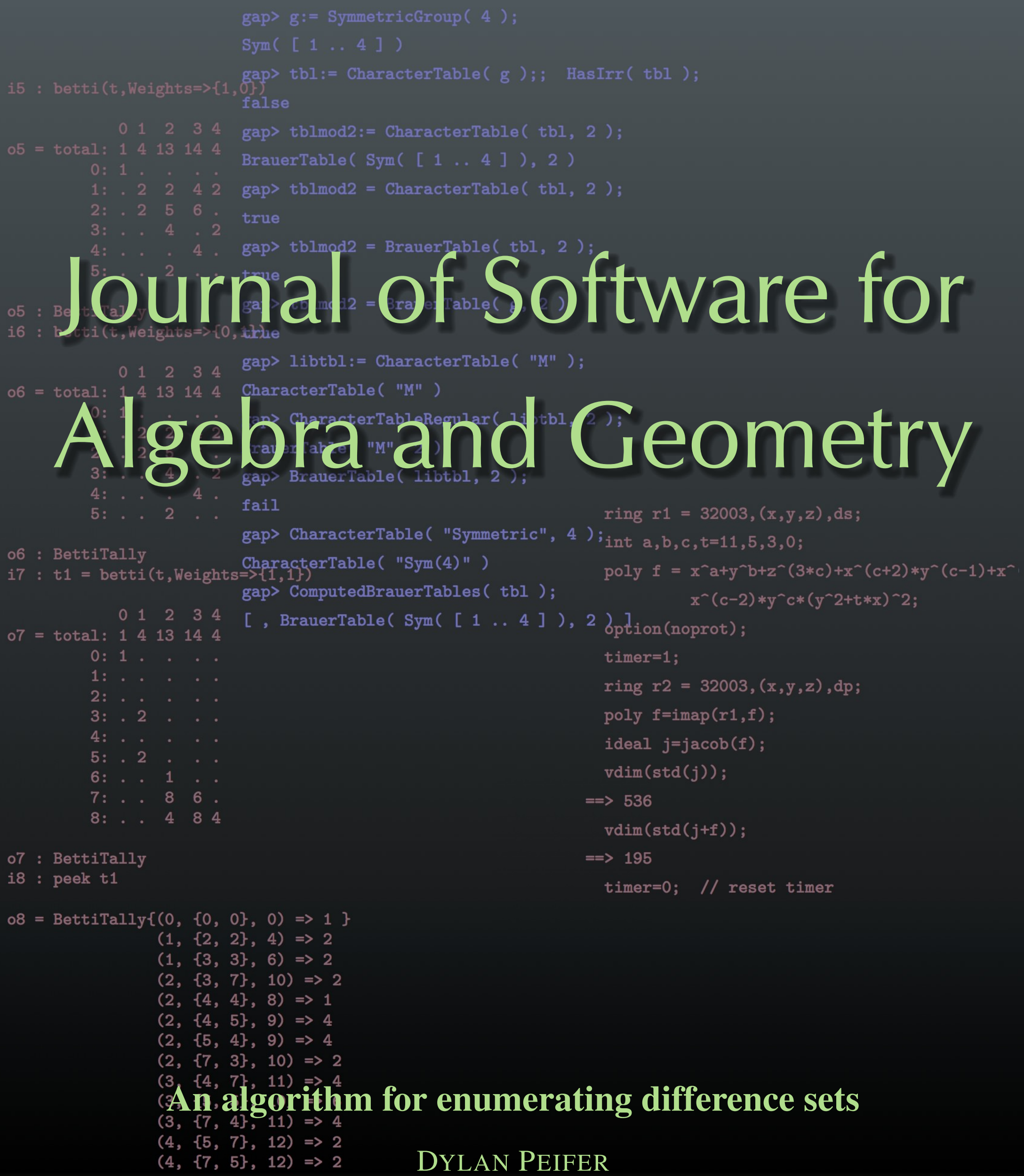
Journal of Software for Algebra and Geometry



```
i5 : betti(t,Weights=>{1,0})

             0 1  2  3 4
o5 = total: 1 4 13 14 4
         0: 1 .  .  . .
         1: . 2  2  4 2
         2: . 2  5  6 .
         3: . .  4  . 2
         4: . .  .  4 .
         5: . .  2  . .

o5 : BettiTally
i6 : betti(t,Weights=>{0,1})

             0 1  2  3 4
o6 = total: 1 4 13 14 4
         0: 1 .  .  . .
         1: . 2  2  4 2
         2: . 2  5  6 .
         3: . .  4  . 2
         4: . .  .  4 .
         5: . .  2  . .

o6 : BettiTally
i7 : t1 = betti(t,Weights=>{1,1})

             0 1  2  3 4
o7 = total: 1 4 13 14 4
         0: 1 .  .  . .
         1: . .  .  . .
         2: . .  .  . .
         3: . 2  .  . .
         4: . .  .  . .
         5: . 2  .  . .
         6: . .  1  . .
         7: . .  8  6 .
         8: . .  4  8 4

o7 : BettiTally
i8 : peek t1

o8 = BettiTally{(0, {0, 0}, 0) => 1 }
                (1, {2, 2}, 4) => 2
                (1, {3, 3}, 6) => 2
                (2, {3, 7}, 10) => 2
                (2, {4, 4}, 8) => 1
                (2, {4, 5}, 9) => 4
                (2, {5, 4}, 9) => 4
                (2, {7, 3}, 10) => 2
                (3, {4, 7}, 11) => 4
                [illegible]
                (3, {7, 4}, 11) => 4
                (4, {5, 7}, 12) => 2
                (4, {7, 5}, 12) => 2
```

```
gap> g:= SymmetricGroup( 4 );
Sym( [ 1 .. 4 ] )
gap> tbl:= CharacterTable( g );;  HasIrr( tbl );
false
gap> tblmod2:= CharacterTable( tbl, 2 );
BrauerTable( Sym( [ 1 .. 4 ] ), 2 )
gap> tblmod2 = CharacterTable( tbl, 2 );
true
gap> tblmod2 = BrauerTable( tbl, 2 );
true
gap> tblmod2 = BrauerTable( g, 2 );
true
gap> libtbl:= CharacterTable( "M" );
CharacterTable( "M" )
gap> CharacterTableRegular( libtbl, 2 );
BrauerTable( "M", 2 )
gap> BrauerTable( libtbl, 2 );
fail
gap> CharacterTable( "Symmetric", 4 );
CharacterTable( "Sym(4)" )
gap> ComputedBrauerTables( tbl );
[ , BrauerTable( Sym( [ 1 .. 4 ] ), 2 ) ]
```

```
  ring r1 = 32003,(x,y,z),ds;
  int a,b,c,t=11,5,3,0;
  poly f = x^a+y^b+z^(3*c)+x^(c+2)*y^(c-1)+x^
           x^(c-2)*y^c*(y^2+t*x)^2;
  option(noprot);
  timer=1;
  ring r2 = 32003,(x,y,z),dp;
  poly f=imap(r1,f);
  ideal j=jacob(f);
  vdim(std(j));
==> 536
  vdim(std(j+f));
==> 195
  timer=0;  // reset timer
```

An algorithm for enumerating difference sets

Dylan Peifer

vol 9

2019

JSAG **9** (2019), 35–41
dx.doi.org/10.2140/jsag.2019.9.35

The Journal of Software for
Algebra and Geometry

An algorithm for enumerating difference sets

DYLAN PEIFER

ABSTRACT: The `DifSets` package for GAP implements an algorithm for enumerating all difference sets in a group up to equivalence and provides access to a library of results. The algorithm functions by finding difference sums, which are potential images of difference sets in quotient groups of the original group, and searching their preimages. In this way, the search space can be dramatically decreased, and searches of groups of relatively large order (such as order 64 or order 96) can be completed.

1. INTRODUCTION. Let G be a finite group of order v and D a subset of G with k elements. Then D is a (v, k, λ)*-difference set* if each nonidentity element of G can be written as $d_i d_j^{-1}$ for $d_i, d_j \in D$ in exactly λ different ways. Difference sets were first studied in relation to finite geometries [Singer 1938] and have connections to symmetric designs, coding theory, and many other fields of mathematics [Moore and Pollatsek 2013; Davis and Jedwab 1996; Colbourn and Dinitz 1996; Beth et al. 1999].

Large libraries of difference sets are useful for developing conjectures and building examples. Gordon provides an extensive library of difference sets in abelian groups [Gordon], but has no results for nonabelian groups, which do show distinct behavior [Smith 1995]. A wide variety of techniques can be used to construct difference sets for these libraries (see, for example, [Dillon 1985] and [Davis and Jedwab 1997]), but fully enumerating all difference sets in a given group requires some amount of exhaustive search, which can quickly become computationally infeasible. Kibler [1978] performed the first major exhaustive enumeration of difference sets, and considered groups where difference sets could be found with $k < 20$. In recent years, AbuGhneim [2013; 2016] has performed almost complete enumerations for all groups of order 64, and several authors have found difference sets in groups of order 96 [Golemac et al. 2005; 2007; AbuGhneim and Smith 2007]. The `DifSets` package for the computer algebra system [GAP] efficiently and generally implements the techniques used by these and many other authors to

MSC2010: 05B10.
Keywords: difference sets, exhaustive search, GAP.
`DifSets` version 2.2.0

exhaustively enumerate all difference sets up to equivalence in a group. With the package loaded, a search of a given group can be performed with a single command.

```
gap> DifferenceSets(CyclicGroup(7));
[ [ 1, 2, 4 ] ]
```

The package has been used to give the first complete enumeration of all difference sets up to equivalence in groups of order 64 and 96, and in total provides a library of results for 1006 of the 1032 groups of order less than 100. Results are organized by their id in the `SmallGroups` library [SmallGrp] and can be easily loaded by GAP.

```
gap> LoadDifferenceSets(16,5);  # results for SmallGroup(16,5)
[ [ 1, 2, 3, 4, 8, 15 ], [ 1, 2, 3, 4, 11, 13 ] ]
```

The ease of use of these top-level functions is the primary interface difference between the `DifSets` package and a similar GAP package [RDS], which provides a variety of tools to search for difference sets. The functions involving coset signatures in `RDS` provide similar functionality to the `DifSets` package, but require substantial user interaction to perform efficient searches, and are not feasible for searching most groups of order 64 and 96. In addition, `RDS` provides no precomputed results, though it does provide significant additional functionality related to relative difference sets, partial difference sets, and projective planes.

2. Difference sets.

For notational purposes it is useful to consider a subset $D \subseteq G$ as an element of the group ring $\mathbb{Z}[G]$. We will abuse notation to define the group ring elements

$$G = \sum_{g \in G} g, \quad D = \sum_{d \in D} d, \quad D^{(-1)} = \sum_{d \in D} d^{-1}, \quad gD = \sum_{d \in D} gd, \quad D^{\phi} = \sum_{d \in D} \phi(d),$$

where $g \in G$ and ϕ is a homomorphism from G. Then the statement that D is a (v, k, λ)-difference set is equivalent to the equation

$$DD^{(-1)} = (k - \lambda)1_G + \lambda G,$$

where D is an element of $\mathbb{Z}[G]$ with coefficients in $\{0, 1\}$. With this definition it is a quick exercise to prove the following (see page 298 of [Beth et al. 1999] and Theorem 4.2 and 4.11 of [Moore and Pollatsek 2013]).

Proposition 1. *Let G be a group of order v. Then*:

(1) *Any one element subset of G is a $(v, 1, 0)$-difference set.*

(2) *The complement of a (v, k, λ)-difference set in G is a $(v, v - k, \lambda + v - 2k)$-difference set in G.*

(3) *If D is a (v, k, λ)-difference set in G, $g \in G$, and $\phi \in \mathrm{Aut}(G)$, then gD^{ϕ} is also a (v, k, λ)-difference set in G.*

In addition, an immediate consequence of the definition is that $k(k-1) = \lambda(v-1)$ for any valid set of parameters of a difference set, so that for a given value of v there are typically only a few possible values of k and λ. More sophisticated results, such as the Bruck–Ryser–Chowla theorem, can reduce the number of possibilities even further.

As a result of Proposition 1, in enumerating difference sets we ignore the trivial one element difference sets, only take the smaller of each complementary pair of sets, and only consider sets distinct up to an equivalence given by part (3).

Definition 2. Let D_1 and D_2 be difference sets in G. Then D_1 and D_2 are *equivalent difference sets* if $D_1 = gD_2^{\phi}$ for some $g \in G$ and $\phi \in \mathrm{Aut}(G)$.

In the `DifSets` package, difference sets are stored as lists of integers. These integers represent indices in the list returned by the GAP function `Elements(G)`, which is a sorted[1] list of elements of the group `G`. For example, consider the group $C_7 = \langle x \mid x^7 = 1\rangle$. In GAP we have

```
gap> C7 := CyclicGroup(7);;
gap> Elements(C7);
[ <identity> of ..., f1, f1^2, f1^3, f1^4, f1^5, f1^6 ]
```

where clearly `f1` is the generator corresponding to our x. Then the subset $D = \{x, x^2, x^4\}$ corresponds to the set consisting of the second, third, and fifth elements of `Elements(C7)`, which we can represent in indices as `[2, 3, 5]`. We can check that this is a difference set and also note that it is equivalent to the difference set $xD = \{x^2, x^3, x^5\}$, which is represented as `[3, 4, 6]`.

```
gap> IsDifferenceSet(C7, [2, 3, 5]);
true
gap> IsEquivalentDifferenceSet(C7, [2, 3, 5], [3, 4, 6]);
true
```

3. DIFFERENCE SUMS. A basic method for enumerating all difference sets in a group G is to enumerate all subsets of G and check if each is a difference set by definition. But since the number of subsets in a group is exponential in its order, we cannot feasibly enumerate and test all subsets for groups of even a modest size. The key to decreasing the search space is the following well-known lemma, which motivates our definition of a *difference sum*.[2]

[1] Element comparison (and thus the list `Elements(G)`) is instance-independent in GAP for permutation and pc groups, which includes, for example, all groups in the `SmallGroups` library.

[2] Concepts similar to difference sums are elsewhere referred to as difference lists, intersection numbers, or signatures. However, difference sums require both a group G and normal subgroup N, not just the group structure of the quotient G/N used in some other definitions. This precision is needed for specifying induced automorphisms in Definition 7 so that we can prove Lemma 8.

<table>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>G/N_3</td></tr>
<tr><td colspan="3">3</td><td colspan="3">1</td><td colspan="3">1</td><td colspan="3">1</td><td colspan="3">1</td><td>G/N_2</td></tr>
<tr><td colspan="15">7</td><td>G/N_1</td></tr>
</table>

Figure 1. A difference set of size 7 in the group G of order 15 and the difference sums it induces in G/N_i where $G = N_1 \rhd N_2 \rhd N_3 = \{1\}$. Each row in the diagram is a group, with each block a coset.

Lemma 3. *Suppose D is a (v, k, λ)-difference set in G and θ is a homomorphism of G with* $|\ker(\theta)| = w$. *Let $S = D^\theta$ and $H = G^\theta$. Then*

$$SS^{(-1)} = (k - \lambda)1_H + \lambda w H.$$

Definition 4. Given a finite group G and normal subgroup N, a (v, k, λ)-*difference sum* is an element S of $\mathbb{Z}[G/N]$ such that $SS^{(-1)} = (k - \lambda)1_{G/N} + \lambda|N|G/N$ and the coefficients of S have values in $\{0, 1, \dots, |N|\}$.

By construction, any difference set in G induces difference sums under the natural projection in quotients of G, as seen in Figure 1. Precisely, we have:

Lemma 5. *Suppose G is a finite group with normal subgroup N and natural projection $\pi : G \to G/N$. Then any (v, k, λ)-difference set D in G induces a (v, k, λ)-difference sum D^π in G/N.*

Lemma 6. *Suppose G is a finite group with normal subgroups N_1 and N_2 such that $N_2 \subseteq N_1$ and $\pi : G/N_2 \to G/N_1$ is the natural projection. Then any (v, k, λ)-difference sum S in G/N_2 induces a (v, k, λ)-difference sum S^π in G/N_1.*

Lemma 5 means that our search for difference sets only requires checking the subsets of G that induce difference sums in some quotient. In finding these difference sums, Lemma 6 additionally allows us to only test sums that induce difference sums in further quotients. In each case the search space is dramatically decreased. Since our search is for difference sets up to equivalence, we also define a complementary equivalence of difference sums such that equivalent difference sums are induced by equivalent collections of difference sets.

Definition 7. Let S_1 and S_2 be difference sums in G/N. Then S_1 and S_2 are *equivalent difference sums* if $S_1 = gS_2^\phi$ for some $g \in G/N$ and ϕ an automorphism of G/N induced by an automorphism of G.

Lemma 8. *Suppose S_1 and S_2 are equivalent difference sums in G/N. Then if D_1 is any difference set in G that induces S_1, there exists a difference set D_2 in G that induces S_2 such that D_1 and D_2 are equivalent.*

In the `DifSets` package, difference sums are stored as lists of integers representing the coefficients of the group ring elements, with position in the list given by the position of the coset in the list returned by the GAP function `Elements(G/N)`. For example, `[3, 1, 1, 1, 1]` represents a difference sum in `SmallGroup(15, 1)` mod its normal subgroup of order 3 with coefficient 3 on the identity coset and coefficient 1 on all other cosets.

```
gap> G := SmallGroup(15, 1);; N := NormalSubgroups(G)[2];;
gap> IsDifferenceSum(G, N, [3, 1, 1, 1, 1]);
true
```

4. ALGORITHM. The basic structure of the algorithm is to start at the bottom of Figure 1 and travel upwards. Given a group G, first compute $v = |G|$ and then find all values of k that give solutions satisfying the Bruck–Ryser–Chowla theorem to the equation $k(k-1) = \lambda(v-1)$ mentioned in Section 2. For example,

```
gap> G := SmallGroup(15, 1);;
gap> PossibleDifferenceSetSizes(G);
[ 7 ]
```

Each value of k will be handled separately. The algorithm starts with the normal subgroup $N_1 = G$, where the only difference sum of size k in $G/N_1 = \{1\}$ is `[k]`.

```
gap> N1 := G;;
gap> difsums := [ [7] ];;
```

Given a normal subgroup N_2 of G such that $N_2 \subseteq N_1$, first enumerate all preimages in G/N_2 of current difference sums in G/N_1 and return those that are themselves difference sums. Then remove all but one representative of each equivalence class from this collection.

```
gap> N2 := NormalSubgroups(G)[2];;
gap> difsums := AllRefinedDifferenceSums(G, N1, N2, difsums);
[ [ 1, 1, 1, 1, 3 ], [ 1, 1, 1, 3, 1 ], [ 1, 1, 3, 1, 1 ],
  [ 1, 3, 1, 1, 1 ], [ 3, 1, 1, 1, 1 ] ]
gap> difsums := EquivalentFreeListOfDifferenceSums(G, N2, difsums);
[ [ 3, 1, 1, 1, 1 ] ]
```

In the general case, the above step is repeated along a chief series

$$G = N_1 \rhd \cdots \rhd N_r = \{1\}$$

of G with N_{r-1} a nontrivial normal subgroup of minimal possible size in G. At N_{r-1}, enumerate sets and remove equivalents to leave the final result.

```
gap> difsets := AllRefinedDifferenceSets(G, N2, difsums);
[ [ 1, 2, 4, 3, 8, 11, 12 ], [ 1, 2, 4, 3, 10, 13, 12 ],
  [ 1, 2, 4, 5, 6, 9, 14 ], [ 1, 2, 4, 5, 10, 13, 14 ],
  [ 1, 2, 4, 7, 6, 9, 15 ], [ 1, 2, 4, 7, 8, 11, 15 ] ]
gap> difsets := EquivalentFreeListOfDifferenceSets(G, difsets);
[ [ 1, 2, 4, 7, 8, 11, 15 ] ]
```

These steps are encapsulated in the function `DifferenceSets` mentioned in Section 1, with two modifications. First, since every difference set is equivalent to some difference set containing the identity, the algorithm does not enumerate some preimages that are guaranteed to be equivalent to others. Second, the final elimination of all but one representative of equivalence classes of difference sets uses the `SmallestImageSet` function [Linton 2004] from the GAP package [GRAPE]. Although roughly 20% slower than the function given above for most cases, `SmallestImageSet` gives a unique minimal result and handles groups with large automorphism groups much more efficiently.

5. Results.

The `DifSets` package successfully computed results for 1006 of the 1032 groups of order less than 100, including all groups of order 64 and 96. Full results with timings and comments can be found in the package and its documentation. Here we include a summary for order 64 and 96. All computations were performed with GAP 4.9.1 on a 4.00GHz i7-6700K using 8GB of RAM.

Order	Groups	Difference sets	Median time per group	Total time
64	267	330159	0.415 hours	295.811 hours
96	231	2627	3.133 hours	1568.746 hours

Timing comparisons with the `RDS` package mentioned in Section 1 are difficult since `RDS` provides a variety of tools rather than a single algorithm. Ordered coset signatures in `RDS` correspond to difference sums in `DifSets`, but, unlike difference sums, coset signatures cannot be refined through multiple stages, which makes the generation of good coset signatures in `RDS` infeasible for most order 64 and order 96 groups. However, if an ordered signature is available, building difference sets through partial difference sets in `RDS` can in some cases be much faster than searching the corresponding difference sum using `DifSets`. In particular, replacing the final step in Section 4 with a search using `RDS` can significantly improve times for some groups of order 96. Further work to combine the refining of difference sums used by `DifSets` and the generation of difference sets through partial difference sets used by `RDS` could lead to significantly better times than either package could manage alone.

Acknowledgements. The author thanks Ken Smith, Alexander Hulpke, and an anonymous reviewer for helpful comments that improved the `DifSets` package and this article.

Supplement. The online supplement contains version 2.2.0 of `DifSets`.

References.

[AbuGhneim 2013] O. AbuGhneim, “On (64, 28, 12) difference sets”, *Ars Combin.* **111** (2013), 401–419. MR Zbl

[AbuGhneim 2016] O. AbuGhneim, “All (64, 28, 12) difference sets and related structures”, *Ars Combin.* **125** (2016), 271–285. MR Zbl

[AbuGhneim and Smith 2007] O. AbuGhneim and K. Smith, “Nonabelian groups with (96, 20, 4) difference sets”, *Electron. J. Combin.* **14**:1 (2007), art. id. R8, 17. MR Zbl

[Beth et al. 1999] T. Beth, D. Jungnickel, and H. Lenz, *Design theory, Vol. I*, 2nd ed., Encyclopedia of Mathematics and its Applications **69**, Cambridge University Press, 1999. MR Zbl

[Colbourn and Dinitz 1996] C. J. Colbourn and J. H. Dinitz (editors), *The CRC handbook of combinatorial designs*, CRC Press, Boca Raton, FL, 1996. MR Zbl

[Davis and Jedwab 1996] J. A. Davis and J. Jedwab, “A survey of Hadamard difference sets”, pp. 145–156 in *Groups, difference sets, and the Monster* (Columbus, OH, 1993), edited by K. T. Arasu et al., Ohio State Univ. Math. Res. Inst. Publ. **4**, de Gruyter, Berlin, 1996. MR Zbl

[Davis and Jedwab 1997] J. A. Davis and J. Jedwab, “A unifying construction for difference sets”, *J. Combin. Theory Ser. A* **80**:1 (1997), 13–78. MR Zbl

[Dillon 1985] J. F. Dillon, “Variations on a scheme of McFarland for noncyclic difference sets”, *J. Combin. Theory Ser. A* **40**:1 (1985), 9–21. MR Zbl

[GAP] The GAP Group, “GAP – Groups, Algorithms, and Programming”, available at https://www.gap-system.org.

[Golemac et al. 2005] A. Golemac, T. Vučičić, and J. Mandić, “One (96, 20, 4)-symmetric design and related nonabelian difference sets”, *Des. Codes Cryptogr.* **37**:1 (2005), 5–13. MR

[Golemac et al. 2007] A. Golemac, J. Mandić, and T. Vučičić, “On the existence of difference sets in groups of order 96”, *Discrete Math.* **307**:1 (2007), 54–68. MR Zbl

[Gordon] D. Gordon, “La Jolla difference set repository”, available at https://www.dmgordon.org/diffset/.

[GRAPE] L. H. Soicher, “GRAPE – GRaph Algorithms using PErmutation groups”, GAP package version 4.7, available at http://www.maths.qmul.ac.uk/∼leonard/grape/.

[Kibler 1978] R. E. Kibler, “A summary of noncyclic difference sets, $k < 20$”, *J. Combinatorial Theory Ser. A* **25**:1 (1978), 62–67. MR Zbl

[Linton 2004] S. Linton, “Finding the smallest image of a set”, pp. 229–234 in *ISSAC 2004*, edited by J. Gutierrez, ACM, New York, 2004. MR Zbl

[Moore and Pollatsek 2013] E. H. Moore and H. S. Pollatsek, *Difference sets: connecting algebra, combinatorics, and geometry*, Student Mathematical Library **67**, American Mathematical Society, Providence, RI, 2013. MR

[RDS] M. Roeder, “RDS – a package for searching relative difference sets”, GAP package version 1.6, available at http://csserver.evansville.edu/∼mroeder.

[Singer 1938] J. Singer, “A theorem in finite projective geometry and some applications to number theory”, *Trans. Amer. Math. Soc.* **43**:3 (1938), 377–385. MR Zbl

[SmallGrp] E. O. B. Eick, H. U. Besche, “SmallGrp – the GAP small groups library”, GAP package, version 1.3, available at https://gap-packages.github.io/smallgrp/.

[Smith 1995] K. W. Smith, “Non-abelian Hadamard difference sets”, *J. Combin. Theory Ser. A* **70**:1 (1995), 144–156. MR Zbl

RECEIVED: 5 Jul 2018 REVISED: 4 Sep 2018 ACCEPTED: 26 Feb 2019

DYLAN PEIFER:
djp282@cornell.edu
Department of Mathematics, Cornell University, Ithaca, NY, United States