KU LEUVEN

STADIUS
Center for Dynamical Systems, Signal Processing and Data Analytics

Citation/Reference	Narayanan A.M., Bertrand A. (2020), **Group-utility metric for efficient sensor selection and removal in LCMV beamformers** *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Barcelona, Spain, 2020, pp. 4950-4954
Archived version	Author manuscript;
Published version	https://ieeexplore.ieee.org/document/9053467
Journal homepage	https://ieeexplore.ieee.org/xpl/conhome/9040208/proceeding
Author contact	abhijith@esat.kuleuven.be
Abstract	
IR	

(article begins on next page)

GROUP-UTILITY METRIC FOR EFFICIENT SENSOR SELECTION AND REMOVAL IN LCMV BEAMFORMERS

Abhijith Mundanad Narayanan and Alexander Bertrand

Dept. of Electrical Engineering (ESAT), Stadius Center for Dynamical Systems, Signal Processing and Data Analytics (STADIUS), KU Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium

ABSTRACT

In sensor arrays or sensor networks, tracking each sensors utility helps in excluding those which do not sufficiently contribute to the task at hand, thereby reducing energy consumption or avoiding model overfitting. In a linearly-constrained minimum variance (LCMV) beamformer, the utility of a sensor is defined as the increase in the beamformer's output noise power when the sensor would be removed and the beamformer coefficients re-optimized. An expression to efficiently compute this utility metric has been found for the case where each sensor removal corresponds to a single beamformer coefficient. However, in a filter-and-sum implementation, a single sensor is filtered by a group of beamformer coefficients. Furthermore, sometimes one wants to track the joint utility of a group of sensors. In this paper we derive a generalized expression to efficiently calculate the utility of such groups as a whole, called the group-utility. We show that the computational complexity of this generalized expression is negligible if the number of groups is larger than the group sizes, leading to a very efficient group-utility computation compared to the straightforward implementation. Furthermore, an efficient updating equation re-optimizing the LCMV beamformer when a group of G beamformer inputs is removed is found as a by-product.

***Index Terms*—** Sensor subset selection, beamforming, group-utility, sensor networks

1. INTRODUCTION

An array of sensors often improves the detection of physical phenomena or the estimation of signals compared to single sensor systems [1, 2]. Tracking the usefulness of individual sensors in a sensor array or a sensor network allows to perform sensor subset selection to reduce energy consumption [3], to avoid injecting noise in the system to avoid model overfitting or for sensor deployment topology identification [4]. In this paper, we address the issue of finding this usefulness of sensors in the context of beamforming.

A beamformer aims to estimate a target signal that is captured by an array of sensors in the presence of noise and interfering signals [5]. Although, beamforming was originally formulated for analysis of radar and sonar signals [5], it has been used in various other applications ranging from speech enhancement in linear arrays [6] or acoustic sensor networks [7] to analysis of event-related potentials (ERPs) in electorencephalography (EEG) [8]. Specifically, we consider the popular linearly-constrained minimum variance (LCMV) beamformer which minimizes the output power of the beamformer under a set of linear constraints designed to pass the signals coming from the desired directions while signals coming from the interfering directions are rejected.

In literature, the 'utility' metric has been proposed to track the usefulness of sensors for various signal estimation problems [9, 10]. In [9], the utility of an individual sensor in an LCMV beamformer was defined as the increase in the total power of the beamformer output signal if the input variable corresponding to this sensor is removed from the estimation problem and the remaining beamformer coefficients are re-optimized using the remaining input variables. However, in many practical scenarios, it makes sense to consider a group of input variables as a single entity. For instance, when a single sensor device consists of many sensors thereby providing multiple input variables or when multiple time-delayed versions of a single sensor signal are combined by the beamformer, thereby allowing it to perform time-domain filtering. Therefore, in these cases it is essential to obtain the utility of a *group* of input variables in the LCMV beamformer, which is defined as the increase in the beamformer output signal power when the entire group of variables are removed at once. Note that this is different than the sum of the utilities of the individual input variables in the group. Similar to [10], which focuses on the case of minimum mean squared error estimation, we term this utility of a group of input variables in an LCMV beamformer as the 'group-utility'. In this paper we derive an expression to efficiently compute it, thereby generalizing the derivation in [9] to a groupwise metric. As a by-product in the derivation, we also obtain an expression to efficiently update the LCMV beamformer coefficients when a group of input variables is removed, which extends existing efficient LCMV beamformer updating methods for a single sensor removal [9, 11] to a group-wise removal. We also perform a complexity analysis, which proves that the computational cost to compute the group-utility metric for K groups of size G can be brought down to only $O(KG^3)$, i.e., linearly in the number of groups K. This is to be compared with the straightforward calculation directly based on the group-utility definition, which would have having a complexity of $O(K(K-1)^3G^3)$.

This work was carried out at the ESAT laboratory of KU Leuven and has received funding from KU Leuven Special Research Fund C14/16/057, FWO project nrs. 1.5.123.16N and G0A4918N. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 802895).

The outline of the paper is as follows. The LCMV beamforming problem is introduced in Section 2.1 followed by Section 2.2 which defines group-utility. In Section 3, efficient expressions to compute the group-utility and to update the LCMV beamformer coefficients are derived. In Section 4 an analysis of the computational complexity is performed and compared to the naive computation of the group-utility. Finally, conclusions are drawn in Section 5.

2. PROBLEM FORMULATION

2.1. LCMV Beamforming

A filter-and-sum beamformer using an array of N sensors estimates the desired signal d at time t as:

$$d(t) = \sum_{n=1}^{N} \sum_{\tau=0}^{J-1} w_n(\tau) y_n(t-\tau) \tag{1}$$

where y_n is the signal from sensor n, w_n is the n-th beamformer coefficient and J is the number of filter taps used for each sensor signal, i.e., the length of the delay line [5]. Expression (1) can be re-written in vector form as:

$$d(t) = \mathbf{w}^T \mathbf{y}(t) \tag{2}$$

where $\mathbf{w}$ is the M-dimensional vector containing $M = NJ$ filter coefficients $w_n(\tau)$, $\mathbf{y}(t)$ is the M-dimensional vector stacked with signals from N sensors and their J time-delayed versions $y_n(t-\tau)$ for $\tau = 0, \cdots, J-1$ [5]. We further simplify (2) by the following expression which estimates the desired signal d at $t = 1, 2 \cdots T$ time samples in the vector $\mathbf{d} \in \mathbb{R}^T$ as:

$$\mathbf{d} = \mathbf{Y}\mathbf{w} \tag{3}$$

where $\mathbf{Y}$ is a $T \times M$ matrix containing in its T rows $\mathbf{y}(t)^T$ for $t = 0, 1, \cdots (T-1)$. In the remaining of this paper, we will refer to the M columns of $\mathbf{Y}$ as input variables which in the present case represents sensor signals and their time-delayed versions.

LCMV beamforming is achieved by calculating the multichannel filter $\hat{\mathbf{w}}$ which minimizes the output signal power $P(\mathbf{w})$ which is defined as:

$$P(\mathbf{w}) = \sum_{t=0}^{T-1} |\mathbf{w}^T \mathbf{y}(t)|^2 = \mathbf{w}^T \mathbf{R}_{YY} \mathbf{w} \tag{4}$$

where $\mathbf{R}_{YY} = \mathbf{Y}^T\mathbf{Y}$, while satisfying a set of Q linear constraints. Thus, the LCMV beamformer $\hat{\mathbf{w}}$ is the solution of:

$$\begin{aligned} \hat{\mathbf{w}} &= \arg \min_{\mathbf{w}} P(\mathbf{w}) \\ &\text{subject to } \mathbf{C}^T\mathbf{w} = \mathbf{f} \end{aligned} \tag{5}$$

where $\mathbf{C}$ is the $M \times Q$ matrix containing the Q constraints and $\mathbf{f}$ is the response vector of length Q [5]. Typically, the columns of $\mathbf{C}$ consist of known transfer functions to specific sources, and $\mathbf{f}$ contains the desired beamformer response for each of these sources (e.g., 1 for a desired source, and 0 for an interfering source) The solution to (5) is given by

$$\hat{\mathbf{w}} = \mathbf{R}_{YY}^{-1}\mathbf{C}\left(\mathbf{C}^T\mathbf{R}_{YY}^{-1}\mathbf{C}\right)^{-1}\mathbf{f} \tag{6}$$

where, $\mathbf{R}_{YY}$ is assumed to be of full rank.

2.2. Definition of Group-Utility

We define the utility of a group of G input variables as the increase in output power P when the G corresponding columns of $\mathbf{Y}$ are removed and the remaining coefficients in $\mathbf{w}$ are re-optimized. In practice, this scenario could correspond to the removal of a subarray with multiple sensors, or a single sensor with multiple time lags or a combination of both. To contrast from the utility of a single variable defined in [9], we term this utility of a group of variables as the *group-utility*.

Let us term the new LCMV solution after removal of G input variables as $\hat{\mathbf{w}}_{-G}$. Then by definition, the group-utility of G variables is:

$$U_G = P(\hat{\mathbf{w}}_{-G}) - P(\hat{\mathbf{w}}). \tag{7}$$

The straightforward method to compute this group-utility is to use (4) to calculate costs after obtaining the new LCMV solution $\hat{\mathbf{w}}_{-G}$ which is defined as:

$$\hat{\mathbf{w}}_{-G} = \left(\mathbf{R}_{YY_{-G}}\right)^{-1}\bar{\mathbf{C}}_{-G}\left(\bar{\mathbf{C}}_{-G}^T\left(\mathbf{R}_{YY_{-G}}\right)^{-1}\bar{\mathbf{C}}_{-G}\right)^{-1}\mathbf{f} \tag{8}$$

where $\mathbf{R}_{YY_{-G}} = \mathbf{Y}_{-G}^T\mathbf{Y}_{-G}$ with $\mathbf{Y}_{-G}$ defined as $\mathbf{Y}$ with the G columns removed that correspond to the G removed input variables, and similarly $\bar{\mathbf{C}}_{-G}$ is the matrix $\mathbf{C}$ with the corresponding G rows removed. To obtain $\left(\mathbf{R}_{YY_{-G}}\right)^{-1}$ which is required to solve (8), an inversion of the submatrix $\mathbf{R}_{YY_{-G}}$ of $\mathbf{R}_{YY}$ can be carried out. This operation is associated with a computational complexity of $O\left((M-G)^3\right)$. Moreover, to evaluate the group-utility of K such groups, the inversion of an $\mathbf{R}_{YY_{-G}}$ matrix has to be carried out K times. Assuming each group consists of G variables, the total complexity for obtaining the group-utility of all the K groups would be $O(K(K-1)^3G^3)$. In this paper, we derive an efficient expression for calculating this group-utility with a computational complexity of only $O(KG^3)$.

Remark: We note that, in practice, the target signal is constrained to be preserved by the LCMV beamformer through the constraint matrix $\mathbf{C}$, in which case any change in beamformer output power due to the removal of columns of $\mathbf{Y}$ is purely due to an increase in output noise power. As such, the utility metric can be straightforwardly related to the SNR through a strictly monotonic relationship (details omitted), i.e., the group of variables that has the lowest utility will also lead to the least decrease in SNR when these variables would be removed.

3. EFFICIENT CALCULATION OF GROUP-UTILITY

3.1. Efficient group-utility equation

To efficiently calculate the group-utility metric, we introduce the variables $\mathbf{T}$ and $\mathbf{L}$ here:

$$\mathbf{T} = \mathbf{R}_{YY}^{-1}\mathbf{C}\ ; \quad \mathbf{T}_{-G} = \left(\mathbf{R}_{YY_{-G}}\right)^{-1}\bar{\mathbf{C}}_{-G} \tag{9}$$

$$\mathbf{L} = \left(\mathbf{C}^T\mathbf{T}\right)^{-1}; \quad \mathbf{L}_{-G} = \left(\bar{\mathbf{C}}_{-G}^T\mathbf{T}_{-G}\right)^{-1} \tag{10}$$

With this new notation, (6) and (8) can be written as:

$$\hat{\mathbf{w}} = \mathbf{TLf}\,, \;\; \hat{\mathbf{w}}_{-G} = \mathbf{T}_{-G}\mathbf{L}_{-G}\mathbf{f} \tag{11}$$

We also define a few block partitionings of some of the matrices where we assume without loss of generality that the group of G input variables of which we aim to compute the group-utility corresponds to the last G columns of $\mathbf{Y}$ (this can always be obtained by reshuffling the input variables).

The matrix $\mathbf{R}_{YY}$ and $\mathbf{R}_{YY}^{-1}$ are partitioned as follows:

$$\mathbf{R}_{YY} = \left[\begin{array}{c|c} \mathbf{R}_{YY_{-G}} & \mathbf{B}_G \\ \hline \mathbf{B}_G^T & \mathbf{D}_G \end{array}\right] \quad \mathbf{R}_{YY}^{-1} = \left[\begin{array}{c|c} * & * \\ \hline * & \mathbf{\Gamma}_G \end{array}\right] \tag{12}$$

where $\mathbf{D}_G$ and $\mathbf{\Gamma}_G$ are $G \times G$ matrices and $\mathbf{B}_G$ is an $(M - G) \times G$ matrix. The parts indicated with $*$ will not be used in the sequel. The variables $\hat{\mathbf{w}}, \mathbf{C}$ and $\mathbf{T}$ are partitioned as follows:

$$\hat{\mathbf{w}} = \begin{bmatrix} \bar{\mathbf{w}}_{-G} \\ \bar{\mathbf{w}}_G \end{bmatrix} \tag{13}$$

$$\mathbf{C} = \begin{bmatrix} \bar{\mathbf{C}}_{-G} \\ \bar{\mathbf{C}}_G \end{bmatrix} \tag{14}$$

$$\mathbf{T} = \begin{bmatrix} \bar{\mathbf{T}}_{-G} \\ \bar{\mathbf{T}}_G \end{bmatrix} \tag{15}$$

At this point, it is important to notice the difference between a variable with and without overline notation, e.g., the difference between $\bar{\mathbf{T}}_{-G}$ and $\mathbf{T}_{-G}$. When using an overline notation as in $\bar{\mathbf{T}}_{-G}$ in (15), we refer to the result of removing the last G rows of the matrix $\mathbf{T}$. $\mathbf{T}_{-G}$ in (9), on the other hand is a new matrix of which the coefficients do not overlap with the first $M - G$ rows of $\mathbf{T}$ and would be used in (11) to compute the optimal beamformer coefficients $\hat{\mathbf{w}}_{-G}$ *after* removal of G input variables. Using the variables defined previously, the following expression to efficiently compute the group-utility can be obtained (the derivation of this equation is provided in Section 3.2):

$$\boxed{U_G = \bar{\mathbf{w}}_G^T \left(\mathbf{\Gamma}_G - \bar{\mathbf{T}}_G \mathbf{L} \bar{\mathbf{T}}_G^T\right)^{-1} \bar{\mathbf{w}}_G} \tag{16}$$

where the variables $\mathbf{L}$, $\mathbf{\Gamma}_G$, $\bar{\mathbf{w}}_G$ and $\bar{\mathbf{T}}_G$, are already available from the computation of the full beamformer $\hat{\mathbf{w}}$ through (10), (12), (13) and (15) respectively. It is noted that the sensor utility expression derived in [9] is a special case of (16) when the group size $G = 1$.

The computational complexity for the group-utility evaluation of a single group of G variables is only $O(G^3)$, for the inversion of the $G \times G$ matrix in (16) or $O(QG^3)$ for the multiplication of $\bar{\mathbf{T}}_G$ and $\mathbf{L}$ if $Q > G$. Therefore, for the evaluation of the group-utility of K such groups, the computational complexity is $O(KG^3)$. Comparing this with the computational complexity of straightforward group-utility evaluation given in Section 2, (16) is clearly less complex.

In the next section we derive the expression in (16). As a by-product, this derivation will also provide two computationally efficient equations to compute $\mathbf{T}_{-G}$, $\mathbf{L}_{-G}$ and hence the new optimal LCMV beamformer $\hat{\mathbf{w}}_{-G}$ in (11) after removal of the G columns of $\mathbf{Y}$.

3.2. Derivation of efficient group-utility equation

Applying the matrix inversion lemma [12] to invert the block matrix form of $\mathbf{R}_{YY}$ given in (12), it can be verified that

$$\mathbf{R}_{YY}^{-1} = \left[\begin{array}{c|c} \left(\mathbf{R}_{YY_{-G}}\right)^{-1} + \mathbf{V}_G \mathbf{\Gamma}_G \mathbf{V}_G^T & -\mathbf{V}_G \mathbf{\Gamma}_G \\ \hline -\mathbf{\Gamma}_G \mathbf{V}_G^T & \mathbf{\Gamma}_G \end{array}\right] \tag{17}$$

where,

$$\mathbf{V}_G = \left(\mathbf{R}_{YY_{-G}}\right)^{-1} \mathbf{B}_G, \tag{18}$$

$$\mathbf{\Gamma}_G = \left(\mathbf{D} - \mathbf{B}_G^T \mathbf{V}_G\right)^{-1}. \tag{19}$$

We now derive the efficient expression for $\mathbf{L}_{-G}$ which will lead to (16). Since $\mathbf{T} = \mathbf{R}_{YY}^{-1}\mathbf{C}$ and using (17) and the partitioning defined in (15), we write $\mathbf{T}$ in block matrices as:

$$\begin{bmatrix} \bar{\mathbf{T}}_{-G} \\ \bar{\mathbf{T}}_G \end{bmatrix} = \begin{bmatrix} \left(\mathbf{R}_{YY_{-G}}\right)^{-1} \bar{\mathbf{C}}_{-G} + \mathbf{V}_G \mathbf{\Gamma}_G \mathbf{V}_G^T \bar{\mathbf{C}}_{-G} - \mathbf{V}_G \mathbf{\Gamma}_G \bar{\mathbf{C}}_G \\ -\mathbf{\Gamma}_G \mathbf{V}_G^T \bar{\mathbf{C}}_{-G} + \mathbf{\Gamma}_G \bar{\mathbf{C}}_G \end{bmatrix}. \tag{20}$$

Comparing the upper block in (20) and using (9):

$$\bar{\mathbf{T}}_{-G} = \mathbf{T}_{-G} + \mathbf{V}_G \left(\mathbf{\Gamma}_G \mathbf{V}_G^T \bar{\mathbf{C}}_{-G} - \mathbf{\Gamma}_G \bar{\mathbf{C}}_G\right) \tag{21}$$

Plugging in the lower block of (20) in (21), we get:

$$\boxed{\mathbf{T}_{-G} = \bar{\mathbf{T}}_{-G} + \mathbf{V}_G \bar{\mathbf{T}}_G} \tag{22}$$

Substituting the block representations of $\mathbf{R}_{YY}^{-1}$ and $\mathbf{C}$ from (17) and (14) respectively in the definition of $\mathbf{L}$ from (10) it can be verified that

$$\begin{aligned} \mathbf{C}^T \mathbf{R}_{YY}^{-1} \mathbf{C}^T = {} & \bar{\mathbf{C}}_{-G}^T \left(\mathbf{R}_{YY_{-G}}\right)^{-1} \bar{\mathbf{C}}_{-G} \\ & + \left(\bar{\mathbf{C}}_{-G}^T \mathbf{V}_G - \bar{\mathbf{C}}_G^T\right) \mathbf{\Gamma}_G \left(\bar{\mathbf{C}}_{-G}^T \mathbf{V}_G - \bar{\mathbf{C}}_G^T\right)^T. \end{aligned} \tag{23}$$

For notational purposes, we introduce the matrix $\mathcal{V}_G = \begin{bmatrix} \mathbf{V}_G \\ -\mathbf{I}_G \end{bmatrix}$ where $\mathbf{I}_G$ is an identity matrix of size $G \times G$. Combining (23) and (10) we get:

$$\mathbf{L}_{-G} = \left(\mathbf{L}^{-1} - \mathbf{C}^T \mathcal{V}_G \mathbf{\Gamma}_G \mathcal{V}_G^T \mathbf{C}\right)^{-1} \tag{24a}$$

$$= \left(\mathbf{L}^{-1} - \bar{\mathbf{T}}_G^T \mathbf{\Gamma}_G^{-1} \bar{\mathbf{T}}_G\right)^{-1} \tag{24b}$$

where the second step uses: $\mathcal{V}_G^T \mathbf{C} = -\mathbf{\Gamma}_G^{-1} \bar{\mathbf{T}}_G$, which follows from combining (14), (15), (9) with the lower part of (17). Applying the matrix inversion lemma on (24b), we find

$$\boxed{\mathbf{L}_{-G} = \mathbf{L} + \mathbf{L} \bar{\mathbf{T}}_G^T \left(\mathbf{\Gamma}_G - \bar{\mathbf{T}}_G \mathbf{L} \bar{\mathbf{T}}_G^T\right)^{-1} \bar{\mathbf{T}}_G \mathbf{L}} \tag{25}$$

Finally, the new LCMV beamformer when a group of G variables is removed can be computed using (11), where $\mathbf{L}_{-G}$ and $\mathbf{T}_{-G}$ can be efficiently calculated using readily available variables from the computation of the full beamformer as shown in (22) and (25).

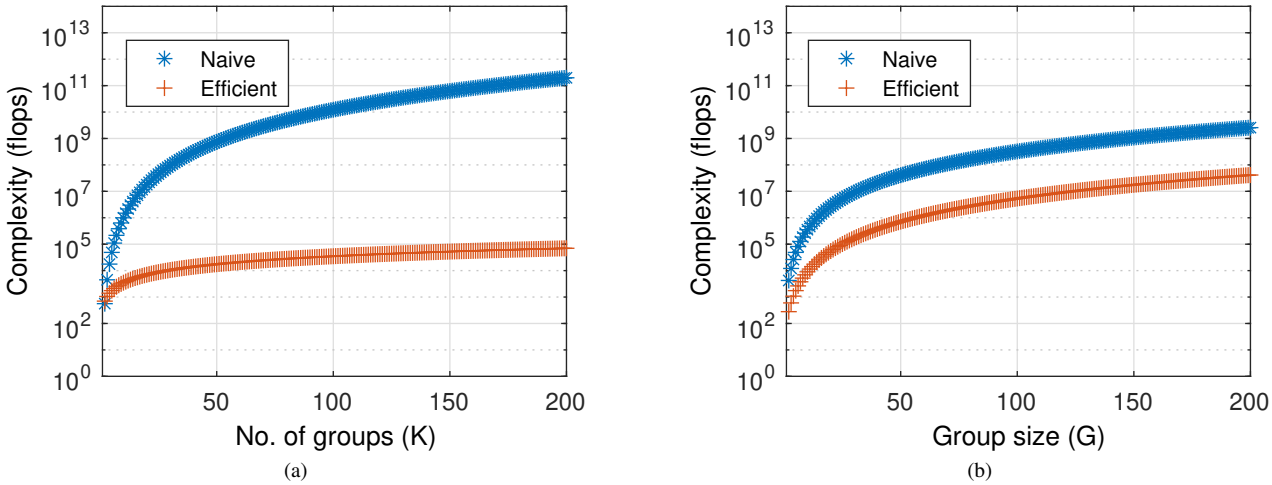

Fig. 1. Comparison between the naive computation (26) and efficient computation (16) of the group-utility of all K groups of size G in an LCMV beamformer. (a) $G = 5$, K is varied (b) $K = 5$, G is varied. The number of constraints is $Q = 2$ in both cases.

We now derive the expression for group-utility U_G as defined in (7). By plugging (6) into (4), it can be verified that $\mathbf{P}(\hat{\mathbf{w}}) = \mathbf{LTf}$, such that (7) becomes:

$$U_G = \mathbf{f}^T(\mathbf{L}_{-G} - \mathbf{L})\mathbf{f} \tag{26}$$

Now, we substitute (25) in (26) and use the fact that $\bar{\mathbf{w}}_G = \bar{\mathbf{T}}_G\mathbf{Lf}$ to obtain (16).

4. COMPLEXITY ANALYSIS

In Section 3.2, expression (16) was derived for the evaluation of group-utility which is significantly less complex than a naive computation by straightforward application of the definition (7), leading to (26). In this section we present a comparison of the computational complexity between the naive and efficient group-utility calculation in terms of floating-point operations (flops). A flop is an addition or a multiplication of two floating point numbers. For the analysis, we assume $\mathbf{R}_{YY}^{-1}$, $\mathbf{L}$, $\mathbf{T}$, and $\hat{\mathbf{w}}$, are already known from computing the full beamformer (11). We further assume the following number of flops associated to common matrix operations:

1. Inversion of an $M \times M$ matrix requires M^3 flops.
2. Multiplication of an $M \times N$ matrix with an $N \times P$ matrix requires $2MNP$ flops.
3. Addition or subtraction of two $M \times N$ matrices require MN flops.

Using these assumptions, the naive calculation of group-utility according to (26) for a single group of G variables requires $(M-G)^3+2(M-G)^2Q+2(M-G)Q^2+Q^3+3Q^2+2Q$ flops, which translates to an $O(K(K-1)^3G^3)$ complexity for all K groups. However, the efficient expression (16), requires only $G^3+G^2(3+2Q)+2G(Q^2+1)$ flops, which is of $O(KG^3)$ complexity when the group-utility computation of all K groups are considered.

In Fig. 1, the number of flops are plotted as a function of the number of groups K and the group size G for both the naive implementation (26) and the efficient implementation (16). In Fig. 1a, we fixed the group size G to $G = 5$ and varied the number of groups K from 2 to 200. It can be observed from this figure that when the number of groups $K >> G$, the efficient computation of group-utility using (16) requires significantly less flops than the naive computation using (26), up to 6 orders of magnitude difference. In most of the practical scenarios where sensor arrays are deployed, the number of groups (K) would be considerably larger than the number of input variables per group (G). This significant reduction in complexity is very useful in real-time tracking of the importance of sensor nodes using group-utility.

For the sake of completeness, we would like to note that, in the case of $G >> K$, the advantage of efficient computation is not as large as when $K >> G$. This is demonstrated in Fig. 1b where we fixed the number of groups to $K = 5$ and group size G was varied from 2 to 200. Here, the gain in complexity is smaller, yet the efficient computation still offers a computational advantage.

5. CONCLUSIONS

In this paper, we have defined a group-utility metric for the LCMV beamforming problem and derived an elegant expression to efficiently compute it. The efficient expression has significantly less computational complexity compared to a straightforward naive calculation of group-utility. We have also verified and demonstrated this by calculating complexity in terms of flops. Thus the online tracking of importance of sensors in a sensor array using group-utility can be carried out easily with minimal computational overhead.

6. REFERENCES

[1] H Kim and Mats Viberg, "Two decades of array signal processing research," *IEEE signal processing magazine*, vol. 13, no. 4, pp. 67–94, 1996.

[2] Ian F Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.

[3] Jie Zhang, Sundeep Prabhakar Chepuri, Richard Christian Hendriks, and Richard Heusdens, "Microphone subset selection for MVDR beamformer based noise reduction," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 3, pp. 550–563, 2017.

[4] A. M. Narayanan and A. Bertrand, "Analysis of miniaturization effects and channel selection strategies for EEG sensor networks with application to auditory attention detection," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 1, pp. 234–244, Jan 2020.

[5] Barry D Van Veen and Kevin M Buckley, "Beamforming: A versatile approach to spatial filtering," *IEEE assp magazine*, vol. 5, no. 2, pp. 4–24, 1988.

[6] Sharon Gannot, Emmanuel Vincent, Shmulik Markovich-Golan, Alexey Ozerov, Sharon Gannot, Emmanuel Vincent, Shmulik Markovich-Golan, and Alexey Ozerov, "A consolidated perspective on multimicrophone speech enhancement and source separation," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 25, no. 4, pp. 692–730, 2017.

[7] Alexander Bertrand and Marc Moonen, "Distributed LCMV beamforming in a wireless sensor network with single-channel per-node signal transmission," *IEEE Transactions on Signal Processing*, vol. 61, no. 13, pp. 3447–3459, 2013.

[8] Marijn Van Vliet, Nikolay Chumerin, Simon De Deyne, Jan Roelf Wiersema, Wim Fias, Gerrit Storms, and Marc M Van Hulle, "Single-trial ERP component analysis using a spatiotemporal LCMV beamformer," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 1, pp. 55–66, 2015.

[9] Alexander Bertrand, Joseph Szurley, Peter Ruckebusch, Ingrid Moerman, and Marc Moonen, "Efficient calculation of sensor utility and sensor removal in wireless sensor networks for adaptive signal estimation and beamforming," *IEEE Transactions on Signal Processing*, vol. 60, no. 11, pp. 5857–5869, 2012.

[10] Alexander Bertrand, "Utility metrics for assessing and selecting input variables in linear estimation algorithms," *IEEE Signal Processing Magazine*, vol. 35, no. 6, pp. 93–99, November 2018.

[11] S. Markovich-Golan, S. Gannot, and I. Cohen, "Low-complexity addition or removal of sensors/constraints in LCMV beamformers," *IEEE Transactions on Signal Processing*, vol. 60, no. 3, pp. 1205–1214, 2011.

[12] Gene Howard Golub and Charles F. Van Loan, *Matrix Computations*, Baltimore: The Johns Hopkins University Press, 3rd edition, 1996.