Eng. Sci., 2025, **38**, 1734

Engineered Science
DOI: https://dx.doi.org/10.30919/es1734

Efficient Greedy Algorithm for Multi-Objective Sensor Placement Optimization

Wangchun Zhang,[1, 2] Ying He,[1] Kun You,[1] Hao Xie[1] and Yujun Zhang[1,*]

Abstract

The sensor placement problem is a complex combinatorial optimization challenge, where greedy selection provides performance guarantees for submodular objective functions, yielding suboptimal yet efficient solutions. Multi-objective optimization addresses the trade-offs among individual objectives, often requiring solutions on the Pareto Front. This paper introduces two multi-objective optimization functions based on optimal experimental design, incorporating normalization and weight update strategies. Leveraging the definition of submodularity, we revalidate that the non-negative linear combination of submodular functions preserves submodularity, applicable to one of the proposed functions. The effectiveness of the methods is validated across four datasets, demonstrating superior performance in sensor placement tasks. Notably, for bi-objective optimization involving A-optimality and D-optimality, adaptive-weight greedy solutions not only provide performance guarantees but also dominate most fixed-weight greedy solutions and lie on the Pareto Front.

Keywords: Multi-objective optimization; Submodularity; Greedy algorithm; Data driven; Sensor placement.

Received: 09 April 2025; Revised: 03 June 2025; Accepted: 14 July 2025.
Article type: Research article.

1. Introduction

Optimal Sensor Placement (OSP) has become a key research area for optimizing data collection efficiency, enhancing system performance, and reducing costs. It finds important applications in fields such as flow fields,[1-6] structural health monitoring,[7-9] and actuator control,[10,11] among others. Recent advances in structural health monitoring have introduced novel approaches such as sub-clustering strategies to address sensor redundancy issues and load-dependent placement methods for model updating under uncertainty.[12,13] The primary goal of OSP is to extract valuable global insights from limited data generated by sensors.

In the early days of sensor placement problems, simple fixed sensing radii were often used,[14] or sensors were placed at the edges of measurement areas, such as in CT scanning.[15] The latter approaches are considered to follow the principle of maximizing information entropy.[8,16,17]

The OSP problem can be described as selecting p sensors from a pool of n candidate sensors to minimize some cost function. It is a combinatorial optimization problem and is known to be NP-hard, involving a total of $\binom{n}{p}$ possible combinations.[18] While global optimal solutions can be found through brute force search and branch-and-bound method,[19] the computational cost is often prohibitive. Evolutionary algorithms have shown remarkable success in addressing such complex optimization challenges. Recent developments include real-valued encoding strategies for large-scale feature selection,[20] and improved grey wolf optimization with guided mutation strategies,[21] demonstrating superior performance in escaping local optima while maintaining solution diversity.

Researchers have turned to seek suboptimal solutions for OSP problem. Convex optimization employs convex relaxation technique to find suboptimal solutions,[22] but the performance gap between suboptimal and optimal solutions is not clear. Additionally, convex optimization techniques are computationally expensive, particularly for high-dimensional

[1] *Key Laboratory of Environmental Optics & Technology, Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei, 230031, China*

[2] *University of Science and Technology of China, Hefei, 230026, China*

**Email:* yjzhang@aiofm.ac.cn (Y. Zhang)

problems. Greedy methods are a popular approach for solving combinatorial optimization problems, and they have proven to be effective in numerous applications, including OSP. Greedy methods provide suboptimal solutions to the OSP problem from the perspective of information theory.[16,17] However, information-theoretic approaches often require statistical analysis, which can be computationally burdensome for large-scale problems.

Modeling the OSP problem as a linear measurement system and greedily selecting sensors to maximize a proxy function associated with the model is a recent and popular approach.[23-26] This proxy function is typically based on the Fisher information matrix,[7,23] which is closely related to optimal experimental design.[22,23,27,28] This model-based approach is now commonly data-driven, where the original high-dimensional signal is low-rank approximated through Proper Orthogonal Decomposition (POD),[27] and then optimization is performed based on proxy functions related to parameter estimation or the reconstruction error of the original signal. Recent innovations in sensor number determination strategies have further enhanced these approaches by integrating clustering avoidance mechanisms with interval uncertainty modeling.[29]

Researchers like Uciński,[30] and Yamada *et al*.,[31] have extended the OSP problem to scenarios with correlated noise. Saito *et al*.[32] and Nakai *et al*.[33] have extended it to undersampling from oversampling. Inoue *et al*.[34] (2023) and Guo *et al*.[6] have redefined the OSP problem from a machine learning perspective and provided corresponding cost functions. Nair and Gozahas extended the linear inverse problem to the case of nonlinear mappings.[2] Farazmand and Saibaba,[1] and Saito *et al*.[35] process the input data from tensor and vector perspectives, respectively. Li *et al*.[36] directly use the reconstruction error of the signal as the cost function for sensor placement. Group greedy[23] and convex optimization.[22] have studied improvements to greedy algorithm solutions from the input and output perspectives, respectively. In parallel, bio-inspired optimization techniques have demonstrated significant potential, with improved whale optimization algorithms showing enhanced coverage optimization in sensor networks and deep learning approaches enabling predictive modeling of complex systems.[37,38] An exception is QR Pivoting (QRP),[10,27] which requires only one matrix decomposition through the QR factorization with column pivoting of the low-rank representation's basis. Clark *et al*.[39] introduced the cost of sensor placement into the row-norm ordering of QRP, leading to OSP algorithms with cost constraints. Zhang *et al*.[3] used information entropy as weights and then implemented weighted column pivoting in QR factorization.

Benefiting from extensive research in the submodularity field,[11,40-43] greedy selection can provide performance guarantees when facing certain proxy functions. Nemhauser *et al*.[40] has demonstrated that the performance of greedy methods is within $1-1/e$ of optimal performance. Kohara *et al*.[42] has leveraged weak submodularity to provide approximate performance guarantees for non-submodular functions. Different forms of objective functions focus on various aspects in the optimization task, such as various reconstruction quality metrics based on information theory, the number of sensors, and the placement cost of sensors. Therefore, in the field of OSP, to balance the performance of various metrics, multi-objective optimization tasks have also been studied.[3,5,7,9,33,44] Nakai *et al*.[33] proposed a Pareto-optimal front based on the non-dominated solution perspective. Yang transformed multi-objective tasks into single-objective optimizations using weight factors, [7] avoiding the problem of analyzing the Pareto front. For multi-objective scenarios, interval Pareto front-based approaches and two-step non-probabilistic optimization strategies have shown particular promise in handling competing objectives under uncertainty.[45,46] Currently, in multi-objective sensor placement research, the focus is on how to combine it with other metaheuristic algorithms to achieve the Pareto-optimal solution.

This paper focuses on the multi-objective optimization problem of three objective functions related to optimal experimental design. Existing research has proven that mean squared error (MSE) and volume of the confidence ellipsoid (VCE) are submodular functions and has provided fast computation algorithms for these objective functions.[23,41] The worst case error variance (WCEV) is not submodular. The set of submodular functions is closed under both linear combinations and multiplication by positive scalars, similar to the properties of convex functions.[47] Building on the submodularity perspective, this paper revisits the definition and reaffirms that the non-negative linear combination of submodular functions preserves submodularity. Subsequently, we formulate a multi-objective optimization problem by combining submodular objective functions and implement an adaptive weighting strategy. Unlike traditional multi-objective optimization approaches that rely on optimization algorithms to explore the Pareto front, our method leverages the submodularity of the objective functions and employs a greedy algorithm to iteratively select sensor locations.

The key contributions of this work are fourfold. First, we provide a theoretical guarantee that non-negative combinations of A- and D-optimality criteria preserve

submodularity, thereby enabling a provable $1-1/e$ approximation bound for the proposed MFADG algorithm. Second, we introduce an adaptive weighting strategy that dynamically balances multiple objectives, achieving almost Pareto dominance over fixed-weight approaches. Finally, we conduct extensive validation on both synthetic and real-world datasets, demonstrating consistent improvements over baselines such as QRP and Convex Relaxation.

The remaining content of this paper is organized as follows. In Section 2, we first introduce the mathematical formulation of the OSP problem and the concept of submodularity. Building on the definition of submodularity, we then provide a lemma and its proof, reaffirming that the non-negative linear combination of submodular functions preserves submodularity. Finally, we present the mathematical formulations of two multi-objective optimization algorithms proposed in this paper, including efficient implementation strategies for A-, D-, and E-optimality objectives. In Section 3, we validate the effectiveness of the proposed methods on two simulation datasets and two real-world datasets. Sections 4 and 5 will be dedicated to discussion and summarization.

If not otherwise specified, this document will use uppercase bold letters $\mathbf{X}$ to represent matrices, lowercase bold letters $\boldsymbol{\alpha}$ to represent vectors, uppercase calligraphic letters $\mathcal{S}$ to denote sets, and regular lowercase letters α_1 to represent variables. Text with subscripts and superscripts like $\mathcal{S}_{\text{text}}$ is used for annotations.

2. Materials and methods

2.1 The mathematical description of OSP

We consider a high-dimensional physical field $\mathbf{x} \in \mathbb{R}^n$ that can be approximated using a reduced-order model as shown in Eq. (1):

$$\mathbf{x} \approx \boldsymbol{\Phi}_r \boldsymbol{\alpha} \tag{1}$$

where $\boldsymbol{\Phi}_r \in \mathbb{R}^{n\times r}$ is a known basis matrix with orthonormal columns derived from the truncated singular value decomposition (SVD) of the data matrix $\mathbf{X} \in \mathbb{R}^{n\times m}$ as shown in Eq. (2):

$$\mathbf{X} = \boldsymbol{\Phi}\boldsymbol{\Sigma}\mathbf{V}^T \tag{2}$$

and $\boldsymbol{\Phi}_r$ consists of the first r columns of $\boldsymbol{\Phi}$. Each column of $\mathbf{X}$ is a snapshot of the physical field at a given time, and rows correspond to spatial locations.

To reconstruct using sparse measurements, we introduce a linear measurement model as shown in Eq. (3):

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{w} \approx \mathbf{C}\boldsymbol{\Phi}_r\boldsymbol{\alpha} + \mathbf{w} = \boldsymbol{\Phi}\boldsymbol{\alpha} + \mathbf{w} \tag{3}$$

where $\mathbf{y} \in \mathbb{R}^p$ represents sensor observations, $\mathbf{w} \sim \mathcal{N}(0, \sigma^2\mathbf{I})$ is Gaussian noise, and $\mathbf{C} \in \mathbb{R}^{p\times n}$ is the selection matrix consisting of p row vectors from the identity matrix that select sensing locations $\mathcal{S} = \{s_1, s_2, \ldots, s_p\}$, with $p \geq r$. The measurement matrix $\boldsymbol{\Phi} = \mathbf{C}\boldsymbol{\Phi}_r \in \mathbb{R}^{p\times r}$ contains the observation vectors corresponding to the selected sensor positions.

Given the linear model and noise assumption, the minimum variance unbiased estimator of $\boldsymbol{\alpha}$ is given by the least-squares solution as shown in Eq. (4):

$$\boldsymbol{\alpha}' = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\mathbf{y} \tag{4}$$

The estimator's covariance is $\mathrm{Cov}(\boldsymbol{\alpha}') = \sigma^2(\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}$. To assess reconstruction quality, we introduce three classical optimality criteria based on the Fisher information matrix $I(\boldsymbol{\alpha}') = 1/\sigma^2\boldsymbol{\Phi}^T\boldsymbol{\Phi}$:

A-optimality: Minimizes the mean squared error (MSE) of the estimator as shown in Eq. (5):

$$\arg\min \mathrm{tr}\left((\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\right) = \arg\min \sum_i \frac{1}{\lambda_i(\boldsymbol{\Phi}^T\boldsymbol{\Phi})} \tag{5}$$

where $\lambda_i(\boldsymbol{\Phi}^T\boldsymbol{\Phi})$ represents the eigenvalues of the matrix $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$.

D-optimality: Maximizes the log-determinant of the information matrix, related to the volume of the confidence ellipsoid as shown in Eq. (6):

$$\arg\max |\det(\boldsymbol{\Phi}^T\boldsymbol{\Phi})| = \arg\max \prod_i \lambda_i(\boldsymbol{\Phi}^T\boldsymbol{\Phi}) \tag{6}$$

E-optimality: Minimizes the worst-case estimation variance as shown in Eq. (7):

$$\arg\min \| (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1} \|_2 = \arg\max \lambda_{\min}(\boldsymbol{\Phi}^T\boldsymbol{\Phi}) \tag{7}$$

First, formalize the optimization problems corresponding to Eq. (5)-(7) as a maximization problem in subset selection as shown in Eq. (8):

$$\begin{aligned} \max \quad & f(\mathcal{S}) \\ \text{subject to} \quad & |\mathcal{S}| \leq p \end{aligned} \tag{8}$$

where $\mathcal{S}$ represents the selected set of sensors, $|\cdot|$ denotes the cardinality of the set, and p corresponds to the maximum

allowable number of sensors. The objective functions $f(\mathcal{S})$ corresponding to Eqs. (5)-(7) are defined as Eqs. (9)-(11):

$$f_{\mathrm{A}} = \beta - \mathrm{tr}\left((\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\right) \tag{9}$$

$$f_{\mathrm{D}} = \log\det|\boldsymbol{\Phi}^T\boldsymbol{\Phi}| \tag{10}$$

$$f_{\mathrm{E}} = \lambda_{\min}(\boldsymbol{\Phi}^T\boldsymbol{\Phi}) \tag{11}$$

where β is a constant. Before solving the subset selection problem using a greedy algorithm, the concept of submodularity is introduced.

Definition 1: A set function $f: 2^{\mathcal{V}} \to \mathbb{R}$ is said to be nondecreasing if $\forall \mathcal{P} \subseteq \mathcal{Q} \subseteq \mathcal{V}$ as shown in Eq. (12),[40]

$$f(\mathcal{P}) \le f(\mathcal{Q}) \tag{12}$$

Definition 2: A set function $f: 2^{\mathcal{V}} \to \mathbb{R}$ is said to be submodular if $\mathcal{V}$ is a finite set, $\forall \mathcal{P} \subseteq \mathcal{Q} \subseteq \mathcal{V}$ and $j \in \mathcal{V} \setminus \mathcal{Q}$ as shown in Eq. (13),

$$f(\mathcal{P}+\{j\}) - f(\mathcal{P}) \ge f(\mathcal{Q}+\{j\}) - f(\mathcal{Q}) \tag{13}$$

If $f(\mathcal{S})$ is a non-decreasing submodular function, a greedy algorithm can be used to obtain an approximate solution for Eq. (8) in Algorithm 1.

If $\mathcal{S}_{\text{optimal}}$ is the optimal solution to the combinatorial optimization problem in Eq. (9) and satisfies $f(\varnothing) = 0$, then they satisfy the bound in Eq. (14):[40]

$$\frac{f(\mathcal{S}_{\text{greedy}})}{f(\mathcal{S}_{\text{optimal}})} \ge 1 - \frac{1}{e} \tag{14}$$

The greedy algorithm has two advantages. Firstly, as seen from Algorithm 1, it reduces the search space from $\binom{n}{p}$ possibilities to $\sum_{i=1}^{p}(N-i+1)$, reducing the algorithm's complexity from $\mathrm{O}(n^p)$ to $\mathrm{O}(p^2)$. Secondly, if the objective function is submodular, the greedy solution provides performance guarantees. However, in the case of a single objective function, such as Eqs. (5)-(7), which often focuses on optimizing a specific aspect of the problem, and considering that the greedy algorithm's solution is suboptimal, a single-objective greedy solution may not meet practical needs. This is where multi-objective optimization becomes important.

This study focuses on A-optimality, D-optimality, and E-optimality. The literature also provides three fast algorithms for optimal experimental design criteria, replacing the objective function in step five of Algorithm 1 with equivalent forms. Rewriting the basis $\boldsymbol{\Phi}$ as a combination of r-dimensional column vectors $\boldsymbol{\Phi} = [\phi_1, \phi_2, \ldots, \phi_N]^T$, the equivalent fast implementations are as shown in Eqs. (15)-(17):[23, 41]

$$g_A(\mathcal{S}\cup\{j\}) = \frac{\phi_j^T\boldsymbol{\Psi}_{\mathcal{S}}^{-2}\phi_j}{1+\phi_j^T\boldsymbol{\Psi}_{\mathcal{S}}^{-1}\phi_j} \tag{15}$$

$$g_D(\mathcal{S}\cup\{j\}) = \phi_j^T\boldsymbol{\Psi}_{\mathcal{S}}^{-1}\phi_j \tag{16}$$

$$g_E(\mathcal{S}\cup\{j\}) = (\phi_j^T\mathbf{v}_{\lambda_{min}}(\boldsymbol{\Psi}_{\mathcal{S}}))^2 \tag{17}$$

Here, $\boldsymbol{\Psi}_{\mathcal{S}} = \boldsymbol{\Phi}^T\boldsymbol{\Phi} + \epsilon\mathbf{I}$, where ϵ is a sufficiently small positive value. This is done to prevent singular matrices when selecting the i-th($i < r$) sensor in Algorithm algorithm1. $\mathbf{v}_{\lambda_{min}}(\boldsymbol{\Psi}_{\mathcal{S}})$ represents the eigenvector corresponding to the smallest eigenvalue λ_{min} of $\boldsymbol{\Psi}_{\mathcal{S}}$.

Algorithm 1: General greedy algorithm for solving sensor placement problems.

Input: $\hat{\boldsymbol{\Phi}}_r \in \mathbb{R}^{n\times r}, p$

Output: $\mathcal{S}_{\text{greedy}}$

1: Initialization: $\mathcal{S} \leftarrow \varnothing, \mathcal{V} \leftarrow \{1, 2, \ldots, n\}$
2: for $i \leftarrow 1$ to p do
3: while sensor number satisfies constraint do
4: $\mathcal{V}_n \leftarrow \mathcal{V} \setminus \mathcal{S}$
5: $j^\star \leftarrow \arg\max_{j\in\mathcal{V}_n} f(\mathcal{S}\cup\{j\})$
6: $\mathcal{S} \leftarrow \mathcal{S}\cup\{j^\star\}$
7: end while
8: end for
9: $\mathcal{S}_{\text{greedy}} \leftarrow \mathcal{S}$

$$\begin{aligned} f_{\text{multi}}(\mathcal{P}\cup\{j\})-f_{\text{multi}}(\mathcal{P}) &= \alpha_1 f_1(\mathcal{P}\cup\{j\})+\alpha_2 f_2(\mathcal{P}\cup\{j\})+\ldots+\alpha_k f_k(\mathcal{P}\cup\{j\}) \\ &\quad -(\alpha_1 f_1(\mathcal{P})+\alpha_2 f_2(\mathcal{P})+\ldots+\alpha_k f_k(\mathcal{P})) \\ &= \alpha_1(f_1(\mathcal{P}\cup\{j\})-f_1(\mathcal{P}))+\alpha_2(f_2(\mathcal{P}\cup\{j\})-f_2(\mathcal{P}))+\ldots \\ &\quad +\alpha_k(f_k(\mathcal{P}\cup\{j\})-f_k(\mathcal{P})) \end{aligned} \tag{19}$$

$$\begin{aligned} f_{\text{multi}}(\mathcal{P}\cup\{j\})-f_{\text{multi}}(\mathcal{P}) &= \alpha_1(f_1(\mathcal{P}\cup\{j\})-f_1(\mathcal{P}))+\alpha_2(f_2(\mathcal{P}\cup\{j\})-f_2(\mathcal{P})) \\ &\quad +\ldots+\alpha_k(f_k(\mathcal{P}\cup\{j\})-f_k(\mathcal{P}))+\ldots+\alpha_k(f_k(\mathcal{P}\cup\{j\})-f_k(\mathcal{P})) \\ &\geq \alpha_1(f_1(\mathcal{Q}\cup\{j\})-f_1(\mathcal{Q}))+\alpha_2(f_2(\mathcal{Q}\cup\{j\})-f_2(\mathcal{Q}))+\ldots \\ &\quad +\alpha_k(f_k(\mathcal{Q}\cup\{j\})-f_k(\mathcal{Q})) \end{aligned} \tag{22}$$

2.2 Multi-target sensor placement and submodularity

Before introducing multi-objective optimization, we first introduce a lemma.[47]

Lemma: Let $f_1, f_2, \ldots, f_k$ be a set of submodular set functions, and let $\alpha_1, \alpha_2, \ldots, \alpha_k$ be positive numbers. The function combination $f_{\text{multi}} = \alpha_1 f_1 + \alpha_2 f_2 + \ldots + \alpha_k f_k$ remains submodular.

Proof: According to Definition 2, for all $\mathcal{P} \subseteq \mathcal{Q} \subseteq \mathcal{V}$ and $j \in \mathcal{V} \setminus \mathcal{Q}$:

$$f_i(\mathcal{P}\cup\{j\})-f_i(\mathcal{P}) \geq f_i(\mathcal{Q}\cup\{j\})-f_i(\mathcal{Q}), \quad i=1,2,\ldots,k \tag{18}$$

Following the same logic as in Definition 2, we have Eq. (19):

Since $\alpha_i > 0$ and by Inequality Eq. (18), we can see that:

$$\alpha_i(f_i(\mathcal{P}\cup\{j\})-f_i(\mathcal{P})) \geq \alpha_i(f_i(\mathcal{Q}\cup\{j\})-f_i(\mathcal{Q})), \quad i=1,\ldots,k \tag{20}$$

By Definition 1, the submodular functions $f_1, f_2, \ldots, f_k$ are non-decreasing, *i.e.*, $f_i(\mathcal{P}\cup\{j\})-f_i(\mathcal{P}) \geq 0$. Combining this with $\alpha_i > 0$, we have, for Eq. (19):

$$f_{\text{multi}}(\mathcal{P}\cup\{j\})-f_{\text{multi}}(\mathcal{P}) \geq 0 \tag{21}$$

Inequality Eq. (21) shows that f_{multi} is non-decreasing. Using the multiplication inequality and multiplying both sides of Inequality Eq. (20) for all i, we can derive:

By Eq. (19), the right side of Inequality Eq. (22) simplifies to $\alpha_1(f_1(\mathcal{Q}\cup\{j\})-f_1(\mathcal{Q}))+\alpha_2(f_2(\mathcal{Q}\cup\{j\})-f_2(\mathcal{Q}))+\ldots+\alpha_k(f_k(\mathcal{Q}\cup\{j\})-f_k(\mathcal{Q})) = f_{\text{multi}}(\mathcal{Q}\cup\{j\})-f_{\text{multi}}(\mathcal{Q})$, so Inequality Eq. (22) is, in fact:

$$f_{\text{multi}}(\mathcal{P}\cup\{j\})-f_{\text{multi}}(\mathcal{P}) \geq f_{\text{multi}}(\mathcal{Q}\cup\{j\})-f_{\text{multi}}(\mathcal{Q}) \tag{23}$$

Returning to Definition 2, Inequality Eq. (23) implies that f_{multi} is submodular. Therefore, we can conclude that a non-negative linear combination of submodular functions remains submodular. Furthermore, since it is also non-decreasing, the performance guarantee in Inequality Eq. (14) holds.

Combining Eqs. (15)-(17), we can define two types of objective functions for the multi-objective optimization tasks that we will use later. The first type is a combination of the three model-based single objective functions, without involving the concept of submodularity, as shown in Eq. (24):

$$F_{\text{multi}}^{\text{ADE}} = \alpha_{\text{A}} g_{\text{A}} + \alpha_{\text{D}} g_{\text{D}} + \alpha_{\text{E}} g_{\text{E}} \tag{24}$$

The second type is a multi-objective function with submodularity, as shown in Eq. (25):

$$F_{\text{multi}}^{\text{AD}} = \alpha_{\text{A}} g_{\text{A}} + \alpha_{\text{D}} g_{\text{D}} \tag{25}$$

2.3 Normalization and weight update

Considering potential differences in the magnitudes of objective functions for different single-objective optimization problems, normalization becomes necessary. Eq. (26) illustrates the normalization process:

$$\mathbf{g}_{\text{single}}^{\text{normalized}} = \frac{\mathbf{g}_{\text{single}} - g_{\text{single}}^{\min}}{g_{\text{single}}^{\max} - g_{\text{single}}^{\min}}, \text{ for single} \in \{\text{A-, D-, E-optimality}\} \tag{26}$$

Here, $\mathbf{g}_{\text{single}}$ represents the vector of objective function values corresponding to all possible sensor positions j during the i -th greedy iteration. $g_{\text{single}}^{\max}$ and $g_{\text{single}}^{\min}$ denote the maximum and minimum values of $\mathbf{g}_{\text{single}}$, respectively. We substitute $\mathbf{g}_{\text{single}}^{\text{normalized}}$ into Eqs. (24) and (25) for the single objective function values.

For weight coefficients, we have two options. We can use uniform weights, where $\alpha_{\text{A}} = \alpha_{\text{D}} = 1/2$ for Eq. (25) and $\alpha_{\text{A}} = \alpha_{\text{D}} = \alpha_{\text{E}} = 1/3$ for Eq. (24). Alternatively, given that the objective functions for different single-objective optimizations may have varying scales and to ensure that different single-objective functions have distinct weights in multi-objective optimization, we can employ an adaptive weight update strategy, as described in Eq. (27).

Here, g_{single}^{i-1} represents the single objective function value when selecting the $(i-1)$ -th sensor, indicating the

contribution of a single objective function value g to the overall performance in the multi-objective sensor placement configuration $f_{\text{multi}}^{\text{best}}$. $\alpha_{\text{single}}^{i-1}$ is the weight coefficient used in the $(i-1)$ -th selection to prevent situations where $g_{\text{single}}^{i-1} / \sum g_{\text{single}}^{i-1}$ becomes close to zero, making the single objective function value ineffective in the i -th selection. The weight coefficients should satisfy $\sum \alpha_{\text{single}} = 1$. The recursive weight update process follows an exponentially weighted moving average. As the number of iterations increases, the influence of the initial value gradually vanishes, and the weights converge to a narrow interval around the final value. This behavior underscores the robustness of the method and its insensitivity to the choice of initial settings.

The specific implementation steps of the multi-target sensor placement algorithm are shown in Algorithm 2.

We also incorporated a algorithmic flowchart following Algorithm 2, which visually summarizes the proposed multi-objective greedy framework, as shown in Fig. 1.

2.4 Evaluation metrics

The choice of evaluation metrics is closely related to the quality of the solution $\boldsymbol{\alpha}$ and the Fisher information matrix. Specifically, we consider the following metrics: MSE, VCE, WCEV, and CN, as shown in Eqs. (28)-(31):

(1) MSE:

$$\text{MSE} = \text{tr}((\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}) \tag{28}$$

(2) VCE:

$$\text{VCE} = \log\det|(\boldsymbol{\Phi}^T\boldsymbol{\Phi})| \tag{29}$$

(3) WCEV:

$$\text{WCEV} = \frac{1}{\lambda_{\min}(\boldsymbol{\Phi}^T\boldsymbol{\Phi})} \tag{30}$$

(4) Condition Number (CN):

$$\text{CN} = \frac{\lambda_{\max}(\boldsymbol{\Phi}^T\boldsymbol{\Phi})}{\lambda_{\min}(\boldsymbol{\Phi}^T\boldsymbol{\Phi})} \tag{31}$$

To analyze the performance of the linear estimator in the reconstruction process of the original signal $\mathbf{X}$ under different scenarios, we define the reconstruction error as follows as shown in Eq. (32):

(5) Reconstruction Error:

$$\text{E}(\mathcal{S}_{\text{greedy}}) = \frac{\|\mathbf{X} - \boldsymbol{\Phi}_r\boldsymbol{\alpha}'\|_F}{\|\mathbf{X}\|_F} = \frac{\|\mathbf{X} - \boldsymbol{\Phi}_r\boldsymbol{\Phi}^{\dagger}\mathbf{C}\mathbf{X}\|_F}{\|\mathbf{X}\|_F} \tag{32}$$

Here, $\boldsymbol{\Phi}^{\dagger}$ represents the Moore-Penrose pseudoinverse of $\boldsymbol{\Phi}$, and $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. $\Phi^{\dagger}C$ is determined by the sensor positions from the training set $\mathbf{X}_{\text{train}}$. We can calculate the reconstruction error on the training set and test set as Eqs. (33) and (34) below:

Reconstruction Error on the Training Set:

$$\text{E}_{\text{train}}(\mathcal{S}_{\text{greedy}}) = \frac{\|\mathbf{X}_{\text{train}} - \boldsymbol{\Phi}_r\boldsymbol{\Phi}^{\dagger}\mathbf{C}\mathbf{X}_{\text{train}}\|_F}{\|\mathbf{X}_{\text{train}}\|_F} \tag{33}$$

Reconstruction Error on the Test Set:

$$\text{E}_{\text{test}}(\mathcal{S}_{\text{greedy}}) = \frac{\|\mathbf{X}_{\text{test}} - \boldsymbol{\Phi}_r\boldsymbol{\Phi}^{\dagger}\mathbf{C}\mathbf{X}_{\text{test}}\|_F}{\|\mathbf{X}_{\text{test}}\|_F} \tag{34}$$

2.5 Comparison Settings

(1) **RS** (Random Selection): Randomly selects p sensors from the candidate sensor set $|\mathcal{V}| = n$.

(2) **Convex**:[22] Utilizes convex relaxation to solve the D-optimality problem and assesses the performance gap between suboptimal and optimal solutions. The implementation of convex optimization can be found in the code[1].

(3) **QRP**:[27] The objective function of QR with column pivoting is based on D-optimality, aiming to maximize the volume of a matrix. QRP is a data-driven sensor placement strategy, and most popular programming software integrates implementations, such as the **qr** function in MATLAB.

(4) **REG**:[36] REG is based on reconstruction error and is a greedy sensor selection method. It directly assumes a linear estimator $\mathbf{X} = \mathbf{T}\mathbf{Y}$. The specific implementation of REG is available online[2].

(5) **MFADG**: A fast implementation algorithm based on A- and D-optimality, balancing the performance metrics of the two single-objective optimizations while providing optimality guarantees for greedy solutions, as defined in Eq. (25).

$$\alpha_{\text{single}}^{i} = \begin{cases} 1/z & \text{if } i = 1, (z = 2 \text{ or } 3) \\ \dfrac{g_{\text{single}}^{i-1} / \sum g_{\text{single}}^{i-1} + \alpha_{\text{single}}^{i-1}}{2} & \text{if } i = 2, 3, 4, \ldots, p \end{cases} \quad \text{for single} \in \{\text{A-, D-, E-optimality}\} \tag{27}$$

[1] https://web.stanford.edu/~boyd/papers/sensor_selection.html

[2] https://github.com/sjtulbj/REG_RELO

Algorithm 2: Greedy algorithm for multi-objective sensor placement optimization.

Input: $\hat{\mathbf{\Phi}}_r = [\phi_1, \phi_2, \ldots, \phi_n] \in \mathbb{R}^{n \times r}, p$

Output: $\mathcal{S}_{\text{greedy}}$

1: Initialization: $\mathcal{S} \leftarrow \varnothing, \mathcal{V} \leftarrow \{1, 2, \ldots, n\}, \mathbf{\Psi}_{\mathcal{S}} = \epsilon \mathbf{I}$, and $\alpha_{\text{single}} = 1/z, z = 2$ or 3,

2: for single $\in$ {A, D} or {A, D, E}

3: for $i \leftarrow 1$ to p do

4: **while** sensor number satisfies constraint do

5: $\mathcal{V}_n \leftarrow \mathcal{V} \setminus \mathcal{S}$

6: for $j \in \mathcal{V}_n$ do

7: $$g_{\text{single}}(\mathcal{S} \cup \{j\}) = \frac{g_{\text{single}}(\mathcal{S} \cup \{j\}) - g_{\text{single}}^{\min}(\mathcal{S} \cup \{j\})}{g_{\text{single}}^{\max}(\mathcal{S} \cup \{j\}) - g_{\text{single}}^{\min}(\mathcal{S} \cup \{j\})}$$

8: end for

9: $$j^{\star} \leftarrow \arg\max f_{\text{multi}}(\mathcal{S} \cup \{j\}) = \sum \alpha_{\text{single}} g_{\text{single}}(\mathcal{S} \cup \{j\})$$

10: $$\mathcal{S} \leftarrow \mathcal{S} \cup \{j^{\star}\}, \mathbf{\Psi}_{\mathcal{S}} = \mathbf{\Psi}_{\mathcal{S}} + \phi_{j^{\star}} \phi_{j^{\star}}^{T}$$

11: $$\alpha_{\text{single}} = \frac{g_{\text{single}}(\mathcal{S}) / \sum_{\text{single}} g_{\text{single}}(\mathcal{S}) + \alpha_{\text{single}}}{2}$$

12: end while

13: end for

14: $\mathcal{S}_{\text{greedy}} \leftarrow \mathcal{S}$

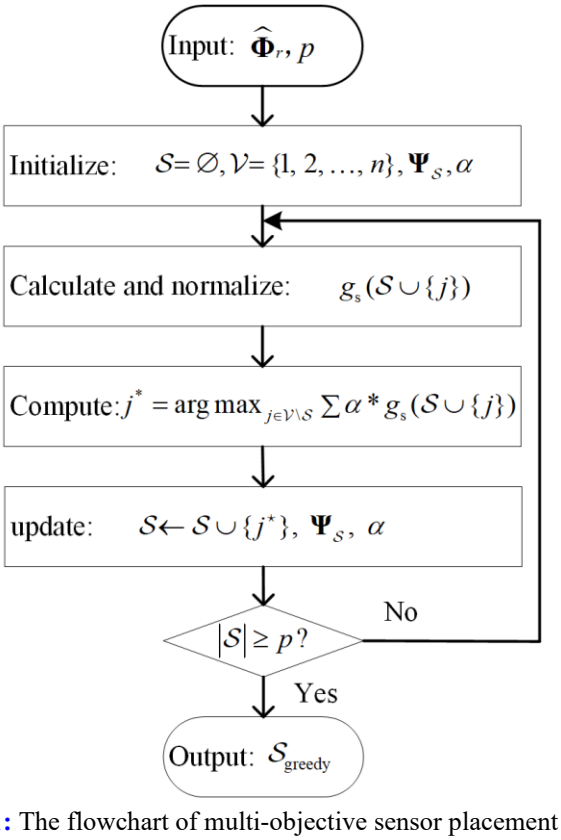

Fig. 1: The flowchart of multi-objective sensor placement optimization.

(6) **MFADEG**: An extension of MFADG based on A-, D-, and E-optimality, which also considers the impact of WCEV as described in Eq. (24). However, it sacrifices the performance guarantee of MFADG.

As shown in Algorithm 2 and Eqs. (15)-(17), the computational cost of the proposed MFADG algorithm mainly comes from matrix-vector multiplications and inverse matrix updates. By employing the Sherman–Morrison formula, the inverse matrix update in each iteration requires only $O(r^2)$ operations. Therefore, the overall computational complexity of MFADG is $O(pnr^2)$. For MFADEG, the additional cost arises from computing the minimal eigenvector required for the metric, which incurs a cost of $O(r^3)$. As a result, the total complexity of MFADEG is $O(p(r^3+nr^2))$. We summarize the computational costs of the proposed algorithms and several baseline methods introduced in this section in Table 1 below.

Here, i is the number of iterations, typically $i \approx \sqrt{r}$, and m is the number of samples used in the SVD operation, which is generally large in practice. Therefore, MFADG achieves lower computational cost compared to Convex and REG methods, and is only slightly more expensive than QRP, which only requires a single QR factorization.

Before proceeding with the method validation, two details are noteworthy. Firstly, the evaluation metrics introduced above are obtained through several Monte-Carlo simulations on the dataset. What we present is the statistical mean of these metrics, as shown in Eq. (35):

$$\overline{\text{Mean Index}} = \frac{\sum_{i=1}^{\text{MonteCarlo number}} \text{Index}_i}{\text{MonteCarlo number}} \tag{35}$$

Secondly, in Eq. (4), to use the least squares method to estimate the parameters α, it is necessary to satisfy $p \geq r$. We follow the same setting as QRP in all datasets, *i.e.*, we set $p = 2r$. To ensure reproducibility, a fixed random seed was used across all four datasets for 100 Monte Carlo simulations. These datasets include two simulated matrices: Gaussian random matrix and Bernoulli random matrix, as well as two real-world datasets: the Yale Face Database B (Yala B) and the Sea Surface Temperature (SST) dataset.

In each Monte-Carlo simulation run, the greedy determination of sensor locations was implemented on a randomly sampled 70% of the dataset, followed by testing on the remaining 30% of the dataset. The computations were performed on a personal computer with an Intel (R) Core (TM) i5-9300H CPU @ 2.40GHz and 16GB of RAM.

3. Results

3.1 Gaussian random matrix

The dataset $\mathbf{X}$ is a 100×100 matrix, where $\mathbf{X} \in \mathbb{R}^{100\times100}$. The elements in $\mathbf{X}$ are randomly generated from a standard normal distribution. We conducted 100 Monte-Carlo simulations using various methods listed in the section 2.5 to compare performance metrics related to the model: MSE, VCE, WCEV, CN and Error, as shown in Fig. 2.

The experiment examined the variation in these metrics as the number of sensors increased from 20 to 50 in increments of 2. With Fig. 2a, the Convex Relaxation method performed slightly better than MFADG and MFADEG in the VCE index. However, in the other four performance metrics, MSE, WCEV, CN, and Error, the proposed multi-objective optimization methods outperformed the previous methods. The objective function of the Convex Relaxation method is related to D-optimality, which may explain its slightly better performance in VCE. It's important to note that the VCE index is the only evaluation metric in the section 2.4 that needs to be maximized, so a higher VCE value indicates a better sensor configuration. The QRP method's performance on the random matrix was as expected, as its advantage lies in performing a single matrix decomposition. As the number of selected sensors increased, only the trend in Figs. 2c and e perfectly matched the expectations for greedy methods, where the corresponding WCEV and Error indices improved as the number of sensors increased. Figs. 2a, b, and d, as more sensors were added by the greedy algorithm, the corresponding index may worsen compared to configurations with fewer sensors. This also highlights the limitation of greedy algorithms, as they aim for local optimization when selecting the current sensor but may not result in a globally optimal sensor configuration.

The two proposed model-based multi-objective optimization strategies perform similarly in terms of performance. Regarding VCE, MSE, and Error indices, MFADG is slightly better, while for WCEV and CN, MFADEG yields better results. The two proposed model-based multi-objective optimization strategies exhibit comparable overall performance. In terms of VCE, MSE, and Error indices, the MFADG method demonstrates slightly better results, indicating superior signal reconstruction quality. Conversely, MFADEG performs better in terms of WCEV and CN, suggesting a stronger emphasis on system stability.

The MFADG sensor placement algorithm achieves the best reconstruction performance while simultaneously ensuring excellent system stability and solution reliability. To ensure transparency, complete statistical test results and have been added as Supplementary information.

Table 1: Computational effort comparison of sensor placement methods.

Algorithm	MFADG	MFADEG	Convex	QRP	REG
Cost	$O(pnr^2)$	$O(p(r^3+nr^2))$	$O(i(r^2+n^3))$	$O(n^3)$	$O(mpn^2)$

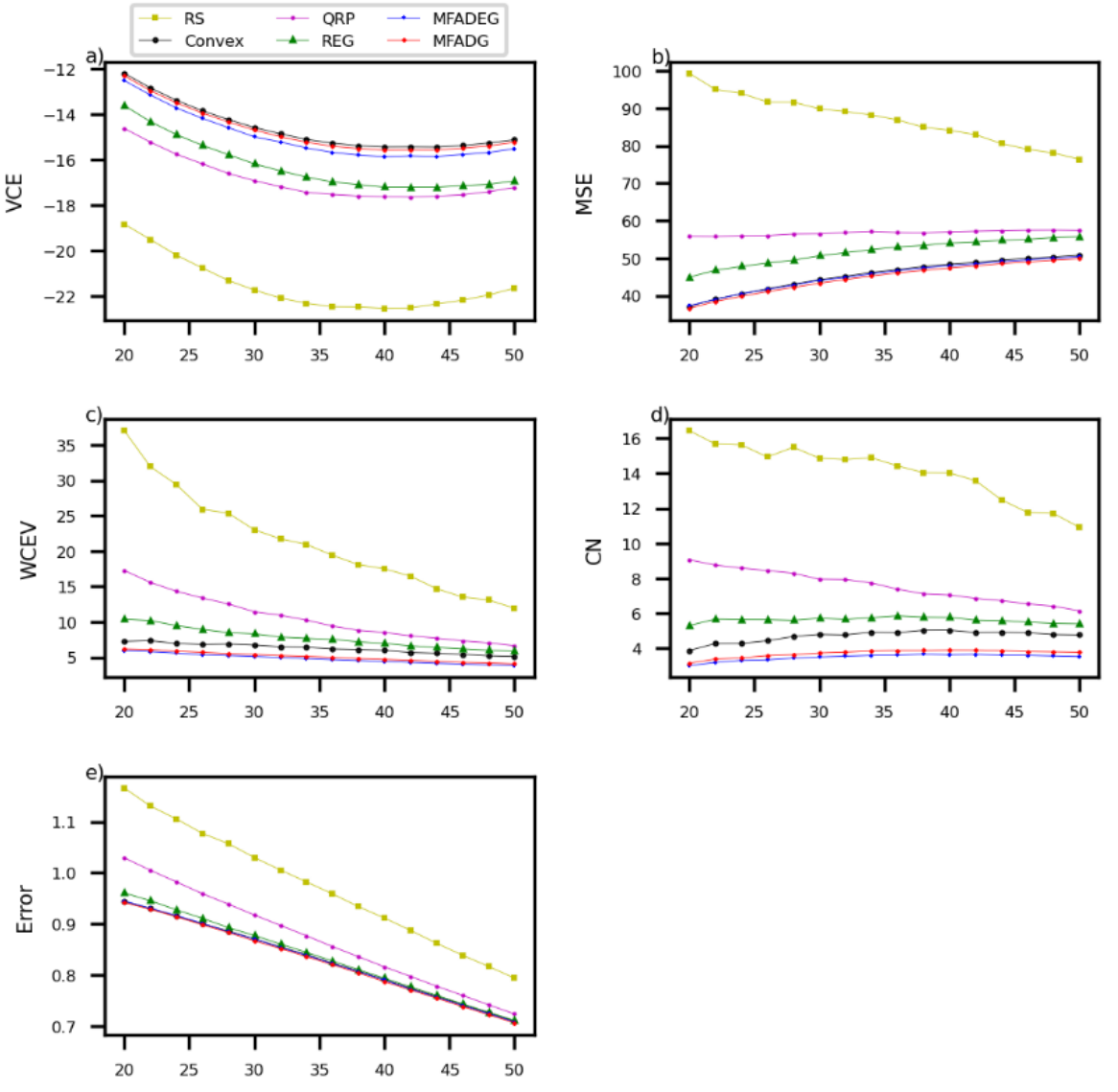

Fig. 2: Comparison of Model Performance between MFADG, MFADG, and State-of-the-Art Methods for Gaussian Random Matrix Input. a) VCE Comparison, b) MSE Comparison, c) WCEV Comparison, d) CN Comparison, e) Error Comparison.

One of the main characteristics of multi-objective optimization is the ability to produce a Pareto front, which comprises a set of equally suitable solutions, also known as Pareto optimal solutions. For multi-objective optimization problems, scalarization transforms them into single-objective optimization problems, where the solutions obtained through greedy algorithms should lie on the Pareto front. In this study, 50 sets of evenly spaced weights $(w, 1-w)$ were generated. Using a greedy algorithm, optimal solutions were computed for fixed weights, along with greedy solutions obtained through the adaptive weights of MFADG. The results are illustrated in Fig. 3.

Fig. 3 provides a detailed comparison of solutions for 16 different sensor placement problems shown in Fig. 2. In each subplot, the red points represent the solutions obtained by the proposed MFADG algorithm, while the black points correspond to the greedy optimal solutions under 50 different fixed weights for multi-objective optimization. These include the single-objective greedy solutions derived with weights $(1,0)$ and $(0,1)$. To facilitate a direct comparison of the relative positions between the fixed-weight solutions and the adaptive-weight solutions, the Pareto front is not explicitly extracted in the figure. However, it can be easily observed based on the definition of the Pareto front. It can be visually observed from the figure that in 14 out of the 16 subplots, the red points representing the adaptive weight solutions are located on the Pareto front. Additionally, the solution quality shows significant improvement compared to the black points representing fixed-weight strategy solutions. For the cases where the number of sensors is $S = 22$ and $S = 44$, although the adaptive weight solutions are dominated by the Pareto optimal solutions of the fixed-weight strategy, they are still positioned near the Pareto front and demonstrate better quality compared to most solutions derived from fixed-weight combinations.

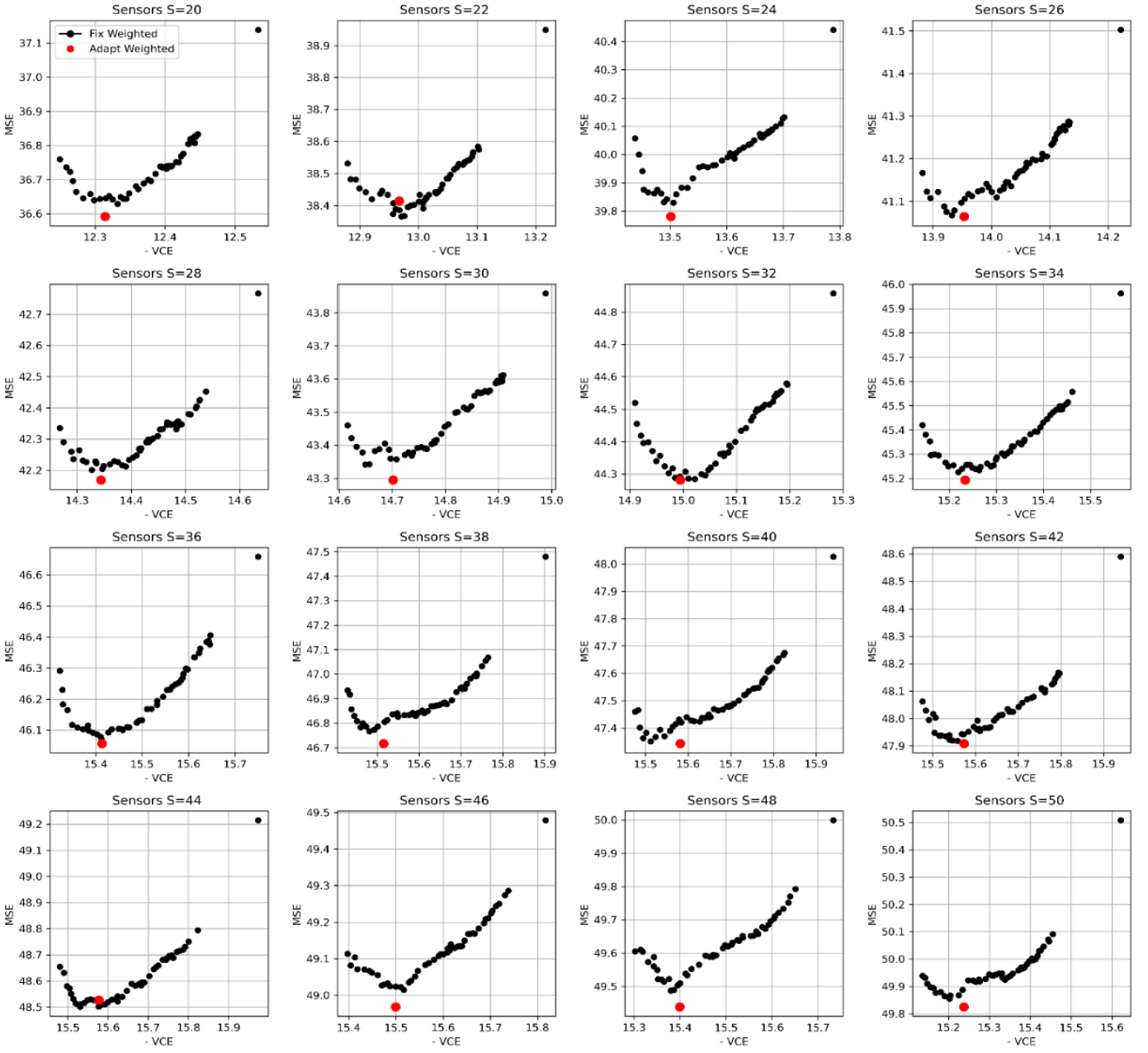

Fig. 3: Fixed-weight and adaptive-weight greedy solutions for the bi-objective optimization problem of A and D under different numbers of sensors in Gaussian random matrices.

3.2 Bernoulli random matrix

Another simulated dataset is generated from a Bernoulli distribution. Each binary matrix $\mathbf{X}$ in this dataset is of size 100×100, where $\mathbf{X}\in\{0,1\}^{100\times100}$. The elements are independently sampled with a success probability of 0.5. A total of 100 binary matrices are produced through Monte Carlo simulations, providing a comprehensive testbed for evaluating sensor placement strategies.

This dataset was used to compare the performance of the proposed method against several state-of-the-art greedy algorithms. Similarly, the performance metrics comparison among various sensor methods is visualized in Fig. 4.

Similarly, the performance of various metrics on Bernoulli random matrices in Fig. 4 is comparable to that in Fig. 2. It is evident that, apart from the convex relaxation method slightly outperforming in the VCE metric, the proposed method demonstrates superior performance in terms of reconstruction accuracy and stability.

This study also investigated the placement of 16 different numbers of sensors, comparing the positions of fixed-weight greedy solutions across 50 different combinations and adaptive-weight greedy solutions, as shown in Fig. 5.

In the placement problems involving 16 different numbers of sensors, the adaptive-weight strategy produced greedy solutions located on the Pareto front in 14 cases. For the remaining two cases, $S=42$ and $S=50$, the adaptive-weight solutions still dominated the majority of the fixed-weight solutions.

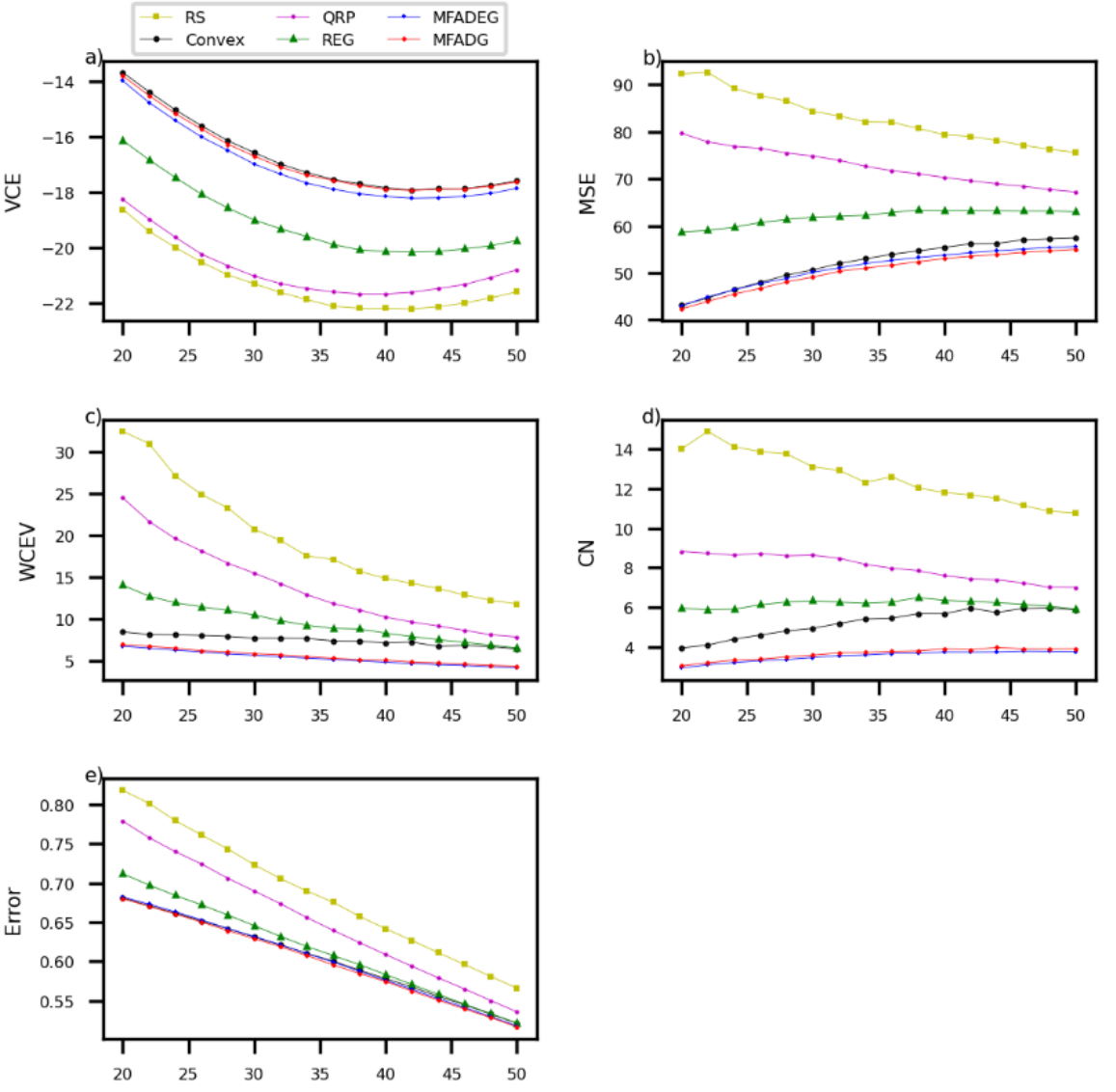

Fig. 4: Comparison of model performance between MFADG, MFADEG, and state-of-the-art methods for Bernoulli random matrix input. a) VCE comparison, b) MSE comparison, c) WCEV comparison, d) CN comparison, e) Error comparison.

Table 2: Statistical comparison of MFADG and MFADEG on Gaussian random matrices (Mann-Whitney U test).

	VCE	MSE	WCEV	CN	Erro
20	0.0016	0.0081	0.0004	0.0027	0.0288
22	0.0073	0.0299	0	0.001	0.2829
24	0.0011	0.0047	0	0.0025	0.0787
26	0.0011	0.03	0	0	0.0434
28	0.0001	0.0009	0	0.0001	0.2384
30	0	0.0004	0	0	0.0063
32	0.0015	0.0064	0	0	0.1228
34	0.0009	0.0047	0	0	0.0065
36	0.0001	0.0054	0	0	0.0727
38	0.0002	0.0012	0	0	0.0328
40	0.0001	0.0013	0	0	0.0027
42	0.0002	0.0018	0	0	0.0568
44	0	0.0031	0	0	0.0419
46	0	0.0013	0	0	0.0201
48	0	0.0005	0	0	0.0432
50	0	0.001	0	0	0.0604

Table 3: Statistical comparison of MFADG and MFADEG on Bernoulli random matrices (Mann-Whitney U test).

	VCE	MSE	WCEV	CN	Erro
20	0.0001	0.0014	0.0046	0.0247	0.0221
22	0	0	0.0004	0.0493	0.0048
24	0	0	0	0.0197	0.0075
26	0	0	0.0442	0.0920	0.0012
28	0	0	0	0.0028	0.0022
30	0	0	0.0001	0.0034	0.0007
32	0	0	0	0.0014	0.0069
34	0	0	0.0004	0.0017	0.0013
36	0	0	0.0009	0.0159	0
38	0	0	0.0005	0.0044	0.0009
40	0	0	0	0.0002	0.0983
42	0	0	0	0.0002	0.0029
44	0	0	0	0	0.0735
46	0	0	0	0	0.0213
48	0	0	0	0.0006	0.1582
50	0	0	0	0	0.0800

To further assess the performance differences between MFADG and its variant MFADEG, we conducted Mann–Whitney U tests on both Gaussian and Bernoulli random matrices, based on 100 Monte Carlo trials. The test results across various sensor configurations (from 20 to 50) are summarized in Tables 2 and 3.

For Gaussian matrices (Table 2), MFADEG achieves significantly better performance in WCEV and CN ($p < 0.001$), indicating improved numerical stability and robustness against ill-conditioning. Meanwhile, MFADG consistently outperforms MFADEG in VCE and MSE ($p < 0.01$), reflecting better data-fitting performance under idealized assumptions. In both cases, no statistically significant difference is observed in the Erro metric ($p > 0.05$), suggesting similar reconstruction accuracy under varied settings. Furthermore, to quantitatively assess the degree of (non-)submodularity in practice, we estimated the submodularity ratio γ for both MFADG and MFADEG. The submodularity ratio is defined as shown in Eq. (36):

$$\gamma = \min_{\substack{P \subseteq Q \subseteq V, \\ j \in V \setminus Q}} \frac{F(P \cup \{j\}) - F(P)}{F(Q \cup \{j\}) - F(Q)} \tag{36}$$

We conducted evaluations 100 Monte Carlo trials and computed γ across different numbers of sensors. The results are presented in Fig. 6, illustrating the empirical behavior of submodularity under varying conditions for both algorithms.

For MFADG, we consistently observed $\gamma \geq 1$ across all trials and sensor numbers. This aligns well with its theoretical submodularity. For MFADEG, the estimated submodularity ratio γ is slightly below 1 in the majority of settings. While the inclusion of the E-optimality criterion leads to mild non-submodular behavior in some cases, the overall performance degradation is marginal. The E-optimality term complements A- and D-optimality by improving worst-case error bounds, albeit at the cost of violating strict submodular structure.

3.3 YaleB dataset

The Extended Yale B dataset is an extension of the original Yale Face Database B (Yale B) and is commonly used for experiments and research in face recognition algorithms.[48] The Yale B dataset has dimensions $\mathbf{X} \in \mathbb{R}^{1024 \times 2414}$, with each snapshot having 1024 features. We applied the proposed MFAD and MFADE to the Yale B facial dataset, as illustrated in Figs. 7 and 8.

Observing Fig. 7, it is evident that in terms of MSE, WCEV, and CN metrics, both MFADG and MFADEG outperform other methods. In WCEV and CN, MFADE performs slightly better than MFADG. In VCE and MSE, MFADG outperforms MFADEG slightly. However, the performance of MFADG and MFADEG in VCE is slightly worse, and they perform worse than Convex Relaxation in most sensors. But in VCE metric, the performances of Convex Relaxation, QRP, REG, MFADG, and MFADEG are very close, unlike in other metrics where there are more noticeable differences. This behavior also aligns with the expectations for multi-objective optimization tasks, as it is unlikely to achieve perfection in all aspects. This can be viewed from the perspective of Pareto Optimal Solutions; if we want to optimize one of the objectives, such as the MSE metric in subfigure b) of Fig. 7, it is difficult to simultaneously optimize VCE as well. The multi-objective methods proposed in this study did not aim to find the Pareto Front, as it generally requires tools such as genetic algorithms or particle swarm optimization, accompanied by considerable computational costs.

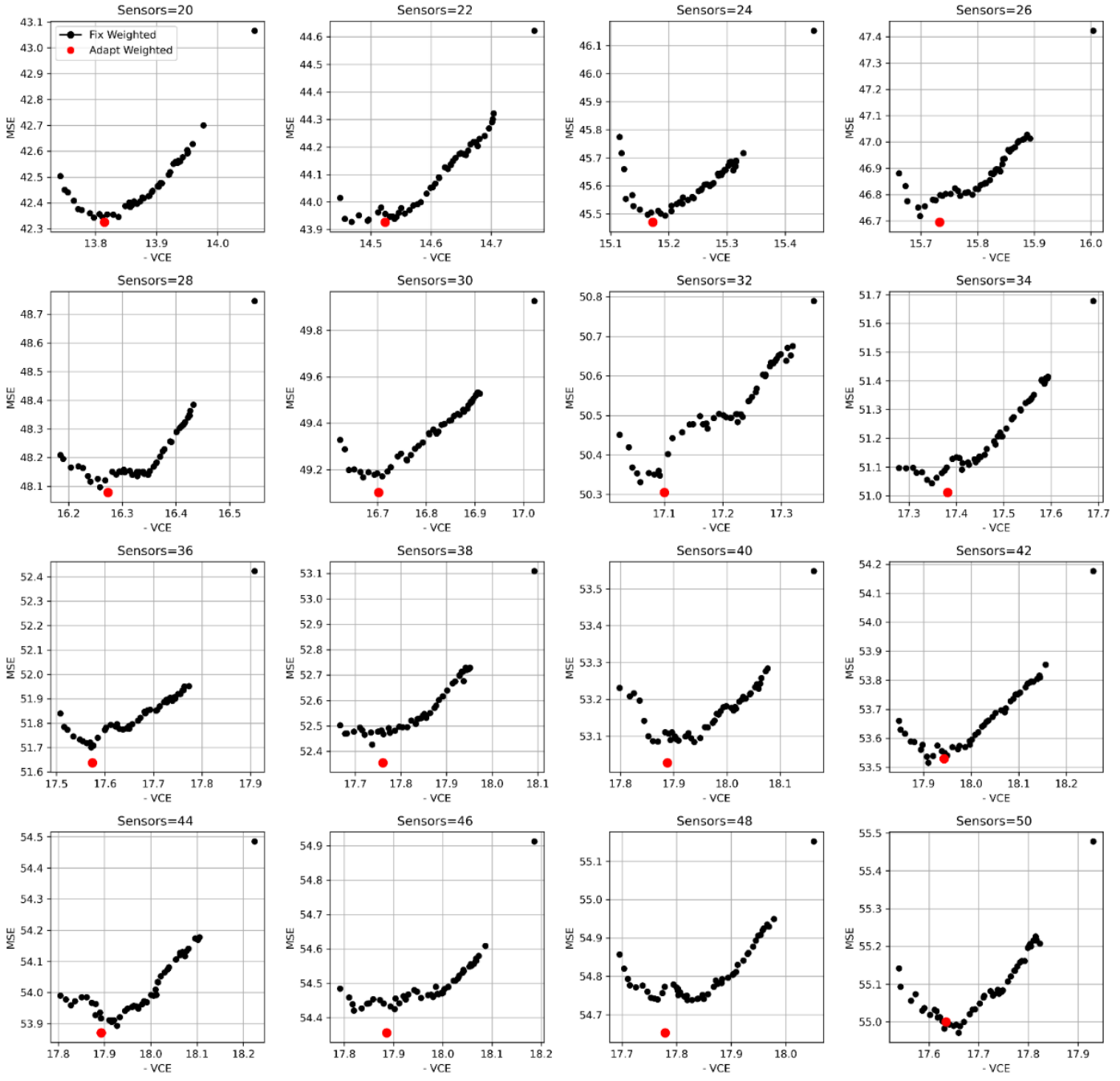

Fig. 5: Fixed-weight and adaptive-weight greedy solutions for the Bi-objective optimization problem of A and D under different numbers of sensors in Bernoulli random matrices.

Fig. 8 illustrates the training and testing errors of the original signals in the Yale B dataset. The testing set exhibits a similar trend in reconstruction errors as the training set. In the testing set, MFADG and MFADEG demonstrate reconstruction errors very close to REG, a greedy method based on reconstruction error, and significantly outperform the Convex Relaxation and QRP methods. When a smaller number of sensors are selected, REG slightly outperforms the MFADG and MFADEG algorithms. However, when $n > 160$, MFADEG's reconstruction performance surpasses that of the REG method. Furthermore, when $n > 180$, MFADG's reconstruction performance also overtakes the REG method.

To gain a more intuitive understanding of the specific implementation of the MFADG and MFADEG methods, we selected a snapshot from the testing set and plotted the reconstruction results and the specific sensor locations when $p = 170$, as shown in Fig. 9.

Fig. 9 illustrates the reconstruction performance of MFADG and MFADEG on specific facial images. Subfigures b) and c) of Fig. 9 show that the proposed MFADG and MFADEG methods reconstruct the main contours of the facial images quite well. Subfigure a) shows that there is a significant overlap in the recommended image sampling points between the two methods. Some sensor locations differ

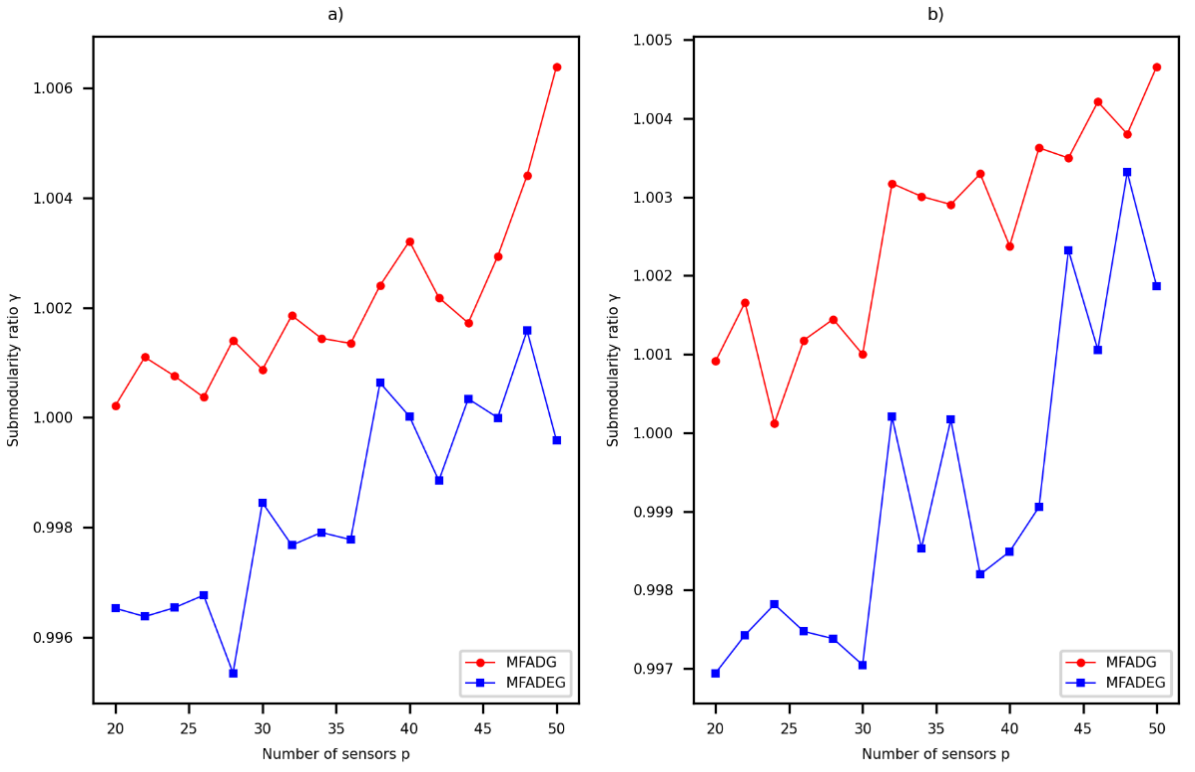

Fig. 6: Estimated submodularity ratio. (a) Gaussian random matrices, (b) Bernoulli random matrices.

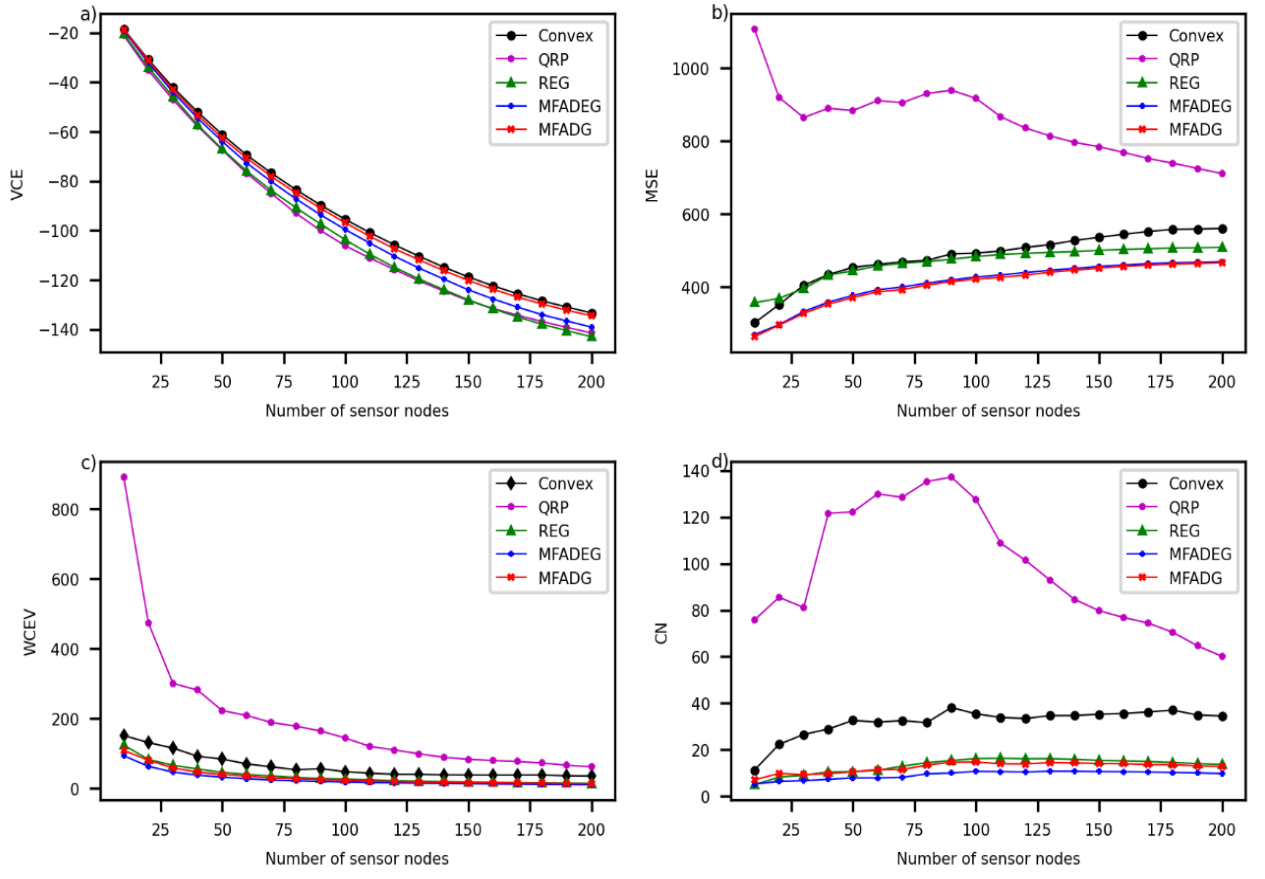

Fig. 7: Comparison of model performance between MFADG and MFADEG and state-of-the-art methods on the Yale B dataset. a) VCE comparison. b) MSE comparison. c) WCEV comparison. d) CN comparison.

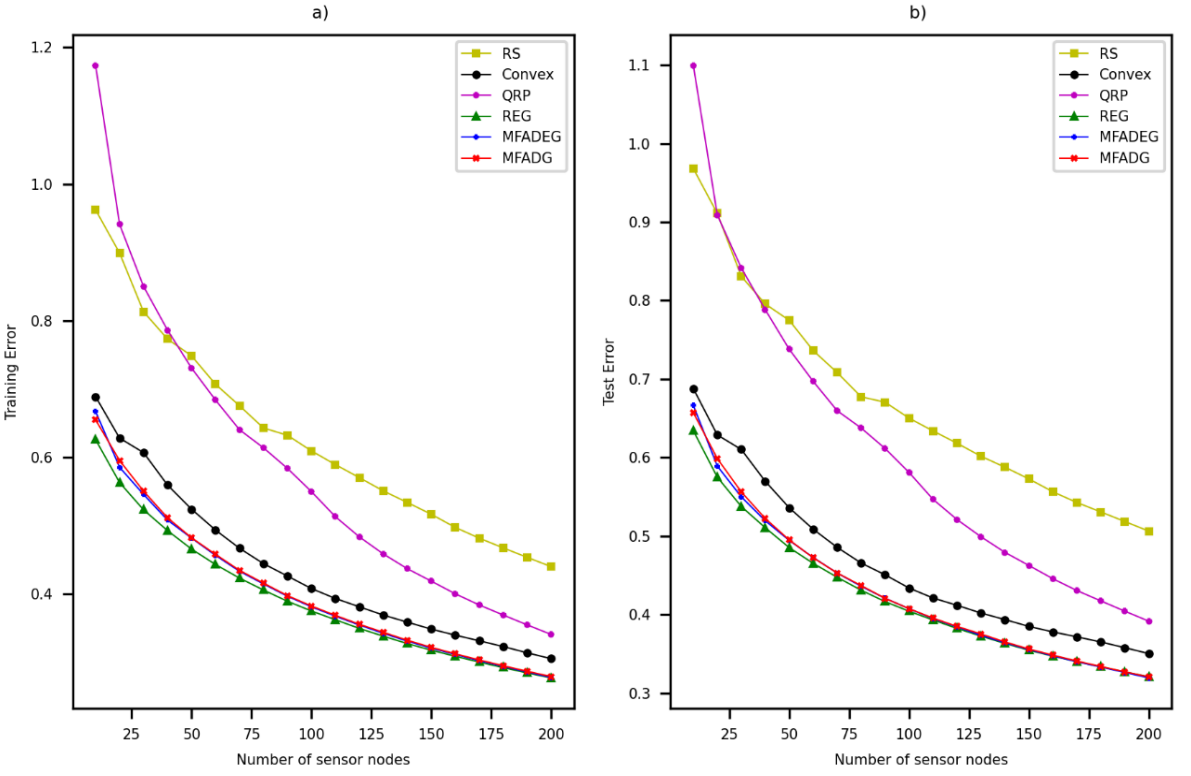

Fig. 8: Comparison of reconstruction errors between MFADG and MFADEG and state-of-the-art methods on the Yale B dataset. a) Training error. b) Test error.

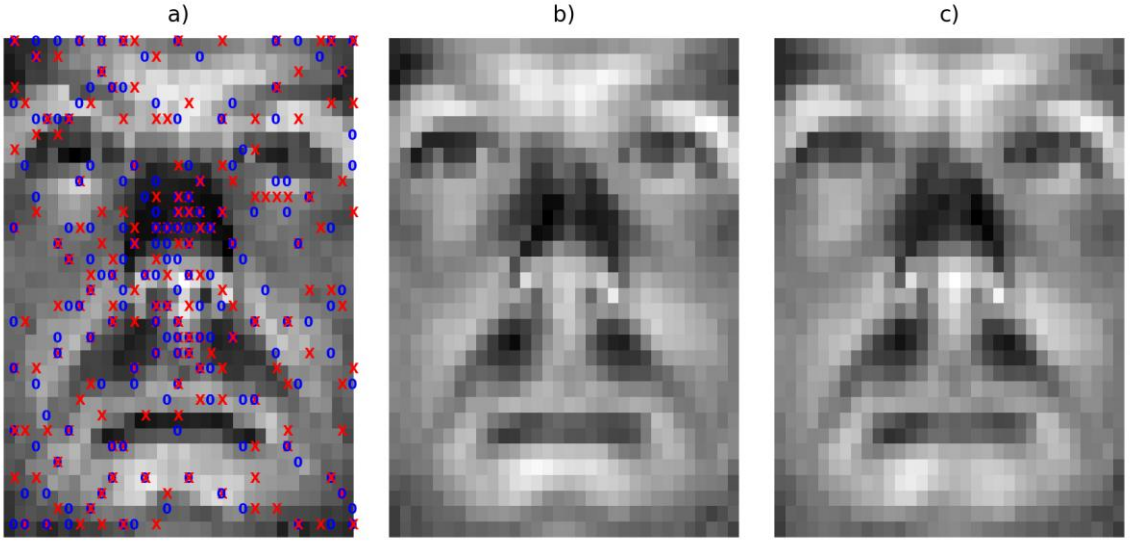

Fig. 9: a) Original image along with sensor locations determined by MFADG and MFADEG algorithms. Red crosses represent sensor locations placed by MFADEG blue circles represent sensor locations placed by MFADG. b) Reconstructed face image using the MFADEG method. c) Reconstructed face image using the MFADG method.

slightly, but both methods perform well in restoring the high-frequency edge portions and low-frequency flat areas of the images. The optimal rank r for Frobenius norm in Equation 2 for the Yale B dataset, obtained using the hard thresholding method,[27] is $r = 166$. The recommended methods use $r = p / 2 = 85$, and they can achieve high-probability recovery of the original signal's maximum information with relatively fewer sampling points through data-driven MFADG and MFADEG greedy sampling of data points in facial images. The advantage of data-driven MFADG and MFADEG methods lies in their ability to guide the sampling point positions using the features of training set images to obtain the maximum information content of the original image with fewer sampling points.

3.4 SST Dataset

The NOAA_OISST_V2 global ocean surface temperature dataset, spanning the period from 1990 to 2023 in weeks, is publicly available online[3]. The original dataset consists of snapshots with a spatial grid size of 360×180 . Due to computational limitations, this study follows the settings of the REG method, reducing the spatial resolution of the dataset. The grid density at this stage is 108×54, resulting in 5832 features per snapshot. This corresponds to reshaping the matrix to $\mathbf{X} \in \mathbb{R}^{5832 \times 1727}$, while simultaneously zero-centering the data. The model performance of all methods on the SST dataset is shown in Fig.10.

Fig. 10 illustrates that the model performance of the sensor configurations recommended by MFADG and MFADEG on the SST dataset is similar to that observed in the Yale B dataset, with both methods performing exceptionally well. In the SST dataset, the performance metrics for VCE are even better than those in the Yale B dataset, approaching the level of convex relaxation while outperforming the other two methods. In terms of MSE, WCEV, and CN, MFAD and MFADE consistently outperform the other three methods across different numbers of sensors. REG approaches the performance of MFAD when a large number of sensors are used in WCEV and CN. In contrast, convex relaxation and QRP perform poorly in terms of the condition number, possibly due to numerical instability when solving the linear measurement system derived from these methods. The performance of all methods in terms of reconstruction error on the SST dataset is illustrated in Fig. 11.

On the SST dataset, Fig. 11 shows that training and testing errors are roughly the same. Moreover, compared to the reconstruction performance on the Yale B dataset, the reconstruction errors of MFADG and MFADEG are closer to those of REG when a large number of sensors are selected. In contrast, when fewer sensors are chosen, the reconstruction performance of MFADG and MFADEG is better than that of REG. Furthermore, in the SST dataset, with a much higher feature dimension than Yale B, when selecting $p \geq 130$ sensors, the reconstruction error is less than 0.2, indicating better adaptability of data-driven sensor methods on the SST dataset.

4. Discussion

In all three datasets, concerning the model performance of linear estimators, except for the slight advantage of Convex Relaxation over MFADG and MFADEG on the VCE index,

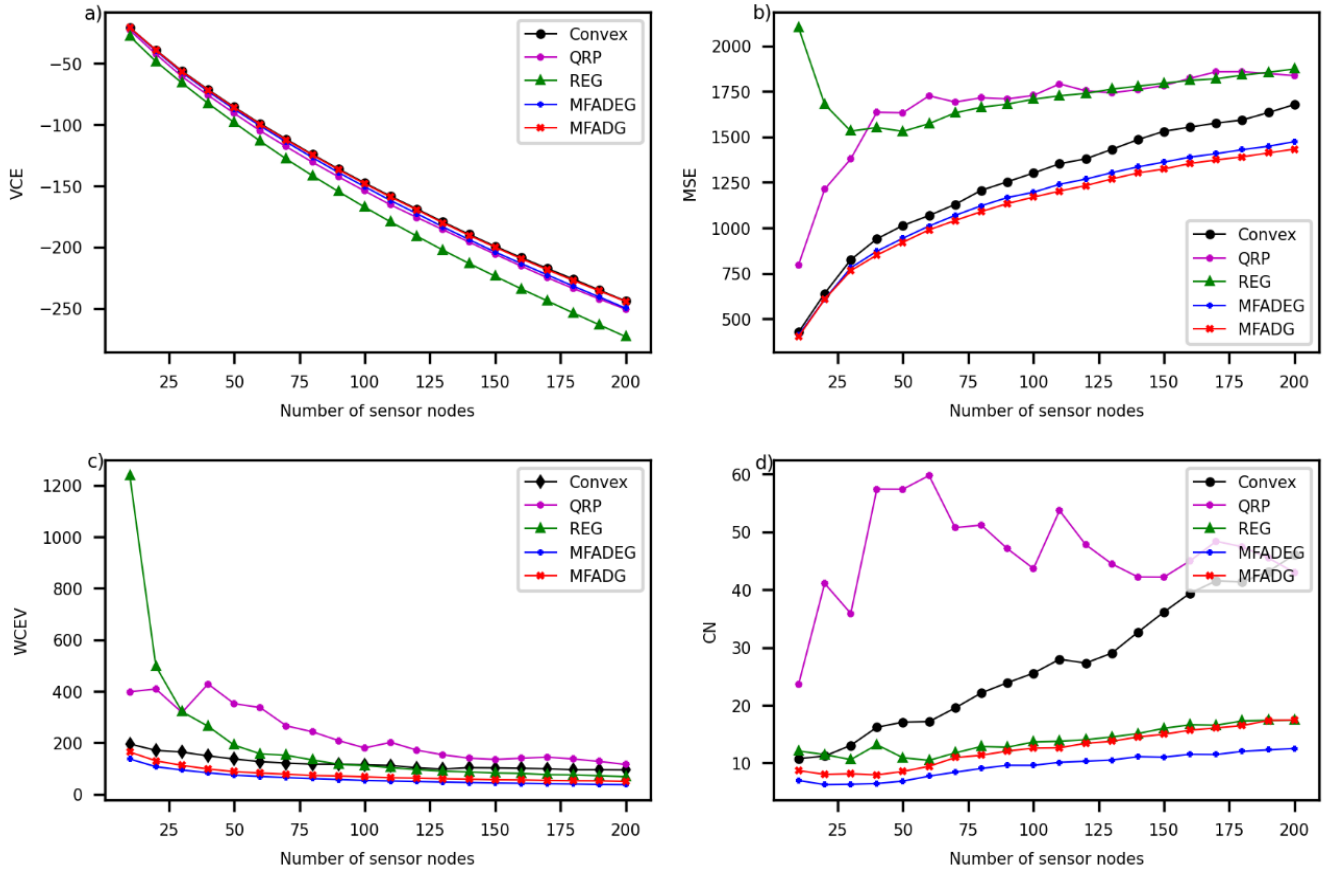

Fig. 10: Comparison of model performance among MFADG, MFADEG, and state-of-the-art methods using the SST dataset. a) VCE comparison, b) MSE comparison, c) WCEV comparison, d) CN comparison.

[3] https://psl.noaa.gov/data/gridded/data.noaa.oisst.v2.html (accessed September 24, 2024)

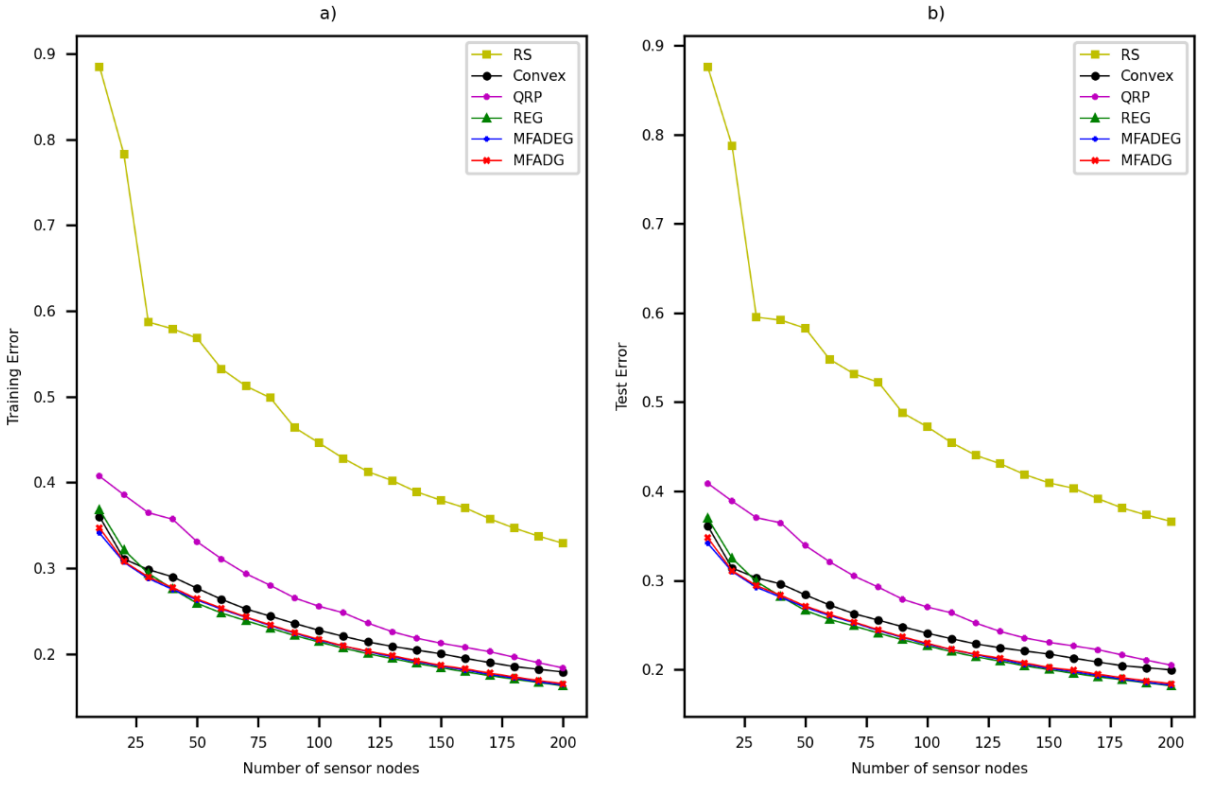

Fig. 11: Comparison of reconstruction errors among MFADG, MFADEG, and state-of-the-art methods using the SST dataset. a) Training error, b) Test error.

MFAD and MFADE outperform the other two methods on the other three metrics, namely MSE, WCEV, and CN. The comparison of algorithm performance here is for the same number of selected sensors p, where VCE is the objective function to be maximized, while MSE, WCEV, and CN are functions to be minimized. However, as the number of selected sensors increases, all these model performance metrics do not improve with the increase in p. The expected behavior is that VCE increases with p, while MSE, WCEV, and CN decrease with p.

One reason for this discrepancy is that greedy solving methods can only identify the next best sensor location $j^\star$ for the current sensor configuration $\mathcal{S}$ in the linear estimation system, but they cannot guarantee that the sensor configuration $\mathcal{S} \cup j^\star$ is also optimal. This suboptimality of greedy sensors can be addressed either at the input stage through group greedy methods or at the output stage through local optimization techniques. For specific approaches, please refer to Joshi and Boyd and Jiang *et al.*[22,23] Another reason is that in several examples, the rank r of the basis changes with the number of selected sensors, *i.e.*, $p = 2r$, leading to trends in certain metrics that deviate from expectations. A possible solution is to determine the optimal rank r, such as using a hard thresholding method, and maintain a fixed basis size r to study the oversampling effects as the number of sensors increases.

From the perspective of reconstruction error, MFADG and MFADEG's performance is very close to the REG method, which is specifically designed for minimizing reconstruction error. They even outperform this sensor placement method designed for minimizing reconstruction error on real datasets Yale B and SST. For all greedy methods, the reconstruction error aligns with our intuition; as the number of selected sensors increases, the reconstruction error decreases.

Although MFADEG and MFADG are fast implementations of multi-objective methods for A-, D-, and E-optimality, the fast implementation of E-optimality in Eq. (17) still requires solving the eigenvector $\mathbf{v}_{\lambda_{min}}(\mathbf{\Psi}_{\mathcal{S}})$ of the minimum eigenvalue $\lambda_{\min}$, which has algorithmic complexities ranging from $O(n^3)$ to $O(n^4)$ for an n-dimensional matrix, and the inversion of the matrix also has a complexity of $O(n^3)$. However, based on the results of solving the sensor placement problem with the MFADG and MFADEG algorithms on the three datasets, MFADEG is only slightly better than MFADG in WCEV and CN. Therefore, one can choose one of the two methods based on the actual performance requirements for the linear estimator.

Furthermore, according to Lemma, Eq. (25) corresponding to the MFAD method has performance guarantees as given in Eq. (14). This is something that Convex Relaxation, QRP, REG, and MFADEG methods do not possess. At the same time, the MFADG method exhibits excellent performance metrics.

5. Conclusion

Solving the problem of reconstructing physical fields in the real world through sensor discretized sampling is a challenging combinatorial optimization problem. One possible solution is to guide the selection of sensor locations through data-driven greedy sensor placement strategies, especially when the objective functions exhibit submodularity, as these greedy methods can provide guarantees on the performance of the selected sensor set.

This study first reiterates that the combination of non-negative linear combinations of single-objective functions with submodularity results in a multi-objective function that also retains submodularity. This property is revalidated from the perspective of the definition of submodular functions. Subsequently, we propose two sensor placement strategies, MFADG and MFADEG, tailored for three objective functions related to optimal experimental design. By introducing a weight update strategy, these multi-objective problems are transformed into single-objective optimization problems, circumventing the need to solve the Pareto front of the multi-objective problem. Among these, the MFADG method is equipped with submodularity guarantees.

We validate our proposed methods on two simulated datasets and two real-world datasets, demonstrating their effectiveness. Experimental results show that MFADG and MFADEG outperform classical sensor placement methods across most of the five performance metrics. If computational complexity is not a primary concern, the MFADEG method can be chosen based on specific requirements. Meanwhile, the MFADG method is recommended as a robust and potentially widely applicable greedy sensor placement strategy with submodularity guarantees. This approach not only allows for real-time updates of sensor location sets based on simulation and measurement data but also avoids the complexity of searching for Pareto-optimal solutions. Furthermore, the solutions obtained by the MFADG algorithm are almost always located on the Pareto front and dominate the majority of fixed-weight solutions.

Acknowledgments

This work was supported by the National Key Research and Development Project (2024YFD2001203); the key Research and Development plan of Anhui Province (2022107020036); the National Natural Science Foundation of China (62033012).

Conflict of Interest

There is no conflict of interest.

Supporting Information

Not applicable.

CRediT Statement

Wangchun Zhang: Conceptualization, Methodology, Software, and Writing-original draft. **Yujun Zhang**: supervision, writing—review, editing, and funding acquisition. **Ying He**: project administration, supervision, validation, writing—review, and editing. **Hao Xie** and **Kun You**: writing—review and editing.

References

[1] M. Farazmand, A. K. Saibaba, Tensor-based flow reconstruction from optimally located sensor measurements, *Journal of Fluid Mechanics*, 2023, **962**, p. A27 doi: 10.1017/jfm.2023.269.

[2] N. J. Nair, A. Goza, Leveraging reduced-order models for state estimation using deep learning, *Journal of Fluid Mechanics*, 2020, **897**, R1, doi: 10.1017/jfm.2020.409.

[3] Q. Zhang, H. Wu, X. Mei, D. Han, M. D. Marino, K. Li, S. Guo, A sparse sensor placement strategy based on information entropy and data reconstruction for ocean monitoring, *IEEE Internet of Things Journal*, 2023, **10**, 19681-19694, doi: 10.1109/JIOT.2023.3281831.

[4] Y. Zi, L. Fan, X. Wu, J. Chen, Z. Han, Distributionally robust optimal sensor placement method for site-scale methane-emission monitoring, *IEEE Sensors Journal*, 2022, **22**, 23403-23412, doi: 10.1109/JSEN.2022.3214176.

[5] J. Gao, L. Zeng, C. Cao, W. Ye, X. Zhang, Multi-objective optimization for sensor placement against suddenly released contaminant in air duct system, *Building Simulation*, 2018, **11**, 139-153, doi: 10.1007/s12273-017-0374-z.

[6] Y. Guo, R. Lin, M. D. Sacchi, Optimal seismic sensor placement based on reinforcement learning approach: an example of OBN acquisition design, *IEEE Transactions on Geoscience and Remote Sensing*, 2023, **61**, 5904112, doi: 10.1109/TGRS.2023.3247593.

[7] C. Yang, An adaptive sensor placement algorithm for structural health monitoring based on multi-objective iterative optimization using weight factor updating, *Mechanical Systems and Signal Processing*, 2021, **151**, 107363, doi: 10.1016/j.ymssp.2020.107363.

[8] B. Zhang, Y. Ni, A data-driven sensor placement strategy for reconstruction of mode shapes by using recurrent Gaussian process regression, *Engineering Structures*, 2023, **284**, 115998, doi: 10.1016/j.engstruct.2023.115998.

[9] J. L. J. Pereira, M. B. Francisco, L. A. de Oliveira, J. A. S. Chaves, S. S. Cunha Jr, G. F. Gomes, Multi-objective sensor placement optimization of helicopter rotor blade based on Feature Selection, *Mechanical Systems and Signal Processing*, 2022, **180**, 109466, doi: 10.1016/j.ymssp.2022.109466.

[10] K. Manohar, J. N. Kutz, S. L. Brunton, Optimal sensor and

actuator selection using balanced model reduction, *IEEE Transactions on Automatic Control*, 2022, **67**, 2108-2115, doi: 10.1109/TAC.2021.3082502.

[11] T. H. Summers, F. L. Cortesi, J. Lygeros, On submodularity and controllability in complex dynamical networks, *IEEE Transactions on Control of Network Systems*, 2016, **3**, 91-101, doi: 10.1109/TCNS.2015.2453711.

[12] C. Yang, K. Liang, X. Zhang, X. Geng, Sensor placement algorithm for structural health monitoring with redundancy elimination model based on sub-clustering strategy, *Mechanical Systems and Signal Processing*, 2019, **124**, 369-387, doi: 10.1016/j.ymssp.2019.01.057.

[13] C. Yang, H. Ouyang, A novel load-dependent sensor placement method for model updating based on time-dependent reliability optimization considering multi-source uncertainties, *Mechanical Systems and Signal Processing*, 2022, **165**, 108386, doi: 10.1016/j.ymssp.2021.108386.

[14] W. Zhang, Y. Zhang, Y. He, K. You, D. Yu, H. Xie, B. Fan, B. Lei, Quantifying gas emissions through vertical radial plume mapping with embedded radial basis function interpolation, *Measurement*, 2023, **217**, 113019, doi: 10.1016/j.measurement.2023.113019.

[15] F. A. Mettler Jr, P. W. Wiest, J. A. Locken, C. A. Kelsey, CT scanning: patterns of use and dose, *Journal of Radiological Protection*, 2000, **20**, 353-359, doi: 10.1088/0952-4746/20/4/301.

[16] H. Wang, K. Yao, G. Pottie, D. Estrin, Entropy-based sensor selection heuristic for target localization, *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, Berkeley California USA, ACM, 2004: 36-45., doi: 10.1145/984622.984628.

[17] C. Guestrin, A. Krause, A. P. Singh, Near-optimal sensor placements in Gaussian processes, *Proceedings of the 22nd International Conference on Machine Learning - ICML '05*, August 7-11, 2005, Bonn, Germany, ACM, 2005, 265-272, doi: 10.1145/1102351.1102385.

[18] D. S. Johnson, C. H. Papadimitriou, K. Steiglitz, Combinatorial optimization: algorithms and complexity, *The American Mathematical Monthly*, 1984, **91**, 209, doi: 10.2307/2322374.

[19] E. L. Lawler, D. E. Wood, Branch-and-bound methods: a survey, *Operations Research*, 1966, **14**, 699-719, doi: 10.1287/opre.14.4.699.

[20] Q. Fu, Q. Li, X. Li, H. Wang, J. Xie, Q. Wang, MOFS-REPLS: a large-scale multi-objective feature selection algorithm based on real-valued encoding and preference leadership strategy, *Information Sciences*, 2024, **667**, 120483, doi: 10.1016/j.ins.2024.120483.

[21] X. Li, Q. Fu, Q. Li, W. Ding, F. Lin, Z. Zheng, Multi-objective binary grey wolf optimization for feature selection based on guided mutation strategy, *Applied Soft Computing*, 2023, **145**, 110558, doi: 10.1016/j.asoc.2023.110558.

[22] S. Joshi, S. Boyd, Sensor selection *via* convex optimization, *IEEE Transactions on Signal Processing*, 2008, **57**, 451-462, doi: 10.1109/TSP.2008.2007095.

[23] C. Jiang, Z. Chen, R. Su, Y. C. Soh, Group greedy method for sensor placement, *IEEE Transactions on Signal Processing*, 2019, **67**, 2249-2262, doi: 10.1109/TSP.2019.2903017.

[24] T. Nagata, K. Yamada, K. Nakai, Y. Saito, T. Nonomura, Randomized Group-greedy method for large-scale sensor selection problems, *IEEE Sensors Journal*, 2023, **23**, 9536-9548, doi: 10.1109/JSEN.2023.3258223.

[25] C. Jiang, Y. C. Soh, H. Li, Sensor placement by maximal projection on minimum eigenspace for linear inverse problems, *IEEE Transactions on Signal Processing*, 2016, **64**, 5595-5610, doi: 10.1109/TSP.2016.2573767.

[26] J. Ranieri, A. Chebira, M. Vetterli, Near-optimal sensor placement for linear inverse problems, *IEEE Transactions on Signal Processing*, 2014, **62**, 1135-1146, doi: 10.1109/TSP.2014.2299518.

[27] K. Manohar, B. W. Brunton, J. N. Kutz, S. L. Brunton, Data-driven sparse sensor placement for reconstruction: demonstrating the benefits of exploiting known patterns, *IEEE Control Systems Magazine*, 2018, **38**, 63-86, doi: 10.1109/MCS.2018.2810460.

[28] K. Wu, P. Chen, O. Ghattas, A fast and scalable computational framework for large-scale high-dimensional Bayesian optimal experimental design, *ASA Journal on Uncertainty Quantification*, 2023, **11**, 235-261, doi: 10.1137/21m1466499.

[29] C. Yang, K. Liang, X. Zhang, Strategy for sensor number determination and placement optimization with incomplete information based on interval possibility model and clustering avoidance distribution index, *Computer Methods in Applied Mechanics and Engineering*, 2020, **366**, 113042, doi: 10.1016/j.cma.2020.113042.

[30] D. Uciński, D-optimal sensor selection in the presence of correlated measurement noise, *Measurement*, 2020, **164**, 107873, doi: 10.1016/j.measurement.2020.107873.

[31] K. Yamada, Y. Saito, K. Nankai, T. Nonomura, K. Asai, D. Tsubakino, Fast greedy optimization of sensor selection in measurement with correlated noise, *Mechanical Systems and Signal Processing*, 2021, **158**, 107619, doi: 10.1016/j.ymssp.2021.107619.

[32] Y. Saito, T. Nonomura, K. Yamada, K. Nakai, T. Nagata, K. Asai, Y. Sasaki, D. Tsubakino, Determinant-based fast greedy sensor selection algorithm, *IEEE Access*, 2021, **9**, 68535-68551, doi: 10.1109/ACCESS.2021.3076186.

[33] K. Nakai, Y. Sasaki, T. Nagata, K. Yamada, Y. Saito, T. Nonomura, Nondominated-solution-based multi-objective

greedy sensor selection for optimal design of experiments, *IEEE Transactions on Signal Processing*, 2022, **70**, 5694-5707, doi: 10.1109/TSP.2022.3224643.

[34] T. Inoue, T. Ikami, Y. Egami, H. Nagai, Y. Naganuma, K. Kimura, Y. Matsuda, Data-driven optimal sensor placement for high-dimensional system using annealing machine, *Mechanical Systems and Signal Processing*, 2023, **188**, 109957, doi: 10.1016/j.ymssp.2022.109957.

[35] Y. Saito, T. Nonomura, K. Nankai, K. Yamada, K. Asai, Y. Sasaki, D. Tsubakino, Data-driven vector-measurement-sensor selection based on greedy algorithm, *IEEE Sensors Letters*, 2020, **4**, 7002604, doi: 10.1109/LSENS.2020.2999186.

[36] B. Li, H. Liu, R. Wang, Efficient sensor placement for signal reconstruction based on recursive methods, *IEEE Transactions on Signal Processing*, 2021, **69**, 1885-1898, doi: 10.1109/TSP.2021.3063495.

[37] Y. Yue, H. You, S. Wang, L. Cao, Improved whale optimization algorithm and its application in heterogeneous wireless sensor networks, *International Journal of Distributed Sensor Networks*, 2021, **17**, 155014772110181, doi: 10.1177/15501477211018140.

[38] M. Chen, W. Du, The predicting public sentiment evolution on public emergencies under deep learning and Internet of Things, *The Journal of Supercomputing*, 2023, **79**, 6452-6470, doi: 10.1007/s11227-022-04900-x.

[39] E. Clark, T. Askham, S. L. Brunton, J. Nathan Kutz, Greedy sensor placement with cost constraints, *IEEE Sensors Journal*, 2019, **19**, 2642-2656, doi: 10.1109/JSEN.2018.2887044.

[40] G. L. Nemhauser, L. A. Wolsey, M. L. Fisher, An analysis of approximations for maximizing submodular set functions: I, *Mathematical Programming*, 1978, **14**, 265-294, doi: 10.1007/BF01588971.

[41] M. Shamaiah, S. Banerjee, H. Vikalo, Greedy sensor selection: leveraging submodularity, *49th IEEE Conference on Decision and Control (CDC)*, December 15-17, 2010, Atlanta, GA, USA, IEEE, 2010, 2572-2577, doi: 10.1109/CDC.2010.5717225.

[42] A. Kohara, K. Okano, K. Hirata, Y. Nakamura, Sensor placement minimizing the state estimation mean square error: Performance guarantees of greedy solutions, *2020 59th IEEE Conference on Decision and Control (CDC)*, December 14-18, 2020, Jeju, Korea, IEEE, 2020, 1706-1711, doi: 10.1109/cdc42340.2020.9304166.

[43] R. Li, N. Mehr, R. Horowitz, Submodularity of optimal sensor placement for traffic networks, *Transportation Research Part B: Methodological*, 2023, **171**, 29-43, doi: 10.1016/j.trb.2023.02.008.

[44] Z. Xu, Y. Guo, J. Homer Saleh, Multi-objective optimization for sensor placement: an integrated combinatorial approach with reduced order model and Gaussian process, *Measurement*, 2022, **187**, 110370, doi: 10.1016/j.measurement.2021.110370.

[45] C. Yang, Y. Xia, Interval Pareto front-based multi-objective robust optimization for sensor placement in structural modal identification, *Reliability Engineering & System Safety*, 2024, **242**, 109703, doi: 10.1016/j.ress.2023.109703.

[46] C. Yang, Y. Xia, A novel two-step strategy of non-probabilistic multi-objective optimization for load-dependent sensor placement with interval uncertainties, *Mechanical Systems and Signal Processing*, 2022, **176**, 109173, doi: 10.1016/j.ymssp.2022.109173.

[47] F. Bach, Learning with submodular functions: a convex optimization perspective, *Foundations and Trends® in Machine Learning*, 2013, **6**, 145-373, doi: 10.1561/2200000039.

[48] A. S. Georghiades, P. N. Belhumeur, D. J. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2001, **23**, 643-660, doi: 10.1109/34.927464.

Publisher's Note: Engineered Science Publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access

This article is licensed under a Creative Commons Attribution 4.0 International License, which permits the use, sharing, adaptation, distribution and reproduction in any medium or format, as long as appropriate credit to the original author(s) and the source is given by providing a link to the Creative Commons license and changes need to be indicated if there are any. The images or other third-party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit http://creativecommons.org/ licenses/by/4.0/.

©The Author(s) 2025