Article

A Gradient-Based Method for Robust Sensor Selection in Hypothesis Testing

Ting Ma ¹, Bo Qian ², Dunbiao Niu ¹, Enbin Song ^{1,*} and Qingjiang Shi ³

¹ College of Mathematics, Sichuan University, Chengdu 610064, China; majiawan27@163.com (T.M.); dunbiaoniu_sc@163.com (D.N.)

² School of Electronic Science and Engineering, Nanjing University, Nanjing 210023, China; boqian@smail.nju.edu.cn

³ School of Software Engineering, Tongji University, Shanghai 201804, China; shiqj@tongji.edu.cn

* Correspondence: e.b.song@163.com

Received: 20 December 2019; Accepted: 23 January 2020; Published: 27 January 2020

Abstract: This paper considers the binary Gaussian distribution robust hypothesis testing under a Bayesian optimal criterion in the wireless sensor network (WSN). The distribution covariance matrix under each hypothesis is known, while the distribution mean vector under each hypothesis drifts in an ellipsoidal uncertainty set. Because of the limited bandwidth and energy, we aim at seeking a subset of p out of m sensors such that the best detection performance is achieved. In this setup, the minimax robust sensor selection problem is proposed to deal with the uncertainties of distribution means. Following a popular method, minimizing the maximum overall error probability with respect to the selection matrix can be approximated by maximizing the minimum Chernoff distance between the distributions of the selected measurements under null hypothesis and alternative hypothesis to be detected. Then, we utilize Danskin's theorem to compute the gradient of the objective function of the converted maximization problem, and apply the orthogonal constraint-preserving gradient algorithm (OCPGA) to solve the relaxed maximization problem without 0/1 constraints. It is shown that the OCPGA can obtain a stationary point of the relaxed problem. Meanwhile, we provide the computational complexity of the OCPGA, which is much lower than that of the existing greedy algorithm. Finally, numerical simulations illustrate that, after the same projection and refinement phases, the OCPGA-based method can obtain better solutions than the greedy algorithm-based method but with up to 48.72% shorter runtimes. Particularly, for small-scale problems, the OCPGA-based method is able to attain the globally optimal solution.

Keywords: wireless sensor network; robust sensor selection; hypothesis testing; Chernoff distance; Danskin's theorem; orthogonal constraint-preserving gradient algorithm

1. Introduction

Wireless sensor networks (WSNs) are extensively used to collect and transmit data in many applications, such as autonomous driving [1], disaster detection [2], target tracking [3], etc. In the WSN, it is usually unaffordable to collect and process all sensor data due to the limitations of power and communication resources [4,5]. Therefore, it is of great significance to choose an optimal subset of sensors such that the best performance is attained only based on data collected by the selected sensors, which is the so-called sensor selection problem.

In the past dozen years, sensor selection has been widely studied in various fields, e.g., estimation [6], target tracking [7], condition monitoring [8], to name a few. For parameter estimation in Kalman filtering dynamic system, [6] chose the optimal subset of sensors in each iteration via minimizing the error covariance matrix of the next iteration. The sensor selection problem for target

tracking in large sensor networks was addressed in [7] based on generalized information gain. In [8], it provided an entropy-based sensor selection method for condition monitoring and prognostics of aircraft engine, which can describe the information contained in the sensor data sets.

Meanwhile, the sensor selection problem in hypothesis testings has also attracted a lot of attention [9–11]. For this type of hypothesis testings, only part of sensors in WSN are activated to transmit observation data, and then decisions are made based on the measurements of selected sensors to achieve the best detection performance. When the optimal sensor selection matrix is fixed, the corresponding hypothesis testing problem is reduced to a common one, which is easy to be dealt with. Hence, it is crucial to solve the involved sensor selection problem.

Work [9] studied the sensor selection for the binary Gaussian distribution hypothesis testing in the Neyman–Pearson framework, where the true distribution under each hypothesis is exactly known. It approximately converted the minimization of the false alarm error probability to the maximization of the Kullback–Leibler (KL) divergence between the distributions of the selected measurements under null hypothesis and alternative hypothesis to be detected. Additionally, [9] proposed a sensor selection framework of first relaxation and then projection for the first time, and provided the greedy algorithm to solve the relaxed problem by optimizing each column vector of the selection matrix.

In practical applications, the events to be detected (i.e., parameters of the hypothesis testing) are usually estimated from training data and affected by some uncertainty factors, such as poorly observation environment and system errors. Then, these parameters are not known precisely, but assumed to lie in some given uncertainty sets [10,11]. In these scenarios, the minimax robust sensor selection problem is formulated to cope with the parameter uncertainty. For the binary Gaussian distribution hypothesis testing under the Neyman–Pearson framework, following the framework in [9], work [10] investigated the involved sensor selection problem with distribution mean under each hypothesis falling in an ellipsoidal uncertainty set (the distribution covariance is known). Furthermore, [11] considered the sensor selection problems involved in the Gaussian distribution robust hypothesis testings with both Neyman–Pearson and Bayesian optimal criteria, where the distribution mean under each hypothesis drifts in an ellipsoidal uncertainty set. For the Bayesian framework, minimizing the maximum overall error probability is approximately converted to maximizing the minimum Chernoff distance. Then the corresponding greedy algorithm together with projection and refinement is also proposed to solve the robust sensor selection problem in the above hypothesis testing.

It has been shown in [11] that the robust sensor selection problem in the hypothesis testing is NP-hard under the Bayesian framework. Nevertheless, when the size of the sensor selection problem is small, its optimal solution can be obtained by the exhaustive method via traversing all possible choices. However, for a large-scale problem, the exhaustive method is not affordable due to its huge computation complexity. Although the aforementioned greedy algorithm-based method (i.e., greedy algorithm, projection and refinement) admits a lower computation complexity than the exhaustive method, it can not arrive at the globally optimal solution in many cases, and its computation complexity is still high for large-scale problems. Therefore, it is significant to seek a more efficient algorithm for solving the robust sensor selection problem in the hypothesis testing of WSN. To our surprise, even though other general sensor selection problems have been continuously investigated, for instance, sparse sensing [12] and sensor selection in sequential hypothesis testing [13], there is little progress for this type of sensor selection problems since [11] was published in 2011, which motivates our research.

In this paper, we consider the same binary Gaussian distribution robust hypothesis testings under a Bayesian framework as in [11], where the distribution mean under each hypothesis lies in an ellipsoidal uncertainty set. We attempt to select an optimal subset of sensors such that the maximum overall error probability is minimized. Following the similar idea in [11], minimizing the maximum overall error probability is approximated by maximizing the minimum Chernoff distance between two distributions under null hypothesis and alternative hypothesis to be detected. Then, our main contributions can be summarized as follows.

- First, we succeed in converting the maximinimization of the Chernoff distance to a maximization problem, and adopting the orthogonal constraint-preserving gradient algorithm (OCPGA) [14] to obtain a stationary point of the relaxed maximization problem without 0/1 constraints.
- Specifically, when implementing the OCPGA to the relaxed maximization problem, we utilize the Danskin's theorem [15] to acquire its gradient. Furthermore, the efficient bisection is applied to get the means of distributions under null hypothesis and alternative hypothesis to be detected corresponding to the minimum Chernoff distance.
- The computational complexity of the OCPGA is shown to be lower than that of the greedy algorithm in [11] from the theoretical point of view, while numerical simulations show that the OCPGA-based method (i.e., OCPGA, projection and refinement) can obtain better solutions than the greedy algorithm-based method (i.e., R-C algorithm in [11]) with up to 48.72% shorter runtimes. Therefore, better solutions are available for our proposed OCPGA-based method in some scenarios.

The remainder of this paper is organized as follows. Section 2 states the problem formulation. The proposed OCPGA, as well as projection and refinement phases, is characterized in Section 3. In Section 4, the existence and computation of gradient are provided. Section 5 presents some numerical experiments to corroborate our theoretical results, while Section 6 concludes the paper.

Notations: Denote $\mathbb{R}^m$ and $\mathbb{R}^{m \times p}$ as the m -dimensional real vector space and $m \times p$ dimensional real matrix space, respectively. Let $\mathbf{I}$ and $\mathbf{0}$ be the identity matrix and the zero matrix whose dimensions will be clear from the context; bold-face lower-case letters are used for vectors, while bold-face upper-case letters are for matrices. $\mathcal{N}(\boldsymbol{\mu}, \mathbf{S})$ represents the Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\mathbf{S}$. For matrix $\mathbf{A}$, $\text{tr}(\mathbf{A})$, $|\mathbf{A}|$, $\|\mathbf{A}\|_F$, $\mathbf{A}^H$ and $A_{i,j}$ denote its trace, determinant, Frobenius norm, conjugate transpose and the (i, j) -th entry, respectively. For square matrices $\mathbf{A}$ and $\mathbf{B}$, $\mathbf{A} \succeq (\succ) \mathbf{B}$ represents that $\mathbf{A} - \mathbf{B}$ is semi-positive (positive) definite. For a semi-positive definite matrix $\mathbf{A}$, $\mathbf{A}^{\frac{1}{2}}$ stands for its square-rooting matrix.

2. Problem Formulation

2.1. System Model

Define $\mathbf{x} = (x_1, x_2, \dots, x_m)^T \in \mathbb{R}^m$ as the observation vector of all m sensors (each sensor corresponds to one-dimensional measurement). Consider the same binary Gaussian distribution robust hypothesis testing as in [11]:

$$\mathcal{H}_0 : \mathbf{x} \sim \mathcal{N}(\mathbf{m}_0, \mathbf{S}_0), \quad \mathcal{H}_1 : \mathbf{x} \sim \mathcal{N}(\mathbf{m}_1, \mathbf{S}_1), \quad (1)$$

where mean vector $\mathbf{m}_i$ falls in a given ellipsoidal uncertainty set $\mathcal{E}(\bar{\mathbf{m}}_i, k_i \mathbf{S}_i^{-1}) = \{\mathbf{x} \in \mathbb{R}^m | (\mathbf{x} - \bar{\mathbf{m}}_i)^T \mathbf{S}_i^{-1} (\mathbf{x} - \bar{\mathbf{m}}_i) \leq \frac{1}{k_i}\}$, $\bar{\mathbf{m}}_i$ denotes the mean estimated by training data, covariance $\mathbf{S}_i$ is a known matrix, and $k_i \in (0, +\infty]$ is the robustness parameter, $i = 0, 1$. Obviously, when $k_i = +\infty$, the ellipsoidal uncertainty set is reduced to a single point, and thereby $\mathbf{m}_i = \bar{\mathbf{m}}_i$, i.e., there is no uncertainty.

Sensor Selection: In the WSN, sensors transmit their observations to a fusion node, which then performs the hypothesis testing based on its received measurements. Due to power constraints, suppose that only p out of m sensors are chosen ($p < m$) to transmit the observations to the fusion node. We aim at selecting p sensors to guarantee the best detection performance, that is, seeking a selection matrix $\mathbf{E} \in \mathbb{R}^{m \times p}$ with 0/1 elements such that the best detection performance based on measurements $\mathbf{y} = \mathbf{E}^T \mathbf{x} \in \mathbb{R}^p$ is achieved. It is easy to see that, $\mathbf{E}$ has exactly one unit entry per column (corresponds to a selected sensor) and at most one unit entry per row (each sensor is selected at most once). Therefore, $\mathbf{E}$ should be a column orthogonal matrix, i.e., $\mathbf{E}^T \mathbf{E} = \mathbf{I}$.

Hypothesis Testing Induced by E : Owing to $\mathbf{y} = E^T \mathbf{x}$, the original hypothesis testing (1) about $\mathbf{x}$ in the high dimensional space $\mathbb{R}^m$ is converted to one about $\mathbf{y}$ in a lower-dimensional space $\mathbb{R}^p$:

$$\mathcal{H}_0 : \mathbf{y} \sim \mathcal{N}(E^T \mathbf{m}_0, E^T \mathbf{S}_0 E), \quad \mathcal{H}_1 : \mathbf{y} \sim \mathcal{N}(E^T \mathbf{m}_1, E^T \mathbf{S}_1 E). \quad (2)$$

Without loss of generality, we assume that

$$\mathcal{N}(E^T \mathbf{m}_0, E^T \mathbf{S}_0 E) \neq \mathcal{N}(E^T \mathbf{m}_1, E^T \mathbf{S}_1 E). \quad (3)$$

Otherwise, the hypothesis testing (2) makes no sense. For hypothesis testing (2), when E and $\mathbf{m}_i$ are determined, the fusion node executes the following likelihood ratio (LR) test:

$$l(\mathbf{y}) := \frac{f_1(\mathbf{y}; E)}{f_0(\mathbf{y}; E)} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\geq}} \gamma,$$

where $f_i(\mathbf{y}; E)$ is the density of $\mathcal{N}(E^T \mathbf{m}_i, E^T \mathbf{S}_i E)$, $i = 0, 1$, and γ is the test threshold [16].

Sensor Selection Optimality Criteria: Under a Bayesian framework, detection performance is quantified by the overall error probabilities P_e , where $P_e := P(\mathcal{H}_0)P_F + P(\mathcal{H}_1)P_M$ with $P_F := P(l(\mathbf{y}) > \gamma | \mathcal{H}_0)$ and $P_M := P(l(\mathbf{y}) < \gamma | \mathcal{H}_1)$ being the false alarm and the miss detection probabilities respectively. Then, the robust sensor selection problem in the hypothesis testing under a Bayesian framework is to seek the selection matrix E such that the maximum overall error probability P_e is minimized when making decisions with respect to hypothesis testing (2), that is, solving the following optimization problem:

$$\begin{aligned} \min_{E \in \mathbb{R}^{m \times p}} \quad & \max_{(\mathbf{m}_0, \mathbf{m}_1) \in \mathcal{E}(\bar{\mathbf{m}}_0, k_0 \mathbf{S}_0^{-1}) \times \mathcal{E}(\bar{\mathbf{m}}_1, k_1 \mathbf{S}_1^{-1})} P_e \\ \text{s.t.} \quad & E^T E = \mathbf{I}, E_{i,j} \in \{0, 1\}. \end{aligned} \quad (4)$$

2.2. Problem Transformation

Since computing P_e in problem (4) is usually difficult due to the involved integrals, we follow the popular approach in [11] to approximately optimize P_e . Based on the Chernoff lemma [17], when the number of independent identically distributed (i.i.d.) measurements increases, the rate of exponential decay of P_e is equal to the Chernoff distance between the two distributions $\mathcal{N}(E^T \mathbf{m}_0, E^T \mathbf{S}_0 E)$ and $\mathcal{N}(E^T \mathbf{m}_1, E^T \mathbf{S}_1 E)$. On the other hand, according to the definition of Chernoff distance between two probability densities f_0 and f_1 :

$$D_C(f_1, f_0) := \max_{s \in [0, 1]} \log \int f_1^s(x) f_0^{1-s}(x) dx,$$

we derive that the Chernoff distance between distributions $\mathcal{N}(E^T \mathbf{m}_0, E^T \mathbf{S}_0 E)$ and $\mathcal{N}(E^T \mathbf{m}_1, E^T \mathbf{S}_1 E)$ is

$$f_C(E, \mathbf{m}_0, \mathbf{m}_1) := \max_{s \in [0, 1]} f(E, s, \mathbf{m}_0, \mathbf{m}_1) \quad (5)$$

with

$$\begin{aligned} f(E, s, \mathbf{m}_0, \mathbf{m}_1) &:= \frac{1}{2} s(1-s)(\mathbf{m}_1 - \mathbf{m}_0)^T E (E^T (s \mathbf{S}_0 + (1-s) \mathbf{S}_1) E)^{-1} E^T \\ &\quad \times (\mathbf{m}_1 - \mathbf{m}_0) - \frac{1}{2} \log \frac{|E^T \mathbf{S}_0 E|^s |E^T \mathbf{S}_1 E|^{1-s}}{|E^T (s \mathbf{S}_0 + (1-s) \mathbf{S}_1) E|}. \end{aligned} \quad (6)$$

Therefore, as in [11], minimizing the maximum overall error probability P_e can be approximately converted to maximizing the minimum Chernoff distance $f_C(E, \mathbf{m}_0, \mathbf{m}_1)$. Accordingly, problem (4) is transformed into

$$\begin{aligned} \max_E \quad & \min_{(\mathbf{m}_0, \mathbf{m}_1) \in \mathcal{E}(\tilde{\mathbf{m}}_0, k_0 S_0^{-1}) \times \mathcal{E}(\tilde{\mathbf{m}}_1, k_1 S_1^{-1})} f_C(E, \mathbf{m}_0, \mathbf{m}_1) \\ \text{s.t.} \quad & E^T E = \mathbf{I}, E_{i,j} \in \{0, 1\}, \end{aligned} \quad (7)$$

or equivalently,

$$\begin{aligned} \max_E \quad & \min_{(\mathbf{m}_0, \mathbf{m}_1) \in \mathcal{E}(\tilde{\mathbf{m}}_0, k_0 S_0^{-1}) \times \mathcal{E}(\tilde{\mathbf{m}}_1, k_1 S_1^{-1})} \max_{s \in [0, 1]} f(E, s, \mathbf{m}_0, \mathbf{m}_1) \\ \text{s.t.} \quad & E^T E = \mathbf{I}, E_{i,j} \in \{0, 1\}. \end{aligned} \quad (8)$$

Work [11] has proven that problem (7) is NP-hard, and proposed a suboptimal greedy method along with projection and refinement phases to deal with problem (7). Although the greedy algorithm-based method (i.e., R-C algorithm in [11]) admits a lower computation complexity than the exhaustive method, it can not arrive at the globally optimal solution in many cases, and still remains high computation complexity for large-scale problems. Therefore, we endeavor to propose a more efficient method to obtain a better solution of problem (7).

It is not difficult to see that, solving problem (7) can be sequentially divided into the inner minimization and outer maximization. For a given selection matrix E , defining

$$\tilde{f}_C(E) := \min_{(\mathbf{m}_0, \mathbf{m}_1) \in \mathcal{E}(\tilde{\mathbf{m}}_0, k_0 S_0^{-1}) \times \mathcal{E}(\tilde{\mathbf{m}}_1, k_1 S_1^{-1})} f_C(E, \mathbf{m}_0, \mathbf{m}_1), \quad (9)$$

and denoting $(\hat{\mathbf{m}}_0(E), \hat{\mathbf{m}}_1(E))$ as the optimal solution of the following subproblem

$$\min_{(\mathbf{m}_0, \mathbf{m}_1) \in \mathcal{E}(\tilde{\mathbf{m}}_0, k_0 S_0^{-1}) \times \mathcal{E}(\tilde{\mathbf{m}}_1, k_1 S_1^{-1})} f_C(E, \mathbf{m}_0, \mathbf{m}_1), \quad (10)$$

then it holds $\tilde{f}_C(E) = f_C(E, \hat{\mathbf{m}}_0(E), \hat{\mathbf{m}}_1(E))$. Correspondingly, problem (7) can be equivalently transformed into

$$\begin{aligned} \text{Primal Problem (PP):} \quad & \max_{E \in \mathbb{R}^{m \times p}} \tilde{f}_C(E) \\ \text{s.t.} \quad & E^T E = \mathbf{I}, E_{i,j} \in \{0, 1\}. \end{aligned} \quad (\text{PP})$$

Remarkably, the optimal solutions of problems (7) and (PP) are the same. Hence, we will discuss how to solve problem (PP) in the following. Taking the orthogonal constraint $E^T E = \mathbf{I}$ into account, we adopt the OCPGA-based method to deal with problem (PP).

3. OCPGA-Based Method

Referring to the greedy algorithm-based method proposed in [11], solving problem (PP) can be also successively divided into three phases: relaxation, projection and refinement. Then we will utilize the OCPGA-based method to solve problem (PP). That is, in the relaxation phase the OCPGA is used to handle the relaxed problem without 0/1 constraints, while the projection and refinement phases are the same as in the greedy algorithm-based method. First we provide the working flow of the OCPGA-based method for solving problem (PP) in Figure 1, and the details are shown in the following Sections 3.1 and 3.2.

Figure 1. Working Flow of the OCPGA Based Method for Solving Problem (PP).

3.1. Relaxation Phase

By relaxing the 0/1 constraints, problem (PP) is reduced to

$$\begin{aligned} \text{Relaxed Problem (RP):} \quad & \max_{E \in \mathbb{R}^{m \times p}} \tilde{f}_C(E) \\ \text{s.t.} \quad & E^T E = I. \end{aligned} \quad (\text{RP})$$

Taking into consideration the orthogonal constraint in problem (RP), once the gradient $\nabla \tilde{f}_C(E)$ of the objective function $\tilde{f}_C(E)$ exists and is computable, we can implement the OCPGA in [14] to solve problem (RP), which is presented in Algorithm 1.

Algorithm 1: OCPGA

Input: Column orthogonal matrix E_0 , $\tau > 0$, τ_l , τ_u , ρ_1 , θ , η , $\epsilon \in (0, 1)$, C_0 , $Q_0 = 1$, $n = 0$.

Output:

Compute $\nabla \tilde{f}_C(E_0)$, $G_0 = -\nabla \tilde{f}_C(E_0) + E_0 \nabla \tilde{f}_C(E_0)^T E_0$.

while $\|G_n\|_F > \epsilon$ **do**

$H_n = -\nabla \tilde{f}_C(E_n) E_n^T + E_n \nabla \tilde{f}_C(E_n)^T$, $Y_n(\tau) = (I + \frac{\tau}{2} H_n)^{-1} (I - \frac{\tau}{2} H_n) E_n$;

while $-\tilde{f}_C(Y_n(\tau)) \geq C_n - \rho_1 \tau \frac{1}{2} \|H_n\|_F^2$ **do**

$\tau = \theta \tau$, $Y_n(\tau) = (I + \frac{\tau}{2} H_n)^{-1} (I - \frac{\tau}{2} H_n) E_n$.

end

$E_{n+1} = Y_n(\tau)$, $Q_{n+1} = \eta Q_n + 1$, $C_{n+1} = (\eta Q_n C_n - \tilde{f}_C(E_{n+1})) / Q_{n+1}$;

Compute the gradient $\nabla \tilde{f}_C(E_{n+1})$, $G_{n+1} = -\nabla \tilde{f}_C(E_{n+1}) + E_{n+1} \nabla \tilde{f}_C(E_{n+1})^T E_{n+1}$;

$T_n = G_{n+1} - G_n$, $S_n = E_{n+1} - E_n$, $\tau_{n+1} = \frac{\text{tr}(S_n^T S_n)}{|\text{tr}(S_n^T T_n)|}$;

Let $\tau = \max(\min(\tau_{n+1}, \tau_u), \tau_l)$, $n = n + 1$.

end

return $E^* = E_n$.

The update formula of $Y_n(\tau)$ in Algorithm 1 is the Cayley transformation [18], and thereby $Y_n(\tau)$ always satisfies the orthogonal constraint $Y_n(\tau)^T Y_n(\tau) = I$ for each iteration n . Meanwhile, the stepsize τ in Algorithm 1 is chosen by curvilinear search algorithms [19] combined with the Barzilai-Borwein (BB) [20] nonmonotonic line search [21]. It has been shown by Lemma 2.2 and Remark 2.3 in [22] that the sequence generated by the OCPGA is globally convergent to a stationary point. For clarity and integrity, we provide the convergence result of Algorithm 1 in the following theorem without proof.

Theorem 1 ([22]). When $\tilde{f}_C(E)$ in problem (RP) is differentiable and the gradient $\nabla \tilde{f}_C(E)$ is derived, then E^* obtained by the OCPGA in Algorithm 1 is a stationary point of problem (RP).

Moreover, the computation complexity of the OCPGA in Algorithm 1 is $\mathcal{O}(mp^2)$ [14]. Combined with the fact that p is generally much smaller than m , the computation complexity of the OCPGA is much lower than that $\mathcal{O}(m^3 p)$ of the greedy algorithm in [11]. Hence, our proposed OCPGA is more efficient than the greedy algorithm, particularly for a quite large m .

3.2. Projection and Refinement Phases

Generally, the solution E^* obtained by the OCPGA in Algorithm 1 is not a selection matrix, because the elements of E^* are not guaranteed to be 0/1. Thus we need to further execute the projection and refinement phases as in [11].

First, we seek matrix $\tilde{E}$ closest to the range space of E^* by solving the following problem

$$\begin{aligned} \min_E \quad & \|EE^T - E^*(E^*)^T\|_F \\ \text{s.t.} \quad & E_{i,j} \in \{0, 1\}, \quad E^T E = \mathbf{I}. \end{aligned} \quad (11)$$

It has been shown in [11] that problem (11) admits a closed-form solution. Specifically, let $(j_1, \dots, j_p)$ be the indexes of the p largest entries on the diagonal of $E^*(E^*)^T$, and then $\tilde{E} = (\mathbf{i}_{j_1}, \dots, \mathbf{i}_{j_p})$ is the optimal solution of problem (11), where $\mathbf{i}_j$ stands for the j -th column of the identity matrix $\mathbf{I}_m$.

Subsequently, after projecting to the set of 0/1 selection matrices, we further implement a refinement around $\tilde{E}$. Setting $E = \tilde{E}$, the first column of E is viewed as the optimization variable, while all other columns are fixed. Then we sweep through all canonical vectors (i.e., columns of the identity matrix $\mathbf{I}_m$) different from the remaining $p - 1$ columns of E , and choose the one corresponding to the maximum $\tilde{f}_C$ as the first column. In the next step, the procedure is repeated for the second column, and so on, up to the p -th step. Finally, one refinement is finished and the matrix $\hat{E} = E$ is regarded as a solution of problem (PP). Obviously, the more we refine, the better solution we can achieve. When $p = 1$, the solution achieved by one refinement is indeed the globally optimal selection matrix of problem (PP). If time permits, we can execute the refinement phase for several times until the objective function value $\tilde{f}_C$ keeps unchanged.

Notice that, the computation cost for projection is very small (as there exists an analytical solution), while in refinement we need to calculate the objective function value $\tilde{f}_C$ for $(m - p)p$ times. Since the OCPGA is more efficient than the greedy algorithm, and the remaining projection and refinement phases are the same, the OCPGA-based method naturally possesses higher efficiency than the greedy algorithm-based method.

Remark 1. For the OCPGA in Algorithm 1, its initial point is randomly chosen from column orthogonal matrices. Experiments illustrate that different initial points may result in different outcomes. Therefore, in order to improve the performance, we implement the OCPGA with different initial points several times, and after projection and refinement, choose the best solution as the output.

Remark 2. According to the above discussion, the key to implement the OCPGA-based method is computing the gradient $\nabla \tilde{f}_C(E)$ in problem (RP). In the next section, we will show how to obtain the gradient $\nabla \tilde{f}_C(E)$ and discuss when it exists.

4. Existence and Computation of the Gradient in Problem (RP)

As can be easily seen from Algorithm 1, it is essential to compute the gradient $\nabla \tilde{f}_C(E)$ in problem (RP). Invoking the definition of $\tilde{f}_C(E)$ in Equation (9), we have

$$\tilde{f}_C(E) = \min_{(\mathbf{m}_0, \mathbf{m}_1) \in \mathcal{E}(\tilde{\mathbf{m}}_0, k_0 \mathbf{S}_0^{-1}) \times \mathcal{E}(\tilde{\mathbf{m}}_1, k_1 \mathbf{S}_1^{-1})} f_C(E, \mathbf{m}_0, \mathbf{m}_1),$$

with $f_C(E, \mathbf{m}_0, \mathbf{m}_1) = \max_{s \in [0, 1]} f(E, s, \mathbf{m}_0, \mathbf{m}_1)$. To proceed, we first prove the strict concavity of $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ with respect to s , which forms a key ingredient of our later arguments.

Proposition 1. Given E and $(\mathbf{m}_0, \mathbf{m}_1)$, $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ defined by Equation (6) is a strictly concave function of s .

Proof. It is easily seen that $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ is continuously differentiable with respect to s for given E and $(\mathbf{m}_0, \mathbf{m}_1)$. Hence we turn to show the positiveness of the second derivative $\nabla_s^2 f(E, s, \mathbf{m}_0, \mathbf{m}_1)$.

For notation brevity, define $\Lambda := (\mathbf{m}_1 - \mathbf{m}_0)(\mathbf{m}_1 - \mathbf{m}_0)^T$ and $B(s) := E(E^T(s\mathbf{S}_0 + (1-s)\mathbf{S}_1)E)^{-1}E^T$. It is clear that both Λ and $B(s)$ are semi-positive definite. By taking derivative of $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ with respect to s , we have

$$\begin{aligned} \nabla_s f(E, s, \mathbf{m}_0, \mathbf{m}_1) = & \left(\frac{1}{2} - s\right) \text{tr}[B(s)\Lambda] + \frac{1}{2} \text{tr}[B(s)(\mathbf{S}_0 - \mathbf{S}_1)] \\ & - \frac{1}{2} s(1-s) \text{tr}[B(s)(\mathbf{S}_0 - \mathbf{S}_1)B(s)\Lambda] - \frac{1}{2} \log \frac{|E^T \mathbf{S}_0 E|}{|E^T \mathbf{S}_1 E|}. \end{aligned} \quad (12)$$

Then, taking derivative of $\nabla_s f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ with respect to s once again, we obtain

$$\begin{aligned} \nabla_s^2 f(E, s, \mathbf{m}_0, \mathbf{m}_1) = & -\text{tr}[B(s)\Lambda] + (2s-1) \text{tr}[B(s)(\mathbf{S}_0 - \mathbf{S}_1)B(s)\Lambda] \\ & + s(1-s) \text{tr}\left\{[B(s)(\mathbf{S}_0 - \mathbf{S}_1)]^2 B(s)\Lambda\right\} - \frac{1}{2} \text{tr}\left\{[B(s)(\mathbf{S}_0 - \mathbf{S}_1)]^2\right\} \\ \stackrel{(a)}{=} & -\text{tr}[B(s)\Lambda] + (s-1) \text{tr}[B(s)(\mathbf{S}_0 - \mathbf{S}_1)B(s)\Lambda] \\ & + s \text{tr}[B(s)\mathbf{S}_0 B(s)(\mathbf{S}_0 - \mathbf{S}_1)B(s)\Lambda] - \frac{1}{2} \text{tr}\left\{[B(s)(\mathbf{S}_0 - \mathbf{S}_1)]^2\right\} \\ \stackrel{(b)}{=} & -\frac{1}{1-s} \text{tr}[B(s)\mathbf{S}_0 B(s)\Lambda] + \frac{s}{1-s} \text{tr}[B(s)\mathbf{S}_0 B(s)\mathbf{S}_0 B(s)\Lambda] - \frac{1}{2} \text{tr}\left\{[B(s)(\mathbf{S}_0 - \mathbf{S}_1)]^2\right\} \\ \stackrel{(c)}{=} & -\frac{1}{1-s} \text{tr}\left\{\Lambda^{\frac{1}{2}} B(s) \mathbf{S}_0 E \Gamma(s) E^T \mathbf{S}_0 B(s) \Lambda^{\frac{1}{2}}\right\} - \frac{1}{2} \text{tr}\left\{[B(s)(\mathbf{S}_0 - \mathbf{S}_1)]^2\right\}, \end{aligned} \quad (13)$$

where $\Gamma(s) = (E^T \mathbf{S}_0 E)^{-1} - (E^T \mathbf{S}_0 E + \frac{1-s}{s} E^T \mathbf{S}_1 E)^{-1}$, equalities (a) and (b) both result from $B(s) = B(s)[s\mathbf{S}_0 + (1-s)\mathbf{S}_1]B(s)$ and thus $(1-s)B(s)(\mathbf{S}_0 - \mathbf{S}_1)B(s) = B(s)\mathbf{S}_0 B(s) - B(s)$, and equality (c) owing to the definition of $B(s)$ and the relationship $\text{tr}(AB) = \text{tr}(BA)$ for matrices A and B .

If $E^T \mathbf{m}_0 \neq E^T \mathbf{m}_1$, then we have $E^T \Lambda^{\frac{1}{2}} \neq \mathbf{0}$. Because of $E^T \mathbf{S}_0 E \succ \mathbf{0}$, it holds

$$E^T \mathbf{S}_0 B(s) \Lambda^{\frac{1}{2}} = E^T \mathbf{S}_0 E [A(s)]^{-1} E^T \Lambda^{\frac{1}{2}} \neq \mathbf{0}$$

with $A(s) = E^T(s\mathbf{S}_0 + (1-s)\mathbf{S}_1)E \succ \mathbf{0}$. Combined with $\Gamma(s) \succ \mathbf{0}$, $B(s) \succeq \mathbf{0}$, and $\text{tr}\{[B(s)(\mathbf{S}_0 - \mathbf{S}_1)]^2\} = \text{tr}\{B(s)[(\mathbf{S}_0 - \mathbf{S}_1)B(s)(\mathbf{S}_0 - \mathbf{S}_1)]\} \geq 0$, we conclude $\nabla_s^2 f(E, s, \mathbf{m}_0, \mathbf{m}_1) < 0$ from Equation (13).

On the other hand, when $E^T \mathbf{m}_0 = E^T \mathbf{m}_1$, then it follows from Equation (3) that $E^T \mathbf{S}_0 E \neq E^T \mathbf{S}_1 E$. If $[B(s)(\mathbf{S}_0 - \mathbf{S}_1)]^2 = \mathbf{0}$, Then we have $B(s)(\mathbf{S}_0 - \mathbf{S}_1)B(s) = \mathbf{0}$, that is

$$E[A(s)]^{-1}(E^T \mathbf{S}_0 E - E^T \mathbf{S}_1 E)[A(s)]^{-1} E^T = \mathbf{0}.$$

Then it immediately follows $E^T \mathbf{S}_0 E - E^T \mathbf{S}_1 E = \mathbf{0}$, which leads to a contradiction. Therefore, when $E^T \mathbf{m}_0 = E^T \mathbf{m}_1$, we deduce $\text{tr}\{[B(s)(\mathbf{S}_0 - \mathbf{S}_1)]^2\} > 0$, which means $\nabla_s^2 f(E, s, \mathbf{m}_0, \mathbf{m}_1) < 0$ by Equation (13).

As a consequence, $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ is a strictly concave function of s . $\square$

Moreover, the following Proposition 2 shows that $f_C(E, \mathbf{m}_0, \mathbf{m}_1)$ is continuously differentiable with respect to E for given $(\mathbf{m}_0, \mathbf{m}_1)$, which is also necessary for follow-up analysis.

Proposition 2. Given $(\mathbf{m}_0, \mathbf{m}_1)$, $f_C(E, \mathbf{m}_0, \mathbf{m}_1)$ defined by (5) is a continuously differentiable function of E .

Proof. For given E and $(\mathbf{m}_0, \mathbf{m}_1)$, denote $s^* := \arg \max_{s \in [0,1]} f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ with $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ given by Equation (6). Note that $f(E, 0, \mathbf{m}_0, \mathbf{m}_1) = f(E, 1, \mathbf{m}_0, \mathbf{m}_1) = 0$ and

$f_C(E, \mathbf{m}_0, \mathbf{m}_1) = \max_{s \in [0,1]} f(E, s, \mathbf{m}_0, \mathbf{m}_1) > 0$ under condition (3). Hence, $s^* \in (0, 1)$, which, combined with the strict concavity of $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ in Proposition 1, implies that s^* is unique with satisfying $\nabla_s f(E, s, \mathbf{m}_0, \mathbf{m}_1) = 0$.

Recalling the expression of $\nabla_s f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ in (12), since all the involved terms $B(s)$, $\text{tr}(\cdot)$, $|\cdot|$, $\log(\cdot)$ and $E^T S_0 E \neq 0$ are continuously differentiable functions of E , then $\nabla_s f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ as a composite function of the above functions is also continuously differentiable with respect to E . Similarly, $\nabla_s^2 f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ in (13) is a composition of functions $\frac{1}{1-s}$, $B(s)$ and $\text{tr}(\cdot)$, which are all continuous with respect to s . Hence, $\nabla_s^2 f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ is continuous with respect to s , that is, $\nabla_s f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ is continuously differentiable with respect to s . Due to $\nabla_s^2 f(E, s, \mathbf{m}_0, \mathbf{m}_1) \neq 0$, it follows from the implicit function theorem [23] that s^* is an implicit function of E . Hence, we rewrite s^* as

$$s^*(E) := \arg \max_{s \in [0,1]} f(E, s, \mathbf{m}_0, \mathbf{m}_1). \quad (14)$$

Furthermore, based on the implicit function theorem [23], $s^*(E)$ is a continuously differentiable function of E , which means that $\nabla s^*(E)$ is continuous with respect to E .

Because of $f_C(E, \mathbf{m}_0, \mathbf{m}_1) = \max_{s \in [0,1]} f(E, s, \mathbf{m}_0, \mathbf{m}_1)$, we have $f_C(E, \mathbf{m}_0, \mathbf{m}_1) = f(E, s^*(E), \mathbf{m}_0, \mathbf{m}_1)$ immediately. In addition, $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ is continuously differentiable with s and E . Combined with the fact that $s^*(E)$ is a continuously differentiable function of E , thereby $f_C(E, \mathbf{m}_0, \mathbf{m}_1)$ is differentiable with respect to E .

Moreover, by leveraging on the chain rule [24] to $f_C(E, \mathbf{m}_0, \mathbf{m}_1) = f(E, s^*(E), \mathbf{m}_0, \mathbf{m}_1)$, it holds

$$\nabla_E f_C(E, \mathbf{m}_0, \mathbf{m}_1) = [\nabla_E f(E, s, \mathbf{m}_0, \mathbf{m}_1) + \nabla_s f(E, s, \mathbf{m}_0, \mathbf{m}_1) \nabla s^*(E)]|_{s=s^*(E)},$$

where

$$\begin{aligned} \nabla_E f(E, s, \mathbf{m}_0, \mathbf{m}_1) &= s(1-s)[\mathbf{I} - \mathbf{W}(s)E\mathbf{A}^{-1}(s)E^T]\mathbf{\Lambda}E\mathbf{A}^{-1}(s) \\ &\quad - s\mathbf{S}_0E(E^T\mathbf{S}_0E)^{-1} - (1-s)\mathbf{S}_1E(E^T\mathbf{S}_1E)^{-1} + \mathbf{W}(s)E\mathbf{A}^{-1}(s) \end{aligned} \quad (15)$$

with $\mathbf{W}(s) = s\mathbf{S}_0 + (1-s)\mathbf{S}_1$, $\mathbf{A}(s) = E^T\mathbf{W}(s)E$, and $\mathbf{\Lambda} = (\mathbf{m}_1 - \mathbf{m}_0)(\mathbf{m}_1 - \mathbf{m}_0)^T$. Since $\nabla s^*(E)$, $\nabla_s f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ and $\nabla_E f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ are all continuous with respect to E , then $\nabla_E f_C(E, \mathbf{m}_0, \mathbf{m}_1)$ is also continuous. In conclusion, $f_C(E, \mathbf{m}_0, \mathbf{m}_1)$ is a continuously differentiable function of E . $\square$

4.1. Compute the Gradient in Problem (RP) by Danskin's Theorem

In the sequel, we will exploit Danskin's theorem in Appendix A to compute the gradient $\nabla \tilde{f}_C(E)$, where $\tilde{f}_C(E)$ is defined by Equation (9).

On basis of Proposition 2, the following results hold true for function $f_C(E, \mathbf{m}_0, \mathbf{m}_1)$ and set $\mathcal{E}(\bar{\mathbf{m}}_0, k_0\mathbf{S}_0^{-1}) \times \mathcal{E}(\bar{\mathbf{m}}_1, k_1\mathbf{S}_1^{-1})$:

1. $\mathcal{E}(\bar{\mathbf{m}}_0, k_0\mathbf{S}_0^{-1}) \times \mathcal{E}(\bar{\mathbf{m}}_1, k_1\mathbf{S}_1^{-1})$ is a compact set because it is a finite-dimensional bounded closed set. Meanwhile, $\forall (\mathbf{m}_0, \mathbf{m}_1)$, mapping $(t, \mathbf{m}_0, \mathbf{m}_1) \rightarrow f_C(E + t\mathbf{h}, \mathbf{m}_0, \mathbf{m}_1)$ is continuous at point $(0, \mathbf{m}_0, \mathbf{m}_1)$ due to the continuity of $f_C(E, \mathbf{m}_0, \mathbf{m}_1)$;
2. For arbitrary given $(\mathbf{m}_0, \mathbf{m}_1)$ and sufficiently small $t > 0$, since $f_C(E, \mathbf{m}_0, \mathbf{m}_1)$ is differentiable, there exists a bounded directional derivative

$$D_1 f_C(E + t\mathbf{h}, \mathbf{m}_0, \mathbf{m}_1; \mathbf{h}) = \lim_{\tau \rightarrow 0^+} \frac{1}{\tau} \times [f_C(E + (t + \tau)\mathbf{h}, \mathbf{m}_0, \mathbf{m}_1) - f_C(E + t\mathbf{h}, \mathbf{m}_0, \mathbf{m}_1)];$$

3. Mapping $(t, \mathbf{m}_0, \mathbf{m}_1) \rightarrow D_1 f_C(E + t\mathbf{h}, \mathbf{m}_0, \mathbf{m}_1)$ is continuous at point $(0, \mathbf{m}_0, \mathbf{m}_1)$, which is from the continuity of $\nabla_E f_C(E, \mathbf{m}_0, \mathbf{m}_1)$.

With identifications $E \sim u$, $(\mathbf{m}_0, \mathbf{m}_1) \sim v$, $\mathbb{R}^{m \times p} \sim U$, $\mathcal{E}(\bar{\mathbf{m}}_0, k_0\mathbf{S}_0^{-1}) \times \mathcal{E}(\bar{\mathbf{m}}_1, k_1\mathbf{S}_1^{-1}) \sim V$, $-f_C(E, \mathbf{m}_0, \mathbf{m}_1) \sim J(u, v)$, and $-\tilde{f}_C(E) \sim \bar{J}(u)$, all conditions of Danskin's theorem in Appendix A

are satisfied. Subsequently, for a given selection matrix E , if the optimal solution $(\hat{m}_0(E), \hat{m}_1(E))$ of problem (10) is unique, then the gradient of $\tilde{f}_C(E)$ exists. Furthermore, on basis of the Danskin's theorem, we have

$$\nabla \tilde{f}_C(E) = \nabla_E f_C(E, \mathbf{m}_0, \mathbf{m}_1)|_{(\mathbf{m}_0, \mathbf{m}_1) = (\hat{m}_0(E), \hat{m}_1(E))}. \quad (16)$$

Recall $f_C(E, \mathbf{m}_0, \mathbf{m}_1) = \max_{s \in [0,1]} f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ from Equation (5). For given $(\mathbf{m}_0, \mathbf{m}_1)$, we can again utilize the Danskin's theorem in Appendix A to obtain $\nabla_E f_C(E, \mathbf{m}_0, \mathbf{m}_1)$. Obviously, $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ is continuously differentiable with respect to E . Therefore, it is easy to verify that conditions 1)–3) of Danskin's theorem in Appendix A are all satisfied with identifications $E \sim u$, $s \sim v$, $\mathbb{R}^{m \times p} \sim U$, $[0, 1] \sim V$, $f(E, s, \mathbf{m}_0, \mathbf{m}_1) \sim J(u, v)$, and $f_C(E, \mathbf{m}_0, \mathbf{m}_1) \sim \bar{J}(u)$. Combined with the uniqueness of $s^*(E)$ defined by Equation (14), we can deduce

$$\nabla_E f_C(E, \mathbf{m}_0, \mathbf{m}_1) = \nabla_E f(E, s, \mathbf{m}_0, \mathbf{m}_1)|_{s=s^*(E)}. \quad (17)$$

On the other hand, due to the equivalence of Problems (7) and (8), for a given matrix E , a solution $(\hat{m}_0(E), \hat{m}_1(E))$ of Problem (10) corresponds to a solution $(\hat{s}(E), \hat{m}_0(E), \hat{m}_1(E))$ of problem

$$\text{Inner Problem (IP):} \quad \min_{(\mathbf{m}_0, \mathbf{m}_1) \in \mathcal{E}(\bar{\mathbf{m}}_0, k_0 S_0^{-1}) \times \mathcal{E}(\bar{\mathbf{m}}_1, k_1 S_1^{-1})} \max_{s \in [0,1]} f(E, s, \mathbf{m}_0, \mathbf{m}_1), \quad (\text{IP})$$

satisfying $f_C(E, \hat{m}_0(E), \hat{m}_1(E)) = f(E, \hat{s}(E), \hat{m}_0(E), \hat{m}_1(E))$ and

$$\hat{s}(E) = \arg \max_{s \in [0,1]} f(E, s, \hat{m}_0(E), \hat{m}_1(E)). \quad (18)$$

Combined with Equations (16) and (17), when the solution $(\hat{s}(E), \hat{m}_0(E), \hat{m}_1(E))$ is unique, we have

$$\nabla \tilde{f}_C(E) = \nabla_E f(s, E, \mathbf{m}_0, \mathbf{m}_1)|_{(s, \mathbf{m}_0, \mathbf{m}_1) = (\hat{s}(E), \hat{m}_0(E), \hat{m}_1(E))}, \quad (19)$$

where $\nabla_E f(s, E, \mathbf{m}_0, \mathbf{m}_1)$ is given by Equation (15) and $(\hat{s}(E), \hat{m}_0(E), \hat{m}_1(E))$ is the optimal solution of problem (IP).

4.2. Compute the Optimal Solution of Problem (IP)

Owing to the Sion's minimax theorem [25] and Lemma 5 in [11], for given E , we can exchange the orders of the minimization with respect to $(\mathbf{m}_0, \mathbf{m}_1)$ and the maximization with respect to s in problem (IP). Correspondingly, problem (IP) can be equivalently converted into

$$\max_{s \in [0,1]} \min_{(\mathbf{m}_0, \mathbf{m}_1) \in \mathcal{E}(\bar{\mathbf{m}}_0, k_0 S_0^{-1}) \times \mathcal{E}(\bar{\mathbf{m}}_1, k_1 S_1^{-1})} f(E, s, \mathbf{m}_0, \mathbf{m}_1). \quad (20)$$

Therefore, we will solve Problem (20) to attain the solution $(\hat{s}(E), \hat{m}_0(E), \hat{m}_1(E))$ of Problem (IP).

First, for given matrix E and parameter s , denote

$$(\mathbf{m}_0^*(s), \mathbf{m}_1^*(s)) := \arg \min_{(\mathbf{m}_0, \mathbf{m}_1) \in \mathcal{E}(\bar{\mathbf{m}}_0, k_0 S_0^{-1}) \times \mathcal{E}(\bar{\mathbf{m}}_1, k_1 S_1^{-1})} f(E, s, \mathbf{m}_0, \mathbf{m}_1).$$

Combining with the expression of $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ in Equation (6) and removing items irrelevant to $(\mathbf{m}_0, \mathbf{m}_1)$, then we can get $(\mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ by solving the following subproblem

$$\min_{(\mathbf{m}_0, \mathbf{m}_1) \in \mathcal{E}(\bar{\mathbf{m}}_0, k_0 S_0^{-1}) \times \mathcal{E}(\bar{\mathbf{m}}_1, k_1 S_1^{-1})} (\mathbf{m}_1 - \mathbf{m}_0)^T B (\mathbf{m}_1 - \mathbf{m}_0) \quad (21)$$

with $B := E(E^T(sS_0 + (1-s)S_1)E)^{-1}E^T \succeq 0$. Clearly, Problem (21) is convex, which can be directly solved by the CVX toolbox [26] with computation complexity $\mathcal{O}(p^3)$ [27]. Thus, once s is fixed, the corresponding $(\mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ follows. Particularly, $(\hat{\mathbf{m}}_0(E), \hat{\mathbf{m}}_1(E)) = (\mathbf{m}_0^*(\hat{s}(E)), \mathbf{m}_1^*(\hat{s}(E)))$.

Next, we will show how to determine the optimal $\hat{s}(E)$ given by Equation (18). Based on Proposition 1, $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ is concave with respect to s . Combined with the fact that $f(E, s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ is the minimum of a family of functions $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ over the uncertainty set $\mathcal{E}(\bar{\mathbf{m}}_0, k_0 S_0^{-1}) \times \mathcal{E}(\bar{\mathbf{m}}_1, k_1 S_1^{-1})$, we conclude that $f(E, s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ is also a concave function of s [28]. Therefore, $\nabla_s f(E, s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ is monotonically decreasing with respect to s . Subsequently, we apply the efficient bisection method to search $\hat{s}(E)$ such that $\nabla_s f(E, \hat{s}(E), \mathbf{m}_0^*(\hat{s}(E)), \mathbf{m}_1^*(\hat{s}(E))) = 0$.

For given E , we can once again use the Danskin's theorem in Appendix A to derive $\nabla_s f(E, s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$. Since $f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ is continuously differentiable with respect to s , all conditions of Danskin's theorem in Appendix A are met. Meanwhile, even if the optimal solution $(\mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ of Problem (21) is not unique, $\nabla_s f(E, s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ in Equation (12) is always the same no matter which $(\mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ is substituted. Therefore, $f(E, s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ is differentiable with respect to s , and we have

$$\nabla_s f(E, s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s)) = \nabla_s f(E, s, \mathbf{m}_0, \mathbf{m}_1)|_{(\mathbf{m}_0, \mathbf{m}_1) = (\mathbf{m}_0^*(s), \mathbf{m}_1^*(s))}, \quad (22)$$

where $\nabla_s f(E, s, \mathbf{m}_0, \mathbf{m}_1)$ is given by Equation (12), and $(\mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ is an arbitrary optimal solution of Problem (21).

After we get the optimal $\hat{s}(E)$, the corresponding $(\mathbf{m}_0^*(\hat{s}), \mathbf{m}_1^*(\hat{s}))$, i.e., $(\hat{\mathbf{m}}_0(E), \hat{\mathbf{m}}_1(E))$, is acquired by solving Problem (21). That is, the optimal solution $(\hat{s}(E), \hat{\mathbf{m}}_0(E), \hat{\mathbf{m}}_1(E))$ of Problem (IP) is obtained. We summarize the process of solving $(\hat{s}(E), \hat{\mathbf{m}}_0(E), \hat{\mathbf{m}}_1(E))$ as the following **Inner Procedure 1**.

Inner Procedure 1: Computing the Optimal Solution of Problem (IP)

Input: E , threshold η_0 , lower bound s_l and upper bound s_u of s , f , $n = 0$.

Output:

Let $s = \frac{s_l + s_u}{2}$, solve problem (21) to obtain $(\mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$.

Compute $\nabla_s f(E, s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ by Equation (22).

while $|\nabla_s f(E, s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s))| > \eta_0$ **do**

if $\nabla_s f(E, s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s)) > 0$ **then**

$s_l = s$;

else

$s_u = s$;

end

 Let $s = \frac{s_l + s_u}{2}$, compute a solution $(\mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ of problem (21);

 Compute $\nabla_s f(E, s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$ by (22). $n = n + 1$.

end

return $(\hat{s}(E), \hat{\mathbf{m}}_0(E), \hat{\mathbf{m}}_1(E)) = (s, \mathbf{m}_0^*(s), \mathbf{m}_1^*(s))$.

Remark 3. If the true distribution under each hypothesis is exactly known, i.e., there is no uncertainty in the mean vector, then we omit the process of computing solution $(\mathbf{m}_0^*(\hat{s}), \mathbf{m}_1^*(\hat{s}))$ of problem (21), and directly regard the true mean vector as $(\mathbf{m}_0^*(\hat{s}), \mathbf{m}_1^*(\hat{s}))$. Consequently, the robust sensor selection problem is reduced to one without uncertainty.

Based on the previous discussion, if the optimal solution $(\hat{s}(E), \hat{\mathbf{m}}_0(E), \hat{\mathbf{m}}_1(E))$ of Problem (IP) is unique, then the gradient $\nabla \tilde{f}_C(E)$ exists and can be computed by Equation (19). Next we will discuss the existence of the gradient $\nabla \tilde{f}_C(E)$ detailedly.

4.3. Existence of the Gradient in Problem (RP)

It has been shown that the uniqueness of the optimal solution $(\hat{s}(E), \hat{m}_0(E), \hat{m}_1(E))$ leads to the existence of the gradient $\nabla \tilde{f}_C(E)$. Therefore, in the following, we turn to show when $(\hat{s}(E), \hat{m}_0(E), \hat{m}_1(E))$ is unique. First the following lemma demonstrates the uniqueness of $\hat{s}(E)$.

Lemma 1. For given E , let $(\hat{s}(E), \hat{m}_0(E), \hat{m}_1(E))$ be the optimal solution of Problem (IP). Then $\hat{s}(E)$ is unique.

Proof. We will prove by contradiction. Assume that $(\tilde{s}(E), \tilde{m}_0(E), \tilde{m}_1(E))$ is also the optimal solution of problem (IP) with $\tilde{s}(E) \neq \hat{s}(E)$.

It is worth noting that, both $(\hat{s}(E), \hat{m}_0(E), \hat{m}_1(E))$ and $(\tilde{s}(E), \tilde{m}_0(E), \tilde{m}_1(E))$ are the saddle points of problem (IP). According to Proposition 1.4 in [29], the set of saddle points admits the Cartesian product form. Hence, $(\tilde{s}(E), \hat{m}_0(E), \hat{m}_1(E))$ is also a saddle point. That is, for given $(\hat{m}_0(E), \hat{m}_1(E))$, both $\hat{s}(E)$ and $\tilde{s}(E)$ are the optimal solutions of problem $\max_{s \in [0,1]} f(E, s, \hat{m}_0(E), \hat{m}_1(E))$, which contradicts with the strict concavity of $f(E, s, m_0, m_1)$ with respect to s . Consequently, $\hat{s}(E)$ is unique. $\square$

With the uniqueness of $\hat{s}(E)$, we additionally make the following assumption to guarantee the existence of $\nabla \tilde{f}_C(E)$.

Assumption 1. For given matrix E , the saddle point $(\hat{s}(E), m_0^*(\hat{s}(E)), m_1^*(\hat{s}(E)))$ of Problem (IP) is unique.

Remark 4. Because Lemma 1 has proven that $\hat{s}(E)$ is unique, if the solution $(m_0^*(\hat{s}(E)), m_1^*(\hat{s}(E)))$ of Problem (21) is unique, then Assumption 1 is satisfied. Under some conditions (e.g., the distance of the two uncertainty sets $\mathcal{E}(\tilde{m}_0, k_0 S_0^{-1})$ and $\mathcal{E}(\tilde{m}_1, k_1 S_1^{-1})$ is large), $(m_0^*(\hat{s}(E)), m_1^*(\hat{s}(E)))$ of Problem (21) is naturally unique. Therefore, Assumption 1 is not very restrictive.

Under Assumption 1, $(\hat{s}(E), \hat{m}_0(E), \hat{m}_1(E))$ is unique and achievable by Inner Procedure 1. Hence, $\nabla \tilde{f}_C(E)$ exists and can be obtained by Equation (19). Subsequently, Algorithm 1 can be executed. Furthermore, on basis of Theorem 1 in Section 3.1, we conclude that E^* obtained by the OCPGA in Algorithm 1 is a stationary point of problem (RP), which lays a foundation to attain a better solution of the original problem (PP).

Remark 5. If Assumption 1 is not satisfied, we could use the Clark generalized gradient [30] to replace the gradient in Algorithm 1. Based on the Danskin's theorem in Appendix A, the Clark generalized gradient can also be attained by Equation (19), where an arbitrary solution of Problem (21) is used. Then Algorithm 1 is still applicable and preserves the orthogonal constraint in each iteration. Although the resulting solution is not guaranteed to be a stationary point of Problem (RP), however, after projection and refinement phases, numerical simulations illustrate that the performance of the final result is also acceptable.

5. Numerical Simulations

In this section, numerical examples are carried out to show that the OCPGA-based method can obtain better solutions than the greedy algorithm-based method in [11]. To this end, (1) for fixed-size sensor selection problems, i.e., the total number and selected number of sensors are fixed, with randomly generated 50 (or 20) cases with different uncertainty sets, we exhibit the proportions that the OCPGA-based method performs better than, the same as, and worse than the greedy algorithm-based method; (2) for small-scale sensor selection problems, we compare the OCPGA-based method with the greedy algorithm based method and the exhaustive method; (3) for larger-scale sensor selection problems, the OCPGA-based method is compared with the greedy algorithm based method; (4) for specific small-scale and larger-scale sensor selection problems, the corresponding receiver operating characteristic (ROC) curves are depicted. Notably, in cases of (1)–(3), the performance of the method is measured by the resulting Chernoff distance (i.e., $\tilde{f}_C(E)$ in Problem (PP)), that is, the method with

larger Chernoff distance admits better performance. All the procedures are coded in MATLAB R2014b on an ASUS notebook with the Intel(R) Core(TM) i3-2310M CPU of 2.10GHz and memory of 6GB.

Assume that we need to choose p out of m sensors. In all simulations, for given (m, p) , the ellipsoidal uncertainty sets $\mathcal{E}(\bar{\mathbf{m}}_i, k_i \mathbf{S}_i^{-1})$ in Problem (IP), $i = 0, 1$, which contain the true distribution under each hypothesis, are generated as follows. Elements of the estimated mean vectors $\bar{\mathbf{m}}_0$ and $\bar{\mathbf{m}}_1$ are randomly generated from $(0, 1)$ and $(0, 2)$, respectively. The covariance matrix $\mathbf{S}_i$ is generated by $\mathbf{S}_i = \mathbf{P}_i \boldsymbol{\Sigma}_i \mathbf{P}_i^T$, where $\mathbf{P}_i$ is an orthogonal basis of $m \times m$ -dimensional matrices whose elements are generally drawn from $(0, 1)$, and $\boldsymbol{\Sigma}_i$ is a diagonal matrix with diagonal entries randomly generated in $(0, 2)$, $i = 0, 1$. The robustness parameters $k_0 = k_1 = 1$.

When (m, p) and the ellipsoidal uncertainty sets $\mathcal{E}(\bar{\mathbf{m}}_i, k_i \mathbf{S}_i^{-1})$ are given, we adopt **Inner Procedure 1** to compute $(\hat{s}(E), \hat{\mathbf{m}}_0(E), \hat{\mathbf{m}}_1(E))$, use Equation (19) to obtain the gradient $\nabla \tilde{f}_C(E)$, and then apply the OCPGA in Algorithm 1 to get the stationary point E^* of Problem (RP). After the projection and refinement phases described in Section 3.2, the final solution $\hat{E}$ of Problem (PP) is achieved. Meanwhile, the greedy algorithm-based method is a deterministic approach, that is, for given (m, p) and ellipsoidal uncertainty set, its outputs are all the same no matter how many times it is recalled. On the contrary, since the initial point of the OCPGA in Algorithm 1 is randomly chosen, the outputs of the OCPGA-based method may vary with initial points of the OCPGA. If time permits, the OCPGA-based method can be recalled for several times to achieve better performance. Moreover, the OCPGA and the greedy algorithm based methods both execute one refinement only.

Fixed-Size Examples: For 8 fixed pairs of small (m, p) , we give the proportions that the OCPGA-based method performs better than, the same as, and worse than the greedy algorithm-based method. For each (m, p) , by implementing the two methods with randomly generated 50 different ellipsoidal uncertainty sets (only 20 different ellipsoidal uncertainty sets for $(50, 5)$, $(80, 5)$, and $(100, 5)$ due to their long runtimes), the corresponding results are listed in Table 1. Here, with each ellipsoidal uncertainty set, the OCPGA-based method is recalled for two times and the better result is selected as the output. As shown in Table 1, for each (m, p) , the OCPGA-based method performs as well as the greedy algorithm-based method in most cases, while the “better” proportion is more than twice as many as the “worse” proportion. Actually, simulations show that, for “worse” cases, if we recall the OCPGA-based method for more times, then it can perform as well as even better than the greedy algorithm-based method.

Table 1. Performance Comparison Proportions.

(m,p)	Better	Same	Worse
(10,3)	20%	74%	6%
(10,4)	24%	72%	4%
(10,5)	20%	76%	4%
(12,3)	18%	74%	8%
(15,3)	22%	74%	4%
(50,5)	20%	70%	10%
(80,5)	15%	80%	5%
(100,5)	10%	85%	5%

Small-Scale Examples: we consider small-scale sensor selection problems, whose globally optimal solutions can be attained by the exhaustive method. Hence, we compare the optimal Chernoff distance obtained by the OCPGA-based method with those of the greedy algorithm-based method and the exhaustive method. Suppose that $p = 3, 4, 5$ out of $m = 10, 12, 15$ sensors are chosen. The corresponding outputs of the three methods are given in Table 2, and the corresponding runtimes of the three methods are listed in Table 3. It can be easily seen from Table 2 that, the Chernoff distances achieved by the OCPGA-based method are larger than those of the existing greedy algorithm-based method. For the case of $(15, 3)$, the Chernoff distance achieved by the OCPGA-based method is even 100% larger than that of the greedy algorithm-based method. In particular, our proposed

OCPGA-based method can attain the same performance as the exhaustive method. Meanwhile, it is shown in Table 3 that, the OCPGA-based method is more efficient than the greedy algorithm-based method, and both of them possess much shorter runtimes than the exhaustive method, which is coincident with the theoretical computation complexity analyses. Via simple computations, we can see from Table 3 that the runtime of the OCPGA-based method can be up to 48.72% shorter than that of greedy algorithm-based method (for the case of (10,4)). Therefore, our proposed OCPGA-based method admits better performance in terms of not only the objective value but also the runtime.

Table 2. Optimal Chernoff Distances of Three Methods.

m	$p = 3$			$p = 4$			$p = 5$		
	Exh	Gre	OCP	Exh	Gre	OCP	Exh	Gre	OCP
10	0.15	0.11	0.15	0.34	0.28	0.34	0.29	0.26	0.29
12	0.51	0.35	0.51	0.25	0.21	0.25	0.62	0.34	0.62
15	0.30	0.15	0.30	0.77	0.76	0.77	0.20	0.19	0.20

Exh: Exhaustive Method; Gre: Greedy Algorithm Based Method; OCP: OCPGA Based Method.

Table 3. Runtimes of Three Methods (s).

m	$p = 3$			$p = 4$			$p = 5$		
	Exh	Gre	OCP	Exh	Gre	OCP	Exh	Gre	OCP
10	2.59×10^3	52.12	51.07	1.10×10^4	204.77	105.01	6.49×10^4	87.07	80.39
12	4.17×10^3	128.75	109.35	2.66×10^4	82.41	75.94	2.04×10^5	106.53	98.31
15	4.33×10^4	92.68	89.28	1.50×10^5	193.45	188.95	1.20×10^6	153.09	150.41

Exh: Exhaustive Method; Gre: Greedy Algorithm Based Method; OCP: OCPGA Based Method.

Larger-Scale Examples: Now we consider larger-scale problems, where $m = 50, 80, 100$ and $p = 5, 10, 15$. Since m and p are large, which leads to the failure of the exhaustive method, we only compare the obtained Chernoff distances of our OCPGA-based method with those of the greedy algorithm-based method. The corresponding results are exhibited in Table 4, while the corresponding runtimes of the two methods are displayed in Table 5. As we can see from Table 4, the OCPGA-based method can attain a better solution than the greedy algorithm-based method. In the case of (50,5), the Chernoff distance obtained by the OCPGA-based method can be 13.14% larger than that of the greedy algorithm-based method. Moreover, Table 5 illustrates that, the OCPGA-based method admits higher efficiency than the greedy algorithm-based method, and the runtime of the OCPGA-based method can reduce up to 42.19% in the case of (50,5). Compared with small-scale cases in Table 3, we can see from Table 5 that the improvement in runtime is more obvious, which is due to the larger gap between m and p for larger-scale cases. Hence, for larger-scale cases, the OCPGA-based method also can achieve better solutions than the greedy algorithm-based method with shorter runtimes.

Table 4. Optimal Chernoff Distances of Two Methods.

m	$p = 5$		$p = 10$		$p = 15$	
	Gre	OCP	Gre	OCP	Gre	OCP
50	1.37	1.55	3.59	3.68	3.82	3.85
80	1.32	1.34	3.36	3.41	5.86	5.92
100	1.69	1.71	4.46	4.56	5.74	6.06

Gre: Greedy Algorithm Based Method; OCP: OCPGA Based Method.

Table 5. Runtimes of Two Methods (s).

m	$p = 5$		$p = 10$		$p = 15$	
	Gre	OCP	Gre	OCP	Gre	OCP
50	1499.76	866.96	2083.39	1956.96	3794.17	3635.68
80	1839.27	1718.73	5435.69	5251.96	5350.99	5240.56
100	2290.26	1752.71	5589.49	5273.85	11966.38	11433.33

Gre: Greedy Algorithm Based Method; OCP: OCPGA Based Method.

ROC Curves for Specific Examples: By Monte Carlo simulations with 200,000 instantiations of the LR tests and calculating P_D/P_F with $P_D := 1 - P_M$ being the detection probability, we depict the ROC curves to show the validity of the OCPGA-based method. It is well known that the higher the ROC curve, the better the detection performance. For the case of $(m, p) = (10, 3)$ in Table 2, we display the corresponding ROC curves of the exhaustive method, the greedy algorithm-based method and the OCPGA-based method. It can be seen from Figure 2 that the OCPGA-based method is superior to the greedy algorithm-based method, while it can attain the same performance as that of the exhaustive method. Similarly, for the case of $(m, p) = (50, 5)$ in Table 4, Figure 3 illustrates that our proposed approach performs better than the existing greedy algorithm-based method.

Figure 2. ROC curve for the small-scale case. **Figure 3.** ROC curve for the larger-scale case.

In summary, compared with the greedy algorithm-based method, the OCPGA-based method not only admits a lower theoretical computation complexity, but also can obtain better solutions with shorter runtimes in numerical simulations.

6. Conclusions

We address the minimax robust sensor selection in the binary Gaussian distribution hypothesis testing of WSN with the distribution mean vector under each hypothesis drifting in an ellipsoidal uncertainty set. Under a Bayesian optimal criterion, minimizing the maximum overall error probability with respect to the selection matrix is approximately converted to maximizing the minimum Chernoff distance between the distributions under a null hypothesis and alternative hypothesis to be detected. Then, the gradient of the objective function of the converted maximization problem is computed by Danskin's theorem. Furthermore, we apply the OCPGA to solve the relaxed maximization problem without 0/1 constraints, which can get a stationary point of the relaxed problem with lower computational complexity than the existing greedy algorithm. Numerical simulations demonstrate that the OCPGA-based method can attain better solutions than the greedy algorithm-based method with up to 48.72% shorter runtimes, and the OCPGA-based method is able to attain the globally optimal solution obtained by the exhaustive method for small-scale problems. In future work, we can consider cases where the distribution mean falls in other types of uncertainty sets such as the band model. In addition, the cases with not precisely known distribution covariance can also be a future research direction.

Author Contributions: Conceptualization and methodology, T.M., B.Q. and E.S.; software, T.M., D.N. and B.Q.; supervision, E.S.; writing—original draft, T.M.; writing—review and editing, E.S. and Q.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Sichuan Science and Technology Program under Grant 2019YJ0115, the National Natural Science Foundation of China under Grants 61473197, 11871357 and 61671411, and the National Key Research and Development Project under Grants 2018YFC0830303 and 2017YFE0119300.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

WSN	Wireless sensor network
KL	Kullback-Leibler
LR	Likelihood ratio
OCPGA	Orthogonal constraint-preserving gradient algorithm
PP	Primal Problem
RP	Relaxed Problem
IP	Inner Problem
ROC	Receiver operating characteristic

Appendix A. Danskin's Theorem (Theorem D1 and Its Corollary [15])

Let U and V be subsets of a Banach space $\mathcal{U}$ and a topological space $\mathcal{V}$ respectively. J is a mapping from $U \times V$ into $\mathbb{R}$. Assume $\bar{J}(u) = \sup_{v \in V} J(u, v)$, and $\hat{V}(u) = \{v \in V | J(u, v) = \bar{J}(u)\}$. Moreover, suppose the following conditions hold true

- (1) V is compact, $\forall v \in V$, the application $(t, v) \rightarrow J(u + th, v)$ is upper semi-continuous at $(0, v)$;
- (2) $\forall v \in V$ and $\forall t$ in a right neighborhood of 0, there exists a bounded directional derivative

$$D_1 J(u + th, v; h) = \lim_{\tau \rightarrow 0^+} \frac{1}{\tau} [J(u + (t + \tau)h, v) - J(u + th, v)];$$

- (3) the map $(t, v) \rightarrow D_1 J(u + th, v)$ is upper semi-continuous at $(0, v)$.

Then the function $\bar{J}$ has a directional derivative at u in the direction h , given by the formula

$$D\bar{J}(u; h) = \max_{v \in \hat{V}(u)} D_1 J(u, v; h).$$

Moreover, If $u \rightarrow J(u, v)$ has a Gateaux derivative J'_u , and if the max is unique: $\hat{V}(u) = \{\hat{v}\}$, then $\bar{J}$ has a Gateaux derivative $\bar{J}'(u)$ given by the simple formula

$$\bar{J}'(u) = J'_u(u, \hat{v}).$$

References

- Gerla, M.; Lee, E.K.; Pau, G.; Lee, U. Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds. In Proceedings of the IEEE World Forum on Internet of Things, Seoul, Korea, 6–8 March 2014; pp. 241–246.
- Bahrepor, M.; Meratnia, N.; Poel, M.; Taghikhaki, Z.; Havinga, P.J. Distributed event detection in wireless sensor networks for disaster management. In Proceedings of the IEEE International Conference on Intelligent Networking and Collaborative Systems, Thessaloniki, Greece, 24–26 November 2010; pp. 507–512.
- Rowaihy, H.; Eswaran, S.; Johnson, M.; Verma, D.; Bar-Noy, A.; Brown, T.; La Porta, T. A survey of sensor selection schemes in wireless sensor networks. In Proceedings of the Unattended Ground, Sea, and Air Sensor Technologies and Applications IX, Orlando, FL, USA, 11 May 2007; p. 65621A.
- Guy, C. Wireless sensor networks. In Proceedings of the Sixth International Symposium on Instrumentation and Control Technology: Signal Analysis, Measurement Theory, Photo-Electronic Technology, and Artificial Intelligence, Beijing, China, 24 October 2006; p. 63571I.
- Bhattacharya, D.; Krishnamoorthy, R. Power optimization in wireless sensor networks. *Int. J. Comput. Sci. Issues* **2011**, *8*, 415–419.
- Weimer, J.E.; Sinopoli, B.; Krogh, B.H. A relaxation approach to dynamic sensor selection in large-scale wireless networks. In Proceedings of the IEEE International Conference on Distributed Computing Systems Workshops, Beijing, China, 17–20 June 2008; pp. 501–506.
- Shen, X.; Varshney, P.K. Sensor selection based on generalized information gain for target tracking in large sensor networks. *IEEE Trans. Signal Process.* **2013**, *62*, 363–375. [\[CrossRef\]](#)
- Liu, L.; Wang, S.; Liu, D.; Zhang, Y.; Peng, Y. Entropy-based sensor selection for condition monitoring and prognostics of aircraft engine. *Microelectron. Reliab.* **2015**, *55*, 2092–2096. [\[CrossRef\]](#)
- Bajovic, D.; Sinopoli, B.; Xavier, J. Sensor selection for hypothesis testing in wireless sensor networks: A Kullback-Leibler based approach. In Proceedings of the 48th IEEE Conference on Decision Control, Shanghai, China, 15–18 December 2009; pp. 1659–1664.
- Bajovic, D.; Sinopoli, B.; Xavier, J. Robust linear dimensionality reduction for hypothesis testing with application to sensor selection. In Proceedings of the 47th Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, USA, 30 September–2 October 2009; pp. 363–370.
- Bajovic, D.; Sinopoli, B.; Xavier, J. Sensor selection for event detection in wireless sensor networks. *IEEE Signal Process. Mag.* **2011**, *59*, 4938–4953. [\[CrossRef\]](#)
- Chepuri, S.P.; Leus, G. Sparsity-promoting sensor selection for non-linear measurement models. *IEEE Signal Process. Mag.* **2015**, *63*, 684–698. [\[CrossRef\]](#)
- Bai, C.Z.; Katewa, V.; Gupta, V.; Huang, Y.F. A stochastic sensor selection scheme for sequential hypothesis testing with multiple sensors. *IEEE Trans. Signal Process.* **2015**, *63*, 3687–3699. [\[CrossRef\]](#)
- Wen, Z.; Yin, W. A feasible method for optimization with orthogonality constraints. *Math. Program.* **2013**, *142*, 397–434. [\[CrossRef\]](#)
- Bernhard, P.; Rapaport, A. On a theorem of Danskin with an application to a theorem of Von Neumann-Sion. *Nonlinear Anal.* **1995**, *24*, 1163–1181. [\[CrossRef\]](#)
- Scharf, L.L. *Statistical Signal Processing: Detection, Estimation and Time Series Analysis*; Addison-Wesley Publishing Company: New York, NY, USA, 1991.
- Cover, T.M.; Thomas, J.A. *Elements of Information Theory*; Wiley: New York, NY, USA, 1991.
- Wu, F.C.; Wang, Z.H.; Hu, Z.Y. Cayley transformation and numerical stability of Calibration equation. *Int. J. Comput. Vis.* **2009**, *82*, 156–184. [\[CrossRef\]](#)
- Goldfarb, D.; Wen, Z.; Yin, W. A curvilinear search method for p-harmonic flows on spheres. *SIAM J. Imaging Sci.* **2009**, *2*, 84–109. [\[CrossRef\]](#)
- Barzilai, J.; Borwein, J.M. Two-point step size gradient methods. *IMA J. Numer. Anal.* **1988**, *8*, 141–148. [\[CrossRef\]](#)

21. Zhang, H.C.; Hager, W.W. A nonmonotone line search technique and its application to unconstrained optimization. *SIAM J. Optim.* **2004**, *14*, 1043–1056. [[CrossRef](#)]
22. Gao, B.; Liu, X.; Chen, X.; Yuan, Y. A new first-order framework for orthogonal constrained optimization problems. *SIAM J. Optim.* **2018**, *28*, 302–332. [[CrossRef](#)]
23. Xia, D.X.; Shu, W.C.; Yan, S.Z.; Tong, Y.S. *Second Course for Functional Analysis*; Higher Education Press: Beijing, China, 1986.
24. Bonnans, J.F.; Shapiro, A. *Perturbation Analysis of Optimization Problems*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013.
25. Ricceri, B.; Simons, S. *Minimax Theory and Applications*; Kluwer Academic Pub: Dordrecht, The Netherlands, 1998.
26. Grant, M.; Boyd, S.; Ye, Y. *CVX Toolbox*; Stanford University Press: Stanford, CA, USA, 2009.
27. Nemirovskii, A.; Nesterov, Y. *Interior-Point Polynomial Algorithms in Convex Programming*; SIAM: Philadelphia, PA, USA, 1994.
28. Boyd, S.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press: Cambridge, UK, 2004.
29. Ekeland, I.; Temam, R. *Convex Analysis and Variational Problems*; SIAM: Philadelphia, PA, USA, 1999.
30. Clarke, F.H. Nonsmooth analysis and optimization. In *Proceedings of the International Congress of Mathematicians, Helsinki, Finland, 15–23 August 1978*; pp. 847–853.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).