ENUMERATING LEFT DISTRIBUTIVE GROUPOIDS

Jaroslav Ježek

Abstract. Groupoids satisfying the equation $x(yz) = (xy)(xz)$ are called left distributive, or LD-groupoids. We give an algorithm for their enumeration and prove several results on the collection of LD-groupoids extending a given monounary algebra.

0. Introduction

Suppose that we want to enumerate all groupoids (i.e., algebras with one multiplicatively denoted binary operation) on a given finite set A of n elements that satisfy a given finite collection E of equations. The groupoids can be identified with their multiplication tables, and we can suppose that $A = \{0, 1, \dots, n-1\}$. A simple algorithm can do the task: generate the tables of all groupoids on A according to their lexicographic order, and for each groupoid check if it satisfies the equations from E. Of course, this can work in a reasonable time for very small numbers n only, as the number of all tables is n^{n^2}. With $n = 5$, we would have to check 5^{25} tables, which is already too much.

If the collection E is sufficiently strong, the number of groupoids satisfying E on a set of n elements can be essentially smaller than the number of all groupoids. So, one can ask for a faster algorithm, avoiding large intervals in the lexicographic ordering for which it is clear from a simple reason that they cannot contain a table satisfying E. One such algorithm is given in Section 3 of this paper. The algorithm is written in the language C, but does not use any hard to understand features of the language. It is formulated for the special case of E consisting of the left distributive law.

This algorithm makes it possible to find the number of isomorphism types of n-element left distributive groupoids for both $n = 5$ and $n = 6$. The numbers are given in Section 4.

Let P be a partial groupoid. By an LD-extension of P we mean a left distributive groupoid G with the underlying set P, such that $xy = z$ in P implies $xy = z$ in G. If xy is defined in P for the pairs x, y with $x = y$ and no other ones, then P can be identified with a monounary algebra. In Section 1 we investigate the collection of LD-extensions of a given monounary algebra. We determine under what conditions there exists precisely one LD-extension, and for some monounary algebras we are able to give a simple description of all the LD-extensions. The conjectures to some of the results were obtained by running a version of the algorithm given in Section 3.

In Section 2 we are concerned with finite zeropotent left distributive groupoids. We prove that all of them satisfy the equation $x(yz) = 0$, and find recursive formulas for their enumeration.

In two related papers [6] and [7], finite left distributive groupoids with one generator are completely described. Related are also the papers [1], [2], [3], [4], [5], [8], [9], and [10].

1. Finite left distributive groupoids extending a given monounary algebra

Let (A, x') be a monounary algebra (i.e., an algebra with one unary operation). By an LD-extension of (A, x') we shall mean an LD-groupoid (A, xy) such that $xx = x'$ for all $x \in A$.

Let (A, x') be given. If $a \in A$, then $a^{(i)}$ is defined for any nonnegative integer i recursively in the following way: $a^{(0)} = a$ and $a^{(i+1)} = a'^{(i)}$. The set $\{a^{(i)} : i \geq 0\}$ is called the orbit of a. Two elements of A are called connected if their orbits are not disjoint. This relation on A is an equivalence; its blocks will be called components of (A, x').

Let C be a component of (A, x'). The intersection of all orbits of elements of C is called the cycle of C. The cycle is nonempty and coincides with the orbit of any of its elements. An element $a \in A$ is called irreducible if there is no element b with $a = b'$.

Of course, every LD-groupoid is an LD-extension of precisely one monounary algebra. Given an LD-groupoid G, we put $x' = xx$ for every $x \in G$, introduce the notation $x^{(i)}$ and speak about orbits and components with respect to this monounary algebra.

1.1. Lemma. *Let (A, xy) be an LD-extension of (A, x'). Then:*

(1) *$(ab)' = ab'$ for any $a, b \in A$;*
(2) *if b is in the orbit of a, then $ab = b'$.*

Proof. (1) $(ab)' = ab \cdot ab = a \cdot bb = ab'$.

(2) Let us prove $aa^{(i)} = a^{(i+1)}$ by induction on i. For $i = 0$ it is clear. If $i > 0$, then $aa^{(i)} = a \cdot a^{(i-1)}a^{(i-1)} = aa^{(i-1)} \cdot aa^{(i-1)} = a^{(i)}a^{(i)} = a^{(i+1)}$. $\square$

1.2. Theorem. *Every monounary algebra (A, x') has at least one LD-extension (A, xy), e.g., $xy = y'$.*

A monounary algebra (A, x') has a unique LD-extension if and only if it has only one component and, for any irreducible element $a \in A$, $a'' = b'$ implies $b = a'$.

Proof. If $xy = y'$ for all $x, y \in A$, then $x \cdot yz = z'' = xy \cdot xz$ and (A, xy) is an LD-extension of (A, x').

Let (A, x') have more than one component. We are going to show that then (A, x') has at least two different LD-extensions. If every component consists of a single element (so that $x' = x$ for all x), then $xy = x$ and $xy = y$ are two different LD multiplications on A, both satisfying $xx = x$. So, we can assume that there is a component D with at least two elements. Take another component $C \neq D$. We can define multiplication on A by $xy = y$ if $x \in C$ and $y \in D$, and $xy = y'$ in all other cases. Clearly, $xx = x'$ for all x, and there are pairs x, y with $xy \neq y'$. It remains to prove $x \cdot yz = xy \cdot xz$ for all x, y, z. If $z \notin D$, then $x \cdot yz = z'' = xy \cdot xz$. Let $z \in D$. If $x \notin C$ and $y \notin C$, then $x \cdot yz = z'' = xy \cdot xz$. If $x \in C$ and $y \in C$, then $x \cdot yz = z = xy \cdot xz$. If precisely one of the elements x and y belongs to C, then $x \cdot yz = z' = xy \cdot xz$.

Let $a \in A$ be an irreducible element and suppose that there is an element $b \neq a'$ such that $a'' = b'$. Define multiplication on A by $a''a = b$ and $xy = y'$ whenever $(x, y) \neq (a'', a)$. If $b \neq a$, then it is easy to see that $x \cdot yz = z''$ for all x, y, z; but then also $xy \cdot xz = z''$ and we get $x \cdot yz = xy \cdot xz$. If b cannot be taken different from a, then A has only two elements and it is easy to verify that the multiplication is also left distributive in this case. Clearly, $xx = x'$ for all x and $xy = y'$ fails for $(x, y) = (a'', a)$. Hence (A, x') has at least two different LD-extensions.

Now let (A, x') have only one component and let $a'' = b'$ imply $b = a'$ for any irreducible element a. Let (A, xy) be an LD-extension of (A, x'). It remains to prove that $xy = y'$ for all $x, y \in A$. By Lemma 1.1(2), if a belongs to the cycle, then $xa = a'$ for any $x \in A$.

Let b be an irreducible element. Put $b_i = b^{(i)}$ for all i. By Lemma 1.1(2), $b_i b_j = b'_j$ for $i \leq j$. Let us prove, by induction on i, that $b_i b_j = b'_j$ for all j. For $i = 0$ this has been proved, so let $i > 0$. If $j > 0$, then

$$b_i b_j = b_{i-1} b_{i-1} \cdot b_{i-1} b_{j-1} = b_{i-1} \cdot b_{i-1} b_{j-1} = b_{i-1} b_j = b'_j,$$

so it remains to prove $b_i b_0 = b'_0$. We have $(b_i b_0)' = b_i b'_0 = b_i b_1 = b_2 = b''$; since b is irreducible, this implies $b_i b_0 = b' = b'_0$.

In particular, we get $ay = y'$ for any element a of the cycle and any element y. By the depth of an element $x \in A$ we shall mean the least nonnegative integer i such that $x^{(i)}$ belongs to the cycle. Let us prove $xy = y'$ by induction on the depth of x. If x is of depth 0, then x belongs to the cycle and we are through. Let x be of a positive depth. The depth of x' is less then the depth of x, so $x'y = y'$ for all y by induction. Let b be an irreducible element. As before, it is sufficient to prove $xb_i = b_{i+1}$ for all i, where $b_i = b^{(i)}$. Let us proceed by induction on i. Take any element a in the cycle. Since $(xb)' = a \cdot xb = ax \cdot ab = x'b' = b''$ and b is irreducible, we have $xb = b'$, i.e., $xb_0 = b_1$. For $i > 0$,

$$xb_i = x \cdot b_{i-1} b_{i-1} = xb_{i-1} \cdot xb_{i-1} = b_i b_i = b_{i+1}.$$

This proves $xy = y'$ for all $x, y \in A$. $\square$

1.3. Lemma. *Let G be a finite LD-groupoid and let a, b be two elements of G such that b belongs to a cycle and $ab = b^{(k)}$ for some $k \geq 0$. Then $a^{(i)} b^{(j)} = b^{(k+j)}$ for all i and j.*

Proof. By Lemma 1.1, $ab^{(j)} = b^{(k+j)}$ for all j. Now it is sufficient to prove $aa \cdot b = b^{(k)}$. We have $aa \cdot b^{(k)} = aa \cdot ab = a \cdot ab = ab^{(k)} = b^{(k+k)}$, so that $aa \cdot b^{(k+j)} = b^{(k+k+j)}$ for all $j \geq 0$. Since b belongs to a cycle, for a suitable j this means that $aa \cdot b = b^{(k)}$. $\square$

Let $n_1, \ldots, n_d$ (where $d \geq 1$) be positive integers; for every pair i, j of elements of $\{1, \ldots, d\}$ with $i \neq j$ let $m_{i,j}$ be a number from $\{0, \ldots, n_j - 1\}$. We denote by $H_{n_i, m_{i,j}}$ the groupoid with the underlying set $\{(i, k) : i \in \{1, \ldots, d\}, k \in \{0, \ldots, n_i - 1\}\}$ and multiplication given by

$$(i, k)(j, l) = (j, l +_j m_{i,j})$$

where $m_{i,i} = 1$ and $+_j$ denotes addition modulo n_j.

1.4. Theorem. *$H_{n_i,m_{i,j}}$ is an LD-groupoid of cardinality $n_1 + \cdots + n_d$ and with cycles of cardinalities $n_1, \ldots, n_d$.*

Let $n_1, \ldots, n_d$ be positive integers such that n_i is not a multiple of n_j for any $i \neq j$. Then every LD-groupoid of cardinality $n_1+\cdots+n_d$ with cycles of cardinalities $n_1, \ldots, n_d$ is isomorphic to $H_{n_i,m_{i,j}}$ for some collection of numbers $m_{i,j}$ $(i \neq j)$ as above. There are precisely $(n_1 n_2 \ldots n_d)^{d-1}$ isomorphism types of such groupoids.

Proof. The first assertion is straightforward. Let G be an LD-groupoid which is the disjoint union of cycles of mutually prime cardinalities $n_1, \ldots, n_d$. Define a bijection f of $\{(i,k) : i \in \{1, \ldots, d\}, k \in \{0, \ldots, n_i - 1\}\}$ onto G in the following way. For every $i \in \{1, \ldots, d\}$ let $f(i,0)$ be an arbitrarily chosen element of the cycle of cardinality n_i, and put $f(i,k) = f(i,0)^{(k)}$ for all k. By Lemma 1.1, $f(i,k)f(i,l) = f(i, l +_i 1)$.

Let $i \neq j$. By Lemma 1.1, $(f(i,0)f(j,0))^{(n_j)} = f(i,0)f(j,0)$; since n_j is not a multiple of any other number from $\{n_1, \ldots, n_d\}$, it follows that $f(i,0)f(j,0) = (j, m_{i,j})$ for precisely one number $m_{i,j} \in \{0, \ldots, n_j - 1\}$. By Lemma 1.3 we get $f(i,k)f(j,l) = (j, l +_j m_{i,j})$ for all k and l. The last assertion follows easily. □

1.5. Example. The partition of an LD-groupoid into components is not necessarily a congruence. The following five-element LD-groupoid is a counterexample:

	a	b	c	d	e
a	b	a	d	d	d
b	b	a	e	d	d
c	a	b	c	d	e
d	a	b	c	d	e
e	a	b	c	d	e

Let us take $n+2$ elements $a, b, c_1, \ldots, c_n$ for some positive integer n. For every pair M, N of subsets of $\{c_1, \ldots, c_n\}$ and every reflexive relation R on $\{c_1, \ldots, c_n\}$ denote by $E_{M,N,R}$ the groupoid with the underlying set $\{a, b, c_1, \ldots, c_n\}$ and multiplication given by

$$\begin{aligned}
&xa = a \\
&xb = a \\
&ac_i = \begin{cases} b & \text{for } c_i \in M \\ a & \text{for } c_i \notin M \end{cases} \\
&bc_i = \begin{cases} b & \text{for } c_i \in N \\ a & \text{for } c_i \notin N \end{cases} \\
&c_i c_j = \begin{cases} b & \text{for } (i,j) \in R \\ a & \text{for } (i,j) \notin R \end{cases}
\end{aligned}$$

1.6. Theorem. *Let (A, x') be the monounary algebra with the underlying set $A = \{a, b, c_1, \ldots, c_n\}$ and the operation $a' = b' = a$, $c_i' = b$. The collection of groupoids $E_{M,N,R}$ is just the collection of all LD-extensions of (A, x').*

Proof. The groupoids $E_{M,N,R}$ are left distributive, since $x \cdot yz = a$ for all x, y, z; clearly, they are LD-extensions of (A, x'). Conversely, let G be an LD-extension of (A, x').

By Lemma 1.1 we have $xa = a$ for all x, and $c_i b = a$.

Suppose $ab = b$. If also $bc_i = b$ for some i, then $a = bb = b \cdot bc_i = bb \cdot bc_i = ab = b$, a contradiction. Since $b \cdot c_i c_i = a$, we cannot have $bc_i = c_j$. Hence $bc_i = a$ for all i. Now $a = c_i a = c_i \cdot bc_i = c_i b \cdot c_i c_i = ab = b$, a contradiction.

We have proved $ab \neq b$. If $ab = c_i$, then $a \cdot bb = c_i c_i$, i.e., $aa = b$, a contradiction. Hence $ab = a$. In total, $xb = a$ for all x.

Since $xc_i \cdot xc_i = x \cdot c_i c_i = xb = a$, we get $xc_i \in \{a, b\}$ for all x. Now it is clear that $G = E_{M,N,R}$ for some M, N, R. □

1.7. Corollary. *There are precisely $2^{n(n+1)}$ LD-extensions of the monounary algebra (A, x') from 1.6. The number of isomorphism types of the LD-extensions is the same as the number of isomorphism types of n-element relation systems with two unary relations and one binary reflexive relation.* □

2. Finite zeropotent left distributive groupoids

A groupoid is called zeropotent if it satisfies $xx \cdot y = y \cdot xx = xx$. Equivalently, a groupoid G is zeropotent if it contains a zero element 0 and $xx = 0$ for all $x \in G$.

2.1. Theorem. *A finite zeropotent groupoid (with zero 0) is left distributive if and only if it satisfies $x \cdot yz = 0$.*

Proof. Let G be a finite zeropotent groupoid with zero 0. If G satisfies $x \cdot yz = 0$, then $x \cdot yz = 0 = xy \cdot xz$ and G is left distributive. Conversely, let G be left distributive. By a bad triple we shall mean a triple a, b, c of elements of G with $a \cdot bc \neq 0$; we need to prove that there is no bad triple.

Since $ab \cdot ac = a \cdot bc$, we get: if a, b, c is a bad triple, then also ab, a, c is a bad triple.

For any pair a, b of elements of G, define an infinite sequence $P_i^{a,b}$ $(i = 0, 1, \cdots)$ of elements of G as follows:

$$P_0^{a,b} = a, \qquad P_1^{a,b} = ab, \qquad P_i^{a,b} = P_{i-1}^{a,b} P_{i-2}^{a,b} \text{ for } i \geq 2.$$

It is easy to see that

$$P_i^{ab,a} = P_{i+1}^{a,b}$$

for any $a, b \in G$ and any $i \geq 0$.

Let us prove by induction on i that $aP_i^{a,b} = 0$. We have $aP_0^{a,b} = aa = 0$ and $aP_1^{a,b} = a \cdot ab = aa \cdot ab = 0 \cdot ab = 0$. For $i \geq 2$, $aP_i^{a,b} = a \cdot P_{i-1}^{a,b} P_{i-2}^{a,b} = aP_{i-1}^{a,b} \cdot aP_{i-2}^{a,b} = 00 = 0$.

Let us prove by induction on $i \geq 1$ that $P_i^{a,b} \cdot P_{i-1}^{a,b} c = a \cdot bc$ for any elements $a, b, c \in G$. For $i = 1$ this is just the left distributive law. If $i \geq 2$, then we can use the induction hypothesis:

$$P_i^{a,b} \cdot P_{i-1}^{a,b} c = P_{i-1}^{a,b} P_{i-2}^{a,b} \cdot P_{i-1}^{a,b} c = P_{i-1}^{a,b} \cdot P_{i-2}^{a,b} c = a \cdot bc.$$

Let us prove by induction on $n \geq 0$ that if a, b, c is a bad triple, then the elements $P_0^{a,b}, \cdots P_n^{a,b}$ are pairwise different. For $n = 0$ there is nothing to prove. Let $n \geq 1$. By the induction hypothesis applied to a, b, c, the elements $P_0^{a,b}, \cdots, P_{n-1}^{a,b}$ are pairwise different. By the induction hypothesis applied to the triple ab, a, c (which is also bad), the elements $P_0^{ab,a}, \cdots, P_{n-1}^{ab,a}$ are also pairwise different. Since

$P_i^{ab,a} = P_{i+1}^{a,b}$, the elements $P_1^{a,b}, \cdots, P_n^{a,b}$ are pairwise different. So, it remains to prove $P_0^{a,b} \neq P_n^{a,b}$. Suppose, on the contrary, that $a = P_n^{a,b}$. Then $a \cdot bc = P_n^{a,b} \cdot P_{n-1}^{a,b} c = a \cdot P_{n-1}^{a,b} c = aP_{n-1}^{a,b} \cdot ac = 0 \cdot ac = 0$, a contradiction.

So, we have proved that if there is a bad triple a, b, c, then all members of the infinite sequence $P_i^{a,b}$ $(i = 0, 1, \cdots)$ are pairwise different, a contradiction with the finiteness of G. □

2.2. Corollary. *The number of zeropotent left distributive groupoids with a given underlying set of n elements and a given zero element 0 is given by*

$$\sum_{i=0}^{n-2} \binom{n-1}{i} C_{i,i}$$

where

$$C_{k,0} = 1,$$

$$C_{k,m} = (m+1)^{(n-k-1)(n-2)} - \sum_{i=0}^{m-1} \binom{m}{i} C_{k,i} \quad \textit{for } 0 < m \leq k.$$

Proof. It is easy. Let G be a set of n elements with a fixed element $0 \in G$. For a given k-element subset K of $G - \{0\}$ and a given m-element subset M of K, $C_{k,m}$ is the number of the groupoids such that $K = \{xy : x, y \in G\} - \{0\}$ and $xa = 0$ for all $x \in G$, $a \in K$. □

2.3. Example. *For $n = 2, 3, 4, 5, 6$ the numbers are, respectively,* 1*,* 3*,* 52*,* 5681*, and* 6026496*.*

Although it seems probable that the assumption of finiteness in Theorem 2.1 cannot be eliminated, the author has not been able to find the corresponding example. The question remains open:

2.4. Conjecture. *There are infinite zeropotent left distributive groupoids not satisfying $x \cdot yz = 0$.*

This is equivalent to the following:

2.5. Conjecture. *The term $x \cdot yz$ is not equivalent to any term containing a subterm tt, for any term t, with respect to the equational theory of left distributive groupoids.*

It is easy to see that the two conjectures are equivalent. The negation of 2.4, together with Theorem 2.1, would be equivalent to saying that $x \cdot yz = uu$ is a consequence of the left distributive law together with $xx \cdot y = y \cdot xx = xx$. If $x \cdot yz = uu$ is a consequence, then there exists a formal proof of the equation, a finite sequence of terms $w_0, \cdots, w_n$ such that $w_0 = x \cdot yz$, $w_n = uu$ and, for any $i > 0$, the equation $w_{i-1} = w_i$ is an immediate consequence of either $x \cdot yz = xy \cdot xz$ or $xx \cdot y = xx$ or $y \cdot xx = xx$; if n is the last index such that $w_{i-1} = w_i$ is an immediate consequence of the distributive law, then clearly w_n contains a subterm tt for some term t, and $x \cdot yz = w_n$ is a consequence of the left distributive law. The converse implication is clear.

3. The enumeration algorithm

The following fragment of a C program can be used to find the number of LD-extensions of a given monounary algebra on n elements. We need two arrays `A` and `B` of the size n^2. `A` holds the multiplication table, which is generated in the lexicographic order. The array `B` holds information on the current state of `A`; if `B[`i`]`$= j < i$, then the value of `A[`i`]` was forced by `A[`j`]` and should be kept unchanged until `A[`i`]` is changed; this makes it possible to skip over intervals in the lexicographic order. In particular, `B[`i`]`$= -1$ means that `A[`i`]` should never be changed.

The function `Verify()` deals with the table `A` which may also contain the number -1, meaning that the corresponding place is undefined and `A` is the table of a partial groupoid. The function returns

-1	if the partial groupoid was found contradictory,
0	if no completion was done,
1	if at least one completion was done.

Here is the fragment (for $n \leq 26$):

```
#include <stdio.h>
int n,n1,nn,nn1,P; long int N=0L,NI=0L; int A[676]; int B[676];

int Verify(I){ int i,j,k,a,b,c,d,e,p,q,Z=0;
for(i=0;i<n;i++)for(j=0;j<n;j++){
  c=A[n*i+j];if(c>=0)for(k=0;k<n;k++){
    d=A[n*i+k];a=A[n*j+k];if(a>=0&&d>=0){
      p=n*i+a;q=n*c+d;b=A[p];e=A[q];
      if(b>=0&&e<0){A[q]=b;B[q]=I;Z=1;}
      else if(b<0&&e>=0){A[p]=e;B[p]=I;Z=1;}
      else if(b>=0&&b!=e)return -1;}}}
return Z;}

int FindNext(I){
while(I>=0&&(A[I]==n1||B[I]<I))I--;
if(I>=0)A[I]++; return I;}

void MakePartial(I){ int i;
for(i=I+1;i<nn;i++)if(B[i]>=I){A[i]=-1;B[i]=nn;P++;}}

void MakeComplete(I){ int i;
P=0;for(i=I+1;i<nn;i++)if(A[i]==-1)A[i]=0;}

void Process(){ N++;}

void main(){ int i,I,V,a,b,c; char s[80];]
printf("\nInput cardinality of groupoid:  "); scanf("%d",&n);
n1=n-1;nn=n*n;nn1=nn-1;
for(i=0;i<nn;i++){A[i]=0;B[i]=nn;}
do{scanf("%s",s);
  if(strlen(s)==3){a=s[0]-'a';b=s[1]-'a';c=s[2]-'a';
```

```
    A[n*a+b]=c;B[n*a+b]=-1;}}
  while(strlen(s)==3);
  I=nn1;P=0;
  while(I>=0){
    do V=Verify(I); while(V==1&&P);
    if(!V&&P){MakeComplete(I);I=nn1;V=Verify(I);}
    if(!V)Process();
    I=FindNext(I);MakePartial(I);}
  printf("N=%ld\n",N);}
```

Of course, one may ask for more than simply counting the number of multiplication tables. The changes should be done in the function `Process()`, which is called each time when a new valid table was found. For example, one may be interested in finding the number of isomorphism types. It is not necessary to store all the preceding tables in order to check if the current table is isomorphic to one of them. Since the tables are found in the lexicographic order, it is sufficient to ask if the new table can be isomorphic to one which came in the lexicographic order earlier, and this can be done by checking all permutations of the underlying aset. There are no requirements on either space or memory in the program. To make the program more user friendly, changes in the function `main()` should be done.

One is often interested in enumerating not all tables, but only those that are extensions of a given partial groupoid. For that purpose one can append

```
A[n*i+j]=k; B[n*i+j]=-1;
```

to the fifth line of the function `main()`, for each item $ij = k$ of the given partial groupoid.

4. Six-element groupoids

The numbers of all left distributive groupoids and of all isomorphism types of left distributive groupoids on a given set of two, three, four, five and six elements are given in the following table:

Elements:	2	3	4	5	6
Groupoids:	9	224	14067	3717524	
Iso types:	6	48	720	33425	35527485

For six elements, we did not count the number of groupoids. Instead, the isomorphism types were divided into 130 groups according to the isomorphism types of their diagonals. There are 130 isomorphism types of monounary algebras on six elements. For each of them, the number of isomorphism types of LD-extensions was computed using the algorithm described in Section 3.

Let us denote the six elements by a, b, c, d, e, f. For a 6-tuple $a_1, \ldots, a_6$, denote by $N(a_1, \ldots, a_6)$ the ordered pair (n, m) where n is the number of LD-extensions of the monounary algebra on a, b, c, d, e, f with $a' = a_1, b' = a_2, \ldots, f' = a_6$, and m is the number of their isomorphism types. Among the 130 cases, there were nineteen with $n \geq 10^5$:

$N(baaabb) = (122263, 15426)$ $\quad$ $N(aaaaaa) = (160006292, 1342744)$
$N(acaaaa) = (47321604, 7902069)$ $\quad$ $N(addaaa) = (30826684, 7721940)$

$N(acaeaa) = (15399116, 7701169)$
$N(addafa) = (14348907, 7184295)$
$N(abbbbb) = (1403331, 61166)$
$N(abeebb) = (226518, 114520)$
$N(aacccc) = (178839, 31039)$
$N(abcddd) = (294766, 31215)$
$N(abcdfe) = (121736, 5891)$
$N(abcdef) = (17711155, 32541)$
$N(aeeeaa) = (15405702, 2581806)$
$N(affffa) = (1048576, 45960)$
$N(abdbbb) = (384558, 193263)$
$N(abdbfb) = (138240, 69489)$
$N(abcccc) = (250078, 24965)$
$N(abccee) = (110569, 29150)$
$N(abcdee) = (1049690, 51254)$

References

1. R. Bashir, A. Drápal, *Quasitrivial left distributive groupoids*, Commentationes Math. Univ. Carolinae **35** (1994), 597–606.
2. P. Dehornoy, *The adjoint representations of left distributive structures*, Comm. Algebra **20** (1992), 1201–1215.
3. P. Dehornoy, *A normal form for the free left distributive law*, preprint.
4. P. Dehornoy, *On the syntactic algorithm for the word problem of left distributivity*, preprint.
5. P. Dehornoy, *Construction of left distributive operations*, preprint.
6. A. Drápal, *Finite left distributive algebras with one generator*, preprint.
7. A. Drápal, *Finite left distributive groupoids with one generator*, preprint.
8. A. Drápal, T. Kepka, M. Musílek, *Group conjugation has non-trivial LD-identities*, Commentationes Math. Univ. Carolinae **35** (1994), 219–222.
9. D. M. Larue, *Left-distributive idempotent algebras*, preprint.
10. R. Laver, *The left distributive law and the freeness of an algebra of elementary embeddings*, Advances in Math. **91** (1992), 209–231.

MFF UK, Sokolovská 83, 18600 Praha 8