Region-Restricted Sensor Placement Based on Gaussian Process for Sound Field Estimation

Tomoya Nishida, *Member, IEEE*, Natsuki Ueno, *Member, IEEE*, Shoichi Koyama, *Member, IEEE*, and Hiroshi Saruwatari, *Member, IEEE*

***Abstract*—Sensor placement methods for field estimation based on Gaussian processes are proposed. Generally, sensor placement methods determine the appropriate placement positions by selecting them from predefined candidate positions. Many criteria for the selection have been proposed, with which the quality of the placements is evaluated with regard to the field at the candidate positions. This means that these sensor placement methods seek to find the positions that can estimate the field at the candidate positions accurately. In practical situations, however, the candidate sensor placement region can be different from the target region for field estimation. In this paper, to make sensor placement methods applicable to this situation, we propose two sensor placement methods based on the mean squared error and on conditional entropy that can be applied to cases in which the sensor placement region is arbitrarily restricted. After formulating the sensor placement problems, two approximate algorithms are derived: the greedy algorithm and the convex-relaxation-based algorithm. The application of the proposed methods to sound field estimation is also illustrated, and their effectiveness was confirmed through numerical experiments.**

***Index Terms*—Sensor placement, sound field estimation, Gaussian process, greedy algorithms, convex optimization.**

I. Introduction

ESTIMATING or interpolating physical fields, such as temperature [1], [2], acoustic field [3]–[5], and pollution [6], from values measured by sensors is an important problem with many applications. For example, sound field estimation is a problem used in many situations, such as reproducing the sound field using loudspeakers or headphones, visualization, and active noise control. To estimate the field, multiple sensors are distributed among the target region, and the entire field is estimated from their measurements.

The number of sensors that can be placed is often restricted owing to physical or economical restraints. In this case, the locations of the sensors greatly affect the estimation accuracy. Thus, one major problem in field estimation is how to determine the placement of the distributed sensors. To deal with this problem, extensive research studies on sensor placement methods have been conducted. In the context of acoustic problems, for example, sensor placement methods have been applied to problems such as sound field estimation [7], microphone array design [8], and sound field control [9], [10]. In most methods, the region of interest is first discretized and then the appropriate sensor locations up to a limited number are determined from these discretized locations. Many criteria for the choice of the locations have been proposed in the literature, which are based on observation models, such as Gaussian processes (GPs) [11]–[13] and finite-dimensional linear inverse problems [14]–[16]. Although there are various criteria, most of them evaluate the field at the discretized locations that are also used as the candidate sensor locations.

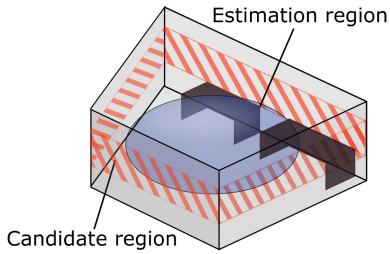

Fig. 1. Example of the sensor placement problem where the candidate placement region is separated from the estimation region.

When estimating fields in practical situations, however, the region for sensor placement, (i.e., the *candidate region*), might also be restricted and can be different from the target region for field estimation, (i.e., the *estimation region*). For example, when estimating sound fields in a room, there can be a situation in which the sensors (microphones) could be placed only around the wall, while the field interior of the room is to be estimated (Fig. 1). Since in previous methods the same discretized locations are used for candidate positions of sensor placement and evaluation positions for field estimation, they cannot be used for such situations. We will discuss this in detail in the next section.

A. Prior Work

Sensor placement methods can be classified by the model they assume for physical fields. In methods based on GPs [17], which

Manuscript received March 26, 2021; revised October 23, 2021 and January 20, 2022; accepted February 15, 2022. Date of publication March 3, 2022; date of current version April 11, 2022. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Florian Roemer. This work was supported by JST PRESTO under Grant JPMJPR18J4. *(Corresponding author: Tomoya Nishida.)*

Tomoya Nishida, Shoichi Koyama, and Hiroshi Saruwatari are with the Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-8656, Japan (e-mail: nisshee1996@gmail.com; koyama.shoichi@ieee.org; hiroshi_saruwatari@ipc.i.u-tokyo.ac.jp).

Natsuki Ueno is with the Graduate School of Systems Design, Tokyo Metropolitan University, Tokyo 191-0065, Japan (e-mail: natsuki.ueno@ieee.org).

Digital Object Identifier 10.1109/TSP.2022.3156012

we mainly focus on, it is assumed that all finite dimensional vectors extracted from any finite set of positions in the domain follow a Gaussian distribution. Using this model, Shewry and Wynn [12] proposed a method based on maximizing the entropy of the selected placement positions, which seeks to choose the most uncertain positions. In [11], the mutual information between the selected and unselected positions is to be maximized, which is equal to choosing the positions most informative about the field at unselected positions. In [13], the mean squared error of the estimation at the candidate locations was evaluated. By appropriately modeling the GP for sound fields with the kernel functions used for the interpolation method in [18], Ariga *et al.* [7] proposed an application of the mutual-information-based method to the sound field estimation problem. All these methods aim to estimate the field at the candidate placement positions; thus, the case where the candidate region and the estimation region are not identical is not considered.

Another major model describes the estimation problem as a finite linear inverse problem. The inverse problem is described as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{\theta} + \boldsymbol{v}, \tag{1}$$

where $\boldsymbol{y}$ is the observation, $\boldsymbol{\theta}$ is the parameter to be estimated, $\boldsymbol{H}$ is the measurement matrix, and $\boldsymbol{v}$ is the observation noise. Here, each element of $\boldsymbol{y}$ corresponds to each observation of candidate sensors. The parameter $\boldsymbol{\theta}$ can be the discretized field itself or any low-dimensional representation of the field. There are cases where a prior distribution for $\boldsymbol{\theta}$ is either assumed or not. By selecting the sensors, one can observe a subset of $\boldsymbol{y}$ and the parameter vector $\boldsymbol{\theta}$ is estimated from those observations. In the sensor selection problem for this model, the covariance matrix of the estimation error of $\boldsymbol{\theta}$ is evaluated and the cost functions are formulated as a scalarization of that error covariance matrix, many of which are derived from experimental design [14], [15]. For example, the trace that represents the mean squared error [19], the log determinant that represents the volume of the confidence region ellipsoid [16], [20], [21], and the maximum eigen value that represents the worst case variance of the estimation error [22] are the typical cost functions. Note that the applications of sensor selection methods based on linear inverse problems are not limited to field estimation; the methods are also applied in various contexts, including robotics [23] and target tracking [24]. When the field is modeled as a Gaussian process, the linear model in (1) can only be used by setting $\boldsymbol{\theta}$ as the field at candidate positions and $\boldsymbol{H}$ as an identity matrix. In this case, since estimating $\boldsymbol{\theta}$ can only obtain the field at the candidate positions, the field at the reconstruction domain cannot be estimated. Thus, the model cannot be straightforwardly used for cases where the candidate and estimation regions are independently and separately defined. There are also cases using nonlinear models, such as those in [25].

Aside from cost functions or the models used in field estimation, sensor placement methods can be distinguished by the algorithms used to approximately solve the problem, which is formulated as a combinatorial problem. There are mainly three strategies to obtain a solution to the combinatorial problem: greedy selection, convex relaxation, and heuristic approaches.

In greedy methods, the sensors are selected one by one, each time selecting the one that provides the best improvement in the cost function. For submodular cost functions, such as the entropy, mutual information [11], or the log determinant of the error covariance matrix [20], [21], the greedy algorithm is guaranteed to obtain a near-optimal solution with a fixed approximation factor [26]. Even when the cost function is not submodular nor supermodular, a degraded approximation factor for the greedy algorithm can be obtained [27]–[29] by measuring how far the cost function is from being submodular (or supermodular). The submodularity ratio and weak supermodularity have been introduced in [27] and [28], [29], respectively, and the approximation factors for subset selection for linear regression problems and sampling in graph signal reconstruction have been evaluated. We also note that there are extended algorithms such as the group greedy method [30].

Convex relaxation methods relax the combinatorial problem to a convex continuous optimization problem [16], [19], [24], [31], [32]. After solving the relaxed problem, the solution is rounded to a solution of the combinatorial problem. Although no performance guarantee for the solution can be given beforehand, the lower bound of the cost function can be obtained after conducting the method using the optimal value of the relaxed problem [16]. In some cases, the relaxed problem is reformulated to semidefinite programming (SDP) [19], [31], [33]. Both the relaxed problem itself and problems formulated as SDPs are typically solved using algorithms based on Newton's method, which is a second-order method [16], [19], [31], [33]. Although second-order methods converge to the optimal solution with a relatively small number of iterations, they require a high spatial complexity and are thus not applicable to large-scale problems.

The last strategy is to use heuristics [34]–[38]. Although methods using heuristics might produce good solutions, no guarantees or bounds for the solution can be obtained.

B. Our Contributions

We propose sensor placement methods for field estimation based on GPs, where the candidate and estimation regions can be independently set. In addition to providing methods to solve this as yet unsolved problem, our contributions can be summarized as follows.

- In the greedy algorithm, by using the notion of weak supermodularity, we provide near-optimal guarantees for the solution. The conditions under which the guarantees become tight are also discussed.
- In the convex-relaxation-based algorithm, by introducing the mirror descent algorithm, which is a first-order method, to solve the relaxed problem, we reduce the spatial complexity of the method and thus make it more applicable to large-scale problems than before.
- We apply the methods to the sound field estimation problem and show their applicability.

Preliminary results are given in [39], but a new cost function based on conditional entropy, the derivation of the weak supermodularity of the cost functions, the derivation of a new optimization algorithm for the convex relaxation method, and

the application to sound fields were added to this study. This paper is organized as follows. The problem of field estimation, using an independently defined candidate region and estimation region is introduced in Section II. Subsequently, two cost functions for sensor placement are proposed in Section III, and two approximate algorithms to obtain a solution to those problems are introduced in Section IV. The special case of sensor placement for sound field estimation is described in Section V, and numerical experiments for this case are described in Section VI.

C. Notations

In this paper, we use the following notations. Italic letters denote scalars, lower-case boldface letters denote vectors, and upper-case boldface letters denote matrices. Calligraphic letters denote sets. Subscripts of scalars, vectors, and matrices stand for their indexes or sets of indexes. For example, x_i is the ith entry of vector $\boldsymbol{x}$, and $\boldsymbol{X}_{\mathcal{A},\mathcal{B}}$ is the submatrix of $\boldsymbol{X}$ formed by rows and columns corresponding to $\mathcal{A}$ and $\mathcal{B}$, respectively. We also use $[\boldsymbol{X}]_{i,j}$ to denote the (i,j)th element of the matrix $\boldsymbol{X}$. The notations $\boldsymbol{X} \succeq \boldsymbol{0}$ and $\boldsymbol{X} \succeq \boldsymbol{Y}$ for matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively indicate that $\boldsymbol{X}$ and $\boldsymbol{X} - \boldsymbol{Y}$ are positive semidefinite.

II. Problem Statement

A. Field Estimation and Estimation Error

To describe the probabilistic field, we use complex GPs, which can be seen as a generalization of Gaussian distributions of finite-dimensional variables to functions. Let $\boldsymbol{u}_{\mathcal{A}}$ be a vector of the physical quantities of the field at any finite set of locations $\mathcal{A} = \{\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_J\} \subset D$, where D is the domain of a physical field, which is typically $\mathbb{R}^2$ or $\mathbb{R}^3$. Then, GP assumes that $\boldsymbol{u}_{\mathcal{A}}$ follows the complex Gaussian distribution

$$\boldsymbol{u}_{\mathcal{A}} \sim \mathcal{N}_{\mathbb{C}}\left(\bar{\boldsymbol{u}}_{\mathcal{A}}, \boldsymbol{K}_{\mathcal{A},\mathcal{A}}\right), \tag{2}$$

where the mean vector $\bar{\boldsymbol{u}}_{\mathcal{A}}$ is described as $\bar{\boldsymbol{u}}_{\mathcal{A}} = [f_u(\boldsymbol{r}_1), \ldots, f_u(\boldsymbol{r}_J)]^{\mathsf{T}} \in \mathbb{C}^J$ with the mean function $f_u(\cdot)$, and the covariance matrix $\boldsymbol{K}_{\mathcal{A},\mathcal{A}} \in \mathbb{C}^{J\times J}$ is the Gram matrix whose (i,j)th element is represented as $[\boldsymbol{K}_{\mathcal{A},\mathcal{A}}]_{i,j} = \kappa(\boldsymbol{r}_i, \boldsymbol{r}_j)$ with the positive definite kernel function $\kappa(\cdot,\cdot)$. The kernel function can be learned from the observed data [40] [41] or be designed using a priori information on the field [18]. A frequently used kernel function is the Gaussian kernel [17].

We consider estimating the field inside a predefined estimation region with sensors placed within a candidate region, where these two regions can be set independently and arbitrarily. To formulate the sensor placement problem, both regions are independently discretized, as illustrated in Fig. 2. We define the estimation locations as $\mathcal{E} = \{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_M\} \subset D$ and the candidate locations as $\mathcal{C} = \{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_N\} \subset D$. Note that these sets can share the same locations when the candidate and estimation regions have a common area. The locations of the selected sensors $\mathcal{S}$ would be a subset of $\mathcal{C}$. Thus, $\mathcal{S}$ can be described as $\mathcal{S} = \{\boldsymbol{q}_{j_1}, \ldots, \boldsymbol{q}_{j_K}\}$ when $|\mathcal{S}| = K$. The observations at $\mathcal{S}$, denoted as $\boldsymbol{y}_{\mathcal{S}}$, can be expressed as

$$\boldsymbol{y}_{\mathcal{S}} = \boldsymbol{u}_{\mathcal{S}} + \boldsymbol{v}_{\mathcal{S}}, \tag{3}$$

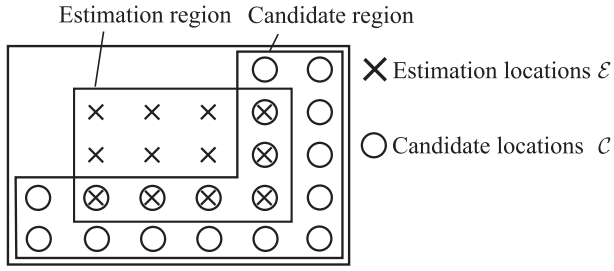

Fig. 2. Definitions of estimation and candidate locations.

where $\boldsymbol{v}_{\mathcal{S}}$ is the Gaussian noise of mean $\boldsymbol{0}$ and covariance $\lambda\boldsymbol{I}$, i.e., $\boldsymbol{v}_{\mathcal{S}} \sim \mathcal{N}_{\mathbb{C}}(\boldsymbol{0}, \lambda\boldsymbol{I})$.

We consider the problem of estimating the true field at $\mathcal{E}$ using the observations $\boldsymbol{y}_{\mathcal{S}}$, which is sampled from $\mathcal{C}$. From the GP model, the joint probability distribution of $\boldsymbol{u}_{\mathcal{E}}$ and $\boldsymbol{y}_{\mathcal{S}}$ is described by the Gaussian distribution

$$\begin{bmatrix}\boldsymbol{u}_{\mathcal{E}} \\ \boldsymbol{y}_{\mathcal{S}}\end{bmatrix} \sim \mathcal{N}_{\mathbb{C}}\left(\begin{bmatrix}\bar{\boldsymbol{u}}_{\mathcal{E}} \\ \bar{\boldsymbol{u}}_{\mathcal{S}}\end{bmatrix}, \begin{bmatrix}\boldsymbol{K}_{\mathcal{E},\mathcal{E}} & \boldsymbol{K}_{\mathcal{E},\mathcal{S}} \\ \boldsymbol{K}_{\mathcal{S},\mathcal{E}} & \boldsymbol{K}_{\mathcal{S},\mathcal{S}} + \lambda\boldsymbol{I}\end{bmatrix}\right), \tag{4}$$

where $\boldsymbol{K}_{\mathcal{E},\mathcal{S}} \in \mathbb{C}^{M\times K}$ is composed of the elements defined as $[\boldsymbol{K}_{\mathcal{E},\mathcal{S}}]_{i,k} = \kappa(\boldsymbol{p}_i, \boldsymbol{q}_{j_k})$ and $\boldsymbol{K}_{\mathcal{S},\mathcal{E}} \in \mathbb{C}^{K\times M}$ is also defined similarly. Thus, the posterior distribution of the true field given the observations also follows a Gaussian distribution given by

$$p\left(\boldsymbol{u}_{\mathcal{E}}|\boldsymbol{y}_{\mathcal{S}}\right) = \mathcal{N}_{\mathbb{C}}\left(\hat{\boldsymbol{u}}_{\mathcal{E}}, \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})\right), \tag{5}$$

$$\hat{\boldsymbol{u}}_{\mathcal{E}} = \bar{\boldsymbol{u}}_{\mathcal{E}} + \boldsymbol{K}_{\mathcal{E},\mathcal{S}}\left(\boldsymbol{K}_{\mathcal{S},\mathcal{S}} + \lambda\boldsymbol{I}\right)^{-1}\left(\boldsymbol{y}_{\mathcal{S}} - \bar{\boldsymbol{u}}_{\mathcal{S}}\right), \tag{6}$$

$$\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S}) = \boldsymbol{K}_{\mathcal{E},\mathcal{E}} - \boldsymbol{K}_{\mathcal{E},\mathcal{S}}\left(\boldsymbol{K}_{\mathcal{S},\mathcal{S}} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{K}_{\mathcal{S},\mathcal{E}}. \tag{7}$$

Here, $\hat{\boldsymbol{u}}_{\mathcal{E}}$ is the *maximum a posteriori* (MAP) estimate of the field at $\mathcal{E}$, and the covariance matrix $\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})$ can also be interpreted as the covariance matrix of the estimation error $\boldsymbol{n}_{\mathcal{E}} = \hat{\boldsymbol{u}}_{\mathcal{E}} - \boldsymbol{u}_{\mathcal{E}}$. Note that $\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})$ is determined only by the sensor selection $\mathcal{S}$, and not by the observations. In the following Sects. III and IV, optimization problems for sensor placement and approximation algorithms are formulated using this estimation error covariance matrix, which is a similar strategy to existing sensor placement methods.

III. Cost Functions for Sensor Placement

In this section, we propose two cost functions for optimizing the sensor placement: the mean-squared-error-based cost function and the conditional-entropy-based cost function, which are commonly used criteria in the literature.

A. Sensor Placement Based on Mean Squared Error

We propose the sensor placement problem as an optimization problem formulated as

$$\underset{\mathcal{S}\subset\mathcal{C}}{\text{minimize}}\ \operatorname{tr}\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})\right)\ \ \text{s.t.}\ |\mathcal{S}| = K. \tag{8}$$

Since $\operatorname{tr}(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})) = \operatorname{tr}(\mathrm{E}[\boldsymbol{n}_{\mathcal{E}}\boldsymbol{n}_{\mathcal{E}}^{\mathsf{H}}]) = \mathrm{E}[\boldsymbol{n}_{\mathcal{E}}^{\mathsf{H}}\boldsymbol{n}_{\mathcal{E}}]$, this optimization problem requires the minimization of the (expected) mean squared error at the estimation positions. In order to make approximation algorithms easily applicable, we reformulate this

sensor placement problem using a binary-valued vector $\boldsymbol{w} \in \{0,1\}^N$ that represents the selected positions, where $w_j = 1$ if $\boldsymbol{q}_j \in \mathcal{S}$. Let $\boldsymbol{\Phi}_{\mathcal{S}} \in \{0,1\}^{N \times K}$ be a binary-valued matrix that extracts the rows of the sensor indexes of $\mathcal{S}$ by multiplication from the left. The covariance matrix $\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})$ in (7) can be rewritten as

$$\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S}) = \boldsymbol{K}_{\mathcal{E},\mathcal{E}} - \boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{\Phi}_{\mathcal{S}} \left(\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{\Phi}_{\mathcal{S}} + \lambda \boldsymbol{I}\right)^{-1} \boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}. \tag{9}$$

Then, by using the matrix inversion lemma and $\boldsymbol{\Phi}_{\mathcal{S}}\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}} = \mathrm{diag}\,(\boldsymbol{w})$, where $\mathrm{diag}\,(\boldsymbol{w})$ is a diagonal matrix with the diagonal components $\boldsymbol{w}$, we obtain

$$\begin{aligned}
&\boldsymbol{\Phi}_{\mathcal{S}} \left(\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{\Phi}_{\mathcal{S}} + \lambda \boldsymbol{I}\right)^{-1} \boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}} \\
&= \boldsymbol{\Phi}_{\mathcal{S}} \big(\lambda^{-1}\boldsymbol{I} \\
&\quad -\lambda^{-1}\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}} \left(\boldsymbol{I} + \lambda^{-1}\boldsymbol{\Phi}_{\mathcal{S}}\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\right)^{-1} \boldsymbol{\Phi}_{\mathcal{S}}\lambda^{-1}\Big) \boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}} \\
&= \lambda^{-1}\boldsymbol{\Phi}_{\mathcal{S}}\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}} \\
&\quad \cdot \left(\boldsymbol{I} - \boldsymbol{K}_{\mathcal{C},\mathcal{C}} \left(\boldsymbol{I} + \lambda^{-1}\boldsymbol{\Phi}_{\mathcal{S}}\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\right)^{-1} \lambda^{-1}\boldsymbol{\Phi}_{\mathcal{S}}\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}}\right) \\
&= \lambda^{-1}\boldsymbol{\Phi}_{\mathcal{S}}\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}} \left(\boldsymbol{I} + \lambda^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{\Phi}_{\mathcal{S}}\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}}\right)^{-1} \\
&= \mathrm{diag}\,(\boldsymbol{w}) \left(\lambda \boldsymbol{I} + \boldsymbol{K}_{\mathcal{C},\mathcal{C}}\,\mathrm{diag}\,(\boldsymbol{w})\right)^{-1}.
\end{aligned} \tag{10}$$

Using this formula, we reformulate the error covariance matrix using $\boldsymbol{w}$ as

$$\begin{aligned}
&\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S}) \\
&= \boldsymbol{K}_{\mathcal{E},\mathcal{E}} - \boldsymbol{K}_{\mathcal{E},\mathcal{C}}\,\mathrm{diag}\,(\boldsymbol{w}) \left(\lambda \boldsymbol{I} + \boldsymbol{K}_{\mathcal{C},\mathcal{C}}\,\mathrm{diag}\,(\boldsymbol{w})\right)^{-1} \boldsymbol{K}_{\mathcal{C},\mathcal{E}}.
\end{aligned} \tag{11}$$

We also note that a similar formulation can be derived when the noise is correlated, i.e., $\boldsymbol{v}_{\mathcal{S}} \sim \mathcal{N}_{\mathbb{C}}(\boldsymbol{0}, \boldsymbol{R})$, where $\boldsymbol{R} \succeq \boldsymbol{0}$, as in [39].

Using $\mathrm{tr}(\boldsymbol{AB}) = \mathrm{tr}(\boldsymbol{BA})$, we can formulate the sensor placement problem on the basis of the mean squared error as

$$\underset{\boldsymbol{w} \in \{0,1\}^N}{\text{minimize}}\ J_{\text{MSE}}(\boldsymbol{w})\ \text{ s.t. } \boldsymbol{1}^{\mathsf{T}}\boldsymbol{w} = K, \tag{P1}$$

where the cost function is defined as

$$J_{\text{MSE}}(\boldsymbol{w}) = \mathrm{tr}\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})\right). \tag{12}$$

Since there is a one-to-one correspondence between $\boldsymbol{w}$ and $\mathcal{S}$, we can also describe the cost function as $J_{\text{MSE}}(\mathcal{S})$. We will use either of the notions in later sections, depending on its convenience.

We note that when the candidate and estimation positions are identical, i.e., $\mathcal{C} = \mathcal{E}$, the proposed problem corresponds to that in [13]. Thus, the proposed problem can be seen as a natural extension of that in [13]. Also, compared with mean-squared-error-based sensor selection problems for finite-dimensional linear inverse problems [19], the proposed method for $\mathcal{C} = \mathcal{E}$ corresponds to that for the case where the measurement matrix is an identity matrix.

B. Sensor Placement Based on Entropy

Many sensor placement methods using cost functions related to entropy, such as informational gain and mutual information, have been proposed in the literature [11], [42]. This is natural considering the fact that we want to select the most "informative" locations. In this section, we propose a cost function based on entropy in the case where $\mathcal{C}$ and $\mathcal{E}$ are independently defined. Since we want to estimate $\boldsymbol{u}_{\mathcal{E}}$ using the observed data $\boldsymbol{y}_{\mathcal{S}}$, the problem will be

$$\underset{\mathcal{S} \subset \mathcal{C}}{\text{minimize}}\ H(\boldsymbol{u}_{\mathcal{E}} \mid \boldsymbol{y}_{\mathcal{S}}) \text{ s.t. } |\mathcal{S}| = K. \tag{13}$$

$H(\boldsymbol{u}_{\mathcal{E}} \mid \boldsymbol{y}_{\mathcal{S}})$ is the conditional entropy of $\boldsymbol{u}_{\mathcal{E}}$ given $\boldsymbol{y}_{\mathcal{S}}$. By evaluating the entropy of a Gaussian random variable, we can reformulate this optimization problem as

$$\underset{\mathcal{S} \subset \mathcal{C}}{\text{minimize}}\ J_{\text{ENT}}(\mathcal{S}) \text{ s.t. } |\mathcal{S}| = K, \tag{P2}$$

where the cost function is defined as

$$J_{\text{ENT}}(\mathcal{S}) = \log\det\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})\right). \tag{14}$$

Using the notion of the covariance matrix in (11), we can also describe the cost function as $J_{\text{ENT}}(\boldsymbol{w})$, which we will also use in later sections. Note that in the context of experimental design for finite-dimensional inverse problems, minimizing the log determinant of the covariance matrix is named D-optimal design and is equivalent to minimizing the volume of the confidence region ellipsoid [16].

To confirm that the cost function we proposed is appropriate for our issue of interest, we discuss other possible cost functions related to entropy by applying existing criteria to the $\mathcal{C} \neq \mathcal{E}$ case.

First, using directly the entropy criteria can be considered, where the entropy at the selected locations $H(\boldsymbol{u}_{\mathcal{S}})$ or $H(\boldsymbol{y}_{\mathcal{S}})$ is to be maximized. Since this criteria can be described as

$$H(\boldsymbol{u}_{\mathcal{S}}) = H(\boldsymbol{u}_{\mathcal{C}}) - H(\boldsymbol{u}_{\mathcal{C}\backslash\mathcal{S}} \mid \boldsymbol{u}_{\mathcal{S}}), \tag{15}$$

it is equivalent to minimizing $H(\boldsymbol{u}_{\mathcal{C}\backslash\mathcal{S}} \mid \boldsymbol{u}_{\mathcal{S}})$ and thus can only consider $\mathcal{C}$, not $\mathcal{E}$.

Next, we consider modifying the criteria such as the conditional entropy and mutual information in a formal manner by simply replacing the candidate positions $\mathcal{C}$ with the estimation positions $\mathcal{E}$. Then, the cost functions can be expressed as

$$H(\boldsymbol{u}_{\mathcal{E}\backslash\mathcal{S}} \mid \boldsymbol{u}_{\mathcal{S}}) = H(\boldsymbol{u}_{\mathcal{E}\cup\mathcal{S}}) - H(\boldsymbol{u}_{\mathcal{S}}), \tag{16}$$

$$I(\boldsymbol{u}_{\mathcal{E}\backslash\mathcal{S}}; \boldsymbol{u}_{\mathcal{S}}) = H(\boldsymbol{u}_{\mathcal{E}\backslash\mathcal{S}}) - H(\boldsymbol{u}_{\mathcal{E}\backslash\mathcal{S}} \mid \boldsymbol{u}_{\mathcal{S}}), \tag{17}$$

where $I(\boldsymbol{u}_{\mathcal{E}\backslash\mathcal{S}}; \boldsymbol{u}_{\mathcal{S}})$ denotes the mutual information between $\boldsymbol{u}_{\mathcal{E}\backslash\mathcal{S}}$ and $\boldsymbol{u}_{\mathcal{S}}$. Although these cost functions might also seem appropriate for our problem, the dimension of $\boldsymbol{u}_{\mathcal{E}\backslash\mathcal{S}}$ varies among the same number of selected locations $|\mathcal{S}|$, since $|\mathcal{E} \cap \mathcal{S}|$ does not always take the same number. Thus, these cost functions become meaningless. Note that this issue did not occur in the $\mathcal{C} = \mathcal{E}$ case, since $\mathcal{S} \subset \mathcal{E} = \mathcal{C}$ and the dimension of $\boldsymbol{u}_{\mathcal{E}\backslash\mathcal{S}}$ is always $|\mathcal{E}\backslash\mathcal{S}| = M - K$. Considering these facts, compared with these cost functions, the proposed cost function can be seen as appropriate.

Algorithm 1: Greedy Algorithm.

Require: $\mathcal{S}^{(0)} = \emptyset$ and $\boldsymbol{M}_{\boldsymbol{w}}^{-1} = \boldsymbol{K}_{\mathcal{C},\mathcal{C}}$
  for $k = 1, \ldots, K-1$ **do**
    Find $\boldsymbol{q} \in \mathcal{C}\backslash\mathcal{S}$ that maximizes $J(\mathcal{S}^{(k)}) - J(\mathcal{S}^{(k)} \cup \{\boldsymbol{q}\})$ using (19) or (20), and denote it as $\boldsymbol{q}_{j_{(k+1)}}$.
    Update sensor selection $\mathcal{S}^{(k+1)} = \mathcal{S}^{(k)} \cup \{\boldsymbol{q}_{j_{(k+1)}}\}$.
    Update required inverse matrices using (24) or (25).
  end for

IV. Algorithms

In this section, we propose two algorithms to obtain an approximate solution to (P1) and (P2): the greedy method and the convex-relaxation-based method.

A. Greedy Method

The greedy algorithm selects the sensor location that minimizes the cost function one by one to obtain an approximate solution with a low cost. In this section, we first describe the greedy algorithm and its efficient computation. Then, we provide the performance guarantees of the algorithm using the concept of supermodularity ratio.

Let the set of selected sensor locations at the kth step of the method be $\mathcal{S}^{(k)}$. As mentioned above, at the $k+1$th step of the greedy algorithm, the sensor placement that minimizes the cost function $J(\mathcal{S}^{(k)})$, i.e.,

$$\boldsymbol{q}_{j_{(k+1)}} = \underset{\boldsymbol{q}\in\mathcal{C}\backslash\mathcal{S}}{\arg\max}\ J(\mathcal{S}^{(k)}) - J(\mathcal{S}^{(k)} \cup \{\boldsymbol{q}\}), \tag{18}$$

is selected and added to the selected set $\mathcal{S}^{(k)}$. Here, $J(\mathcal{S})$ denotes either $J_{\mathrm{MSE}}(\mathcal{S})$ or $J_{\mathrm{ENT}}(\mathcal{S})$. Because of its simplicity, the greedy algorithm has been applied to many sensor placement problems [11], [20], [43]. Although the cost function $J(\mathcal{S})$ requires the calculation of an inverse matrix, which seems to increase the computational complexity at first glance, this can be avoided using the rank-1 update of an inverse matrix.

The resulting decrease in the cost functions can be calculated as

$$J_{\mathrm{MSE}}(\mathcal{S}) - J_{\mathrm{MSE}}(\mathcal{S} \cup \{\boldsymbol{q}_j\}) = b_j \left[\boldsymbol{P}_{\boldsymbol{w}} \boldsymbol{K}_{\mathcal{C},\mathcal{E}} \boldsymbol{K}_{\mathcal{E},\mathcal{C}} \boldsymbol{P}_{\boldsymbol{w}}^{\mathsf{H}}\right]_{j,j}, \tag{19}$$

$$J_{\mathrm{ENT}}(\mathcal{S}) - J_{\mathrm{ENT}}\left(\mathcal{S} \cup \{\boldsymbol{q}_j\}\right) = -\log\left(1 - b_j \boldsymbol{a}_j^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})^{-1} \boldsymbol{a}_j\right), \tag{20}$$

where $\boldsymbol{P}_{\boldsymbol{w}}$, $\boldsymbol{a}_j$, and b_j are respectively defined as

$$\boldsymbol{P}_{\boldsymbol{w}} = \left(\lambda \boldsymbol{I} + \boldsymbol{K}_{\mathcal{C},\mathcal{C}} \operatorname{diag}(\boldsymbol{w})\right)^{-1}, \tag{21}$$

$$\boldsymbol{a}_j = \boldsymbol{K}_{\mathcal{E},\mathcal{C}} \boldsymbol{P}_{\boldsymbol{w}}^{\mathsf{H}} \boldsymbol{e}_j, \tag{22}$$

$$b_j = \frac{\lambda}{1 + \left[\boldsymbol{P}_{\boldsymbol{w}} \boldsymbol{K}_{\mathcal{C},\mathcal{C}}\right]_{j,j}}. \tag{23}$$

Here, $\boldsymbol{e}_j$ denotes the unit vector with the jth element being equal to 1. See Appendix A for the derivation of (19) to (23). Note that both cost functions decrease monotonically, which means that they always decrease with an addition of a sensor. This is because $\boldsymbol{P}_{\boldsymbol{w}} \boldsymbol{K}_{\mathcal{C},\mathcal{C}}$ is semidefinite (see Appendix B). The inverse matrices in (20) and (21) can also be calculated efficiently by using the following rank-1 updates:

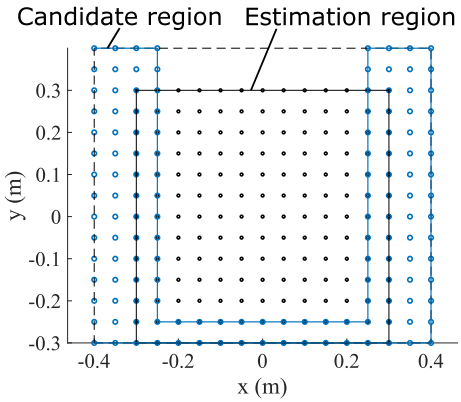

Fig. 3. Estimation positions and candidate sensor positions for numerical experiments. The blue and black dots represent $\mathcal{C}$ and $\mathcal{E}$, respectively.

$$\boldsymbol{P}_{\boldsymbol{w}+\boldsymbol{e}_j} = \boldsymbol{P}_{\boldsymbol{w}} - \frac{\boldsymbol{P}_{\boldsymbol{w}} \boldsymbol{K}_{\mathcal{C},\mathcal{C}} \boldsymbol{e}_j \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{P}_{\boldsymbol{w}}}{1 + \left[\boldsymbol{P}_{\boldsymbol{w}} \boldsymbol{K}_{\mathcal{C},\mathcal{C}}\right]_{j,j}}, \tag{24}$$

$$\begin{aligned}&\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S} \cup \{\boldsymbol{q}_j\})^{-1} \\ &= \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})^{-1} \\ &\quad + \frac{b_j}{1 - b_j \boldsymbol{a}_j^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})^{-1} \boldsymbol{a}_j} \cdot \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})^{-1} \boldsymbol{a}_j \boldsymbol{a}_j^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})^{-1}.\end{aligned} \tag{25}$$

The algorithm is summarized in Algorithm 1.

Owing to the efficient calculation, the computational complexities for both cost functions are $O(N^2)$ for each $\boldsymbol{q} \in \mathcal{C}\backslash\mathcal{S}$, whereas calculating the cost functions directly will cost $O(N^3)$. We also note that the complexity is the same as that in the $\mathcal{C} = \mathcal{E}$ case, and no increase in the complexity occurs when $\mathcal{C}$ and $\mathcal{E}$ are independently defined.

Now, we give the performance guarantee for the greedy algorithm. Recently, a way to derive bounds for the suboptimality of the greedy algorithm for non-submodular (supermodular) functions using the notion of approximate submodularity (supermodularity) was proposed [27], [29]. By using the definition of supermodularity ratio held in [29], we will establish approximation bounds for the greedy algorithm used for (P1) and (P2).

A set function $J : 2^{\mathcal{C}} \to \mathbb{R}$ is α-supermodular if

$$J(\mathcal{A} \cup \{\boldsymbol{q}\}) - J(\mathcal{A}) \leq \alpha \left(J(\mathcal{B} \cup \{\boldsymbol{q}\}) - J(\mathcal{B})\right) \quad \forall \mathcal{A} \subseteq \mathcal{B} \subseteq \mathcal{C}, \forall \boldsymbol{q} \notin \mathcal{B}, \tag{26}$$

for $\alpha \geq 0$. This concept can be seen as an extension of supermodularity, since if $\alpha \geq 1$, J is supermodular. α represents how much the function degrades from being supermodular. When J monotonically decreases, our interest is in the largest α that

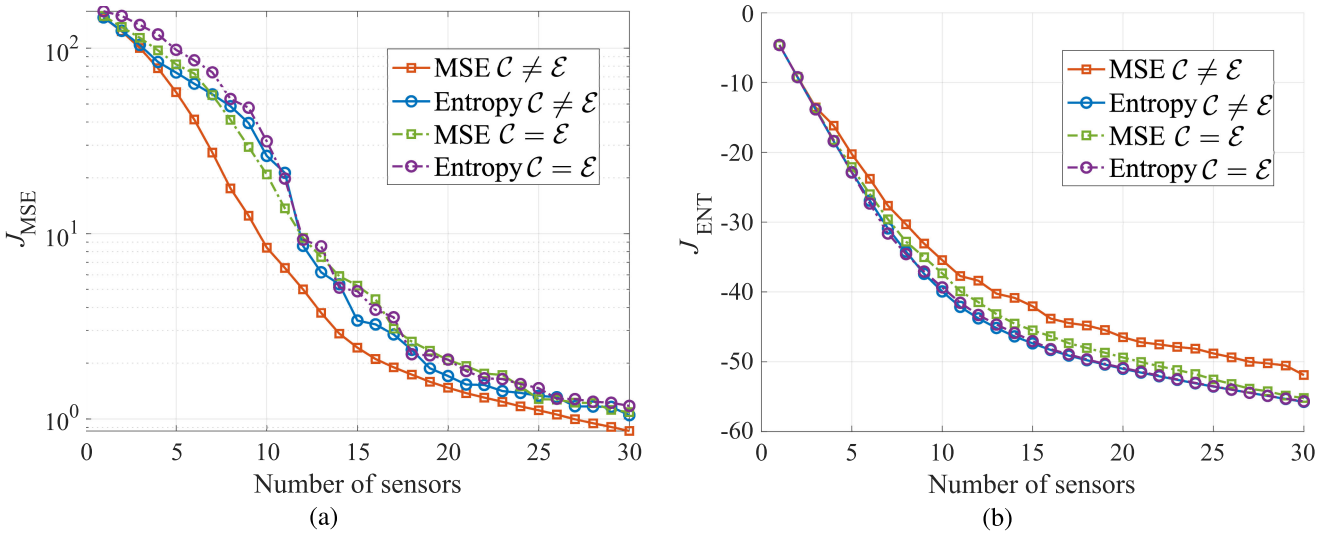

Fig. 4. Values of cost functions with respect to number of sensors for greedy algorithms. The case for 600 Hz. (a) MSE cost; (b) Entropy cost.

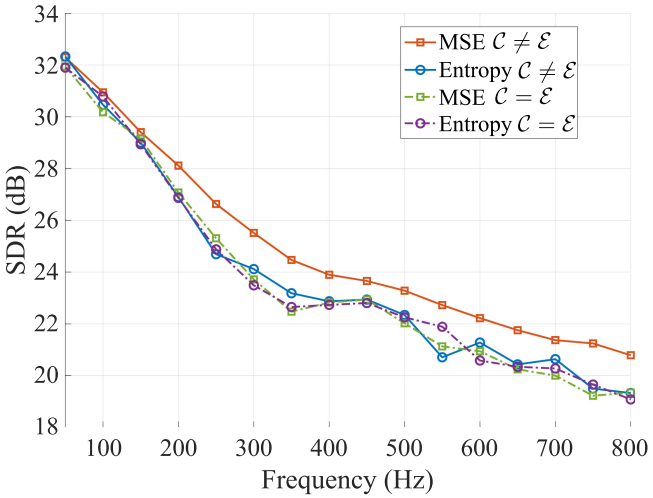

Fig. 5. SDR for greedy algorithms. The case for 24 sensors.

satisfies (26), i.e.,

$$\alpha = \min \left\{ \frac{J(\mathcal{A} \cup \{\boldsymbol{q}\}) - J(\mathcal{A})}{J(\mathcal{B} \cup \{\boldsymbol{q}\}) - J(\mathcal{B})} | \mathcal{A} \subseteq \mathcal{B} \subseteq \mathcal{C},\ \boldsymbol{q} \notin \mathcal{B} \right\}. \tag{27}$$

Using this parameter α, the suboptimality of the greedy algorithm is established in the following theorem.

Theorem 1: [29] Let $J(\mathcal{S})$ be a monotonically decreasing and α-supermodular function, and J^* be the optimal solution of the problem

$$\underset{\mathcal{S} \subset \mathcal{C}, |\mathcal{S}| \leq K}{\text{minimize}}\, J(\mathcal{S}). \tag{28}$$

Then, the solution of the greedy algorithm $\mathcal{S}^{(k)}$ $(|\mathcal{S}| = k)$ satisfies

$$\frac{J(\mathcal{S}^{(k)}) - J^*}{J(\emptyset) - J^*} \leq e^{-\alpha k / K}. \tag{29}$$

Proof: See [29]. □

Here, we show that $J_{\mathrm{MSE}}(\mathcal{S})$ and $J_{\mathrm{ENT}}(\mathcal{S})$ are α-supermodular functions in the following theorems.

Theorem 2: Suppose that $\boldsymbol{K}_{\mathcal{C},\mathcal{C}}$ is positive definite. J_{MSE} is α-supermodular with

$$\alpha \geq \frac{\lambda_{\min}(\bar{\boldsymbol{K}})}{\lambda_{\max}(\bar{\boldsymbol{K}})} \cdot \frac{\lambda_{\min}(\boldsymbol{K}_{\mathcal{C},\mathcal{C}})}{\lambda_{\max}(\boldsymbol{K}_{\mathcal{C},\mathcal{C}})} \cdot \frac{1}{1 + \lambda^{-1} \lambda_{\min}(\boldsymbol{K}_{\mathcal{C},\mathcal{C}})}, \tag{30}$$

where $\lambda_{\min}(\boldsymbol{X})$ and $\lambda_{\max}(\boldsymbol{X})$ denote the minimum and maximum eigen values of $\boldsymbol{X}$, respectively, and $\bar{\boldsymbol{K}} = \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1} \boldsymbol{K}_{\mathcal{C},\mathcal{E}} \boldsymbol{K}_{\mathcal{E},\mathcal{C}} \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}$.

Proof: See Appendix C. □

Theorem 3: Suppose that $\boldsymbol{K}_{\mathcal{C},\mathcal{C}}$ and $\boldsymbol{K}_{\mathcal{E},\mathcal{E}}$ is positive definite. J_{ENT} is α-supermodular with

$$\alpha \geq \frac{\lambda_{\min}\left(\boldsymbol{K}_{\mathcal{E},\mathcal{E}}\right)}{\lambda_{\max}\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{C})\right)} \cdot \frac{\lambda_{\min}\left(\bar{\boldsymbol{K}}\right)}{\lambda_{\max}\left(\bar{\boldsymbol{K}}\right)} \frac{\lambda_{\min}(\boldsymbol{K}_{\mathcal{C},\mathcal{C}})}{\lambda_{\max}(\boldsymbol{K}_{\mathcal{C},\mathcal{C}})} \cdot \frac{1}{1 + \lambda^{-1} \lambda_{\min}(\boldsymbol{K}_{\mathcal{C},\mathcal{C}})} \cdot \frac{1}{1 + \lambda^{-1} \max_j [\boldsymbol{K}_{\mathcal{C},\mathcal{C}}]_{j,j}}. \tag{31}$$

Proof: See Appendix D. □

Thus, using these bounds on α and Theorem 1, we can obtain the performance guarantee for the greedy algorithms. We also describe the situations in which the guarantees would be good. Since the bound includes the inverse of the condition number of $\bar{\boldsymbol{K}}$ and $\boldsymbol{K}_{\mathcal{C},\mathcal{C}}$, when these condition numbers are smaller, the guarantees would be better. In particular, when $\mathcal{C} = \mathcal{E}$, the condition number of $\bar{\boldsymbol{K}} = \boldsymbol{I}$ would be 1, and thus the performance guarantee would be better than the $\mathcal{C} \neq \mathcal{E}$ case. On the other hand, when the number of candidate positions is larger than the number of evaluation positions, i.e., $|\mathcal{C}| > |\mathcal{E}|$, the number of rows of $\boldsymbol{K}_{\mathcal{C},\mathcal{E}}$ would be larger than that of columns and $\bar{\boldsymbol{K}}$ becomes singular. In this case, only a trivial bound $\alpha \geq 0$ can be obtained where there is no information about the performance guarantee. Also, when $\boldsymbol{K}_{\mathcal{C},\mathcal{C}}$ gets close to singular, i.e., $\lambda_{\min}$ gets close to 0, the bound will get close to the trivial bound $\alpha \geq 0$. As λ gets smaller, i.e., as the noise decreases, the bounds get worse.

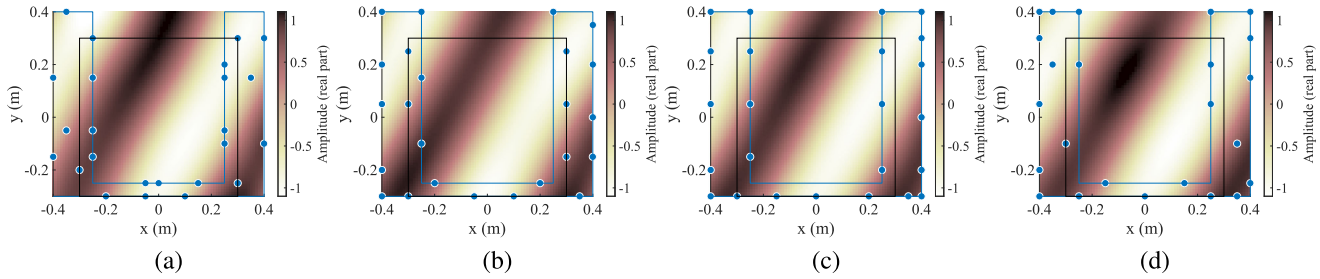

Fig. 6. Reconstructed pressure distributions at 600 Hz with 24 sensors chosen by the greedy algorithm for the narrowband case. (a) MSE $\mathcal{C} \neq \mathcal{E}$; (b) MSE $\mathcal{C} = \mathcal{E}$; (c) Entropy $\mathcal{C} \neq \mathcal{E}$; (d) Entropy $\mathcal{C} = \mathcal{E}$.

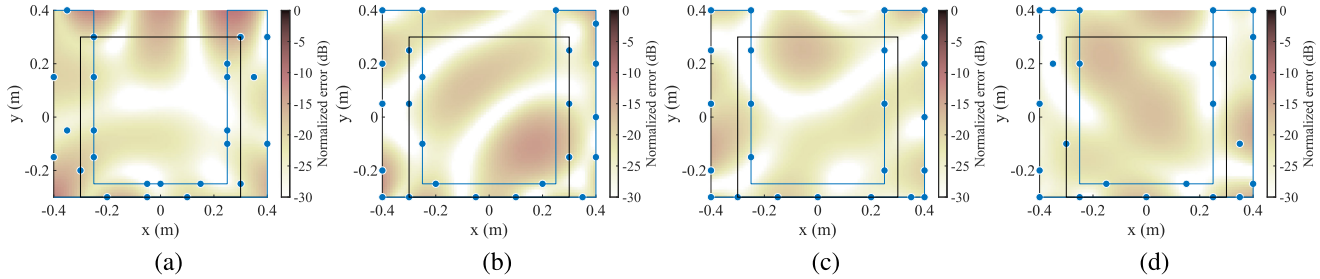

Fig. 7. Normalized error distributions at 600 Hz with 24 sensors chosen by the greedy algorithm for the narrowband case. (a) MSE $\mathcal{C} \neq \mathcal{E}$; (b) MSE $\mathcal{C} = \mathcal{E}$; (c) Entropy $\mathcal{C} \neq \mathcal{E}$; (d) Entropy $\mathcal{C} = \mathcal{E}$. The SDRs were (a) 23.6, (b) 20.5, (c) 23.4, and (d) 21.5 dB.

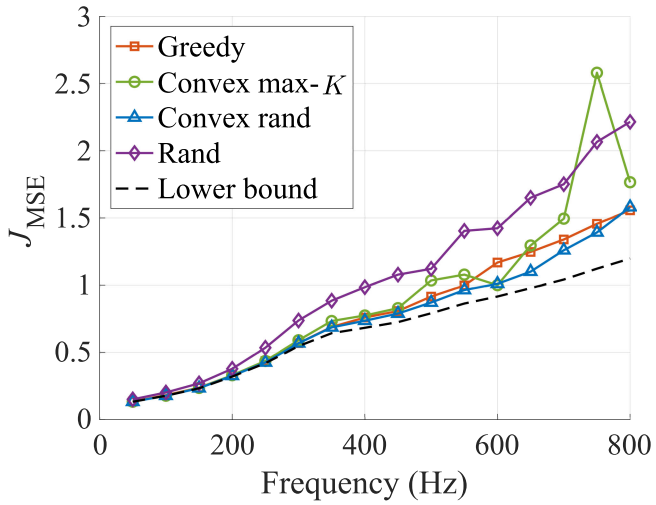

Fig. 8. Cost function values of MSE $\mathcal{C} \neq \mathcal{E}$ for each algorithm for 24 sensors. Lower bound stands for the lower bound of the cost function obtained from the convex relaxation algorithm.

Algorithm 2: Mirror Descent Algorithm for Solving (P3).

Require: Initial variable $\boldsymbol{w}^{(0)} \in [0,1]^N$, step sizes $\{\alpha^{(l)}\}_{l=1}^{\infty}$, and $l = 0$.
while Stopping condition is not satisfied **do**
$l \leftarrow l + 1$.
$y_j^{(l+1)} = w_j^{(l)} \exp\left(-\alpha^{(l)} g_j^{(l)}\right)$ $(j = 1, \ldots, N)$.
$\bar{w}_j^{(l+1)} = \frac{K}{\|\boldsymbol{y}^{(l+1)}\|_1} y_j^{(l+1)}$ $(j = 1, \ldots, N)$.
while $\bar{\boldsymbol{w}}^{(l+1)} \notin [0,1]^N$ **do**
Set $L = |\{\bar{w}_j^{(l+1)} \mid \bar{w}_j^{(l+1)} \geq 1,\ j = 1, \ldots, N\}|$.
Compute

$$\bar{w}_j^{(l+1)} \leftarrow \begin{cases} 1, & (\bar{w}_j^{(l+1)} \geq 1) \\ \frac{K}{K-L} \bar{w}_j^{(l+1)}, & (\text{otherwise}), \end{cases} \tag{32}$$

for $j = 1, \ldots, N$.
end while
Set $\boldsymbol{w}^{(l+1)} = \bar{\boldsymbol{w}}^{(l+1)}$.
end while

B. Convex Relaxation Method

The convex relaxation method relaxes combinatorial optimization problems to continuous convex optimization problems, which can be globally solved. By using this globally optimal solution, we can obtain an approximate solution to the original problem. For the proposed problems, we relax the boolean constraint $\boldsymbol{w} \in \{0,1\}^N$ in (P1) and (P2) to $\boldsymbol{w} \in [0,1]^N$ to obtain a continuous optimization problem. Then, both problems become convex optimization problems. By assuming that the optimal solution of the original problem lies near the optimal solution of the relaxed one, we apply some sort of rounding algorithm, such as those in [16], [25] and [19], after solving the relaxed problem to obtain a near optimal solution. The advantage of this method is that we can always obtain the global optimal solution to the relaxed problem, which can be used as a lower bound of the cost function in the original problem. By comparing this lower bound with the cost function value of the rounded solution, we can evaluate the optimality of the solution we have obtained. This procedure was thoroughly investigated in [16].

By relaxing the restriction on $\boldsymbol{w}$ from $\boldsymbol{w} \in \{0,1\}^N$ to $\boldsymbol{w} \in [0,1]^N$, we formulate the continuous optimization problem

$$\underset{\boldsymbol{w}}{\text{minimize}}\ J(\boldsymbol{w}) \quad \text{s.t. } \mathbf{0} \leq \boldsymbol{w} \leq \mathbf{1},\ \mathbf{1}^{\mathsf{T}}\boldsymbol{w} = K, \tag{P3}$$

where $\mathbf{0} \leq \boldsymbol{w} \leq \mathbf{1}$ denotes that all the elements of $\boldsymbol{w}$ satisfy $0 \leq w_j \leq 1$ for $j = 1, \ldots, N$. Here, $J(\boldsymbol{w})$ denotes either $J_{\text{MSE}}(\boldsymbol{w})$ or $J_{\text{ENT}}(\boldsymbol{w})$. Under the constraints in (P3), these cost functions can be proven to be convex (see Appendix C). Thus, the optimal solution can be obtained using convex optimization algorithms.

As mentioned in Section I, conventional convex relaxation methods were mainly based on second-order optimization algorithms, such as Newton's method. However, second-order methods typically become impractical when dealing with large-scale problems. Moreover, when the relaxed problem is reformulated to SDP, extra optimization variables are introduced, which makes the spatial complexity larger than necessary. To make convex relaxation methods applicable to large-scale problems, we here propose a first-order algorithm based on the mirror descent algorithm (MDA) [44]. One can also use zeroth-order algorithms such as in [45] to avoid the calculation of the gradients. We also note that other first-order optimization algorithms, such as the primal-dual splitting algorithm [46], can also be applied to the relaxed problem. In this paper, we use MDA because of its convenience in two ways: 1) the constraints of (P3) are exactly satisfied at each iteration, whereas the primal-dual algorithm only converges to a solution that satisfies the constraints, and 2) MDA has fewer parameters to be optimized.

MDA solves the convex optimization problem

$$\underset{\boldsymbol{w} \in C}{\text{minimize}}\ J(\boldsymbol{w}). \tag{33}$$

MDA assumes a function $h(\boldsymbol{w})$ that is differentiable and strongly convex over some norm, and derives the Bregman divergence from h as

$$D_h(\boldsymbol{w}, \boldsymbol{w}') = h(\boldsymbol{w}) - \left(h(\boldsymbol{w}') + \nabla h(\boldsymbol{w}')^{\mathsf{T}}(\boldsymbol{w}' - \boldsymbol{w})\right). \tag{34}$$

Then, the algorithm is described as

$$\boldsymbol{y}^{(l+1)} = \underset{\boldsymbol{y} \in \mathbb{R}^N}{\arg\min} \left\{ J(\boldsymbol{w}^{(l)}) + \boldsymbol{g}^{(l)\mathsf{T}}\left(\boldsymbol{y} - \boldsymbol{w}^{(l)}\right) + \frac{1}{\alpha^{(l)}} D_h(\boldsymbol{y}, \boldsymbol{w}^{(l)}) \right\}, \tag{35}$$

$$\boldsymbol{w}^{(l+1)} = P_C^h(\boldsymbol{y}^{(l+1)}), \tag{36}$$

where $\alpha^{(l)} > 0$, $\boldsymbol{g}^{(l)}$ is the subgradient of $J(\boldsymbol{w}^{(l)})$, and $P_C^h(\boldsymbol{y})$ is the Bregman projection defined by

$$P_C^h(\boldsymbol{y}) = \underset{\boldsymbol{x} \in C}{\arg\min}\ D_h(\boldsymbol{x}, \boldsymbol{y}). \tag{37}$$

The problem of interest in (P3) is the case where

$$C = \{\boldsymbol{w} \in \mathbb{R}^N \mid \mathbf{0} \leq \boldsymbol{w} \leq \mathbf{1},\ \mathbf{1}^{\mathsf{T}}\boldsymbol{w} = K\}. \tag{38}$$

To calculate (35) and (36), we define the function $h(\boldsymbol{w})$ and obtain the Bregman divergence D_h as

$$h(\boldsymbol{w}) = \sum_{j=1}^{N} w_j \log w_j, \tag{39}$$

$$D_h(\boldsymbol{w}, \boldsymbol{y}) = \sum_{j=1}^{N} w_j \log \frac{w_j}{y_j} - (w_j - y_j), \tag{40}$$

respectively. These are the same functions proposed in [44] for the optimization over the unit simplex

$$C = \left\{\boldsymbol{w} \in \mathbb{R} \mid w_j \geq 0,\ \mathbf{1}^{\mathsf{T}}\boldsymbol{w} = 1\right\}. \tag{41}$$

However, this constraint is slightly different from (P3) in that the sum of the elements is 1, not $K \in \mathbb{N}$. Thus, the algorithm will be modified from the one proposed in [44].

First, solving (35) yields

$$y_j^{(l+1)} = w_j^{(l)} \exp\left(-\alpha^{(l)} g_j^{(l)}\right) \quad (j = 1, \ldots, N). \tag{42}$$

Here, the computation of $\boldsymbol{g}$, which is the subgradient of J_{MSE} or J_{ENT}, is straightforward, since both cost functions have gradients calculated as

$$\nabla J_{\text{MSE}}(\boldsymbol{w}) = -\lambda \operatorname{diag}\left(\boldsymbol{P}_{\boldsymbol{w}} \boldsymbol{K}_{\mathcal{C},\mathcal{E}} \boldsymbol{K}_{\mathcal{E},\mathcal{C}} \boldsymbol{P}_{\boldsymbol{w}}^{\mathsf{H}}\right), \tag{43}$$

$$\nabla J_{\text{ENT}}(\boldsymbol{w}) = -\lambda \operatorname{diag}\left(\boldsymbol{P}_{\boldsymbol{w}} \boldsymbol{K}_{\mathcal{C},\mathcal{E}} \boldsymbol{\Sigma}_{\mathcal{E}}^{-1} \boldsymbol{K}_{\mathcal{E},\mathcal{C}} \boldsymbol{P}_{\boldsymbol{w}}^{\mathsf{H}}\right). \tag{44}$$

Next, to find the Bregman projection in (36), we first adjust the elements by multiplying a constant factor to them, so that the sum would be equal to K, which is denoted as

$$\bar{w}_j^{(l+1)} = \frac{K}{\|\boldsymbol{y}^{(l+1)}\|_1} y_j^{(l+1)} \quad (j = 1, \ldots, N). \tag{45}$$

After this procedure, we will round the elements larger than 1 to 1, and adjust the rest by multiplying a constant factor to make the sum of the elements equal to K. By repeating this procedure until every element belongs to [0,1], we obtain the Bregman projection. We note that the obtained solution can be confirmed to be the Bregman projection by writing down the Karush-Kuhn-Tucker conditions of (36). The projection can be completed with the complexity of $O(N^2)$. The algorithm is summarized in Algorithm 2.

We discuss the difference between the proposed algorithm and the conventional convex relaxation methods based on SDP, as shown in [39]. Since SDP is formulated by introducing additional variables and is typically solved using second-order optimization algorithms in solvers [47], the spatial complexity is $O(M^4)$. In contrast, the spatial complexity for the proposed algorithm is $O(M^2 + N^2)$. When the sizes of N and M are approximately the same, the proposed algorithm requires much smaller memories. The difference becomes even larger when considering a broadband sound field estimation, as discussed in Section V.

After obtaining the solution of (P3), defined as $\hat{\boldsymbol{w}}$, an approximate solution for (P1) or (P2) is obtained by either selecting the K largest entries of $\hat{\boldsymbol{w}}$ or by conducting a randomized rounding algorithm [19]. A simple randomized rounding algorithm is summarized in Algorithm 3.

V. Application to Sound Field Estimation

A. Gaussian Process for Sound Field Estimation

The choice of the kernel function is essential for the sensor placement problem. Here, we provide a way of formulating

Algorithm 3: Randomized Rounding Algorithm.

Require: Solution $\hat{\boldsymbol{w}}$ of (P3)
for $l = 1, \ldots, L$ **do**
Generate a random vector $\boldsymbol{\eta}^{(l)} \sim \mathcal{N}(\mathbf{0}, \mathrm{diag}\,(\hat{\boldsymbol{w}}))$.
Select the K largest entries (in absolute terms) of $\boldsymbol{\eta}^{(l)}$ and define the set of those indices as $\mathcal{L}^{(l)} = \{j_1, \ldots, j_K\}$.
Generate $\boldsymbol{w}^{(l)} \in \{0,1\}^N$, defined as

$$\begin{cases} w_j^{(l)} & = 1, \quad j \in \mathcal{L}^{(l)} \\ w_j^{(l)} & = 0, \quad \text{otherwise.} \end{cases} \tag{46}$$

end for
Choose the vector in $\{\boldsymbol{w}^{(l)}\}_{l=1}^{I}$ that yields the smallest objective value of (P1) or (P2).

appropriate kernel functions, which was also considered in [7], and show some examples.

The sound field u satisfies the Helmholtz equation

$$(\Delta + \bar{k}^2)u = 0, \tag{47}$$

where $\bar{k}$ denotes the wave number and Δ denotes the Laplacian. Thus, when modeling a sound field as a GP, the kernel function should be chosen so that the estimated field, i.e., the MAP estimate, always satisfies the above equation. This can be achieved when the kernel function with one of the inputs fixed satisfies the Helmholtz equation, since the MAP estimate is given as a weighted sum of kernel functions. For example, the two positive definite kernel functions proposed in [48] defined as

$$\kappa(\boldsymbol{r}, \boldsymbol{r}') = \begin{cases} J_0(\bar{k}\|\boldsymbol{r} - \boldsymbol{r}'\|) \\ j_0(\bar{k}\|\boldsymbol{r} - \boldsymbol{r}'\|) \end{cases} \tag{48}$$

for the two-dimensional and three-dimensional cases, respectively, can be used. Here, J_0 is the zeroth-order Bessel function of the first kind and j_0 is the zeroth-order spherical Bessel function of the first kind.

The GP using these kernel functions can be interpreted as a diffused field with amplitude 1 [48], where plane waves arrive from all directions with independent complex Gaussian amplitudes. For example, the two-dimensional case of (48) can be written as

$$J_0(\bar{k}\|\boldsymbol{r} - \boldsymbol{r}'\|) = \frac{1}{2\pi} \int_{\mathbb{S}} e^{-\mathrm{j}\bar{k}\boldsymbol{\xi}^{\mathsf{T}}\boldsymbol{r}} \left(e^{-\mathrm{j}\bar{k}\boldsymbol{\xi}^{\mathsf{T}}\boldsymbol{r}'}\right)^* \mathrm{d}\boldsymbol{\xi}, \tag{49}$$

where j denotes the imaginary unit, $(\cdot)^*$ denotes the complex conjugate, $e^{-\mathrm{j}\bar{k}\boldsymbol{\xi}^{\mathsf{T}}\boldsymbol{r}}$ is the plane wave function with the arrival direction being $\boldsymbol{\xi}$, and $\mathbb{S}$ is the unit circle. The integrand of (49) is the covariance between $\boldsymbol{r}$ and $\boldsymbol{r}'$ of a plane wave arriving from the direction $\boldsymbol{\xi}$ with the variance of the amplitude 1. By integrating this over $\mathbb{S}$, the covariance of plane waves arriving from all directions is calculated.

In the same way, by constructing a kernel function that takes some prior knowledge of the sound field into consideration, such as the direction of the sound source, we can model the GP incorporating that prior knowledge. For example, in [49], the kernel function is constructed with a directional weighting function incorporated into (49). Using this kernel function, we can obtain a GP model in which the sound source is expected to be in a specific direction.

B. Broadband Case

The kernel function for expressing sound fields depends on the frequency, as in (48). However, the frequency range of interest in sound field estimation is often broadband, and the proposed algorithms cannot be directly applied in these cases. In this subsection, we consider applying the proposed methods to the broadband case.

We first discretize the frequency range of interest and obtain a set of frequency bins $\mathcal{F} = \{f_1, \ldots, f_F\}$ ($|\mathcal{F}| = F$). Now, the problem we consider is to estimate $\boldsymbol{u}_{\mathcal{E},\mathcal{F}} = \left[\boldsymbol{u}_{\mathcal{E},f_1}^{\mathsf{T}}, \ldots, \boldsymbol{u}_{\mathcal{E},f_F}^{\mathsf{T}}\right]^{\mathsf{T}}$, where $\boldsymbol{u}_{\mathcal{E},f} \in \mathbb{C}^M$ is defined as the sound field at estimation locations $\mathcal{E}$ of frequency $f \in \mathcal{F}$. Since the priority of estimating the sound field might differ among frequencies, we define the cost functions as a weighted sum of those for the single frequency case, i.e.,

$$J_{\mathrm{MSE},\mathcal{F}}(\mathcal{S}) = \sum_{f \in \mathcal{F}} a_f J_{\mathrm{MSE},f}(\mathcal{S}), \tag{50}$$

$$J_{\mathrm{ENT},\mathcal{F}}(\mathcal{S}) = \sum_{f \in \mathcal{F}} a_f J_{\mathrm{ENT},f}(\mathcal{S}), \tag{51}$$

where $J_{\mathrm{MSE},f}$ and $J_{\mathrm{ENT},f}$ stand for the cost functions in (P1) and (P2) using the kernel function corresponding to the frequency f and $a_f > 0$ denotes the weight for each frequency bin $f \in \mathcal{F}$. Here, $J_{\mathrm{MSE},\mathcal{F}}(\mathcal{S})$ can be seen as the weighted mean squared error of estimating $\boldsymbol{u}_{\mathcal{E},\mathcal{F}}$ with the weights being a_f for sound fields of frequency $f \in \mathcal{F}$. Also, by assuming that the sound fields of different frequencies are statistically independent and by using the additivity of entropy, $J_{\mathrm{ENT},\mathcal{F}}(\mathcal{S})$ can be similarly interpreted as the weighted conditional entropy.

Both the greedy and convex relaxation algorithms proposed in this paper can be applied to optimize these cost functions, since any calculation related to the cost functions can be replaced with the weighted sum of those in (P1) and (P2).

C. Computational Complexity of Broadband Case

The algorithms proposed in Section IV for approximately optimizing (50) or (51) involves F times as many calculations as the single frequency case. Thus, the computational complexity for the greedy algorithm becomes $O(FN^2)$, and the spatial complexity for MDA becomes $O(F(N^2 + M^2))$. Note that the spatial complexity for solving the SDP problem becomes $O(F^2\, M^4)$. Since the complexity with respect to F of SDP is also larger than that of the MDA, the SDP-based method becomes even more infeasible than in the single frequency case. For example, consider the situation in the numerical experiments. In the broadband case, 7.49×10^5 variables are required to be stored for the calculation of the gradient, whereas 5.22×10^{10} variables are required for the Hessian in SDP.

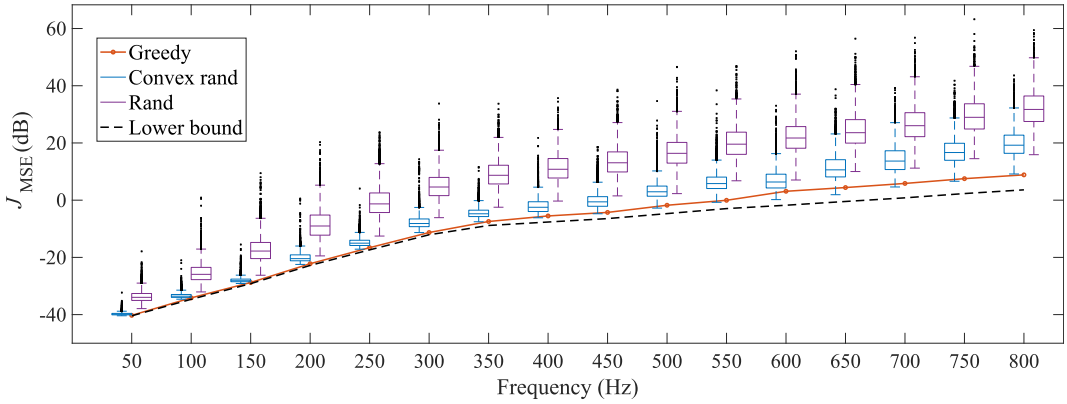

Fig. 9. Distributions of the cost function value of MSE $\mathcal{C} \neq \mathcal{E}$ for randomized methods with 24 sensors.

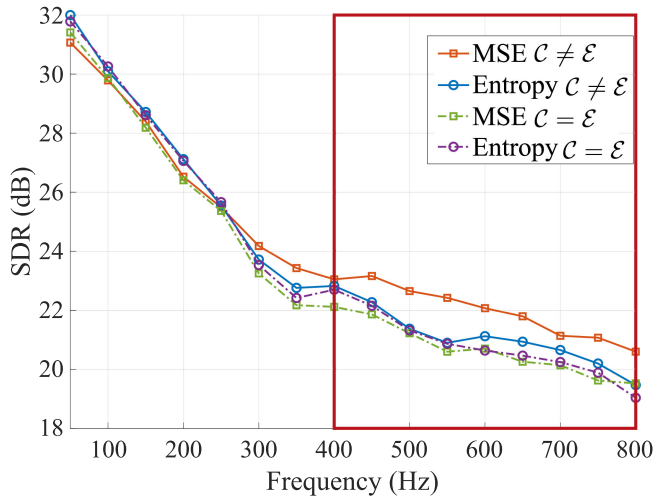

Fig. 10. SDR with respect to frequency for broadband case for 24 sensors. The red rectangle illustrates the frequency of interest for sensor placement.

VI. Numerical Experiments

We evaluated the proposed algorithms by numerical experiments. We especially focus on sensor placement for two-dimensional sound field estimation. The candidate and estimation regions are defined as in Fig. 3. Here, $\mathcal{C}$ and $\mathcal{E}$ were defined by discretizing each area at intervals of 0.05 m. The cardinalities of each set were $|\mathcal{C}| = 138$ and $|\mathcal{E}| = 169$. The Bessel function kernel in (48) was used for the GP model. The variance of the observation noise was $\lambda = 10^{-2}$. The sound speed was set to 340.0 m/s.

We evaluated the methods in two ways. One is a theoretical evaluation where we compared the achieved cost function values. The other is a more practical evaluation where we compared the reconstruction accuracy of plane wave fields. For the latter evaluation, the reconstruction accuracy with respect to the frequency is measured using the signal-to-distortion ratio (SDR) defined as

$$\mathrm{SDR}(f) = 10\log_{10}\frac{\int_{\Omega}|u_{\mathrm{true}}(\boldsymbol{r},\bar{k}_f)|^2\mathrm{d}\boldsymbol{r}}{\int_{\Omega}|u_{\mathrm{true}}(\boldsymbol{r},\bar{k}_f)-\hat{u}(\boldsymbol{r},\bar{k}_f)|^2\mathrm{d}\boldsymbol{r}}, \quad (52)$$

where u_{true} and $\hat{u}$ are the true and estimated sound fields, respectively, $\bar{k}_f$ is the wave number of the frequency f, and Ω denotes the estimation region. The integral was done by discretizing Ω at an interval of 0.01 m. The true sound field to be estimated was a sinusoidal plane wave field with an amplitude of 1 arriving from angles of 0° to 359° every 1°.

The narrowband case is considered in Sects. VI-A and VI-B. In Section VI-A, the two proposed optimization problems approximately solved by the greedy method are compared. To investigate the effects of independently setting the estimation and candidate positions, we also compare them with the optimization problems where $\mathcal{C}$ and $\mathcal{E}$ are fixed to the same positions. In Section VI-B, the two proposed approximation methods are compared. Also, we evaluate the optimality of the solutions by using the lower bound of the optimization problem obtained from the convex-relaxation-based method. Finally, the broadband case is investigated in Section VI-C.

A. Comparison of Greedy Algorithms in Narrowband Case

We evaluate the proposed methods in the narrowband case, where sensor positions are determined at each frequency. To investigate the effects of independently setting the estimation and candidate positions, the proposed mean-squared-error-based problem (MSE) and entropy-based problem (Entropy) were both used along with the problems where the estimation positions are fixed to the candidate positions. The proposed problems and the problems for comparison are denoted as $\mathcal{C} \neq \mathcal{E}$ and $\mathcal{C} = \mathcal{E}$, respectively. Note that in the $\mathcal{C} = \mathcal{E}$ case, both the candidate and estimation positions were fixed to the candidate positions of the proposed methods. Thus, for example, for MSE $\mathcal{C} = \mathcal{E}$, the MSE in $\mathcal{C}$ is calculated for the cost function (the sensor positions are also selected from $\mathcal{C}$). This is different from the proposed MSE $\mathcal{C} \neq \mathcal{E}$, where the MSE in $\mathcal{E}$ is calculated for the cost function. In particular, the $\mathcal{C} = \mathcal{E}$ case of Entropy is equivalent to the optimization problem that maximizes $H(\boldsymbol{y}_S)$. For the implementation of the greedy algorithm for Entropy $\mathcal{C} \neq \mathcal{E}$, a small diagonal matrix $10^{-7} \times \boldsymbol{I}$ was added to $\boldsymbol{K}_{\mathcal{E},\mathcal{E}}$ when calculating $\boldsymbol{\Sigma}_{\mathcal{E}}(\emptyset)^{-1} = \boldsymbol{K}_{\mathcal{E},\mathcal{E}}^{-1}$ to stabilize the computation.

Fig. 4(a) and (b) illustrate the cost function values for J_{MSE} and J_{ENT} described in (12) and (14), respectively, with respect to the

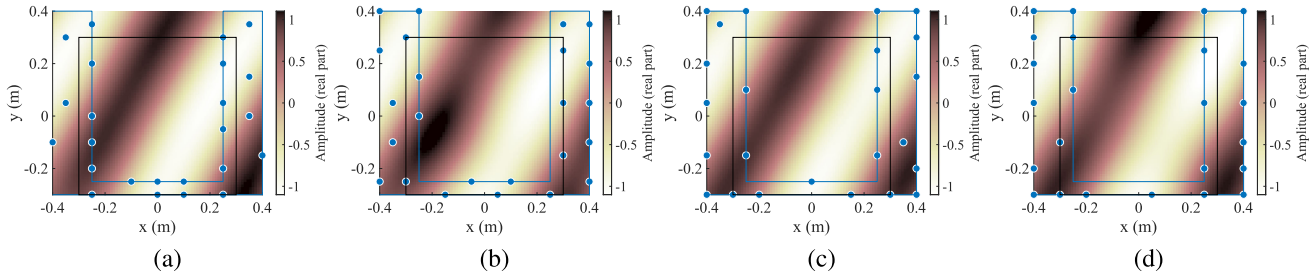

Fig. 11. Reconstructed pressure distributions at 600 Hz with 24 sensors chosen by the greedy algorithm for the broadband case. (a) MSE $\mathcal{C} \neq \mathcal{E}$; (b) MSE $\mathcal{C} = \mathcal{E}$; (c) Entropy $\mathcal{C} \neq \mathcal{E}$; (d) Entropy $\mathcal{C} = \mathcal{E}$.

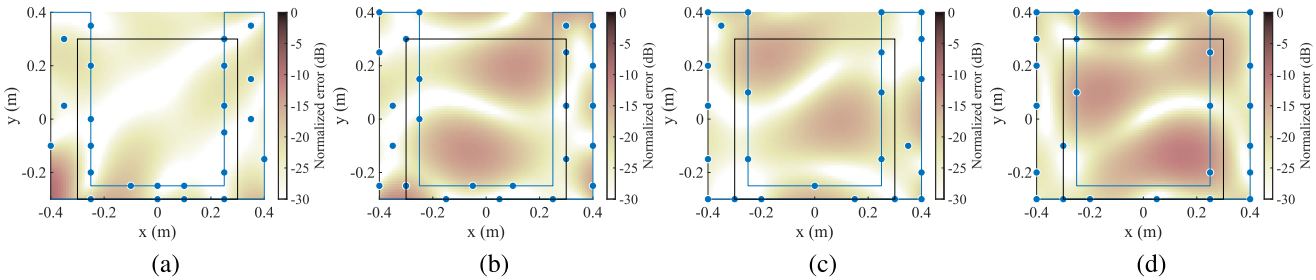

Fig. 12. Normalized error distributions at 600 Hz with 24 sensors chosen by the greedy algorithm for the broadband case. (a) MSE $\mathcal{C} \neq \mathcal{E}$; (b) MSE $\mathcal{C} = \mathcal{E}$; (c) Entropy $\mathcal{C} \neq \mathcal{E}$; (d) Entropy $\mathcal{C} = \mathcal{E}$. The SDRs were (a) 24.7, (b) 18.5, (c) 20.1, and (d) 16.7 dB.

number of sensors. MSE $\mathcal{C} \neq \mathcal{E}$ obtained its own objective values lower than Entropy $\mathcal{C} \neq \mathcal{E}$, and vice versa. In Fig. 4(a), MSE $\mathcal{C} \neq \mathcal{E}$ obtained a lower MSE cost value than MSE $\mathcal{C} = \mathcal{E}$, which implies the usefulness of the proposed method. In contrast, Fig. 4(b) shows that Entropy $\mathcal{C} \neq \mathcal{E}$ and Entropy $\mathcal{C} = \mathcal{E}$ had little difference in J_{ENT}. This can be explained by the fact that the cost function of Entropy $\mathcal{C} \neq \mathcal{E}$ can be written as

$$H(\boldsymbol{u}_{\mathcal{E}} \mid \boldsymbol{y}_{\mathcal{S}}) = H(\boldsymbol{u}_{\mathcal{E}}, \boldsymbol{y}_{\mathcal{S}}) - H(\boldsymbol{y}_{\mathcal{S}}), \tag{53}$$

where the second term is the cost function of Entropy $\mathcal{C} = \mathcal{E}$. It is likely that the second term of (53) is much larger than the first term, and thus Entropy $\mathcal{C} \neq \mathcal{E}$ and Entropy $\mathcal{C} = \mathcal{E}$ almost take the same values.

In Fig. 5, the SDRs for 24 sensors are plotted against frequency. SDRs of all methods decrease as the frequency gets higher, which means that estimating sound fields gets harder as the frequency increases. The figure shows that $\mathcal{C} \neq \mathcal{E}$ achieves a higher SDR than $\mathcal{C} = \mathcal{E}$ in most cases, both for MSE and Entropy. MSE $\mathcal{C} \neq \mathcal{E}$ obtained a higher SDR than Entropy $\mathcal{C} \neq \mathcal{E}$. This can be explained by comparing J_{MSE} with the definition of SDR in (52). The denominator inside the integral of (52) denotes the mean squared error at the estimation region, which is approximated by J_{MSE}. Thus, when J_{MSE} is efficiently reduced, a high SDR can be expected.

Figs. 6 and 7 illustrate the estimated sound field and the estimation error along with the sensor placement, respectively. MSE $\mathcal{C} \neq \mathcal{E}$ mostly selected positions close to the estimation area. As a result, the lowest estimation error was achieved. Entropy $\mathcal{C} \neq \mathcal{E}$ also tended to select positions inside the estimation area, but it also selected many positions near the corner of the candidate area. This is because the second term in (53) requires the placement of sensors to be away as possible from each other, where this property is also described in [11].

B. Comparison of Greedy Method and Convex Relaxation Method

We compare the different algorithms proposed for the sensor placement problems and also evaluate the optimality of the solutions. We only provide the results of the narrowband case for MSE owing to lack of space, although MDA can also be applied to Entropy or the broadband case. For the convex relaxation method, two rounding algorithms were conducted: a simple algorithm that chooses the largest K entries (Convex max-K) and a randomized rounding algorithm described in Algorithm 3 (Convex rand). We also investigated random sensor placement where the placements of K sensors were selected uniformly randomly from $\mathcal{C}$ (Rand). By comparing this with Convex rand, we evaluate the effect of using the solution of convex relaxed problems for random rounding. The random procedures for Convex rand and Rand were conducted for $I = 5000$ times and the solution that obtained the lowest cost function value were selected.

Fig. 8 illustrates the cost function value for each algorithm. Greedy and Convex rand achieved the lowest values among them, whereas Convex max-K obtained a higher value than the others when the number of sensors was small. Since the values of Greedy and Convex rand were close to the lower bound of the optimal solution (Lower bound), which was obtained by the convex-relaxation-based method, we can confirm the near-optimality of the two algorithms.

Fig. 9 shows the boxplot of the obtained cost function values for the two randomized methods. Mostly, Convex rand achieved lower values than Rand, which shows the effectiveness of solving the relaxed convex problem and using the solution for the randomized rounding algorithm compared with choosing the sensor placements uniformly randomly.

Overall, Greedy and Convex rand were the two best algorithms for solving (P1). Although Greedy is much faster than Convex rand, Convex rand is still useful, since we can know the lower bound of the optimal value and can evaluate the optimality of the approximate solution using that lower bound.

C. Comparison of Optimization Problems in Broadband Case

We evaluate the proposed methods in the broadband case, where the solutions were obtained by the greedy algorithm. The frequency of interest is set from 400 to 800 Hz at intervals of 50 Hz. The weights in (50) and (51) are equally set to $a_f = 1$ for $f \in \mathcal{F}$. Fig. 10 illustrates the SDR with respect to frequency. Here, the placements of 24 sensors were determined and were used for estimating sound fields of frequencies from 50 to 800 Hz. For both MSE and Entropy, $\mathcal{C} \neq \mathcal{E}$ obtained higher SDRs than $\mathcal{C} = \mathcal{E}$ at most frequencies of interest, although the difference was small for Entropy. The highest SDR was achieved by MSE $\mathcal{C} \neq \mathcal{E}$, as in the narrowband case. Figs. 11 and 12 illustrate the sensor placement with an example of the estimated sound field and the estimation error. The tendencies of the sensor placements were the same as those in the narrowband case.

VII. Conclusion

We studied the sensor placement problem for field estimation where the field follows GPs. Whereas many sensor placement methods for GPs have been proposed, to the best of our knowledge, the case where the candidate and estimation regions are not identical was not considered. In this paper, we proposed two sensor placement methods based on mean squared error and conditional entropy that can set the candidate locations and the estimation locations independently. Subsequently, we derived two algorithms to obtain an approximate solution to the problems: the greedy method and the convex relaxation method. In addition, we derived a bound for the suboptimality of the greedy method, and in the convex relaxation method, the relaxed problem was solved by a first-order algorithm, which enables this method to be applied to large-scale problems compared with the second-order-algorithm-based methods. By using the proposed methods, we can determine sensor placements in broader situations. Experimental results for sound field estimation where the candidate placement region differs from the estimation region showed that the proposed methods were more suitable for this situation than current methods. We also confirmed the near optimality of the methods by using the bound obtained by the convex relaxation method.

Appendix A
Derivation of Update Rules for Greedy Algorithms

First, we define $\boldsymbol{A}_{\boldsymbol{w}} = \mathrm{diag}\,(\boldsymbol{w})\boldsymbol{P}_{\boldsymbol{w}}$. The update of $\boldsymbol{A}_{\boldsymbol{w}}$ by adding a new index j can be calculated as

$$\begin{aligned}
&\boldsymbol{A}_{\boldsymbol{w}+\boldsymbol{e}_j} \\
&= \left(\mathrm{diag}\,(\boldsymbol{w}) + \boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\right)\left(\lambda\boldsymbol{I} + \boldsymbol{K}_{\mathcal{C},\mathcal{C}}\,\mathrm{diag}\,(\boldsymbol{w}) + \boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\right)^{-1} \\
&= \left(\mathrm{diag}\,(\boldsymbol{w}) + \boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\right)\left(\boldsymbol{P}_{\boldsymbol{w}} - \frac{\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}}{1 + \boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{e}_j}\right). \quad (54)
\end{aligned}$$

Using this, the update rule of $\boldsymbol{A}_{\boldsymbol{w}}$ can be written as

$$\begin{aligned}
&\boldsymbol{A}_{\boldsymbol{w}+\boldsymbol{e}_j} - \boldsymbol{A}_{\boldsymbol{w}} \\
&= \boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}} - \frac{\left(\mathrm{diag}\,(\boldsymbol{w}) + \boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\right)\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}}{1 + \boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{e}_j}. \quad (55)
\end{aligned}$$

Here, the first term of (55) can be modified as

$$\begin{aligned}
\boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}} &= \frac{\left(1 + \boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{e}_j\right)\boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}}{1 + \boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{e}_j} \\
&= \frac{\left(\boldsymbol{I} + \boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\right)\boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}}{1 + \boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{e}_j}. \quad (56)
\end{aligned}$$

By substituting (56) to (55), the update rule of $\boldsymbol{A}_{\boldsymbol{w}}$ can be rewritten as

$$\begin{aligned}
\boldsymbol{A}_{\boldsymbol{w}+\boldsymbol{e}_j} - \boldsymbol{A}_{\boldsymbol{w}} &= \frac{\left(\boldsymbol{I} - \mathrm{diag}\,(\boldsymbol{w})\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\right)\boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}}{1 + \boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{e}_j} \\
&= b_j\boldsymbol{P}_{\boldsymbol{w}}^{\mathsf{H}}\boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}. \quad (57)
\end{aligned}$$

Here, the last equality can be confirmed by applying the matrix inversion lemma to $\boldsymbol{P}_{\boldsymbol{w}}^{\mathsf{H}}$ as

$$\begin{aligned}
\boldsymbol{P}_{\boldsymbol{w}}^{\mathsf{H}} &= \left(\lambda\boldsymbol{I} + \mathrm{diag}\,(\boldsymbol{w})\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\right)^{-1} \\
&= \lambda^{-1}\left(\boldsymbol{I} - \mathrm{diag}\,(\boldsymbol{w})\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\right). \quad (58)
\end{aligned}$$

Using (57), the update of the posterior covariance matrix by adding a new sensor $\boldsymbol{q}_j$ can be written as

$$\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S}) - \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S} \cup \{\boldsymbol{q}_j\}) = b_j\boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{P}_{\boldsymbol{w}}^{\mathsf{H}}\boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}. \quad (59)$$

Thus, the decrease in the cost functions can be derived as

$$\begin{aligned}
&J_{\mathrm{MSE}}(\mathcal{S}) - J_{\mathrm{MSE}}(\mathcal{S} \cup \{\boldsymbol{q}_j\}) \\
&\quad = \mathrm{tr}\left(b_j\boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{P}_{\boldsymbol{w}}^{\mathsf{H}}\boldsymbol{e}_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\right) \\
&\quad = b_j\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{P}_{\boldsymbol{w}}^{\mathsf{H}}\boldsymbol{e}_j, \\
&J_{\mathrm{ENT}}(\mathcal{S}) - J_{\mathrm{ENT}}(\mathcal{S} \cup \{\boldsymbol{q}_j\}) \\
&\quad = \log\det\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})\right) - \log\det\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S}) - b_j\boldsymbol{a}_j\boldsymbol{a}_j^{\mathsf{H}}\right) \\
&\quad = -\log\left(1 - b_j\boldsymbol{a}_j^{\mathsf{H}}\boldsymbol{\Sigma}_{\mathcal{E}}^{-1}(\mathcal{S})\boldsymbol{a}_j\right). \quad (60)
\end{aligned}$$

Here, since $\boldsymbol{P}_{\boldsymbol{w}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}$ is semidefinite (see Appendix B), $b_j \geq 0$. From this, we can confirm that both cost functions decrease by adding a new sensor, as

$$J_{\mathrm{MSE}}(\mathcal{S}) - J_{\mathrm{MSE}}(\mathcal{S} \cup \{\boldsymbol{q}_j\})$$

$$= b_j \|\boldsymbol{K}_{\mathcal{E},\mathcal{C}} \boldsymbol{P}_{\boldsymbol{w}}^{\mathsf{H}} \boldsymbol{e}_j\|^2 \geq 0, \tag{61}$$

$$J_{\text{ENT}}(\mathcal{S}) - J_{\text{ENT}}(\mathcal{S} \cup \{\boldsymbol{q}_j\}) \geq -\log(1) = 0. \tag{62}$$

Appendix B
Proof of Semidefiniteness of $\boldsymbol{P}_{\boldsymbol{w}} \boldsymbol{K}_{\mathcal{C},\mathcal{C}}$

By using the matrix inversion lemma, $\boldsymbol{P}_{\boldsymbol{w}} \boldsymbol{K}_{\mathcal{C},\mathcal{C}}$ can be rewritten as

$$\begin{aligned}\boldsymbol{P}_{\boldsymbol{w}} \boldsymbol{K}_{\mathcal{C},\mathcal{C}} &= \left(\lambda \boldsymbol{I} + \boldsymbol{K}_{\mathcal{C},\mathcal{C}} \operatorname{diag}(\boldsymbol{w}) \operatorname{diag}(\boldsymbol{w})\right)^{-1} \boldsymbol{K}_{\mathcal{C},\mathcal{C}} \\ &= \lambda^{-1} \left(\boldsymbol{K}_{\mathcal{C},\mathcal{C}} - \boldsymbol{K}_{\mathcal{C},\mathcal{C}} \operatorname{diag}(\boldsymbol{w}) \right. \\ &\quad \cdot \left(\lambda \boldsymbol{I} + \operatorname{diag}(\boldsymbol{w}) \boldsymbol{K}_{\mathcal{C},\mathcal{C}} \operatorname{diag}(\boldsymbol{w})\right)^{-1} \\ &\quad \left. \cdot \operatorname{diag}(\boldsymbol{w}) \boldsymbol{K}_{\mathcal{C},\mathcal{C}}\right). \end{aligned} \tag{63}$$

Here, let $\boldsymbol{X}$ be

$$\begin{aligned}\boldsymbol{X} &= \begin{bmatrix} \boldsymbol{K}_{\mathcal{C},\mathcal{C}} & \boldsymbol{K}_{\mathcal{C},\mathcal{C}} \operatorname{diag}(\boldsymbol{w}) \\ \operatorname{diag}(\boldsymbol{w}) \boldsymbol{K}_{\mathcal{C},\mathcal{C}} & \lambda \boldsymbol{I} + \operatorname{diag}(\boldsymbol{w}) \boldsymbol{K}_{\mathcal{C},\mathcal{C}} \operatorname{diag}(\boldsymbol{w}) \end{bmatrix} \\ &= \begin{bmatrix} \sqrt{\boldsymbol{K}_{\mathcal{C},\mathcal{C}}} \\ \operatorname{diag}(\boldsymbol{w}) \sqrt{\boldsymbol{K}_{\mathcal{C},\mathcal{C}}} \end{bmatrix} \begin{bmatrix} \sqrt{\boldsymbol{K}_{\mathcal{C},\mathcal{C}}} & \sqrt{\boldsymbol{K}_{\mathcal{C},\mathcal{C}}} \operatorname{diag}(\boldsymbol{w}) \end{bmatrix} \\ &\quad + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \lambda \boldsymbol{I} \end{bmatrix} \\ &\succeq \boldsymbol{0}. \end{aligned} \tag{64}$$

Since (63) is the schur complement of $\lambda^{-1}\boldsymbol{X} \succeq \boldsymbol{0}$, $\boldsymbol{P}_{\boldsymbol{w}} \boldsymbol{K}_{\mathcal{C},\mathcal{C}}$ is positive semidefinite.

Appendix C
Proof of Theorem 2

Since $\boldsymbol{K}_{\mathcal{C},\mathcal{C}}$ is invertible, $\boldsymbol{P}_{\boldsymbol{w}}$ can be reformulated as

$$\boldsymbol{P}_{\boldsymbol{w}} = \left(\lambda \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1} + \operatorname{diag}(\boldsymbol{w})\right)^{-1} \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}. \tag{65}$$

Using the above equation and (19), we can describe the supermodularity ratio as

$$\alpha = \min_{\substack{\mathcal{A} \subseteq \mathcal{B} \subset \mathcal{C}, \\ \boldsymbol{q}_j \in \mathcal{C} \setminus \mathcal{B}}} \frac{\lambda + \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{B}}^{-1} \boldsymbol{e}_j}{\lambda + \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{A}}^{-1} \boldsymbol{e}_j} \cdot \frac{\boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{A}}^{-1} \bar{\boldsymbol{K}} \boldsymbol{M}_{\mathcal{A}}^{-1} \boldsymbol{e}_j}{\boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{B}}^{-1} \bar{\boldsymbol{K}} \boldsymbol{M}_{\mathcal{B}}^{-1} \boldsymbol{e}_j}, \tag{66}$$

where $\boldsymbol{M}_{\boldsymbol{w}} = \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1} + \lambda^{-1} \operatorname{diag}(\boldsymbol{w})$ and $\bar{\boldsymbol{K}} = \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1} \boldsymbol{K}_{\mathcal{C},\mathcal{E}} \boldsymbol{K}_{\mathcal{E},\mathcal{C}} \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}$. By considering the fact that

$$\boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{S}}^{-1} \bar{\boldsymbol{K}} \boldsymbol{M}_{\mathcal{S}}^{-1} \boldsymbol{e}_j = \operatorname{tr}\left(\bar{\boldsymbol{K}} \boldsymbol{M}_{\mathcal{S}}^{-1} \boldsymbol{e}_j \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{S}}^{-1}\right) \tag{67}$$

and that $\bar{\boldsymbol{K}}$ and $\boldsymbol{M}_{\mathcal{S}}^{-1} \boldsymbol{e}_j \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{S}}^{-1}$ are positive semidefinite, we can apply the theorem in [50], i.e.,

$$\lambda_{\min}(\boldsymbol{A}) \operatorname{tr}(\boldsymbol{B}) \leq \operatorname{tr}(\boldsymbol{A}\boldsymbol{B}) \leq \lambda_{\max}(\boldsymbol{A}) \operatorname{tr}(\boldsymbol{B}). \tag{68}$$

Then, we obtain

$$\begin{aligned}&\lambda_{\min}(\bar{\boldsymbol{K}}) \lambda_{\min}(\boldsymbol{M}_{\mathcal{S}}^{-1}) \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{S}}^{-1} \boldsymbol{e}_j \\ &\quad \leq \lambda_{\min}(\bar{\boldsymbol{K}}) \operatorname{tr}\left(\boldsymbol{M}_{\mathcal{S}}^{-1} \boldsymbol{e}_j \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{S}}^{-1}\right) \\ &\quad \leq \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{S}}^{-1} \bar{\boldsymbol{K}} \boldsymbol{M}_{\mathcal{S}}^{-1} \boldsymbol{e}_j \\ &\quad \leq \lambda_{\max}(\bar{\boldsymbol{K}}) \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{S}}^{-2} \boldsymbol{e}_j \\ &\quad \leq \lambda_{\max}(\bar{\boldsymbol{K}}) \lambda_{\max}(\boldsymbol{M}_{\mathcal{S}}^{-1}) \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{S}}^{-1} \boldsymbol{e}_j. \end{aligned} \tag{69}$$

Using this inequality and on the basis of the fact that

$$\left(\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1} + \lambda^{-1} \boldsymbol{I}\right)^{-1} \preceq \boldsymbol{M}_{\mathcal{B}}^{-1} \preceq \boldsymbol{M}_{\mathcal{A}}^{-1} \preceq \boldsymbol{K}_{\mathcal{C},\mathcal{C}}, \tag{70}$$

we can bound (66) as

$$\begin{aligned}&\frac{\lambda + \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{B}} \boldsymbol{e}_j}{\lambda + \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{A}} \boldsymbol{e}_j} \cdot \frac{\lambda_{\min}(\bar{\boldsymbol{K}}) \lambda_{\min}(\boldsymbol{M}_{\mathcal{A}}^{-1}) \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{A}}^{-1} \boldsymbol{e}_j}{\lambda_{\max}(\bar{\boldsymbol{K}}) \lambda_{\max}(\boldsymbol{M}_{\mathcal{B}}^{-1}) \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{B}}^{-1} \boldsymbol{e}_j} \\ &\quad \geq \frac{\lambda / [\boldsymbol{M}_{\mathcal{B}}^{-1}]_{j,j} + 1}{\lambda / [\boldsymbol{M}_{\mathcal{A}}^{-1}]_{j,j} + 1} \cdot \frac{\lambda_{\min}(\bar{\boldsymbol{K}}) \lambda_{\min}(\boldsymbol{M}_{\mathcal{A}}^{-1})}{\lambda_{\min}(\bar{\boldsymbol{K}}) \lambda_{\min}(\boldsymbol{M}_{\mathcal{B}}^{-1})} \\ &\quad \geq \frac{\lambda_{\min}(\bar{\boldsymbol{K}}) \lambda_{\min}(\boldsymbol{M}_{\mathcal{A}}^{-1})}{\lambda_{\max}(\bar{\boldsymbol{K}}) \lambda_{\max}(\boldsymbol{M}_{\mathcal{B}}^{-1})} \\ &\quad \geq \frac{\lambda_{\min}(\bar{\boldsymbol{K}}) \lambda_{\min}\left(\left(\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1} + \lambda^{-1} \boldsymbol{I}\right)^{-1}\right)}{\lambda_{\max}(\bar{\boldsymbol{K}}) \lambda_{\max}\left(\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\right)}. \end{aligned} \tag{71}$$

Finally by substituting

$$\lambda_{\min}\left(\left(\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1} + \lambda^{-1} \boldsymbol{I}\right)^{-1}\right) = \frac{1}{1/\lambda_{\min}(\boldsymbol{K}_{\mathcal{C},\mathcal{C}}) + \lambda^{-1}}, \tag{72}$$

Theorem 2 is derived.

Appendix D
Proof of Theorem 3

Using (20), the supermodularity ratio of J_{ENT} can be described by

$$\alpha = \min_{\mathcal{A} \subseteq \mathcal{B} \subset \mathcal{C}} \frac{\log\left(1 - b_{\mathcal{A},j} \boldsymbol{a}_{\mathcal{A},j}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{A})^{-1} \boldsymbol{a}_{\mathcal{A},j}\right)}{\log\left(1 - b_{\mathcal{B},j} \boldsymbol{a}_{\mathcal{B},j}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1} \boldsymbol{a}_{\mathcal{B},j}\right)}. \tag{73}$$

By using the inequality of the logarithm

$$\frac{x}{1+x} \leq \log(1+x) \leq x \tag{74}$$

and the fact that $\log\left(1 - b_{\mathcal{B},j} \boldsymbol{a}_{\mathcal{B},j}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1} \boldsymbol{a}_{\mathcal{B},j}\right) \leq 0$ and $\log\left(1 - b_{\mathcal{A},j} \boldsymbol{a}_{\mathcal{A},j}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{A})^{-1} \boldsymbol{a}_{\mathcal{A},j}\right) \leq 0$, we can bound α by

$$\begin{aligned}\alpha &\geq \min_{\mathcal{A} \subseteq \mathcal{B} \subset \mathcal{C}} \left(-b_{\mathcal{A},j} \boldsymbol{a}_{\mathcal{A},j}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{A})^{-1} \boldsymbol{a}_{\mathcal{A},j}\right) \\ &\quad \cdot \frac{1 - b_{\mathcal{B},j} \boldsymbol{a}_{\mathcal{B},j}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1} \boldsymbol{a}_{\mathcal{B},j}}{-b_{\mathcal{B},j} \boldsymbol{a}_{\mathcal{B},j}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1} \boldsymbol{a}_{\mathcal{B},j}} \\ &= \min_{\mathcal{A} \subseteq \mathcal{B} \subset \mathcal{C}} \frac{b_{\mathcal{A},j} \boldsymbol{a}_{\mathcal{A},j}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{A})^{-1} \boldsymbol{a}_{\mathcal{A},j}}{b_{\mathcal{B},j} \boldsymbol{a}_{\mathcal{B},j}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1} \boldsymbol{a}_{\mathcal{B},j}} \\ &\quad \cdot \left(1 - b_{\mathcal{B},j} \boldsymbol{a}_{\mathcal{B},j}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1} \boldsymbol{a}_{\mathcal{B},j}\right). \end{aligned} \tag{75}$$

Here, by using (68), we can bound some components of the above equation as

$$\begin{aligned}&\boldsymbol{a}_{\mathcal{B},j}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1} \boldsymbol{a}_{\mathcal{B},j} \\ &= \lambda^{-2} \boldsymbol{e}_j^{\mathsf{T}} \boldsymbol{M}_{\mathcal{B}}^{-1} \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1} \boldsymbol{K}_{\mathcal{C},\mathcal{E}} \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1} \boldsymbol{K}_{\mathcal{E},\mathcal{C}} \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1} \boldsymbol{M}_{\mathcal{B}}^{-1} \boldsymbol{e}_j \end{aligned}$$

$$
\begin{aligned}
&\leq \lambda^{-2}\lambda_{\max}\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1}\right)\lambda_{\max}\left(\bar{\boldsymbol{K}}\right)\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{M}_{\mathcal{B}}^{-2}\boldsymbol{e}_j\\
&\leq \lambda^{-2}\lambda_{\max}\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1}\right)\lambda_{\max}\left(\bar{\boldsymbol{K}}\right)\lambda_{\max}\left(\boldsymbol{M}_{\mathcal{B}}^{-1}\right)[\boldsymbol{M}_{\mathcal{B}}^{-1}]_{j,j}\\
&\leq \lambda^{-2}\lambda_{\max}\left(\boldsymbol{K}_{\mathcal{E},\mathcal{E}}^{-1}\right)\lambda_{\max}\left(\bar{\boldsymbol{K}}\right)\lambda_{\max}\left(\boldsymbol{M}_{\mathcal{B}}^{-1}\right)[\boldsymbol{M}_{\mathcal{B}}^{-1}]_{j,j}\\
&\leq \lambda^{-2}\lambda_{\max}\left(\boldsymbol{K}_{\mathcal{E},\mathcal{E}}^{-1}\right)\lambda_{\max}\left(\bar{\boldsymbol{K}}\right)\lambda_{\max}\left(\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\right)[\boldsymbol{M}_{\mathcal{B}}^{-1}]_{j,j},
\end{aligned}
\tag{76}
$$

$$
\begin{aligned}
&\boldsymbol{a}_{\mathcal{A},j}^{\mathsf{H}}\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{A})^{-1}\boldsymbol{a}_{\mathcal{A},j}\\
&\geq \lambda^{-2}\lambda_{\min}\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{A})^{-1}\right)\lambda_{\min}\left(\bar{\boldsymbol{K}}\right)\lambda_{\min}\left(\boldsymbol{M}_{\mathcal{A}}^{-1}\right)[\boldsymbol{M}_{\mathcal{A}}^{-1}]_{j,j}\\
&\geq \lambda^{-2}\lambda_{\min}\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{C})^{-1}\right)\lambda_{\min}\left(\bar{\boldsymbol{K}}\right)\lambda_{\min}\left(\boldsymbol{M}_{\mathcal{A}}^{-1}\right)[\boldsymbol{M}_{\mathcal{A}}^{-1}]_{j,j}\\
&\geq \lambda^{-2}\frac{\lambda_{\min}\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{C})^{-1}\right)\lambda_{\min}\left(\bar{\boldsymbol{K}}\right)[\boldsymbol{M}_{\mathcal{A}}^{-1}]_{j,j}}{1/\lambda_{\min}(\boldsymbol{K}_{\mathcal{C},\mathcal{C}})+\lambda^{-1}}\\
&= \lambda^{-2}\frac{\lambda_{\min}\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{C})^{-1}\right)\lambda_{\min}\left(\bar{\boldsymbol{K}}\right)\lambda_{\min}(\boldsymbol{K}_{\mathcal{C},\mathcal{C}})[\boldsymbol{M}_{\mathcal{A}}^{-1}]_{j,j}}{1+\lambda^{-1}\lambda_{\min}(\boldsymbol{K}_{\mathcal{C},\mathcal{C}})},
\end{aligned}
\tag{77}
$$

where $\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{C}) = \boldsymbol{K}_{\mathcal{E},\mathcal{E}} - \boldsymbol{K}_{\mathcal{E},\mathcal{C}}(\boldsymbol{K}_{\mathcal{C},\mathcal{C}}+\lambda\boldsymbol{I})^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}$. Here, we used the fact that the covariance matrix $\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S})$ satisfies

$$
\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{C}) \succeq \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{S}) \succeq \boldsymbol{K}_{\mathcal{E},\mathcal{E}}. \tag{78}
$$

The second term of (75) can modified as

$$
\begin{aligned}
&1 - b_{\mathcal{B},j}\boldsymbol{a}_{\mathcal{B},j}^{\mathsf{H}}\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1}\boldsymbol{a}_{\mathcal{B},j}\\
&= 1 - \frac{\lambda^{-1}\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{M}_{\mathcal{B}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1}\boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{M}_{\mathcal{B}}^{-1}\boldsymbol{e}_j^{\mathsf{T}}}{1+\lambda^{-1}[\boldsymbol{M}_{\mathcal{B}}^{-1}]_{j,j}}\\
&= \frac{1}{\lambda+[\boldsymbol{M}_{\mathcal{B}}^{-1}]_{j,j}}\left(\lambda+\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{M}_{\mathcal{B}}^{-1}\boldsymbol{e}_j\right.\\
&\quad\left.-\boldsymbol{e}_j^{\mathsf{T}}\boldsymbol{M}_{\mathcal{B}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1}\boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{M}_{\mathcal{B}}^{-1}\boldsymbol{e}_j^{\mathsf{T}}\right)\\
&= \frac{1}{\lambda+[\boldsymbol{M}_{\mathcal{B}}^{-1}]_{j,j}}\left(\lambda+\boldsymbol{e}_j^{\mathsf{T}}\left(\boldsymbol{M}_{\mathcal{B}}^{-1}\right.\right.\\
&\quad\left.\left.-\boldsymbol{M}_{\mathcal{B}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1}\boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{M}_{\mathcal{B}}^{-1}\right)\boldsymbol{e}_j^{\mathsf{T}}\right).
\end{aligned}
\tag{79}
$$

Here, we reformulate $\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})$ as

$$
\begin{aligned}
\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B}) &= \boldsymbol{K}_{\mathcal{E},\mathcal{E}} - \boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\\
&\quad + \boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{M}_{\mathcal{B}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}.
\end{aligned}
\tag{80}
$$

This can be derived by applying

$$
\begin{aligned}
&\boldsymbol{\Phi}_{\mathcal{S}}\left(\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{\Phi}_{\mathcal{S}}+\lambda\boldsymbol{I}\right)^{-1}\boldsymbol{\Phi}_{\mathcal{S}}^{\mathsf{T}}\\
&\quad= \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1} - \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\left(\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}+\lambda^{-1}\operatorname{diag}(\boldsymbol{w})\right)^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}
\end{aligned}
\tag{81}
$$

to (9). Here, the following matrix which appears in (79) can be shown to be positive seimidefinite as

$$
\boldsymbol{M}_{\mathcal{B}}^{-1} - \boldsymbol{M}_{\mathcal{B}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1}\boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{M}_{\mathcal{B}}^{-1} \succeq \boldsymbol{0}. \tag{82}
$$

This is because the above matrix is the schur complement of a positive semidefinite matrix

$$
\begin{aligned}
&\begin{bmatrix}\boldsymbol{M}_{\mathcal{B}}^{-1} & \boldsymbol{M}_{\mathcal{B}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\\ \boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{M}_{\mathcal{B}}^{-1} & \boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})\end{bmatrix}\\
&= \begin{bmatrix}\boldsymbol{M}_{\mathcal{B}}^{-1} & \boldsymbol{M}_{\mathcal{B}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\\ \boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{M}_{\mathcal{B}}^{-1} & \boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{M}_{\mathcal{B}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\end{bmatrix}\\
&\quad+\begin{bmatrix}\boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{K}_{\mathcal{E},\mathcal{E}}-\boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\end{bmatrix}\\
&\succeq \boldsymbol{0}.
\end{aligned}
\tag{83}
$$

From (82) and (79), we obtain

$$
\begin{aligned}
1 - b_{\mathcal{B},j}\boldsymbol{a}_{\mathcal{B},j}^{\mathsf{H}}\boldsymbol{\Sigma}_{\mathcal{E}}(\mathcal{B})^{-1}\boldsymbol{a}_{\mathcal{B},j} &\geq \frac{\lambda}{\lambda+[\boldsymbol{M}_{\mathcal{B}}^{-1}]_{j,j}}\\
&\geq \frac{1}{1+\lambda^{-1}\max_j[\boldsymbol{K}_{\mathcal{C},\mathcal{C}}]_{j,j}}.
\end{aligned}
\tag{84}
$$

Finally, by substituting (76), (77) and (84) to (75) and by using the fact that $\boldsymbol{M}_{\mathcal{B}}^{-1} \preceq \boldsymbol{M}_{\mathcal{A}}^{-1}$, Theorem 3 is derived.

Appendix E Proof of Convexity of (P3)

The Hessian of J_{MSE} can be calculated as

$$
\Delta_{\boldsymbol{w}}J_{\text{MSE}}(\boldsymbol{w}) = 2\lambda^{-1}\boldsymbol{M}_{\boldsymbol{w}}^{-1}\odot\left(\boldsymbol{M}_{\boldsymbol{w}}^{-1}\bar{\boldsymbol{K}}\boldsymbol{M}_{\boldsymbol{w}}^{-1}\right). \tag{85}
$$

Here, since the Hessian is constructed as the Hadamard product of two positive semidefinite matrices, it is positive semidefinite. Thus, J_{MSE} is convex.

Next, the Hessian of J_{ENT} can be calculated as

$$
\begin{aligned}
\Delta_{\boldsymbol{w}}J_{\text{ENT}}(\boldsymbol{w}) &= \Delta_{\boldsymbol{w}}\log\det\left(\boldsymbol{\Sigma}_{\mathcal{E}}(\boldsymbol{w})\right)\\
&= \lambda^{-2}\left(\boldsymbol{A}_{\boldsymbol{w}}\boldsymbol{\Sigma}_{\mathcal{E}}^{-1}(\boldsymbol{w})\boldsymbol{A}_{\boldsymbol{w}}^{\mathsf{T}}\right)\\
&\quad\odot\left(2\boldsymbol{M}_{\boldsymbol{w}}^{-1}-\boldsymbol{A}_{\boldsymbol{w}}\boldsymbol{\Sigma}_{\mathcal{E}}^{-1}(\boldsymbol{w})\boldsymbol{A}_{\boldsymbol{w}}^{\mathsf{T}}\right),
\end{aligned}
\tag{86}
$$

where $\boldsymbol{A}_{\boldsymbol{w}} = \boldsymbol{M}_{\boldsymbol{w}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}$. Here, since $\boldsymbol{A}_{\boldsymbol{w}}\boldsymbol{\Sigma}_{\mathcal{E}}^{-1}(\boldsymbol{w})\boldsymbol{A}_{\boldsymbol{w}}^{\mathsf{T}}$ is positive semidefinite, if the right term of the Hadamard product in (86) is also positive semidefinite, the Hessian matrix would also be positive semidefinite.

Since Gram matrices are always positive semidefinite,

$$
\begin{aligned}
&\begin{bmatrix}\boldsymbol{K}_{\mathcal{E},\mathcal{E}} & \boldsymbol{K}_{\mathcal{E},\mathcal{C}}\\ \boldsymbol{K}_{\mathcal{C},\mathcal{E}} & \boldsymbol{K}_{\mathcal{C},\mathcal{C}}\end{bmatrix} \succeq \boldsymbol{0}\\
&\Leftrightarrow \boldsymbol{K}_{\mathcal{E},\mathcal{E}} - \boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}} \succeq \boldsymbol{0}.
\end{aligned}
\tag{87}
$$

Using the fact that $\boldsymbol{K}_{\mathcal{E},\mathcal{E}} - \boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}$ is positive semidefinite, we can define another positive semidefinite matrix as

$$
\begin{aligned}
&\boldsymbol{K}_{\boldsymbol{w}} + \boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\left(\boldsymbol{K}_{\mathcal{E},\mathcal{E}} - \boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{E}}\right)^{-1}\boldsymbol{K}_{\mathcal{E},\mathcal{C}}\boldsymbol{K}_{\mathcal{C},\mathcal{C}}^{-1}\\
&= \left(\boldsymbol{M}_{\boldsymbol{w}}^{-1} - \boldsymbol{A}_{\boldsymbol{w}}\boldsymbol{\Sigma}_{\boldsymbol{w}}^{-1}\boldsymbol{A}_{\boldsymbol{w}}^{\mathsf{T}}\right)^{-1}.
\end{aligned}
\tag{88}
$$

This shows that $\boldsymbol{M}_{\boldsymbol{w}}^{-1} - \boldsymbol{A}_{\boldsymbol{w}}\boldsymbol{\Sigma}_{\boldsymbol{w}}^{-1}\boldsymbol{A}_{\boldsymbol{w}}$ is also positive semidefinite, and thus that the right term of the Hadamard product in (86) is positive semidefinite. Thus, J_{ENT} is convex.

REFERENCES

[1] M. R. Holdaway, "Spatial modeling and interpolation of monthly temperature using kriging," *Clim. Res.*, vol. 6, no. 3, pp. 215–225, 1996.

[2] R. Fablet and F. Rousseau, "Joint interpolation of multisensor sea surface temperature fields using nonlocal and statistical priors," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 6, pp. 2665–2675, Jun. 2016.

[3] N. Ueno, S. Koyama, and H. Saruwatari, "Sound field recording using distributed microphones based on harmonic analysis of infinite order," *IEEE Signal Process. Lett.*, vol. 25, no. 1, pp. 135–139, Jan. 2018.

[4] A. Laborie, R. Bruno, and S. Montoya, "A new comprehensive approach of surround sound recording," in *Proc. 114th AES Int. Conf.*, 2003, pp. 1–20.

[5] P. Samarasinghe, T. Abhayapala, and M. Poletti, "Wavefield analysis over large areas using distributed higher order microphones," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 3, pp. 647–658, Mar. 2014.

[6] J. V. Zidek, W. Sun, and N. D. Le, "Designing and integrating composite networks for monitoring multivariate Gaussian pollution fields," *J. R. Statist. Soc., Ser. C*, vol. 49, no. 1, pp. 63–79, 2000.

[7] K. Ariga, T. Nishida, S. Koyama, N. Ueno, and H. Saruwatari, "Mutual-information-based sensor placement for spatial sound field recording," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2020, pp. 166–170.

[8] G. Chardon, W. Kreuzer, and M. Noisternig, "Design of spatial microphone arrays for sound field interpolation," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 5, pp. 780–790, Aug. 2015.

[9] S. Koyama, G. Chardon, and L. Daudet, "Joint source and sensor placement for sound field control based on empirical interpolation method," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 501–505.

[10] S. Koyama, G. Chardon, and L. Daudet, "Optimizing source and sensor placement for sound field control: An overview," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 696–714, Jan. 2020.

[11] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies," *J. Mach. Learn. Res.*, vol. 9, pp. 235–284, 2008.

[12] M. C. Shewry and H. P. Wynn, "Maximum entropy sampling," *J. Appl. Stat.*, vol. 14, no. 2, pp. 165–170, 1987.

[13] Y. Yang and R. S. Blum, "Sensor placement in gaussian random field via discrete simulation optimization," *IEEE Signal Process. Lett.*, vol. 15, pp. 729–732, Nov. 2008.

[14] D. M. Steinberg and W. G. Hunter, "Experimental design: Review and comment," *Technometrics*, vol. 26, no. 2, pp. 71–97, 1984.

[15] K. Chaloner and I. Verdinelli, "Bayesian experimental design: A review," *J. Statist. Sci.*, vol. 10, no. 3, pp. 273–304, 1995.

[16] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 451–462, Feb. 2009.

[17] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[18] N. Ueno, S. Koyama, and H. Saruwatari, "Kernel ridge regression with constraint of Helmholtz equation for sound field interpolation," in *Proc. Int. Workshop Acoust. Signal Enhancement*, 2018, pp. 436–440.

[19] S. Liu, S. P. Chepuri, M. Fardad, E. Maşazade, G. Leus, and P. K. Varshney, "Sensor selection for estimation with correlated measurement noise," *IEEE Trans. Signal Process.*, vol. 64, no. 13, pp. 3509–3522, Jul. 2016.

[20] M. Shamaiah, S. Banerjee, and H. Vikalo, "Greedy sensor selection: Leveraging submodularity," in *Proc. IEEE Conf. Decis. Control*, 2010, pp. 2572–2577.

[21] A. Konar and N. D. Sidiropoulos, "A simple and effective approach for transmit antenna selection in multiuser massive MIMO leveraging submodularity," *IEEE Trans. Signal Process.*, vol. 66, no. 18, pp. 4869–4883, Sep. 2018.

[22] C. Jiang, Y. C. Soh, and H. Li, "Sensor placement by maximal projection on minimum eigenspace for linear inverse problems," *IEEE Trans. Signal Process.*, vol. 64, no. 21, pp. 5595–5610, Nov. 2016.

[23] G. E. Hovland and B. J. McCarragher, "Dynamic sensor selection for robotic systems," in *Proc. IEEE Int. Conf. Robot. Automat.*, 1997, pp. 272–277.

[24] X. Shen and P. K. Varshney, "Sensor selection based on generalized information gain for target tracking in large sensor networks," *IEEE Trans. Signal Process.*, vol. 62, no. 2, pp. 363–375, Jan. 2014.

[25] S. P. Chepuri and G. Leus, "Sparsity-promoting sensor selection for non-linear measurement models," *IEEE Trans. Signal Process.*, vol. 63, no. 3, pp. 684–698, Feb. 2015.

[26] G. Nemhauser, L. Wolsey, and M. Fisher, "An analysis of approximations for maximizing submodular set functions," *Math. Program.*, vol. 14, no. 1, pp. 265–294, 1978.

[27] A. Das and D. Kempe, "Approximate submodularity and its applications: Subset selection, sparse approximation and dictionary selection," *J. Mach. Learn. Res.*, vol. 19, no. 1, pp. 74–107, 2018.

[28] L. Chamon and A. Ribeiro, "Approximate supermodularity bounds for experimental design," in *Proc. Conf. Neural Inform. Process. Syst.*, 2017, pp. 5403–5412.

[29] L. Chamon and A. Ribeiro, "Greedy sampling of graph signals," *IEEE Trans. Signal Process.*, vol. 66, no. 1, pp. 34–47, Jan. 2018.

[30] C. Jiang, Z. Chen, R. Su, and Y. C. Soh, "Group greedy method for sensor placement," *IEEE Trans. Signal Process.*, vol. 67, no. 9, pp. 2249–2262, May 2019.

[31] C. Rusu, J. Thompson, and N. M. Robertson, "Sensor scheduling with time, energy, and communication constraints," *IEEE Trans. Signal Process.*, vol. 66, no. 2, pp. 528–539, Jan. 2018.

[32] S. P. Chepuri and G. Leus, "Continuous sensor placement," *IEEE Signal Process. Lett.*, vol. 22, no. 5, pp. 544–548, May 2015.

[33] M. Coutino, S. P. Chepuri, and G. Leus, "Subset selection for kernel-based signal reconstruction," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 4014–4018.

[34] H. Wang, G. Pottie, K. Yao, and D. Estrin, "Entropy-based sensor selection heuristic for target localization," in *Proc. Int. Symp. Inf. Process. Sensor Netw.*, 2004, pp. 36–45.

[35] M. Al-Obaidy, A. Ayesh, and A. F. Sheta, "Optimizing the communication distance of an ad hoc wireless sensor networks by genetic algorithms," *Artif. Intell. Rev.*, vol. 29, no. 3/4, pp. 183–194, 2008.

[36] W. J. Welch, "Branch-and-bound search for experimental designs based on *D* optimality and other criteria," *Technometrics*, vol. 24, no. 1, pp. 41–48, 1982.

[37] S. Lau, R. Eichardt, L. D. Rienzo, and J. Haueisen, "Tabu search optimization of magnetic sensor systems for magnetocardiography," *IEEE Trans. Magn.*, vol. 44, no. 6, pp. 1442–1445, Jun. 2008.

[38] M. Naeem, S. Xue, and D. C. Lee, "Cross-entropy optimization for sensor selection problems," in *Proc. Int. Symp. Commun. Inf. Technol.*, 2009, pp. 396–401.

[39] T. Nishida, N. Ueno, S. Koyama, and H. Saruwatari, "Sensor placement in arbitrarily restricted region for field estimation based on Gaussian process," in *Proc. Eur. Signal Process. Conf.*, 2021, pp. 2289–2293.

[40] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan, "Learning the kernel matrix with semidefinite programming," *J. Mach. Learn. Res.*, vol. 5, pp. 27–72, 2004.

[41] A. G. Wilson and R. P. Adams, "Gaussian process kernels for pattern discovery and extrapolation," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1067–1075.

[42] D. J. C. MacKay, "Information-based objective functions for active data selection," *Neural Comput.*, vol. 4, no. 4, pp. 590–604, 1992.

[43] J. Ranieri, A. Chebira, and M. Vetterli, "Near-optimal sensor placement for linear inverse problems," *IEEE Trans. Signal Process.*, vol. 62, no. 5, pp. 1135–1146, Mar. 2014.

[44] A. Beck and M. Teboulle, "Mirror descent and nonlinear projected subgradient methods for convex optimization," *Oper. Res. Lett.*, vol. 31, no. 3, pp. 167–175, 2003.

[45] S. Liu, J. Chen, P. Chen, and A. Hero, "Zeroth-order online alternating direction method of multipliers: Convergence analysis and applications," in *Proc. Int. Conf. AI, Stat.*, 2018, pp. 288–297.

[46] L. Condat, "A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms," *J. Optim. Theory. Appl.*, vol. 158, no. 2, pp. 460–479, 2013.

[47] R. H. Tütüncü, K. C. Toh, and M. J. Todd, "SDPT3 matlab software package for semidefinite-quadratic-linear programming, version 3.0," 2001. [Online]. Available: https://www.math.cmu.edu/∼reha/sdpt3.html

[48] R. K. Cook, R. V. Waterhouse, R. D. Berendt, S. Edelman, and M. C. Thompson Jr., "Measurement of correlation coefficients in reverberant sound fields," *J. Acoust. Soc. Amer.*, vol. 27, no. 6, pp. 1072–1077, 1955.

[49] H. Ito, S. Koyama, N. Ueno, and H. Saruwatari, "Spatial active noise control based on kernel interpolation with directional weighting," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2020, pp. 8404–8408.

[50] B. Wang and F. Zhang, "Some inequalities for the eigenvalues of the product of positive semidefinite hermitian matrices," *Linear Algebra Appl.*, vol. 160, pp. 113–118, 1992.

Tomoya Nishida (Member, IEEE) received the B.E. and M.S. degrees in mathematical engineering and information physics from The University of Tokyo, Tokyo, Japan, in 2019 and 2021, respectively. He is currently a Researcher with Hitachi, Ltd, Tokyo, Japan. His research focuses on acoustic signal processing.

Natsuki Ueno (Member, IEEE) received the B.E. degree in engineering from Kyoto University, Kyoto, Japan, in 2016, and the M.S. and Ph.D. degrees in information science and technology from The University of Tokyo, Tokyo, Japan, in 2018 and 2021, respectively. He is currently a Project Assistant Professor with Tokyo Metropolitan University, Tokyo, Japan. His research interests include spatial audio and acoustic signal processing.

Shoichi Koyama (Member, IEEE) received the B.E., M.S., and Ph.D. degrees from The University of Tokyo, Tokyo, Japan, in 2007, 2009, and 2014, respectively. In 2009, he joined Nippon Telegraph and Telephone Corporation, as a Researcher in acoustic signal processing. In 2014, he moved to The University of Tokyo, where he has been a Lecturer since 2018. From 2016 to 2018, he was also a Visiting Researcher with Institut Langevin, Paris Diderot University (Paris7), Paris, France. His research interests include audio signal processing, acoustic inverse problems, and spatial audio. He was the recipient of the Itakura Prize Innovative Young Researcher Award by the Acoustical Society of Japan, in 2015, and Research Award by Funai Foundation for Information Technology, in 2018.

Hiroshi Saruwatari (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Nagoya University, Nagoya, Japan, in 1991, 1993, and 2000, respectively. In 1993, he joined SECOM IS Laboratory, Tokyo, Japan, and in 2000 the Nara Institute of Science and Technology, Ikoma, Japan. Since 2014, he has been a Professor with The University of Tokyo, Tokyo, Japan. His research interests include statistical audio signal processing, blind source separation, and speech enhancement. He has put his research into the world's first commercially available independent-component-analysis-based BSS microphone in 2007. He was the recipient of several paper awards from IEICE in 2001 and 2006, from TAF in 2004, 2009, 2012, and 2018, from IEEE-IROS2005 in 2006, and from APSIPA in 2013 and 2018. He was also the recipient of the DOCOMO Mobile Science Award in 2011, Ichimura Award in 2013, Commendation for Science and Technology by the Minister of Education in 2015, Achievement Award from IEICE in 2017, and Hoko-Award in 2018. He has been professionally involved in various volunteer works for IEEE, EURASIP, IEICE, and ASJ. Since 2018, he has been an APSIPA Distinguished Lecturer.