On classifying objects with specified groups of automorphisms, friendly subgroups, and Sylow tower groups

Leonard H. Soicher

School of Mathematical Sciences
Queen Mary, University of London
Mile End Road, London E1 4NS, UK

email: L.H.Soicher@qmul.ac.uk

December 16, 2008

Abstract

We describe some group theory which is useful in the classification of combinatorial objects having given groups of automorphisms. In particular, we show the usefulness of the concept of a friendly subgroup: a subgroup H of a group K is a *friendly* subgroup of K if every subgroup of K isomorphic to H is conjugate in K to H. We explore easy-to-test sufficient conditions for a subgroup H to be a friendly subgroup of a finite group K, and for this, present an algorithm for determining whether a finite group H is a Sylow tower group.

1 Introduction

The following situation is common when classifying combinatorial objects. We have some appropriate finite G-set Ω, and we want to classify the elements of Ω satisfying a given property, up to the action of G. For example, suppose we are classifying t-(v, k, λ) designs up to isomorphism, for some given t, v, k, λ. Then $G = \mathrm{Sym}(v)$ and Ω can be taken to be the set of all size $b := \lambda\binom{v}{t}/\binom{k}{t}$ multisets of k-subsets of $V := \{1, \ldots, v\}$. The action of G on Ω is the natural one on multisets of subsets of V (the natural action on multisets is like the natural action on sets but with multiplicities preserved), and the property we require of an element $\mathcal{B} \in \Omega$ is that every t-subset of V is contained in exactly λ elements of $\mathcal{B}$; that is, we require that $(V, \mathcal{B})$ is a t-(v, k, λ) design. Often a complete classification is too difficult, and we impose the additional condition that each object we seek is invariant under at least one of some specified (non-conjugate) subgroups of G. This is the situation we consider in this paper (see also [2, 5, 8, 9]). We are especially interested in the avoidance of unnecessary G-equivalence checks of objects (which are usually done via the determination of canonical G-orbit representatives of the given objects (see [10, 11])), when usually much easier $N_G(H)$-equivalence checks suffice.

Our approach is group-theoretical, and we show the usefulness of friendly subgroups: a subgroup H of a group K is a *friendly* subgroup of K if every subgroup of K isomorphic to H is conjugate in K to H. (Our approach generalizes one of Laue's methods described in [9], in which Sylow subgroups are employed in a similar way to one way we employ friendly subgroups.) Next, we explore easy-to-test sufficient conditions for a subgroup H to be a friendly subgroup of a finite group K, and for this, present an algorithm for determining whether a finite group H is a Sylow tower group.

The theory in this paper is implemented in the DESIGN package [12]. This GAP [3] package can construct, classify, partition and study block designs satisfying a very wide range of user-specified properties, including being invariant under a given group H of automorphisms. The designs may be t-designs (with or without repeated blocks), but in general need not have constant block-size nor constant replication-number.

2 Friendly subgroups

The usefulness of friendly subgroups is based on the following two results.

Proposition 2.1 *Suppose G acts on a set Ω, and let $\mathcal{A}, \mathcal{B} \in \Omega$, with H a friendly subgroup of $G_{\mathcal{A}}$ (the stabilizer in G of $\mathcal{A}$), and H a subgroup of $G_{\mathcal{B}}$. Then $\mathcal{A}$ and $\mathcal{B}$ are in the same G-orbit if and only if they are in the same $N_G(H)$-orbit.*

Proof. The if-part is trivial. For the converse, suppose $x \in G$ with $\mathcal{A}^x = \mathcal{B}$. Then $G_{\mathcal{B}} = (G_{\mathcal{A}})^x = x^{-1}G_{\mathcal{A}}x$, and so H^x is a friendly subgroup of $G_{\mathcal{B}}$. Since $H \leq G_{\mathcal{B}}$, it must be conjugate in $G_{\mathcal{B}}$ to H^x, and so there is a $y \in G_{\mathcal{B}}$ with $H^{xy} = H$. We thus have $xy \in N_G(H)$ and $\mathcal{A}^{xy} = \mathcal{B}^y = \mathcal{B}$. ■

Proposition 2.2 *Suppose G acts on a set Ω, and let $\mathcal{A}, \mathcal{B} \in \Omega$, with H a friendly subgroup of $G_{\mathcal{A}}$. Then if $\mathcal{B}$ is in the same G-orbit as $\mathcal{A}$, every subgroup of $G_{\mathcal{B}}$ that is isomorphic to H is conjugate in G to H.*

Proof. Suppose $x \in G$ with $\mathcal{A}^x = \mathcal{B}$. Then $G_{\mathcal{B}} = (G_{\mathcal{A}})^x$, and so H^x is a friendly subgroup of $G_{\mathcal{B}}$. Thus, if $J \leq G_{\mathcal{B}}$ with $J \cong H$, then J is conjugate in $G_{\mathcal{B}}$ to H^x, and so J is conjugate in G to H. ■

When classifying H-invariant objects up to G-equivalence (that is, up to being in the same G-orbit), for a given $H \leq G$, one often first classifies the objects up to $N_G(H)$-equivalence (see [2, 12]). Proposition 2.1 allows us to avoid many tests to determine G-equivalence when $N_G(H)$-orbit representatives of the H-invariant objects have already been determined. For such an $N_G(H)$-orbit representative $\mathcal{A}$, if H is a friendly subgroup of $G_{\mathcal{A}}$ then no G-equivalence tests involving $\mathcal{A}$ are required.

When classifying objects such that each is invariant under at least one of the groups in a given set $\{H_1, \ldots, H_m\}$ of pairwise isomorphic, but non-conjugate, subgroups of G (such as a set of conjugacy class representatives of the subgroups of some prime order p), Proposition 2.2 allows us to avoid many tests to determine when an H_i-invariant object is G-equivalent to an H_j-invariant one. For a given H_i-invariant $\mathcal{A}$, if H_i is a friendly subgroup of $G_{\mathcal{A}}$, then $\mathcal{A}$ cannot be in the same G-orbit as an H_j-invariant object, when $i \neq j$.

We note that, unlike the methods given in [9], we generally need to be able to explicitly compute the G-stabilizers (automorphism groups) of the objects we classify, but unlike [9], we do not have to list *all* objects invariant under a given H. Depending on the problem under consideration, determining G-stabilizers seems often to be easier in practice than determining canonical G-orbit representatives (which is what we aim to avoid having to do). For example, in GAP, determining stabilizers of sets is faster in practice than Linton's important procedure [10] to find the lexicographically least set in a G-orbit of sets. On the other hand, computing the automorphism group of a graph and canonically labelling that graph using nauty [11] appears to take much the same time. It is, however, sometimes possible to avoid computing G-stabilizers altogether, for we may be able to deduce that a subgroup J of our our assumed group of automorphisms H must be a friendly subgroup of any possible automorphism group of the objects we seek (see [9] for some instances of this in the case of the objects being t-(v, k, λ) designs and J being a Sylow p-subgroup of H for certain primes p). Then Proposition 2.1 tells us that to classify our H-invariant objects up to isomorphism, it suffices to classify them up to $N_G(J)$-equivalence.

3 Verifying friendliness

Given finite groups H and K, with H a subgroup of K, it is often possible to use relatively cheap computational tests to verify that H is a friendly subgroup of K (when it is such a subgroup). Typically, in our applications, H is small and fixed for many overgroups K, and each such K is a permutation group of relatively low degree. One useful test for H is to determine whether it is a Sylow tower group (defined below), since a Sylow tower group which is a Hall subgroup of a finite group K is a friendly subgroup of K (see Theorem 3.1).

Suppose H is a finite group, and $1 = T_0 < T_1 < ... < T_k = H$ is a normal series for H, such that, for $i = 1, \ldots, k$, T_i/T_{i-1} is isomorphic to a Sylow p_i-subgroup of H, for some prime p_i dividing $|H|$. Then H is called a *Sylow tower group* having *complexion* $(p_1, \ldots, p_k)$ (unlike some authors, we do not require that a Sylow tower group has a complexion $(p_1, \ldots, p_k)$ satisfying $p_1 > \cdots > p_k$). In the next section we give an algorithm to determine whether a given finite group is a Sylow tower group. Note that if a finite group H is a Sylow tower group then H is soluble, but the converse does not hold in general. However, each finite supersoluble group is a Sylow tower group (see [7]).

The following result details many instances of friendly subgroups of finite groups.

Theorem 3.1 *Let K be a finite group and H a subgroup of K. Then H is a friendly subgroup of K if one or more of the following holds:*

1. *$H = K$;*

2. *K is cyclic;*

3. *H is a Hall subgroup of K (i.e.* $\gcd(|H|, |K{:}H|) = 1$*) and H is a Sylow tower group;*

4. *H is a nilpotent Hall subgroup of K (such as a Sylow subgroup), or more generally, H is a friendly subgroup of a nilpotent Hall subgroup of K;*

5. *K is soluble and H is a Hall subgroup of K, or more generally, K is soluble and H is a friendly subgroup of a Hall subgroup of K;*

6. *H is a simple normal subgroup of K and $|H|^2$ does not divide $|K|$.*

Proof.

1. Trivial.

2. If K is cyclic then H is the unique subgroup of K of order $|H|$.

3. Suppose H is a Sylow tower group of complexion $(p_1, \ldots, p_k)$, and that $J \leq K$ with $J \cong H$. Then J is a Sylow tower group of complexion $(p_1, \ldots, p_k)$, and by [7, Theorem A1], J is conjugate to H.

4. Suppose that H is a friendly subgroup of a nilpotent Hall subgroup L of K, and that $J \leq K$ with $J \cong H$. Then, by [13], J is conjugate to a subgroup J^* of L, and since H is a friendly subgroup of L, J^* (being isomorphic to H) is conjugate to H.

5. Suppose K is soluble, that H is a friendly subgroup of a Hall subgroup L of K, and that $J \leq K$ with $J \cong H$. Then, by [6], J is conjugate to a subgroup J^* of L, and since H is a friendly subgroup of L, J^* (being isomorphic to H) is conjugate to H.

6. Suppose H is a simple normal subgroup of K. Let $J \leq K$ with $H \neq J \cong H$. Since $H \cap J$ is a normal subgroup of J, we must have $H \cap J = 1$, and so HJ is a subgroup of K of order $|H|^2$. Thus, if $|H|^2$ does not divide $|K|$, then H is the only subgroup of its isomorphism class in K.

∎

It is worth noting that F. Gross, using the Classification of Finite Simple Groups, shows that every odd-order Hall subgroup of a finite group is a friendly subgroup of that group (see [4]), but I prefer not to use this sledgehammer to crack the odd nut.

4 Determining whether H is a Sylow tower group

We now present an algorithm which determines whether or not a given finite group H is a Sylow tower group. The algorithm given here is implemented in GAP and works well in practice for permutation groups and PC-groups (groups with polycyclic presentations), making use of the ability to compute a set of representatives of the Sylow subgroups of a given finite soluble group.

Suppose H is a finite group, and suppose $1 = T_0 < T_1 < ... < T_m \leq H$, such that, for $i = 1, \ldots, m$, T_i is a normal subgroup of H and T_i/T_{i-1} is isomorphic to a Sylow p_i-subgroup of H, for some prime p_i dividing $|H|$. Then $(T_0, \ldots, T_m)$ is called a *partial Sylow tower* for H, having *complexion* $(p_1, \ldots, p_m)$, and a partial Sylow tower for H is a *Sylow tower* for H if its last element is equal to H.

Lemma 4.1 *Suppose $(T_0, \ldots, T_m)$ is a partial Sylow tower, having complexion $(p_1, \ldots, p_m)$, for a finite group H, and let $i \in \{1, \ldots, m\}$. Then T_i is equal to the semidirect product $T_{i-1}{:}P$, for every Sylow p_i-subgroup P of H.*

Proof. Since T_i/T_{i-1} is isomorphic to a Sylow p_i-subgroup of H, the order of T_i is such that T_i contains a Sylow p_i-subgroup of H, and since T_i is normal in H, T_i contains every Sylow p_i-subgroup of H. In addition, since p_i does not divide $|T_{i-1}|$, we conclude that T_i is equal to the semidirect product $T_{i-1}{:}P$, for every Sylow p_i-subgroup P of H. ∎

The following theorem forms the theoretical basis of our iterative algorithm, which, given a finite group H, either ascends a Sylow tower for H, or when failing to do so, provides a proof that none such exists. The algorithm is detailed in Figure 1, using the algorithmic notation of [1] and the LaTeX package `clrscode`. It would be easy to modify this algorithm to compute a Sylow tower for H when H is a Sylow tower group.

Theorem 4.2 *Let H be a Sylow tower group, let $p_1, \ldots, p_k$ be the distinct primes dividing $|H|$, and let $\{P_1, \ldots, P_k\}$ be a set of representative Sylow subgroups of H, with P_i a Sylow p_i-subgroup.*

Suppose that $(T_0, \ldots, T_m)$ is a partial Sylow tower for H, with $m < k$. Then there exists an $i \in \{1, \ldots, k\}$ such that $T_m P_i$ is a normal subgroup of H, and properly contains T_m (in which case $(T_0, \ldots, T_m, T_m P_i)$ is a partial Sylow tower for H, extending $(T_0, \ldots, T_m)$).

Proof. Let $(U_0, \ldots, U_k)$ be a Sylow tower for H, having complexion $(q_1, \ldots, q_k)$ (thus $q_1, \ldots, q_k$ is some particular ordering of $p_1, \ldots, p_k$). Now let j be the least element of $\{1, \ldots, m+1\}$ such that q_j does not divide $|T_m|$ (such a j exists), and define i by $p_i = q_j$.

By the preceding lemma, U_{j-1} is generated by the union of the sets of all Sylow subgroups for the primes $q_1, \ldots, q_{j-1}$, and by the lemma and the definition of j, we see that U_{j-1} is a subgroup of T_m. By the definition of i, $U_j = U_{j-1} P_i$, and so $T_m P_i = T_m U_j$, the product of two normal subgroups of H. Thus $T_m P_i$ is normal in H, and since p_i does not divide $|T_m|$, we have that $T_m P_i / T_m$ is isomorphic to P_i. ∎

4.1 Some counts of Sylow tower groups

The algorithm given in Figure 1 has been implemented as a GAP function, and we use this to determine the number of Sylow tower groups amongst groups of small order, transitive permutation groups of low degree, and primitive permutation groups of modest degree. All timings are in CPU-seconds on a 2.8 GHz Pentium 4 PC running Linux.

Using the GAP library of small groups, we obtain a list of the 7012 groups (up to isomorphism) of order at most 255. We determine that 6836 of these are Sylow tower groups in about 16 seconds. This compares with 6998 of the groups being soluble, 6590 being supersoluble, and 3722 being nilpotent.

Using the GAP library of transitive groups, we obtain a list of the 4953 transitive groups (up to permutation isomorphism) of degrees 1 to 23. We determine that 2773 of these are Sylow tower groups in about 320 seconds. This compares with 4213 of the groups being soluble, 1820 being supersoluble, and 1488 being nilpotent.

Using the GAP library of primitive groups, we obtain a list of the 2270 primitive groups (up to permutation isomorphism) of degrees 1 to 256. We

Figure 1: Algorithm to determine whether H is a Sylow tower group

```
IsSylowTowerGroup(H)
  ▷ Given a finite group H, this function returns TRUE
  ▷ if H is a Sylow tower group, and returns FALSE if not.
  if H is not soluble
     then return FALSE;
  if H is nilpotent
     then return TRUE;
  P ← a list of representatives of the non-trivial Sylow subgroups of H;
  ▷ P contains exactly one Sylow p-subgroup for each prime p dividing |H|.
  T ← {1_H};
  while Length(P) > 1
        do ▷ T is the highest term of a partial Sylow tower for H,
           ▷ T ≠ H, and |T| is coprime to the order of each element of P.
           if there is no Q in P such that ⟨T, Q⟩ is normal in H
              then ▷ By Theorem 4.2, H cannot be a Sylow tower group.
                     return FALSE;
           Q ← the first Q in P such that ⟨T, Q⟩ is normal in H;
           T ← ⟨T, Q⟩;
           Remove Q from P;
  ▷ There is now just one element left in P, T is the last term
  ▷ of a partial Sylow tower for H, and this partial Sylow tower
  ▷ can be extended to a Sylow tower by adjoining H itself.
  return TRUE;
```

determine that 703 of these are Sylow tower groups in about 87 seconds. This compares with 940 of the groups being soluble, 439 being supersoluble, and 55 being nilpotent.

5 Acknowledgements

I thank Peter J. Cameron (Queen Mary, University of London) for coming up with the concept of a friendly subgroup, and I am grateful to Reinhard Laue (University of Bayreuth) for reading an early draft of this paper and making interesting and useful comments.

References

[1] T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, *Introduction to Algorithms* (Second edition), MIT Press, 2001.

[2] Z. Eslami, Classification of designs with nontrivial automorphism groups, *J. Combin. Designs* **14** (2006), 479–489.

[3] The GAP Group, GAP — Groups, Algorithms, and Programming, Version 4.4.9; Aachen, St Andrews, 2006, `http://www.gap-system.org/`

[4] F. Gross, Conjugacy of odd order Hall subgroups, *Bull. London Math. Soc.* **19** (1987), 311–319.

[5] E. Haberberger, A. Betten and R. Laue, Isomorphism classification of t-designs with group theoretical localisation techniques applied to some Steiner quadruple systems on 20 points, *Congr. Numerantium* **142** (2000), 75–96.

[6] P. Hall, A note on soluble groups, *J. London Math. Soc.* **3** (1928), 98–105.

[7] P. Hall, Theorems like Sylow's, *Proc. London Math. Soc. (3)* **6** (1956), 286–304.

[8] P. Kaski, Isomorph-free exhaustive generation of designs with prescribed groups of automorphisms, *SIAM J. Discrete Math.* **19** (2005), 664–690.

[9] R. Laue, Solving isomorphism problems for t-designs, *Designs 2002. Further computational and constructive design theory* (W.D. Wallis, ed.), Kluwer, Boston, 2003, 277–300.

[10] S. Linton, Finding the smallest image of a set, *ISSAC 2004* (J. Gutierrez, ed.), ACM Press, New York, 2004, 229–234.

[11] B.D. McKay, nauty, `http://cs.anu.edu.au/people/bdm/nauty/`

[12] L.H. Soicher, The DESIGN package for GAP, Version 1.3, 2006, `http://designtheory.org/software/gap_design/`

[13] H. Wielandt, Zum Satz von Sylow, *Math. Z.* **60** (1954), 407–408.