Detecting Arrays for Main Effects

Charles J. Colbourn(✉) and Violet R. Syrotiuk

School of Computing, Informatics, and Decision Systems Engineering,
Arizona State University, Tempe, AZ, USA
{colbourn,syrotiuk}@asu.edu

Abstract. Determining correctness and performance for complex engineered systems necessitates testing the system to determine how its behaviour is impacted by many factors and interactions among them. Of particular concern is to determine which settings of the factors (main effects) impact the behaviour significantly. Detecting arrays for main effects are test suites that ensure that the impact of each main effect is witnessed even in the presence of d or fewer other significant main effects. Separation in detecting arrays dictates the presence of at least a specified number of such witnesses. A new parameter, corroboration, enables the fusion of levels while maintaining the presence of witnesses. Detecting arrays for main effects, having various values for the separation and corroboration, are constructed using error-correcting codes and separating hash families. The techniques are shown to yield explicit constructions with few tests for large numbers of factors.

1 Introduction

Combinatorial testing [21,31] addresses the design and analysis of test suites in order to evaluate correctness (and, more generally, performance) of complex engineered systems. To set the stage, we introduce some basic definitions. There are k factors $F_1, \ldots, F_k$. Each factor F_i has a set $S_i = \{v_{i1}, \ldots, v_{is_i}\}$ of s_i possible *levels* (or *values* or *options*). A *test* is an assignment of a level from $v_{i1}, \ldots, v_{is_i}$ to F_i, for each $1 \leq i \leq k$. The execution of a test yields a measurement of a *response*. When $\{i_1, \ldots, i_t\} \subseteq \{1, \ldots, k\}$ and $\sigma_{i_j} \in S_{i_j}$, the set $\{(i_j, \sigma_{i_j}) : 1 \leq j \leq t\}$ is a *t-way interaction*. The value of t is the *strength* of the interaction. A *main effect* is a 1-way interaction. A test on k factors *covers* $\binom{k}{t}$ t-way interactions. A *test suite* is a collection of tests. A test suite is typically represented as an $N \times k$ array $A = (\sigma_{i,j})$ in which $\sigma_{i,j} \in S_j$ when $1 \leq i \leq N$ and $1 \leq j \leq k$. The *size* of the test suite is N and its *type* is $(s_1, \ldots, s_k)$. Tests correspond to rows of A, and factors correspond to its columns.

When the response of interest can depend on one or more interactions, each having strength at most t, a test suite must cover each interaction in at least one row (test). To make this precise, let $A = (\sigma_{i,j})$ be a test suite of size N and type $(s_1, \ldots, s_k)$. Let $T = \{(i_j, \sigma_{i_j}) : 1 \leq j \leq t\}$ be a t-way interaction. Then $\rho_A(T)$ denotes the set $\{r : a_{ri_j} = \sigma_{i_j}, 1 \leq j \leq t\}$ of rows of A in which the interaction

M. Ćirić et al. (Eds.): CAI 2019, LNCS 11545, pp. 112–123, 2019.
https://doi.org/10.1007/978-3-030-21363-3_10

is covered. A t-way interaction T must have $|\rho_A(T)| \geq 1$ in order to impact the response. For a set $\mathcal{T}$ of interactions, $\rho_A(\mathcal{T}) = \bigcup_{T \in \mathcal{T}} \rho_A(T)$.

When used in practical testing applications, as in [1,18,33], further requirements arise. First, if we suppose that some set $\mathcal{T}$ of interactions are those that significantly impact the response, yet there is another interaction $T \notin \mathcal{T}$ for which $\rho_A(T) \subseteq \bigcup_{S \in \mathcal{T}} \rho_A(S)$, the responses are inadequate to determine whether or not T impacts the response significantly. This requirement was explored in [14], and later in [15,16,27]. Secondly, one or more tests may fail to execute correctly, and yield no response or yield outlier responses. To mitigate this, Seidel *et al.* [34] impose stronger 'separation' requirements on the test suite.

Extending definitions in [14,16,34], we formally define the test suites with which we are concerned. Let A be a test suite of size N and type $(s_1, \ldots, s_k)$. Let $\mathcal{I}_t$ be the set of all t-way interactions for A. Our objective is to identify the set $\mathcal{T} \subseteq \mathcal{I}_t$ of interactions that have significant impact on the response. In so doing, we assume that at most d interactions impact the response. Without limiting d, it can happen that no test suite of type $(s_1, \ldots, s_k)$ exists for any value of N [27].

An $N \times k$ array A of type $(s_1, \ldots, s_k)$ is (d, t, δ)-*detecting* if $|\rho_A(T) \setminus \rho_A(\mathcal{T})| < \delta \Leftrightarrow T \in \mathcal{T}$ whenever $\mathcal{T} \subseteq \mathcal{I}_t$, and $|\mathcal{T}| = d$. To record all of the parameters, we use the notation $\mathsf{DA}_\delta(N; d, t, k, (s_1, \ldots, s_k))$. To emphasize that different factors may have different numbers of levels, this is a *mixed* detecting array. When all factors have the same number, v, of levels, the array is *uniform* and the notation is simplified to $\mathsf{DA}_\delta(N; d, t, k, v)$. The parameter δ is the *separation* of the detecting array [34], and the definition in [14] is recovered by setting $\delta = 1$. Rows in $\rho_A(T) \setminus \rho_A(\mathcal{T})$ are *witnesses* for T that are not masked by interactions in $\mathcal{T}$. A separation of δ necessitates δ witnesses, ensuring that fewer than δ missed or incorrect measurements cannot result in an interaction's impact being lost.

Setting $d = 0$ in the definition, $\mathcal{T} = \emptyset$ and $\rho_A(\emptyset) = \emptyset$. Then a $(0, t, \delta)$-detecting array is an array in which each t-way interaction is covered in at least δ rows. This leads to a standard class of testing arrays for testing: A *covering array* $\mathsf{CA}_\delta(N; t, k, (s_1, \ldots, s_k))$ is equivalent to a $\mathsf{DA}_\delta(N; 0, t, k, (s_1, \ldots, s_k))$. Again the simpler notation $\mathsf{CA}_\delta(N; t, k, v)$ is employed when it is uniform.

In this paper we focus on detecting arrays for main effects. In Sect. 2, we develop a further parameter, corroboration, for detecting arrays to facilitate the construction of mixed detecting arrays from uniform ones. In Sect. 3 we briefly summarize what is known about the construction of detecting arrays. In Sect. 4 we develop constructions of $(1, 1)$-detecting arrays with specified corroboration and separation using results on perfect hash families of strength two and higher index, or (equivalently) using certain error-correcting codes. In Sect. 5 we extend these constructions to $(d, 1)$-detecting arrays for $d > 1$ using a generalization of perfect hash families, the separating hash families.

2 Fusion and Corroboration

Covering arrays have been much more extensively studied [10,21,31] than have detecting arrays and their variants; they are usually defined only in the case when

$\delta = 1$, and in a more direct manner than by exploiting the equivalence with certain detecting arrays. Often constructions of covering arrays focus on the uniform cases. In part this is because a $\mathsf{CA}_\delta(N; t, k, (s_1, \ldots, s_{i-1}, s_{i-1}, s_{i+1}, \ldots, s_k))$ can be obtained from a $\mathsf{CA}_\delta(N; t, k, (s_1, \ldots, s_{i-1}, s_i, s_{i+1}, \ldots, s_k))$ by making any two levels of the ith factor identical. This operation is *fusion* (see, e.g., [11]).

When applied to detecting arrays with $\delta \geq 1$, however, fusion may reduce the number of witnesses. Increasing the separation cannot overcome this problem. Because techniques for uniform covering arrays are better developed than for mixed ones, generalizations to detecting arrays can be expected to be again more tractable for uniform cases. As with covering arrays, fusion for detecting arrays promises to extend uniform constructions to mixed cases.

In order to facilitate this, we propose an additional parameter for detecting arrays. We begin with a useful characterization. Let A be an $N \times k$ array. Let $T = \{(i_j, \sigma_{i_j}) : 1 \leq j \leq t\}$ be a t-way interaction for A. Let $C = \{c_i : 1 \leq i \leq d\}$ be a set of d column indices of A with $\{i_1, \ldots, i_t\} \cap \{c_1, \ldots, c_d\} = \emptyset$. A set system $\mathcal{S}_{A,T,C}$ is defined on the ground set $\{(c, f) : c \in C, f \in S_c\}$ containing the collection of sets $\{\{(c_1, v_1), \ldots, (c_d, v_d)\} : T \cup \{(c_1, v_1), \ldots, (c_d, v_d)\}$ is covered in $A\}$.

Lemma 1. *An array A is (d, t, δ)-detecting if and only if for every t-way interaction T and every set C of d disjoint columns, every subset X of elements of the set system $\mathcal{S}_{A,T,C}$, whose removal (along with all sets containing an element of X) leaves fewer than δ sets in $\mathcal{S}_{A,T,C}$, satisfies $|X| > d$.*

Proof. First suppose that for some t-way interaction $T = \{(i_j, \sigma_{i_j}) : 1 \leq j \leq t\}$ and some set $C = \{c_i : 1 \leq i \leq d\}$ of d disjoint columns, in the set system $\mathcal{S}_{A,T,C}$ there is a set of elements $X = \{(c_1, v_1), \ldots, (c_d, v_d)\}$ for which fewer than δ sets in the set system contain no element of X. Define $T_i = \{(i_j, \sigma_{i_j}) : 1 \leq j \leq t-1\} \cup \{(c_i, v_i)\}$. Set $\mathcal{T} = \{T_1, \ldots, T_d\}$. Then $T \notin \mathcal{T}$ but $|\rho_A(T) \setminus \rho_A(\mathcal{T})| < \delta$, so A is not (d, t, δ)-detecting.

In the other direction, suppose that A is not (d, t, δ)-detecting, and consider a set $\mathcal{T} = \{T_1, \ldots, T_d\}$ of d t-way interactions and a t-way interaction T for which $T \notin \mathcal{T}$ but $|\rho_A(T) \setminus \rho_A(\mathcal{T})| < \delta$. Without loss of generality, there is no interaction $T' \in \mathcal{T}$ for which T and T' share a factor set to different levels in each (and so, because $T \neq T'$, T' contains a factor not appearing in T). For each $T_i \in \mathcal{T}$, let c_i be a factor in T_i that is not in T', and suppose that $(c_i, v_i) \in T_i$ for $1 \leq i \leq d$. Then the set $X = \{(c_i, v_i) : 1 \leq i \leq d\}$, when removed from $\mathcal{S}_{A,T,C}$, leaves fewer than δ sets. □

Lemma 1 implies that a (d, t, δ)-detecting array must cover each t-way interaction at least $d + \delta$ times; indeed when $d \geq 1$, for each t-way interaction T and every column c not appearing in T, interaction T must be covered in at least $d + 1$ rows containing distinct levels in column c. In particular, a necessary condition for a $\mathsf{DA}_\delta(N; d, t, k, (s_1, \ldots, s_{i-1}, s_{i-1}, s_{i_1}, \ldots, s_k)))$ to exist is that $d < \min(s_i : 1 \leq i \leq k)$ (see also [14]).

These considerations lead to the parameter of interest. For array A, with t-way interaction T and set C of d disjoint columns, suppose that in $\mathcal{S}_{A,T,C}$, for each column in C one performs fewer than s fusions of elements within those arising from that column. Further suppose that, no matter how these fusions are done, the resulting set system has the property that every subset X of elements of the set system, whose removal (along with all sets containing an element of X) leaves fewer than δ sets, satisfies $|X| \geq d+1$. Then (T, C) has *corroboration* s in A. When every choice of (T, C) has corroboration (at least) s in a $\mathsf{DA}_\delta(N; d, t, k, (s_1, \dots, s_k))$, it has *corroboration* s. We extend the notation as $\mathsf{DA}_\delta(N; d, t, k, (s_1, \dots, s_k), s)$ to include corroboration s as a parameter.

3 Covering Arrays and Sperner Partition Systems

As observed in [14], one method to construct detecting arrays is to use covering arrays of higher strength. The following records consequences for separation and corroboration.

Lemma 2. *A* $\mathsf{CA}_\lambda(N; t, k, v)$ *is*

1. *a* $\mathsf{DA}_\delta(N; d, t-d, k, v, 1)$ *with* $\delta = \lambda(v-d)v^{d-1}$*, and*
2. *a* $\mathsf{DA}_\delta(N; d, t-d, k, v, v-d)$ *with* $\delta = \lambda(d+1)^{d-1}$

whenever $1 \leq d < \min(t, v)$*.*

Proof. Let A be a $\mathsf{CA}_\lambda(N; t, k, v)$. Let d satisfy $1 \leq d < \min(t, v)$. Let T be a $(t-d)$-way interaction, and let C be a set of d columns not appearing in T. Using the parameters of the covering array, $\mathcal{S}_{A,T,C}$ contains at least λv^d sets, and each element appears in at least λv^{d-1} of them. Suppose that d elements of $\mathcal{S}_{A,T,C}$ are removed, and further suppose that the numbers of elements deleted for the d factors are $e_1, \dots, e_d$ (so that $d = \sum_{i=1}^{d} e_i$). Then the number of remaining sets is $\lambda \prod_{i=1}^{d}(v-e_i)$, which is minimized at $\delta = \lambda(v-d)v^{d-1}$. This establishes the first statement. For the second, performing at most $v-d-1$ fusions within each factor of $\mathcal{S}_{A,T,C}$ and then deleting at most d elements leaves at least $\delta = \lambda(d+1)^{d-1}$ sets by a similar argument. □

The effective construction of detecting arrays is well motivated by practical testing applications, in which the need for higher separation to mitigate the effects of outlier responses, and higher corroboration to support fusion of levels, arise. Despite this, other than the construction from covering arrays of higher strength, few constructions are available. In [43] uniform $(1, t)$-detecting arrays with separation 1, corroboration 1, and few factors are studied. This was extended in [36,38] to (d, t)-detecting arrays, and further to mixed detecting arrays in [37]. Each of these focuses on the determination of a lower bound on the number of rows in terms of d, t, and v, and the determination of cases in which this bound can be met. For $d + t \geq 2$, however, the number of rows must grow at least logarithmically in k, because every two columns must be distinct. Hence the study of arrays meeting bounds that are independent of k necessarily considers only small values of k. In addition, none of these addresses separation or corroboration.

For larger values of k, algorithmic methods are developed in [34]. The algorithms include randomized methods based on the Stein-Lovász-Johnson framework [20,25,40], and derandomized algorithms using conditional expectations (as in [7,8]); randomized methods based on the Lovász Local Lemma [3,19] and derandomizations using Moser-Tardos resampling [30] (as in [12]). Although these methods produce $(1, t)$-mixed detecting arrays for a variety of separation values, they have not been applied for $d > 1$ or to increase the corroboration. Extensions to larger d for locating arrays are considered in [23].

When $t = 1$, one is considering detecting arrays for main effects. A *Sperner family* is a family of subsets of some ground set such that no set in the family is a subset of any other. Meagher et al. [28] introduced Sperner partition systems as a natural variant of Sperner families. An (n, v)-*Sperner partition system* is a collection of partitions of some n-set, each into v nonempty classes, such that no class of any partition is a subset of a class of any other. In [24,28], the largest number of classes in an (n, v)-Sperner partition system is determined exactly for infinitely many values of n for each v. In [9], lower and upper bounds that match asymptotically are established for all n and each v. As noted there, given an (n, v)-Sperner partition system with k partitions, if we number the elements using $\{1, \ldots, n\}$ and number the sets in each partition with $\{1, \ldots, v\}$, we can form an $n \times k$ array in which cell (r, c) contains the set number to which element r belongs in partition c. This array is a $\mathsf{DA}_1(n; 1, 1, k, v, 1)$, and indeed every such DA arises in this way. Even when $d = t = s = \delta = 1$, the largest value of k as a function of n is not known precisely. Therefore it is natural to seek useful bounds and effective algorithms for larger values of the parameters.

4 (1, 1, δ)-Detecting Arrays

In this section, we consider the case when $d = t = 1$. As noted, Sperner partition systems address the existence of such detecting arrays when the separation $\delta = 1$. A naive way to increase the separation simply forms δ copies of each row in a $\mathsf{DA}_1(N; 1, 1, k, v, 1)$ to form a $\mathsf{DA}_\delta(\delta N; 1, 1, k, v, 1)$. This leaves the corroboration unchanged; in addition, it employs more rows than are needed to obtain the increase in separation. In order to treat larger values of separation and corroboration, we employ further combinatorial arrays.

An $(N; k, v)$-*hash family* is an $N \times k$ array on v symbols. A *perfect hash family* $\mathsf{PHF}_\lambda(N; k, v, t)$ is an $(N; k, v)$-hash family, in which in every $N \times t$ subarray, at least λ rows each consist of distinct symbols. Mehlhorn [29] introduced perfect hash families, and they have subsequently found many applications in combinatorial constructions [41].

Colbourn and Torres-Jiménez [17] relax the requirement that each row have the same number of symbols. An $N \times k$ array is a *heterogeneous hash family*, or $\mathsf{HHF}(N; k, (v_1, \ldots, v_N))$, when the ith row contains (at most) v_i symbols for $1 \leq i \leq N$. The definition for PHF extends naturally to perfect *heterogeneous* hash families; we use the notation $\mathsf{PHHF}_\lambda(N; k, (v_1, \ldots, v_N), t)$.

Returning to detecting arrays, we first consider larger separation.

Lemma 3. *Whenever a* $\mathsf{PHF}_\delta(N; k, v, 2)$ *exists, a* $\mathsf{DA}_\delta(v(N+\delta); 1, 1, k, v, 1)$ *exists.*

Proof. Let A be a $\mathsf{PHF}_\delta(N; k, v, 2)$ on symbols $\{0, \ldots, v-1\}$. Let A_i be the array obtained from A by adding i modulo v to each entry of A. Let B be the $\delta v \times k$ array consisting of δ rows containing only symbol i, for each $i \in \{0, \ldots, v-1\}$. Vertically juxtapose $A_0, \ldots, A_{v-1}$, and B to form a $v(N+\delta) \times k$ array D. To verify that D is a $\mathsf{DA}_\delta(v(N+\delta); 1, 1, k, v, 1)$, consider a main effect (c, σ) and let $c' \neq c$ be a column. Among the rows of D covering (c, σ), we find σ exactly δ times in the rows of B (and perhaps among rows of one or more of the $\{A_i\}$). Further, each of the δ rows in the PHF having different symbols in columns c and c' yield a row in one of the $\{A_i\}$ in which (c, σ) appears but c' contains a symbol different from σ. Hence no symbol in c' can cover all but $\delta - 1$ rows containing (c, σ). □

When does a $\mathsf{PHF}_\delta(N; k, v, 2)$ exist? Treating columns as codewords of length N on a v-ary alphabet, two different codewords are at Hamming distance at least δ. Hence such a $\mathsf{PHF}_\delta(N; k, v, 2)$ is exactly a v-ary code of length N and minimum distance δ, having k codewords. (See [26] for definitions in coding theory.) When $\delta = 1$, the set of all v^N codewords provides the largest number of codewords, while for $\delta = 2$, the set of v^{N-1} codewords having sum 0 modulo v provides the largest code. For $\delta \geq 3$, however, the existence question for such codes is far from settled, particularly when $v > 2$ (see [22], for example). As applied here, this fruitful connection with codes permits increase in the separation but not the corroboration. We address this next.

Construction 1 *(h-***inflation***). Let* v *be a prime power and let* $1 \leq h \leq v$*. Let* $\{e_0, \ldots, e_{v-1}\}$ *be the elements of* $\mathbb{F}_v$*. Let* A *be an* $(N; k, v+1)$*-hash family on* $\{e_0, \ldots, e_{v-1}\} \cup \{\infty\}$*. Define* 2×1 *column vectors* $\mathcal{C}_h$ *containing* $\mathbf{c}_\infty = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ *and* $\mathbf{c}_x = \begin{pmatrix} x \\ 1 \end{pmatrix}$ *for* $x \in \mathbb{F}_v$*. Form a set of* r_h *row vectors* $\mathcal{R}_h = (\mathbf{r}_1, \ldots, \mathbf{r}_{r_h})$ *so that for every* $\mathbf{c}_a \in \mathcal{C}_h$*, each* $\mathbf{d}_a = (\mathbf{r}_i \mathbf{c}_a : 1 \leq i \leq r_h)$ *contains each entry of* $\mathbb{F}_v$ *at least* h *times. Form* B *by replacing each element* a *in array* A *by the column vector* $\mathbf{d}_a^T$*. Then* B *is a* $(r_h N; k, v)$*-hash family, an* h-inflation *of* A*.*

In Construction 1, each column vector $\mathbf{d}_a$ contains each element of the field at least h times. Moreover, if $a \neq b$, the h coordinates in which $\mathbf{d}_a$ contains a specific element of the field contain h different elements in these coordinates in $\mathbf{d}_a$. Both facts can be easily checked.

Lemma 4. *Whenever* v *is a prime power, a* $\mathsf{PHF}_\delta(N; k, v+1, 2)$ *exists, and* $1 \leq s \leq v-1$*, a* $\mathsf{DA}_{\delta s}((r_{s+1} N; 1, 1, k, v, s)$ *exists.*

Proof. Using Construction 1 and the subsequent facts, any $(s+1)$-inflation, B, of a $\mathsf{PHF}_\delta(N; k, v+1, 2)$, A, is a $\mathsf{DA}_{\delta s}((r_{s+1} N; 1, 1, k, v, s)$. □

Given a $\mathsf{PHF}_\delta(N; k, v+1, 2)$, Lemma 4 produces a $\mathsf{DA}_{\delta(v-1)}(v^2 N; 1, 1, k, v, v-1)$ that is, in fact, a covering array $\mathsf{CA}_\delta(Nv; 2, k, v)$. Although this does not lead to the largest number of columns in a covering array with these parameters when $\delta = 1$ (compare with [13]), it is competitive and applies for all δ. More importantly, one can make detecting arrays for a variety of separation and corroboration values.

To illustrate this, we adapted the 'replace-one-column–random extension' randomized algorithm from [12] in order to construct PHFs of index δ. In the interests of space, we do not describe the method here, noting only that it is an heuristic technique that is not expected to produce optimal sizes. In Table 1 we report the largest number of columns found for a $\mathsf{PHF}_\delta(N; k, 6, 2)$ for various values of N and $1 \leq \delta \leq 4$. Recall that each is equivalent to a 6-ary code of length N and minimum distance δ with k codewords.

Table 1. Number k of columns found for a $\mathsf{PHF}_\delta(N; k, 6, 2)$

$\delta \downarrow N \rightarrow$	1	2	3	4	5	6	7	8	9	10
1	6	36	216	1296	7776	46656				
2		6	36	216	1296	7776	46656			
3			6	33	156	704	3156	14007		
4				6	30	116	429	1776	7406	26374

Suppose that we are concerned with a large (but fixed) number of factors, such as 10000. Together with the Lemma 4, the results in Table 1 imply, for example, the existence of the following:

$$\begin{array}{lll} \mathsf{DA}_1(84; 1, 1, 10000, 5, 1) & \mathsf{DA}_2(114; 1, 1, 10000, 5, 2) & \mathsf{CA}_1(150; 2, 10000, 5) \\ \mathsf{DA}_2(98; 1, 1, 10000, 5, 1) & \mathsf{DA}_4(133; 1, 1, 10000, 5, 2) & \mathsf{CA}_2(175; 2, 10000, 5) \\ \mathsf{DA}_3(112; 1, 1, 10000, 5, 1) & \mathsf{DA}_6(152; 1, 1, 10000, 5, 2) & \mathsf{CA}_3(200; 2, 10000, 5) \\ \mathsf{DA}_4(140; 1, 1, 10000, 5, 1) & \mathsf{DA}_8(190; 1, 1, 10000, 5, 2) & \mathsf{CA}_4(250; 2, 10000, 5) \end{array}$$

These examples demonstrate not only that increases in both separation and corroboration can be accommodated with a reasonable increase in the number of rows, but also that detecting arrays for main effects can be constructed for very large numbers of factors.

5 $(d, 1, \delta)$-Detecting Arrays

Next we extend these methods to treat higher values of d. To do so, we employ a generalization of PHFs. An $(N; k, v, \{w_1, w_2, \ldots, w_t\})$-*separating hash family* of *index* λ is an $(N; k, v)$-hash family A that satisfies the property: For any $C_1, C_2, \ldots, C_t \subseteq \{1, 2, \ldots, k\}$ such that $|C_1| = w_1$, $|C_2| = w_2$, $\ldots$, $|C_t| = w_t$, and $C_i \cap C_j = \emptyset$ for every $i \neq j$, whenever $c \in C_i$, $c' \in C_j$, and $i \neq j$, different symbols appear in columns c and c' in each of at least λ rows. The notation $\mathsf{SHF}_\lambda(N; k, v, \{w_1, w_2, \ldots, w_t\})$ is used. See, for example, [2,32,39]; and see [4] for the similar notion of 'partially hashing'. When heterogeneous, we use the notation $\mathsf{SHHF}_\lambda(N; k, (v_1, \ldots, v_N), \{w_1, w_2, \ldots, w_t\})$. In the particular case of $\mathsf{SHF}_1(N; k, v, \{1, d\})$, these are *frameproof codes* (see, for example, [39,42]).

Theorem 1. *Let v be a prime power. When an $\mathsf{SHF}_\delta(N; k, v+1, \{1, d\})$ exists, and $1 \leq s \leq v - d$, a $\mathsf{DA}_{\delta s}(r_{s+d}N; d, 1, k, v, 1)$ and a $\mathsf{DA}_\delta(r_{s+d}N; d, 1, k, v, \lfloor(s + d - 1)/d\rfloor)$ exist.*

Proof. Using Construction 1, let B be an $(s+d)$-inflation of an $\mathsf{SHF}_\delta(N; k, v + 1, \{1, d\})$, A. Then B is a $r_{s+d}N \times k$ array with entries from $\mathbb{F}_v$. Now consider a set of distinct columns $\{c, c_1, \ldots, c_d\}$ of A. Let R be the set of (at least δ) rows of A in which the entry in column c does not appear in any of columns $\{c_1, \ldots, c_d\}$. For each $\sigma \in \mathbb{F}_v$, the inflation of a row in R yields at least $s + d$ rows in which column c contains σ and each of $\{c_1, \ldots, c_d\}$ contains $d + s$ distinct symbols. Indeed, setting $T = \{(c, \sigma)\}$ and $C = \{c_1, \ldots, c_d\}$, the inflation of each row in R places $d + s$ *mutually disjoint* sets in $\mathcal{S}_{B,T,C}$. Consequently, any removal of d elements from $\mathcal{S}_{B,T,C}$ can remove at most d of the $s + d$ sets arising from a row in R. Hence at least δs must remain, and B is a detecting array with separation (at least) δs. Identification of fewer than $\lfloor(s + d - 1)/d\rfloor$ levels for each factor of $\mathcal{S}_{B,T,C}$ leaves at least δ sets, giving the second DA. □

In order to apply Theorem 1, we require $\{1, d\}$-separating hash families. Their existence is well studied for $\delta = 1$ (see [35] and references therein), but they appear not to have been studied when $\delta > 1$. When $\delta = 1$, Blackburn [6] establishes that an $\mathsf{SHF}_1(N; k, v, \{1, d\})$ can exist only when $k \leq dv^{\lceil \frac{N}{d} \rceil} - d$. Stinson et al. [42] use an expurgation technique to establish lower bounds on k for which an $\mathsf{SHF}_1(N; k, v, \{1, d\})$ exists. One consequence of their results is that an $\mathsf{SHF}_1(N; k, v, \{1, 2\})$ exists for $k = \left\lceil \frac{1}{2}\left(\frac{v^2}{2v-1}\right)^{\frac{N}{2}} \right\rceil$.

Let us consider a concrete set of parameters. Suppose that we are to construct an $\mathsf{SHF}_1(13; k, 6, \{1, 2\})$. The bounds ensure that the largest value of k for which one exists satisfies $1112 \leq k \leq 559870$. A straightforward computation yields such an SHF with $k = 8014$. Naturally one hopes to improve on both the lower and upper bounds, and to generalize them to cases with separation more than $\delta = 1$. Error-correcting codes are not equivalent to the SHF s required when $d > 1$, but they again provide constructions; we leave this discussion for later work. Nevertheless, there appears to be a need to resort to computation as well.

Table 2 gives the largest values of k that we found for an $\mathsf{SHF}_\delta(N; k, 6, \{1, 2\})$ for $1 \leq \delta \leq 4$ and various values of N. Each yields a $\mathsf{DA}_\delta(15N; 2, 1, k, 5, 1)$ (and other detecting arrays, from Theorem 1).

The entries in Table 2 have again been determined using a variant of the 'replace-one-column–random-extension' algorithm developed in [12]. This heuristic method is not expected in general to yield the largest possible number of columns (and the lower and upper bounds on such largest numbers are currently far apart). When the number of rows is small, however, we can make some comparisons, and we do this next. First we establish an upper bound on k when $N \leq d + \delta - 1$.

Table 2. Number k of columns found for an $\mathsf{SHF}_\delta(N; k, 6, \{1, 2\})$

$\delta \downarrow N \rightarrow$	1	2	3	4	5	6	7	8	9	10	11	12
1	6	10	36	51	154	201	373	634	1003	1751	2825	4578
2		6	7	34	39	142	152	262	342	529	805	1257
3			6	6	30	32	72	80	168	195	328	486
4				6	6	27	27	56	58	125	134	231
$\delta \downarrow N \rightarrow$	13	14	15	16	17	18	19	20	21	22	23	24
1	8068	10000										
2	2041	3163	4920	8431	10000							
3	716	1086	1695	2543	3891	6290	9878	10000				
4	311	466	696	1005	1540	2310	3387	5181	8242	10000		

Lemma 5. *Let $d \geq 2$, $\delta \geq 1$, and $\alpha \geq 1$. Then*

$$k \leq \max\left(v_1, \ldots, v_{d+\delta-\alpha}, \left\lfloor \frac{\sum_{i=1}^{d+\delta-\alpha}(v_i - 1)}{\delta} \right\rfloor\right)$$

whenever an $\mathsf{SHHF}_\delta(d + \delta - \alpha; k, (v_1, \ldots, v_{d+\delta-\alpha}), \{1, d\})$ exists.

Proof. Let A be an $\mathsf{SHHF}_\delta(d + \delta - 1; k, (v_1, \ldots, v_{d+\delta-\alpha}), \{1, d\})$. An entry in A is a *private* entry if it contains the only occurrence of a symbol in its row. If some row contains only private entries, then $k \leq \max(v_1, \ldots, v_{d+\delta-\alpha})$. If some column c were to contain $d + 1 - \alpha$ entries that are not private, for each of $d+1-\alpha$ such rows choose a column that contains the same symbol as in column c. Let X be the set of at most $d + 1 - \alpha$ columns so chosen. There could be at most $\delta - 1$ rows separating $\{c\}$ from X, which cannot arise. Consequently every column of A contains at least δ private entries, and at most $d - \alpha$ that are not private. Row i employs v_i symbols and hence contains at least $k - v_i + 1$ entries that are not private. It follows that $(d-\alpha)k \geq \sum_{i=1}^{d+\delta-\alpha}(k - v_i + 1)$. Hence $\sum_{i=1}^{d+\delta-\alpha}(v_i - 1) \geq \delta k$ and the bound follows. □

When $\delta = 1$ and N is larger, Blackburn [6] partitions the N rows into d classes; then when the largest class has r rows in it, he amalgamates all rows in the class into a single row on v^r symbols. He employs a version of Lemma 5, using $\delta = 1$ and not exploiting heterogeneity, to obtain the upper bound on k already mentioned. Our heterogeneous bound underlies an improvement in the upper bound in some situations. In particular, in the example given before, an $\mathsf{SHF}_1(13; k, 6, \{1, 2\})$ must have $k \leq 326590$. Unfortunately, although the amalgamation strategy cannot reduce a separation $\delta \geq 2$ to zero, it can nonetheless reduce it to 1. Hence Lemma 5 does not lead to an effective upper bound on k as a function of N when $\delta > 1$. Despite this, Lemma 5 implies that the upper bounds on k match the lower bounds found computationally for the entries in Table 2 when $N = 2 + \delta - 1$, showing their optimality.

Proceeding to the next diagonal, when $N = d + \delta$, we employ a general observation: Whenever there exists an $\mathsf{SHF}_{\delta+1}(N; k, v, \{1, d\})$, one can delete any of the N rows to produce an $\mathsf{SHF}_{\delta}(N-1; k, v, \{1, d\})$. An elementary argument shows that $k \leq v^2$ in an $\mathsf{SHF}_1(d+1; k, v, \{1, d\})$ when $d \leq v$, and hence this upper bound on k extends to $\mathsf{SHF}_{\delta}(d+\delta; k, v, \{1, d\})$. Equality is met if and only if there exist $d + \delta - 2$ mutually orthogonal latin squares of side v (via their equivalence with "$(d+\delta)$-nets", see [5]); we omit the proof here. The non-existence of two orthogonal latin squares of side 6 explains in part the entries on this diagonal in Table 2.

For few rows, these observations indicate that the SHF s found in Table 2 are optimal, or nearly so. We do not anticipate that the numbers of columns given are optimal for larger numbers of rows, but they provide explicit solutions that are better than known general lower bounds, and often substantially better.

6 Concluding Remarks

Certain separating hash families, the frameproof codes, can be used to produce detecting arrays for main effects supporting larger separation (to cope with outlier and missing test results) and corroboration (to permit fusion of some levels). Although such SHFs have been extensively researched for index one, the generalization to larger indices is not well studied. Because we require explicit presentations of detecting arrays for testing applications, we examine constructions for SHFs for small indices, and demonstrate that a randomized algorithm can be used to provide useful detecting arrays.

Acknowledgements. This work is supported in part by the U.S. National Science Foundation grants #1421058 and #1813729, and in part by the Software Test & Analysis Techniques for Automated Software Test program by OPNAV N-84, U.S. Navy. Thanks to Randy Compton, Ryan Dougherty, Erin Lanus, and Stephen Seidel for helpful discussions. Thanks also to three very helpful reviewers.

References

1. Aldaco, A.N., Colbourn, C.J., Syrotiuk, V.R.: Locating arrays: a new experimental design for screening complex engineered systems. SIGOPS Oper. Syst. Rev. **49**(1), 31–40 (2015)
2. Alon, N., Cohen, G., Krivelevich, M., Litsyn, S.: Generalized hashing and parent-identifying codes. J. Comb. Theory Ser. A **104**, 207–215 (2003)
3. Alon, N., Spencer, J.H.: The Probabilistic Method. Wiley, Hoboken (2008)
4. Barg, A., Cohen, G., Encheva, S., Kabatiansky, G., Zémor, G.: A hypergraph approach to the identifying parent property: the case of multiple parents. SIAM J. Discret. Math. **14**, 423–431 (2001)
5. Beth, T., Jungnickel, D., Lenz, H.: Design Theory. Encyclopedia of Mathematics and its Applications, vol. 69, 2nd edn. Cambridge University Press, Cambridge (1999)
6. Blackburn, S.R.: Frameproof codes. SIAM J. Discret. Math. **16**(3), 499–510 (2003)

7. Bryce, R.C., Colbourn, C.J.: The density algorithm for pairwise interaction testing. Softw. Test. Verif. Reliab. **17**, 159–182 (2007)
8. Bryce, R.C., Colbourn, C.J.: A density-based greedy algorithm for higher strength covering arrays. Softw. Test. Verif. Reliab. **19**, 37–53 (2009)
9. Chang, Y., Colbourn, C.J., Gowty, A., Horsley, D., Zhou, J.: New bounds on the maximum size of Sperner partition systems (2018)
10. Colbourn, C.J.: Covering arrays and hash families. In: Information Security and Related Combinatorics, pp. 99–136. NATO Peace and Information Security, IOS Press (2011)
11. Colbourn, C.J., Kéri, G., Rivas Soriano, P.P., Schlage-Puchta, J.C.: Covering and radius-covering arrays: constructions and classification. Discret. Appl. Math. **158**, 1158–1190 (2010)
12. Colbourn, C.J., Lanus, E., Sarkar, K.: Asymptotic and constructive methods for covering perfect hash families and covering arrays. Des. Codes Cryptogr. **86**, 907–937 (2018)
13. Colbourn, C.J., Martirosyan, S.S., Mullen, G.L., Shasha, D.E., Sherwood, G.B., Yucas, J.L.: Products of mixed covering arrays of strength two. J. Comb. Des. **14**, 124–138 (2006)
14. Colbourn, C.J., McClary, D.W.: Locating and detecting arrays for interaction faults. J. Comb. Optim. **15**, 17–48 (2008)
15. Colbourn, C.J., Syrotiuk, V.R.: Coverage, location, detection, and measurement. In: 2016 IEEE Ninth International Conference on Software Testing, Verification and Validation Workshops (ICSTW), pp. 19–25. IEEE Press (2016)
16. Colbourn, C.J., Syrotiuk, V.R.: On a combinatorial framework for fault characterization. Math. Comput. Sci. **12**(4), 429–451 (2018)
17. Colbourn, C.J., Torres-Jiménez, J.: Heterogeneous hash families and covering arrays. Contemp. Math. **523**, 3–15 (2010)
18. Compton, R., Mehari, M.T., Colbourn, C.J., De Poorter, E., Syrotiuk, V.R.: Screening interacting factors in a wireless network testbed using locating arrays. In: IEEE INFOCOM International Workshop on Computer and Networking Experimental Research Using Testbeds (CNERT) (2016)
19. Erdős, P., Lovász, L.: Problems and results on 3-chromatic hypergraphs and some related questions. In: Infinite and Finite Sets, pp. 609–627. North-Holland, Amsterdam (1975)
20. Johnson, D.S.: Approximation algorithms for combinatorial problems. J. Comput. Syst. Sci. **9**, 256–278 (1974)
21. Kuhn, D.R., Kacker, R., Lei, Y.: Introduction to Combinatorial Testing. CRC Press, Boca Raton (2013)
22. Laaksonen, A., Östergård, P.R.J.: New lower bounds on error-correcting ternary, quaternary and quinary codes. In: Barbero, Á.I., Skachek, V., Ytrehus, Ø. (eds.) ICMCTA 2017. LNCS, vol. 10495, pp. 228–237. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66278-7_19
23. Lanus, E., Colbourn, C.J., Montgomery, D.C.: Partitioned search with column resampling for locating array construction. In: 2019 IEEE Ninth International Conference on Software Testing, Verification and Validation Workshops (ICSTW). IEEE Press (2019, to appear)
24. Li, P.C., Meagher, K.: Sperner partition systems. J. Comb. Des. **21**(7), 267–279 (2013)
25. Lovász, L.: On the ratio of optimal integral and fractional covers. Discret. Math. **13**(4), 383–390 (1975)

26. MacWilliams, F.J., Sloane, N.J.A.: The Theory of Error-Correcting Codes. North-Holland Publishing Co., Amsterdam (1977)
27. Martínez, C., Moura, L., Panario, D., Stevens, B.: Locating errors using ELAs, covering arrays, and adaptive testing algorithms. SIAM J. Discret. Math. **23**, 1776–1799 (2009)
28. Meagher, K., Moura, L., Stevens, B.: A Sperner-type theorem for set-partition systems. Electron. J. Comb. **12**, 20 (2005)
29. Mehlhorn, K.: Data Structures and Algorithms 1: Sorting and Searching. Springer, Berlin (1984). https://doi.org/10.1007/978-3-642-69672-5
30. Moser, R.A., Tardos, G.: A constructive proof of the general Lovász local lemma. J. ACM **57**(2), 11 (2010)
31. Nie, C., Leung, H.: A survey of combinatorial testing. ACM Comput. Surv. **43**(2), 11 (2011)
32. Sarkar, P., Stinson, D.R.: Frameproof and IPP codes. In: Rangan, C.P., Ding, C. (eds.) INDOCRYPT 2001. LNCS, vol. 2247, pp. 117–126. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45311-3_12
33. Seidel, S.A., Mehari, M.T., Colbourn, C.J., De Poorter, E., Moerman, I., Syrotiuk, V.R.: Analysis of large-scale experimental data from wireless networks. In: IEEE INFOCOM International Workshop on Computer and Networking Experimental Research Using Testbeds (CNERT), pp. 535–540 (2018)
34. Seidel, S.A., Sarkar, K., Colbourn, C.J., Syrotiuk, V.R.: Separating interaction effects using locating and detecting arrays. In: Iliopoulos, C., Leong, H.W., Sung, W.-K. (eds.) IWOCA 2018. LNCS, vol. 10979, pp. 349–360. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-94667-2_29
35. Shangguan, C., Wang, X., Ge, G., Miao, Y.: New bounds for frameproof codes. IEEE Trans. Inf. Theory **63**(11), 7247–7252 (2017)
36. Shi, C., Tang, Y., Yin, J.: The equivalence between optimal detecting arrays and super-simple OAs. Des. Codes Cryptogr. **62**(2), 131–142 (2012)
37. Shi, C., Tang, Y., Yin, J.: Optimum mixed level detecting arrays. Ann. Stat. **42**(4), 1546–1563 (2014)
38. Shi, C., Wang, C.M.: Optimum detecting arrays for independent interaction faults. Acta Math. Sin. (Engl. Ser.) **32**(2), 199–212 (2016)
39. Staddon, J.N., Stinson, D.R., Wei, R.: Combinatorial properties of frameproof and traceability codes. IEEE Trans. Inf. Theory **47**, 1042–1049 (2001)
40. Stein, S.K.: Two combinatorial covering theorems. J. Comb. Theory Ser. A **16**, 391–397 (1974)
41. Stinson, D.R., Van Trung, T., Wei, R.: Secure frameproof codes, key distribution patterns, group testing algorithms and related structures. J. Stat. Plan. Inference **86**, 595–617 (2000)
42. Stinson, D.R., Wei, R., Chen, K.: On generalized separating hash families. J. Comb. Theory (A) **115**, 105–120 (2008)
43. Tang, Y., Yin, J.X.: Detecting arrays and their optimality. Acta Math. Sin. (Engl. Ser.) **27**(12), 2309–2318 (2011)
44. Wigmore, J.H.: Required numbers of witnesses: a brief history of the numerical system in England. Harv. Law Rev. **15**(2), 83–108 (1901)