DOI 10.4171/8ECM/19

Detecting arrays for effects of single factors

Charles J. Colbourn and Violet R. Syrotiuk

Abstract. Determining correctness and performance for complex engineered systems necessitates testing the system to determine how its behavior is impacted by factors and interactions among them. Of particular concern is to determine which settings of single factors (main effects) impact the behavior 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 [33,45] addresses the design and analysis of test suites in order to evaluate correctness (and, more generally, performance) of complex engineered systems. We first 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 $1 \le i \le k$. 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 $\{(i_j, \sigma_{i_j}) : 1 \le j \le t\}$ 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 $1 \le i \le N$ and $1 \le j \le k$. 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.

2020 Mathematics Subject Classification. Primary 05B40; Secondary 05B15, 68R05, 62K99.
Keywords. Combinatorial testing, detecting array, covering array, orthogonal array, error-correcting code.

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 $A = (a_{i,j})$ be a test suite of size N and type $(s_1, \ldots, s_k)$. Let $T = \{(i_j, \sigma_{i_j}) : 1 \le j \le t\}$ be a t-way interaction. Then $\rho_A(T)$ denotes the set $\{r : a_{r,i_j} = \sigma_{i_j}, 1 \le j \le t\}$ of rows of A in which the interaction is covered. A t-way interaction T must have $|\rho_A(T)| \ge 1$ 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 [1,24,50], 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 [17], and later in [19,40]. Secondly, one or more tests may fail to execute correctly, and yield no response or yield outlier responses. To mitigate this, Seidel et al. [51] impose stronger "separation" requirements on the test suite.

Extending definitions in [17,19,51], 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 [40].

An $N \times k$ array A of type $(s_1, \ldots, s_k)$ is $(\bar{d}, t, \delta)$-*locating* if $|\rho_A(\mathcal{R}) \cap \rho_A(\mathcal{T})| < \delta \Leftrightarrow \mathcal{R} = \mathcal{T}$ whenever $\mathcal{R}, \mathcal{T} \subseteq \mathcal{I}_t$, $|\mathcal{R}| \le d$, and $|\mathcal{T}| \le d$. In this paper, we enforce a condition that is stronger [17]. 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 called 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 [51], and the definition in [17] 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: a *covering array* $\mathsf{CA}_\delta(N; t, k, (s_1, \ldots, s_k))$ is equivalent to a $\mathsf{DA}_\delta(N; 0, t, k, (s_1, \ldots, s_k))$. The simpler notation $\mathsf{CA}_\delta(N; t, k, v)$ is employed when it is uniform.

An *orthogonal array* $\mathsf{OA}_\delta(N; t, k, v)$, A, enforces the stronger condition that for every t-way interaction T, we have that $|\rho_A(T)| = \delta$. Orthogonal arrays are the subject of a vast literature [28], in part because of their applications in experimental design and error-correcting codes. Covering arrays have also been much more extensively studied [13, 33, 45] than 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., [15]).

Supporting fusion for detecting arrays motivates the definition of a further parameter [20]. When applied to detecting arrays with $d \geq 1$, fusion may reduce the number of witnesses. Increasing the separation cannot overcome this problem, unless the number of *distinct* witnesses increases.

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 $\{i_1, \ldots, i_t\} \cap \{c_1, \ldots, c_d\} = \emptyset$. Define a set system on the ground set $\{(c, f) : c \in C, f \in S_c\}$ by

$$\mathcal{S}_{A,T,C} = \left\{\left\{(c_1, v_1), \ldots, (c_d, v_d)\right\} : T \cup \left\{(c_1, v_1), \ldots, (c_d, v_d)\right\} \text{ is covered in } A\right\}.$$

Lemma 1.1. *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 the ground set of $\mathcal{S}_{A,T,C}$ that is disjoint from 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 1.1 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 $\mathsf{DA}_\delta(N; d, t, k, (s_1, \ldots, s_k))$ to exist is that $d < \min(s_i : 1 \le i \le k)$ (see also [17]).

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 the ground set of $\mathcal{S}_{A,T,C}$ that is disjoint from fewer than δ sets in $\mathcal{S}_{A,T,C}$ satisfies $|X| \ge d + 1$. Then (T, C) has *corroboration* s in A. When every choice of (T, C) has corroboration (at least) s in a $\mathsf{DA}_\delta(N; d, t, k, (s_1, \ldots, s_k))$, it has *corroboration* s. We extend the notation as $\mathsf{DA}_\delta(N; d, t, k, (s_1, \ldots, s_k), s)$ to include corroboration s as a parameter.

In this paper, we focus on detecting arrays for single factors, or main effects. In Section 2, we briefly summarize what is known about the construction of detecting arrays. In Section 3, we define and construct certain arrays, perfect and separating hash families, which are subsequently used to construct detecting arrays with different values of separation and corroboration. In Section 4, we unify a number of constructions for detecting arrays that employ hash families by providing a general column replacement method, and present the small detecting arrays needed. In Section 5, we examine the consequences of applying the general construction.

2. Covering arrays and Sperner partition systems

As observed in [17], one method to construct detecting arrays is to use covering arrays of higher strength. The following records consequences for separation and corroboration.

Lemma 2.1. *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 $1 \le d < \min(t, v)$.

Proof. Let A be a $\mathsf{CA}_\lambda(N; t, k, v)$. Let d satisfy $1 \le d < \min(t, v)$. 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 $e_1, \ldots, e_d$ (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 [60], uniform $(1,t)$-detecting arrays with separation 1, corroboration 1, and few factors are studied. This was extended in [53, 55] to (d,t)-detecting arrays, and further to mixed detecting arrays in [54]. Each of these focusses 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 [51]. The algorithms include randomized methods based on the Stein–Lovász–Johnson framework [31, 37, 57], and derandomized algorithms using conditional expectations (as in [10, 11]); randomized methods based on the Lovász local lemma [3, 25] and derandomizations using Moser–Tardos resampling [44] (as in [16]). 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 [35].

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, Moura, and Stevens [42] 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 [36, 42], 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 [12, 26], lower and upper bounds 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.

3. Perfect and separating hash families

3.1. Separating hash families: Definitions

An $(N; k, v)$-*hash family* is an $N \times k$ array on v symbols. Colbourn and Torres-Jiménez [23] 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 $1 \le i \le N$.

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, 48, 56]; and see [5] for the similar notion of "partially hashing." The notation $\mathsf{SHHF}_\lambda(N; k, (v_1, \ldots, v_N), \{w_1, w_2, \ldots, w_t\})$ is used for heterogeneous arrays. We remark that an $\mathsf{SHF}_1(N; k, v, \{1, d\})$ is a *frameproof code* (see, for example, [56,59]), a type of *strong* separating hash family [47,48].

When $w_1 = \cdots = w_k = 1$, we recover a more widely studied class of arrays. 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 consisting of distinct symbols. Mehlhorn [43] introduced perfect hash families, and they have subsequently found many applications in combinatorial constructions [58]. The definition for PHF extends naturally to perfect *heterogeneous* hash families; we use the notation $\mathsf{PHHF}_\lambda(N; k, (v_1, \ldots, v_N), t)$.

We employ a further extension that incorporates two types of symbols, as proposed in [14]. Let $\Sigma_v = \{0, \ldots, v-1\}$. An $\mathsf{SHHF}_\lambda(N; k, (v_1, \ldots, v_N), \{1, d^\circ\})$ is an $N \times k$ array for which

(1) the jth row contains symbols from $\Sigma_{v_j} \cup \{\circ\}$;

(2) for every $C_1, C_2 \subseteq \{1, 2, \ldots, k\}$ with $|C_1| = 1, |C_2| = d$, and $C_1 \cap C_2 = \emptyset$, there are λ rows, indexed by $\{\rho_1, \ldots, \rho_\lambda\}$, so that for each ρ_j, the set S of symbols appearing in columns of C_2 in row ρ_j is a subset of $\Sigma_{v_{\rho_j}} \cup \{\circ\}$, and the symbol in the column of C_1 in row ρ_j belongs to $\Sigma_{v_{\rho_j}} \setminus S$.

When the array is homogeneous, the notation $\mathsf{SHF}_\lambda(N; k, v, \{1, d^\circ\})$ is used. Every $\mathsf{SHF}_\lambda(N; k, v, \{1, d\})$ is an $\mathsf{SHF}_\lambda(N; k, v, \{1, d^\circ\})$, and by treating $\circ$ as a symbol like the rest, every $\mathsf{SHF}_\lambda(N; k, v, \{1, d^\circ\})$ is an $\mathsf{SHF}_\lambda(N; k, v+1, \{1, d\})$.

3.2. Separating hash families: Some constructions

Existence of SHFs is well studied for $\delta = 1$ (see [52] and references therein), but these appear not to have been studied when $\delta > 1$. We employ a number of standard ideas to construct SHHFs from other SHHFs in the following lemma.

Lemma 3.1. *Suppose that an* $\mathsf{SHHF}_\delta(N; k, \{v_1, \ldots, v_N\}, \{1, d^\circ\})$ *exists, in which some symbol in the* j*th row appears in* c *columns. Then*

(1) *an* $\mathsf{SHHF}_\delta(N; k, \{v_1, \ldots, v_{j-1}, v_j + 1, v_{j+1}, \ldots, v_N\}, \{1, d^\circ\})$ *exists;*

(2) *when* $\delta > 1$, *an* $\mathsf{SHHF}_{\delta-1}(N - 1; k, \{v_1, \ldots, v_{j-1}, v_{j+1}, \ldots, v_N\}, \{1, d^\circ\})$ *exists;*

(3) *when* $c = k$, *an* $\mathsf{SHHF}_{\delta-1}(N - 1; k, \{v_1, \ldots, v_{j-1}, v_{j+1}, \ldots, v_N\}, \{1, d^\circ\})$ *exists;*

(4) *an* $\mathsf{SHHF}_\delta(N; k - c, \{v_1, \ldots, v_{j-1}, v_j - 1, v_{j+1}, \ldots, v_N\}, \{1, d^\circ\})$ *exists;*

(5) *an* $\mathsf{SHHF}_\delta(N - 1; c, \{v_1, \ldots, v_{j-1}, v_{j+1}, \ldots, v_N\}, \{1, d^\circ\})$ *exists;*

(6) *an* $\mathsf{SHHF}_\delta(N; k + 1, \{v_1 + 1, \ldots, v_j + 1, \ldots, v_N + 1\}, \{1, d^\circ\})$ *exists;*

(7) *if an* $\mathsf{SHHF}_{\delta'}(M; k, \{w_1, \ldots, w_M\}, \{1, d^\circ\})$ *also exists, then an* $\mathsf{SHHF}_{\delta+\delta'}(N + M; k, \{v_1, \ldots, v_N, w_1, \ldots, w_M\}, \{1, d^\circ\})$ *exists.*

Proof. Let A be the stated SHHF. Then (1) holds because permitting an additional symbol in row j does not require its use. Deleting any row of A can reduce its index by at most one, so (2) holds. When $c = k$ (and in particular when $v_j = 1$), row j accomplishes no separations in A, so (3) holds. To establish (4), choose a symbol that occurs c times in row j, and delete all columns containing that symbol in row j. For (5), choose a symbol that occurs c times in row j, and delete all columns containing any other symbol in row j; then apply (3). For (6), add a column to A that, in each row, contains a symbol not appearing in A. For (7), vertically juxtapose the two arrays. ∎

Stinson, Wei, and Chen [59] 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 the following.

Theorem 3.2 ([59]). *An* $\mathsf{SHF}_1(N; k, v, \{1, 2\})$ *exists for* $k = \lceil \frac{1}{2}(\frac{v^2}{2v-1})^{\frac{N}{2}} \rceil$.

Unfortunately, Theorem 3.2 does not provide competitive lower bounds on the achievable numbers of columns in our applications. We therefore develop a number of other constructions.

Lemma 3.3. *An* $\mathsf{SHF}_1(N + 1; N \cdot v^N, v, \{1, 1^\circ\})$ *exists whenever* $N \geq 1$ *and* $v \geq 1$.

Proof. Form an $N \times v^N$ array A consisting of all distinct column vectors from Σ_v^N. For $0 \leq i \leq N$, form an $(N + 1) \times v^N$ array A_i by inserting a row consisting entirely of $\circ$ after row i when $1 \leq i \leq N$, or before row 1 when $i = 0$. Horizontally juxtapose $A_0, \ldots, A_N$ to form B, the $\mathsf{SHF}_1(N + 1; N \cdot v^N, v, \{1, 1^\circ\})$. The verification is routine, as follows. Consider two distinct columns γ and c of B. When γ and c are from the same A_i, the two columns disagree in at least one row because such a row appears

in A. On the other hand, when γ is in A_i and c is in A_j with $j \neq i$, row $j+1$ of the resulting array contains $\circ$ in column c, but contains an element of Σ_v in column γ, so the desired separation is ensured. ∎

We consider cases with "few" rows next.

Lemma 3.4. *Let $d \geq 2$, $\delta \geq 1$, and $d > \alpha \geq 1$. Then*

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

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

Proof. Let A be an $\mathsf{SHHF}_\delta(d+\delta-\alpha; k, (v_1, \ldots, v_{d+\delta-\alpha}), \{1, d\})$. 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. Hence $(d-\alpha)k \geq \sum_{i=1}^{d+\delta-\alpha}(k - v_i + 1)$. Hence $\sum_{i=1}^{d+\delta-\alpha}(v_i - 1) \geq \delta k$ and the bound follows. ∎

When $\delta = 1$, Blackburn [6] establishes that an $\mathsf{SHF}_1(N; k, v, \{1, d\})$ can exist only when $k \leq dv^{\lceil \frac{N}{d} \rceil} - d$. To establish this, partition the N rows into d classes; then when the largest class has r rows in it, amalgamate all rows in the class into a single row on v^r symbols. He employs a version of Lemma 3.4, 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. Unfortunately, although the amalgamation strategy cannot reduce a separation $\delta \geq 2$ to zero, it can nonetheless reduce it to 1. Hence Lemma 3.4 does not lead to an effective upper bound on k as a function of N when $\delta > 1$.

For certain SHHFs, this bound can be met by generalizing a well-known construction for perfect hash families [6, 41, 61]; indeed, it can be extended to employ $\circ$ symbols.

Lemma 3.5. *Let $\delta \geq 1$ and $d > \alpha \geq 1$. Let $v_1 = \cdots = v_\delta \geq v_{\delta+1} \geq \cdots \geq v_{d+\delta-\alpha}$. Then an* $\mathsf{SHHF}_\delta(d+\delta-\alpha; \max(v_\delta, \lfloor \frac{1}{\delta} \sum_{j=1}^{d+\delta-\alpha} v_j \rfloor), \{v_1, \ldots, v_{d+\delta-\alpha}\}, \{1, d^\circ\})$ *exists.*

Proof. If $k = v_\delta$, form δ rows that contain only private entries, and adjoin $d-\alpha$ arbitrary rows to produce the SHHF. Henceforth, we suppose that $k > v_\delta = v_1$. In a

$(d + \delta - \alpha) \times k$ array, place $\circ$ entries so that (1) each of the k columns contains exactly $d - \alpha$ $\circ$ entries, and (2) for $1 \le j \le d + \delta - \alpha$, row j contains at least $k - v_j$ $\circ$ entries. When this can be done, in each row fill the remaining entries with distinct symbols. Then no matter how $C_1 = \{\gamma\}$ is chosen, there are δ rows in which γ contains a private entry, so the array is an $\mathsf{SHHF}_\delta(d+\delta-\alpha; k, \{v_1, \ldots, v_{d+\delta-\alpha}\}, \{1, d^\circ\})$.

We next determine the values of k for which this is possible. Because each column contains $d-\alpha$ entries equal to $\circ$, the array contains $k(d-\alpha)$ entries equal to $\circ$. On the other hand, row j contains at least $k-v_j$ entries equal to $\circ$. Hence $\sum_{j=1}^{d+\delta-\alpha}(k-v_j) \le k(d-\alpha)$, leading to the stated bound on k. It remains to ensure that the $\circ$ entries can be placed to meet the row and column constraints simultaneously; this follows from classical work on $\{0, 1\}$-matrices with fixed row and column sums ([9], for example). ∎

For larger numbers of rows, the elementary constructions of Lemma 3.1 are useful, but they typically decrease the number of columns or the index. Hence further constructions are needed. One addition method follows.

Lemma 3.6. *If an* $\mathsf{SHF}(N; k, v, \{1, d^\circ\}; \delta)$ *and an* $\mathsf{SHF}(N'; k', v, \{1, d^\circ\}; \delta)$ *both exist, then an* $\mathsf{SHF}(N + N'; k + k', v, \{1, d^\circ\}; \delta)$ *exists.*

Proof. Let A be an $\mathsf{SHF}(N; k, v, \{1, d^\circ\}; \delta)$ and let B be an $\mathsf{SHF}(N'; k', v, \{1, d^\circ\}; \delta)$. Let $E_{n\times\kappa}$ denote an $n \times \kappa$ array in which every entry is $\circ$. Then the array $\begin{pmatrix} A & E_{N\times k'} \\ E_{N'\times k} & B \end{pmatrix}$ is an $\mathsf{SHF}(N + N'; k + k', v, \{1, d^\circ\}; \delta)$. The verification is routine. ∎

To yield a larger increase in the number of columns, we also employ a *composition* [4] or *column replacement* [13] method.

Theorem 3.7. *Suppose that an* $\mathsf{SHHF}_\delta(N; k, \{k_1, \ldots, k_N\}, \{1, d\})$ *exists, Further suppose that an* $\mathsf{SHHF}_\beta(M_i; k_i, \{v_{i1}, \ldots, v_{iM_i}\}, \{1, d^\circ\})$ *exists for each* $1 \le i \le N$. *Then an* $\mathsf{SHHF}_{\delta\beta}(\sum_{i=1}^N M_i; k, \{v_{ij} : 1 \le i \le N, 1 \le j \le M_i\}, \{1, d^\circ\})$ *exists.*

Proof. Let A be the $\mathsf{SHHF}_\delta(N; k, \{k_1, \ldots, k_N\}, \{1, d\})$. Form an arbitrary bijection between the k_i *symbols* permitted in row i of A and the k_i *columns* of B_i, the $\mathsf{SHHF}_\beta(M_i; k_i, \{v_{i1}, \ldots, v_{iM_i}\}, \{1, d^\circ\})$. Replace each symbol of A by its associated column in B_i to form an array D_i. Vertically juxtapose $D_1, \ldots, D_N$ to form D. Then D has $\sum_{i=1}^N M_i$ rows and k columns; the largest numbers of symbols that can appear in its rows is given by $\{v_{ij} : 1 \le i \le N, 1 \le j \le M_i\}$. Now consider an arbitrary column γ and a set C_2 of d columns of D not containing γ. Column γ is separated from C_2 in δ rows of A, say $\rho_1, \ldots, \rho_\delta$. Consider a particular such row, ρ_j. Suppose that A contains symbol ν in column γ; let S be the set of symbols appearing in row ρ_j in columns of C_2. Then S does not contain ν, and column ν is separated from all columns in S in β rows of B_{ρ_j} that do not contain $\circ$ in column ν. But then in D_{ρ_j},

there are β rows in which column γ does not contain $\circ$, and the symbol in column γ is not the same as in any column of C_2. This establishes that D is the desired SHHF. ∎

Restricting to SHFs, an $\mathsf{SHF}_\delta(N;k,\kappa,\{1,d\})$ and an $\mathsf{SHF}_\beta(M;\kappa,v,\{1,d^\circ\})$ yield an $\mathsf{SHF}_{\delta\beta}(NM;k,v,\{1,d^\circ\})$. Theorem 3.7 is particularly useful because the array in which replacements are made is a separating, rather than a perfect, hash family. It is also effective because the construction of v-ary SHHFs can employ SHHFs with much larger alphabets.

3.3. Separating hash families: Codes for $d = 1$

Here we consider the case when $d = 1$, i.e., the case of perfect hash families. A v-ary code C of *length* N is a subset of Σ_v^n. (See [38,39] for definitions in coding theory.) Each $c \in C$ is a *codeword*. The *size* of C is the number $|C|$ of codewords, and its *minimum distance* is the smallest Hamming distance between any two distinct codewords. A v-ary code with length N, size k, and minimum distance δ is an $(N,k,\delta)_v$ *code*. Let $A_v(N,d)$ denote the maximum size of an $(N,k,\delta)_v$ code. Treating columns of a $\mathsf{PHF}_\delta(N;k,v,2)$ as codewords, one obtains an $(N,k,\delta)_v$ code; the converse also holds. More generally, we have the following lemma.

Lemma 3.8. *An* $\mathsf{SHF}_\delta(N;k,v,\{1,d\})$ *exists whenever* $A_v(N, N-\lfloor\frac{N-\delta}{d}\rfloor) \geq k$.

Proof. Treat codewords of an $(N, k, N-\lfloor\frac{N-\delta}{d}\rfloor)_v$ code as columns of an $N \times k$ array on v symbols. Consider any set $C_1 = \{\gamma\}$ of one column, and any disjoint set C_2 of d columns. For each $\gamma' \in C_2$, there can be at most $\lfloor\frac{N-\delta}{d}\rfloor$ rows in which γ and γ' share a symbol. Then there can be at most $N-\delta$ rows in which γ shares a symbol with one or more columns of C_2, and hence at least δ rows in which γ shares a symbol with no column of C_2. This establishes the result. ∎

For $\delta \geq 3$, the existence question for such codes is far from settled, particularly when $v > 2$. In Section 5, we use constructions of $\mathsf{SHF}_\lambda(N;k,5,\{1,d\})$s, $\mathsf{SHF}_\lambda(N;k,6,\{1,d\})$s, and $\mathsf{SHF}_\lambda(N;k,5,\{1,d^\circ\})$s. Therefore, for $d = 1$, in Table 1 we provide lower bounds on the number of columns achieved for these parameters.

To justify these entries, first consider the cases when $\delta \in \{1,2\}$. For type $\{1,1\}$ with $v \in \{5,6\}$, the values are exact and arise from the easy observations that all distinct column vectors form a code of distance 1, while all column vectors whose total weight is $0 \pmod v$ form a code of distance 2. Considering the case of type $\{1,1^\circ\}$ for $v = 5$ with $\delta \in \{1,2\}$, only one example is given for an $\mathsf{SHF}_\delta(N;k,5,\{1,1^\circ\})$ having more columns than the $\mathsf{SHF}_\delta(N;k,5,\{1,1\})$, namely the $\mathsf{SHF}_1(6;18750,5,\{1,1^\circ\})$. This SHF was initially found by computation, but Lemma 3.3 provides an easier construction.

	$\delta = 1$			$\delta = 2$			$\delta = 3$			$\delta = 4$		
x	$(5,1)$	$(5,1^\circ)$	$(6,1)$	$(5,1)$	$(5,1^\circ)$	$(6,1)$	$(5,1)$	$(5,1^\circ)$	$(6,1)$	$(5,1)$	$(5,1^\circ)$	$(6,1)$
1	25	25	36	25	25	36	25	25	34	25	25	30
2	125	125	216	125	125	216	125	125	159	125	125	146
3	625	625	1296	625	625	1296	625	625	953	263	263	819
4	3125	3125	7776	3125	3125	7776	1597	1597	5718	1225	1225	4914
5	15625	18750	46656	15625	15625	46656	7985	7985	34278	4375	4375	22719
6										17500	17500	28320

Table 1. Lower bounds on k for $\mathsf{SHF}_\delta(x+\delta; k, v, \{1, d\})$ for $v \in \{5, 6\}$ and $d \in \{1, 1^\circ\}$.

Now let us turn to $\delta \geq 3$. Most research effort for the construction of codes has concentrated on "small" values of v. Indeed, for $v = 6$ there appears to have been no systematic effort to construct 6-ary codes. However, for $v = 5$, the situation is quite different. For large values of N, typically one resorts to using linear codes, as tabulated in [27]. In addition, Bogdanova and Östergård [7] tabulate lower bounds on $A_5(N, d)$ for $N \leq 11$ obtained by standard code constructions, by computation, and certain explicit constructions [32]. Some entries were subsequently improved upon in [34]. In particular, Laaksonen and Östergård [34] show that $A_5(8, 5) \geq 165$ and $A_5(9, 5) \geq 725$; in the repository of codes associated with their paper they provide explicit solutions to establish that $A_5(8, 5) \geq 257$ and $A_5(9, 5) \geq 857$. In [7], it is shown that $A_5(7, 5) \geq 53$ and $A_5(8, 6) \geq 45$. We improve these two bounds next, obtained via computations using `cliquer` [46].

Lemma 3.9. *$A_5(7, 5) \geq 57$ and $A_5(8, 6) \geq 50$.*

Proof. We write codewords omitting the commas and parentheses. Consider the nine codewords $C_7 = \{1111111, 1242342, 1200224, 1324443, 1333020, 1420300, 1432233, 1043404, 1004032\}$. When $a_0 \cdots a_{N-1}$ is a codeword, any vector $b_0 \cdots b_{N-1}$ with $b_{i+s \bmod N} = a_i$ for some s is a cyclic shift of the codeword. The 57 distinct cyclic shifts of the codewords in C_7 form a $(7, 57, 5)_5$ code. In the same manner, the 50 cyclic shifts of $\{11214402, 11023313, 12001200, 13441344, 14330040, 10322424, 22030434, 23232323\}$ form an $(8, 50, 6)_5$ code. ∎

Known codes provide powerful constructions for $\mathsf{SHF}_\delta(N; k, 5, \{1, 1\})$s, which in turn yield lower bounds on the number of columns in $\mathsf{SHF}_\delta(N; k, 5, \{1, 1^\circ\})$s and $\mathsf{SHF}_\delta(N; k, 6, \{1, 1\})$s. We failed to find any cases with $\delta \geq 3$ in which an $\mathsf{SHF}_\delta(N; k, 5, \{1, 1^\circ\})$ has more columns than an $\mathsf{SHF}_\delta(N; k, 5, \{1, 1\})$, although we expect that this can happen for larger sizes.

We have no tables of 6-ary codes. Lemma 3.1 (6) constructs 6-ary codes from 5-ary ones, to which the remaining constructions of the lemma can be applied. In addition, we adapted the "replace-one-column–random extension" randomized algorithm from [16] in order to construct SHFs of index δ. We do not describe the method here, noting only that it is a heuristic technique that is not expected to produce optimal sizes. We also report some 6-ary codes, again found using `cliquer` [46].

Lemma 3.10. *For* $v = 6$, $A_6(13, 10) \geq 78$, $A_6(12, 9) \geq 108$, $A_6(11, 8) \geq 132$, $A_6(10, 7) \geq 186$, $A_6(9, 6) \geq 258$, $A_6(16, 12) \geq 96$, $A_6(15, 11) \geq 180$, $A_6(14, 10) \geq 546$, *and* $A_6(11, 7) \geq 660$.

Proof. Form the following sets of codewords:

N, d	Starter codewords
13,10	0001105451315
12,9	000000000000, 002322454401, 012345012345, 013415321203
11,8†	00205141502, 00541020145
10,7	0000143153, 0023442314, 0140504324, 0303030303
9,6	000214121, 000300342, 004135045, 004343154, 010232454, 025403241, 042042042
16,12†	0000414354453414
15,11†	000125304403521, 000450523325054
14,10†	00043515451534, 00103445544301, 00134204402431, 00214412003553, 00452540045254, 01014503530541, 01025245354252
11,7†	00015524122, 00133100525, 00150303051, 00224151422, 00314010413, 00342404243, 00501242105, 01035134235

When a † is shown, adjoin the codeword $a_{N-1} \cdots a_0$ whenever $a_0 \cdots a_{N-1}$ is a codeword. Form all distinct cyclic shifts (as in the proof of Lemma 3.9). Then develop each codeword under the additive action of $\mathbb{Z}_6$. ∎

3.4. Separating hash families: $d = 2$

For separations $\{1, 2\}$ and $\{1, 2^\circ\}$, we report results for $v \in \{5, 6\}$ in Table 2. To produce Table 2, we apply Lemma 3.8 to the $(n, k, \delta)_5$ and $(n, k, \delta)_6$ codes described earlier. We also employ certain linear codes, noting that a linear code with parameters $[n, k, \delta]_q$ yields an $(n, q^k, \delta)_q$ code [38, 39]. In particular, we employ linear codes with parameters $[6, 3, 4]_5$, $[10, 3, 7]_5$, $[12, 4, 8]_5$, $[15, 5, 9]_5$, $[18, 5, 11]_5$, $[15, 6, 8]_5$,

	$\delta = 1$			$\delta = 2$			$\delta = 3$			$\delta = 4$		
x	$(5,2)$	$(5,2^\circ)$	$(6,2)$	$(5,2)$	$(5,2^\circ)$	$(6,2)$	$(5,2)$	$(5,2^\circ)$	$(6,2)$	$(5,2)$	$(5,2^\circ)$	$(6,2)$
1	8	10	10	6	7	7		6			6	
2	25	25	36	25	25	34	25	25	30	25	25	27
3	34	46	51	26	36	39		26	32			28
4	125	125	162	125	125	146	57	57	132	50	50	126
5		170	202			152		62				
6	263	274	819	174	174	702	135	135	258	125	125	186
7	352	356		177	179							
8	857	868	2106	625	625	1944	625	625	660	625	625	626
9		900	2592			2106						
10	3125	3125	3564	780	780	2592			864			
11			5346	1000	1125	3564		730	1296			864
12		3645	8423	3125	3125	5184	3125	3125	3126		730	1296
13	5000	6562	14124		3645	7776					1125	1944
14	15625	15625	44172	4096	6562	11664		3645	4374	3125	3125	3126
15				5000	10125	17496	4096	6562	6562		3645	4374
16				15625	18225	42984	15625	15625	15626	15625	15625	15626

Table 2. Lower bounds on k for $\mathsf{SHF}_\delta(x+\delta; k, v, \{1, d\})$ for $v \in \{5, 6\}$ and $d \in \{2, 2^\circ\}$.

$[20, 6, 12]_5$ [27]. Lemmas 3.5 and 3.1 are applied. In order to apply Theorem 3.7, we employ SHFs on larger alphabets. The primary source of these is the following standard construction of orthogonal arrays.

Theorem 3.11. *Let q be a prime power, where $q \geq s \geq 2$, and $d \geq 1$. Then an $\mathsf{OA}_1(q^s; s, q+1, q)$ exists. Hence, in addition, for $1 \leq \delta \leq (q+1) - (s-1)d$, an $\mathsf{SHF}_\delta((s-1)d + \delta; q^s, q, \{1, d\})$ exists.*

Proof. The construction of the $\mathsf{OA}_1(q^s; s, q+1, q)$ is very well known [28, 38, 39], but we repeat it here. To form the orthogonal array A, index the q^s rows by the q^s polynomials with coefficients in $\mathbb{F}_q$ of degree less than s. Index the columns by elements of $\mathbb{F}_q \cup \{\infty\}$. In a row indexed by the polynomial $f(x) = \sum_{i=0}^{s-1} a_i x^i$, and column indexed by r, place the entry $f(r)$ when $r \in \mathbb{F}_q$, or the entry a_{s-1} when $r = \infty$. This is the required orthogonal array. To form the SHF, B, first select $R \subseteq \mathbb{F}_q \cup \{\infty\}$ with $|R| = (s-1)d + \delta$, and let A_R be the array obtained from A by including only columns whose indices are in R. Transpose A_R to form B. Then two columns can agree in at most $s-1$ rows, and so one column disagrees with each of d other columns in at least δ rows, so we have the desired SHF. ∎

Despite applying each of the constructions discussed thus far, for many parameter sets the bound obtained is weak. We also employ a heuristic computational method using random extension (as in [16]) to establish lower bounds on k for these situations. We expect that many or most of the entries can be increased, particularly when $v = 6$.

3.5. Separating hash families: $d = 3$

For separations $\{1, 3\}$ and $\{1, 3^\circ\}$, we report results for $v \in \{5, 6\}$ in Table 3. We follow the same strategy as when $d = 2$. In this case, we use linear codes with parameters $[11, 3, 8]_5$, $[22, 4, 16]_5$, $[25, 5, 17]_5$, $[29, 5, 20]_5$, $[34, 5, 24]_5$, $[16, 4, 12]_8$, and $[16, 4, 12]_9$ [27]. When the number of symbols is not a prime power, we also apply the natural extension of Theorem 3.11 to transversal designs and incomplete transversal designs (see [13], for example). A $(6, 225, 5)_{15}$ code results from the existence of four mutually orthogonal latin squares of order 15 [49], and a $(6, 98, 5)_{10}$ code results from four mutually incomplete orthogonal latin squares of order 10 with a hole of size 2 [8].

It is worthwhile to remark that in order to produce an $\mathsf{SHF}_3(N; 5^6, 5, \{1, 3\})$ one could use an $[N, 6, \Delta]_5$ linear code with $3\Delta - 2N \geq 3$. According to [27], the smallest N for which such a linear code is known has $N = 39$. Nevertheless, an $\mathsf{SHF}_3(9; 25^3, 25, \{1, 3\})$ and an $\mathsf{SHF}_1(4; 25, 5, \{1, 3\})$ both exist by Theorem 3.11, and hence an $\mathsf{SHF}_3(36; 5^6, 5, \{1, 3\})$ exists by Theorem 3.7.

4. Constructing detecting arrays from hash families

Now we return to detecting arrays. Perhaps the easiest connection with hash families is the following.

Lemma 4.1. *If an* $\mathsf{SHF}_1(N; k, v, \{1, 1\})$ *exists, a* $\mathsf{DA}_1(v(N + 1); 1, 1, k, v, 1)$ *exists.*

Proof. Form the $\mathsf{SHF}_1(N; k, v, \{1, 1\})$ on symbols Σ_v, append a row consisting of all 0s, and apply the action of the cyclic group $\mathbb{Z}_v$. To verify that this works, consider a column γ and a symbol σ, and let R be the rows in which σ appears in column γ. Let $\gamma' \neq \gamma$. Some row in R in the orbit of the all-0 row contains σ in column γ'. A different row of R contains a symbol not equal to σ in column γ', from the orbit of a row of the SHF in which columns γ and γ' contain different symbols. ∎

In [20], a second approach is explored, there called *h-inflation*, which uses an $\mathsf{SHF}_\delta(N; k, v + 1, \{1, d\})$ to make a detecting array on v symbols. In [21], yet another approach is developed for general d and general t; when $t = 1$ it employs an array that is equivalent to an $\mathsf{SHF}_\delta(N; k, v, \{1, d^\circ\})$. Rather than reviewing each approach, we develop a common generalization of all three, in the case that $t = 1$. Later we revisit these constructions.

	$\delta = 1$			$\delta = 2$			$\delta = 3$			$\delta = 4$		
x	$(5,3)$	$(5,3^\circ)$	$(6,3)$	$(5,3)$	$(5,3^\circ)$	$(6,3)$	$(5,3)$	$(5,3^\circ)$	$(6,3)$	$(5,3)$	$(5,3^\circ)$	$(6,3)$
1	8	10	10	6	7	7						
2	12	15	15	8	10	10	6	8	8	6	7	7
3	25	25	34	25	25	30	25	25	27	15	15	19
4	26	36	37		28	32			28		16	
5	29	54	54		41	41			29		20	20
6	57	67	132	50	50	126	41	41	72	25	25	64
7	64	98									27	
8		103		64	98			50			31	
9	125	125	186	125	125	132	65	98	108	65	65	78
10		150	216			144					82	82
11	200	225	324		130	216	80		144			108
12	320	405	552	200	225	552	125	125	216	125	125	192
13	512	730	730	320	405		200	225	324			216
14		760	864	512	730	730	320	405	486	200	225	324
15	625	830	1296				512	730	730	320	405	486
16		1125	1944		760	760				512	730	730
17	807	2025	2916		830	1130						
18	1270	3645	4374	625	910	1866	625	760	1033	625		
19	2032	6562	6562		2578	3092		1186	1701			
20		6567	6567	704	5141	5141		2345	2813		760	760
21		6833	6833	968	5651	5651		4700	4700		830	841
22		7515	7515	1332	6214	6214	704	5151	5151		1043	1043
23	2179	8265	8265		6833	6833	968	5655	5655		1304	1304
24	3125	9091	9091	1600	7515	7515	1332	6214	6214	704	1663	1663
25	4096	10000	10240	2560	8265	8265	1600	6833	6833	1000	2134	2134
26	5632	13000	15360	4096	9091	9091	2560	7515	7515	1600	2749	2916
27	15625	16900	32770		10000	10000	4096	8265	8265	2560	3645	4374
28		22926		5632	13000	13000		9091	9091	4096	6562	6562
29		34386	34386	7744	16900	16900		10000	10000			
30				15625	21970	32770	5632	13000	13000		7696	7696
31					32250		7744	16900	16900		10000	10000
32							10648	21970	21970		13000	13000
33							15625	30255	32770	4516	16900	16900
34										6753	21970	21970
35										10118	28563	28563
36										15625		32770

Table 3. Lower bounds on k for $\mathsf{SHF}_\delta(x+\delta; k, v, \{1, d\})$ for $v \in \{5, 6\}$ and $d \in \{3, 3^\circ\}$.

To begin, we extend the notion of detecting arrays for single factors to permit a column in which not all symbols need appear. A $\mathsf{DA}_\delta(N; d, 1, k^\circ, v)$ is an $N \times (k+1)$ array A in which the columns contain symbols from Σ_v, where the first k columns are indexed by Σ_k and the last is indexed by $\circ$, so that when $T = \{\gamma\}$ with $0 \le \gamma < k$, $|\rho_A(T) \setminus \rho_A(\mathcal{T})| < \delta \Leftrightarrow T \in \mathcal{T}$ whenever $\mathcal{T} \subseteq \mathcal{I}_1$, and $|\mathcal{T}| = d$. In plain English, we require that each of the first k columns be separated from any set of d other columns (possibly including the last) at least δ times, *but no such requirement is placed on the last column.*

Evidently, every $\mathsf{DA}_\delta(N; d, 1, k+1, v)$ is a $\mathsf{DA}_\delta(N; d, 1, k^\circ, v)$; moreover, by deleting column $k+1$, every $\mathsf{DA}_\delta(N; d, 1, k^\circ, v)$ yields a $\mathsf{DA}_\delta(N; d, 1, k, v)$. Incorporating corroboration as a parameter parallels the definitions provided at the outset.

The general construction that we use for detecting arrays for single factor effects follows.

Construction 4.2. *Suppose that there exist*

(1) *an* $\mathsf{SHHF}_\delta(N; k, (\ell_1, \ldots, \ell_N), \{1, d^\circ\})$, *and*

(2) *a* $\mathsf{DA}_\beta(M_j; d, 1, \ell_j^\circ, v, s)$ *for each* $1 \le j \le N$.

Then a $\mathsf{DA}_{\beta\delta}(\sum_{j=1}^N M_j; d, 1, k, v, s)$ *exists.*

Proof. Let the symbols of the $\mathsf{SHHF}_\delta(N; k, (\ell_1, \ldots, \ell_N), \{1, d^\circ\})$, A, in row j be $\Sigma_{\ell_j} \cup \{\circ\}$. For $1 \le j \le N$, let B_j be a $\mathsf{DA}_\beta(M_j; d, 1, \ell_j^\circ, v, s)$, in which the first ℓ_j columns are indexed by Σ_{ℓ_j}, and the last by $\circ$. (There is a natural bijection between the symbols in row j of A and the column indices of B_j.) Replace each symbol of row j of A by the corresponding column of B_j to form an $M_j \times k$ array, E_j, on v symbols, for $1 \le j \le N$. Then vertically juxtapose $E_1, \ldots, E_N$ to form E. For any column γ of E and any set C_2 of d disjoint columns of A, there are (at least) δ rows $\{\rho_1, \ldots, \rho_\delta\}$ in which column γ contains a non-$\circ$ symbol ψ, and the columns of C_2 contain symbols S with $\psi \notin S$. Let $\nu \in \Sigma_v$. For each $1 \le j \le \delta$, in E_{ρ_j} there is a set R of $r \ge d + \beta$ rows in which the column ψ (arising from symbol ψ of row ρ_j of A) contains ν so that no selection $\mathcal{T}$ of d (column,value) pairs within the columns of S have $|\rho_{E_{\rho_j}}(R) \setminus \rho_{E_{\rho_j}}(\mathcal{T})| < \beta$. This establishes the desired separation. Corroboration is limited by the number of *distinct* witnesses; each of $E_{\rho_1}, \ldots, E_{\rho_\delta}$ ensures corroboration s individually, but each may employ the same witnesses. Hence the corroboration of E is (at least) s. ∎

Using an $\mathsf{SHHF}_\delta(N; k, \kappa, \{1, d\})$ and a $\mathsf{DA}_\beta(M; d, 1, \kappa, v, s)$, the variant of Construction 4.2 enables one to use ingredient arrays not involving $\circ$.

Although we have already explored constructing the $\mathsf{SHF}_\delta(N; k, \kappa, \{1, d^\circ\})$, the effective application of Construction 4.2 requires that we establish the existence of suitable $\mathsf{DA}_\delta(M; d, 1, \kappa^\circ, v, s)$s, at least for small values of κ. We resort to one basic construction using orthogonal arrays.

Lemma 4.3. *Suppose that an* $\mathsf{OA}_1(q^2; 2, q+1, q)$ *exists. Let* $d \geq 1$, $\delta \geq 1$, *and* $s \geq 1$ *satisfy* $sd + \delta \leq q$. *Then a* $\mathsf{DA}_\delta((sd+\delta)q; 1, d, q^\circ, q, s)$ *exists.*

Proof. Let R be the $\mathsf{OA}_1(q^2; 2, q+1, q)$. Set $\phi = sd + \delta$. Choose any set L_ϕ of ϕ symbols in the last column, and delete all rows from R that contain a symbol not in L_ϕ to form an array S having ϕq rows. In the last column, each of ϕ symbols appears q times; in each of the remaining columns of S, every symbol appears precisely ϕ times. Let $T = \{(\gamma, \nu)\}$ with $0 \leq \gamma < q$ and $\nu \in \Sigma_q$. Consider the ϕ rows in $\rho_S(T)$; these rows agree in column γ *but in no other column*. No matter how fewer than s fusions are performed in at most d columns to form array A (so that ν remains a symbol in column γ), it follows that $|\rho_A(T)| \geq \phi - (s-1)d$, and moreover that $\rho_A(T)$ contains a set of at least $\phi - (s-1)d$ rows that mutually disagree on all other columns. Because $\phi - (s-1)d \geq d + \delta$, S is a $\mathsf{DA}_\delta((sd+\delta)q; 1, d, q^\circ, q, s)$. ∎

Lemma 4.3 produces arrays that need not contain all q symbols in the final column. In these cases, we obtain a $\mathsf{DA}_\delta((sd+\delta)q; 1, d, q^\circ, q, s)$ but not a $\mathsf{DA}_\delta((sd+\delta)q; 1, d, q+1, q, s)$. To obtain various DAs on $q+1$ symbols, we employ definitions and results from finite projective geometry. (For relevant background, see [29,30].) In the projective plane $PG(2,q)$, an (n,r)-*arc* is a set A of n points with at most r on a line; the largest n for which there is an (n,r)-arc in $PG(2,q)$ is denoted by $m_r(2,q)$. A t-*blocking set* in $PG(2,q)$ is a set B of points so that every line meets B in at least t points. Whenever A is an (m,n)-arc in $PG(2,q)$, $B = PG(2,q) \setminus A$ is a $(q+1-n)$-blocking set of size $q^2+q+1-m$.

Lemma 4.4. *Let* q *be a prime power. Let* $d \geq 1$, $\delta \geq 1$, *and* $s \geq 1$ *satisfy* $sd + \delta \leq q$. *Then a* $\mathsf{DA}_\delta(q^2 - m_{q-sd-\delta}(2,q); 1, d, q+1, q, s)$ *exists.*

Proof. Use an $(m, q-sd-\delta)$-arc in $PG(2,q)$ with $m = m_{q-sd-\delta}(2,q)$ to form an $(sd+\delta+1)$-blocking set of size $q^2+q+1-m$. The dual blocking set (i.e., the configuration obtained from the blocking set by interchanging points and lines) is a set of $q^2+q+1-m$ lines so that every point belongs to at least $sd+\delta+1$. Delete any point and the $q+1$ lines through it to form a set of (at least) q^2-m lines so that every remaining point belongs to at least $sd+\delta$. Use the $q+1$ deleted lines, omitting the deleted point, to form the columns of the DA.

Let $T = \{(\gamma, \nu)\}$ with $0 \leq \gamma < q+1$ and $\nu \in \Sigma_q$. Then $\rho(T)$ contains $sd+\delta$ rows that agree in column γ but in no other column. So similar arguments to those used in the proof of Lemma 4.3 show that the array is in fact a DA with the required parameters. ∎

When $q = sd + \delta$, Lemma 4.4 yields precisely q^2 rows, the entire orthogonal array. Exact values for $m_r(2,q)$ are not known in general and form the focus of much research. For our running examples with $q = 5$, however, exact values are known:

$m_0(2,5) = 0, m_1(2,5) = 1, m_2(2,5) = 6, m_3(2,5) = 11, m_4(2,5) = 16, m_5(2,5) = 25$, and $m_6(2,5) = 31$ [29, Table 25].

Lemmas 4.3 and 4.4, together with Construction 4.2, unify earlier constructions, as follows. The h-inflation developed in [20] is equivalent to applying Construction 4.2 using the DAs from Lemma 4.4 along with an $\mathsf{SHF}_\delta(N; k, q+1, \{1, d\})$. The method of [21] is equivalent *when restricted to* $t = 1$ to applying Construction 4.2 using the DAs from Lemma 4.3 along with an $\mathsf{SHF}_\delta(N; k, q, \{1, d^\circ\})$. Instead by removing the last column of the DA from Lemma 4.3 and using an $\mathsf{SHF}_\delta(N; k, q, \{1, d\})$, we recover a construction in the same vein as Lemma 4.1. However, there are important differences. First, Lemma 4.1 needs no assumption that v is a prime power. More importantly, where the application of Construction 4.2 requires a DA_1 (having at least $2q$ rows), Lemma 4.1 instead modifies the SHF by adding an all-0 row, so that instead of the DA_1 we can employ only q rows. To reconcile this apparent discrepancy, form the $\mathsf{DA}_1(2q; 1, 1, q, q, 1)$ so that it contains all q constant rows (rows in which all symbols are the same). Apply Construction 4.2 using an $\mathsf{SHF}_1(N; k, q, \{1, d\})$ to form an $2Nq \times q$ array E. The manner of construction ensures that each of the constant rows appears (at least) N times in E. Because these are only useful when $\mathcal{T}$ contains no main effects using the symbol used in T, $N-1$ copies of each of these constant rows are unnecessary and can be deleted. This recovers Lemma 4.1 (when q is a prime power), and indeed leads to a useful generalization of Construction 4.2.

By choosing symbol 0 to be in L_ϕ, Lemma 4.3 produces a $\mathsf{DA}_\delta((sd+\delta)q; 1, d, q, q, s)$ having q constant rows. Using this, we provide a construction (stated for the homogeneous case) for corroboration $s = 1$.

Construction 4.5. *If an* $\mathsf{SHF}_\delta(N; k, \kappa, \{1, d\})$ *and a* $\mathsf{DA}_\beta(M; d, 1, \kappa, v, 1)$ *having* v *constant rows exist, a* $\mathsf{DA}_{\beta\delta}(NM - (N - \beta\delta)v; d, 1, k, v, 1)$ *exists.*

Each replacement of columns of the DA into a row of the SHF yields (at least) v constant rows. Then the verification follows that of Construction 4.2, after deleting all but $\beta\delta$ copies of each constant row. We leave the details to the reader. Instead, we explore a powerful construction employing the detecting arrays of Lemma 4.3, restricting to a prime power number of symbols. A row in $\Sigma_v^{\kappa+1}$ is *nearly constant* if each of the first κ entries contains the same symbol, and the last entry is 0.

Construction 4.6. *Let* q *be a prime power. If an* $\mathsf{SHF}_\delta(N; k, q, \{1, d^\circ\})$ *exists, then when* $d + \beta \le q$, *a* $\mathsf{DA}_{\beta\delta}((N(d+\beta-1)+\delta)q; d, 1, k, q, 1)$ *exists.*

Proof. Let the symbols of the $\mathsf{SHF}_\delta(N; k, q, \{1, d^\circ\})$, A, be $\Sigma_q \cup \{\circ\}$. Form a $\mathsf{DA}_\beta((d+\beta)q; d, 1, q^\circ, q, 1)$, B, using Lemma 4.3 choosing $L_\phi = \{0, \ldots, d+\beta-1\}$. Let the first q columns of the $\mathsf{DA}_\beta(M; d, 1, q^\circ, q, 1)$ be indexed by Σ_q, and index the $(q+1)$st by $\circ$. The q rows containing 0 in the last column are nearly constant. Remove them from B to form a $(d+\beta-1)q \times (q+1)$ array B'. Replace each sym-

bol of row j of A by the corresponding column of B' to form an $(d+\beta-1)q \times k$ array, E_j, on q symbols, for $1 \le j \le N$. Form a $q \times k$ array D_1 consisting of each *constant* row. For $1 \le i \le \delta$, let D_i be a copy of D. Then vertically juxtapose $E_1, \ldots, E_N$ and $D_1, \ldots, D_\delta$ to form F.

To verify that F is the desired $\mathsf{DA}_{\beta\delta}((N(d+\beta-1)+\delta)q; d, 1, k, q, 1)$, consider $T = \{(\gamma, \nu)\}$. It is necessary and sufficient that in every column $g \neq \gamma$, and every set X of d symbols, at least $\beta\delta$ rows of F contain ν in column γ but contain no symbol of X in column g. To establish this for a specific T and g, partition the N rows of A into classes, as follows:

(1) A_1 contains the τ_1 rows in which columns γ and g contain distinct symbols from Σ_q;

(2) A_2 contains the τ_2 rows in which column γ contains a symbol from Σ_q, and column g contains $\circ$;

(3) A_3 contains the τ_3 rows in which column g contains a symbol from Σ_q, and column γ contains $\circ$;

(4) A_4 contains the τ_4 rows in which columns γ and g contain the same symbol from Σ_q;

(5) A_5 contains the τ_5 rows in which columns γ and g both contain $\circ$.

The separation requirements of the SHF ensure that $\tau_1 + \tau_2 \ge \delta$ in order to separate column γ from column g, and that $\tau_1 + \tau_3 \ge \delta$ in order to separate column g from column γ.

Next we define disjoint classes of rows of F as follows.

(1) For $1 \le j \le \min(\delta, \tau_1)$, F_j contains $(d+\beta)q$ rows of F consisting of
- the $(d+\beta-1)q$ arising from the jth row of A_1, and
- the q rows of D_j.

(2) For $1 \le j \le \delta - \tau_1$, G_j contains $(2d+2\beta-1)q$ rows of F consisting of
- the $(d+\beta-1)q$ arising from the jth row of A_2,
- the $(d+\beta-1)q$ arising from the jth row of A_3, and
- the q rows of $D_{j+\tau_1}$.

It suffices to check that in each of $F_1, \ldots, F_{\tau_1}$ and each of $G_1, \ldots, G_{\delta-\tau_1}$, at least $d+\beta$ rows have ν in column γ and distinct symbols in column g. Let us check F_j. In the $(d+\beta-1)q$ rows arising from the column replacement of B' into the jth row of A_1, each ν in column γ appears in exactly $d+\beta-1$ rows, with a different symbol in column g in each. Because nearly constant rows have been deleted to form B', none of these $d+\beta-1$ symbols is ν, so the row from D_j that is constant equal to ν provides the final symbol in column g. Initially, we proceed in the same manner

for G_j. In the $(d+\beta-1)q$ rows arising from the column replacement of B' into the jth row of A_2, each v in column γ appears in exactly $d+\beta-1$ rows; in these rows, column g contains each of $\{1,\ldots,d+\beta-1\}$. When $v \notin \{1,\ldots,d+\beta-1\}$, the all-$v$ row from $D_{j+\tau_1}$ provides the final row needed. When $v \in \{1,\ldots,d+\beta-1\}$, consider the $(d+\beta-1)q$ rows arising from the column replacement of B' into the row chosen from A_3 (these rows have not been considered before). In these rows, every $v \in \{1,\ldots,d+\beta-1\}$ appears in column γ with *every* symbol of Σ_q in column g. This completes the verification. ∎

When the DA from Lemma 4.3 is used, Construction 4.6 improves on Construction 4.5. By imposing the further condition on the SHF that the class A_4 of rows not be empty, one could eliminate some of the constant rows added. Even without this restriction, Construction 4.6 may include more constant rows than are needed in certain cases. We do not pursue this further here.

5. Consequences

Now we consider some consequences of Constructions 4.2 and 4.6 using detecting arrays from Lemmas 4.3 and 4.4, along with the SHFs tabulated in Tables 1, 2, and 3, and SHHFs produced from these by Lemma 3.1 (4). Of course we do not attempt to list all of the detecting arrays generated; instead we compare different approaches. Our interest is in constructing detecting arrays for complex engineered systems of moderate to large sizes. In Table 4, we report upper bounds on the number N of rows in a $\mathsf{DA}_\delta(N;1,d,k,5)$ (with corroboration 1) for various values of d, k, and δ.

In constructing Table 4, we apply Lemma 3.1 (7) and the observation that a DA need not have more rows than a DA having larger index but the same parameters otherwise.

The effectiveness of the methods employed in producing detecting arrays for single factor effects with many columns enables us to produce such arrays for larger systems. Although we do not expect that the arrays found have the fewest possible rows in general, it is striking how few rows suffice for large numbers of columns.

Comparing the results from Constructions 4.2 and 4.6 in Table 4, one finds that Construction 4.6 almost always yields the fewest rows. Perhaps this is no surprise, because Construction 4.6 typically succeeds in eliminating many rows using the nearly constant rows of the detecting array ingredient. Despite this, Construction 4.2 often remains competitive, because it uses hash families in which the unusual $\circ$ symbol does not appear, and which can be heterogeneous. Indeed Construction 4.2 can lead to the better result, as we illustrate next. Using an $\mathsf{SHF}_2(8;126,6,\{1,3\})$ and $\mathsf{DA}_2(25;1,3,6,5)$, Construction 4.2 yields a $\mathsf{DA}_4(200;1,3,100,5)$. Using an

		$k = 10$		$k = 100$		$k = 1000$		$k = 10000$	
d	δ	4.6,4.3	4.2,4.4	4.6,4.3	4.2,4.4	4.6,4.3	4.2,4.4	4.6,4.3	4.2,4.4
1	1	15	20	20	30	30	50	35	60
1	2	25	30	30	40	40	60	45	70
1	3	35	40	40	50	50	70	60	90
1	4	40	45	50	60	60	80	70	100
1	8	70	75	80	90	100	120	120	125
2	1	25	38	55	75	115	165	155	225
2	2	35	48	70	90	150	186	190	270
2	3	45	50	100	120	165	225	205	285
2	4	70	80	100	120	200	225	220	300
2	8	110	120	170	188	290	360	320	400
3	1	35	48	155	164	290	380	425	560
3	2	45	50	175	175	325	400	490	640
3	3	90	100	230	200	360	500	555	720
3	4	90	100	230	200	410	500	620	775
3	8	160	175	400	400	540	650	820	950

Table 4. Upper bounds on N for a $\mathsf{DA}_\delta(N; 1, d, k, 5)$. Two upper bounds are given for each. The first employs Construction 4.6 using Lemma 4.3, and SHFs of type $\{1, d^\circ\}$ with $v = 5$. The second employs Construction 4.2 using Lemma 4.4, and SHHFs of type $\{1, d\}$ with five or six symbols per row whose existence is implied by the SHF tables.

$\mathsf{SHF}_2(11; 125, 5, \{1, 3^\circ\})$ and $\mathsf{DA}_2(25; 1, 3, 5^\circ, 5)$, Construction 4.6 yields a $\mathsf{DA}_4(230; 1, 3, 100, 5)$. The hash family with 6 symbols is enough smaller than that with five symbols in addition to $\circ$ that the usual advantage of exploiting nearly constant rows is overcome. Naturally finding hash families with fewer rows might impact such comparisons. Although we do not believe that the hash families here have the fewest rows (or the most columns), we do believe that Construction 4.2 can, in certain cases, yield fewer rows than Construction 4.6.

Potential improvements in the sizes of the detecting arrays could result from finding better SHFs and SHHFs. They could also arise from a more detailed analysis of the redundant rows produced by Constructions 4.2 and 4.6; for this purpose, a post-optimization strategy from [18] may prove useful computationally, but we have not employed that here. In this paper, we applied the constructions only to DAs from Lemmas 4.3 and 4.4, which have v or $v + 1$ columns. Naturally, the same constructions can be applied to DAs having more columns (permitting the hash families to have more symbols and hence fewer rows). We expect that such an extension would be effective, given a larger collection of detecting arrays to use as ingredients.

Of most importance from the standpoint of applications is that the column replacement techniques and associated ingredients developed underlie effective and efficient methods to produce detecting arrays for the effects of single factors so that specified values of separation and corroboration can be achieved. Finally, many of the techniques developed here extend in a natural manner to detecting t-way interactions, not just the effects of single factors [21, 22].

Acknowledgments. Thanks to Yasmeen Akhtar, Randy Compton, Erin Lanus, and Stephen Seidel for helpful discussions. Special thanks to an anonymous referee for improvements in the presentation.

Funding. This research was supported by NSF grant #1813729, and by the Software Test & Analysis Techniques for Automated Software Test program by OPNAV N-84, U.S. Navy.

References

[1] A. N. Aldaco, C. J. Colbourn, and V. R. Syrotiuk, Locating arrays: A new experimental design for screening complex engineered systems. *SIGOPS Oper. Syst. Rev.* **49** (2015), no. 1, 31–40

[2] N. Alon, G. Cohen, M. Krivelevich, and S. Litsyn, Generalized hashing and parent-identifying codes. *J. Combin. Theory Ser. A* **104** (2003), no. 1, 207–215 Zbl 1036.94015 MR 2018429

[3] N. Alon and J. H. Spencer, *The Probabilistic Method. With an appendix on the life and work of Paul Erdős*. 3rd edn., Wiley-Interscience Series in Discrete Mathematics and Optimization, John Wiley & Sons, Hoboken, NJ, 2008 Zbl 1148.05001 MR 2437651

[4] M. Atici, S. S. Magliveras, D. R. Stinson, and W.-D. Wei, Some recursive constructions for perfect hash families. *J. Combin. Des.* **4** (1996), no. 5, 353–363 Zbl 0914.68087 MR 1402122

[5] A. Barg, G. Cohen, S. Encheva, G. Kabatiansky, and G. Zémor, A hypergraph approach to the identifying parent property: the case of multiple parents. *SIAM J. Discrete Math.* **14** (2001), no. 3, 423–431 Zbl 1011.94014 MR 1857594

[6] S. R. Blackburn, Frameproof codes. *SIAM J. Discrete Math.* **16** (2003), no. 3, 499–510 Zbl 1041.68063 MR 2002175

[7] G. T. Bogdanova and P. R. J. Östergård, Bounds on codes over an alphabet of five elements. *Discrete Math.* **240** (2001), no. 1-3, 13–19 Zbl 1005.94032 MR 1855043

[8] A. E. Brouwer, Four MOLS of order 10 with a hole of order 2. *J. Statist. Plann. Inference* **10** (1984), no. 2, 203–205 Zbl 0553.05022 MR 760405

[9] R. A. Brualdi, Matrices of zeros and ones with fixed row and column sum vectors. *Linear Algebra Appl.* **33** (1980), 159–231 Zbl 0448.05047 MR 585770

[10] R. C. Bryce and C. J. Colbourn, The density algorithm for pairwise interaction testing. *Software Testing, Verification, and Reliability* **17** (2007), 159–182

[11] R. C. Bryce and C. J. Colbourn, A density-based greedy algorithm for higher strength covering arrays. *Software Testing, Verification, and Reliability* **19** (2009), 37–53

[12] Y. Chang, C. J. Colbourn, A. Gowty, D. Horsley, and J. Zhou, New bounds on the maximum size of Sperner partition systems. *European J. Combin.* **90** (2020), 103165, 18 Zbl 1458.05028 MR 4125527

[13] C. J. Colbourn, Covering arrays and hash families. In *Information Security, Coding Theory and Related Combinatorics*, pp. 99–135, NATO Sci. Peace Secur. Ser. D Inf. Commun. Secur. 29, IOS, Amsterdam, 2011 Zbl 1341.68134 MR 2963127

[14] C. J. Colbourn, D. Horsley, and V. R. Syrotiuk, A hierarchical framework for recovery in compressive sensing. *Discrete Appl. Math.* **236** (2018), 96–107 Zbl 1431.94014 MR 3739778

[15] C. J. Colbourn, G. Kéri, P. P. R. Soriano, and J.-C. Schlage-Puchta, Covering and radius-covering arrays: constructions and classification. *Discrete Appl. Math.* **158** (2010), no. 11, 1158–1180 Zbl 1231.05033 MR 2629893

[16] C. J. Colbourn, E. Lanus, and K. Sarkar, Asymptotic and constructive methods for covering perfect hash families and covering arrays. *Des. Codes Cryptogr.* **86** (2018), no. 4, 907–937 Zbl 1383.05045 MR 3770276

[17] C. J. Colbourn and D. W. McClary, Locating and detecting arrays for interaction faults. *J. Comb. Optim.* **15** (2008), no. 1, 17–48 Zbl 1149.90090 MR 2375213

[18] C. J. Colbourn and P. Nayeri, Randomized post-optimization for t-restrictions. In *Information Theory, Combinatorics, and Search Theory*, pp. 597–608, Lecture Notes in Comput. Sci. 7777, Springer, Heidelberg, 2013 Zbl 1309.05034 MR 3076131

[19] C. J. Colbourn and V. R. Syrotiuk, On a combinatorial framework for fault characterization. *Math. Comput. Sci.* **12** (2018), no. 4, 429–451 Zbl 1433.68268 MR 3870157

[20] C. J. Colbourn and V. R. Syrotiuk, Detecting arrays for main effects. In *Algebraic Informatics*, pp. 112–123, Lecture Notes in Comput. Sci. 11545, Springer, Cham, 2019 Zbl 1434.68340 MR 3976191

[21] C. J. Colbourn and V. R. Syrotiuk, Covering strong separating hash families. In *Finite Fields and Their Applications*, pp. 189–198, De Gruyter Proc. Math., De Gruyter, Berlin, 2020 Zbl 1466.05019 MR 4204971

[22] C. J. Colbourn and V. R. Syrotiuk, There must be fifty ways to miss a cover. In *50 Years of Combinatorics, Graph Theory, and Computing*, pp. 319–333, Discrete Math. Appl. (Boca Raton), CRC Press, Boca Raton, FL, 2020 Zbl 1451.05190 MR 4368178

[23] C. J. Colbourn and J. Torres-Jimenez, Heterogeneous hash families and covering arrays. In *Error-Correcting Codes, Finite Geometries and Cryptography*, pp. 3–15, Contemp. Math. 523, Amer. Math. Soc., Providence, RI, 2010 Zbl 1226.05061 MR 2766009

[24] R. Compton, M. T. Mehari, C. J. Colbourn, E. De Poorter, and V. R. Syrotiuk, 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)*, IEEE Press, 2016

[25] P. Erdős and L. Lovász, Problems and results on 3-chromatic hypergraphs and some related questions. In *Infinite and Finite Sets (Colloq., Keszthely, 1973; Dedicated to P. Erdős on His 60th Birthday), Vol. II*, pp. 609–627, North-Holland, Amsterdam, 1975 Zbl 0315.05117 MR 0382050

[26] A. Gowty and D. Horsley, More constructions for Sperner partition systems. *J. Combin. Des.* **29** (2021), no. 9, 579–606 MR 4284176

[27] M. Grassl, Bounds on the minimum distance of linear codes and quantum codes. Available at http://www.codetables.de. Accessed on 2019-08-30

[28] A. S. Hedayat, N. J. A. Sloane, and J. Stufken, *Orthogonal Arrays*. Springer Ser. Statist., Springer, New York, 1999 Zbl 0935.05001 MR 1693498

[29] J. W. P. Hirschfeld and L. Storme, The packing problem in statistics, coding theory and finite projective spaces. *J. Statist. Plann. Inference* **72** (1998), no. 1-2, 355–380 Zbl 0958.51013 MR 1655203

[30] J. W. P. Hirschfeld and L. Storme, The packing problem in statistics, coding theory and finite projective spaces: update 2001. In *Finite Geometries*, pp. 201–246, Dev. Math. 3, Kluwer Acad. Publ., Dordrecht, 2001 Zbl 1025.51012 MR 2061806

[31] D. S. Johnson, Approximation algorithms for combinatorial problems. *J. Comput. System Sci.* **9** (1974), 256–278 Zbl 0296.65036 MR 449012

[32] J. G. Kalbfleisch, R. G. Stanton, and J. D. Horton, On covering sets and error-correcting codes. *J. Combinatorial Theory Ser. A* **11** (1971), 233–250 Zbl 0181.22401 MR 290860

[33] D. R. Kuhn, R. Kacker, and Y. Lei, *Introduction to Combinatorial Testing*. CRC Press, Boca Raton, FL, 2013 Zbl 1272.68004

[34] A. Laaksonen and P. R. J. Östergård, New lower bounds on error-correcting ternary, quaternary and quinary codes. In *Coding Theory and Applications*, pp. 228–237, Lecture Notes in Comput. Sci. 10495, Springer, Cham, 2017 Zbl 1429.94090 MR 3705120

[35] E. Lanus, C. J. Colbourn, and D. C. Montgomery, Partitioned search with column resampling for locating array construction. In *2019 IEEE Ninth International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pp. 214–223, IEEE Press, 2019

[36] P. C. Li and K. Meagher, Sperner partition systems. *J. Combin. Des.* **21** (2013), no. 7, 267–279 Zbl 1269.05107 MR 3055150

[37] L. Lovász, On the ratio of optimal integral and fractional covers. *Discrete Math.* **13** (1975), no. 4, 383–390 Zbl 0323.05127 MR 384578

[38] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes. I*. North-Holland Mathematical Library 16, North-Holland Publishing, Amsterdam, 1977 Zbl 0369.94008 MR 0465509

[39] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes. II*. North-Holland Mathematical Library 16, North-Holland Publishing, Amsterdam, 1977 Zbl 0369.94008 MR 0465510

[40] C. Martínez, L. Moura, D. Panario, and B. Stevens, Locating errors using ELAs, covering arrays, and adaptive testing algorithms. *SIAM J. Discrete Math.* **23** (2009/10), no. 4, 1776–1799 Zbl 1210.94126 MR 2570203

[41] S. Martirosyan and T. V. Trung, On t-covering arrays. *Des. Codes Cryptogr.* **32** (2004), no. 1-3, 323–339 Zbl 1046.05020 MR 2072336

[42] K. Meagher, L. Moura, and B. Stevens, A Sperner-type theorem for set-partition systems. *Electron. J. Combin.* **12** (2005), Note 20, 6 Zbl 1077.05097 MR 2180805

[43] K. Mehlhorn, *Data structures and algorithms. 1: Sorting and searching*. EATCS Monographs on Theoretical Computer Science, Springer, Berlin, 1984 Zbl 0556.68001 MR 756413

[44] R. A. Moser and G. Tardos, A constructive proof of the general Lovász local lemma. *J. ACM* **57** (2010), no. 2, Art. 11, 15 Zbl 1300.60024 MR 2606086

[45] C. Nie and H. Leung, A survey of combinatorial testing. *ACM Comput. Surv.* **43** (2011), no. 2, 29 Zbl 1293.68080

[46] S. Niskanen and P. R. J. Östergård, Cliquer user's guide, version 1.0. Tech. Rep. T48, Communications Laboratory, Helsinki University of Technology, Espoo, Finland, 2003

[47] X. Niu and H. Cao, Constructions and bounds for separating hash families. *Discrete Math.* **341** (2018), no. 9, 2627–2638 Zbl 1422.94033 MR 3828774

[48] P. Sarkar and D. R. Stinson, Frameproof and IPP codes. In *Progress in Cryptology—INDOCRYPT 2001 (Chennai)*, pp. 117–126, Lecture Notes in Comput. Sci. 2247, Springer, Berlin, 2001 Zbl 1011.94547 MR 1934490

[49] P. J. Schellenberg, G. H. J. Van Rees, and S. A. Vanstone, Four pairwise orthogonal Latin squares of order 15. *Ars Combin.* **6** (1978), 141–150 Zbl 0433.05014 MR 526098

[50] S. A. Seidel, M. T. Mehari, C. J. Colbourn, E. De Poorter, I. Moerman, and V. R. Syrotiuk, 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, IEEE, 2018

[51] S. A. Seidel, K. Sarkar, C. J. Colbourn, and V. R. Syrotiuk, Separating interaction effects using locating and detecting arrays. In *International Workshop on Combinatorial Algorithms*, pp. 349–360, Springer, Cham, 2018 Zbl 06932716

[52] C. Shangguan, X. Wang, G. Ge, and Y. Miao, New bounds for frameproof codes. *IEEE Trans. Inform. Theory* **63** (2017), no. 11, 7247–7252 Zbl 1390.94884 MR 3724426

[53] C. Shi, Y. Tang, and J. Yin, The equivalence between optimal detecting arrays and super-simple OAs. *Des. Codes Cryptogr.* **62** (2012), no. 2, 131–142 Zbl 1283.05045 MR 2886266

[54] C. Shi, Y. Tang, and J. Yin, Optimum mixed level detecting arrays. *Ann. Statist.* **42** (2014), no. 4, 1546–1563 Zbl 1297.62177 MR 3262460

[55] C. Shi and C. M. Wang, Optimum detecting arrays for independent interaction faults. *Acta Math. Sin. (Engl. Ser.)* **32** (2016), no. 2, 199–212 Zbl 1331.05046 MR 3441302

[56] J. N. Staddon, D. R. Stinson, and R. Wei, Combinatorial properties of frameproof and traceability codes. *IEEE Trans. Inform. Theory* **47** (2001), no. 3, 1042–1049 Zbl 1001.94032 MR 1829330

[57] S. K. Stein, Two combinatorial covering theorems. *J. Combinatorial Theory Ser. A* **16** (1974), 391–397 Zbl 0287.05002 MR 340062

[58] D. R. Stinson, T. van Trung, and R. Wei, Secure frameproof codes, key distribution patterns, group testing algorithms and related structures. *J. Statist. Plann. Inference* **86** (2000), no. 2, 595–617 Zbl 1054.94013 MR 1768292

[59] D. R. Stinson, R. Wei, and K. Chen, On generalized separating hash families. *J. Combin. Theory Ser. A* **115** (2008), no. 1, 105–120 Zbl 1131.68070 MR 2378859

[60] Y. Tang and J. X. Yin, Detecting arrays and their optimality. *Acta Math. Sin. (Engl. Ser.)* **27** (2011), no. 12, 2309–2318 Zbl 1260.05022 MR 2853789

[61] R. A. Walker II and C. J. Colbourn, Perfect Hash families: constructions and existence. *J. Math. Cryptol.* **1** (2007), no. 2, 125–150 Zbl 1128.05012 MR 2345113

Charles J. Colbourn
School of Computing and Augmented Intelligence, Arizona State University,
P.O. Box 878809, Tempe, AZ 85287, USA; colbourn@asu.edu

Violet R. Syrotiuk
School of Computing and Augmented Intelligence, Arizona State University,
P.O. Box 878809, Tempe, AZ 85287, USA; syrotiuk@asu.edu