]

is called the water level. By taking a close look at (24), we can see that \( p_1 \geq \ldots \geq p_r > 0 \) and \( p_{r+1} = \ldots = p_M = 0 \). Figure [1] illustrates the relation among \( \Lambda_i := \sigma^2 M \lambda_i, p_i \), and water level \( \mu \).

For the sake of transparency, we make the following remark.

Let \( V := \tilde{G} \tilde{G}^T \). Then

\[
\det(I_M + \tilde{G}^T \tilde{\Lambda} \tilde{G}) = \det(I_n + \Lambda V).
\]

Denote the eigenvalues of \( V \) as \( \mu_1 \geq \mu_2 \geq \ldots \geq \mu_n \geq 0 \).
Since
\[ V = \sigma^2 M^{-1} U G G^T U^T = \sigma^2 M^{-1} U \left( \sum_{i=1}^{M} a_i a_i^T \right) U^T, \]

at most \( M \) of the \( \mu_i \) are positive. Moreover, \( \sum_{i=1}^{n} \mu_i = \text{tr} G^2 \leq 1 \). By [18, Lemma 3],
\[ \det(I_n + AV) \leq \prod_{i=1}^{n} (1 + \sigma^2 M \lambda_i \mu_i) = \prod_{i=1}^{M} (1 + \lambda_i \mu_i'), \]
where \( \mu_i' = \sigma^2 M \mu_i \). Equality holds if and only if \( \Lambda \) and \( V \) commute. Since \( \Lambda \) is a diagonal matrix, if \( V = \sigma^2 M^{-1} U G G^T U \) is also diagonal, they commute.

In practice, if \( G = \sigma^{-1}[a_1, a_2, \ldots, a_M] \) is obtained from a realization of the greedy algorithm, its columns are from the set of orthonormal eigenvectors. This implies that \( \Lambda \) and \( V \) commute, and [25] holds with equality. Furthermore, each \( \mu_i' \) is a multiple of \( \sigma^2 \) because it depends on the multiplicity of appearance of a specific eigenvector \( v \in \{v_1, \ldots, v_n\} \) as a column of \( G \).

IV. WHEN GREEDY IS OPTIMAL

In the preceding sections, we have discussed three types of policies: the greedy policy, the optimal policy, and the relaxed optimal policy. Denote by \( H_G, H_O, \) and \( H_R \) the net information gains associated with these three policies respectively. Clearly,
\[ H_G \leq H_O \leq H_R. \]

We now provide two sufficient conditions under which the greedy policy is optimal (i.e., \( H_G = H_O \)) for the sequential scalar measurements problem [8].

As defined before, \( \lambda_1 \geq \ldots \geq \lambda_n \) are the eigenvalues of \( P_0 \). \( M \) is the number of measurements, and \( r \) is defined as in Lemma [1].

**Theorem 3:** Assume that each row of \( \Phi \) can be selected to be any row vector \( a^T \) with \( ||a|| \leq 1 \). If \( \lambda_{k-1}^k - \lambda_k^{-1} = n_k \sigma^{-2} \), where \( n_k \) is some nonnegative integer, for \( k = 1, \ldots, r \), then the greedy policy is optimal.

**Proof:** In the \( k \)th iteration of the greedy algorithm, the algorithm changes the largest eigenvalue \( \lambda_{k-1}^k \) into \( \lambda_{k-1}^k/1 + \sigma^{-2} \lambda_k^{-1} \). This is equivalent to adding \( \sigma^{-2} \) to \( 1/\lambda_{k-1}^k \). If we consider the whole process of the greedy algorithm, it simply allocates \( M \) blocks of size \( \sigma^{-2} \) one by one to the channel corresponding to the biggest eigenvalue of \( P_{k-1} \).

Since \( \lambda_{k-1}^k - \lambda_k^{-1} = n_k \sigma^{-2} \), \( k = 1, \ldots, r \), the greedy solution fills the blocks of size \( \sigma^{-2} \) will give the same maximal value of \( \prod_{k=1}^{M} (1 + \lambda_k \mu_k') \) as the water-filling solution. Therefore, \( H_G = H_R \). Form (29), we conclude that \( H_G = H_O \).

The next result provides an alternative sufficient condition for greedy to be optimal, based on eigenvectors of \( P_0 \).

**Theorem 4:** Suppose that the rows of \( \Phi \) can be picked from a set \( S \subseteq \{v_1^T, \ldots, v_r^T\} \), which is a subset of the orthonormal eigenvectors of \( P_0 \). If \( \{v_i^T, \ldots, v_r^T\} \subseteq S \), then the greedy policy is optimal.

**Proof:** If we pick rows of \( \Phi \) from \( \{v_1^T, \ldots, v_r^T\} \), \( V \) is a diagonal matrix as we analyzed in (27). Then \( \Lambda \) and \( V \) commute and the equality in (28) holds. Hence, the net information gain is \( \prod_{k=1}^{M} (1 + \lambda_k \mu_k') \). We have claimed that each \( \mu_k' \) is a multiple of \( \sigma^{-2} \) and \( \sum_{k=1}^{M} \mu_k' = \sigma^{-2} M \). The optimal solution is simply the best allocation of \( M \) blocks which maximizes \( \prod_{k=1}^{M} (1 + \lambda_k \mu_k') \). Assume that \( \gamma = (\gamma_1, \gamma_2, \ldots, \gamma_M) \) gives an optimal solution. If \( \gamma_i + \lambda_i^{-1} \mu_i < \sigma^{-2} \) for some \( 1 \leq i \leq M \), where \( \mu_i \) is the water level defined in (25), this means there exist some channel \( 1 \leq j \leq M \), such that \( (\gamma_j + \lambda_j^{-1} - \mu) < 0 \).

Now move the top block of the \( i \)th channel to the \( j \)th channel to get another allocation \( \eta = (\eta_1, \ldots, \eta_M) \). Clearly, \( \eta \) and \( \gamma \) have the same entries except the \( i \)th and \( j \)th ones. The argument in this paragraph is illustrated in Figure 2.

Write \( \gamma_k + \lambda_k^{-1} = \mu + \delta_k \) for \( k = 1, \ldots, M \). So
\[ \frac{\prod_{k=1}^{M} (1 + \lambda_k \eta_k)}{\prod_{i=k}^{M} (1 + \lambda_i \gamma_k)} = \frac{(1 + \lambda_i (\mu + \delta_i - \lambda_i^{-1} - \sigma^{-2}))}{(1 + \lambda_i (\mu + \delta_i - \lambda_i^{-1}))} \times \frac{(1 + \lambda_j (\mu + \delta_j - \lambda_j^{-1} + \sigma^{-2}))}{(1 + \lambda_j (\mu + \delta_j - \lambda_j^{-1}))} = \frac{(\mu + \delta_i - \sigma^{-2}) (\mu + \delta_j + \sigma^{-2})}{(\mu + \delta_i) (\mu + \delta_j)} > 1, \]

since \( \delta_i - \delta_j > \sigma^{-2} \). Thus \( \eta \) gives a better allocation and a contradiction to the optimality of \( \gamma \). By similar arguments, we obtain that for optimal solution \( \gamma \), there also does not exist \( i \) such that \( \gamma_i + \lambda_i^{-1} - \mu \leq \sigma^{-2} \). In conclusion, in the optimal solution the water level in each channel deviates from \( \mu \) less than \( \sigma^2 \). This also means in an optimal allocation \( \gamma = (\gamma_1, \gamma_2, \ldots, \gamma_M) \), \( \gamma_{r+1} = \ldots = \gamma_M = 0 \). Since we only need to fill blocks to the first \( r \) channels to obtain an optimal allocation, only \( v_1^T, \ldots, v_r^T \) are needed in \( S \).
It turns out that we can show an identical property for the greedy policy. Assume that \( \eta = (\eta_1, \eta_2, \ldots, \eta_n) \) gives a greedy solution. If \( \eta_i > 0 \) and \((\eta_i + \lambda_i^{-1} - \mu) \geq \sigma^{-2}\), for some \( 1 \leq i \leq n \), this implies that there exists a channel \( 1 \leq j \leq M \), such that \((\eta_j + \lambda_j^{-1} - \mu) < 0\). Therefore, when the greedy algorithm fills the last block to channel \( i \), it does not add that block to a channel whose level is lower. This gives a contradiction. By similar arguments, there does not exist some channel \( i \) such that \((\eta_i + \lambda_i^{-1} - \mu) \leq -\sigma^{-2}\). This implies that in the greedy solution the water level in each channel deviates from \( \mu \) less than \( \sigma^{-2} \). Moreover, the greedy algorithm will never fill blocks to channels other than the first \( r \) of them. This implies that only \( v_i^1, \ldots, v_i^r \) are needed in \( S \).

Consequently, both the greedy algorithm and optimal allocation meet the following stage in the allocating process: if we add one block to any channel \( i \) for \( 1 \leq i \leq r \), the water level in that channel will be above \( \mu \). Assume that at this stage we have already allocated \( M' \) blocks and there are still \( r' \) more blocks to be allocated. Easy to check with the argument of the total volume \((M - M') \alpha \), we can verify that the optimal solution is simply adding those \( r' \) blocks to the channels with \( r' \) lowest water levels respectively. Otherwise, use above arguments we can always find a better allocation. On the other hand, the greedy algorithm will do the same thing with those \( r' \) blocks. So the greedy policy is exactly the optimal policy.

V. WHEN GREEDY IS NOT OPTIMAL

A. An Example with Non-Scalar Measurements

In this subsection we give an example where the greedy policy is not optimal. Indeed, as we see, the greedy policy can be arbitrarily worse than the optimal policy. Suppose that we are restricted to a set of only three choices of \( A_k \):

\[
A = \{ \text{diag}(1,0), \text{diag}(0,1), \frac{1}{2} \text{diag}(1,1) \}.
\]

Note that \( \text{diag}(1,1) = I \). In this case, \( L = N = 2 \). Moreover, set \( M = 2 \), \( P_0 = 16I \), and \( R = I \).

Let us see what the greedy policy would do in this case. For \( k = 1 \), it would pick \( A_1 \in A \) to maximize

\[
\det \left( \frac{1}{16} I + (A_1)^2 \right).
\]

A quick calculation shows that for \( A_1 = \text{diag}(1,0) \) or \( \text{diag}(0,1) \), we have

\[
\det \left( \frac{1}{16} I + (A_1)^2 \right) = \frac{17}{256},
\]

whereas for \( A_1 = \frac{1}{2} \text{diag}(1,1) \),

\[
\det \left( \frac{1}{16} I + (A_1)^2 \right) = \frac{25}{256},
\]

So the greedy policy picks \( A_1 = \frac{1}{2} \text{diag}(1,1) \), which leads to \( P_1 = \frac{16}{7} I \).

For \( k = 2 \), we go through the same calculations: for \( A_2 = \text{diag}(1,0) \) or \( \text{diag}(0,1) \), we have

\[
\det \left( \frac{5}{16} I + (A_2)^2 \right) = \frac{105}{256}
\]

whereas for \( A_2 = \frac{1}{2} \text{diag}(1,1) \),

\[
\det \left( \frac{5}{16} I + (A_2)^2 \right) = \frac{81}{256}.
\]

So, this time the greedy policy picks \( A_2 = \text{diag}(1,0) \) (or \( \text{diag}(0,1) \)), after which \( \det(P_2) = 256/105 \).

Consider the alternative policy that picks \( A_1 = \text{diag}(1,0) \) and \( A_2 = \text{diag}(0,1) \). In this case,

\[
P_2^{-1} = \frac{1}{16} I + \text{diag}(1,0) + \text{diag}(0,1) = \frac{17}{16} I
\]

and so \( \det(P_2) = 256/289 \), which is clearly less than what was obtained with the greedy policy. Call this alternative policy the alternating policy (because it alternates between \( \text{diag}(1,0) \) and \( \text{diag}(0,1) \)).

Conclusion: For this example, the greedy policy is not optimal with respect to the objective of maximizing the net information gain.

How much worse is the objective function of the greedy policy relative to that of the alternating policy? Suppose that we set \( P_0 = \alpha^{-1} I \) and let the third choice in \( A \) be \( \alpha^{1/4} I \), where \( \alpha > 0 \) is some small number. (Note that the numerical example above is a special case with \( \alpha = 1/16 \). In this case, it is straightforward to check that the greedy policy picks \( A_1 = \alpha^{1/4} I \) and \( A_2 = \text{diag}(1,0) \) (or \( \text{diag}(0,1) \)) if \( \alpha \) is sufficiently small, resulting in

\[
\det(P_2) = \frac{1}{\sqrt{\alpha}(1 + \sqrt{\alpha})(1 + \sqrt{\alpha} + \alpha)},
\]

which increases unboundedly as \( \alpha \to 0 \). However, the alternating policy results in

\[
\det(P_2) = \frac{1}{(1 + \alpha)^2},
\]

which converges to 1 as \( \alpha \to 0 \). Hence, letting \( \alpha \) get arbitrarily small, the ratio of the objective function for the greedy policy to that of the alternating policy can be made arbitrarily large. This means that the greedy policy is arbitrarily worse than the alternating policy.

What went wrong? The greedy policy was “fooled” into picking \( A_1 = \alpha^{1/4} I \) in the first stage, because this choice maximizes the per-stage information gain in the first stage. But once it does that, it is stuck with its resulting covariance matrix \( P_1 \). The alternating policy trades off the per-stage information gain in the first stage for the sake of better net information gain over two stages. The first measurement matrix \( \text{diag}(1,0) \) “sets up” the covariance matrix \( P_1 \) so that the second measurement matrix \( \text{diag}(0,1) \) can take advantage of it to obtain a superior covariance \( P_2 \) after the second stage, embodying a form of “delayed gratification.”

Interestingly, the argument above depends on the value of \( \alpha \) being sufficiently small. For example, if \( \alpha = 1/4 \), then the greedy policy has the same net information gain as the alternating policy, and is in fact optimal.

B. An Example with Scalar Measurements

Assume that

\[
P_0 = \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix},
\]

\[
p_0 = \begin{bmatrix} \frac{1}{16} & 0 \\ 0 & \frac{1}{16} \end{bmatrix} \begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix} = \begin{bmatrix} \frac{5}{16} & \frac{5}{16} \\ \frac{5}{16} & \frac{5}{16} \end{bmatrix}
\]

\[
\det(\frac{5}{16} I + (A_2)^2) = \frac{81}{256}.
\]

So, this time the greedy policy picks \( A_2 = \text{diag}(1,0) \) (or \( \text{diag}(0,1) \)), after which \( \det(P_2) = 256/105 \).
$R = I$, and set $M = 2$. Our goal is to find $\|a\|, \|b\| \leq 1$ such that $a, b$ maximize the net information gain:

$$H_0 - H_2 = \frac{1}{2} \log \det(P_0) \det(P_0^{-1} + aa^T + bb^T). \quad (32)$$

By simple computation, we know that the eigenvalues of $P_0$ are $\lambda_1^2 = 5$ and $\lambda_2^2 = 1$. If we follow the greedy policy, the eigenvalues of $P_1$ are $\lambda_1^2 = 1$ and $\lambda_2^2 = 5/6$. By (14), the net information gain for the greedy policy is

$$H_0 - H_2 = \frac{1}{2} \log(1 + 5)(1 + 1) = \frac{1}{2} \log(12).$$

Next we solve for the optimal solution. Let $a = [a_1, a_2]^T$. By (3), we have

$$P_1 = \begin{bmatrix}
\frac{5a_2^2 + 3}{3a_1^2 + 4a_1a_2 + 3a_2^2 + 1} & -\frac{(5a_1a_2 - 2)}{3a_1^2 + 4a_1a_2 + 3a_2^2 + 1} \\
-\frac{(5a_1a_2 - 2)}{3a_1^2 + 4a_1a_2 + 3a_2^2 + 1} & \frac{5a_1^2 - 3}{3a_1^2 + 4a_1a_2 + 3a_2^2 + 1}
\end{bmatrix}.$$

We compute that

$$\lambda_1^1 = \frac{25a_1^4 + 50a_1^2a_2^2 - 80a_1a_2 + 25a_2^4 + 16}{6a_1^2 + 8a_1a_2 + 6a_2^2 + 2}$$

$$+ \frac{5a_1^2 + 5a_2^2 + 6}{6a_1^2 + 8a_1a_2 + 6a_2^2 + 2}. \quad (33)$$

When we choose $b$ in the second stage, we can simply maximize the information gain in that stage. In this special case when $M = 2$, the second stage is the actually the last one. If $a$ is given, maximizing the net information gain is equivalent to maximizing the information gain in the second stage. Therefore, the second step is equivalent to a greedy step. By (14),

$$H_1 - H_2 = -\frac{1}{2} \log \left(1 - \frac{1}{1 + 1/\lambda_1^1}\right) = \frac{1}{2} \log(1 + \lambda_1^1). \quad (34)$$

By (11), we know

$$H_0 - H_1 = -\frac{1}{2} \log \det \left(I - \frac{P_0aa^T}{a^TP_0a + 1}\right)$$

$$= \frac{1}{2} \log(4 + 4a_1a_2). \quad (35)$$

Using $\|a\| = 1$, we simplify the sum of (34) and (35) and obtain

$$H_0 - H_2 = \frac{1}{2} \log \left(\frac{1}{2} \left[(41 - 80a_1a_2)^{1/2} + 19 + 8a_1a_2\right]\right). \quad (36)$$

This expression reaches its maximal value when $a_1a_2 = 1/5$. So the optimal net information gain is $1/2 \log(12.8)$, when

$$a = \left[\frac{-\sqrt{21} + 5}{10}, \frac{\sqrt{21} + 5}{10}\right]^T$$

and

$$b = \left[\frac{\sqrt{21} + 5}{10}, \frac{-\sqrt{21} + 5}{10}\right]^T.$$

This implies that the greedy policy is not optimal.