

# CONJUGACY CLASS REPRESENTATIVES IN THE MONSTER GROUP

R. W. BARRACLOUGH AND R. A. WILSON

## *Abstract*

The paper describes a procedure for determining (up to algebraic conjugacy) the conjugacy class in which any element of the Monster lies, using computer constructions of representations of the Monster in characteristics 2 and 7. This procedure has been used to calculate explicit representatives for each conjugacy class.

## 1. Introduction

The Monster group,  $\mathbb{M}$ , is the largest of the 26 sporadic simple groups, having order

$$2^{46} \cdot 3^{20} \cdot 5^9 \cdot 7^6 \cdot 11^2 \cdot 13^3 \cdot 17 \cdot 19 \cdot 23 \cdot 29 \cdot 31 \cdot 41 \cdot 47 \cdot 59 \cdot 71$$

$$= 808,017,424,794,512,875,886,459,904,961,710,757,005,754,368,000,000,000.$$

It is useful to find conjugacy class representatives in groups, and this has been done explicitly for the other twenty-five sporadic simple groups (see [5, 8]), but the Monster is more difficult to deal with. The main obstacle to effective computation in the Monster is the large degree of the smallest faithful representation: 196883 over  $\mathbb{C}$ , or a mere 196882 over  $\mathbb{F}_2$  or  $\mathbb{F}_3$ . We make use of the original computer construction of the Monster [4] and of an analogous construction over  $\mathbb{F}_7$  due to the second author [7] to find words in the generators for an element in each conjugacy class of  $\mathbb{M}$ , up to algebraic conjugacy. (Two classes  $X$  and  $Y$  are algebraically conjugate if  $\langle x \rangle$  and  $\langle y \rangle$  are conjugate whenever  $x \in X$  and  $y \in Y$ .) In the Monster the only non-trivial algebraic conjugacies are between classes  $X$  and  $Y = X^{**}$ , defined by  $x \in X \iff x^{-1} \in X^{**}$ . In practice, this means that we can distinguish such classes only by using a character that takes non-real values on  $X$  and  $X^{**}$ , and for the Monster such characters have enormous degree. We are also unable to distinguish the classes of elements of order 27, as they have the same character values and power maps. Incidentally, these are precisely the same sets of classes whose McKay–Thompson series are identical [2]. However, we are able to produce an element in class 27A and another element in class 27B, and the corresponding words, by a different method.

## 2. Computer constructions of the Monster

We recapitulate some useful information from [4]. The Monster is constructed from subgroups of shape  $N(3B) = 3^{1+12} \cdot 2 \cdot \text{Suz}:2$  and  $3^{2+5+10} \cdot (\text{M}_{11} \times 2 \cdot \text{S}_4)$  intersecting in a group of shape  $3^{2+5+10} \cdot (\text{M}_{11} \times \text{S}_3)$ . Importantly for us, the 196882-dimensional module over  $\mathbb{F}_2$  decomposes under the action of  $N(3B)$  into submodules either of small dimension or with useful structure. This means that we can compute with relative ease in this group.

---

Calculations performed on equipment provided by EPSRC grant GR/R95265.  
 Received 25 January 2005, revised 14 June 2005; published 6 September 2005.  
 2000 Mathematics Subject Classification 20E45, 20D08  
 © 2005, R. W. Barraclough and R. A. Wilson

Our generators for  $N(3B)$  are  $B, C$  and  $E$ , where  $B$  and  $C$  are standard generators for  $6\text{-Suz}:2$  in the sense of [9, 6], and  $E$  is a specified element of  $3^{1+12}$ . A fourth element  $T$ , of order 2, extends a suitable subgroup  $3^{2+5+10} \cdot (\mathbb{M}_{11} \times S_3)$  to  $3^{2+5+10} \cdot (\mathbb{M}_{11} \times 2 \cdot S_4)$ . We have  $\langle B, C, T \rangle = \langle B, C, E, T \rangle = \mathbb{M}$ , but it is not possible to multiply  $T$  with any of the other three generators. To work in the Monster, we are reduced to acting on vectors by words of the form  $W_1 T W_2 T \dots$  for  $W_i \in N(3B)$ . Acting on a random vector allows element orders to be calculated with high probability, and using two vectors whose stabilisers intersect trivially increases this probability to 1. The trace of an element can be found by acting on each basis vector in turn and adding the appropriate entry of the result to a running total. The number of occurrences of  $T$  in a word is the main factor determining the speed of the calculation. On a Pentium III 700MHz, computing the trace for a word  $W_1 T$  takes about 30 minutes.

The construction given in [7] uses the same generators  $B, C, E$ , and  $T$  to make the 196883-dimensional representation over  $\mathbb{F}_7$ . In this case, calculating the trace for a word  $W_1 T$  takes about seven hours.

### 3. Implementation

One reason why the modulo 2 construction is considerably faster is the efficiency of binary arithmetic on a binary computer. The programs that implement the  $\mathbb{F}_2$  construction of [4] are based on R. Parker’s MeatAxe, which stores an element of  $\mathbb{F}_2$  as a single bit, so that addition and multiplication can be done for 32 elements at once (on a 32-bit computer) using the operations `xor` and `and`, respectively. With this in mind, we modified the modulo 7 programs as follows. Originally, each byte stored two field elements (say  $(a, b)$ ) as  $7a + b$ , and arithmetic was done with a  $256 \times 256$  lookup table, two field elements at a time. In our modified packing, we store  $a$  in the first nibble of the byte and  $b$  in the second, that is,  $32a + b$ . Since  $7 + 7 < 16$ , we can safely add such bytes together, as one nibble can never overflow into the next; see the following examples.

$$\begin{array}{ll}
 (2, 6) + (3, 1) \rightarrow 00100110 + 00110001 & (2, 6) + (6, 6) \rightarrow 00100110 + 01100110 \\
 = 01010111 & = 10001100 \\
 \rightarrow (5, 7) & \rightarrow (8, 12)
 \end{array}$$

To perform the modular reduction on a byte, we first add the byte 00010001 to make 7 overflow to 8 in each nibble. Now take the bitwise `and` of this byte and 10001000, subtract it, shift it four places to the right, and then add it on (this performs reduction modulo 8). Finally subtract 00010001; see the following examples.

$$\begin{array}{ll}
 01010111 + 00010001 = 01101000 & 10001100 + 00010001 = 10011101 \\
 01101000 \text{ and } 10001000 = 00001000 & 10011101 \text{ and } 10001000 = 10001000 \\
 01101000 - 00001000 = 01100000 & 10011101 - 10001000 = 00010101 \\
 00001000 \xrightarrow{4} = 00000001 & 10001000 \xrightarrow{4} = 00010001 \\
 01100000 + 00000001 = 01100001 & 00010101 + 00010001 = 00100110 \\
 01100001 - 00010001 = 01010000 & 00100110 - 00010001 = 00010101 \\
 \rightarrow (5, 0) & \rightarrow (1, 5)
 \end{array}$$

This procedure is considerably slower than the old lookup table. However, we can apply it to 32 bits at once instead of just bytes, thus processing 4 times as many field elements at once and realising approximately a two-fold speed increase.

#### 4. Conjugacy class representatives

##### 4.1. Distinguishing conjugacy classes

The ordinary character of degree 196883 is rational, and therefore cannot be used to distinguish algebraically conjugate classes. Moreover, the classes  $27A$  and  $27B$  have the same character value and power maps, so these classes also cannot be distinguished in this way. It is straightforward to check that, up to these ambiguities, the conjugacy class of an element is determined by its order and the character values of it and its powers. Indeed, it is sufficient to use the reduction of the character value modulo 14; that is, mod 2 and mod 7.

We obtain our character values from trace calculations performed on explicit elements of  $\mathbb{M}$ . Note that modulo 2 the 196883-dimensional module splits as  $1 + 196882$ , and so the traces obtained from calculations have the other parity to the character values tabulated in the ATLAS [1] for  $\chi_2$ . Modulo 7 there is no such splitting, and the traces are the 7-modular reductions of the values of  $\chi_2$ . Wherever possible, the modulo 2 values are used, since the calculations are much quicker.

For each order of elements in  $\mathbb{M}$ , we tabulate the traces that we must find in order to identify each class of elements of that order. These data allow us to answer a second question: ‘Given an element of order  $n$  what information must we calculate in order to determine its conjugacy class?’

These ‘conjugacy class identification data’ are displayed in Table 1. The format is easily read by machine (and thus is a little cryptic). For each order  $n$  of elements in  $\mathbb{M}$ , there is a header row followed by one line for each class of elements of order  $n$ . The header row shows the element order, followed by what needs to be calculated: ‘2’ for a trace modulo 2, ‘7’ for a trace modulo 7, and ‘a**P**’ for the trace modulo  $a$  of the  $b$ th power. The subsequent rows give the traces for the class indicated in the first column. A header row containing an asterisk indicates that classes of order  $n$  cannot be distinguished. A header row containing the number 1 indicates that there is only one conjugacy class of elements of order  $n$ , so no traces need to be calculated.

##### 4.2. Searching for conjugacy class representatives

In exact analogy with finding conjugacy class representatives in other sporadic simple groups [5], a random search is conducted. The strategy is to generate a long list of words, calculate the orders of the corresponding elements of  $\mathbb{M}$ , and then use the conjugacy class-identification information to work out in which classes these elements lie.

It is sufficient to find a generator for each conjugacy class of maximal cyclic subgroup, that is, a representative for every class that cannot be obtained by powering up an element of greater order. Excluding the classes of algebraically conjugate elements, these are:  $18E$ ,  $24A$ ,  $24C$ ,  $24D$ ,  $24E$ ,  $24F$ ,  $24G$ ,  $24H$ ,  $24I$ ,  $24J$ ,  $30A$ ,  $30F$ ,  $32A$ ,  $32B$ ,  $36A$ ,  $36B$ ,  $36C$ ,  $36D$ ,  $38A$ ,  $39B$ ,  $40A$ ,  $40B$ ,  $41A$ ,  $42D$ ,  $45A$ ,  $48A$ ,  $50A$ ,  $51A$ ,  $52B$ ,  $54A$ ,  $56A$ ,  $57A$ ,  $60A$ ,  $60B$ ,  $60C$ ,  $60D$ ,  $60E$ ,  $60F$ ,  $66A$ ,  $66B$ ,  $68A$ ,  $70A$ ,  $70B$ ,  $78A$ ,  $78C$ ,  $84A$ ,  $84B$ ,  $84C$ ,  $105A$ , and  $110A$ . So we must definitely find representatives for these classes. However, powering up produces words with many instances of  $T$  that are difficult and slow to work with. It is therefore preferable to find representatives for as many classes as possible, not just those whose elements generate maximal cyclic subgroups. Of course, whenever possible it is preferable to find words in  $\langle B, C, E \rangle$ , since they are much faster to calculate with.



4.3. *Word-generation strategies*

Product replacement [3] has been shown to produce genuinely random words in exactly our situation, but the word length soon becomes very large, which would make our calculations too slow. Instead, we conducted a search through words of the form  $W_1T$  and  $W_2$ , where  $W_1$  is a word in  $B$  and  $C$ , and  $W_2$  is a word in  $B$ ,  $C$ , and  $E$ . Eliminating words containing  $B^3$ ,  $C^4$ , or  $E^3$ , we obtained 1.4 million words of shape  $W_1T$ , and we calculated element orders for 160,000 of these. For each element order, we produced trace-calculation jobs according to the data in Table 1. We stored the results of all these calculations in a database that was used to produce further jobs until we had witnessed all the classes of elements of that order. For the elements of order 12, for example, we ran the following jobs.

1. Calculate the trace modulo 7.
2. If the trace modulo 7 is 5 or 6, then calculate the trace modulo 2.
3. If the trace modulo 7 is 5 and the trace modulo 2 is 0, then calculate the trace modulo 7 of the square.
4. If the trace modulo 7 is 6 and the trace modulo 2 is 0, then calculate the trace modulo 7 of the cube.

Representatives for most of the conjugacy classes were found in this way, using words of the form  $W_1T$ . A few classes of elements of orders 24 and 36 did not turn up, and required more sophisticated methods. These classes were  $24C$ ,  $24G$ ,  $24F$ ,  $24J$ , and  $36B$ . In fact, only 43 elements of the form  $W_1T$  have been found in the classes  $24J$ ,  $24G$ ,  $24F$  and  $24E$  (most of them in  $24E$ ), but finding these took over 60 days of computer time.

4.4. *Example: finding an element in class 24C*

If  $x \in 24C$ , then  $x^8 \in 3B$ , so  $C_{\mathbb{M}}(x)$  is the centraliser in  $3^{1+12} \cdot 2 \cdot \text{Suz}$  of  $x^3$ . Since  $x^3$  has order 8, we may think of it as being an element of  $2 \cdot \text{Suz}$ , which has the following conjugacy classes of elements of order 8:

- class  $8A$  with centraliser in  $2 \cdot \text{Suz}$  of order  $192 = 2^6 \cdot 3$ ;
- classes  $+8B$  and  $-8B$ , both with centraliser in  $2 \cdot \text{Suz}$  of order  $2 \cdot 64 = 2^7$ ;
- class  $8C$  with centraliser in  $2 \cdot \text{Suz}$  of order  $32 = 2^5$ .

Since  $|C_{\mathbb{M}}(x)| = 3456 = 2^7 \cdot 3$  and  $3^{1+12}$  is a 3-group, the 2-powers in these centraliser orders show that  $x^3$  must lie above class  $8B$  of  $\text{Suz}$ .

Now,  $C$  and  $B = D$  are standard generators for  $6 \cdot \text{Suz}:2$  and standard generators for  $6 \cdot \text{Suz}$  are  $A$  and  $B$ , where

$$A = (CDD)^{-2}(CD)^{14}(CDD)^2$$

as in [4]. Using  $A$  and  $B$  in the recipe in [9] for the conjugacy classes of  $\text{Suz}$  yields elements of  $6 \cdot \text{Suz}$  lying above the corresponding classes of  $\text{Suz}$ . From this, we obtain elements  $g_8$  lying above  $8B$  of order 8, and  $g_{10}$  lying above  $10B$  of order 60. Therefore  $g_6 = g_{10}^{10}$  is a central element of order 6 in  $6 \cdot \text{Suz}$ . To find all classes lying above  $8B$ , we calculate the order and traces for  $g_6^i g_8$  for  $0 \leq i \leq 5$ , and thus find elements of  $\mathbb{M}$  in classes  $8A$ ,  $8C$ ,  $24C$ , and  $24I$ .

As a word in  $B$  and  $C$ , the length of the element in  $24C$  is 161455, which is too long for our results table and takes about 30 minutes to make on machine (using a naïve algorithm). As a word in  $A$  and  $B$ , the length is reduced to 205. Looking at elements lying above class  $8C$  of  $\text{Suz}$  yields words in  $24G$  and  $8C$ , but we already have good words for these classes.

4.5. Elements of order 27

Class 27A has the same character values and power maps as does class 27B, so they cannot be distinguished mechanically, using our method. However,  $|C_{\mathbb{M}}(27A)|$  is divisible by 2, whereas  $|C_{\mathbb{M}}(27B)|$  is not, and we can exploit this to find words in  $\langle B, C, E \rangle$  for elements in each of these classes.

To find such elements, we work in the three-fold cover  $G$  of  $N(3B)$ . This is a group of shape  $G \cong 3^{1+12} \cdot 6 \cdot \text{Suz}:2$ , and it has a representation in 38 dimensions over  $\mathbb{F}_3$  that was constructed in [4]. Using this representation, we find words in  $\langle B, C \rangle$  for an element  $g$  of order 9 in  $6 \cdot \text{Suz}:2$ . There are 3 such conjugacy classes in  $6 \cdot \text{Suz}:2$ , and they all project onto the same conjugacy class in  $2 \cdot \text{Suz}:2$ , so we may choose whichever we like.

Now, extending downwards to  $G$ , we find two conjugacy classes of elements of order 27, and four classes of elements of order 9. One of these classes of elements of order 27 must become 27A in the Monster, and the other must become 27B. This is because these two classes are the only two classes of elements of order 27 in the quotient group  $N(3B)$ . We compute  $C_G(g)$  to find which is centralised by an involution, and thus distinguish the classes.

Straight-line programs for these elements are included in the archive.

Note that an element in class 27A can also be obtained by squaring an element in class 54A.

5. Results

The results consist of a list of conjugacy class names, with corresponding words. Table 2 gives, for each conjugacy class of  $\mathbb{M}$ , the shortest word that we have found for an element in that class.

The class names are in the second column, and in the third column is the number of words found in the class (if there is a sum, then the summands give the number of words of the form  $W_1T$  and  $W_2$  respectively). The fourth column gives either a word or, where no element has been found, the name of a class that powers up to the particular class, followed in brackets by the number of  $T$ 's that the power has. Any element that powers up to class 3B is conjugate to an element in  $\langle B, C, E \rangle$ , as are all other powers. Such conjugacy classes are indicated by a bullet in the first column. Of course this test does not identify all of the conjugacy classes of  $\mathbb{M}$  that are represented in  $\langle B, C, E \rangle$ .

Table 2: Words for conjugacy class representatives

$\langle B, C, E \rangle$	Class	Number	Word
•	2B	8	6C(0)
•	2A	0	4B(0)
•	3B	0	6D(0)
	3A	0	B
	3C	0	BCCCBBBCBCBCBCBCBCCBBBC– –BBCCCBBBCBCBCBCBCBCCBBBC– –BCBBCCBBCCBBCC
•	4D	6982	BCCBB
•	4A	8	BCCBBT or 24A(0)

Continued on the next page



Table 2: Words for conjugacy class representatives, *continued*

$\langle B, C, E \rangle$	Class	Number	Word
	14B	0	28B(2)
	14A	0	28A, 42A(3)
	15A	0	30B(2)
	15D	0	30E(2)
•	15B	0	30D(2)
•	15C	3	<i>BCBCBCBCBBCBBCBCT</i>
	16A	48 + 1	<i>BCCCBBCBCBCBCBCBBCBBC–</i> <i>–BBCCCBBCBCBCBCBBCBBC–</i> <i>–BBCCCBCBCBBCBCBCBCBBCB</i>
	16B	0	32A(2)
	16C	0	32B(2)
	17A	55	<i>BCBCBCBCBCBCBCT</i>
•	18A	0	54A(3)
•	18E	26	<i>BBCBCBCBBCBCCT</i>
•	18D	48	<i>BCBCBCBCCBBT</i>
•	18B	57	<i>CBBCBBCBCBBT</i>
•	18C	0	36D(2)
	19A	21	<i>BBCBBCBBCBCCCBBCBCBCBBT</i>
	20A	0	40B(2)
	20B	0	40A(2)
•	20C	0	60C(3)
•	20D	12	<i>BCBCBBCBCBBCBCCT</i>
	20E	4	<i>BBCBBCBCBCBBCBCBBT</i>
•	20F	4	<i>BBCBBCBCBBCBBT</i>
	21C	0	42C(2)
	21A	3	<i>BBCBBCBCBCBBCBCBBCBCT</i>
•	21D	1	<i>BBCBBCBBCBCBBCBCBCT</i>
	21B	1	<i>BBCBBCBBCBBCBBCBCBBT</i>
•	22B	9	<i>BCCBCBCBCBCBCBCBCBCB</i>
	22A	21	<i>BBCBBCBBCBCT</i>
	23AB	2361	<i>BCCBCCT</i>
•	24H	623	<i>BCCBCBCBCBBCBCBCB</i>
	24A	1	<i>BCBCBBCBCBBCBCT</i> and see page 209 of the text
	24B	1	<i>BCBCBBCBCBBCBCBT</i>
•	24I	31	<i>BBCBBCBCBCBCBBT</i>
•	24D	222 + 23	<i>BCCBCBCBBCBCBCBBCB</i>
•	24C	0	see page 209 of the text
	24J	1	<i>BCBCBCBBCBCCCT</i>
•	24G	9923	<i>EBBCBCBCBC</i>

*Continued on the next page*

Table 2: Words for conjugacy class representatives, *continued*

$\langle B, C, E \rangle$	Class	Number	Word
•	24F	9	<i>BBCBBCBCBCBBCBBCBCBT</i>
	24E	33	<i>BBCBCBCBCCCBT</i>
	25A	949	<i>BBCBBCBCBBCBBT</i>
	26A	8	<i>BBCBBCBCBCBCCCBCBBCBCCT</i>
•	26B	78	<i>BBCBCCCBCBCBBT</i>
•	27A	0	54A(2)
•	27AB	454	<i>BBCCBCBCBCBBCT</i>
•	28D	47	<i>BBCCBCBCBCCBCBT</i>
	28B	27	<i>BBCCCBT</i>
	28A	0	84A(3)
	28C	31	<i>BCBBT</i>
	29A	1553	<i>BCCBCBCBBCBT</i>
	30B	1	<i>BBCBBCCBCBCBBCBCBCBBCBCCCT</i>
	30C	5	<i>BBCBBCBCBCBCBCBCBBCBCBBT</i>
	30E	107	<i>BCBCBBCBCBBCBBCT</i>
•	30D	2	<i>BBCBBCBCBCBCBT</i>
•	30G	1	<i>BBCBCBCBBCBBCT</i>
•	30F	13	<i>BBCCBCBCT</i>
•	30A	1	<i>BCBBCBCBCBBCCT</i>
	31AB	3034	<i>BBCT</i>
	32B	3	<i>BCBBCCCT</i>
	32A	1	<i>BCBCBCBCCT</i>
	33B	5	<i>BBCBBCCBBCBBCBCBCBBCBCT</i>
•	33A	35	<i>BBCBCBBCBBCBCBT</i>
	34A	3221	<i>BCCBCBCBT</i>
	35B	1316	<i>BCBCBCCBCBT</i>
	35A	0	70A(2)
•	36C	3	<i>BCBCBBCBCBBCBBT</i>
•	36D	590	<i>BCBCBBT</i>
•	36A	1	<i>BBCCBCBCBCBCCCT</i>
•	36B	1	<i>BBCBBCBBCBBCBBCBCBCBBT</i>
	38A	1628	<i>BBCBCBCBCBT</i>
	39A	102	<i>BCBBCBCBCCCT</i>
	39B	3	<i>BBCCBCBBCBBCBCT</i>
•	39CD	44	<i>BBCBCBBBT</i>
	40A	18	<i>BBCBBCBBCBCT</i>
	40B	1	<i>BBCBBCBCBBCBBCBCT</i>
	40CD	688	<i>BCCBCBBT</i>

*Continued on the next page*

Table 2: Words for conjugacy class representatives, *continued*

$\langle B, C, E \rangle$	Class	Number	Word
	41A	4949	<i>BCCT</i>
	42C	75	<i>BCBBCBCBBCBCBT</i>
	42A	1	<i>BBCBBCBCCBCBBCBCBCBT</i>
•	42B	1	<i>BCBCBCBBCBCBBCBT</i>
	42D	43	<i>BCBCBCCBCBT</i>
	44AB	1657	<i>BCBCBCBCBCBT</i>
•	45A	2049	<i>BCBT</i>
	46CD	21	<i>BBCCT</i>
	46AB	5	<i>BBCBBCBT</i>
	47AB	4670	<i>BCBCBCBBT</i>
	48A	617	<i>BCBBCBCBBCBBT</i>
	50A	3342	<i>BBCBBCCT</i>
	51A	1102	<i>BCBBCBCBBT</i>
	52A	5	<i>BBCCBCBCBCBT</i>
	52B	33	<i>BBCBCBBT</i>
•	54A	2614	<i>BCBBCBCBBCBBT</i>
	55A	2035	<i>BBCBBCBCBBCBT</i>
	56BC	8	<i>BBCBBCBCCBCBBCBBT</i>
	56A	6	<i>BBCCBBCBCBBT</i>
	57A	1744	<i>BCCCT</i>
	59AB	3573	<i>BCBCBCBBCCT</i>
	60A	4	<i>BCBBCBCBCCBCBT</i>
	60B	1	<i>BCBCBCBCBCBBT</i>
	60F	80	<i>BBCT</i>
•	60E	35	<i>BCBCCBCBT</i>
•	60C	5	<i>BBCBCCBCBBT</i>
•	60D	2	<i>BCBCBCBCBCCCT</i>
	62AB	7404	<i>BCBCBT</i>
	66A	5	<i>BBCBBCBCBBCBCCBCBBCBCBT</i>
•	66B	290	<i>BBCBCBBCBCBT</i>
	68A	2273	<i>BCBCBCBT</i>
	69AB	3406	<i>BBCBCCCT</i>
	70B	3	<i>BBCBBCBCBBCBCBBT</i>
	70A	41	<i>BBCBCBCBT</i>
	71AB	2588	<i>BCBBCBCBT</i>
•	78BC	14	<i>BBCBCBBCBBT</i>

*Continued on the next page*

Table 2: Words for conjugacy class representatives, *continued*

$\langle B, C, E \rangle$	Class	Number	Word
	78A	30	BBCBCBCBBBT
	84C	2	BBCBBCBBCBBCBCCCT
	84A	3	BBCBBCBBCBCBBCBCCCT
•	84B	8	BBCBBCBBCBCCBBT
	87AB	3521	BCBBCBCT
	88AB	2130	BCBCBCCBCT
	92AB	5731	BBCBCBCBT
	93AB	6910	BCBBCBBBT
	94AB	1452	BCCBCBCT
	95AB	2313	BCBCBBCBBBT
	104AB	1367	BCCBCBCT
	105A	888	BBCCBCBCBCBCBCT
	110A	238	BCBCBBCBCBCBCCBT
	119AB	3248	BBCBBCBBBT

Appendix A. *Monster element database*

Due to the large number of words being used, it became obvious early on that some sort of database was needed. We store elements and their trace information in text files, one file for each order of elements in  $\mathbb{M}$ . A suite of Perl scripts manage the database; in particular, they can determine what needs to be calculated, and they produce input data files for the trace-finding programs.

The main database programs and data are included with this paper, and can be found at

<http://www.lms.ac.uk/jcm/8/lms2005-007/appendix-a>.

See the README.txt file included there, for an explanation of how to use the programs and data.

References

1. J. H. CONWAY, R. T. CURTIS, S. P. NORTON, R. A. PARKER and R. A. WILSON, *ATLAS of finite groups* (Oxford University Press, 1985; reprinted with corrections, 2004). 207
2. J. H. CONWAY and S. P. NORTON, ‘Monstrous moonshine’, *Bull. London Math. Soc.* 11 (1979) 308–339. 205
3. P. E. HOLMES, S. A. LINTON and S. H. MURRAY, ‘Product replacement in the monster’, *Experiment. Math.* 12 (2003) 123–126. 209
4. S. A. LINTON, R. A. PARKER, P. WALSH and R. A. WILSON, ‘Computer construction of the monster’, *J. Group Theory* 1 (1998) 307–337. 205, 206, 209, 210
5. I. A. I. SULEIMAN, P. G. WALSH and R. A. WILSON, ‘Conjugacy classes in sporadic simple groups’, *Comm. Algebra* 28 (2000) 3209–3222. 205, 207

6. R. A. WILSON, 'Standard generators for sporadic simple groups', *J. Algebra* 184 (1996) 505–515. 206
7. R. A. WILSON, 'A construction of the monster group over  $GF(7)$  and an application', preprint, University of Birmingham, 2000;  
<http://web.mat.bham.ac.uk/R.A.Wilson/pubs/Mcon7.html>. 205, 206
8. R. A. WILSON, 'Conjugacy class representatives in Fischer's baby monster', *LMS J. Comput. Math.* 5 (2002) 175–180. 205
9. R. A. WILSON *et al.*, 'A world wide web atlas of group representations' (1995–2005), <http://brauer.maths.qmul.ac.uk/Atlas/>. 206, 209

R. W. Barraclough [R.W.Barraclough@qmul.ac.uk](mailto:R.W.Barraclough@qmul.ac.uk)

<http://www.maths.qmul.ac.uk/~rwb>

R. A. Wilson [R.A.Wilson@qmul.ac.uk](mailto:R.A.Wilson@qmul.ac.uk)

<http://www.maths.qmul.ac.uk/~raw>

School of Mathematical Sciences

Queen Mary

University of London

Mile End Road

London E1 4NS