

## RESEARCH ARTICLE

# A Fast Algorithm for Constructing Efficient Event-Related fMRI Designs

Ming-Hung Kao\* and Hans D. Mittelmann

*School of Mathematical and Statistical Sciences, Arizona State University, Tempe,  
Arizona 85287, USA*

*(November 2012)*

We propose a novel, efficient approach for obtaining high-quality experimental designs for event-related functional magnetic resonance imaging (ER-fMRI), a popular brain mapping technique. Our proposed approach combines a greedy hill-climbing algorithm and a cyclic permutation method. When searching for optimal ER-fMRI designs, the proposed approach focuses only on a promising restricted class of designs with equal frequency of occurrence across stimulus types. The computational time is significantly reduced. We demonstrate that our proposed approach is very efficient compared with a recently proposed genetic algorithm approach. We also apply our approach in obtaining designs that are robust against misspecification of error correlations.

**Keywords:** *A*-optimality; autoregressive process; cyclic permutation; *D*-optimality; genetic algorithms; hill-climbing technique; maximin designs

**AMS Subject Classification:** 62K05; 62J12

## 1. Introduction

We are concerned with experimental designs for event-related functional magnetic resonance imaging (ER-fMRI), one of the most dominating brain mapping technologies for studying brain activity in response to mental stimuli (e.g. pictures or sounds). An ER-fMRI design determines the presentation order and onset times of the mental stimuli that are to be presented in turn to a subject in an ER-fMRI experiment. It can be written as a finite sequence with elements  $0, 1, \dots, Q$ , where  $Q$  is the total number of stimulus types. For example, a design with two stimulus types may look like  $\xi = \{01120 \dots 2\}$ . The  $\ell$ th element in  $\xi$  corresponds to time  $(\ell - 1)ISI$ , where the *ISI* is a pre-specified inter-stimulus interval (e.g. 4 s); time 0 may be synchronized with the first valid MR scan. A positive integer  $q$  in  $\xi$  indicates an onset of the  $q$ th-type stimulus and a 0 means no stimulus onset at the corresponding time point. Each stimulus may last a short period of time (e.g. 1 s) and the control (e.g. periods of visual fixation or rest) appears during the time periods when there is no stimulus presentation.

A planned ER-fMRI design is presented to an experimental subject while an MR scanner repeatedly scans the subject's brain to acquire fMRI time series for making statistical inference about brain functions. The quality of the collected data

---

\*Corresponding author. Email: mkao3@asu.edu

depends on the selected design. A well-planned design helps to collect informative data to permit valid and precise conclusions. However, an imprudently selected design may hinder statistical inference, making it impossible to correctly answer the scientific questions of interest. Obtaining a good design to allow a successful ER-fMRI experiment is thus crucially important.

When selecting ER-fMRI designs, we need to consider not only statistical efficiencies, but also psychological constraints. Taking both considerations into account, Wager and Nichols [31] evaluated the quality of designs using a weighted sum of individual design criteria, each corresponding to a study objective of interest; the weights are selected to reflect the importance of the corresponding objective. To obtain a design optimizing the weighted sum design criterion, Wager and Nichols advocated the use of a genetic algorithm to search over the enormous space of all ER-fMRI designs. Recently, Kao et al. [16] proposed improvements to the approach of Wager and Nichols [31]. This improved genetic algorithm approach has been applied in some studies to address important ER-fMRI design issues [e.g. 6, 18, 19, 27, 28].

Genetic algorithms [12] are metaheuristic algorithms widely considered in various optimization problems. While powerful, genetic algorithms are often criticized for being computationally expensive, mainly due to a large degree of randomness involved [29]. When obtaining ER-fMRI designs, genetic algorithms also tend to require much CPU time. As reported in Kao et al. [16], a genetic algorithm may take several minutes to hours just to achieve one design. The required CPU time can be significantly increased with the size and complexity of the problem, and the search for optimal ER-fMRI designs can easily become infeasible. For example, Maus et al. [27] proposed a maximin approach for obtaining maximin designs that are robust against misspecified autocorrelation coefficient  $\rho$  of first-order autoregressive (AR1) noise. The obtained designs are of great practical value since the  $\rho$ -value is almost always uncertain at the design stage. However, to obtain such a maximin design, one must implement the genetic algorithm multiple (e.g. 153) times to generate multiple locally optimal ER-fMRI designs, each being optimal for a given  $\rho$ -value (see Subsection 4.2). A highly efficient search algorithm allowing to quickly obtain these ER-fMRI designs is called for.

In this article, we develop a novel, fast approach for searching for optimal ER-fMRI designs by combining the hill-climbing technique and the cyclic permutation procedure considered by Kao et al. [18]. Hill-climbing algorithms typically start with a potential solution, which is an ER-fMRI design in our context, and iteratively make incremental changes to the solution to seek a better solution. The process is repeated until no further improvements can be made. This technique is simple and is widely applied in some non-deterministic polynomial-time (NP) hard optimization problems such as the traveling salesman problem. Empirical studies also suggest that, for some cases, carefully designed hill-climbing algorithms can provide a better performance than the genetic algorithms; see also, Forrest and Mitchell [7], Juels and Wattenberg [14], and Baluja [2]. Here, we demonstrate that the hill-climbing algorithm that we develop can help to efficiently obtain good ER-fMRI designs. The idea of our approach is to operate on a 'short' ER-fMRI design, which is used to generate a full-length design by cyclically permuting the labels of the stimulus types  $(Q - 1)$  times. The obtained designs will thus have a (nearly) equal frequency of appearance across stimulus types. This means that our approach will focus only on a restricted class of designs having such a special pattern. We then propose a hill-climbing algorithm to iteratively replace the current short design by the 'best' short design in a small neighborhood. Here, a short design is said to be the best if, compared with other short designs, it yields a full-length design

that has the best value (design efficiency) of a specific design criterion. The search is terminated if no further improvements can be made.

With various case studies, we demonstrate that our newly proposed approach significantly outperforms the genetic algorithm of Kao et al. [16] in terms of computing time. Designs that we obtain attain similar design efficiencies as those found by the genetic algorithm. Our approach helps to save much computational resource without sacrificing the performance of the obtained designs. We also use the proposed approach to obtain ER-fMRI designs when the noise is modeled by a second-order autoregressive (AR2) process. This extension is of great practical relevance. In addition to AR1 process, the use of an AR2 process is not uncommon [e.g. 21], and can provide improved analysis results [20]. However, the previous approach for obtaining ER-fMRI designs is computationally very expensive, if not infeasible, for this important situation. Our newly proposed approach helps to efficiently obtain high-quality designs.

The remainder of the article is organized as follows. In Section 2, we provide background information. Our proposed approach is described in Section 3. We then present some case studies in Section 4 and compare our approach with the genetic algorithm of Kao et al. [16]. Finally, concluding remarks are provided in Section 5.

## 2. Background

An ER-fMRI design is a sequence of mental stimuli. It can be written as a sequence of finite numbers of length  $L$ ; e.g.  $\xi = \{01120 \dots 2\}$ . The  $\ell$ th element specifies the stimulus type to be presented at time  $(\ell - 1)(ISI)$  with a pre-specified  $ISI$ . Specifically, a positive integer  $q$  represents a  $q$ th-type stimulus and a 0 indicates no stimulus presentation at that time point. While being presented to an experimental subject, each stimulus appears briefly (e.g. 1 s). The control (e.g. visual fixation) is presented from the offset time of a stimulus to the onset time of the next stimulus.

While the subject is cognitively engaged with the stimuli in an ER-fMRI design, an MR scanner scans the subject's brain to collect fMRI time series. At an activated voxel, each stimulus evokes an influx of oxygenated blood and changes the ratio of the oxy- to deoxy-blood. The strength of the local magnetic field is perturbed, leading to an increase in the MR signal intensity. With no additional stimuli, the MR signal intensity falls back to baseline, typically followed by an undershoot before completely returning to baseline. This rise and fall of MR signals may take about 20-30 s, and is described by the hemodynamic response function (HRF). Under a commonly assumed linear time invariance system [e.g. 21], the HRFs evoked by the stimuli of the same  $q$ th type are assumed to be the same throughout the experiment, and will be denoted by  $h_q(\tau)$ , where  $\tau$  is the time elapsed since the stimulus onset;  $q = 1, \dots, Q$  (=total number of stimulus types). In addition, when the stimuli are close in time, the evoked HRFs overlap and the heights of overlapping HRFs are assumed to accumulate linearly. The accumulated HRF height at time  $t$  can be represented as [13]:

$$h(t) = \sum_{q=1}^Q \int_0^t x_q(\tau) h_q(t - \tau) d\tau,$$

where  $x_q(\tau)$  is the stimulus function of the  $q$ th-type stimuli. For ER-fMRI with brief stimuli,  $x_q(\tau) = 1$  if a  $q$ th-type stimulus onsets at time  $\tau$ , and  $x_q(\tau) = 0$  otherwise; see also Josephs et al. [13]. Consequently,  $h(t) = \sum_{q=1}^Q \sum_{\tau \in \mathcal{T}_q(t)} h_q(t - \tau)$ .

Here,  $\mathcal{T}_q(t) = \{\tau_{q,1}, \dots, \tau_{q,J_q(t)}\}$ ,  $\tau_{q,j}$  is the  $j$ th onset time of the  $q$ th-type stimulus, and  $J_q(t)$  is the number of  $q$ th-type stimuli that occur on or before time  $t$ . Making inferences about (some characteristics of)  $h_q(\cdot)$  is of great interest to neuroscientists; studying the HRF helps to understand the underlying brain activity due to the stimuli [21].

The fMRI time series,  $y(t)$ , collected from a voxel is commonly modeled as the sum of 1) the accumulated HRF  $h(t)$ , 2) a commonly observed drift/trend, and 3) correlated noise. With an MR scanning rate of  $TR$  (e.g. 2 s), a model for describing the fMRI time series can be written as:

$$\mathbf{y} = \sum_{q=1}^Q \mathbf{X}_q \mathbf{h}_q + \mathbf{S}\boldsymbol{\gamma} + \mathbf{e}. \quad (1)$$

Here,  $\mathbf{y}$  is the  $T \times 1$  vector with the  $j$ th element  $y_j = y((j-1)TR)$ . The vector  $\mathbf{h}_q = (h_{1,q}, \dots, h_{k,q})'$  is a discretization of  $h_q(\tau)$ . In particular,  $h_{j,q} = h_q((j-1)\Delta T)$ , where  $\Delta T$  is the discretization interval and is the greatest value making both  $(ISI/\Delta T)$  and  $(TR/\Delta T)$  integers. With this  $\Delta T$ , all the HRF heights that can possibly contribute to  $\mathbf{y}$  are included in  $\mathbf{h}_q$ ; see also Kao et al. [16]. The length  $k$  of  $\mathbf{h}_q$  is determined by to the duration of the HRF, counting from the stimulus onset to the HRF's complete return to baseline. Typically, the duration of an HRF evoked by a brief stimulus is no longer 32 s;  $k$  is thus set to  $1 + \lceil 32/\Delta T \rceil$ .  $\mathbf{X}_q$  in (1) is the  $T \times k$ , zero-one design matrix for the  $q$ th stimulus type. A construction of  $\mathbf{X}_q$  can be found in Kao et al. [19]. The vector  $\mathbf{S}\boldsymbol{\gamma}$  allows for a drift/trend of  $\mathbf{y}$ , where  $\mathbf{S}$  is a specified matrix and  $\boldsymbol{\gamma}$  is an unknown parameter vector. The vector  $\mathbf{e} = (e_1, \dots, e_T)'$  represents the correlated noise with mean  $\mathbf{0}$  and variance-covariance matrix  $\sigma^2\boldsymbol{\Sigma}$ .

By setting  $\mathbf{h} = (\mathbf{h}'_1, \dots, \mathbf{h}'_Q)'$  as a unknown parameter vector of interest, Model (1) is widely considered for estimating the HRFs. In general, the main focus is on estimating  $\mathbf{C}_h \mathbf{h}$ , where  $\mathbf{C}_h$  is a specified matrix of coefficients of linear combinations of interest. The design goal is to achieve a design yielding the most precise least squares estimate  $\mathbf{C}_h \hat{\mathbf{h}}$  of  $\mathbf{C}_h \mathbf{h}$ . To that end, we may consider to minimize the sum of the variances of the least squares estimates of the parametric functions of interest. This is to find a design maximizing  $\sigma^2 / \text{trace}[\text{cov}(\mathbf{C}_h \hat{\mathbf{h}})]$ , where  $\text{cov}(\mathbf{C}_h \hat{\mathbf{h}}) = \sigma^2 \mathbf{C}_h [\mathbf{X}'\mathbf{V}'(\mathbf{I}_T - \mathbf{P}_{VS})\mathbf{V}\mathbf{X}]^{-1} \mathbf{C}'_h$ ,  $\mathbf{X} = [\mathbf{X}_1, \dots, \mathbf{X}_Q]$ ,  $\mathbf{V}$  is such that  $\mathbf{V}\boldsymbol{\Sigma}\mathbf{V}' = \mathbf{I}_T$ ,  $\mathbf{I}_T$  is the  $T \times T$  identity matrix, and  $\mathbf{P}_A$  is the orthogonal projection onto the column space of  $\mathbf{A}$ . This trace criterion will be referred to as the  $A$ -optimality criterion hereinafter. We will also consider another popular design criterion, namely the  $D$ -optimality criterion, and aim at designs maximizing  $\sigma^{2r_h} / \det[\text{cov}(\mathbf{C}_h \hat{\mathbf{h}})]$ , where  $r_h$  is the number of rows of  $\mathbf{C}_h$ ; see also Kao et al. [16] and Maus et al. [27].

In addition to estimating the HRF, another study objective of interest to neuroscientists is to detect brain voxels that are activated by the stimuli. For this objective, the HRF is typically modeled as the product of a reference waveform and an unknown scaling parameter (or HRF amplitude); i.e.,  $h_q(\tau) = \theta_q h^*(\tau)$  with a specific  $h^*(\tau)$  and an unknown parameter  $\theta_q$ . The function  $h^*(\tau)$  is commonly set to the double-gamma function of the SPM software package (<http://www.fil.ion.ucl.ac.uk/spm/>), which is widely used for fMRI data analysis. Specifically,  $h^*(\tau) = g(\tau) / \max_s g(s)$ , where

$$g(\tau) = \frac{\tau^5 e^{-\tau}}{5!} - \frac{1}{6} \frac{\tau^{15} e^{-\tau}}{15!}.$$

A general linear model for detection problems can thus be written as:

$$\mathbf{y} = \sum_{q=1}^Q \mathbf{X}_q \mathbf{h}^* \theta_q + \mathbf{S}\boldsymbol{\gamma} + \boldsymbol{\varepsilon}, \quad (2)$$

where  $\mathbf{h}^* = (h_1^*, \dots, h_k^*)'$  with  $h_j^* = h^*((j-1)\Delta T)$ ,  $\theta_q$  is the unknown amplitude of the HRF of the  $q$ th stimulus type, and  $\boldsymbol{\varepsilon} = (\varepsilon_1, \dots, \varepsilon_T)'$  is the correlated noise. All the other terms in Model (2) are as in Model (1). For detection, the focus is normally on  $\mathbf{C}_\theta \boldsymbol{\theta}$  with a given linear combination coefficient matrix  $\mathbf{C}_\theta$ , and  $\boldsymbol{\theta} = (\theta_1, \dots, \theta_Q)'$ . The design goal for this study objective is to obtain a design optimizing some function of  $\text{cov}(\mathbf{C}_\theta \hat{\boldsymbol{\theta}}) = \sigma^2 \mathbf{C}_\theta [(\mathbf{I}_Q \otimes (\mathbf{h}^*)') \mathbf{X}' \mathbf{V}' (\mathbf{I}_T - \mathbf{P}_{\mathbf{V}\mathbf{S}}) \mathbf{V} \mathbf{X} (\mathbf{I}_Q \otimes \mathbf{h}^*)]^{-1} \mathbf{C}_\theta'$ ;  $\mathbf{C}_\theta \hat{\boldsymbol{\theta}}$  is the least squares estimate of  $\mathbf{C}_\theta \boldsymbol{\theta}$ , and  $\otimes$  is the Kronecker product. We will focus on obtaining  $A$ -optimal designs maximizing  $\sigma^2 / \text{trace}[\text{cov}(\mathbf{C}_\theta \hat{\boldsymbol{\theta}})]$  and  $D$ -optimal designs maximizing  $\sigma^{2r_\theta} / \det[\text{cov}(\mathbf{C}_\theta \hat{\boldsymbol{\theta}})]$ , where  $r_\theta$  is the number of rows of  $\mathbf{C}_\theta$ .

Selecting a design to efficiently achieve estimation and detection is important. However, some selected designs may easily incur psychological confounds such as anticipation and habituation to contaminate the data [5, 24]. To avoid such designs, we may consider to optimize the  $R$ th order counterbalancing criterion proposed by Wager and Nichols [31]. Designs optimizing this criterion make it hard for the subject to predict the next stimulus based on the previously presented stimuli. Here, we consider a modified version of the counterbalancing criterion provided by Kao et al. [16] to account for the finiteness of design length. Specifically, the modified  $R$ th order counterbalancing criterion is:

$$\sum_{r=1}^R \sum_{p=1}^Q \sum_{q=1}^Q [|n_{pq}^r - (L_0 - r)/Q^2|]. \quad (3)$$

Here,  $R$  is a specified integer and is typically set to 3. The values of  $n_{pq}^r$  and  $L_0$  are calculated from a subsequence generated by leaving out all the zeros in the design being evaluated. Specifically,  $n_{pq}^r$  is the observed number of occurrences in the subsequence that a  $q$  follows a  $p$  with  $r-1$  elements in between, and  $L_0$  is the length of the subsequence. A design is said to be  $R$ th order counterbalanced if its subsequence minimizes (3). We note that the criterion is defined for  $Q > 1$ .

For studies with one or more previously mentioned objectives, we may consider to obtain designs maximizing the following weighted sum criterion:

$$\begin{aligned} F^* &= w_c \left[ 1 - \frac{F_c}{\max(F_c)} \right] + w_d \left[ \frac{F_d}{\max(F_d)} \right] + w_e \left[ \frac{F_e}{\max(F_e)} \right] \\ &= w_c F_c^* + w_d F_d^* + w_e F_e^*. \end{aligned} \quad (4)$$

Here,  $F_c$  represents counterbalancing criterion in (3);  $F_d$  is the  $A$ - or  $D$ -optimality criterion for detection with Model (2);  $F_e$  is the  $A$ - or  $D$ -optimality criterion for estimation with Model (1);  $w_i$ 's are user-specified weights with  $w_i \geq 0$  and  $\sum_i w_i = 1$ . Following Kao et al. [16], the  $\max(F_c)$  is calculated as the  $F_c$ -value of the design  $\xi_1 = \{1 \cdots 1\}$ , which is the design involving only the first stimulus type;  $\xi_1$  can be replaced by  $\xi_q = \{q \cdots q\}$  for any  $q = 2, \dots, Q$  since we focus only on designs where all the stimulus types appear equally often. The maxima of  $F_d$  and  $F_e$  are generally unavailable, and are approximated by using an optimization algorithm (e.g., the genetic algorithm of Kao et al. [16] or our proposed approach described

in the next section) to maximize  $F_d$  and  $F_e$ , respectively.

### 3. A Novel Approach

We propose a novel, fast approach for searching for optimal ER-fMRI designs that optimize a specific objective function,  $F^\dagger$ , which can be  $F_d$ ,  $F_e$  or  $F^*$  in (4). Our proposed approach combines the cyclic permutation method used by Kao et al. [18] and a greedy hill-climbing algorithm. The cyclic permutation method was originally proposed to obtain multiple sequences of stimuli for multiple MR scanning sessions; each sequence is for one scanning session. Here, we adapt the method for finding one sequence of stimuli for one, single MR scanning session. Our use of the cyclic permutation method here is motivated by the observation that designs achieving the maximal design efficiency  $F^\dagger$  tend to have (nearly) equal stimulus frequency across stimulus types. For simplified models, this design property is proved by Liu and Frank [23]. The empirical results of Kao et al. [17] and Kao et al. [16] also suggest that the property may remain true for the more realistic models presented in the previous section. The cyclic permutation method uses a ‘short design’ of length  $\lceil L/Q \rceil$  to generate a design of full length  $L$ ;  $\lceil a \rceil$  is the smallest integer  $\geq a$ . Starting with such a short design, additional  $Q - 1$  short designs are sequentially constructed by cyclically permuting the labels of stimulus types. Specifically, the label  $q$  in the current short design becomes  $q + 1$  in the next short design,  $q = 1, \dots, Q - 1$ , the label  $Q$  is replaced by 1, and 0’s are kept intact. A full-length design is then generated by concatenating the  $Q$  short designs and trimming off the last  $(Q\lceil L/Q \rceil - L)$  elements. The obtained designs have (nearly) equal frequency across stimulus types. The cyclic permutation method allows us to focus only on this promising class of designs when searching for optimal ER-fMRI designs.

We would like a short design that yields a full-length design maximizing the objective function  $F^\dagger$ . Targeting such a short design, we propose a greedy hill-climbing approach which gradually increases the achieved  $F^\dagger$ -value by iteratively updating a fraction of an initial short design. At the first iteration, we create some ‘neighbors’ by perturbing the first  $b$  elements of the initial short design. If the  $F^\dagger$ -value is improved by the best neighbor, we replace the current short design by this best neighbor; otherwise, the current short design is kept. We then move to the next  $b$  elements and repeat the process until all the  $\lceil L/Q \rceil$  elements are considered once. This finishes the first ‘run’. Another ‘run’ is then started by returning to the first  $b$  elements of the short design achieved in the previous run. This hill-climbing procedure is repeated and terminated if there is no improvement in the last run. We note that, at the last iteration of each run, the number of elements to be perturbed might be  $\leq b$ . Our method for perturbing the selected elements is described below.

To generate neighbors of a short design, we make use of the  $(2^b - 1)$  non-zero  $b$ -element binary sequences. We add (subtract) each of these sequences to (from) the  $b$  elements of the short design being considered at the current iteration; other elements are kept intact. If a resulting element is  $Q + 1$ , it is replaced by 0. We also substitute  $-1$  by  $Q$  so that the obtained neighboring short design is feasible. With these perturbations, we will have  $2^b - 1$  neighboring short designs for  $Q = 1$  since addition and subtraction lead to identical short designs in this case. For  $Q > 1$ , a total of  $2(2^b - 1)$  neighbors will be generated at each iteration .

We now present in details the steps of our proposed approach:

**Step 0.** Specify an initial short design  $\pi^{(0)}$  of length  $\lceil L/Q \rceil$ . Use the cyclic permutation procedure described previously to obtain the full-length design

of length  $L$ . Calculate the  $F^\dagger$ -value of the obtained full-length design. Set  $F_{\max}^\dagger$  to the obtained  $F^\dagger$ -value, and  $t = 0$ .

**Step 1.** For  $\pi^{(t)} = \{\pi_1, \dots, \pi_{\lceil L/Q \rceil}\}$ , create its neighbor by replacing  $(\pi_{tb+1}, \dots, \pi_{(t+1)b})$  with  $(\text{mod}(\pi_{tb+1} + \zeta_1, Q + 1), \text{mod}(\pi_{tb+2} + \zeta_2, Q + 1), \dots, \text{mod}(\pi_{(t+1)b} + \zeta_b, Q + 1))$ , where  $\text{mod}(a_1, a_2)$  is  $a_1$  modulo  $a_2$ , and  $\zeta_i$  is the  $i$ th element of a non-zero binary sequence of length  $b$ ;  $i = 1, \dots, b$ . Use all the  $2^b - 1$  non-zero binary sequences to generate  $2^b - 1$  neighbors. If  $Q > 1$ , obtain additional  $2^b - 1$  neighbors by replacing  $(\pi_{tb+1}, \dots, \pi_{(t+1)b})$  with  $(\text{mod}(\pi_{tb+1} - \zeta_1, Q + 1), \text{mod}(\pi_{tb+2} - \zeta_2, Q + 1), \dots, \text{mod}(\pi_{(t+1)b} - \zeta_b, Q + 1))$ .

**Step 2.** Similarly to Step 0, obtain the  $F^\dagger$ -values for the neighboring short designs generated in Step 1. Find the the best short design,  $\pi_{nbr}$ , and its  $F^\dagger$ -value,  $F_{nbr}^\dagger$ . If  $F_{nbr}^\dagger > F_{\max}^\dagger$ , then set  $F_{\max}^\dagger = F_{nbr}^\dagger$  and  $\pi^{(t+1)} = \pi_{nbr}$ ; otherwise,  $\pi^{(t+1)} = \pi^{(t)}$ . Set  $t = t + 1$ .

**Step 3.** Repeat Steps 1 and 2 until  $t = \lfloor (\lceil L/Q \rceil / b) \rfloor + 1$ . When  $t = \lfloor (\lceil L/Q \rceil / b) \rfloor + 1$  and  $l = \text{mod}(\lceil L/Q \rceil, b) > 0$ , repeat Steps 1 and 2 once more to but with  $b$  replaced by  $l$  in Step 1 to perturb the last  $l$  elements.

**Step 4.** Terminate the search if the  $F^\dagger$ -value is not improved in the current run. Otherwise, set  $\pi^{(0)}$  to  $\pi^{(t)}$  (i.e., the most updated short design), and then reset  $t$  to 0 and repeat Steps 1 to 3 for another run.

In our case studies, we start the search with a sequence of zeros as the initial short design. The integer  $b$  in Step 1 determines the number of neighboring short designs or, equivalently, the size of the neighborhood. With a large  $b$ , a large number of neighbors can be considered, but the computational burden is also increased. In our experience, setting  $b$  to around 4 tends to yield good designs. We also note that, unlike genetic algorithms which often involve many algorithmic parameters, our proposed algorithm only has one algorithmic parameter  $b$  to be determined by users.

In the next section, we consider various case studies and compare the performance of the proposed hill-climbing approach with that of the genetic algorithm of Kao et al. [16]. For these case studies, we set  $b = 4$ . As for the genetic algorithm, the algorithmic parameters are set to their default values as in Table 1 of Kao [15], except for the stopping rule. Here, we considered an efficient stopping rule, which is also built in the program of Kao [15]; see the Appendix for details.

## 4. Case Studies

In this section, we apply our proposed approach to obtain single- and multi-objective ER-fMRI designs. We then follow the maximin procedure of Maus et al. [27] to achieve designs for cases where the autocorrelation between observations is uncertain.

### 4.1. Single- and multi-objective ER-fMRI designs

In the first set of case studies, we consider various  $(Q, L)$  combinations, including (1, 255), (2, 242), (3, 255), (4, 624), and (6, 342), and set the  $ISI$  to 4 s and  $TR$  to 2 s. Following Kao et al. [16], the reference waveform  $\mathbf{h}^*$  of model (2) is set to the double-gamma function of SPM, normalized to have a maximum of 1. The fMRI time series is assumed to have a second-order Legendre polynomial drift. The errors follow an AR1 process and  $\Sigma^{-1}$  is a tri-diagonal matrix with  $((\Sigma^{-1}))_{1,1} = ((\Sigma^{-1}))_{T,T} = 1$ ,  $((\Sigma^{-1}))_{i,i} = 1 + \rho^2$ ,  $i = 2, \dots, T - 1$ ,  $((\Sigma^{-1}))_{i,j} = -\rho$  for  $|i - j| = 1$

and  $((\Sigma^{-1}))_{i,j} = 0$  for  $|i - j| > 1$ . The whitening matrix  $\mathbf{V}$  can be obtained from  $\Sigma^{-1}$  by, e.g., the Cholesky decomposition. Following previous studies, the autocorrelation coefficient is set to  $\rho = 0.3$ . We will relax this assumption in the next subsection. The computations for this first set of case studies are conducted on a desktop computer with a 3.0 GHz Intel Pentium 4 quad-core processor.

Tables 1 and 2 provide comparisons between our proposed hill-climbing algorithm and the genetic algorithm of Kao et al. [16] in optimizing a class of objective functions, including  $F_d$ ,  $F_e$ , and  $F_{de}^*(w) = wF_d^* + (1-w)F_e^*$  for  $w = 0.2, 0.4, 0.5, 0.6$ , and  $0.8$ . As described under (4),  $F_d$  is the design efficiency for detection, and  $F_e$  is used to evaluate the worth of designs for estimating the HRF.  $F_{de}^*$  is defined as  $F^*$  in (4) with  $(w_c, w_d, w_e) = (0, w, 1 - w)$ . Both  $A$ -optimality and  $D$ -optimality are considered. We separately study cases where the individual stimulus effects are of interest, and cases where pairwise comparisons are of interest. When individual stimulus effects are of interest, the focus of estimation is on estimating  $\mathbf{h} = (\mathbf{h}'_1, \dots, \mathbf{h}'_Q)'$  of Model (1), and that of detection is on studying  $\boldsymbol{\theta} = (\theta_1, \dots, \theta_Q)'$  of Model (2). For pairwise comparisons with  $A$ -optimality, the linear combination coefficient matrix  $\mathbf{C}_\theta$  for  $\boldsymbol{\theta}$  contains the coefficients of all the  $Q(Q-1)/2$  comparisons, namely  $(\theta_i - \theta_j)$ ,  $1 \leq i < j \leq Q$ . For pairwise comparisons with  $D$ -optimality, we set  $\mathbf{C}_\theta$  to a  $(Q-1)$ -by- $Q$  matrix with the  $i$ th row corresponding to the comparison between the first and the  $(i+1)$ st stimulus types (i.e.,  $\theta_1 - \theta_{i+1}$ ). In particular, when  $Q = 3$ , we have

$$\mathbf{C}_\theta = \begin{pmatrix} 1 & -1 & 0 \\ 1 & 0 & -1 \\ 0 & 1 & -1 \end{pmatrix} \text{ for } A\text{-optimality; and } \mathbf{C}_\theta = \begin{pmatrix} 1 & -1 & 0 \\ 1 & 0 & -1 \end{pmatrix} \text{ for } D\text{-optimality.}$$

In addition, the linear combination coefficient matrix  $\mathbf{C}_h$  for  $\mathbf{h}$  is  $\mathbf{C}_\theta \otimes \mathbf{I}_k$ ;  $k = 1 + \lfloor 32/\Delta T \rfloor$ .

[[ Table 1 is about here ]]

Table 1 presents the CPU time spent of our proposed approach relative to the genetic algorithm. For most cases, the CPU time used by our approach is less than half of that of the genetic algorithm. In many of these cases, the relative CPU time is no more than 10%. Our newly proposed algorithm is very efficient. In addition, the reduced CPU usage does not hinder the quality of the obtained ER-fMRI designs. As presented in Table 2, the designs obtained by the two approaches yield similar efficiencies.

[[ Table 2 is about here ]]

We further consider cases where avoiding psychological confounds is also of interest and regard  $F^* = (F_c^* + F_d^* + F_e^*)/3$  as the objective function. Figure 1 presents the CPU time spent (A1 & B1) and the  $\max(F^*)$  (A2 & B2) achieved by the two approaches for these cases. In that figure, (A1) and (A2) are for individual stimulus effects and (B1) and (B2) are for pairwise comparisons. The results presented in Figure 1 convey the same information as those in Tables 1 and 2. We again observe that our proposed hill-climbing approach requires much less CPU time than the genetic algorithm and achieves highly efficient designs. In particular, our approach saved about 76%-98% of the CPU time for the cases considered in Figure 1.

[[Figure 1 is about here ]]

We note that, for each case with two or more objectives, we need to obtain three designs. Specifically, a design maximizing  $F_d$  and a design maximizing  $F_e$ . These two designs allow us to calculate the value of the multi-objective design criterion ( $F_{de}^*$  or  $F^*$ ) of each candidate design. We then obtain the third design, which is a multi-objective design maximizing the specified multi-objective design criterion. In



Table 1 and Figure 1, the CPU time spent for obtaining a multi-objective design is thus calculated as the total time for obtaining the three designs. Although both algorithms are used to obtain  $\max\text{-}F_d$  and  $\max\text{-}F_e$  designs for calculating the CPU times, we choose to use the maxima achieved by the genetic algorithm for evaluating the value of the multi-objective design criterion. This is to provide a fair comparison of the achieved designs, and it ensures that the two search algorithms aim at the same objective functions. The  $\max(F_d)$  and  $\max(F_e)$  obtained by our proposed algorithm can also be considered.

#### 4.2. Maximin robust ER-fMRI designs for AR1 noise

As described in Maus et al. [27], the autocorrelation coefficient  $\rho$  of AR1 noise of the fMRI time series may be uncertain, especially at the design stage. Consequently, obtaining an optimal design with a fixed  $\rho$ -value may be unsatisfactory. It is thus desired to achieve a design performing relative well across all possible values of  $\rho$ . For this purpose, we follow Maus et al. [27] to aim at designs maximizing the following maximin criterion:

$$F_{Mm}^*(\xi) = \min_{\rho \in [0,0.5]} RE(\xi | \xi_\rho^*) = \min_{\rho \in [0,0.5]} \frac{F(\xi; \rho)}{F(\xi_\rho^*; \rho)}. \quad (5)$$

Here,  $F(\cdot; \rho)$  can be  $F_d$  for detection,  $F_e$  for estimation, or  $F^*$  (or  $F_{de}^*$ ) for a weighted sum criterion. We note that the criteria  $F_d$ ,  $F_e$ ,  $F^*$  and  $F_{de}^*$  are defined as in the previous sections, and their values depend on  $\rho$ . In addition, we use  $\xi_\rho^*$  to represent a design maximizing the specified  $F(\cdot; \rho)$  with a given  $\rho$ .

In Maus et al. [27], the minimum  $RE(\xi | \xi_\rho^*)$  is obtained over a grid on the range  $[0, 0.5]$  of  $\rho$ . Specifically, they increased the value  $\rho$  from 0 to 0.5 in steps of 0.01 and utilized the genetic algorithm of Kao et al. [16] to generate a ‘locally optimal design’  $\xi_\rho^*$  for each of the 51  $\rho$ -values. With these 51  $\xi_\rho^*$  designs, the value of  $F_{Mm}^*$  can be calculated for any given design  $\xi$ . Maus et al. [27] then evaluated the  $F_{Mm}^*$ -values of the 51  $\xi_\rho^*$  designs and select the one having the greatest  $F_{Mm}^*$ -value. The selected design is identified as the ‘maximin design’.

Here, we compare the performance of our proposed approach with that of the genetic algorithm in implementing this maximin procedure of Maus et al. [27]. Following Maus et al. [27], we set the *ISI* and *TR* to 2 s. Other settings are as in the previous subsection. We obtain maximin designs for: (I) detection with individual stimulus effects; (II) detection with pairwise comparisons; (III) estimation with individual stimulus effects; and (IV) estimation with pairwise comparisons. We also consider bi-objective criteria and assign equal weights to detection and estimation with (V) individual stimulus effects, and (VI) pairwise comparisons; see Table 3. This set of case studies is performed on a desktop computer with a 3.4 GHz Intel Core i7-2600 processor. We note that, Maus et al. [27] only studied cases (I), (II), (III) and (V) with  $(Q, L) = (3, 255)$ .

##### 4.2.1. Cases (I) to (IV)

[[Table 3 is about here]]

For cases (I) to (IV), the focuses are on single objective functions. In particular,  $F(\cdot; \rho)$  in (5) is either  $F_d$  or  $F_e$ . For each case, we follow Maus et al. [27] to apply the genetic algorithm to obtain a set,  $\Xi_{GA}$ , of 51  $\xi_{\rho, GA}^*$  designs that maximize  $F(\cdot; \rho)$  for  $\rho = 0, 0.01, \dots, 0.5$ . The obtained design set is used to provide an approximation of  $F_{Mm}^*$ , denoted by  $F_{Mm, GA}^*$ . With respect to  $F_{Mm, GA}^*$ , a maximin design  $\xi_{GA}^*$  is then selected from the design set  $\Xi_{GA}$ . Following this same procedure, we also use the

proposed hill-climbing algorithm to obtain another design set  $\Xi_{HC} = \{\xi_{\rho,HC}^*, \rho = 0, 0.01, \dots, 0.5\}$ . Another approximation,  $F_{Mm,HC}^*$ , of  $F_{Mm}^*$  is achieved based on  $\Xi_{HC}$ . With this  $F_{Mm,HC}^*$ , a maximin design  $\xi_{HC}^*$  is obtained from  $\Xi_{HC}$ . The CPU time spent on obtaining the design set  $\Xi_{HC}^*$  and that for  $\Xi_{GA}^*$  can be found in Table 3. As presented in the table, the hill-climbing algorithm uses much less CPU time than the genetic algorithm in implementing the maximin procedure of Maus et al. [27]. For some cases, the genetic algorithm needs more than one or two hours while our approach requires only a few minutes.

We also compare the achieved maximin designs  $\xi_{GA}^*$  and  $\xi_{HC}^*$ . A direct comparison between  $F_{Mm,GA}^*(\xi_{GA}^*)$  and  $F_{Mm,HC}^*(\xi_{HC}^*)$  is not reasonable since  $\Xi_{GA}$  and  $\Xi_{HC}$  may differ. To provide a fair comparison, we combine  $\Xi_{GA}$  and  $\Xi_{HC}$  to provide another approximation of  $F_{Mm}^*$ . Specifically, for each  $\rho$ , we select the better design between  $\xi_{\rho,GA}^*$  and  $\xi_{\rho,HC}^*$  that yields a higher  $F(\cdot; \rho)$ -value. The selected 51 designs are then used to provide better approximations of  $F_{Mm}^*(\xi_{GA}^*)$  and  $F_{Mm}^*(\xi_{HC}^*)$ . These two  $F_{Mm}^*$ -values are reported in Table 3. As presented, the maximin efficiencies achieved by the two algorithms do not differ much. The maximin designs obtained by the two approaches are quite efficient for  $\rho \in [0, 0.5]$ .

#### 4.2.2. Cases (V) and (VI)

Cases (V) and (VI) are more complicated than the previous cases since two objectives (detection and estimation) are simultaneously of interest. The  $F(\cdot; \rho)$  in (5) is  $F_{de}^*(0.5) = 0.5F_d^* + 0.5F_e^*$ . For each of the 51  $\rho$ -values, we need a design maximizing  $F_d$  and a design maximizing  $F_e$ ; these two designs allow us to calculate  $F_d^*$  and  $F_e^*$ , respectively, and thus  $F(\cdot; \rho) = F_{de}^*$ ;  $\rho = 0, 0.01, \dots, 0.5$ . To evaluate the  $F_{Mm}^*$  of (5), we need additional 51 locally optimal designs  $\xi_\rho^*$ , each being optimal for  $F(\cdot; \rho)$ . This calls for a total of 153 designs. We then follow Maus et al. [27] to select the design maximizing  $F_{Mm}^*$  from the 51  $\xi_\rho^*$  designs. Without an efficient approach for obtaining each of the required designs, this maximin procedure is computationally very expensive.

We compare the performance of our proposed hill-climbing algorithm with the genetic algorithm in obtaining these 153 designs and the maximin design. As presented in Table 3, our approach consistently requires much less CPU time than the genetic algorithm in achieving a maximin design. To compare the performance of the achieved maximin designs, we evaluate the  $F_{Mm}^*$ -value by using the 153 designs obtained by the genetic algorithm to approximate  $\max(F_d)$ ,  $\max(F_e)$  and  $\max(F_{de}^*)$  for each  $\rho$ -value. This ensures that the designs obtained by different algorithms are compared under the same criterion. The results reported in Table 3 show that the  $F_{Mm}^*$ -values achieved by the two algorithms are very close to 1. This indicates that both algorithms can obtain good maximin designs.

#### 4.2.3. The performance of the $\xi_{0.3}^*$ design

[[Table 4 is about here ]]

We also evaluate the performance of the locally optimal design  $\xi_{0.3,HC}^*$ , which is the best design obtained by the hill-climbing algorithm when  $\rho$  is fixed at 0.3. Table 4 presents the  $F_{Mm}^*$ -value of  $\xi_{0.3,HC}^*$  relative to the maximin design  $\xi_{HC}^*$ . For many cases, the  $\xi_{0.3,HC}^*$  design performs reasonably well even when the true value of  $\rho$  is not 0.3. Nevertheless, a design with an improved  $F_{Mm}^*$ -value can (almost always) be efficiently obtained by combining our proposed algorithm with the maximin approach of Maus et al. [27]. We also report in Table 4 the worst relative value of  $F_{Mm}^*$  obtained among the 51 locally optimal designs. The result suggests that not all the locally optimal designs perform as well as  $\xi_{0.3,HC}^*$  or  $\xi_{HC}^*$ ; an imprudently selected design can result in a great efficiency loss.

### 4.3. Maximin robust ER-fMRI designs for AR2 noise

In addition to AR1 process, modeling the correlated noise using AR2 process is not uncommon [e.g. 21] and is recommended by Lenoski et al. [20] based on empirical results. As demonstrated in the previous subsection, the genetic algorithm is computationally very expensive in obtaining a maximin design for AR1 noise. This approach becomes infeasible when the more complicated AR2 process is considered. Our newly proposed algorithm greatly reduces computational burden, making it an efficient approach for obtaining designs for this important situation. With an AR2 process, the inverse of the variance-covariance matrix is proportional to  $\Sigma^{-1}$  having  $((\Sigma^{-1}))_{1,1} = ((\Sigma^{-1}))_{T,T} = 1$ ;

$$((\Sigma^{-1}))_{i,i} = \begin{cases} 1 + \phi_1^2, & i = 2, T - 1; \\ 1 + \phi_1^2 + \phi_2^2, & 2 < i < T - 1; \end{cases}$$

$$((\Sigma^{-1}))_{i,j} = \begin{cases} -\phi_1, & |i - j| = 1 \text{ and } i + j = 3, 2T - 1; \\ -\phi_1(1 - \phi_2), & |i - j| = 1 \text{ and } 3 < i + j < 2T - 1; \\ -\phi_2, & |i - j| = 2; \\ 0, & |i - j| > 2. \end{cases}$$

Here, the range of the unknown parameter  $\phi_2$  is set to  $[0, 0.5]$ , and that of the unknown parameter  $\phi_1$  is  $[0, 0.5(1 - \phi_2)]$  when  $\phi_2 \in [0, 1/3]$ , and is  $[0, \sqrt{0.5(1 - 2\phi_2)(1 - \phi_2)}]$  when  $\phi_2 \in (1/3, 1/2]$ . This irregular parameter space allows for a stationary process with  $\rho_1 = \phi_1/(1 - \phi_2)$  and  $\rho_2 = \phi_2^2/(1 - \phi_2) + \phi_2$  ranging between 0 and 0.5. Here,  $\rho_i$  is the  $i$ th-order autocorrelation coefficient of an AR2 process, which describes the correlation between  $e_t$  and  $e_{t+i}$  in Model (1) and that of  $\varepsilon_t$  and  $\varepsilon_{t+i}$  in Model (2);  $t = 1, \dots, T - i$ ,  $i = 1, 2$ . We note that the method that we consider here can also be applied to other parameter spaces.

We apply the proposed hill-climbing approach to obtain a locally optimal design  $\xi_{\phi, HC}^*$  for each of the 1798 values of  $\phi = (\phi_1, \phi_2)$  over a grid on the parameter space described previously; the grid size is set to  $0.01 \times 0.01$ . The obtained designs are used to evaluate the following maximin criterion:

$$F_{Mm2}^*(\xi) = \min_{\phi \in \Omega} RE(\xi | \xi_{\phi}^*) = \min_{\phi \in \Omega} \frac{F(\xi; \phi)}{F(\xi_{\phi}^*; \phi)}. \quad (6)$$

Here,  $F$  represents the design criterion for the study objective(s) of interest;  $\Omega$  is the specified parameter space of  $\phi$ ; and  $\xi_{\phi}^*$  is a locally optimal design for a given  $\phi$ . We follow Maus et al. [27] to search over the obtained locally optimal designs for a design maximizing  $F_{Mm2}^*$ . Instead of evaluating the  $F_{Mm2}^*$ -values of all the 1798 locally optimal designs, we consider a much more efficient two-stage procedure to approximate the result; a similar procedure has also been considered in other studies [30]. At the first stage, a design maximizing  $F_{Mm2}^*$  is selected among a subclass of locally optimal designs, each being optimal for a  $\phi$ -value over a coarse grid ( $0.1 \times 0.1$ ) on  $\Omega$ . We then use the original grid size ( $0.01 \times 0.01$ ) to define a neighborhood of the selected design and search over the neighboring locally optimal designs for a better maximin design. For example, when searching for a maximin design for detection with  $Q = 1$  (Case I with  $Q = 1$  in Table 5), a locally optimal design  $\xi_{\phi^*, HC}^*$  for  $\phi^* = (\phi_1^*, \phi_2^*) = (0.4, 0.1)$  is selected at the first stage. The neighboring designs that we consider at the second stage are those  $\xi_{\phi, HC}^*$  designs, where the two coordinates of  $\phi$  can be rounded to the corresponding coordinates of  $\phi^*$ .

[[Table 5 is about here]]

In Table 5, we present  $F_{Mm2}^*$ -values of the designs obtain by using the two-stage approach; the  $\phi$ -values used to achieve these designs are also reported. From the table, designs with a maximin efficiency of at least 95% can be achieved when study objective is on estimation or on both estimation and detection. For cases where detecting brain activation is the only study objective, we obtained designs with about 85-92% maximin efficiency. For detection, the locally optimal designs are (near) block designs. Compared with the other cases reported in Table 5, these (nearly) patterned designs tend to suffer a slightly higher efficiency loss when the  $\phi$ -value of AR2 noise is misspecified. For comparison purposes, we also report in Table 6 the  $F_{Mm2}^*$ -values of the  $\xi_{0.3,HC}^*$  designs obtained in the previous subsection for AR1 noise. While the  $\xi_{0.3,HC}^*$  designs perform quite well under AR1 noise, they are rather vulnerable to a misspecified  $\phi$ -value of AR2 noise, especially for cases where detection is the only study objective.

[[Table 6 is about here]]

## 5. Discussion

High-quality experimental designs are important for ER-fMRI studies, and are in great demand. To help efficiently achieve such designs, we propose a novel approach that combines a cyclic permutation method and a greedy hill-climbing algorithm. We demonstrate that, in terms of computational time, our approach can significantly outperform the genetic algorithm of Kao et al. [16] that has been demonstrated to work better than other methods hitherto published. For many cases, our approach only requires about 10% of the CPU time used by the genetic algorithm to achieve highly efficient ER-fMRI designs.

In addition, we use our proposed approach to implement the maximin procedure of Maus et al. [27] for obtaining designs that allow uncertain autocorrelation coefficient  $\rho$  of the AR1 noise. As argued by Maus et al. [27], obtaining such a maximin robust design is important since the  $\rho$ -value is almost always unknown at the design stage and can vary across brain voxels. As the same design is to be applied on all voxels, the selected design should accommodate all the  $\rho$ -values that can possibly occur. However, finding a maximin robust design can be very time consuming. An efficient approach is thus crucially important. In our case studies, we show that our proposed approach provides a very efficient tool for achieving such designs.

Moreover, we apply our proposed approach to obtain maximin robust designs under AR2 noise. While the AR2 process is commonly considered at the analysis stage for modeling the noise, previous studies on ER-fMRI designs only focused on the AR1 process. This lack is partly due to the fact that the genetic algorithm is computationally very expensive. The proposed hill-climbing approach helps to efficiently obtain good designs for this important situation.

Our proposed approach takes advantage of the knowledge that optimal ER-fMRI designs tend to have equal stimulus frequency across stimulus types [e.g. 23]. We thus focus only on a restricted class of designs in which the  $Q$  stimulus types appear nearly equally often. Our results provide compelling evidence that highly efficient ER-fMRI designs can be found from this restricted design class by applying the hill-climbing technique. A drawback of the hill-climbing technique is that the algorithm might be trapped at a local maximum. Similar to some other optimization problems [e.g. 25], this potential drawback does not seem to hinder our search for efficient ER-fMRI designs. Nevertheless, one may consider to combine our proposed algorithm with a probabilistic method, such as the simulated annealing [e.g. 3], to avoid local

optima. Further research in this direction is needed.

Another advantage of our hill-climbing approach is in the small number of algorithmic parameters. When using the genetic algorithm, users need to decide the size of population, mutation rate, number of immigrants, the stopping rule and the algorithmic parameters involved with the selected stopping rule; see the Appendix for details. On the other hand, our proposed algorithm only has one algorithmic parameter, namely the number of perturbed elements  $b$  for determining the size of the neighborhood to be explored. We recommend to set  $b$  to a divisor of  $\lfloor 16/ISI \rfloor$  that is around 4; this value is linked to the block size (around 16 s) of the commonly suggested 16s-on-16s-off block designs for detection [11]. In our experience, when  $b$  is large, the approach tends to spend much CPU time without improving much in the achieved design efficiencies. For a small  $b$ , the approach may easily be trapped in a local optimum. Moreover, when implementing the hill-climbing algorithm, we used a sequence of zeros as the initial short design. Our results with this initial design seem satisfactory. Nevertheless, one may consider different initial designs to generate several solutions and then select the best one. This procedure is well suited to a parallel computing environment, which is becoming increasingly available.

In our case studies, we consider both  $A$ - and  $D$ -optimality criteria. We observe that, for some cases, the genetic algorithm requires much more time on searching for an  $A$ -optimal design than for a  $D$ -optimal design. We believe that the observation is linked to the quality of the initial designs used in the genetic algorithm. For cases where the interest is in the individual stimulus effects, the initial designs are favorable for  $D$ -optimality. Specifically, these initial designs have the same stimulus frequency of  $1/(Q+1)$  across all the  $Q$  stimulus types. This stimulus frequency is the optimal frequency reported in Maus et al. [27] for  $D$ -optimality with individual stimulus effects. For  $A$ -optimality, Liu and Frank [23] indicated that the optimal stimulus frequency is  $1/(Q+\sqrt{Q})$ .

We also note that our results are mainly for experiments with one, single scanning session. In some experiments, multiple short scanning sessions are implemented in replace of a long scanning session. For multiple scanning sessions, Kao et al. [18] studied and compared several methods for obtaining good ER-fMRI designs. A future research of interest is to extend our proposed hill-climbing algorithm to provide an efficient alternative approach for obtaining ER-fMRI designs for multiple scanning sessions.

## Appendix A. A genetic algorithm

The genetic algorithm of Kao et al. [16] is provided here. This algorithm makes use of some well known ER-fMRI designs. These designs are block designs,  $m$ -sequence, random designs, and mixed designs. In a block design, stimuli of the same type are clustered in blocks. For example, they can be formed by repetitions of  $\{111122220000\}$  or repetitions of  $\{11112222\}$ . Such designs typically yield high  $F_d$ -values and are commonly recommended for detection. The  $m$ -sequences are known to have high  $F_e$ -values and are advocated by Buračas and Boynton [4] for estimation problems. Such a design can be generated from primitive polynomials for a Galois field, and only exists when  $Q+1$  is a prime or prime power [9, 26]. Random designs are randomly generated and are unlikely to have a perceivable pattern. A mixed design is formed by combining part of a block design with part of a random design or an  $m$ -sequence. Mixed designs can attain good compromises between the two competing objectives, namely detection and estimation; see Liu [22] for further details. The genetic algorithm takes advantage of these well known

designs to find good designs. This algorithm is described below.

Step 1. Generate  $2G$  initial designs, including block designs,  $m$ -sequences (if exist), random designs, and mixed designs. Obtain the fitness, or the value of the objective function, of each initial design. These designs form the first generation.

Step 2. With probability proportional to fitness, select with replacement  $G$  pairs of designs to generate  $G$  pairs of offsprings via crossover and mutation. Specifically, the crossover operator randomly selects a cut-point, and exchanges the corresponding subsequences following the cut-point in the paired designs. The mutation operator perturbs a randomly selected portion  $\alpha_m$  of elements of all offspring designs. Obtain the fitness of the resulting designs.

Step 3. Add to the population another  $\lfloor 2G\alpha_i \rfloor$  immigrants drawn from random and block designs, and their combinations. Obtain their fitness.

Step 4. Create a pool to include the designs of the current generation, offspring designs and immigrants. According to their fitness, keep the best  $2G$  designs in the pool to form the next generation, and discard the others.

Step 5. Repeat steps 2 through 4 until a stopping rule is met. Keep track of the best design over generations.

In our case studies, we follow Kao et al. [16] to set  $G = 10$ ,  $\alpha_m = 1\%$  and  $\alpha_i = 20\%$ . In Kao et al. [16], the search is terminated after 10,000 generations. Here, we consider a more efficient stopping rule presented in Kao [15]. The search is then terminated if there is no significant improvement. Specifically, the improvement in the value of the objective function is calculated every 200 generations. If the improvement in the last 200 generations is no more than  $10^{-7}$  of that of the first 200 generations, the search is stopped.

In addition, when calculating the multi-objective criteria  $F_{de}^*$  and  $F^*$ , we need to approximate  $\max(F_d)$  and  $\max(F_e)$ . This is done by using the algorithm to search for a design maximizing  $F_d$ , and a design maximizing  $F_e$ . Denote the former design by  $\xi_d^*$  and the latter one by  $\xi_e^*$ . The  $\max(F_d)$  and  $\max(F_e)$  are approximated by  $F_d(\xi_d^*)$  and  $F_e(\xi_e^*)$ , respectively.

## References

- [1] Atkinson, A. C., Donev, A. N., and Tobias, R. D. (2007), *Optimum Experimental Designs, with SAS*, Great Britain: Oxford University Press.
- [2] Baluja, S. (1997), "Genetic Algorithms and Explicit Search Statistics," in *Advances in Neural Information Processing Systems*, eds. Jordan, M. I., Mozer, M. C., and Petsche, T., Cambridge, Mass.: M. I. T. Press, vol. 9, pp. 319–325.
- [3] Bertsimas, D. and Tsitsiklis, J. (1993), "Simulated Annealing," *Statistical Science*, 8, 10–15.
- [4] Buračas, G. T. and Boynton, G. M. (2002), "Efficient Design of Event-Related fMRI Experiments Using M-Sequences," *NeuroImage*, 16, 801–813.
- [5] Dale, A. M. (1999), "Optimal Experimental Design for Event-Related fMRI," *Human Brain Mapping*, 8, 109–114.
- [6] Delzell, D. A. P., Gunst, R. F., Schucany, W. R., Carmack, P. S., Lin, Q., Spence, J. S., and Haley, R. W. (2012), "Key Properties of D-Optimal Designs for Event-Related Functional MRI Experiments with Application to Nonlinear Models," *Statistics in Medicine*, to appear.
- [7] Forrest, S. and Mitchell, M. (1993), "Relative Building-Block Fitness and the Building-Block Hypothesis," in *Foundations of Genetic Algorithms*, ed. Whitley, D., San Mateo CA: Morgan Kaufmann, vol. 2, pp. 109–126.
- [8] Friston, K. J., Zarahn, E., Josephs, O., Henson, R. N. A., and Dale, A. M. (1999), "Stochastic Designs in Event-Related fMRI," *NeuroImage*, 10, 607–619.
- [9] Godfrey, K. (1993), *Perturbation Signals for System Identification*, New York: Prentice Hall.
- [10] Hedayat, A. S., Jacroux, M., and Majumdar, D. (1988), "Optimal Designs for Comparing Test Treatments with Controls," *Statistical Science*, 3, 462–476.
- [11] Henson, R. N. A. (2007), "Efficient Experimental Design for fMRI," in *Statistical Parametric Mapping: The Analysis of Functional Brain Images*, eds. Friston, K. J., Ashburner, J. T., Kiebel, S. J., Nichols, T. E., and D., P. W., London: Academic, 1st ed., pp. 193–210.

- [12] Holland, J. H. (1975), *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, Ann Arbor: University of Michigan Press.
- [13] Josephs, O., Turner, R., and Friston, K. (1997), "Event-Related fMRI," *Human Brain Mapping*, 5, 243–248.
- [14] Juels, A. and Wattenberg, M. (1995), "Stochastic Hill-climbing as a Baseline Method for Evaluating Genetic Algorithms," in *Advances in Neural Information Processing Systems*, eds. Hasselmo, M. E., Mozer, M. C., and Touretzky, D. S., Cambridge, Mass.: MIT Press, vol. 8, pp. 430–436.
- [15] Kao, M. H. (2009), "Multi-Objective Optimal Experimental Designs for ER-fMRI Using Matlab," *Journal of Statistical Software*, 30, 1–13.
- [16] Kao, M.-H., Mandal, A., Lazar, N., and Stufken, J. (2009), "Multi-Objective Optimal Experimental Designs for Event-Related fMRI Studies," *NeuroImage*, 44, 849–856.
- [17] Kao, M.-H., Mandal, A., and Stufken, J. (2008), "Optimal Design for Event-Related Functional Magnetic Resonance Imaging Considering Both Individual Stimulus Effects and Pairwise Contrasts," *Statistics and Applications*, 6, 225–241.
- [18] Kao, M. H., Mandal, A., and Stufken, J. (2009), "Efficient Designs for Event-Related Functional Magnetic Resonance Imaging with Multiple Scanning Sessions," *Communications in Statistics-Theory and Methods*, 38, 3170–3182.
- [19] Kao, M.-H., Mandal, A., and Stufken, J. (2012), "Constrained Multi-Objective Designs for Functional MRI Experiments Via a Modified Nondominated Sorting Genetic Algorithm," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 61, 515–534.
- [20] Lenoski, B., Baxter, L. C., Karam, L. J., Maisog, J., and Debbins, J. (2008), "On the Performance of Autocorrelation Estimation Algorithms for fMRI Analysis," *IEEE Journal of Selected Topics in Signal Processing*, 2, 828–838.
- [21] Lindquist, M. A. (2008), "The Statistical Analysis of fMRI Data," *Statistical Science*, 23, 439–464.
- [22] Liu, T. T. (2004), "Efficiency, Power, and Entropy in Event-Related fMRI with Multiple Trial Types: Part II: Design of Experiments," *NeuroImage*, 21, 401–413.
- [23] Liu, T. T. and Frank, L. R. (2004), "Efficiency, Power, and Entropy in Event-Related fMRI with Multiple Trial Types: Part I: Theory," *NeuroImage*, 21, 387–400.
- [24] Liu, T. T., Frank, L. R., Wong, E. C., and Buxton, R. B. (2001), "Detection Power, Estimation Efficiency, and Predictability in Event-Related fMRI," *NeuroImage*, 13, 759–773.
- [25] MacFarlane, A., Secker, A., May, P., and Timmis, J. (2010), "An Experimental Comparison of a Genetic Algorithm and a Hill-Climber for Term Selection," *Journal of Documentation*, 66, 513–531.
- [26] MacWilliams, F. J. and Sloane, N. J. A. (1977), *The Theory of Error Correcting Codes*, Amsterdam: Elsevier/North-Holland.
- [27] Maus, B., van Breukelen, G. J. P., Goebel, R., and Berger, M. P. F. (2010), "Robustness of Optimal Design of fMRI Experiments with Application of a Genetic Algorithm," *Neuroimage*, 49, 2433–2443.
- [28] Maus, B., van Breukelen, G. J. P., Goebel, R., and Berger, M. P. F. (2012), "Optimal Design for Nonlinear Estimation of the Hemodynamic Response Function," *Human Brain Mapping*, 33, 1253–1267.
- [29] Sivanandam, S. N. and Deepa, S. N. (2007), *Introduction to Genetic Algorithms*, Berlin ; New York: Springer.
- [30] Stufken, J. and Yang, M. (2012), "On Locally Optimal Designs for Generalized Linear Models with Group Effects," *Statistica Sinica*, 22, 1765–1786.
- [31] Wager, T. D. and Nichols, T. E. (2003), "Optimization of Experimental Design in fMRI: A General Framework Using a Genetic Algorithm," *NeuroImage*, 18, 293–309.
- [32] Worsley, K. J. and Friston, K. J. (1995), "Analysis of fMRI Time-Series Revisited—Again," *NeuroImage*, 2, 173–181.

Table 1. Relative (%) CPU time spent of the proposed hill-climbing approach to the genetic algorithm on maximizing  $F_d$ ,  $F_e$ , or  $F_{de}^*(w) = wF_d^* + (1-w)F_e^*$ .

$(Q, L)$	$F_e$	$F_{de}^*(0.2)$	$F_{de}^*(0.4)$	$F_{de}^*(0.5)$	$F_{de}^*(0.6)$	$F_{de}^*(0.8)$	$F_d$
<i>A-optimality: individual stimulus effects</i>							
(1, 255)	41.4	52.4	49.6	58.9	43.8	47.6	47.8
(2, 242)	22.0	19.8	29.4	29.4	33.1	27.2	35.2
(3, 255)	4.3	7.8	5.4	6.4	9.1	8.2	17.6
(4, 624)	7.6	10.8	10.4	11.5	9.5	10.5	11.3
(6, 342)	5.1	4.3	5.9	5.9	3.4	2.8	10.0
<i>D-optimality: individual stimulus effects</i>							
(1, 255)	19.6	31.9	31.9	43.0	42.1	39.8	66.9
(2, 242)	14.6	24.1	25.1	27.7	27.8	21.3	95.5
(3, 255)	23.4	24.7	18.1	16.7	18.0	17.3	20.3
(4, 624)	9.5	13.0	12.9	14.4	9.3	8.0	26.5
(6, 342)	7.6	5.2	3.9	3.1	2.3	3.7	6.5
<i>A-optimality: pairwise comparisons</i>							
(2, 242)	22.6	17.9	18.0	18.0	20.2	26.6	12.1
(3, 255)	19.3	12.0	13.7	10.1	13.1	11.1	11.7
(4, 624)	12.7	11.1	15.9	13.1	15.8	9.9	49.6
(6, 342)	5.0	5.2	5.6	4.8	4.5	4.6	14.1
<i>D-optimality: pairwise comparisons</i>							
(2, 242)	23.1	27.0	22.0	17.7	21.9	18.5	9.5
(3, 255)	19.7	22.4	12.1	22.4	21.0	12.6	53.2
(4, 624)	17.3	20.7	13.4	13.9	20.5	9.2	98.4
(6, 342)	2.4	3.0	2.6	2.5	2.9	1.9	24.3



Table 2. Relative value (%) of the  $\max(F_d)$ ,  $\max(F_e)$ , or  $\max(F_{de}^*)$  achieved by the proposed hill-climbing approach to that achieved by the genetic algorithm.

$(Q, L)$	$F_e$	$F_{de}^*(0.2)$	$F_{de}^*(0.4)$	$F_{de}^*(0.5)$	$F_{de}^*(0.6)$	$F_{de}^*(0.8)$	$F_d$
<i>A-optimality: individual stimulus effects</i>							
(1, 255)	100.1	100.1	100.1	100.1	100.0	100.0	99.7
(2, 242)	100.1	100.0	99.9	99.9	100.1	100.1	100.0
(3, 255)	100.1	99.9	99.8	100.0	100.0	100.1	100.5
(4, 624)	100.1	100.1	100.0	100.0	100.0	100.3	100.5
(6, 342)	99.8	99.5	100.0	100.0	99.9	100.1	102.1
<i>D-optimality: individual stimulus effects</i>							
(1, 255)	100.0	100.0	100.0	100.0	100.1	100.3	99.8
(2, 242)	99.9	100.1	100.0	100.0	100.1	99.9	98.7
(3, 255)	100.0	99.8	99.8	99.8	99.8	99.9	98.5
(4, 624)	100.0	100.0	100.1	100.1	100.1	100.1	98.9
(6, 342)	99.8	99.6	99.7	99.8	99.7	99.5	98.6
<i>A-optimality: pairwise comparisons</i>							
(2, 242)	100.6	99.8	99.8	100.0	99.7	99.7	100.0
(3, 255)	99.4	99.6	99.8	99.3	99.3	99.3	102.1
(4, 624)	100.0	99.7	99.7	99.7	99.6	99.5	98.6
(6, 342)	98.6	98.2	98.4	98.3	97.9	98.7	102.4
<i>D-optimality: pairwise comparisons</i>							
(2, 242)	99.8	99.8	99.9	99.9	99.6	99.7	100.2
(3, 255)	99.4	99.7	99.5	99.4	99.2	99.0	103.1
(4, 624)	99.9	99.7	99.7	99.7	99.6	99.4	98.5
(6, 342)	98.9	98.9	98.8	98.6	98.8	97.9	101.6

Table 3. Total CPU time spent and the achieved  $F_{Mm}^*$ -value of the proposed hill-climbing algorithm and the genetic algorithm.

$Q =$	A-optimality					D-optimality				
	1	2	3	4	6	1	2	3	4	6
(I) Detection: individual stimulus										
CPU time (minutes)										
hillclimber	2.1	2.6	1.6	7.9	1.6	2.2	2.7	1.5	6.8	1.6
genetic alg.	4.0	8.0	14.8	82.1	54.4	4.1	8.6	14.6	19.4	29.2
$F_{Mm2}^*$ -value (%)										
hillclimber	97.0	96.9	97.4	98.1	97.4	97.0	96.8	96.7	96.9	96.5
genetic alg.	94.9	95.4	97.0	95.4	96.7	95.5	96.2	98.0	98.1	97.5
(II) Detection: pairwise comparison										
CPU time (minutes)										
hillclimber	n/a	1.2	1.3	7.0	1.7	n/a	1.3	1.3	7.0	1.8
genetic alg.	n/a	8.3	18.5	100.6	50.1	n/a	8.4	19.1	96.2	54.4
$F_{Mm}^*$ -value (%)										
hillclimber	n/a	95.2	96.9	97.0	96.7	n/a	95.4	96.6	97.2	96.7
genetic alg.	n/a	96.6	96.0	97.6	97.7	n/a	95.5	96.8	97.5	97.0
(III) Estimation: individual stimulus										
CPU time (minutes)										
hillclimber	1.8	2.1	2.3	14.7	5.0	2.5	3.0	4.2	22.4	13.8
genetic alg.	4.4	10.6	26.6	158.8	149.1	5.7	15.4	42.3	29.0	19.4
$F_{Mm}^*$ -value (%)										
hillclimber	97.2	96.9	96.7	97.1	97.3	99.9	99.7	99.5	100.0	97.7
genetic alg.	96.8	97.3	97.2	96.9	97.1	99.8	100.0	100.0	99.8	100.0
(IV) Estimation: pairwise comparison										
CPU time (minutes)										
hillclimber	n/a	1.7	4.3	16.0	12.6	n/a	2.9	4.9	25.5	18.9
genetic alg.	n/a	12.4	28.4	154.5	149.2	n/a	14.6	37.2	221.6	275.1
Maximin efficiency (%)										
hillclimber	n/a	97.0	96.4	97.1	94.8	n/a	100.0	99.4	99.8	98.5
genetic alg.	n/a	97.0	96.9	96.8	96.2	n/a	99.5	100.0	100.0	100.0
(V) Detection + estimation (equal weights): individual stimulus										
CPU time (minutes)										
hillclimber	6.4	7.8	6.6	40.6	11.6	8.3	10.6	10.7	54.5	27.7
genetic alg.	16.2	31.0	69.5	425.8	396.2	15.9	42.0	112.0	355.2	497.6
$F_{Mm}^*$ -value (%)										
hillclimber	98.8	98.7	98.5	99.0	98.6	99.1	99.3	98.8	99.3	98.9
genetic alg.	98.7	99.0	98.8	98.8	98.4	99.2	99.0	99.3	99.3	99.2
(VI) Detection + estimation (equal weights): pairwise comparison										
CPU time (minutes)										
hillclimber	n/a	4.8	8.2	37.6	19.8	n/a	6.6	9.7	58.5	30.8
genetic alg.	n/a	34.1	76.0	438.4	645.9	n/a	43.0	114.3	673.8	649.3
Maximin efficiency (%)										
hillclimber	n/a	98.5	98.5	98.7	96.6	n/a	99.1	98.3	98.9	98.0
genetic alg.	n/a	98.3	98.7	98.7	98.5	n/a	99.4	99.4	99.2	99.5

Table 4. The relative maximin efficiency,  $[F_{Mm}^*(\xi)/F_{Mm}^*(\xi_{HC}^*)] \times 100\%$ , of the locally optimal design  $\xi_{0.3,HC}^*$  for  $\rho = 0.3$  and the worst relative maximin efficiency over the design class  $\Xi_{HC}^*$  of the 51 locally optimal designs obtained by the hill-climbing approach;  $\xi_{HC}^*$  is the maximin design over  $\Xi_{HC}$ .

$Q =$	A-optimality					D-optimality				
	1	2	3	4	6	1	2	3	4	6
(I) Detection: individual stimulus										
$\rho = 0.3$	92.9	96.5	94.7	96.5	92.5	92.9	96.8	95.4	98.5	94.9
Worst case	83.2	83.3	81.9	85.0	88.5	83.2	85.5	89.4	89.2	91.7
(II) Detection: pairwise comparison										
$\rho = 0.3$	n/a	100.0	98.8	97.7	96.4	n/a	100.0	98.4	98.9	97.5
Worst case	n/a	91.1	87.4	89.6	90.9	n/a	91.1	87.7	90.3	91.4
(III) Estimation: individual stimulus										
$\rho = 0.3$	99.2	98.8	98.8	99.4	96.6	99.8	99.9	99.9	99.9	100.0
Worst case	92.6	91.2	91.5	92.5	88.5	99.5	99.6	99.2	99.7	98.2
(IV) Estimation: pairwise comparison										
$\rho = 0.3$	n/a	99.3	99.6	99.2	100.0	n/a	99.6	99.9	99.8	99.9
Worst case	n/a	91.5	91.5	91.9	86.8	n/a	98.1	98.6	99.5	98.1
(V) Detection + estimation (equal weights): individual stimulus										
$\rho = 0.3$	100.0	99.8	99.1	99.6	100.0	99.7	99.7	100.0	99.8	99.7
Worst case	96.4	95.9	94.8	96.2	93.8	97.3	97.3	96.5	97.3	96.9
(VI) Detection + estimation (equal weights): pairwise comparison										
$\rho = 0.3$	n/a	99.7	98.6	99.4	97.2	n/a	100.0	99.7	99.8	98.4
Worst case	n/a	94.9	39.7	96.3	94.3	n/a	96.8	97.5	97.5	96.8

Table 5. The  $F_{Mm2}^*$ -value (%) of a near maximin design for AR2 noise, and the corresponding  $(\phi_1, \phi_2)$ -value that yields this design.

$Q =$	A-optimality					D-optimality				
	1	2	3	4	6	1	2	3	4	6
(I) Detection: individual stimulus										
$F_{Mm2}^*$	91.9	91.1	90.8	91.2	92.1	91.9	89.9	91.2	92.1	91.7
$\phi_1^*$	0.39	0.15	0.18	0.00	0.07	0.39	0.00	0.31	0.05	0.31
$\phi_2^*$	0.13	0.23	0.21	0.30	0.24	0.13	0.30	0.09	0.22	0.12
(II) Detection: pairwise comparison										
$F_{Mm2}^*$	n/a	85.9	90.2	90.6	90.9	n/a	85.9	90.4	89.4	91.3
$\phi_1$	n/a	0.02	0.00	0.45	0.19	n/a	0.02	0.00	0.00	0.35
$\phi_2$	n/a	0.29	0.25	0.02	0.18	n/a	0.29	0.25	0.27	0.04
(III) Estimation: individual stimulus										
$F_{Mm2}^*$	96.1	96.2	96.1	95.8	95.6	99.5	99.8	99.7	99.8	99.6
$\phi_1$	0.16	0.15	0.20	0.16	0.21	0.14	0.04	0.14	0.20	0.05
$\phi_2$	0.27	0.29	0.31	0.20	0.13	0.44	0.31	0.39	0.30	0.28
(IV) Estimation: pairwise comparison										
$F_{Mm2}^*$	n/a	96.6	96.2	96.1	95.9	n/a	99.5	99.5	99.7	99.3
$\phi_1$	n/a	0.20	0.15	0.20	0.19	n/a	0.31	0.25	0.08	0.30
$\phi_2$	n/a	0.23	0.26	0.26	0.10	n/a	0.09	0.00	0.29	0.00
(V) Detection + estimation (equal weights): individual stimulus										
$F_{Mm2}^*$	97.6	97.9	97.8	97.8	97.6	96.8	96.9	96.7	96.9	97.7
$\phi_1$	0.21	0.15	0.20	0.18	0.18	0.49	0.30	0.00	0.00	0.00
$\phi_2$	0.29	0.33	0.38	0.26	0.21	0.01	0.10	0.28	0.33	0.32
(VI) Detection + estimation (equal weights): pairwise comparison										
$F_{Mm2}^*$	n/a	97.6	97.6	96.3	97.2	n/a	97.2	97.2	96.8	97.7
$\phi_1$	n/a	0.16	0.19	0.36	0.18	n/a	0.01	0.05	0.15	0.22
$\phi_2$	n/a	0.23	0.25	0.20	0.29	n/a	0.30	0.25	0.24	0.18

Table 6. The  $F_{Mm2}^*$ -value (%) of the obtained locally optimal design  $\xi_{0.3,HC}$  for AR1 noise with  $\rho = 0.3$ .

$Q =$	A-optimality					D-optimality				
	1	2	3	4	6	1	2	3	4	6
Case (I)	62.8	66.9	62.8	67.6	56.8	62.8	67.7	66.5	75.4	65.7
Case (II)	n/a	74.5	73.0	72.2	71.4	n/a	74.5	72.7	75.8	72.9
Case (III)	88.7	88.1	88.8	88.2	92.4	98.4	99.0	99.3	99.1	99.0
Case (IV)	n/a	90.7	93.2	89.1	92.1	n/a	98.6	99.5	99.0	99.3
Case (V)	92.6	90.7	92.7	93.1	94.7	91.5	91.6	91.5	91.8	92.6
Case (VI)	n/a	92.7	91.2	92.8	91.9	n/a	93.8	91.9	91.5	90.4

**Figure Caption:**

Figure 1. The CPU time (minutes) and the achieved maximal  $F^* = (F_c^* + F_d^* + F_e^*)/3$  of the proposed hill-climbing approach and the genetic algorithm for individual stimulus effects (A1 & A2) and pairwise comparisons (B1 & B2) with both  $A$ - and  $D$ -optimality criteria and  $(Q, L) = (2, 242), (3, 255), (4, 624)$  and  $(6, 342)$ .