HOMOTOPY 2-TYPES OF LOW ORDER

GRAHAM ELLIS AND LE VAN LUYEN

Abstract. There is a well-known equivalence between the homotopy types of connected CW-spaces \( X \) with \( \pi_nX = 0 \) for \( n \neq 1, 2 \) and the quasi-isomorphism classes of crossed modules \( \partial : M \to P \) [11]. When the homotopy groups \( \pi_1X \) and \( \pi_2X \) are finite one can represent the homotopy type of \( X \) by a crossed module in which \( M \) and \( P \) are finite groups. We define the order of such a crossed module to be \( |\partial| = |M| \times |P| \), and the order of a quasi-isomorphism class of crossed modules to be the least order of any crossed module in the class. We then define the order of a homotopy 2-type \( X \) to be the order of the corresponding quasi-isomorphism class of crossed modules. In this paper we describe an implemented computer function that inputs a finite crossed module of reasonably small order and returns a quasi-isomorphic crossed module of least order. The function is used to enumerate all homotopy 2-types of order \( m \leq 127 \), \( m \neq 32, 64, 81, 96 \). Underlying the function is a catalogue of all isomorphism classes of crossed modules of order \( m \leq 255 \).

1. Introduction

An important resource for finite group theorists is the computer classification of all groups \( G \) of low order. This classification is available in the MAGAM [3] and GAP [7] computer systems and, for example, can be used to: (i) list representatives of all isomorphism classes of groups \( G \) of a given order \( m \); (ii) identify the isomorphism class of a user-defined group \( G \) in terms of a pair \((m, k)\) where \( m \) is the order of \( G \) and \( k \) is a catalogue number.

In this paper we build on work of Alp and Wensley [2] and develop the beginnings of an analogous resource for homotopy types of connected CW-spaces \( X \) with \( \pi_nX = 0 \) for \( n \neq 1, 2 \). The homotopy type of \( X \) is called a homotopy 2-type. It is well-known that such a homotopy type can be modelled by a group homomorphism \( \partial : M \to P \) and group action \((p, m) \mapsto pm\) of \( P \) on \( M \) satisfying

1. \( \partial(pm) = p(\partial m)p^{-1} \)
2. \( \partial m'm = mm'm^{-1} \)

for \( p \in P, m, m' \in M \). Such a homomorphism and action constitute a crossed module. The model is such that \( \pi_nX \cong \pi_n(\partial) \) for \( n = 1, 2 \) where one defines \( \pi_1(\partial) = P/\text{im} \partial \) and \( \pi_2(\partial) = \ker \partial \). A morphism of crossed modules \( \phi_* : (\partial : M \to P) \to (\partial' : M' \to P') \) consists of two group homomorphisms \( \phi_1 : P \to P', \phi_2 : M \to M' \) satisfying \( \partial' \phi_2(m) = \phi_1 \partial(m) \) and \( \phi_2(pm) = \phi_1 \phi_2(m) \) for \( m, m' \in M, p \in P \). A morphism induces canonical homomorphisms \( \pi_n(\phi_*) : \pi_n(\partial) \to \pi_n(\partial') \) for \( n = 1, 2 \). The morphism \( \phi_* \) is said to be an isomorphism if \( \phi_* \) is...
an isomorphism for \( n = 1, 2 \). The morphism \( \phi_* \) is said to be a \textit{quasi-isomorphism} if \( \pi_n(\phi_*) \) is an isomorphism for \( n = 1, 2 \). Two crossed modules \( \partial, \partial' \) are said to be \textit{quasi-isomorphic} if there exists a sequence of quasi-isomorphisms \( \partial \to \partial_1 \leftarrow \partial_2 \to \partial_3 \leftarrow \cdots \to \partial_n \leftarrow \partial' \) of arbitrary length \( n \). We write \( \partial \simeq \partial' \) to denote that \( \partial \) is quasi-isomorphic to \( \partial' \). Note that \( \simeq \) is an equivalence relation on crossed modules; the corresponding equivalence classes are called \textit{quasi-isomorphism classes}. We emphasize that two crossed module \( \partial, \partial' \) could be quasi-isomorphic without the existence of any quasi-isomorphism between \( \partial \) and \( \partial' \).

MacLane and Whitehead [11] showed that there is a one-one correspondence between homotopy 2-types and quasi-isomorphism classes of crossed modules. We define the \textit{order} of a crossed module \( \partial \colon \text{Iso} \to \text{M} \), to be the product \( |\partial| = |\text{Iso}| \times |\text{M}| \) of the orders of the groups \( \text{Iso}, \text{M} \). We define the \textit{order} of a quasi-isomorphism class of crossed modules to be the least order of any crossed module in the class. We define the \textit{order} of a homotopy 2-type \( X \) to be the order of the corresponding quasi-isomorphism class of crossed modules.

A homotopy 2-type \( X \) can also be represented by the fundamental group \( \pi_1 X \), the \( \pi_1 X \)-module \( \pi_2 X \) and a cohomology class \( \kappa \in H^3(\pi_1 X, \pi_2 X) \) known as the Postnikov invariant. The Postnikov invariant \( \kappa \) is the trivial cohomology class if and only if the homotopy 2-type \( X \) can be represented by a crossed module \( \partial \colon \text{M} \to \text{P} \) with \( \partial = 0 \). In this case we deem the homotopy 2-type, and also the quasi-isomorphism type, to be \textit{trivial}.

In this paper we describe two computer functions, both of which have been implemented by the second author in the HAP package [5] for the computer algebra system GAP [7]. The first function lists representatives for all the quasi-isomorphism classes of crossed modules of a given order \( m \leq 127, m \neq 32, 64, 81, 96 \). The second function inputs a user-defined crossed module (of order possibly greater than 127) and tries to return numbers \((m, k)\) that identify the least order \( m \) of any quasi-isomorphic crossed module and a catalogue number \( k \) that uniquely identifies the quasi-isomorphism class of the input; it certainly succeeds if the input is of order \( \leq 127, \neq 32, 64, 81, 96 \). We have used the implementation of these two functions, and related functions, to compile Table 1. The table uses the notation:

\[
\begin{align*}
\text{Iso2}(m) &= \text{number of isomorphism classes of crossed modules of order } m. \\
\text{QIso2}(m) &= \text{number of homotopy 2-types of order } m \\
\text{TQIso2}(m) &= \text{number of quasi-isomorphism classes of order } m.
\end{align*}
\]

It is an easy exercise to see that \( \text{Iso2}(p) = \text{QIso2}(p) = \text{TQIso2}(p) = 2 \) for \( p \) a prime and so we omit prime values of \( m \) from the table. It is also easy to show that for primes \( p < q \) we have \( \text{Iso2}(pq) = \text{QIso2}(pq) = \text{TQIso2}(pq) = 6 \) when \( p \) divides \( q - 1 \) and \( \text{Iso2}(pq) = \text{QIso2}(pq) = \text{TQIso2}(pq) = 4 \) when \( p \) does not divide \( q - 1 \) and so these values of \( m \) are also omitted from the table. (To establish the formulae one uses that: the cyclic group of order \( p \) can act non-trivially on the cyclic group of order \( q \) precisely when \( p \) divides \( q - 1 \); the only groups of order \( p \) or order \( pq \) with \( p \) not dividing \( q - 1 \) are the cyclic groups; the only groups of order \( pq \) with \( p \) dividing \( q - 1 \) are the cyclic group and one non-abelian semi-direct product of cyclic groups.)
Perhaps not surprisingly, the table shows that most of the homotopy 2-types of low order have trivial Postnikov invariant. It shows that the smallest homotopy 2-type with non-trivial Postnikov invariant has order 16, and that there is just one non-trivial homotopy 2-type of this order. A straightforward computer analysis shows that this homotopy type is represented by the crossed module with $M = \langle x \mid x^4 = 1 \rangle$, $P = \langle a \mid a^4 = 1 \rangle$, $a x = x^3$, $\partial(x) = a^2$. It is also represented by the crossed module $M = \langle x, y \mid x^2 = y^2 = [x, y] = 1 \rangle$, $P = \langle a \mid a^4 = 1 \rangle$, $a x = xy$, $a y = y$, $\partial(x) = a^2$, $\partial(y) = 1$. No other crossed module of order 16 represents the unique smallest homotopy 2-type with non-trivial Postnikov invariant.

We remark that it has been observed previously that the second of the crossed modules representing this homotopy type corresponds to a non-trivial Postnikov invariant; see for instance the example of Section 7 in [8] and Example 12.7.12 in [4].

The values of $\text{Iso}_2(m)$ for $m \leq 63$ are available from the software [1] described in [2]. The values of $\text{Iso}_2(m)$ for higher $m$ are obtained from a function for listing non-isomorphic crossed modules of given order which was designed and implemented by the second author. Details are given in Section 3. We are grateful to Alexander Hulpke for providing a key step in the implementation of our algorithm for testing isomorphism of two crossed modules.

For each of the $\text{Iso}_2(m)$ crossed modules of order $m$ we apply a refinement of an algorithm in our previous paper [6] that attempts to find a smaller quasi-isomorphic crossed module. In this way we obtain an upper bound for $\text{QIso}_2(m)$. Details are given in Section 4.

\begin{table}[h]
\centering
\begin{tabular}{c|cccccccccccccccccccccc}
\hline
$m$ & 1 & 4 & 8 & 9 & 12 & 16 & 18 & 20 & 24 & 25 & 27 & 28 & 30 & 32 & 36 & 40 \\
\hline
$\text{Iso}_2(m)$ & 1 & 6 & 18 & 6 & 20 & 62 & 22 & 20 & 73 & 6 & 18 & 18 & 20 & 16 & 20 & 251 & 78 & 72 \\
$\text{QIso}_2(m)$ & 1 & 5 & 14 & 5 & 18 & 43 & 19 & 18 & 61 & 5 & 14 & 16 & 20 & A & 63 & 60 \\
$\text{TQIso}_2(m)$ & 1 & 5 & 14 & 5 & 18 & 42 & 19 & 18 & 61 & 5 & 14 & 16 & 20 & 152 & 63 & 60 \\
\hline
$m$ & 42 & 44 & 45 & 48 & 49 & 50 & 52 & 54 & 56 & 60 & 63 & 64 & 66 & 68 & 70 & 72 \\
\hline
$\text{Iso}_2(m)$ & 26 & 18 & 12 & 296 & 6 & 22 & 20 & 81 & 68 & 77 & 18 & 12 & 76 & 20 & 18 & 20 & 251 \\
$\text{QIso}_2(m)$ & 26 & 16 & 10 & 224 & 5 & 19 & 18 & 65 & 56 & 73 & 16 & B & 20 & 18 & 20 & 251 \\
$\text{TQIso}_2(m)$ & 26 & 16 & 10 & 220 & 5 & 19 & 18 & 65 & 56 & 73 & 16 & 697 & 20 & 18 & 20 & 251 \\
\hline
$m$ & 75 & 76 & 78 & 80 & 81 & 84 & 88 & 90 & 92 & 96 & 98 & 99 & 100 & 102 & 104 \\
\hline
$\text{Iso}_2(m)$ & 14 & 18 & 26 & 302 & 64 & 90 & 66 & 76 & 18 & 1446 & 22 & 12 & 87 & 20 & 20 & 72 \\
$\text{QIso}_2(m)$ & 12 & 16 & 26 & 230 & C & 84 & 54 & 66 & 16 & D & 19 & 10 & 71 & 20 & 60 \\
$\text{TQIso}_2(m)$ & 12 & 16 & 26 & 226 & 5 & 19 & 18 & 65 & 56 & 73 & 16 & 971 & 19 & 10 & 71 & 20 & 60 \\
\hline
$m$ & 105 & 108 & 110 & 112 & 114 & 116 & 117 & 120 & 121 & 124 & 125 & 126 & 128 \\
\hline
$\text{Iso}_2(m)$ & 12 & 308 & 26 & 270 & 26 & 20 & 18 & 342 & 6 & 18 & 18 & 102 & 9120 \\
$\text{QIso}_2(m)$ & 12 & 238 & 26 & 202 & 26 & 18 & 16 & 302 & 5 & 16 & 14 & 92 & ? \\
$\text{TQIso}_2(m)$ & 12 & 238 & 26 & 198 & 26 & 18 & 16 & 302 & 5 & 16 & 14 & 92 & 4668 \\
\end{tabular}
\caption{Table 1}
\end{table}
To prove that the upper bound equals $QIso_2(m)$ we need a method for establishing that two crossed modules $\partial, \partial'$ are not quasi-isomorphic. We do this by computing the quasi-isomorphism invariants $\pi_1(\partial), \pi_2(\partial), H^3(\pi_1(\partial), \pi_2(\partial))$ and $H_n(X, \mathbb{Z})$. The last invariant is the homology of the homotopy 2-type $X$ represented by $\partial$ and is computed using our algorithm described in [6]. We use the group cohomology routines in our HAP package [5] to compute $H^3(\pi_1(\partial), \pi_2(\partial))$. Details are given in Section 5.

We begin by illustrating the functionality of our computer implementation in Section 2.

2. Computer implementation

It is well-known that the notion of a crossed module can be reformulated as a “group with compatible category structure”. We use such a reformulation both for implementing algorithms and for checking correctness of algorithms. There are several variants of the reformulation and we opt to work with the following notion due to J-L. Loday [10].

A $\text{cat}^1$-group consists of a pair of group endomorphisms $s, t: G \to G$ satisfying $ts = s, st = t$ and $[\ker s, \ker t] = 1$. A morphism of $\text{cat}^1$-groups $\phi: (G, s, t) \to (G', s', t')$ consists of a group homomorphism $\phi: G \to G'$ satisfying $\phi s = s' \phi$ and $\phi t = t' \phi$. A $\text{cat}^1$-group gives rise to a crossed module by taking $M = \ker s, P = \im s$ and taking $\partial$ to be the restriction of $t$ to $\ker s$. Conversely, a crossed module gives rise to a $\text{cat}^1$-group by using the action of $P$ on $M$ to form the semi-direct product $G = M \rtimes P$ and defining the endomorphisms $s, t: M \rtimes P \to M \rtimes P$ as $s(m, p) = (1, p), t(m, p) = (1, (\partial m)p)$. It is observed in [10] that these two constructions provide an isomorphism between the category of crossed modules and the category of $\text{cat}^1$-groups. It is thus routine to translate notions of order, homotopy group, quasi-isomorphism and (trivial) quasi-isomorphism class of crossed modules to equivalent notions for $\text{cat}^1$-groups. We leave details to the reader and use these equivalent notions throughout the remainder of the paper.

Our first GAP session begins by setting $G$ equal to the 500th group of order 2000 from the database of small groups. It then computes a list $L$ of all possible non-isomorphic $\text{cat}^1$-group structures on $G$. The list $L$ has length 16.

```
gap> G:=SmallGroup(2000,500);;
gap> L:=CatOneGroupsByGroup(G);
gap> Length(L);
16
```

Our second GAP session involves the homomorphism $\alpha_H: H \to \text{Aut}(H), h \mapsto \iota_h$ from a group $H$ to its automorphism group which sends $h \in H$ to the inner automorphism $\iota_h: H \to H, x \mapsto hxh^{-1}$. The homomorphism $\alpha_H$ is a crossed module with respect to the obvious action of $\text{Aut}(H)$ on $H$. The session begins by constructing the associated $\text{cat}^1$-group $G_1$ for $H$ equal to the dihedral group of order 12. The second command in the session determines that the underlying group of $G_1$ is the 154th group of order 144 in the
small groups database, and that \( G_1 \) is endowed with the 8th cat\(^1\)-structure on this group. The session then constructs the cat\(^1\)-group \( G_2 \) associated to \( \alpha_H \) for \( H \) the dihedral group of order 72 and computes \( |G_2| = 31104 \). The final command identifies the quasi-isomorphism class of \( G_2 \) to be the 55th quasi-isomorphism class of order 24.

\[
\text{gap} > G1:=\text{AutomorphismGroupAsCatOneGroup}(\text{DihedralGroup}(12));;
\]
\[
\text{gap} > \text{IdCatOneGroup}(G1);
[ 144, 154, 8 ]
\]
\[
\text{gap} > G2:=\text{AutomorphismGroupAsCatOneGroup}(\text{DihedralGroup}(72));;
\]
\[
\text{gap} > \text{Size}(G2);
31104
\]
\[
\text{gap} > \text{IdQuasiCatOneGroup}(G2);
[ 24, 55 ]
\]

Our third \text{gap} session begins by constructing the cat\(^1\)-group corresponding to the 2nd homotopy 2-type \( X \) of order 30. It then uses the algorithm from [6] to compute \( H_5(X, \mathbb{Z}) = \mathbb{Z}_{10} \).

\[
\text{gap} > G:=\text{SmallQuasiCatOneGroup}(30, 2);
\]
\[
\text{gap} > \text{Homology}(G, 5);
[ 10 ]
\]

The implementation contains functions for converting a crossed module to a cat\(^1\)-group and vice-versa. Each of the above \text{gap} sessions could thus equally well have been performed using equivalent crossed modules.

3. Enumeration of isomorphism classes

The \text{gap} package [1] of Alp and Wensley provides a list of all non-isomorphic cat\(^1\)-groups and crossed modules of order \( \leq 63 \). To handle larger examples the second author implemented a function which inputs a finite group \( G \) and outputs a list of all non-isomorphic cat\(^1\)-group structures \((G, s, t)\). This implementation uses \text{gap}'s function \text{IdGroup}(H) for identifying certain subgroups \( H \leq G \) by their order \( m \) and catalogue number \( k \) and thus works only in cases for which \text{IdGroup}(H) is implemented.

The algorithm begins by computing a list \( \mathbb{L} \) of all normal subgroups \( N \) in \( G \) and a list \( \mathbb{L}' \) of subgroups \( K \) in \( G \) representing all subgroup conjugacy classes. There are then two steps to the algorithm.

**Step 1.** For each \( N \in \mathbb{L} \) we find all \( K \in \mathbb{L}' \) satisfying
For each action we have adapted a gap stabilizer algorithm can be found in [9].

Let $K$ be a group and to compute the stabilizer subgroup of this element. A description of the orbit-writer by Alexander Hulpke and used it to compute the orbit of an element under the action of $G$, we return `false`.

For each such pair $N, K$ the quotient homomorphism $p: G \rightarrow G/N$ restricts to an isomorphism $p|_K: K \rightarrow G/N$. We form the inverse isomorphism $(p|_K)^{-1}: G/N \rightarrow K$ and set $\sigma = (p|_K)^{-1}p: G \rightarrow G$. By construction we have $\ker \sigma = N$, $\im \sigma = K$ and $\sigma \sigma = \sigma$. For each normal subgroup $N$ we compute the list $\mathbb{L}_N$ of such homomorphisms $\sigma$.

**Step 2.** For each pair of normal subgroups $N, M$ in $G$ satisfying $[N, M] = 1$ we consider all $s \in \mathbb{L}_N$, $t \in \mathbb{L}_M$. If $\im s = \im t$ we add the data $(G, s, t)$ to our list of cat$^1$-group structures on $G$.

In this manner, all possible cat$^1$-group structures on $G$ are produced, though isomorphic copies may have been produced by the algorithm.

To test if two cat$^1$-group structures on a group $G$ are isomorphic we need to access the automorphism group $\text{Aut}(G)$ of the group $G$. As this automorphism group can be large we follow a suggestion of Alexander Hulpke and use:

(i) the action $fK = f(K)$ of $f \in \text{Aut}(G)$ on subgroups $K \leq G$;

(ii) the action $fs(x) = f(s(x)f^{-1}$ of $f \in \text{Aut}(G)$ on endomorphisms $s: G \rightarrow G$.

For each action we have adapted a GAP implementation of an orbit-stabilizer algorithm written by Alexander Hulpke and used it to compute the orbit of an element under the action and to compute the stabilizer subgroup of this element. A description of the orbit-stabilizer algorithm can be found in [9].

To test if two cat$^1$-group structures $(G, s, t)$ and $(G, s', t')$ are isomorphic we perform the following steps.

**Step 1.** We first use GAP’s `IdGroup()` function to check that $\im s \cong \im s'$ and $\ker s \cong \ker s'$ and $\ker t \cong \ker t'$. If this check fails then the two cat$^1$-groups are not isomorphic and we return `false`.

**Step 2.** Otherwise we compute the orbit of $\ker s$ under the action of $\text{Aut}(G)$. If $\ker s'$ is not in this orbit then the two cat$^1$-groups are not isomorphic and we return `false`. Otherwise we can find an element $f \in \text{Aut}(G)$ such that $\ker s' = f(\ker s)$. We then define $s'' = f^{-1}s'$, $t'' = f^{-1}t'$ to obtain a cat$^1$-group $(G, s'', t'')$ which is isomorphic to $(G, s', t')$ and which has the property that $\ker s'' = \ker s$. For ease of notation we redefine $s' := s''$, $t' := t''$. In other words, we replace $(G, s', t')$ by an isomorphic cat$^1$-group satisfying $\ker s' = \ker s$.

**Step 3.** We compute the stabilizer subgroup $\text{Stab}(\ker s) \leq \text{Aut}(G)$ and the orbit of $\im s$ under the action of $\text{Stab}(\ker s)$. If $\im s'$ is not in this orbit then the two cat$^1$-groups are not isomorphic and we return `false`. Otherwise we can find an element $f \in \text{Stab}(\ker s)$ such that $\im s' = f(\im s)$ and then replace $(G, s', t')$ by an isomorphic cat$^1$-group satisfying $\im s' = \im s$ and $\ker s' = \ker s$.

**Step 4.** We compute the stabilizer subgroup $\text{Stab}(\im s) \leq \text{Stab}(\ker s)$ and the orbit of $\ker t$ under the action of $\text{Stab}(\im s)$. If $\ker t'$ is not in this orbit then the two cat$^1$-groups are not isomorphic and we return `false`. Otherwise we replace $(G, s', t')$ by an isomorphic cat$^1$-group satisfying $\ker t' = \ker t$, $\im s' = \im s$ and $\ker s' = \ker s$.

**Step 5.** We compute the stabilizer $\text{Stab}(\ker t) \leq \text{Stab}(\im s)$ and the orbit of $s$ under the action of $\text{Stab}(\ker t)$. If $s'$ is not in this orbit the two cat$^1$-groups are not isomorphic and
we return false. Otherwise we replace \((G, s', t')\) by an isomorphic \(cat^1\)-group satisfying \(\ker t' = \ker t, s' = s\).

**Step 6.** We compute the stabilizer \(Stab(s) \leq Stab(\ker t)\) and the orbit of \(t\) under the action of \(Stab(s)\). If \(t'\) is not in this orbit then the two \(cat^1\)-groups are not isomorphic and we return false. Otherwise we return true.

4. **Construction of small quasi-isomorphic representatives**

Given a \(cat^1\)-group \((G, s, t)\) we attempt to find a smaller quasi-isomorphic \(cat^1\)-group using (a new implementation of) the following procedure which was described in [6].

**Step 1:** Given a finite \(cat^1\)-group \((G, s, t)\) we search through the normal subgroups of \(\ker s\) to find a largest normal subgroup \(K\) in \(\ker s\) such that \(K\) is normal in \(G\) and \(K \cap \ker t = 1\). We can then choose some generating set \(X_K\) for \(K\) and construct the group \(N\) generated by the set \(X_K \cup \{t(x) : x \in X_K\}\). Then \(N\) is a normal subgroup of \(G\) for which the quotient homomorphism \(G \to G/N\) is a quasi-isomorphism.

**Step 2:** Given a finite \(cat^1\)-group \((G, s, t)\) we search through the subgroups of \(G\) to find a smallest \(cat^1\)-group \(K \subset G\) such that the inclusion \(K \to G\) is a quasi-isomorphism. Since we are dealing with finite groups we can test if this inclusion is a quasi-isomorphism simply by checking if the induced homomorphisms \(\pi_n K \to \pi_n G\) are surjections and \(|\pi_n K| = |\pi_n G|\) for \(n = 1, 2\).

**Step 3:** We repeatedly apply (i) followed by (ii) until no more size reduction is achieved.

5. **Establishing distinct quasi-isomorphism classes**

A table of all isomorphism types of \(cat^1\)-groups of order at most 255 has been computed and stored in [5]. This table underlies the function \(IdCatOneGroup(G)\) which returns a triple \((m, k_1, k_2)\) for any \(cat^1\)-group \(G\) of order \(m \leq 255\); the integer \(k_1\) is the number of the underlying group of \(G\) in the small groups database; the integer \(k_2\) is the number of the \(cat^1\)-group structure on \(G\) in our table of small \(cat^1\)-groups.

The table of isomorphism types of \(cat^1\)-groups immediately yields the upper bound \(Iso_2(m) \geq QIso_2(m)\) on the number of quasi-isomorphism types of \(cat^1\)-groups. By applying the procedure of Section 4 to each isomorphism type \(G\) of order \(m\) in the table, and then discarding \(G\) if the procedure succeeds in finding a smaller \(cat^1\)-group quasi-isomorphic to \(G\), we obtain a list \(L\) of representatives of quasi-isomorphism types of \(cat^1\)-groups of order \(m\). It could be that some pair \(G, G' \in L\) are quasi-isomorphic. For each \(G \in L\) we have computed the following quasi-isomorphism invariants:

i) the small groups database identifier for \(\pi_1(G)\);
ii) the small groups database identifier for \(\pi_2(G)\);
iii) the small groups database identifier for the semi-direct product \(\pi_2(G) \rtimes \pi_1(G)\) involving the action of the first homotopy group on the second homotopy group;
iv) the abelian invariants of the integral homology group \(H_n(X, \mathbb{Z})\) for \(n \leq 5\) where \(X\) is the homotopy 2-type represented by \(G\). The homology algorithm from [6] was used for this.
If for every pair of non-isomorphic cat\(^1\)-groups \(G, G' \in L\) at least one of the invariants (i)-(iv) yields distinct values we conclude that our list \(L\) contains precisely one representative for each quasi-isomorphism class of order \(m\).

In cases where invariants (i)-(iv) are identical for two cat\(^1\)-groups \(G, G' \in L\) the order of the cohomology group \(H^3(\pi_1(G), \pi_2(G))\) was computed using group cohomology functions in [5]. By Mac Lane and Whitehead’s result [11], the order of this cohomology group provides an upper bound on the number of quasi-isomorphism types with given fundamental group \(\pi_1(G)\) and given second homotopy group \(\pi_2(G)\). In some cases this upper bound is sufficient to conclude that \(G\) and \(G'\) are quasi-isomorphic.

Using the above strategy a non-redundant list of representatives of all quasi-isomorphism types of order \(m\) has been computed and recorded in the software [5] for all \(m \leq 127\) excluding \(m = 32, 64, 81, 96\). This record is used to implement the function \(\text{IdQuasiCatOneGroup}(G)\) which inputs a cat\(^1\)-group \(G\) and tries to return the the pair of integers \((m, k)\) with \(m\) the order of the smallest cat\(^1\)-group in the quasi-isomorphism class of \(G\) and \(k\) a number uniquely identifying this smallest representative. The function first attempts to find a quasi-isomorphism representative of \(G\) of order \(\leq 127, \neq 32, 64, 81, 96\); if it succeeds it then uses the stored record to produce \(m\) and \(k\).

For \(m = 32, 64, 81, 96\) the above strategy produces the bounds \(158 \leq Q\text{Iso}_2(32) \leq 171, 727 \leq \text{Iso}_2(64) \leq 831, 45 \leq \text{Iso}_2(81) \leq 46\) and \(996 \leq \text{Iso}_2(96) \leq 1052\).

References


Graham Ellis, School of Mathematics, National University of Ireland, Galway
E-mail address: graham.ellis@nuigalway.ie

Le Van Luyen, School of Mathematics, National University of Ireland, Galway
E-mail address: v.le2@nuigalway.ie