Finding 47:23 in the Baby Monster

WILSON, RA; BRAY, JN; PARKER, RA

Submitted exclusively to the London Mathematical Society
doi:10.1112/0000/000000

Finding 47 : 23 in the Baby Monster

John N. Bray, Richard A. Parker and Robert A. Wilson

Abstract

In this paper we describe methods for finding very small maximal subgroups of very large groups, with particular application to the subgroup 47 : 23 of the Baby Monster. This example is completely intractable by standard or naïve methods. The example of finding 31 : 15 inside the Thompson group Th is also discussed, as a test case.

1. *Introduction*

When computing in finite groups, it is often useful to know how to find generating sets for various subgroups, especially maximal ones, of a group G, in terms of the standard generators of G. Usually this means either as words or as straight-line-programs: here we use 'word' loosely to cover both concepts. The www-Atlas [**11**] now includes such words for maximal subgroups of many groups. Some of the small maximal subgroups, however, pose particular challenges.

One place where we thought at one time the www-Atlas would have a permanent gap was for a straight-line program to find generators for the maximal subgroup 47 : 23 of the Baby Monster $\mathbb{B}$, owing to pessimistic forecasts for how long such a search would take. However, in 2003 we managed to fill in this gap by devising a new and more subtle search strategy. In this paper we describe this method, in the hope that it may be of use in other similar searches.

There are various reasons why our main problem is particularly difficult, such as $\mathbb{B}$ being large (order approximately 4.15478×10^{33}) and that we are constrained to compute in a large representation (degree 4370 over $\mathbb{F}_2$). Moreover, the non-trivial elements of 47 : 23 have very small centralisers in $\mathbb{B}$ (order 46 or 47). And every non-trivial proper subgroup of 47 : 23 is cyclic of prime order, so we cannot even search in another subgroup of $\mathbb{B}$ to give us a useful contribution towards generating 47 : 23.

We first tested the new strategy by using it to obtain words for generators of 31 : 15 in terms of standard generators of the Thompson group Th. Since Th has a considerably smaller representation than $\mathbb{B}$ (degree 248 as opposed to 4370 over $\mathbb{F}_2$) this provided an ideal 'dry run' opportunity to try out various methods. It is worth pointing out that finding $31 : 15 < \mathrm{Th}$ is amenable to other techniques, and these include a brute force direct search, which we have also done. One of the main reasons for this is that 15, unlike 23, is not prime. The method Wilson [**10**] employs to find $31 : 15 < \mathrm{Th}$ relies on 15 not being prime.

For the purposes of this paper, we shall assume that we have a standard copy of $\mathbb{B}$, on its standard generators, given as a subgroup of $\mathrm{GL}_{4370}(2)$ (the one whose www-Atlas identifier is `BG1-f2r4370B0`). Likewise, we assume that we have a standard copy of Th as a subgroup of $\mathrm{GL}_{248}(2)$.

The calculations described in this paper can now be performed easily in Magma [**3**, **1**]. Originally, we used the C-Meataxe [**9**, **8**] for the computationally intensive parts of the Baby Monster calculations (mainly calculating the 'fingerprints' in Step 1).

2000 *Mathematics Subject Classification* 00000.

2. *A standard method*

Of the various methods the authors of the WWW-ATLAS [**11**] have commonly employed to find words for subgroups of a group $G = \langle a, b\rangle$, the least implausible method in the present instance (the case when $G = \mathbb{B}$ and the subgroup is $47{:}23$) is to search at random through subgroups generated by two elements of order 23.

Given fixed elements $x, y \in G$ (an arbitrary finite group), and elements x' and y' such that x and x' are G-conjugate and y and y' are G-conjugate, the probability that the pair (x', y') is G-conjugate to (x, y) is

$$\frac{|\mathrm{C}_G(x)|.|\mathrm{C}_G(y)|}{|G|.|\mathrm{C}_G(\langle x, y\rangle)|},$$

with $|\mathrm{C}_G(\langle x, y\rangle)| = 1$ for the cases that interest us here.

Suppose we choose to generate $47{:}23$ by two elements x and y of order 23 that are $\mathbb{B}$-conjugate. Since $\mathbb{B}$ has 242 conjugacy classes of $(23\mathrm{A}, 23\mathrm{A})$ pairs that generate $47{:}23$, the probability of success at each attempt is

$$\frac{46 \cdot 46 \cdot 242}{|\mathbb{B}|} = \frac{253}{2052757648827285667577856000000} \approx \frac{1}{8.114 \times 10^{27}}.$$

A test for success could be that $[[x, y], [[x, y], y]] = 1$ and $[x, y]$ has order 47. Trying to generate $47{:}23$ by two non-conjugate (in $\mathbb{B}$) elements of order 23, gives us the same probability of success as above, and the same test for success still works. We reiterate a point made by Linton [**7**]: the thing we must test quickly is *failure*.

For the $(23\mathrm{A}, 23\mathrm{A})$ case, we know that for $1 \leqslant i \leqslant 22$, the order of xy^i should be 23. So we would first test, using a carefully written 'Monte Carlo order oracle', that xy does not have order 23. If xy does (or might) have order 23, we then perform the same test for xy^2 and (if xy^2 has order 23) for xy^3. At some point, we decide we are not gaining much by performing more of these failure tests, and perform the success test instead. (For the $(23\mathrm{A}, 23\mathrm{B})$ case, the elements xy^i can have orders 23 or 47, and overall the test would take a bit more than twice as long on average.)

Suppose, optimistically, that we can check 10^{12} cases a year (which is more than 30000 per second). Then we would still expect the search to take more than 8×10^{15} years to complete, which compares unfavourably to the current age of the Universe, which is thought to be about 1.38×10^{10} years.

It is worth remarking that this method is essentially a black-box method, even though some of the tests are substantially speeded up by using fast Monte Carlo order oracles, or pseudo-order oracles, which depend on the representation for their implementation.

3. *A new method*

Our method is actually a method for finding the normaliser of a cyclic subgroup (or more generally, any small subgroup) inside a group H that lies in an ambient general linear group or symmetric group G. We suppose that H is generated by a relatively small conjugacy class X of elements of H. Let y be a generator for the cyclic group we wish to normalise.

Step 1. Locate pairs (x_1, y) and (x_2, y^k) that are H-conjugate, with $x_i \in X$ and k a suitable integer, and $\langle x_1, y\rangle = H$.

Step 2. Use standard computational methods to find all elements $g \in G$ that conjugate (x_1, y) to (x_2, y^k).

Step 3. For one such $g \in H$, write g as a word in the generators of H.

Step 1 is accomplished by a fingerprinting (or hashing) technique, described in more detail in Section 5.1, to detect coincidences in a large population. In this context, a fingerprint of a pair

(a, b) of group elements is a suitable collection of conjugacy-invariant data, for example a list of orders of certain elements, such as $ab, ab^2, [a, b], \ldots$. A similar fingerprint for representations of group algebras is described in [**8**]. The essential ingredient of our method is to use the 'birthday paradox' to find coincidences quickly. Suppose that $x_1, \ldots, x_m$ are chosen from a set of size d. Then the probability of at least one coincidence between an x_i and an x_j exceeds 50% if $m^2 > 2(\log_e 2)d$, and it exceeds 90% if $m^2 > 2(\log_e 10)d$.

Step 2 is achieved by a 'standard basis' algorithm, such as can be found in [**5**, Section 7.5.3], see also [**8**]. The 'standard basis' is not in general unique, and g is only determined up to multiplication by of the centraliser of the representation. However, in our examples, in fact the standard basis element is unique.

Step 3 is a 'constructive membership' problem, for which a number of methods, such as Ryba's algorithm [**6**], are available. However, Ryba's algorithm tends to produce quite long words, and we prefer a method which produces shorter words.

Our approach to finding a word for $g \in H$ (where H is a finite group) in terms of the generators of H is straightforward, even naïve: we use a process of successive approximation. So let

$$H = H_0 > H_1 > H_2 > \cdots > H_r = 1,$$

be a chain of subgroups of H, where the generators of H_i are given as words in the generators of H_{i-1}, and thus ultimately as words in the generators of H. The basic idea is to run through the right coset representatives h_0 of H_1 in H_0 until we find one for which $H_1 h_0 = H_1 g$. Thus $g h_0^{-1} \in H_1$ and we recurse the problem into H_1, next finding $h_1 \in H_1$ such that $H_2 g h_0^{-1} = H_2 h_1$, giving $g h_0^{-1} h_1^{-1} \in H_2$. Continuing in this manner, we eventually find elements $h_0 \in H_0$, $h_1 \in H_1, \ldots, h_{r-1} \in H_{r-1}$ such that $g = h_{r-1} \ldots h_1 h_0$. Of course, each h_i is written as a word in the generators of H_i.

Some of the indices we encounter below are somewhat too large to make such a naïve approach feasible, and again, the 'birthday paradox' comes to our rescue. Instead of searching through all the cosets $H_1 h_0$ until we find $H_1 g$, we make a collection of cosets $H_1 x_i$, and another collection $H_1 g y_j$, and look for coincidences between these two sets.

Suppose that $x_1, \ldots, x_m$ and $y_1, \ldots, y_n$ are chosen from a set of size d. Then the probability of at least one coincidence between an x_i and a y_j exceeds 50% if $mn > (\log_e 2)d$, and it exceeds 90% if $mn > (\log_e 10)d$. Therefore if H_1 has index d in H_0 and we make equal numbers of "forward" and "back" cosets (that is, cosets $H_1 x_i$ and $H_1 g y_j$), we need to make about $2\sqrt{(\log_e 2)d}$ cosets for the probability of coincidence to be at least 50%, and probably somewhat more than that to account for insufficient randomness in our production of cosets.

4. *Application to the Thompson group*

We first describe how one can use this new method to find a copy of $31{:}15$, the normaliser of a subgroup of order 31, in Th. We use the embedding $\mathrm{Th} \leqslant \mathrm{GL}_{248}(2)$. For this experimental case, we did try out Ryba's algorithm, regarding the involution centraliser $2_+^{1+8} \cdot \mathrm{A}_9$ as the base case. The result of this is given as the program `ThG1-max15W1` in the WWW-ATLAS. Thereafter, we quickly produced another version (`ThG1-max15W2`), with shorter words, using other methods for constructive membership testing, as described below.

We pick an element y of order 31. The quadratic residules modulo 31 are the powers of 7, that is:

$$1, 7, 18, 2, 14, 5, 4, 28, 10, 8, 25, 20, 16, 19, 9.$$

The conjugacy class X is taken to be Class 2A, which is the class of 976841775 involutions.

Step 1. For the first step, we must fingerprint pairs (x, y) with $x \in X$. It is not quite true that all such pairs must generate Th, for among the 976841775 involutions, one calculates that 3×7740 of them generate with y a copy of $2^5.\mathrm{L}_5(2)$ (for a probability of 16/700245) and that 3×31 of them generate $2^5{:}31$ (for a probability of 1/10503675). So the probability of a proper subgroup is 241/10503675, which is approximately 1 in 43584. This is small enough that it does not materially affect the calculations.

A convenient fingerprint is the list of orders of xy^{7^k}, for $0 \leqslant k \leqslant 14$. This was chosen because 7 is an element of multiplicative order 15 modulo 31. Indeed, the 8 elements of order 15 modulo 31 are just 7^k for k coprime to 15. Replacing y by y^7 then just has the effect of cycling the 15 orders in this fingerprint. Thus a relevant coincidence is detected by finding two fingerprints which differ only by one of the 8 rotations of order 15. We expect around two million different cyclically reduced fingerprints, and therefore expect to find a coincidence after inspecting a few thousand of them.

Step 2. Once we have found a candidate coincidence in this way, Step 2 allows us both to prove that it is a genuine coincidence, and to find an element g of $\mathrm{GL}_{248}(2)$ which conjugates the relevant pairs: (x_1, y) to (x_2, y^{7^k}). Now the fact that this representation of Th is absolutely irreducible, over the field of two elements, implies that there is a unique such element g, and in particular, $g \in \mathrm{Th}$. Of course, we want g to have order 15, which means that the coincidence we use must have k prime to 15. Otherwise, g has smaller order. (An alternative is to find one element of order 3 and one of order 5, rather than an element of order 15.)

Step 3. The final step is now to find a word for g in terms of the standard generators for Th. One method for doing this is Ryba's algorithm, which is effective in this case, but gives words which are perhaps longer than we might hope for. Since this is a one-off calculation, it is worth adopting a rather more labour-intensive method in order to obtain shorter words, as described in Section 3.

We pick a subgroup chain (of length 2):

$$\mathrm{Th} > 2_+^{1+8}{\cdot}\mathrm{A}_9 > 1,$$

with indices 976841775 and 92897280. Our strategy is, essentially, first to identify which coset of the subgroup $2_+^{1+8}{\cdot}\mathrm{A}_9$ the element g lies in, and then to identify which element of that coset it is. In both steps, we use the 'birthday problem' to square root the size of the search.

Now in the given representation of Th, the subgroup $2_+^{1+8}{\cdot}\mathrm{A}_9$ fixes a unique nonzero vector v, and we may use the orbit of v under Th to represent the cosets of $2_+^{1+8}{\cdot}\mathrm{A}_9$ therein. The method is to make around 100000 images of v and v^g under known elements of Th, and search for coincidences. Any such coincidence gives rise to an element w of Th such that gw fixes v, and therefore gw lies in the given subgroup which is the stabiliser of v.

Within the subgroup, we first chopped the representation, and chose a suitable subquotient representing the group $2^8{\cdot}\mathrm{A}_9$. In fact, we were able to choose a subspace of dimension 29, namely the third socle. The dimension was small enough that we could compute enough group elements to locate a coincidence directly. Hence we obtained a word for gw, which now only had to be possibly adjusted by the central involution (of $2_+^{1+8}{\cdot}\mathrm{A}_9$) in order to complete the task.

5. *Application to the Baby Monster*

We apply the same basic method outlined in Section 3 to the problem of finding $47{:}23$ in $\mathbb{B}$ as follows. We assume the 47-subgroup we wish to normalise is generated by y, and that $\mathbb{B}$ sits in some 'universal' group G, which in our case is $G = \mathrm{GL}_{4370}(2)$. Specifically, we let

$\mathbb{B} = \langle a, b \rangle \leqslant \mathrm{GL}_{4370}(2)$, where a and b are standard generators of $\mathbb{B}$, that is a is in Class 2C, b is in Class 3A, ab has order 55 and $abababab^2abab^2ab^2$ has order 23. Our standard 47-element is then taken to be $y = abab^2abababab^2abab^2$, which we shall deem to lie in Class 47A. The normalising element g that we are looking for conjugates y to y^k for some $k \neq 1$ which is a quadratic residue modulo 47. The quadratic residues modulo 47 are the powers of 2. We let x run through the smallest non-trivial conjugacy class in $\mathbb{B}$, so that the number of possibilities for (x, y) is as small as possible. This is the class of *transpositions* (Class 2A in the Atlas [**4**, page 208]).

5.1. *Step 1*

Any $(2\mathrm{A}, 47\mathrm{A})$ pair of elements must generate the whole of $\mathbb{B}$, since the only group H such that $\mathbb{B} > H > \langle y \rangle \cong 47$ is $H = \mathrm{N}_{\mathbb{B}}(\langle y \rangle) \cong 47{:}23$. Now $\mathbb{B}$ has 13571955000 transpositions, and all subgroups of order 47 are self-centralising, so the probability that any $(2\mathrm{A}, 47\mathrm{A})$-pair is conjugate in $\mathbb{B}$ to a given one is:

$$\frac{47}{13571955000} = \frac{1}{288765000}.$$

Naturally, we do not wish to collect fingerprints of the 288765000 distinct $(2\mathrm{A}, 47\mathrm{A})$-pairs (and the same number for the $(2\mathrm{A}, 47\mathrm{B})$-pairs) in order to define a pair of $(2\mathrm{A}, 47\mathrm{A})$ standard generators of $\mathbb{B}$. Nor would we wish to trawl through an expected 288765000 transpositions x_2 until we found one such that (x_2, y^{2^k}) is conjugate to our favourite $(2\mathrm{A}, 47\mathrm{A})$-pair (x_1, y).

We have already observed that we wish to conjugate y to some non-trivial power y^m, where necessarily m is a square modulo 47, and without loss of generality $1 < m < 47$. It is convenient for later computations to arrange the possible values of m as successive powers of 2, and so the set S of allowed values of m modulo 47 is:

$$S := \{2, 4, 8, 16, 32, 17, 34, 21, 42, 37, 27, 7, 14, 28, 9, 18, 36, 25, 3, 6, 12, 24\}.$$

The fingerprint of a pair (x, y) is the list of the orders of the 23 elements xy^{2^k}, as k runs from 0 to 22. We used a Monte Carlo order oracle, which has a very small chance of giving a proper divisor of the actual element order. In principle, this could cause us to miss a genuine coincidence, or to flag up a false coincidence. The former is not material, while the latter is detected at the next stage, so can be thrown away and a new coincidence tested.

The probability that (x_1, y) is conjugate to (x_2, y^m) for one of 22 values of $m \in S$ is $p = 11/144382500$, which is still uncomfortably low. But we are not looking for a particular fingerprint, we are just looking for coincidences between fingerprints. That is, we wish to make sufficient transpositions $x_1, x_2, \ldots, x_r$ such that (x_i, y) and (x_j, y^m) are conjugate for some $m \in S$. As described in Section 3, this is a form of the 'birthday problem', and somewhere around $r = 10000$ gives us a reasonably good chance of finding a coincidence. Note, however, that we are not interested in coincidences with $m = 1$. In the end, we considered slightly fewer than 10000 fingerprints, and found exactly one coincidence of the required type. It transpires that the coincidence we found was between pairs of the form (x_1, y) and (x_2, y^{16}).

5.2. *Step 2*

Since (x_1, y) and (x_2, y^{16}) are conjugate pairs of generators of $\mathbb{B}$, they are also conjugate in $\mathrm{GL}_{4370}(2)$. Hence a 'standard basis' algorithm as described in [**5**, Section 7.5.3] will produce an element g of $\mathrm{GL}_{4370}(2)$ which conjugates one pair to the other. Since the representation of $\mathbb{B}$ is irreducible, g is unique up to scalar multiplication, and since the field has order 2, the element g is actually unique. It follows that g lies in $\mathbb{B}$.

5.3. *Step 3*

For Step 3, it was originally expected that we would use Ryba's algorithm [**6**] to achieve this. In fact, in the Baby Monster case we never tried to used Ryba's algorithm, and used the method described in Section 3, which was designed to give shorter words than Ryba's algorithm was likely to produce.

The best choice for H_1 in the Baby Monster is the largest subgroup, $2^{\cdot 2}\mathrm{E}_6(2):2$, of index 13571955000, and which can easily be found by the standard method of Bray [**2**]. Thus to perform the first step in our Baby Monster calculation, we expect (in the English sense) to create about 200000 cosets before finding the desired coincidence. This is an improvement on the naïve method by a factor of nearly 10^5. Moreover, this subgroup fixes a unique non-zero vector in the given representation, so we can use vectors as labels for the cosets. (Even so, the storage requirement for 200000 vectors is at least 100MB, even when stored in binary format.)

For the remaining steps of the calculation, both the groups and the representations are much smaller, and it is not necessary to take so much care over all the details. Our choice for the group H_2 was a group of shape $2.(2_+^{1+20}:\mathrm{U}_6(2):2)$, with index 3968055 in H_1. For this calculation, we chopped the representation (of H_1) to one of dimension 78, such that there is an involution in the kernel of the (78-dimensional) representation. We adjust for this involution, if necessary, at the end of the whole calculation.

For H_3 we chose $2^2.\mathrm{U}_6(2).2$, with index 1048576 in H_2, and order 73574645760, so it is possible to choose H_4 to be the trivial group. Other chains of subgroups of H_3 are possible and may be easier to use. (Once we had found H_2 we worked in an even smaller representation that faithfully represented $2^{20}:\mathrm{U}_6(2):2$, and so the effective index of H_4 in H_3 was $|\mathrm{U}_6(2):2| = 18393661440$.)

References

1. W. Bosma, J. Cannon and C. Playoust. The Magma algebra system. I. The user language. Computational algebra and number theory (London, 1993). *J. Symbolic Comput.* **24** (1997), no. 3–4, 235–265.

2. J. N. Bray. An improved method for generating the centralizer of an involution. *Arch. Math. (Basel)* **74** (2000), 241–245.

3. J. J. Cannon *et al.* The Magma programming language. Various versions, 1994–present.

4. J. H. Conway, R. T. Curtis, S. P. Norton, R. A. Parker and R. A. Wilson. "An ATLAS of finite groups", Clarendon Press 1985.

5. D. F. Holt, B. Eick and E. A. O'Brien. Handbook of Computational Group Theory. Discrete Mathematics and its Applications (Boca Raton). Chapman & Hall/CRC, Boca Raton, FL, 2005.

6. P. E. Holmes, S. A. Linton, E. A. O'Brien, A. J. E. Ryba and R. A. Wilson. Constructive membership in black-box groups. *J. Group Theory* **11** (2008), no. 6, 747–763.

7. S. A. Linton. The art and science of computing in large groups. *Computational algebra and number theory (Sydney, 1992)*, 91–109, Math. Appl., 325, *Kluwer Acad. Publ., Dordrecht*, 1995.

8. R. A. Parker. The computer calculation of modular characters (the MeatAxe). *Computational group theory (Durham, 1982)*, 267–274, *Academic Press, London*, 1984.

9. M. Ringe. The C-MeatAxe programming language.

10. R. A. Wilson. Some new subgroups of the Baby Monster. *Bull. London Math. Soc.* **25** (1993), no. 1, 23–28.

11. R. A. Wilson, R. A. Parker, S. J. Nickerson, J. N. Bray *et al.* "Atlas of Finite Group Representations, Version 3" available at `http://brauer.maths.qmul.ac.uk/Atlas/v3/`.

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

J.N.Bray@qmul.ac.uk

70 York Street,
Cambridge,
CB1 2PY

richpark54@hotmail.co.uk

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

R.A.Wilson@qmul.ac.uk