J. Math. Crypt. **1** (2007), 125–150

DOI 10.1515 / JMC.2007.008

Perfect Hash Families: Constructions and Existence

Robert A. Walker II and Charles J. Colbourn

Communicated by Spyros S. Magliveras

Abstract. A *perfect hash family* $\mathsf{PHF}(N; k, v, t)$ is an $N \times k$ array on v symbols with $v \geq t$, in which in every $N \times t$ subarray, at least one row is comprised of distinct symbols. Perfect hash families have a wide range of applications in cryptography, particularly to secure frameproof codes, in database management, and indirectly in software interaction testing. New recursive constructions, new direct constructions, and PHFs found using tabu search are provided here. The first general tables of the best known sizes of PHFs are presented; in the process, the known direct and recursive constructions are surveyed.

Keywords. Perfect hash family, interaction testing, three-term arithmetic progression.

AMS classification. 05B15.

1 Introduction

A *perfect hash family* $\mathsf{PHF}(N; k, v, t)$ is an $N \times k$ array on v symbols, in which in every $N \times t$ subarray, at least one row is comprised of distinct symbols. Figure 1 shows a $\mathsf{PHF}(6; 12, 3, 3)$. For instance, in columns 1, 3, and 5, the first row contains `1 0 2`. An older survey on PHFs is given in [15].

$$\begin{bmatrix} 1 & 0 & 0 & 2 & 2 & 2 & 1 & 1 & 2 & 1 & 0 & 2 \\ 2 & 0 & 1 & 1 & 2 & 0 & 2 & 0 & 1 & 1 & 2 & 1 \\ 2 & 0 & 2 & 1 & 2 & 1 & 0 & 2 & 2 & 1 & 1 & 0 \\ 0 & 1 & 2 & 2 & 1 & 2 & 2 & 0 & 1 & 1 & 0 & 0 \\ 2 & 0 & 1 & 2 & 1 & 1 & 2 & 2 & 0 & 1 & 2 & 1 \\ 0 & 2 & 1 & 0 & 2 & 2 & 2 & 1 & 0 & 1 & 2 & 1 \end{bmatrix}$$

Figure 1. A $\mathsf{PHF}(6; 12, 3, 3)$

The smallest N for which a $\mathsf{PHF}(N; k, v, t)$ exists is the *perfect hash family number*, denoted $\mathsf{PHFN}(k, v, t)$.

Mehlhorn [23] defined perfect hash families as follows: A (k, v)-*hash function* is a function $h : A \to B$, where $|A| = k$ and $|B| = v$. For any given subset $X \subseteq A$, the function h is *perfect* if h is injective on X, i.e., if $h|_X$ is one-to-one. Given integers k, v, t so that $k \geq v \geq t \geq 2$, let $\mathcal{H}$ ($|\mathcal{H}| = N$) be a set of (k, v)-hash functions for which $h : A \to B$ for each $h \in \mathcal{H}$, where $|A| = k$ and $|B| = v$. Then $\mathcal{H}$ is a $\mathsf{PHF}(N; k, v, t)$ whenever, for any $X \subseteq A$ with $|X| = t$, there exists at least one $h \in \mathcal{H}$ such that $h|_X$

is one-to-one. This definition is equivalent to the array definition. Consider each row of the array to be a function h, and take $A = \{1, 2, \ldots, k\}$. Then the value in column i of the row for h is the value of $h(i)$.

Mehlhorn [23] introduced perfect hash families as an efficient tool for compact storage and fast retrieval of frequently used information, such as reserved words in programming languages or command names in interactive systems.

Stinson, Trung, and Wei [28] establish that perfect hash families, and a variation known as "separating hash families", can be used to construct separating systems, key distribution patterns, group testing algorithms, cover-free families, and secure frameproof codes. Perfect hash families have also recently found applications in broadcast encryption [16] and threshold cryptography [10]. Finally, perfect hash families arise as ingredients in some recursive constructions for covering arrays [14]. Covering arrays have a wide range of applications, most prominently in software interaction testing.

The goal of this paper is threefold. Primarily, we produce the first comprehensive existence tables for perfect hash families for a wide range of parameters. This is motivated by the need not only to produce explicit sets for applications, but also in order to assess the utility of constructions both known and new. Secondly, we review the known constructions available for PHF construction. Thirdly, we develop new constructions. The new direct methods include a somewhat unexpected construction using sets of integers with no three-term arithmetic progression, and the new recursive constructions include "Roux-type" methods that have proven powerful in the construction of covering arrays.

2 Direct constructions

Previous research on perfect hash families has focused on producing direct constructions based on related combinatorial objects. We first present known results and then turn to new direct constructions.

2.1 Known direct constructions

All optimal perfect hash families are known for strength 1 and 2. Given any k and v it is possible to construct the PHF with minimum possible N, and given any N and v it is possible to construct the PHF with maximum possible k.

For strength 1, one row is always sufficient (any single element set is vacuously composed of distinct elements). For strength 2, the construction is slightly more complicated.

Theorem 2.1. $\mathsf{PHFN}(k, v, 2) = \lceil \log_v k \rceil$.

Proof. To construct an array with N rows, use all possible distinct N-tuples on v symbols as columns. Then we have $k = v^N$ columns. There cannot be an $N \times 2$ sub-array containing no row with distinct symbols, since if there were, the two columns would be identical. Therefore, the array is a perfect hash family. Adding further columns would require duplication of an existing column. □

For strengths 3 and higher exact results are in general not known. The simplest construction gives optimal PHFs for one row.

Lemma 2.2. *There exists a* $\mathsf{PHF}(1; v, v, t)$*, and it is optimal. The array consists of one copy of every symbol.*

The first interesting construction produces a PHF from codes. We first provide a few definitions. Let $x = (x_1, x_2, \ldots, x_N)$ and $y = (y_1, y_2, \ldots, y_N)$ be any q-ary vectors of length N. The *Hamming distance* between x and y is $d(x, y) = |\{i : x_i \neq y_i\}|$. An (N, K, D, q)*-code* is a set $\mathcal{C}$ of K vectors in $\{1, \ldots, q\}^N$ such that the Hamming distance between any two distinct vectors in $\mathcal{C}$ is at least D. Codes over an alphabet of size q are often referred to as q-ary codes.

Theorem 2.3 ([1]). *If there is an* (N, K, D, q) *code* $\mathcal{C}$*, then there is a* $\mathsf{PHF}(N; K, q, t)$ *when* $(N - D)\binom{t}{2} < N$*.*

Using Reed-Solomon codes, we obtain:

Corollary 2.4 ([1]). *Suppose* N *and* v *are given, with* v *a prime power and* $N \leq v + 1$*. Then there exists a* $\mathsf{PHF}(N; v^{\left\lceil \frac{N}{\binom{t}{2}} \right\rceil}, v, t)$*.*

In constructing so-called "IPP-codes" [33], Trung and Martirosyan [33] prove:

Lemma 2.5 ([33]). *For any prime power* $v \geq 3$ *and any* $i \geq 1$*, there exists a* $\mathsf{PHF}((i + 1)^2; v^{i+1}, v, 3)$*.*

Lemma 2.6 ([33]). *For any prime power* $v \geq 4$ *and any* $i \geq 1$*, there exists a* $\mathsf{PHF}(\frac{5}{6}(2i^3 + 3i^2 + i) + i + 1; v^{i+1}, v, 4)$*.*

Trung and Martirosyan [32] develop a class of codes to give:

Theorem 2.7 ([32]). *Let* q_0 *and* q_1 *be prime powers such that* $q_1 \geq q_0$*, and* $i \geq 1$ *is an integer. Then, for any integer* N *with* $N \leq q_0 q_1^i + q_1^i + q_1^{i-1} + \cdots + q_1 + 1$ *there exists a* $\mathsf{PHF}(N; k, v, t)$ *with* $k = q_0^2 q_1^i$*,* $v = q_0 q_1^i$*, and* $t = \left\lceil \frac{\sqrt{8N+1}-1}{2} \right\rceil$*.*

An $m \times n$ *latin rectangle* is an $m \times n$ array, $m \leq n$, in which each cell contains a symbol from an n-set; no symbol occurs twice in any row or in any column. Two $m \times n$ latin rectangles are *orthogonal* if, when superimposed, every ordered pair of symbols arises at most once. A set of k latin rectangles, each $m \times n$, is *mutually orthogonal* if every two latin rectangles in the set is orthogonal; such a set is called k *MOLR*. An equivalent structure, an "$(n; m, k + 2)$-difference function family", is defined in [31]. When $m = n$, this is the more standard combinatorial structure, *mutually orthogonal latin squares*, or *MOLS*.

Theorem 2.8 ([29]). *Suppose there are at least* $s = \binom{t}{2} - 1$ *MOLR of size* $m \times n$*. Then there exists a PHF*$(s + 2; mn, n, t)$*.*

Corollary 2.9 ([29]). *Suppose there are at least* $s = \binom{t}{2} - 1$ *MOLS of order* v*. Then there exists a* $\mathsf{PHF}(s + 2; v^2, v, t)$*.*

Using a class of orthogonal arrays developed by Bierbrauer, the following is proved:

Theorem 2.10 ([29]). *For q a prime power and for any positive integers n, m, i such that $n \geq m$, $2 \leq i \leq q^n$, and $\binom{t}{2} < \frac{q^m}{i-1}$, there exists a* $\mathsf{PHF}(q^n; q^{m+(i-1)n}, q^m, t)$.

Blackburn [9] uses the Cartesian product $\{1, \ldots, a\}^t$ to show:

Theorem 2.11. *For every integer $a \geq 2$ there exists a* $\mathsf{PHF}(t, a^t, a^{t-1}, t)$.

Blackburn [8] gives a construction based on affine planes for $t = 4$ and v prime:

Theorem 2.12. *There exists a* $\mathsf{PHF}(6; v^2, v, 4)$ *for $v = 11$ and every prime $v \geq 17$.*

Finally, Atici *et al.* [4] provide a construction from resolvable balanced incomplete block designs (RBIBDs):

Theorem 2.13. *Suppose there exists a (v, b, r, k, λ)-RBIBD, where $r > \lambda\binom{t}{2}$. Then there exists a* $\mathsf{PHF}(r; v, \frac{v}{k}, t)$.

2.2 A new direct construction

We start not with a construction, but with a lower bound.

Theorem 2.14. $\mathsf{PHFN}(v+1, v, t) > \left\lfloor \frac{t}{2} \right\rfloor$.

Proof. Let A be a $\mathsf{PHF}(N; k = v+1, v, t)$. At least one symbol is duplicated in each row of A since $k > v$. Assume that $N \leq \left\lfloor \frac{t}{2} \right\rfloor$. Choose every column that is part of a duplicate in any row to obtain $c \leq 2N \leq t$ columns. Restricting A to these c columns maintains a duplicate entry in every row, and hence is not a strength c perfect hash family. Since $c \leq t$, A is not a strength t PHF. ∎

Given $\left\lfloor \frac{t}{2} \right\rfloor + 1$ rows, we can do better.

Theorem 2.15 (First-N Construction). *For $s \geq 1$ and $m \geq 2$,* $\mathsf{PHFN}(ms+m, ms+1, 2s+1) = s+1$.

Proof. By Theorem 2.14, $\mathsf{PHFN}(ms+m, ms+1, 2s+1) \geq \mathsf{PHFN}(ms+2, ms+1, 2s+1) > s$. We show that $\mathsf{PHFN}(ms+m, ms+1, 2s+1) \leq s+1$.

Create an array with $ms+m$ columns and $s+1$ rows. Partition the rows into $s+1$ blocks of m symbols each; the jth block of the jth row is the *primary block* for the row. In each column of the primary block of row j, place symbol v where $v = ms+1$. There remain ms unfilled positions in each row, so place the symbols $1, \ldots, v-1$ once each.

Consider any set T of t columns. For $1 \leq i \leq s+1$, the ith row fails to be distinct for T if and only if two or more of the columns are in its primary block. Since $t < 2(s+1)$, at least one primary block contains no two columns of T. Hence, the array is a PHF. ∎

3 Strength three with three rows

We next consider a specific case in which the bound on the number of rows provided by the First-N Construction is exceeded by one.

The *dual* of a $\mathsf{PHF}(N;k,v,t)$ P is obtained as follows. Let K be a set of k elements, the column indices of the PHF. For row i and symbol j, form a set $B_{ij} = \{\ell : P_{i,\ell} = j\}$ called a *block*. Define $\mathcal{B} = \{B_{ij} : 1 \le i \le N, 1 \le j \le v\}$; the *set system* $\mathcal{B}$ is N*-regular* in that each of the k points appears in exactly N blocks. Now $\mathcal{B}_i = \{B_{ij} : 1 \le j \le v\}$ is a partition of K, called a *parallel class* of blocks on K. The set system $\mathcal{B}$ then has a partition into N parallel classes $\mathcal{B}_1, \ldots, \mathcal{B}_N$; this partition is called a *resolution* and the set system is *resolvable* when it admits a resolution. Thus a $\mathsf{PHF}(N;k,v,t)$ gives rise to an N-regular, resolvable set system on k points.

Fix $N = t = 3$. Now suppose that some pair occurs in more than one block. If pair $\{x,y\}$ occurs in both B_{11} and B_{21}, without loss of generality, both x and y must appear as singleton sets in $\mathcal{B}_3$; otherwise if x appears with z in $\mathcal{B}_3$, then there is no row that separates x, y, and z. This does not preclude x and y being together twice, but it does force singleton classes in the partition. A *linear space* is a set system in which no pair occurs in more than one block (sometimes this definition excludes blocks of size 0 or 1; we do not).

Now restrict attention to 3-regular, resolvable linear spaces, and ask: Which are duals of $\mathsf{PHF}(3;k,v,3)$s? Consider three elements $\{x,y,z\} \subseteq K$. If $\{x,y\}$ is contained in a block of $\mathcal{B}_1$, $\{x,z\}$ is contained in a block of $\mathcal{B}_2$, and $\{y,z\}$ is contained in a block of $\mathcal{B}_3$, again this fails to be the dual of a PHF. Hence we also require that $\mathcal{B}$ be *triangle-free*.

Our goal then is to construct a triangle-free, 3-regular, resolvable linear space (a tfrrls for short). The number of symbols v of the $\mathsf{PHF}(3;k,v,3)$ is the largest number of blocks in one of the classes $\mathcal{B}_i$, $i \in \{1,2,3\}$, and the number of columns k is the size of the underlying point set of the tfrrls. So we use *tfrrls*(v,k) to denote a tfrrls on k points with at most v blocks in each parallel class.

3.1 Tfrrls

We examine a specific construction for tfrrls$(v,\ell v)$ over $\mathbb{Z}_v \times \{f_0, f_1, \ldots, f_{\ell-1}\}$. Let $A = (a_0, \ldots, a_{\ell-1})$; we associate the integer a_i modulo v with the point (a_i, f_i). The jth *translate* of a point (a_i, f_i) under $\mathbb{Z}_v$ is the point $(a_i + j \bmod v, f_i)$, and the jth translate of a set of points consists of the jth translates of the points in the set. Form $\mathcal{B}_1$ as the translates of $((0,f_0),(0,f_1),\ldots,(0,f_{\ell-1}))$, $\mathcal{B}_2$ as the points associated with translates of A, and $\mathcal{B}_3$ as the points associated with translates of $-A$ (here and elsewhere arithmetic is done modulo v, so that $-a_b = v - a_b \bmod v$). The result is a 3-regular resolvable set system. It is a linear space when $a_i \not\equiv a_j \pmod{v}$ and $a_i - a_j \not\equiv -(a_i - a_j) \pmod{v}$; in other words, $2a_i \not\equiv 2a_j \pmod{v}$. Hence we require that A contain integers that are distinct modulo v when v is odd, and modulo $v/2$ when v is even.

Now we treat the harder question of when the result is triangle-free. Without loss of generality, in a triangle we may assume that the points associated with A form a block in the triangle. Now suppose that a corner of the triangle involves the point

(a_i, f_i). Then the block of $\mathcal{B}_1$ forming the second side of the triangle is $\{(a_i, f_m) : 0 \le m < \ell\}$. The question then is whether among the blocks arising from translates of $-A$ there is one containing (a_i, f_k) and (a_j, f_j) for some choice of j and k. The only possible translate is $-A + (a_i + a_k)$, and hence to form a triangle we require that $-a_j + a_i + a_k = a_j$, Thus a triangle is formed precisely when two entries in A sum to twice a third element.

A set $A = \{a_0, \dots, a_{\ell-1}\}$ *has no three-term arithmetic progression modulo* v whenever for distinct $i, j, k \in \{0, \dots, \ell-1\}$, $a_i + a_j \not\equiv 2a_k \pmod{v}$ when $i \ne k$. We permit that $i = j$ to exclude cases in which $2a_i \equiv 2a_k \pmod{v}$ as before. If the congruence were to hold, a_k is the "average" of a_i and a_j. The term *non-averaging set* is sometimes applied when the arithmetic mean of some set of two or more elements in the sequence also belongs to the sequence [11]; in our case we are only concerned with sums of two elements.

This gives the main theorem.

Theorem 3.1. *Given a set of size ℓ with no three-term arithmetic progression modulo v, we immediately obtain a tfrrls$(v, \ell v)$ and hence a* $\mathsf{PHF}(3; \ell v, v, 3)$.

Wanless [36] recasts the existence problem for three-row PHFs in terms of partial latin squares and also derives the relationship with integer sequences having no three-term arithmetic progression.

3.2 Constructions

We treat the simple "greedy" construction: start with the empty set A and consider the nonnegative integers in sequence, adding each to A exactly when no three-term arithmetic progression is introduced. This has a well understood behaviour that we exploit here. Let $v \ge 3^\alpha$ and $\ell = 2^\alpha$. We claim that a tfrrls$(v, \ell v)$ exists. Let $0 \le x < 2^\alpha$ be an integer and write $x = \sum_{i=0}^{\alpha-1} b_i 2^i$. Then define $\tau(x) = \sum_{i=0}^{\alpha-1} b_i 3^i$. Now define $A = \{\tau(x) : 0 \le x < 2^\alpha\}$. Then A has no three-term arithmetic progression, as follows. Every ternary representation of an entry of A contains only '0' and '1' entries, and hence summing ternary representations of two causes no carry. Since any two differ in at least one position in the ternary representation, their sum contains at least one position with a '1' entry, and hence is not equal to the average of any two. The largest entry in A is $(3^\alpha - 1)/2$, and hence provided $v \ge 3^\alpha$ the negatives and doubles are all disjoint.

Proceeding in the same manner, any subset of A has no three-term arithmetic progression in the integers, and this remains true modulo v when $v > 2 * \max(A)$. For example, $\{0, 1, 3, 4, 9, 10\}$ has no three-term arithmetic progression modulo 21.

An exhaustive search with $v \le 96$ establishes that the largest size of a set having no three-term arithmetic progression modulo v is 2 for $v \in \{5, 6\}$; 3 for $v \in \{7, 8\}$; 4 for $v \in \{9-16\}$; 5 for $v \in \{17, 18, 20\}$; 6 for $v \in \{19, 21-24\}$; 7 for $v \in \{25, 26\}$; 8 for $v \in \{27-34, 36, 38\}$; 9 for $v \in \{35, 40-44\}$; 10 for $v \in \{37, 39, 45-50\}$; 11 for $v \in \{51, 53-56, 58\}$; 12 for $v \in \{52, 57, 59, 60, 62\}$; 13 for $v \in \{61, 63, 64, 66, 67, 68\}$; 14 for $v \in \{65, 69-78\}$; 15 for $v \in \{79, 80\}$; 16 for $v \in \{81-84, 86-90, 92, 94\}$; 17 for $v \in \{85, 91, 93, 95, 96\}$;

This results in the creation of the following PHFs:

PHF(3; 10, 5, 3)	PHF(3; 12, 6, 3)	PHF(3; 21, 7, 3)	PHF(3; 24, 8, 3)
PHF(3; 36, 9, 3)	PHF(3; 40, 10, 3)	PHF(3; 44, 11, 3)	PHF(3; 48, 12, 3)
PHF(3; 52, 13, 3)	PHF(3; 56, 14, 3)	PHF(3; 60, 15, 3)	PHF(3; 64, 16, 3)
PHF(3; 85, 17, 3)	PHF(3; 90, 18, 3)	PHF(3; 114, 19, 3)	PHF(3; 126, 21, 3)
PHF(3; 132, 22, 3)	PHF(3; 138, 23, 3)	PHF(3; 144, 24, 3)	PHF(3; 175, 25, 3)
PHF(3; 182, 26, 3)	PHF(3; 216, 27, 3)	PHF(3; 224, 28, 3)	PHF(3; 232, 29, 3)
PHF(3; 240, 30, 3)	PHF(3; 248, 31, 3)	PHF(3; 256, 32, 3)	PHF(3; 264, 33, 3)
PHF(3; 272, 34, 3)	PHF(3; 315, 35, 3)	PHF(3; 370, 37, 3)	PHF(3; 390, 39, 3)
PHF(3; 396, 44, 3)	PHF(3; 450, 45, 3)	PHF(3; 460, 46, 3)	PHF(3; 470, 47, 3)
PHF(3; 480, 48, 3)	PHF(3; 490, 49, 3)	PHF(3; 500, 50, 3)	PHF(3; 561, 51, 3)
PHF(3; 624, 52, 3)	PHF(3; 684, 57, 3)	PHF(3; 708, 59, 3)	PHF(3; 720, 60, 3)
PHF(3; 793, 61, 3)	PHF(3; 819, 63, 3)	PHF(3; 832, 64, 3)	PHF(3; 910, 65, 3)
PHF(3; 966, 69, 3)	PHF(3; 980, 70, 3)	PHF(3; 994, 71, 3)	PHF(3; 1008, 72, 3)
PHF(3; 1022, 73, 3)	PHF(3; 1036, 74, 3)	PHF(3; 1050, 75, 3)	PHF(3; 1064, 76, 3)
PHF(3; 1078, 77, 3)	PHF(3; 1092, 78, 3)	PHF(3; 1185, 79, 3)	PHF(3; 1200, 80, 3)
PHF(3; 1296, 81, 3)	PHF(3; 1312, 82, 3)	PHF(3; 1328, 83, 3)	PHF(3; 1344, 84, 3)
PHF(3; 1445, 85, 3)	PHF(3; 1547, 91, 3)	PHF(3; 1581, 93, 3)	PHF(3; 1615, 95, 3)
PHF(3; 1632, 96, 3)			

It is reasonable to ask whether the construction here yields results close to the best possible. In the next subsection we demonstrate that asymptotically it does not; however it appears to be useful for small values of v.

3.3 The connection with additive combinatorics

Let $r(n)$ be the size of the largest subset of $\{0, 1, \ldots, n\}$ that contains no three-term arithmetic progression. It may seem that the greedy algorithm yields a large value of $r(n)$, showing that $r(n)$ is $\Omega(n^{\log_3 2-1})$. In 1946 Behrend [6] improved dramatically on this lower bound. A progression-free set in $\mathbb{R}^\ell$ can be obtained using a sphere. So, consider an d-dimensional cube $[1,\ell]^d \cap \mathbb{Z}^d$ and family of spheres $x_1^2+x_2^2+\cdots+x_d^2 = t$ for $t = 1, \ldots, d\ell^2$. Each point in the cube is contained in one of the spheres, and so at least one of the spheres contains a set A of at least $\ell^d/d\ell^2$ lattice points. Now A does not contain any progressions since the sphere does not. A *Freiman isomorphism* [17] of order s is a bijective mapping $f\colon A \to B$ such that $a_1+a_2+\cdots+a_s = a_1'+a_2'+\cdots+a_s'$ holds if and only if $f(a_1)+f(a_2)+\cdots+f(a_s) = f(a_1')+f(a_2')+\cdots+f(a_s')$.

Set $f(x) = x_1 + x_2(2\ell) + x_3(2\ell)^2 + \cdots + x_d(2\ell)^{d-1}$ for $x = \{x_1, x_2, \ldots, x_d\} \in A$; that is, we treat x_i as i'th digit of $f(x)$ in base 2ℓ. Then f is a Freiman isomorphism of order 2 from A to a subset of $\mathbb{Z}$; $f(A) \subset \{1, 2, \ldots, n = (2n)^d\}$. Set $d = c\sqrt{\ln n}$ to establish that there is a progression-free subset of $\{1, 2, \ldots, n\}$ of size at least $ne^{-\sqrt{\ln n}(c \ln 2 + 2/c + o(1))}$. To maximize, set $c = \sqrt{2/\ln 2}$. Consequently, there exists a progression-free set of size at least $ne^{-\sqrt{8 \ln 2 \ln n}(1+o(1))}$. Related work appears in [3, 25, 26]. Alon [2] treated the modular version of the problem, in the language of cyclic groups $\mathbb{Z}_n$. Generalizations to finite abelian groups appear in [22, 24].

Roth [27] proved that $r(n) < cn/\log\log n$. This was improved by Heath-Brown [21] to $O(n(\log n)^{-c})$, for an unspecified constant $c > 0$. Szemerédi [30] obtains the same bound and shows that $c = \frac{1}{4}$ is admissible; see also [19]. Bourgain [12] improved this to $O(n(\log\log n/\log n)^{1/2})$.

In our context, the importance of this prior research is that for sufficiently large number v of symbols, the greedy approach does not produce the best available tfrrls; however for "small" numbers of symbols, it appears to be a useful technique. It remains interesting to find other constructions of tfrrls that do not require progression-free sequences; also showing that $k = \mathrm{o}(v^2)$ for a $\mathsf{PHF}(3; k, v, 3)$ remains open.

4 Recursive constructions

Recursive constructions take one or more perfect hash families and produce a new perfect hash family. Several recursive constructions are known, and several more are introduced here.

Blackburn [7] gives a simple product construction, *composition*:

Theorem 4.1. *Suppose there exist* $\mathsf{PHF}(N_0; k, x, t)$ A *and* $\mathsf{PHF}(N_1; x, v, t)$ B. *Then there exists a* $\mathsf{PHF}(N_0N_1; k, v, t)$.

Combine Corollary 2.9 and Theorem 4.1 to obtain:

Corollary 4.2 ([29]). *If there exist* $\binom{t}{2} - 1$ *MOLS of order* k *and a* $\mathsf{PHF}(N_0; k, v, t)$, *then there exists a* $\mathsf{PHF}((\binom{t}{2} + 1)N_0; k^2, v, t)$.

The existence of q MOLS of order k implies the existence of at least q MOLS of order k^j for $j \geq 1$ [13]; hence this process can be iterated. Theorem 4.2 is equivalent to Theorem 13 in [31]; it generalizes and improves upon two constructions given in [4]. Tonien and Safavi-Naini [31] give a further generalization:

Theorem 4.3 ([31, Theorem 14]). *If a* $\mathsf{PHF}(N_1; k_1, v_1, t)$, *a* $\mathsf{PHF}(N_2; k_2, v_2, t)$, *and* $\binom{t}{2} - 1$ *MOLR of size* $k_1 \times k_2$ *all exist, then a* $\mathsf{PHF}(\binom{t}{2}N_1 + N_2; k_1k_2, max(v_1, v_2), t)$ *exists.*

Atici *et al.* [4] also give a Kronecker-product type construction:

Theorem 4.4. *Suppose that the following exist:*

- *a* $\mathsf{PHF}(N_1; k_0k_1, v, t)$,
- *a* $\mathsf{PHF}(N_2; k_2, k_1, t - 1)$,
- *a* $\mathsf{PHF}(N_3; k_2, v, t)$.

Then there is a $\mathsf{PHF}(N_1N_2 + N_3; k_0k_2, v, t)$.

We next state two basic constructions that do not seem to appear in the literature, but are almost certainly general knowledge. The first increases the number of columns for "free" while increasing the number of symbols.

Lemma 4.5. $\mathsf{PHFN}(k + 1, v + 1, t) \leq \mathsf{PHFN}(k, v, t)$.

Proof. Let A be a $\mathsf{PHF}(N;k,v,t)$. Appending a column entirely comprised of a new symbol gives us the desired result. Any set of columns not including the last is treated in A. Any set of columns including the last has at least one distinct row because the $t-1$ in A have a distinct row and none of its symbols could possibly be the added one. ∎

Using a $\mathsf{PHF}(8;8,6,6)$ we produce the $\mathsf{PHF}(8;9,7,6)$ in Figure 2.

$$\begin{bmatrix} 4&3&5&0&1&4&2&2\\ 4&5&1&1&3&4&0&2\\ 5&1&3&2&2&4&0&3\\ 5&0&1&0&2&3&1&4\\ 2&5&1&2&5&4&3&0\\ 4&2&2&0&3&5&1&1\\ 0&1&4&2&5&3&3&5\\ 0&0&2&1&5&5&4&3 \end{bmatrix} \qquad \begin{bmatrix} 4&3&5&0&1&4&2&2&6\\ 4&5&1&1&3&4&0&2&6\\ 5&1&3&2&2&4&0&3&6\\ 5&0&1&0&2&3&1&4&6\\ 2&5&1&2&5&4&3&0&6\\ 4&2&2&0&3&5&1&1&6\\ 0&1&4&2&5&3&3&5&6\\ 0&0&2&1&5&5&4&3&6 \end{bmatrix}$$

Figure 2. A $\mathsf{PHF}(8;8,6,6)$ and a $\mathsf{PHF}(8;9,7,6)$

We can also multiply both the number of symbols and the number of columns by the same factor.

Lemma 4.6. $\mathsf{PHFN}(\ell k,\ell v,t) \le \mathsf{PHFN}(k,v,t)$.

Proof. Let A be a $\mathsf{PHF}(N;k,v,t)$. Place ℓ copies of A side by side, using a different set of v symbols for each copy. Any set of t columns arises from t or fewer columns of the original A, and therefore there is a distinct row in the original A. This row is now spread out among copies of A, and may contain duplicate symbols if any of the t columns correspond to the same column in A. However, the copies of A all use disjoint symbol sets, so the duplicate columns arise from different symbol sets. Hence the row is distinct in the new array. ∎

Now we turn our attention to new constructions. We can generalize Lemma 4.6 when $t=3$; juxtapose any two arrays, not just two copies of the same array:

Theorem 4.7. *Let* A *be a* $\mathsf{PHF}(N_1;k_1,v_1,3)$ *and* B *be a* $\mathsf{PHF}(N_2;k_2,v_2,3)$. *There exists a* $\mathsf{PHF}(N_1;k_1+k_2,v_1+v_2,3)$ *when* $N_1 \ge N_2$.

Proof. Ensure that the v_1 symbols used in A are different from the v_2 symbols used for B. If $N_1 > N_2$, extend B to have N_1 rows by filling in the additional rows with any symbol from B. Then juxtapose the arrays horizontally to obtain an array of the desired parameters. Any set of t columns entirely in A or entirely in B is handled by that array. It remains to consider the case of two columns from one block and one column from the other. The two columns are distinct in at least one row, and the remaining column arises from a different symbol set. ∎

We use a similar idea to create a “Roux-type” construction for arrays with many symbols. *Roux-type* constructions are recursive constructions that place copies of an object side by side and handle omitted cases by using additional rows. A comprehensive discussion for covering arrays appears in [14]. In the following proofs we will use arithmetic modulo v to describe symbol manipulation concisely. The first Roux-type construction for perfect hash families is:

Theorem 4.8. $\mathsf{PHFN}(k\ell, v, t) \le \mathsf{PHFN}(k, v, t) + \mathsf{PHFN}(k, \lfloor \frac{v}{l} \rfloor, t-1)$ *whenever* $\ell(t-1) \le v$.

Proof. Suppose that the following exist:

- $\mathsf{PHF}(N_1; k, v, t)$ A,
- $\mathsf{PHF}(N_2; k, \lfloor \frac{v}{l} \rfloor, t-1)$ B

We produce a perfect hash family $\mathsf{PHF}(N'; k\ell, v, t)$ C where $N' = N_1 + N_2$. C is formed by vertically juxtaposing arrays C_1 of size $N_1 \times k\ell$ and C_2 of size $N_2 \times k\ell$. We index $k\ell$ columns by ordered pairs from $\{1, \dots, k\} \times \{1, \dots, \ell\}$.

In row r and column (f, h) of C_1 place the entry in cell (r, f) of A. Thus C_1 consists of ℓ copies of A placed side by side.

Set $v' = \lfloor \frac{v}{l} \rfloor$. In row r and column (f, h) of C_2 place the entry $x + v'(h-1)$ where x is the entry in row r and column f of B. Since $v'\ell \le v$, C_2 consists of ℓ structurally equivalent copies of C_2 on distinct symbol sets placed side by side. This is essentially the construction given in Theorem 4.6.

We show that C is a perfect hash family. Consider $(f_1, h_1), (f_2, h_2), \dots, (f_t, h_t)$, a set of t columns of C. If $f_1, f_2, \dots, f_t$ are all distinct, then these columns restricted to C_1 arise from t distinct columns of A. Hence, at least one row has distinct symbols.

It remains to consider the case where not all columns are distinct. If any $f_i = f_j$ there are w distinct columns for some $w \le t-1$. Since B is a perfect hash family of strength $t-1$, these columns restricted to C_2 arise from w columns of B. Therefore, at least one row r in B contains distinct entries in these w columns. Consider the translated copies of B that make up C_2: if $f_i = f_j$ then $h_i \ne h_j$, so any column equalities come from different copies of B. Since each translate of B is on a different symbol set, row r of C_2 contains distinct values in all t columns. Hence, C is a perfect hash family. ☐

This construction is limited to the case when $v \ge \ell(t-1)$. When v is smaller, a different approach is useful:

Theorem 4.9. $\mathsf{PHFN}(2k, v, 3) \le \mathsf{PHFN}(k, v, 3) + 2\mathsf{PHFN}(k, v, 2)$.

Proof. Suppose that there exist a

- $\mathsf{PHF}(N_1; k, v, 3)$ A, and a
- $\mathsf{PHF}(N_2; k, v, 2)$ B.

We produce a perfect hash family $\mathsf{PHF}(N'; 2k, v, t)$ C where $N' = N_1 + 2N_2$. C is formed by vertically juxtaposing arrays D of size $N_1 \times 2k$ and E_1 and E_2 each of size $N_2 \times 2k$. We index $2k$ columns by ordered pairs from $\{1, \dots, k\} \times \{1, 2\}$.

In row r and column (f,h) of D place the entry in cell (r,f) of A. Thus D consists of 2 copies of A placed side by side.

Set x equal to the entry in cell (r,f) of B. In row r and column $(f,1)$ of E_i place x. In row r and column $(f,2)$ of E_i place $x+i$.

To show that C is a perfect hash family, consider columns $(f_1,h_1),(f_2,h_2),(f_3,h_3)$ of C. If f_1,f_2,f_3 are all distinct, then these columns restricted to D arise from $t=3$ distinct columns of A. Hence, there is at least one row on distinct symbols.

Without loss of generality it remains to treat the case when $f_1=f_2$. Then in the three columns, E_i contains the values $(x,x+i,y)$ in the row that contains distinct values for columns f_1 and f_3 restricted to B. Then we must avoid the case where $x+i=y$. Since this only eliminates one choice of i, the other E array must contain a distinct row. ◻

In order to extend this construction, define a *partial difference covering array* $\mathsf{D}=(d_{ij})$ over a group Γ (a $\mathsf{PDCA}(N,\Gamma;t,k,v,c)$ for short) to be an $N\times k$ array with entries from Γ having the property that for any t distinct columns $j_1,j_2,\ldots,j_t$, the set $\{(d_{i,j_1}\odot d_{i,j_2}^{-1},d_{i,j_1}\odot d_{i,j_3}^{-1},\ldots,d_{i,j_1}\odot d_{i,j_t}^{-1}):1\le i\le N\}$ contains at least c distinct nonzero $(t-1)$-tuples over Γ. When $\Gamma=\mathbb{Z}_v$ we omit it from the notation. We denote by $\mathsf{PDCAN}(t,k,v,c)$ the minimum N for which a $\mathsf{PDCA}(N;t,k,v,c)$ exists. A $\mathsf{PDCA}(N;2,k,v,1)$ is equivalent to a $\mathsf{PHF}(N;k,v,2)$.

Now we extend Theorem 4.9:

Theorem 4.10. *For any integer $\ell\ge 3$,*
$\mathsf{PHFN}(k\ell,v,3)\le\mathsf{PHFN}(k,v,3)+\mathsf{PHFN}(\ell,v,3)+\mathsf{PDCAN}(2,\ell,v,2)\mathsf{PHFN}(k,v,2)$.

Proof. Suppose that the following exist:

- $\mathsf{PHF}(N_1;k,v,3)$ A
- $\mathsf{PHF}(N_2;\ell,v,3)$ B
- $\mathsf{PHF}(N_3;k,v,2)$ K
- $\mathsf{PDCA}(M;2,\ell,v,2)$ R

We produce a perfect hash family $\mathsf{PHF}(N';k\ell,v,t)$ C where $N'=N_1+N_2+MN_3$. C is formed by vertically juxtaposing arrays D of size $N_1\times k\ell$, E of size $N_2\times k\ell$, and F_1 through F_M each of size $N_3\times k\ell$. We describe the construction of each array in turn. We index $k\ell$ columns by ordered pairs from $\{1,\ldots,k\}\times\{1,\ldots,\ell\}$.

In row r and column (f,h) of D, place the entry in cell (r,f) of A. Thus D consists of ℓ copies of A placed side by side.

In row r and column (f,h) of E, place the entry in cell (r,h) of F. Thus E consists of k copies of each column of B.

In row r and column (f,h) of F_i, place $\mathsf{K}_{rf}+\mathsf{R}_{ih}$. Thus the F arrays are obtained from K by cyclic shifts of the symbols as directed by R.

We show that C is a perfect hash family. Consider columns $(f_1,h_1),(f_2,h_2),(f_3,h_3)$ of C. If f_1,f_2,f_3 are all distinct, then these columns restricted to D arise from t distinct columns of A. Hence there is at least one row on distinct symbols. If h_1,h_2,h_3 are all distinct, then these columns restricted to E arise from distinct columns in B. Hence again there is at least one row on distinct symbols.

Without loss of generality, it remains to consider the case where $f_1 = f_2 \neq f_3$, $h_1 = h_3 \neq h_2$, i.e. two columns from one block and one duplicated column from another. Therefore all tuples of the form $(x, x+i, y)$ with $x \neq y$ are covered, where i can be any of the differences found in columns h_1 and h_2 of R. At least one of the two distinct i values results in $(x, x+i, y)$ being a distinct tuple. Therefore, all possible column selections are covered. □

In order to generate values for PDCAN, we use:

Theorem 4.11. $\mathsf{PDCAN}(2,k,v,2) \leq 2\mathsf{PHFN}(k,v,2) = 2\lceil \log_v(k) \rceil$ *for v odd or a prime power, $v > 2$.*

Proof. In either case, begin with a $\mathsf{PHF}(\log_v(k); k, v, 2)$ A.

For v odd, append an array of equal size B where $\mathsf{B}_{ij} = 2\mathsf{A}_{ij} \pmod v$. Then, in any given pair of columns, at least one row in A is distinct, and thus covers one non-zero difference d. Since v is odd, $d \neq 2d \pmod v$ and $2d \neq 0 \pmod v$. Hence, the corresponding row in B covers a second distinct non-zero difference. Thus at least two differences are covered.

For v a prime power, choose an element x of $\mathbf{GF}(v)$ where $x \neq 0$ and $x \neq 1$. We can guarantee a selection of x because $v > 2$. Append to A an array of equal size B where $\mathsf{B}_{ij} = x\mathsf{A}_{ij}$ with arithmetic done in $\mathbf{GF}(v)$. Then, in any given pair of columns, at least one row in A is distinct, and thus covers one non-zero difference d. We know that $xd \neq d$ because $x \neq 1$ and $xd \neq 0$ because $x \neq 0$ and $d \neq 0$. Hence, the corresponding row in B covers a second non-zero difference and at least two differences are covered. □

For strength $t = 4$, Theorem 4.8 does not apply when $v \in \{4, 5\}$. We treat the case when $v = 4$ here. Denote by $[x, y, z]$ a function with $f(0) = x, f(1) = y, f(2) = z$. For the following, assume that the symbol set of an array on three symbols is $\{0, 1, 2\}$.

Theorem 4.12. $\mathsf{PHFN}(2k,4,4) \leq \mathsf{PHFN}(k,4,4) + 3\mathsf{PHFN}(k,3,3) + \mathsf{PHFN}(k,2,2)$.

Proof. Suppose that the following exist:

- $\mathsf{PHF}(N_1; k, 4, 4)$ A
- $\mathsf{PHF}(N_2; k, 3, 3)$ B
- $\mathsf{PHF}(N_3; k, 2, 2)$ K

We produce a perfect hash family $\mathsf{PHF}(N'; 2k, 4, 4)$ C where $N' = N_1 + 3N_2 + N_3$. C is formed by vertically juxtaposing arrays D of size $N_1 \times 2k$, E_1, E_2, and E_3 each of size $N_2 \times 2k$, and F of size $N_3 \times 2k$. We index $2k$ columns by ordered pairs from $\{1, \ldots, k\} \times \{1, 2\}$.

In row r and column (f, h) of D, place the entry in cell (r, f) of A.

Set x equal to the entry in cell (r, f) of B. In row r and column $(f, 1)$ of E_i place x. In row r and column $(f, 2)$ of E_1, place $[3, 1, 2](x)$. In row r and column $(f, 2)$ of E_2, place $[0, 2, 3](x)$. In row r and column $(f, 2)$ of E_3, place $[0, 3, 1](x)$.

Set x equal to the entry in cell (r, f) of K. We use 0 and 1 as the symbols of K. In row r and column $(f, 1)$ of F place x. In column $(f, 2)$ of the same row, we place $x + 2$.

Consider four columns $(f_1, h_1), (f_2, h_2), (f_3, h_3), (f_4, h_4)$ of C. If f_1, f_2, f_3, f_4 are all distinct, then these columns restricted to D arise from $t = 4$ distinct columns of A. Hence at least one row has distinct symbols.

It remains to consider three cases. In the first, three columns from one block and one column with equal f from the other are selected. Without loss of generality, the equal columns are the first and last (i.e. $f_1 = f_4$), and the three columns are from block 1, so $h_1 = h_2 = h_3 = 1, h_4 = 2$.

In columns f_1, f_2, f_3 of B there is at least one distinct row. If there are more than one, consider only the first. We must consider each possible distinct row separately:

(0,1,2) : In E_1 we have the row $(0, 1, 2, [3, 1, 2](0)) = (0, 1, 2, 3)$.

(0,2,1) : In E_1 we have the row $(0, 2, 1, [3, 1, 2](0)) = (0, 2, 1, 3)$.

(2,0,1) : In E_2 we have the row $(2, 0, 1, [0, 2, 3](2)) = (2, 0, 1, 3)$.

(2,1,0) : In E_2 we have the row $(2, 1, 0, [0, 2, 3](2)) = (2, 1, 0, 3)$.

(1,0,2) : In E_3 we have the row $(1, 0, 2, [0, 3, 1](1)) = (1, 0, 2, 3)$.

(1,2,0) : In E_3 we have the row $(1, 2, 0, [0, 3, 1](1)) = (1, 2, 0, 3)$.

Therefore, there is a distinct row on four columns regardless of the distinct row found for the three columns.

The second and third cases arise when selecting two columns from each block. First consider when three of these columns are distinct, and hence one pair is equal. Without loss of generality, f_1, f_2, f_4 are distinct, $f_1 = f_3$, and $h_1 = h_2 = 1, h_3 = h_4 = 2$.

In columns f_1, f_2, f_4 of B there is at least one distinct row. If there are more than one, consider only the first. We consider each possible distinct row separately:

(0,1,2) : In E_1 we have the row $(0, 1, [3, 1, 2](0), [3, 1, 2](2)) = (0, 1, 3, 2)$.

(0,2,1) : In E_1 we have the row $(0, 2, [3, 1, 2](0), [3, 1, 2](1)) = (0, 2, 3, 1)$.

(1,0,2) : In E_2 we have the row $(1, 0, [0, 2, 3](1), [0, 2, 3](2)) = (1, 0, 2, 3)$.

(2,1,0) : In E_2 we have the row $(2, 1, [0, 2, 3](2), [0, 2, 3](0)) = (2, 1, 3, 0)$.

(1,2,0) : In E_3 we have the row $(1, 2, [0, 3, 1](1), [0, 3, 1](0)) = (1, 2, 3, 0)$.

(2,0,1) : In E_3 we have the row $(2, 0, [0, 3, 1](2), [0, 3, 1](1)) = (2, 0, 1, 3)$.

Again all cases are handled. Finally, consider the case selecting two identical columns from each block. Here, employ F. At least one row in K is distinct for these columns; hence the 4-tuple found in that row is also distinct since each block is defined on different symbol sets. ∎

We now discuss several recursive constructions that are not Roux-type. A simple construction exists to increase k by 1:

Theorem 4.13. *For $t \geq 3$,*

$$\mathsf{PHFN}(k + 1, v, t) \leq \mathsf{PHFN}(k, v, t) + \mathsf{PHFN}(k - 1, v - 2, t - 2).$$

Proof. Suppose that there exist a $\mathsf{PHF}(N_1; k, v, t)$ A, and a $\mathsf{PHF}(N_2; k-1, v-2, t-2)$ B. We produce a perfect hash family $\mathsf{PHF}(N'; k+1, v, t)$ C where $N' = N_1 + N_2$. C is formed by vertically juxtaposing arrays C_1 of size $N_1 \times (k+1)$ and C_2 of size $N_2 \times (k+1)$.

In row r and column c with $c \le k$ of C_1, place the entry in cell (r, c) of A. In column $k+1$ place the entry in cell (r, k) of A.

In row r and column c with $c \le k-1$ of C_2, place the entry in cell (r, c) of B. In column k, place N_2 copies of the $(v-1)$-th symbol and in column $k+1$, place N_2 copies of the v-th symbol. These are the symbols not used in B.

We show that C is a perfect hash family. Consider t columns of C. If this set of columns includes at most one of $\{k, k+1\}$ then restricted to C_1 they arise from t distinct columns of A, and hence at least one row has distinct symbols.

It remains to consider when both k and $k+1$ are included. Then, the remaining $t-2$ columns restricted to C_2 arise from $t-2$ distinct columns of B. Hence at least one row r has distinct symbols. Since B does not use the $(v-1)$-th and v-th symbols, the entries in columns k and $k+1$ are also distinct and hence the row r is distinct in the set of t columns. □

In fact, when $t = 3$ we can do better:

Theorem 4.14. $\mathsf{PHFN}(k+v-2, v, 3) \le \mathsf{PHFN}(k, v, 3) + 1$.

Proof. Let A be a $\mathsf{PHF}(N; k, v, 3)$. We produce a perfect hash family $\mathsf{PHF}(N+1; k+v-2, v, 3)$ C. C is formed by vertically juxtaposing arrays C_1 of size $N \times (k+v-2)$ and C_2 of size $1 \times (k+v-2)$.

In row r and column c with $c \le k$ of C_1, place the entry in cell (r, c) of A. In column $k+1$ through $k+v-2$, place the entry in cell (r, k) of A. Thus C_1 consists of A alongside $v-2$ copies of its last column.

In column c with $c \le k-1$ of C_1, place the symbol v. In column $k+i$ for $0 \le i \le v-2$, place the symbol $i+1$.

Consider three columns of C. If these three have at most one among the last $v-1$ columns, then, restricted to C_1, they arise from $t = 3$ distinct columns of A, and hence at least one row has distinct symbols.

C_2 takes care of the case where either all of two of the three columns lie among the last $v-1$ columns, since it is comprised of distinct symbols there and repeats a symbol only within the first $k-1$ columns. □

5 Computational search

Walker and Colbourn [35] introduce a class of arrays and a method to search for arrays in the class. The search employs tabu search, as introduced by Glover in [18]. The class includes any type of array that can be formulated as follows. Let $\mathcal{C} = \{C_i : i = 1, \ldots, \sigma\}$ be a set of subsets of same length tuples over an alphabet of size v. Let the length of the tuples in set C_i be denoted t_i. Define a $\mathcal{C}$-(N, k)-array to be an $N \times k$ array with entries from the same alphabet of size v, in which every $N \times t$ subarray has the property that for every i with $1 \le i \le \sigma$, there exists a row of the subarray equal to a t-tuple in C_i. No assumption is made that C_i and C_j are disjoint, nor that $t_i = t_j$, nor that a given tuple appear in any of the sets.

Taking $\mathcal{C}$ to have a single set C_1 in which t-tuples with distinct entries appear makes a $\mathcal{C}^*$-array a perfect hash family. The description of $\mathcal{C}$ is not an explicit listing of tuples; rather it is an oracle to test membership of a tuple in C_i. Testing a t-tuple for membership in C_1 is trivial. Using a modification of any O$(n \log n)$ sorting algorithm that stops whenever it finds two duplicate elements, we test in O$(t \log t)$ steps. When t is small, it is equally effective to test every element for equality with every other using exactly $\binom{n}{2}$ steps.

The maximum values of k given N, v and t for which PHFs are found are shown in Tables 1 through 4. Explicit solutions for each array appear in [34] and on the web site `http://www.phftables.com`. A selection of the results appear in Figures 3 through 6. The searches themselves took no more than 1 hour per perfect hash family, and often took 30 seconds to 5 minutes.

		N								
		3	**4**	**5**	**6**	**7**	**8**	**10**	**20**	**21**
v	**3**	6	9	10	12	16	19	29	90	95
	4			20	25		42			
	5	12		38	47					
	6	18		50						
	7	22		70						
	8	31								
	9	36								
	10	43								
	11	49								
	12	57								

Table 1. PHF table for k, given N and v, where t = 3

		N					
		3	**4**	**5**	**6**	**7**	**8**
v	**4**	5		6	8		9
	5	7	8	10	11	12	14

Table 2. PHF table for k, given N and v, where t = 4

		N										
		3	**4**	**6**	**7**	**8**	**10**	**11**	**13**	**18**	**23**	**28**
v	**5**	6		7		8		9	10	11	12	13
	6	7	8	9	10	11	12					

Table 3. PHF table for k, given N and v, where t = 5

		N					
		4	**6**	**8**	**11**	**13**	**18**
v	**6**	7		8		9	10
	7	8	9	10	11		

Table 4. PHF table for k, given N and v, where t = 6

$$\begin{bmatrix} 0 & 2 & 0 & 1 & 2 & 2 & 1 & 0 & 0 & 1 \\ 2 & 2 & 0 & 2 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 2 & 2 & 2 & 0 & 2 & 2 & 1 & 0 & 0 \\ 0 & 1 & 1 & 2 & 2 & 0 & 0 & 2 & 2 & 1 \\ 2 & 1 & 0 & 0 & 2 & 0 & 1 & 0 & 1 & 2 \end{bmatrix}$$

Figure 3. A $\mathsf{PHF}(5; 10, 3, 3)$

$$\begin{bmatrix} 2 & 3 & 2 & 0 & 3 & 2 & 4 & 4 & 3 & 1 & 1 \\ 3 & 3 & 1 & 0 & 4 & 4 & 0 & 1 & 2 & 0 & 2 \\ 0 & 1 & 2 & 3 & 3 & 1 & 0 & 4 & 3 & 2 & 4 \\ 1 & 2 & 1 & 0 & 3 & 4 & 2 & 0 & 2 & 3 & 4 \\ 4 & 3 & 0 & 0 & 4 & 0 & 1 & 3 & 2 & 1 & 2 \\ 1 & 2 & 0 & 4 & 4 & 3 & 3 & 4 & 0 & 2 & 1 \end{bmatrix}$$

Figure 4. A $\mathsf{PHF}(6; 11, 5, 4)$

$$\begin{bmatrix} 2 & 0 & 1 & 3 & 3 & 4 & 4 & 1 & 0 \\ 1 & 4 & 1 & 3 & 1 & 0 & 2 & 3 & 4 \\ 3 & 0 & 1 & 4 & 2 & 2 & 0 & 1 & 3 \\ 1 & 2 & 3 & 4 & 4 & 0 & 1 & 2 & 0 \\ 1 & 0 & 3 & 4 & 1 & 2 & 0 & 2 & 4 \\ 4 & 0 & 2 & 1 & 0 & 3 & 1 & 3 & 2 \\ 0 & 4 & 1 & 2 & 3 & 3 & 0 & 2 & 1 \\ 1 & 0 & 2 & 4 & 1 & 3 & 0 & 4 & 0 \\ 3 & 1 & 4 & 0 & 4 & 2 & 0 & 1 & 3 \\ 1 & 3 & 4 & 2 & 3 & 0 & 4 & 1 & 0 \\ 2 & 4 & 0 & 3 & 0 & 1 & 1 & 2 & 3 \end{bmatrix}$$

Figure 5. A $\mathsf{PHF}(11; 9, 5, 5)$

$$\begin{bmatrix} 0 & 3 & 5 & 1 & 4 & 4 & 2 & 0 & 6 \\ 4 & 2 & 6 & 3 & 5 & 3 & 0 & 5 & 1 \\ 0 & 1 & 1 & 5 & 4 & 6 & 2 & 3 & 0 \\ 0 & 3 & 5 & 1 & 6 & 1 & 1 & 2 & 4 \\ 2 & 5 & 6 & 4 & 0 & 1 & 2 & 3 & 0 \\ 4 & 6 & 2 & 3 & 5 & 1 & 2 & 6 & 0 \end{bmatrix}$$

Figure 6. A $\mathsf{PHF}(6; 9, 7, 6)$

Many rows appear to be required when $v = t$, especially as t grows. Figure 2 given earlier shows an array with $N = k = 8$, yet we can find no solution using fewer rows.

6 Tables

We adopt a bottom-up approach to building an existence table for PHFs. We start with PHFs from direct constructions and computational search. We then create new PHFs using recursive constructions. We iterate until we have the best known PHFs. Atici, Stinson, and Wei [5] give algorithms for constructing perfect hash families from known recursive constructions and direct constructions. Their approach is top-down, and therefore produces one PHF at a time. Indeed it may miss complicated interactions among PHFs that are reflected in our tables.

6.1 Implementation

We implemented a table generator in C++, based on a prototype written in Perl using MySQL. It takes less than 45 minutes on a 3 GHz Pentium IV to generate the entire set of tables given in [34]. Updates can be done incrementally, so addition of a new PHF incurs a significantly smaller cost than regenerating the tables.

We start with the set of perfect hash families from direct and computational constructions. We label the entire set as *unprocessed*. We then iterate through the list of unprocessed PHFs. To *process* an array we attempt to use it in every recursive construction. For instance, suppose we are processing $\mathsf{PHF}(7;49,7,4)$, produced by Corollary 2.4.

To use this array in a Roux-type construction, we must consider using it both as the first array and the second array. We combine it with the best known $\mathsf{PHF}(N;49,3,3)$ to form a $\mathsf{PHF}(7+N;98,7,4)$. The second case is more complicated. Since we tabulate ρ values instead of every possible PHF we must consider using this array as a $\mathsf{PHF}(7;48,7,4)$ and so on down to $\mathsf{PHF}(7;\rho(6,4,7)+1,7,4)$.

In addition, whenever inserting a new array into the pool of arrays, we consider two questions:

1. Is this array useful? I.e., does it change any ρ values?
2. Are any older arrays made obsolete by this aray? I.e., are there now arrays that do not affect any ρ values?

If the new array is not useful, we need not consider it any further. Likewise, if other arrays are made obsolete, we may remove them from the pool. Whenever we remove an array from the pool, we also must remove all of its descendants. Often times, an array becomes obsolete before it is processed, saving computation. In order to make this happen frequently, we process arrays in breadth-first order.

Tables of MOLS and RBIBDs are used from [13]. The MOLR table used is in [20].

6.2 Results

To efficiently generate tables, we keep a rich data structure in memory that tracks generated PHFs and links them to their ingredients and children. Complete tables as well as a system to browse these links exist at `http://www.phftables.com`. Using this information we can get sense of which constructions provide the most results.

The *score* of an array is the number of arrays directly or indirectly dependent on that array. The *score* of a construction is the number of arrays directly or indirectly dependent on that construction. Table 5 ranks the constructions by score for the 7313 PHFs in the tables.

The construction based on MOLS is the most useful direct construction, followed closely by the Reed-Solomon codes construction. The most useful "interesting" recursive constructions are the Kronecker product and composition. Strangely, the symbol product construction featured prominently in the tables until the introduction of the tfrrls construction, which overtook it completely.

As a matter of interest, 78.6% of the PHFs constructed depend on an array or construction presented in this paper for the first time.

Construction	Description	Score	Direct	Indirect
Theorem 4.5	Symbol increase	3961	1689	2272
Theorem 2.2	PHF$(1; v, v, t)$	3478	386	3092
Corollary 2.9	MOLS	3258	211	3047
Corollary 2.4	Reed-Solomon	2723	176	2547
Theorem 4.13	Column increase, $t \neq 3$	2373	1811	562
Section 5	Tabu search	2241	49	2192
Theorem 4.4	Kronecker product	2213	769	1444
Theorem 4.1	Composition	2182	566	1616
Theorem 4.8	High Symbol Roux-type	1935	752	1183
Theorem 2.15	First N	1842	151	1691
Theorem 2.13	RBIBD	750	7	743
Theorem 2.12	Affine plane	727	20	707
Theorem 4.14	Column increase, $t = 3$	578	415	163
Theorem 3.1	tfrrls	245	57	188
Theorem 4.9	Roux-type $t = 3, \ell \geq 3$	237	108	129
Theorem 2.7	Martirosyan Code	193	2	191
Corollary 4.2	MOLS composition	148	31	117
Theorem 2.10	Bierbrauer OA	130	4	126
Theorem 4.12	Roux-type $t = 4, v = 4, \ell = 2$	122	20	102
Lemma 2.5	IPP codes, $t = 3$	108	9	99
Theorem 4.7	$t = 3$ Juxtaposition	102	66	36
Lemma 2.6	IPP codes, $t = 4$	91	4	87
Theorem 2.11	Partition	48	6	42
Theorem 4.10	Roux-type $t = 3, \ell \geq 3$	10	2	8
Theorem 4.6	Symbol product	0	0	0

Table 5. Ranking of PHF constructions

6.3 PHF tables

We produce tables of upper bounds on PHFN for $3 \leq t \leq 6$, $t \leq v \leq 50$, $v \leq k \leq 500\,000$. To the best of our knowledge, these are the first general tables for PHFN. In Tables 6–9, we report results for $t = v$ with $t \in \{3, 4, 5, 6\}$. Many further tables from our computations are online at `www.phftables.com`. It is obviously not space-conscious to give 500 000 results for every t and v, and fortunately there is no need to do so. Let $\rho(N; t, v)$ be the largest k for which $\mathsf{PHFN}(k, v, t) \leq N$. As k increases, for many consecutive numbers of factors, the perfect hash family number does not change. Therefore reporting those values of $\rho(N;t,v)$ for which $\rho(N;t,v) > \rho(N-1;t,v)$, along with the corresponding value of N, enables one to determine all perfect hash family numbers when k is no larger than the largest $\rho(N;t,v)$ value tabulated. Since the exact values for perfect hash family numbers are unknown in general, we in fact report lower bounds on $\rho(N;t,v)$.

The authorities used in Tables 6–9 are as follows:

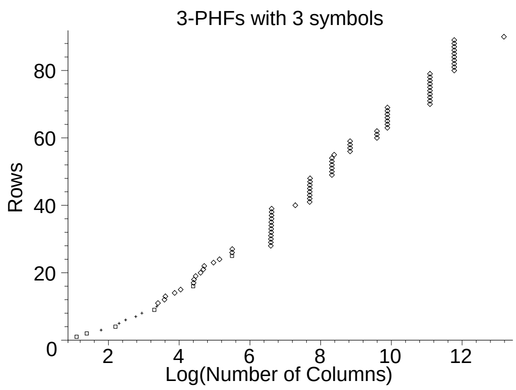
$+$	Column increase	k	Kronecker product
1	$\mathsf{PHF}(1;v,v,t)$	ℓ	Roux-type
c	Composition	m	MOLS or MOLR
f	First N	t	Tabu search
i	IPP codes		

7 Conclusions

Perfect hash families admit a wide variety of constructions; here we have added Roux-type recursive constructions, and the use of integer sequences with no three-term arithmetic progression, to the tools available. However with the richness of constructions, it becomes problematic to determine whether a specific PHF is implied by the available constructions. Hence we have provided a tool for making tables of the best available bounds.

Constructing tables for perfect hash families is beneficial in several ways. First and foremost, it provides people in need of a perfect hash family of specific parameters a resource to find out how to construct the object they need. Second, it causes one to ask questions they might not otherwise ask. The strength three juxtaposition construction and the column increase constructions were created based on a specific need for the tables. Questions about what is possible with three rows were motivated by patterns which emerged from computational search results. Thirdly, and perhaps most importantly, beating the current "world record" is an intriguing challenge.

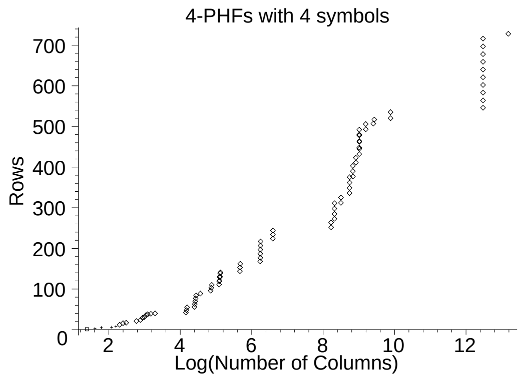
3	1^1	37	13^1
4	2^f	48	14^f
6	3^t	57	15^t
9	4^m	81	16^m
10	5^t	82	17^t
12	6^t	83	18^t
16	7^t	87	19^t
19	8^t	100	20^t
27	9^i	108	21^i
29	10^t	111	22^t
30	11^+	144	23^+
36	12^c	171	24^c



243	25^i	746	39^+	4100	53^+	19687	67^+	130322	81^+
244	26^+	1458	40^ℓ	4101	54^+	19688	68^+	130323	82^+
245	27^+	2187	41^ℓ	4374	55^ℓ	19689	69^+	130324	83^+
729	28^c	2188	42^+	6859	56^c	65536	70^c	130325	84^+
730	29^+	2189	43^+	6860	57^+	65537	71^+	130326	85^+
731	30^+	2190	44^ℓ	6861	58^+	65538	72^+	130327	86^+
732	31^+	2193	45^ℓ	6862	59^+	65539	73^+	130328	87^+
736	32^c	2196	46^ℓ	14642	60^c	65540	74^+	130338	88^c
737	33^+	2208	47^ℓ	14643	61^+	65541	75^+	130339	89^+
738	34^+	2211	48^ℓ	14644	62^+	65542	76^+	531441	90^c
739	35^+	4096	49^c	19683	63^c	65550	77^c		
743	36^c	4097	50^+	19684	64^+	65551	78^+		
744	37^+	4098	51^+	19685	65^+	65552	79^+		
745	38^+	4099	52^+	19686	66^+	130321	80^c		

Table 6. Upper bounds of $\mathsf{PHFN}(k,3,3)$

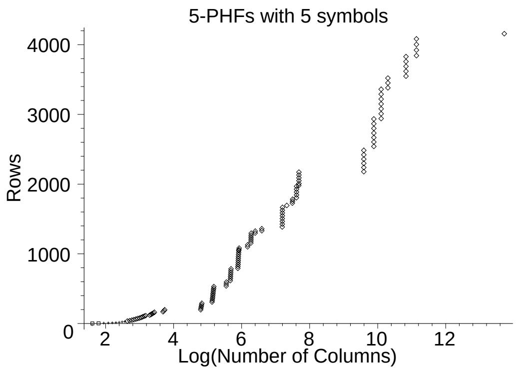
4	1^{1}	21	36^{1}
5	3^{t}	22	38^{t}
6	5^{t}	24	39^{t}
8	6^{t}	27	40^{t}
9	8^{t}	64	42^{t}
10	12^{+}	65	48^{+}
11	16^{+}	66	55^{+}
12	17^{k}	81	56^{k}
16	21^{ℓ}	82	63^{ℓ}
18	24^{ℓ}	83	70^{ℓ}
19	29^{+}	84	77^{+}
20	31^{ℓ}	85	84^{ℓ}



96	89^{k}	292	162^{+}	4097	285^{+}	7450	423^{ℓ}	19683	520^{c}
128	96^{c}	512	168^{c}	4098	298^{+}	8192	432^{ℓ}	19684	535^{+}
130	103^{ℓ}	513	177^{+}	4099	311^{+}	8193	445^{+}	262144	546^{c}
132	110^{ℓ}	514	187^{+}	4914	312^{c}	8194	448^{ℓ}	262145	564^{+}
162	111^{ℓ}	515	197^{+}	4915	325^{+}	8195	462^{+}	262146	583^{+}
163	119^{+}	516	207^{+}	6243	336^{c}	8196	464^{ℓ}	262147	602^{+}
164	121^{ℓ}	517	217^{+}	6244	349^{+}	8197	478^{+}	262148	621^{+}
165	129^{+}	729	224^{c}	6245	362^{+}	8198	480^{ℓ}	262149	640^{+}
166	131^{ℓ}	730	234^{+}	6246	375^{+}	8200	492^{k}	262150	659^{+}
167	139^{+}	731	244^{+}	6859	377^{c}	9828	493^{ℓ}	262151	678^{+}
168	141^{ℓ}	3724	252^{c}	6860	390^{+}	9830	506^{ℓ}	262152	697^{+}
290	144^{c}	3725	264^{+}	6861	403^{+}	12168	507^{c}	262153	716^{+}
291	153^{+}	4096	273^{c}	7448	411^{ℓ}	12486	517^{ℓ}	531441	728^{c}

Table 7. Upper bounds of $\mathsf{PHFN}(k,4,4)$

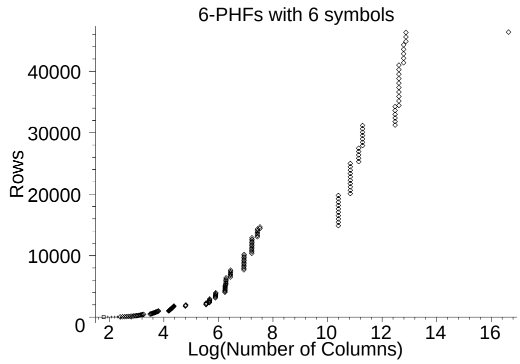
5	1^{1}	17	56^{1}
6	3^{f}	18	64^{f}
7	6^{t}	19	72^{t}
8	8^{t}	20	78^{t}
9	11^{t}	21	87^{t}
10	13^{t}	22	96^{t}
11	18^{t}	23	105^{t}
12	23^{t}	24	114^{t}
13	28^{t}	27	121^{t}
14	35^{+}	28	130^{+}
15	42^{+}	29	140^{+}
16	49^{+}	30	150^{+}



31	161^{+}	289	616^{c}	530	1183^{+}	2024	1845^{+}	19689	2934^{+}
40	169^{c}	290	644^{+}	531	1211^{+}	2025	1886^{+}	24389	2940^{c}
41	183^{+}	291	672^{+}	532	1239^{+}	2026	1927^{+}	24390	3010^{+}
42	197^{+}	292	700^{+}	533	1267^{+}	2027	1968^{+}	24391	3080^{+}
121	198^{c}	293	728^{+}	534	1295^{+}	2166	1980^{k}	24392	3150^{+}
122	221^{+}	294	756^{+}	600	1298^{k}	2172	2008^{k}	24393	3220^{+}
123	244^{+}	295	784^{+}	601	1326^{+}	2173	2049^{+}	24394	3290^{+}
124	267^{+}	361	792^{c}	729	1331^{c}	2174	2090^{+}	24395	3360^{+}
125	290^{+}	362	820^{+}	730	1359^{+}	2175	2131^{+}	29791	3381^{c}
169	308^{c}	363	848^{+}	1331	1386^{c}	2176	2172^{+}	29792	3451^{+}
170	332^{+}	364	876^{+}	1332	1426^{+}	14641	2178^{c}	29793	3521^{+}
171	356^{+}	365	904^{+}	1333	1466^{+}	14642	2238^{+}	50656	3549^{c}
172	380^{+}	366	932^{+}	1334	1506^{+}	14643	2298^{+}	50657	3619^{+}
173	405^{+}	367	960^{+}	1335	1546^{+}	14644	2359^{+}	50658	3689^{+}
174	430^{+}	368	988^{+}	1336	1586^{+}	14645	2421^{+}	50659	3759^{+}
175	455^{+}	369	1016^{+}	1337	1626^{+}	14646	2484^{+}	50660	3829^{+}
176	480^{+}	370	1044^{+}	1338	1666^{+}	19683	2541^{c}	68921	3843^{c}
177	505^{+}	372	1059^{k}	1521	1694^{k}	19684	2604^{+}	68922	3923^{+}
178	530^{+}	375	1082^{k}	1792	1727^{k}	19685	2668^{+}	68923	4003^{+}
256	539^{c}	480	1100^{k}	1799	1755^{k}	19686	2733^{+}	68924	4083^{+}
257	567^{+}	481	1128^{+}	1806	1783^{k}	19687	2799^{+}	912676	4158^{c}
258	595^{+}	529	1155^{c}	2023	1804^{k}	19688	2866^{+}		

Table 8. Perfect Hash Family Numbers $\mathsf{PHFN}(k, 5, 5)$

6	1^1	18	171^1
7	4^t	19	195^t
8	8^t	20	224^t
9	13^t	21	255^t
10	18^t	22	291^t
11	30^+	23	329^+
12	46^+	24	368^+
13	63^+	25	407^+
14	84^+	26	447^+
15	105^+	33	480^+
16	126^+	34	522^+
17	147^+	35	564^+



36	606^+	258	2304^+	630	7632^+	1850	14652^+	79511	30084^+
37	648^+	289	2352^c	1025	7680^c	32769	14880^c	79512	30630^+
38	690^+	290	2496^+	1026	7932^+	32770	15426^+	79513	31176^+
39	732^+	291	2640^+	1027	8184^+	32771	15972^+	262145	31248^c
40	774^+	292	2793^+	1028	8436^+	32772	16518^+	262146	31812^+
41	816^+	293	2955^+	1029	8688^+	32773	17064^+	262147	32395^+
42	858^+	361	3120^c	1030	8940^+	32774	17610^+	262148	32997^+
43	900^+	362	3288^+	1031	9192^+	32775	18156^+	262149	33618^+
44	942^+	363	3456^+	1032	9444^+	32776	18702^+	262150	34258^+
45	984^+	364	3624^+	1033	9696^+	32777	19248^+	300763	34441^c
65	1008^c	365	3792^+	1034	9948^+	32778	19794^+	300764	35169^+
66	1056^+	366	3960^+	1035	10200^+	50653	20088^c	300765	35897^+
67	1111^+	512	4032^c	1369	10368^c	50654	20634^+	300766	36625^+
68	1167^+	513	4200^+	1370	10620^+	50655	21180^+	300767	37353^+
69	1223^+	514	4377^+	1371	10872^+	50656	21726^+	300768	38081^+
70	1279^+	515	4564^+	1372	11124^+	50657	22272^+	300769	38809^+
71	1335^+	516	4761^+	1373	11376^+	50658	22818^+	300770	39537^+
72	1391^+	517	4968^+	1374	11628^+	50659	23364^+	300771	40265^+
73	1447^+	518	5185^+	1375	11880^+	50660	23910^+	300772	40993^+
74	1503^+	529	5264^c	1376	12132^+	50661	24456^+	357911	41385^c
75	1559^+	530	5488^+	1377	12384^+	50662	25002^+	357912	42113^+
76	1615^+	531	5712^+	1378	12636^+	68921	25296^c	357913	42841^+
77	1671^+	532	5936^+	1379	12888^+	68922	25842^+	357914	43569^+
78	1727^+	533	6160^+	1681	13056^c	68923	26388^+	357915	44297^+
79	1783^+	534	6384^+	1682	13308^+	68924	26934^+	389017	44857^c
120	1785^c	625	6512^c	1683	13560^+	68925	27480^+	389018	45585^+
121	1881^+	626	6736^+	1684	13812^+	79507	27900^c	389019	46313^+
122	1977^+	627	6960^+	1685	14064^+	79508	28446^+	16777217	46368^c
256	2016^c	628	7184^+	1686	14316^+	79509	28992^+		
257	2160^+	629	7408^+	1849	14400^c	79510	29538^+		

Table 9. Perfect Hash Family Numbers $\mathsf{PHFN}(k, 6, 6)$

Acknowledgments. Thanks to Sosina Martirosyan for getting us started on this topic.

References

[1] N. Alon, *Explicit construction of exponential sized families of k-independent sets*, Discrete Math. 58 (1986), pp. 191–193.

[2] ———, *Subset sums*, J. Number Theory 27 (1987), pp. 196–205.

[3] N. Alon, T. Kaufman, M. Krivelevich, and D. Ron, *Testing triangle-freeness in general graphs*. Proc. Symposium on Discrete Algorithms (SODA), pp. 279–288, 2006.

[4] M. Atici, S. S. Magliveras, D. R. Stinson, and W. D. Wei, *Some recursive constructions for perfect hash families*, J. Combin. Designs 4 (1996), pp. 353–363.

[5] M. Atici, D. R. Stinson, and R. Wei, *A new practical algorithm for the construction of a perfect hash function*, J. Combin. Math. Combin. Comput. 35 (2000), pp. 127–145.

[6] F. A. Behrend, *On sets of integers which contain no three terms in arithmetical progression*, Proc. Nat. Acad. Sci. U. S. A. 32 (1946), pp. 331–332.

[7] S. R. Blackburn, *Combinatorics and threshold cryptography*, Combinatorial Designs and their Applications, Chapman and Hall, 1999, pp. 49–70.

[8] ———, *Perfect hash families: probabilistic methods and explicit constructions*, J. Comb. Theory (A) 92 (2000), pp. 54–60.

[9] ———, *Perfect hash families with few functions*, unpublished manuscript, 2000.

[10] S. R. Blackburn, M. Burmester, Y. Desmedt, and P. R. Wild, *Efficient multiplicative sharing schemes*, Lecture Notes in Computer Science 1070 (1996), pp. 107–118.

[11] A. P. Bosznay, *On the lower estimation of nonaveraging sets*, Acta Math. Hungar. 53 (1989), pp. 155–157.

[12] J. Bourgain, *On triples in arithmetic progression*, Geom. Funct. Anal. 9 (1999), pp. 968–984.

[13] C. J. Colbourn and J. H. Dinitz (eds.), *The CRC Handbook of Combinatorial Designs*. CRC Press, Boca Raton, FL, 1996.

[14] C. J. Colbourn, S. S. Martirosyan, Tran Van Trung, and R. A. Walker II, *Roux-type constructions for covering arrays of strengths three and four*, Designs, Codes and Cryptography 41 (2006), pp. 35–57.

[15] Z. J. Czech, G. Havas, and B. S. Majewski, *Perfect hashing*, Theor. Comp. Sci. 182 (1997), pp. 1–143.

[16] A. Fiat and M. Naor, *Broadcast encryption*, Lecture Notes in Computer Science 773 (1994), pp. 480–491.

[17] G. Freiman, *Foundations of Structural Theory of Set Addition*, Translations of Mathematical Monographs 37. AMS, 1973.

[18] F. Glover, *Tabu search – Part I*, ORSA J. Comput. 1 (1989), pp. 190–206.

[19] W. T. Gowers, *A new proof of Szemerédi's theorem*, Geom. Funct. Anal. 11 (2001), pp. 465–588.

[20] W. Harvey and T. Winterer, *Solving the MOLR and social golfers problems*, Lecture Notes in Computer Science 3709 (2005), pp. 286–300.

[21] D. R. Heath-Brown, *Integer sets containing no arithmetic progressions*, J. London Math. Soc. (2) 35 (1987), pp. 385–394.

[22] V. F. Lev, *Progression-free sets in finite abelian groups*, J. Number Theory 104 (2004), pp. 162–169.

[23] K. Mehlhorn, *Data Structures and Algorithms 1: Sorting and Searching*. Springer-Verlag, Berlin, 1984.

[24] R. Meshulam, *On subsets of finite abelian groups with no* 3*-term arithmetic progressions*, J. Combin. Theory Ser. A 71 (1995), pp. 168–172.

[25] L. Moser, *On non-averaging sets of integers*, Canadian J. Math. 5 (1953), pp. 245–252.

[26] R. A. Rankin, *Sets of integers containing not more than a given number of terms in arithmetical progression*, Proc. Roy. Soc. Edinburgh Sect. A 65 (1962), pp. 332–344.

[27] K. F. Roth, *On certain sets of integers*, J. London Math. Soc. 28 (1953), pp. 104–109.

[28] D. R. Stinson, Tran van Trung, and R. Wei, *Secure frameproof codes, key distribution patterns, group testing algorithms and related structures*, J. Statist. Plan. Infer. 86 (2000), pp. 595–617.

[29] D. R. Stinson, R. Wei, and L. Zhu, *New constructions for perfect hash families and related structures using combinatorial designs and codes*, J. Combin. Designs 8 (2000), pp. 189–200.

[30] E. Szemerédi, *Integer sets containing no arithmetic progressions*, Acta Math. Hungar. 56 (1990), pp. 155–158.

[31] D. Tonien and R.Safavi-Naini, *Recursive constructions of secure codes and hash families using difference function families*, J. Combinat. Theory (A) 113 (2006), pp. 664–674.

[32] Tran van Trung and S. S. Martirosyan, *On a class of traceability codes*, Designs Codes Crypt. 31 (2004), pp. 125–132.

[33] ———, *New constructions for IPP codes*, Designs Codes Crypt. 35 (2005), pp. 227–239.

[34] R. A. Walker II, *Covering Arrays and Perfect Hash Families*, Ph.D. thesis, Department of Computer Science and Engineering, Arizona State University, USA, 2005.

[35] R. A. Walker II and C. J. Colbourn, *Tabu search for covering arrays using permutation vectors*, submitted, 2006.

[36] I. M. Wanless, *A partial latin squares problem posed by Blackburn*, Bull. Inst. Comb. Appl. 42 (2004), pp. 76–80.

Received 27 March, 2006; revised 27 September, 2006

Author information

Robert A. Walker II, Computer Science and Engineering, Arizona State University, P.O. Box 878809, Tempe, AZ 85287, USA.
Email: robby.walker@gmail.com

Charles J. Colbourn, Computer Science and Engineering, Arizona State University, P.O. Box 878809, Tempe, AZ 85287, USA.
Email: charles.colbourn@asu.edu