FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

MASTER THESIS

Viliam Valent

Small order quasigroups with minimum number of associative triples

Department of Algebra

Supervisor of the master thesis:	prof. RNDr. Aleš Drápal, CSc., DSc.
Study programme:	Mathematics
Study branch:	Mathematics for Information Technologies

Prague 2018

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In .... date .... signature of the author

Title: Small order quasigroups with minimum number of associative triples

Author: Viliam Valent

Department: Department of Algebra

Supervisor: prof. RNDr. Aleš Drápal, CSc., DSc., Department of Algebra

Abstract: This thesis is concerned with quasigroups with a small number of associative triples. The minimum number of associative triples among quasigroups of orders up to seven has already been determined. The goal of this thesis is to determine the minimum for orders eight and nine. This thesis reports that the minimum number of associative triples among quasigroups of order eight is sixteen and among quasigroups of order nine is nine. The latter finding is rather significant and we present a construction of an infinite series of quasigroups with the number of associative triples equal to their order. Findings of this thesis have been a result of a computer search which used improved algorithm presented in this thesis. The first part of the thesis is devoted to the theory that shows how to reduce the search space. The second part deals with the development of the algorithm and the last part analyzes the findings and shows a comparison of the new algorithm to the previous work. It shows that new search program is up to four orders of magnitude faster than the one used to determine the minimum number of associative triples among quasigroups of order seven.

Keywords: quasigroups, latin squares, associativity index

I would like to thank my supervisor prof. RNDr. Aleš Drápal, CSc., DSc. for regular and very helpful consultations, his patience and all his advice that helped me to stay on track during the whole process of writing this thesis.

Access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum provided under the programme "Projects of Large Research, Development, and Innovations Infrastructures" (CESNET LM2015042), is greatly appreciated.

Contents

Introduction

Quasigroups are algebraic structures similar to groups. They differ from groups mainly in that they do not have to be associative. This thesis deals with quasigroups of small orders with a small number of associative triples. Associativity index of quasigroups has already been studied by Drápal [1], Ježek and Kepka [2], Kotzig [3], and Grošek and Horák [4]. This subject has since become the focus of our attention as well [5, 6].

The motivation for finding quasigroups with a small number of associative triples comes from cryptography where those quasigroups can be used hash functions as mentioned in [7, 8, 9].

The goal of this thesis is to determine the minimum number of associative triples among quasigroups of orders eight and nine. This goal seems to be on the limit of current computational powers.

The structure of the thesis is as follows. In the first chapter, we focus on permutations and transformations of finite sets. We enumerate types of transformations on 8 and 9 element sets based on the number of fixed points and their defect. This provides the basis for case analysis in the search for quasigroups of orders eight and nine with the minimum number of associative triples. In Chapter 2 we define basic terminology for dealing with quasigroups and focus on local unit mappings in quasigroups. Chapter 3 shows the distinction between two types of associative triples. One type, elementary associative triples, are fully determined by local unit mappings and can be therefore counted in partially constructed quasigroups. The estimate on a number of elementary associative triples presented in Chapter 3 can be seen as an improvement over the estimate found in [4]. In the same chapter, we also devise a strategy for conquering the search for quasigroups of orders 8 and 9 with a maximum of 16 and 18 associative triples respectively. The strategy consists of dividing the problem into subproblems based on the number of idempotents and using the results of Chapter 1 to reduce the search space.

Chapter 4 focuses on nonelementary associative triples and a new approach for counting them at the time of their creation. Combination of theoretical results from all previous chapters can be found in Chapter 5 where we outline an algorithm for counting associative triples based on their roles in the quasigroup. The main results of this thesis were achieved by implementing and optimizing this algorithm in order to discover the minimum number of associative triples among quasigroups of small orders. Results of our work are presented in Chapter 6 together with the classification of found quasigroups and an analysis of their automorphism groups. In this chapter, we also present several ideas for the direction of future research. One of the most notable results is the construction of infinite series of quasigroups with the number of associative triples equal to their order.

In Chapter 7 we share insights into implementation aspects that significantly improved the performance of our search program. We hope that those implementation improvements might help to tackle higher orders in future. Lastly, in Chapter 8 we measure various speed improvements presented in this thesis and compare the results with our previous work in [5].

1. Permutations and transformations

In this chapter we are going to introduce necessary terms for working with permutations and transformations of finite sets. This will allow us to enumerate them based on their defect and number of fixed points. The results of these enumerations will be used later in the thesis to reduce the search space when searching for quasigroups with a small number of associative triples.

1.1 Permutations

A permutation α of a set X is a bijection from X to itself. All permutations of set X form a group where group operation is a composition of permutations. This group is known as a *symmetric group of* X and is denoted by S_X. If $X = \{1, \ldots, n\}$, then S_X is also written as S_n.

Any permutation α can be expressed as a composition of disjoint cycles. A cycle $(a_1\ a_2\ \ldots\ a_n)$ of length n describes a permutation such that a_i gets mapped to a_{i+1}, $1 \leq i < n$, and a_n gets mapped to a_1. Two cycles are *disjoint* if they do not have any element in common. A cycle of size two is known as a *transposition*.

Elements of the symmetric group on a set X are divided into *conjugacy classes*. Two permutations $\alpha, \beta \in S_X$ are *conjugate* if and only if there exists $\varphi \in S_X$ such that $\beta = \varphi \circ \alpha \circ \varphi^{-1}$. This is true if and only if they consist of the same number of disjoint cycles of the same lengths. For instance, in S_6, permutations $(1\ 2\ 3)(4\ 6)$ and $(1\ 4\ 3)(2\ 5)$ are conjugate. Two permutations in the same conjugacy class are said to have the same *type*.

We will denote the type of a permutation $\alpha \in S_n$ by an m-tuple $(t_1, \ldots, t_m)$ where $t_1 \geq t_2 \geq \ldots \geq t_m \geq 1$ are lengths of cycles in α and $\sum_{i=1}^{m} t_i = n$.

1.2 Transformations

A transformation f of a set X is a mapping from X to itself. A set of all transformations of X closed under composition form a *transformation monoid* which is denoted by T_X. If $X = \{1, \ldots, n\}$, then T_X is also written as T_n.

We are going to extend the notion of the *type* to T_X.

Definition. Transformations $f, g : X \to X$ are of the same *type* if there exists $\varphi \in S_X$ such that $g = \varphi \circ f \circ \varphi^{-1}$.

Definition. The *defect* of a transformation $f : X \to X$ is defined as $|X| - |\operatorname{Im}(f)|$.

Definition. A subset of X defined as $C(f) = \bigcap_{i \geq 1} \operatorname{Im}(f^i)$ is called the *cyclic part* of f. Its complement $N(f) = X \setminus C(f)$ is called the *non-cyclic part* of f.

Note that f restricted upon $C(f)$ is a permutation.

Definition. Let f be a transformation. For every $x \in C(f)$ define a directed tree graph with set of vertices $V = \{y \in N(f);\ \exists i \in \mathbb{N} : f^i(y)=x \text{ and } \forall j < i : f^j(y) \in N(f)\}$ and edges $E = \{(a,b) \in V^2; f(a) = b\}$ called a *tree of x*. It is denoted as $\mathrm{Tr}(x)$.

Lemma 1.1. *Type of transformation $f : X \to X$ with defect 1 is uniquely determined by $(\boldsymbol{t}, l, k)$, where $\boldsymbol{t}$ is the type of the permutation α_f obtained by restricting f to $C(f)$, l is the smallest integer for which $f^l(s) \in C(f)$, where s is the only element in $X \setminus \mathrm{Im}(f)$, and k is the length of the cycle of α_f containing element $f^l(s)$.*

Proof. Let $f, g : X \to X$ be transformations described by the same triple $(\mathbf{t}, l, k)$. This means that $|C(f)| = |C(g)|$ and also that $\alpha_f = f|_{C(f)}$ and $\beta_g = g|_{C(g)}$ have the same type. Let s_f, s_g be the only elements in $X \setminus \mathrm{Im}(f)$ and $X \setminus \mathrm{Im}(g)$ respectively. Define $\varphi : X \to X$ as a bijection between $N(f)$ and $N(g)$ for which $\varphi(s_f)=s_g$ and $\varphi(f^i(s_f))=g^i(s_g), 1 \le i \le l$. Extend φ to $C(f)$ according to the following formula. Fix $a = f^l(s_f) \in C(f)$. Note that $\varphi(a)$ is already defined as $g^l(s_g)$ denoted by b. Define φ on $C(f)$ inductively by $f(x) = y \Rightarrow \varphi(y) = g(\varphi(x)) = g(\varphi(f^{-1}(y)))$. In order to start the induction there has to be chosen first element e from each cycle of $C(f)$ and element d from each cycle of $C(g)$ for which $\varphi(e) = d$. This has been done for the cycles containing a and b. For other cycles C from $C(f)$ pick $e \in C$ arbitrarily and pick any $d \in D$ where D is an unused cycle from $C(g)$ of the same length as C. See that φ is also a bijection on $C(f)$. As we shall observe $f = \varphi^{-1} \circ g \circ \varphi$. For every $x \in N(f), x = f^i(s_f), i \le l$ this holds because:

$$\begin{aligned}
\varphi^{-1}(g(\varphi(f^i(s_f)))) &= \varphi^{-1}(g(g^i(s_g))) \\
&= \varphi^{-1}(g^{i+1}(s_g)) \\
&= f^{i+1}(s_f) \\
&= f(f^i(s_f)) \\
&= f(x).
\end{aligned}$$

For $y \in C(f)$ this holds from the definition of φ. The opposite direction of the proof is easy. Denote by $(\mathbf{t}_f, l_f, k_f)$ and $(\mathbf{t}_g, l_g, k_g)$ triples determining types of f and g. Let φ be a permutation for which $g = \varphi \circ f \circ \varphi^{-1}$. Restriction of φ on $C(f)$ gives $\mathbf{t}_f = \mathbf{t}_g$ from the conjugation of permutations, $l_f = l_g$ from the sizes of $N(f), N(g)$ and defect 1. Lastly, $k_f = k_g$ because the opposite would imply that cycles of different lengths are conjugate. □

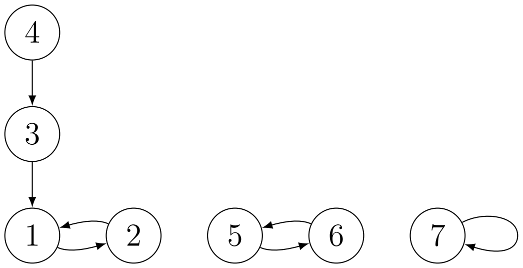


Figure 1.1: Transformation f described by $(\mathbf{t}{=}(2,2,1), l{=}2, k{=}2)$.

Definition. A *partition* of a positive integer n is a way of writing n as a sum of positive integers. Each partition is thus associated with an expression $n = \sum_{i=1}^{r} k_i c_i$, where $c_1 > c_2 > \ldots > c_r \geq 1$ and $k_i \geq 1$. A *partition number* $p(n)$ is equal to the number of such expressions. By convention, $p(0) = 1$.

Denote by $q(n)$ the number of partitions of n that do not include integer 1. This is equal to the number of those sums for which $c_r \geq 2$. Note that $p(n) = \sum_{m=0}^{n} q(m)$. Note also that $p(n)$ is equal to the number of conjugacy classes in S_n and $q(n)$ is the number of those classes that consist of fixed point free permutations.

The first few partition numbers are shown in the table below.

n	0	1	2	3	4	5	6	7	8	9	10
$p(n)$	1	1	2	3	5	7	11	15	22	30	42
$q(n)$	1	0	1	1	2	2	4	4	7	8	12

Table 1.1: Values of $p(n)$ and $q(n)$ for n=0, . . . , 10.

Definition. A transformation $f : X \to X$ is called *near-permutation* if $|X| - |C(f)| = 1$.

Observe that near-permutations are the transformations with defect 1 and l=1. It is easy to see that number of types of near-permutations on n-element set with no fixed point, denoted by $t_0(n)$, is obtained by counting with weight r each partition of $n-1$ associated with the sum $n-1 = \sum_{i=1}^{r} k_i c_i, c_1 \geq c_2 \geq \ldots \geq c_r \geq 2, k_i \geq 1$.

Denote by $t_1(n)$ the number of types of transformations on n elements with no fixed point and defect one. It is clear that $t_1(n) = \sum_{m \leq n} t_0(m)$.

Lastly, denote by $t_2(n)$ the number of types of transformations from T_n with one fixed point and defect one.

Lemma 1.2. *For $n \geq 1, t_2(n) = t_1(n-1) + p(n-1) - q(n-1)$. This is equal to $t_1(n-1) + p(n-2)$ for $n \geq 2$. If $h \geq 1$, then $t_2(n)$ also gives the number of types of transformations from T_{n+h-1} with exactly h fixed points and defect one.*

Proof. Divide transformations $f \in T_n$ with defect one and one fixed point into two sets, according to the value of k from their determining triple $(\mathbf{t}, l, k)$. Those for which k>1 are in one-to-one relations with transformations on $n-1$ elements with defect one and without any fixed point. They amount to $t_1(n-1)$ types.

The other set of transformations for which k=1 contains transformations with one fixed point z having a tree $\mathrm{Tr}(z)$ of height at least one. Defect one implies that $\mathrm{Tr}(z)$ has only one leaf. Associate type of permutation α on $n-1$ elements with at least one fixed point with the type of transformation f from this set. The number of fixed points of α being equal to l, and cycles of length at least 2 yielding the type of $C(f) \setminus \{z\}$. This means that f consists of cycles from α, fixed point z and $\mathrm{Tr}(z)$ of height l.

The last part of the proof is a simple observation. □

Table 1.2 shows computed values of $t_0(n), t_1(n)$, and $t_2(n)$.

n	1	2	3	4	5	6	7	8	9	10
$t_0(n)$	0	0	1	1	2	3	5	7	11	15
$t_1(n)$	0	0	1	2	4	7	12	19	30	45
$t_2(n)$	0	1	1	3	5	9	14	23	34	52

Table 1.2: Values of $t_0(n), t_1(n)$, and $t_2(n)$ for $n = 1, \ldots, 10$.

Lemma 1.3. *For transformations $f \in T_8$ that fix $i \leq 8$ points and have the defect at most one denote by η_i the number of their types. Use ν_i to count different types of transformations $g \in T_9$ that fix $i \leq 9$ points and have the defect at most one. Then values of η_i and ν_i are as in the Table 1.3:*

n	0	1	2	3	4	5	6	7	8	9
η_i	26	27	18	11	7	4	2	1	1	
ν_i	38	41	27	18	11	7	4	2	1	1

Table 1.3: Values of η_i and ν_i for $i = 1, \ldots, 9$.

Proof. In the general case, $\eta_i = q(8{-}i) + t_2(8{-}i{+}1), 1 \leq i \leq 7$ because there are $q(8-i)$ types of those transformations with no defect and the number of types of transformations with i fixed points and defect one on n elements is equal to $t_w(8{-}i{+}1)$. Observe that $\eta_8 = 1$ because the identity is the only choice. Lastly, $\eta_0 = q(8){+}t_1(8)$. A similar argument also gives ν_i. □

Lemma 1.4. *For $i \in \{0, 1, 2, 3, 4\}$ denote by μ_i the number of types of transformations $f \in T_n$ that fix exactly $n{-}i$ points, $n \geq 2i$. Then $\mu_0 = 1$, $\mu_1 = 1$, $\mu_2 = 4$, $\mu_3 = 10$, and $\mu_4 = 30$.*

Proof. The case for $i = 0$ is clear. For $i = 1$ there is only one type transformation possible and is described by $(\mathtt{t}{=}(1, \ldots, 1), l{=}1, k{=}1)$. In case that $i = 2$ there four types in total. One is a permutation of type $(2, 1, \ldots, 1)$. Those other three cases are depicted in Figure 1.2. For $i = 3$ write μ_3 as $d_0 + d_1 + d_2 + d_3$ where d_j is the number of types of f that have defect j. The defect 0 means that f is a permutation and there is only one type: $(3, 1, \ldots, 1)$. By enumeration all types of f with defect 1, illustrated in Figure 1.3, we get $d_1 = 3$. Next, $d_2 = 3$, as shown in Figure 1.4 and $d_3 = 3$ as shown in Figure 1.5. Finally, similar rewriting of μ_4 as a sum based on the defect yields $\mu_4 = 2 + 5 + 11 + 7 + 5 = 30$. □

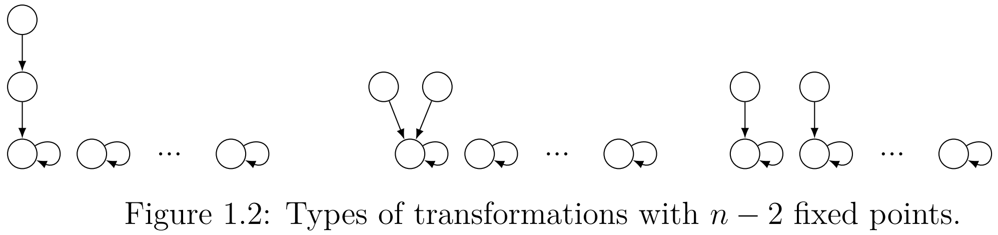


Figure 1.2: Types of transformations with $n - 2$ fixed points.

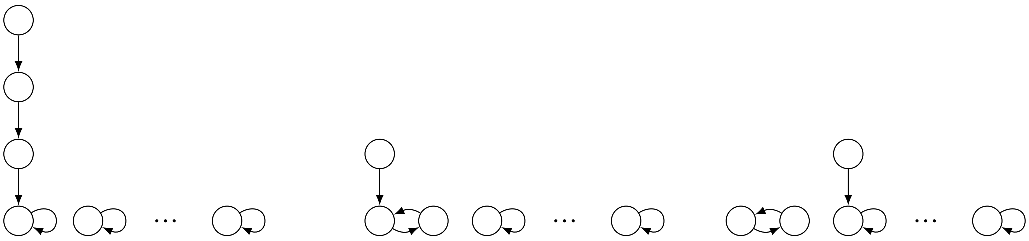


Figure 1.3: Types of transformations with $n-3$ fixed points and defect 1.

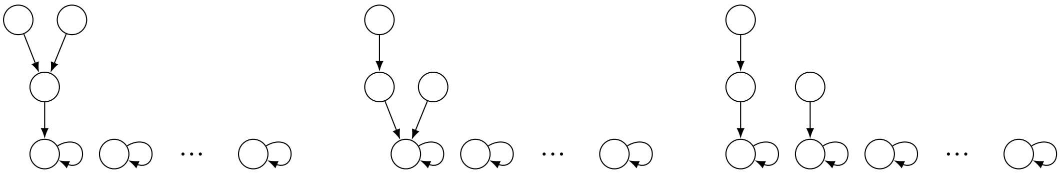


Figure 1.4: Types of transformations with $n-3$ fixed points and defect 2.

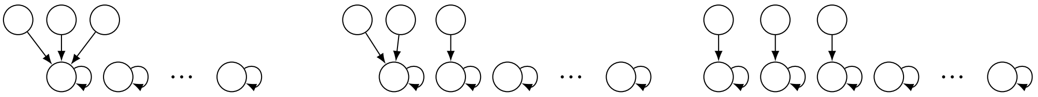


Figure 1.5: Types of transformations with $n-3$ fixed points and defect 3.

i	1	2	3	4	5	6	7	8
$f_1(i)$	1	2	3	4	6	5	1	2
$f_2(i)$	1	2	3	4	6	5	1	5
$f_3(i)$	1	2	3	4	6	5	5	6
$f_4(i)$	1	2	3	4	1	2	5	6
$f_5(i)$	1	2	3	4	1	2	5	7
$f_6(i)$	1	2	3	4	1	5	6	6
$f_7(i)$	1	2	3	4	1	5	5	6

Table 1.4: The transformations on 8 element set with four fixed points and defect two.

Lemma 1.5. *Let $f \in T_8$ satisfying $|f^{-1}(i)| \leq 2, i \in \{1,\dots,8\}$ have four fixed points and defect two. Then f has the same type as one of the following transformations f_j in Table 1.4.*

Proof. The enumeration of all possible types of those transformations is straightforward and can be divided into two cases based on the size of the cyclic part of f. Transformations f_1, f_2 and f_3 correspond to the types with $|C(f)|$=6 and have two trees of height one. The others have $|C(f)| = 4$ where f_4 and f_5 have two trees and f_6 and f_7 have only one tree. This enumerates all possible options. □

Lemma 1.6. *Let $g \in T_9$ satisfying $|g^{-1}(i)| \leq 2, i \in \{1,\dots,9\}$ have four fixed points and defect two. Then g has the same type as one of the following transformations g_i in Table 1.5.*

Proof. Transformations g_1, g_2 and g_3 correspond to the types with $|C(f)| = 7$ and have two trees of height one. Transformations $g_4,\dots,g_9$ have $|C(f)| = 6$ and $g_{10},\dots,g_{15}$ have $|C(f)| = 4$. This enumerates all possible options. □

i	1	2	3	4	5	6	7	8	9
$g_1(i)$	1	2	3	4	6	7	5	1	2
$g_2(i)$	1	2	3	4	6	7	5	1	5
$g_3(i)$	1	2	3	4	6	7	5	5	5
$g_4(i)$	1	2	3	4	6	5	1	2	7
$g_5(i)$	1	2	3	4	6	5	1	7	7
$g_6(i)$	1	2	3	4	6	5	1	5	7
$g_7(i)$	1	2	3	4	6	5	1	5	8
$g_8(i)$	1	2	3	4	6	5	5	7	7
$g_9(i)$	1	2	3	4	6	5	5	6	7
$g_{10}(i)$	1	2	3	4	1	2	5	6	7
$g_{11}(i)$	1	2	3	4	1	2	5	7	8
$g_{12}(i)$	1	2	3	4	1	5	5	6	7
$g_{13}(i)$	1	2	3	4	1	5	5	6	8
$g_{14}(i)$	1	2	3	4	1	5	6	6	7
$g_{15}(i)$	1	2	3	4	1	5	6	7	7

Table 1.5: The transformations on 9 element set with four fixed points and defect two.

Lemma 1.7. *Let $h \in T_9$ satisfying $|h^{-1}(i)| \leq 2, i \in \{1, \dots, 9\}$ have five fixed points and defect two. Then h has the same type as one of the following transformations h_i in Table 1.6.*

i	1	2	3	4	5	6	7	8	9
$h_1(i)$	1	2	3	4	5	1	2	6	7
$h_2(i)$	1	2	3	4	5	1	2	6	8
$h_3(i)$	1	2	3	4	5	1	2	9	8
$h_4(i)$	1	2	3	4	5	1	6	6	7
$h_5(i)$	1	2	3	4	5	1	6	7	7
$h_6(i)$	1	2	3	4	5	1	8	7	7
$h_7(i)$	1	2	3	4	5	7	6	6	7

Table 1.6: The transformations on 9 element set with five fixed points and defect two.

Proof. Follows directly from Lemma 1.5 □

This concludes this chapter about permutations and transformations. Results of this section will be used to enumerate cases that cover the search of highly nonassociative quasigroups of orders 8 and 9. Those cases are connected to the types of local unit mappings in quasigroups. This will allow us to speed up the search by several orders of magnitude. Note, for instance, the difference between the number of all permutations on 9 elements, which is equal to 362880, and only 30 types of those permutations.

2. Properties of quasigroups

This chapter deals with properties of quasigroups with focus on associative triples. It shows the link between types of mappings in quasigroups and isomorphism classes that can be used in the search for nonassociative quasigroups.

Definition. A *quasigroup* $(Q, \cdot)$ is a set Q with a binary operation $\cdot$ closed on Q such that the equations $a \cdot x = b$ and $y \cdot a = b$ have unique solutions for every $a, b \in Q$.

Throughout this thesis, we will write xy instead of $x \cdot y$. Also, it is assumed that the quasigroup operation is $\cdot$ unless stated otherwise. The *order* of Q (i.e. the number of elements of Q) will be denoted by $|Q|$. Unless stated otherwise, Q will mean a finite quasigroup of order n.

Definition. A *loop* is a quasigroup with an *unit*; that is, an element i such that $xi = x$ and $ix = x$ for every $x \in Q$.

Definition. A *Latin square* is an $n \times n$ array filled with n different symbols, each occurring exactly once in each column and exactly once in each row.

Remark. Every multiplication table of a finite quasigroup is a Latin square. Conversely, every Latin square can be considered as the multiplication table of a quasigroup.

Definition. For a quasigroup Q a triple $(a, b, c) \in Q^3$ is called *associative* if $(ab)c = a(bc)$. Denote by $A(Q)$ the set of all associative triples in Q and put $\mathbf{a}(Q) = |A(Q)|$. This number is also called the *associativity index* of Q. Also, we set $\mathbf{a}(n) = \min\{\mathbf{a}(Q); Q \text{ is a quasigroup of order } n\}$.

Definition. An element q of a quasigroup Q is called *idempotent* if $qq = q$. Denote by $I(Q)$ the set of all idempotents of Q and put $\mathbf{i}(Q) = |I(Q)|$. The quasigroup Q is called *idempotent* if every $q \in Q$ is an idempotent element.

Definition. Let Q be a quasigroup. For all $a \in Q$ the *left* and *right translations* are defined as follows:

$$L_a(x) = ax, \forall x \in Q; \text{ and}$$
$$R_a(y) = ya, \forall y \in Q.$$

The inverse mappings are left and right division, that is,

$$L_a^{-1}(x) = a\backslash x; \text{ and}$$
$$R_a^{-1}(y) = y/a.$$

In this notation the identities that describe quasigroup's multiplication and division operations are

$$x(x\backslash y) = y;$$
$$x\backslash(xy) = y;$$
$$(yx)/x = y; \text{ and}$$
$$(y/x)x = y.$$

Definition. Let $(Q,\cdot)$ and $(R,*)$ be two quasigroups of the same order. An ordered triple (α,β,γ) of bijections α,β,γ of the set Q onto the set R is called an *isotopy* or *isotopism* of $(Q,\cdot)$ upon $(R,*)$ if $\alpha(x)*\beta(y)=\gamma(x\cdot y)$ for all $x,y\in Q$. The quasigroups $(Q,\cdot)$ and $(R,*)$ are then said to be *isotopic*. An isotopy (α,α,α) of Q upon R is called an *isomorphism*. If $(Q,\cdot)=(R,*)$, then (α,β,γ) is called an *autotopy* or *autotopisms* of Q and (α,α,α) is called an *automorphism*.

Remark. The set of all autotopisms of a quasigroup Q forms a group.

Remark. Isomorphic quasigroups have the same number of associative triples.

Definition. Denote by $\mathrm{Aut}(Q)$ the group of all automorphisms of Q.

Definition. For a quasigroup $(Q,\cdot)$ and $\delta\in S_3$ define a *parastrophe* to the quasigroup Q as a quasigroup $(Q^\delta,*)$ on the same set Q where $x_1\cdot x_2=x_3\Leftrightarrow x_{\delta(1)}*x_{\delta(2)}=x_{\delta(3)}$. Operations $\cdot$ and $*$ are said to be *conjugates* or *parastrophic*.

A permutation (1 2)(3) creates parastrophe Q^{op} called an *opposite quasigroup* to Q.

Definition. Quasigroups $(Q,\cdot)$ and $(R,*)$ are said to be *paratopic* or *main class isotopic* if R is isotopic to a conjugate of Q. The set of quasigroups paratopic to Q is the *main class* of Q.

2.1 Local units

For $x\in Q$ there exist local units e_x and f_x such that $e_x x=x=xf_x$. In the rest of this thesis we will use e and f as mappings $Q\to Q$, with $e(x)=e_x$ and $f(x)=f_x$, for all $x\in Q$.

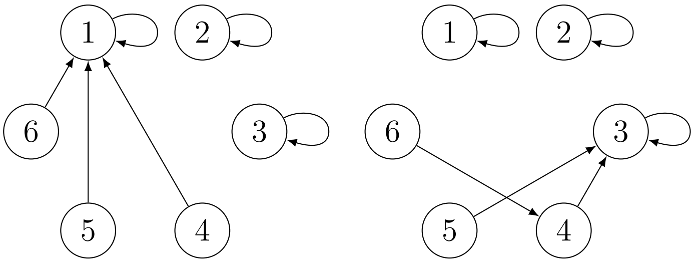
Remark. Mapping e in quasigroup Q becomes mapping f in opposite quasigroup Q^{op} and vice versa.

Remark. In the case of a loop, both mappings e and f send all elements of the loop to the identity element i. For idempotent quasigroup e and f are the identity permutations.

For the illustration purposes, we will be using directed graphs to represent the mappings e and f. A directed graph $G_e=(Q,E)$ represents the mapping e of a finite quasigroup Q if vertices of G_e are the elements of Q, and $(x,y)\in E\Leftrightarrow e(x)=y$. From the structure of the quasigroup some properties of G_e are as follows:

- $|E|=|Q|$,
- outdegree of each vertex is 1,
- idempotent elements of quasigroup will manifest as loops,
- if each vertex has indegree of 1, e is a permutation, and
- number of vertices with indegree 0 is equal to the defect of e.

	1	2	3	4	5	6
1	$\boxed{\mathbf{1}}$	3	2	**4**	**5**	**6**
2	3	$\boxed{\mathbf{2}}$	6	1	4	5
3	4	1	$\boxed{\mathbf{3}}$	5	6	2
4	5	6	$\boxed{4}$	2	1	3
5	6	4	$\boxed{5}$	3	2	1
6	2	5	1	$\boxed{6}$	3	4



(a) The mapping e

(b) The mapping f

Figure 2.1: A quasigroup of order 6 and its mappings $\boldsymbol{e}$ and $\boxed{f}$ highlighted and illustrated.

Observe that fixing mappings e and f allows us to reduce search space when looking for nonassociative quasigroups in the following way:

Lemma 2.1. *For quasigroups $(Q_1, \cdot)$ and $(Q_2, \times)$ denote by e^i, f^i, and $d^i : x \mapsto xx$ mappings in Q_i. If Q_1 and Q_2 are isomorphic, then e^1 and e^2, f^1 and f^2, and d^1 and d^2 are of the same type.*

Proof. Suppose that Q_1 and Q_2 are isomorphic under permutation φ. Mapping $e^1(x) = e^1_x$ such that $e^1_x \cdot x = x$. This is mapped by φ onto $\varphi(e^1_x) \times \varphi(x) = \varphi(x)$ therefore e^1_x is mapped onto $e^2_{\varphi(x)}$ for all $x \in Q_1$. Verify that $e^2 = \varphi e_1 \varphi^{-1}$. Put $e^2(a) = e^2_a$ and denote by $b = \varphi^{-1}(a)$. Observe that $e^1(b) = e^1_b$ and $\varphi(e^1_b) = e^2_{\varphi(b)} = e^2_a$ which concludes the proof. For mappings f and d the proof is analogous. □

This simple observation is equivalent with the statement that two quasigroups with mappings e (or f or d) of different types cannot be isomorphic. Practical implications are that when performing an exhaustive search we can fix one of the mappings from above, for example e, search through all quasigroups with that mapping e and later skip those quasigroups with mapping e of the same type because they are isomorphic to one of the quasigroups we already checked. This corresponds with filling out $|Q|$ cells into the multiplication table of the quasigroup.

In the next chapter, we will use findings from Chapter 1 to enumerate all the cases that cover the search of highly nonassociative quasigroups of orders 8 and 9.

3. Elementary and nonelementary associative triples

In this chapter, we show that associative triples in a quasigroup can be of several types. For some of those types, we can determine their number just from the knowledge of idempotents in the quasigroup and mappings e and f.

Definition. An associative triple $(x, y, z) \in Q^3$ where

- $x \cdot yz = xy = xy \cdot z$ is called *left elementary*;
- $x \cdot yz = yz = xy \cdot z$ is called *right elementary*;
- $x \cdot yz = xz = xy \cdot z$ is called *middle elementary.*

A triple for which at least on of the above holds is called *elementary.*

Lemma 3.1. *An elementary associative triple $T = (x, y, z) \in Q^3$ can be described by means of e and f:*

(a) T is left elementary $\Leftrightarrow xy, y \in f^{-1}(z)$;

(b) T is right elementary $\Leftrightarrow yz, y \in e^{-1}(x)$;

(c) T is middle elementary $\Leftrightarrow f(x) = y = e(z)$.

Proof. $y, xz \in f^{-1}(z) \Leftrightarrow f(y) = z$ and $f(xy) = z \Leftrightarrow yz = y$ and $xy \cdot z = xz \Leftrightarrow xy \cdot z = xy = x \cdot yz$. Proof of (b) is similar to (a). For (c) simply note that $f(x) = y \Leftrightarrow xy = x$ and $e(z) = y \Leftrightarrow yz = z$. Therefore $x \cdot yz = xz = xy \cdot z$. □

Lemma 3.2. *Let $T = (x, y, z) \in Q^3$. Then*

(i) T is left and right elementary if and only if $y \in e^{-1}(x) \cap f^{-1}(z)$;

(ii) T is left and middle elementary if and only if $f(x) = y = z \in I(Q)$;

(iii) T is right and middle elementary if and only if $e(z) = x = y \in I(Q)$; and

(iv) T is left, right, and middle elementary if and only if $x = y = z \in I(Q)$.

Proof. If (a) and (b) from Lemma 3.1 hold for T, then $y \in e^{-1}(x) \cap f^{-1}(z)$. If $y \in e^{-1}(x) \cap f^{-1}(z)$, then $xy = y = yz$ and T is both left and right elementary because $x \cdot yz = xy = y = yz = xy \cdot z$.
If both (a) and (c) hold, then $yz = y = z$ and therefore $f(x) = y = z \in I(Q)$. The other direction follows from $f(x) = y \Leftrightarrow xy = x$ and therefore $x \cdot yz = x \cdot yy = xy = xz = xy \cdot z$.
The statement (iii) mirrors (ii), and (iv) follows directly from (ii) and (iii). □

Lemma 3.3. *Let Q be a finite quasigroup, $I = I(Q)$. The number of elementary associative triples $(x, y, z) \in Q^3$ is equal to*

$$|I| - |Q| - |e^{-1}(I)| - |f^{-1}(I)| + \sum_{q \in Q} (|e^{-1}(q)|^2 + |f^{-1}(q)|^2 + |e^{-1}(q)||f^{-1}(q)|). \quad (3.1)$$

Proof. Put $\mathcal{T}_l = \{T \in Q^3; T \text{ is left elementary associative triple}\}$ and define $\mathcal{T}_r$ and $\mathcal{T}_m$ in the similar fashion. In order to determine the size of $\mathcal{T}_l \cup \mathcal{T}_r \cup \mathcal{T}_m$ start by obtaining their individual sizes. For fixed $z \in Q$ there are $|f^{-1}(z)|$ choices for both y and xy for left elementary triple (x, y, z) according to Lemma 3.1. This yields $|\mathcal{T}_l| = \sum_{q\in Q} |f^{-1}(q)|^2$. The case for $\mathcal{T}_r$ is analogous and gives $|\mathcal{T}_r| = \sum_{q\in Q} |e^{-1}(q)|^2$. Regarding the size of $\mathcal{T}_m$, we fix $y \in Q$ and see that $f(x) = y \Leftrightarrow x \in f^{-1}(y)$. Therefore we have $|f^{-1}(y)|$ choices for x. The same idea gives $|e^{-1}(y)|$ choices for z and from this it follows that $|\mathcal{T}_m| = \sum_{q\in Q} |e^{-1}(q)||f^{-1}(q)|$. To get the size of the union of sets $\mathcal{T}_l$, $\mathcal{T}_r$, and $\mathcal{T}_m$ we need to add their sizes together, subtract the size of their pairwise intersections, and add the size of the intersection of all three. From Lemma 3.2 we have that $\mathcal{T}_l \cap \mathcal{T}_r$ consists of triples for which $y \in e^{-1}(x) \cap f^{-1}(z)$. For fixed y this gives $x = e(y)$ and $z = f(y)$ and $|\mathcal{T}_l \cap \mathcal{T}_r| = |Q|$. In the other two cases the idea is very similar. For fixed $y \in I(Q)$ there are $|f^{-1}(y)|$ choices for x such that $(x, y, y) \in \mathcal{T}_l \cap \mathcal{T}_m$ and $|e^{-1}(y)|$ choices for z such that $(y, y, z) \in T_r \cap \mathcal{T}_m$. This makes $|\mathcal{T}_l \cap \mathcal{T}_m| = |f^{-1}(I)|$ and $|\mathcal{T}_r \cap \mathcal{T}_m| = |e^{-1}(I)|$. Lastly, $|\mathcal{T}_l \cap \mathcal{T}_r \cap \mathcal{T}_m|$ is clearly equal to $|I|$. □

For $Q = \{1, \dots, n\}$, $I(Q) = \{1, \dots, k\}$ put $a = (a_1, \dots, a_n)$, $a_i = |e^{-1}(i)|$, and $b = (b_1, \dots, b_n)$, $b_i = |f^{-1}(i)|$. Then the value of (3.1) is equal to $k - n + S(a, b)$, where

$$S(a, b) = \sum_{i=1}^{n} (a_i^2 + b_i^2 + a_i b_i) - \sum_{i=1}^{k} (a_i + b_i). \tag{3.2}$$

Claim 3.4. *For $n \geq k \geq 0$ consider integers $a_i, b_i \geq 0, 1 \leq i \leq n$ such that $\sum_{i=1}^{n} a_i = n$ and $\sum_{i=1}^{n} b_i = n$. Assume that $a_i, b_i \geq 1$ if $1 \leq i \leq k$. Denote by δ_L the number of i for which $a_i = 0$ and by δ_R the number of i with $b_i = 0$. Put $\delta = \delta_L + \delta_R$. Then*

(i) $S(a, b) = 3n - 2k$ if and only if $\delta = 0$;

(ii) $S(a, b) \geq 3n - 2k + \delta$;

(iii) $S(a, b) \geq 2n - k + 2\delta$ if $\delta \geq n - k$;

(iv) $S(a, b) \geq 4\delta$ if $\delta \geq n - k/2$; and

(v) $S(a, b) \geq 6\delta - 2n$ if $\delta \geq n$.

If any of the inequalities (ii)-(v) holds as an equality, then it is possible to reorder the set such that

$$a_1 \geq \dots \geq a_k,\ a_{k+1} \geq \dots \geq a_n,\ b_1 \leq \dots \leq b_k \text{ and } b_{k+1} \leq \dots \leq b_n.$$

Let this be true and let an equality (ii)-(v) hold as an equality. If $k > 0$, then $a_1 - a_k \leq 1$ and $b_k - b_1 \leq 1$. If $k + \delta_L < n$, then $a_{k+1} - a_{n-\delta_L} \leq 1$, and if $k + \delta_R < n$, then $b_n - b_{k+1-\delta_R} \leq 1$.

Full proof of Claim 3.4 is rather long and can be found in [10]. Still, note that if $\delta = 0$, then $a_i = b_i = 1$ for $1 \leq i \leq n$ which implies that both e and f are permutations.

Remark. Note that Lemma 3.3 together with Claim 3.4 give a lower bound on $\mathbf{a}(n)$. In case that both e and f are permutations, this lower bound is $2|Q| - |I|$, the same as in [4].

Let Q be a quasigroup. Set $\delta_L(Q) = |\{x{\in}Q;\ e^{-1}(x) = \emptyset\}|$ and $\delta_R(Q) = |\{x{\in}Q;\ f^{-1}(x) = \emptyset\}|$, and $\delta(Q) = \delta_L(Q) + \delta_R(Q)$. Note that $\delta_L(Q)$, $\delta_R(Q)$ are equal to the defects of mappings e and f respectively.

Theorem 3.5. *Let Q be a finite quasigroup. Then $\boldsymbol{a}(Q) \geq 2|Q| - \boldsymbol{i}(Q) + \delta(Q)$. If the equality holds, then all associative triples are elementary.*

Proof. Assume that $Q = \{1, \dots, n\}$. Put $k = \mathbf{i}(Q)$, $a_i = |e^{-1}(i)|$, and $b_i = |f^{-1}(i)|$, for every $i \in \{1, \dots, n\}$. Use (3.2) to define $S = S(a, b)$. Observe that δ from Claim 3.4 is equal to $\delta(Q)$. By Lemma 3.3 the number of elementary associative triples is equal to $k - n + S$. Thus, $\mathbf{a}(Q) \geq k - n + S$. From Claim 3.4 we have that $S \geq 3n - 2k + \delta(Q)$. Therefore, $\mathbf{a}(Q) \geq 2n - k + \delta(Q)$. This implies that if the equality holds, then all associative triples must be elementary. □

Throughout this thesis, we will use the following terminology regarding quasigroup with a small number of associative triples.

Definition. A finite quasigroup Q is called *extremely nonassociative* if $\mathbf{a}(Q) = 2|Q| - \mathbf{i}(Q)$. It is called *highly nonassociative* if $\mathbf{a}(Q) \leq 2|Q|$.

Corollary. For a finite extremely nonassociative quasigroup Q the mappings e and f are permutations and all associative triples are elementary.

Theorem 3.5 together with Claim 3.4 give rise to the following statement.

Claim 3.6. *Let Q be a quasigroup defined upon $\{1, \dots, n\}$. Put $a_i = |e^{-1}(i)|$ and $b_i = |f^{-1}(i)|$, $1 \leq i \leq n$. Assume that the idempotents of Q are the elements $1, \dots, k$. Use (3.2) to define $S = S(a, b)$. Put $r = \delta_L$, $s = \delta_R$, and $\delta = \delta(Q)$. Then $\boldsymbol{a}(Q) \geq k - n + S \geq 2n - k + \delta$. If $\boldsymbol{a}(Q) = 2n - k + \delta$, then Q may be reordered in such way that*

(1) $a_i \geq a_{i+1}$ and $b_i \leq b_{i+1}$ if $1 \leq i < k$ or $k \leq i < n$,

(2) $a_k + 1 \geq a_1$ and $b_1 + 1 \geq b_k$ if $k > 0$,

(3) $a_{n-r} + 1 \geq a_{k+1}$ if $n > k + r$, and

(4) $b_{k+1+s} + 1 \geq b_n$ if $n > k + s$.

3.1 Mappings that cannot occur

It is interesting to note that in some cases it is not possible to have both e and f as permutations. This implies that in such cases the lower bound $\mathbf{a}(Q) = 2|Q| - |I(Q)|$ will never be achieved.

Lemma 3.7. *Consider quasigroup Q such that $\boldsymbol{i}(Q) = |Q| - 1$. Mappings e and f are not permutations and the number of elementary associative triples is $|Q| + 4$ or $|Q| + 5$.*

Proof. Suppose that e is a permutation. Permutation e fixes exactly $|Q|-1$ points. The only permutation on $|Q|$ elements which fixes $|Q| - 1$ points is the identity which fixes all $|Q|$ points. Therefore, e cannot be a permutation. Observe that both e and f are near-permutations. This implies that $\delta_L(Q) = 1 = \delta_R(Q)$

and $\delta(Q) = 2$. Claim 3.4 (*iii*) gives the lower bound on number of elementary associative triples equal to $|Q| + 4$. This lower bound can be achieved when $s_e \neq s_f$ where s_e is the only element in $N(e)$ and s_f is the only element in $N(f)$. Substitute values of e and f into equation (3.1). $|I| - |Q| - |e^{-1}(I)| - |f^{-1}(I)| + \sum_{q \in Q}(|e^{-1}(q)|^2 + |f^{-1}(q)|^2 + |e^{-1}(q)||f^{-1}(q)|) = |Q| - 1 - 3|Q| + (|Q| - 3) * 3 + 2 * 7 + 0 = Q + 4$.

If $s_e = s_f$ the same equation gives the number of elementary associative triples to be $|Q| + 5$. □

The table bellow illustrates Lemma 3.7 for Q=$\{1, 2, 3, 4, 5, 6\}$, I=$\{1, 2, 3, 4, 5\}$.

	1	2	3	4	5	6
1	$\boxed{1}$	·	·	·	·	·
2	·	$\boxed{2}$	·	·	·	·
3	·	·	$\boxed{3}$	·	·	·
4	·	·	·	$\boxed{4}$	·	$\boxed{6}$
5	·	·	·	·	$\boxed{5}$	·
6	·	·	$\boxed{6}$	·	·	$\boxed{1}$

Table 3.1: Mappings e and f in quasigroup with 5 idempotents.

Lemma 3.8. *Consider quasigroup Q such that $\boldsymbol{i}(Q) = |Q| - 2$. Then only one of the mappings e and f may be a permutation. The smallest number of elementary associative triples is $|Q| + 4$.*

Proof. Suppose that e is a permutation. Since it fixes exactly $|Q| - 2$ points, it has to be a transposition on the last two points. Denote by a, b two distinct elements for which $aa \neq a$ and $bb \neq b$. Therefore, $e(a) = b$ and $e(b) = a$ which means that $ba = a$ and $ab = b$. Using the same argument for f being a permutation observe that this implies that $f(a) = b$ and $f(b) = a$ which leads to the contradiction since it means that $ab = a$ and $ba = b$. From this it is easy to see that $\delta(Q) \geq 2$. The minimum number of elementary associative triples is obtained from Claim 3.4 and can be achieved when one of mappings e and f is a permutation. □

The table below illustrates Lemma 3.8 for $Q = \{1, 2, 3, 4, 5, 6\}, I = \{1, 2, 3, 4\}$ and f being a permutation.

	1	2	3	4	5	6
1	$\boxed{1}$	·	·	·	·	·
2	·	$\boxed{2}$	·	·	·	·
3	·	·	$\boxed{3}$	·	$\boxed{5}$	·
4	·	·	·	$\boxed{4}$	·	$\boxed{6}$
5	·	·	·	·	$\boxed{1}$	$\boxed{5}$
6	·	·	·	·	$\boxed{6}$	$\boxed{2}$

Table 3.2: Mappings e and f in quasigroup with 4 idempotents.

Lemmas 3.7 and 3.8 show that quasigroup Q of order n cannot be extremely nonassociative if $\mathbf{i}(Q) \in \{n-1, n-2\}$. As mentioned before, in extremely nonassociative quasigroup mapping e is a permutation. The number of different types of permutations e is equal to $q(n - \mathbf{i}(Q))$. By adding up all types of permutation e we obtain the following.

Claim 3.9. *For a finite extremely nonassociative quasigroup Q of order n mapping e is a permutation and there are $p(n)-1$ possible types of those permutations.*

In the next section, we extend this idea to finding highly nonassociative quasigroup.

3.2 Finding highly nonassociative quasigroups

The main objective of this section is to explain how to use Theorem 3.5 to find all highly nonassociative quasigroups of orders 8 and 9.

The idea how to conquer the problem of finding such quasigroups is to divide the problem into a limited number of cases, based on the mappings e, f or d from Lemma 2.1, in each of which it is possible to fill (at least) $|Q|$ cells of the multiplication table. For example, if mapping e is known, then the cells $(e(x), x)$ can be filled by x.

Let Q be a highly nonassociative quasigroup of order n. Put $k = \mathbf{i}(Q)$, $\delta_L = \delta_L(Q)$, $\delta_R = \delta_R(Q)$, and $\delta = \delta(Q)$. By Theorem 3.5, $\delta \leq k$.

3.2.1 Order 8

If $k = 0$, then $\delta = 0$, and there are $q(8) = 7$ possible types of transformation e.

If $k = 1$, then $\delta_L + \delta_R \leq 1$. We can assume that $\delta_L = 0$ because mirroring does not change the number of associative triples. This means to test $q(7) = 4$ permutations e.

If $k = 2$, then $\delta_L + \delta_R \leq 2$. It may be assumed that the defect of e is at most one. This gives $q(6) + t_2(7) = 18$ cases to test, by Lemma 1.3.

If $k = 3$, then the situation is similar since $\delta_L + \delta_R \leq 3$. By Lemma 1.3, there are $q(5) + t_2(6) = 11$ cases to consider.

If $k = 4$, then $\delta_L + \delta_R \leq 4$. The situations with $\delta_L \leq 1$ or $\delta_R \leq 1$ are covered by choosing e with the defect at most one. This yields $q(4) + t_2(5) = 7$ cases. Other 7 cases are those from Lemma 1.5. Let us observe that no other cases are needed. Suppose that $\delta_L = \delta_R = 2$ and put $a_i = |e^{-1}(i)|$, $1 \leq i \leq 8$. Remember that $\sum_{i=1}^{8} a_i = 8$. By Claim 3.6 it may be assumed that: $1, \dots, 4$ are idempotents, $a_1 \geq a_2 \geq a_3 \geq a_4 \geq 1$, $a_5 \geq a_6 \geq a_7 \geq a_8$, $a_1 - a_4 \leq 1$, and $a_5 - a_6 \leq 1$. A case not yet covered has to have i such that $a_i \geq 3$. Observe that this is not possible. If $a_1 \geq 3$, then $a_4 \geq 2$ and $a_1 + a_2 + a_3 + a_4 \geq 9$ which is not possible. Since $a_1 + a_2 + a_3 + a_4 \geq 4$, it is necessary that $a_5 + a_6 + a_7 + a_8 \leq 4$. If $a_5 \geq 3$, then $a_6 \geq 2$ and $a_5 + a_6 \geq 5$. Thus, $a_i \leq 2$ for every $i \in \{1, \dots, 8\}$.

If $k \geq 5$ consider diagonal mapping $d : x \mapsto xx$. By Lemma 1.4 there are 16 cases to be considered for filling the diagonal.

Together, there are $7 + 4 + 18 + 11 + 14 + 16 = \mathbf{70}$ cases that cover the whole search.

3.2.2 Order 9

If $k = 0$, then $\delta = 0$, and there are $q(9) = 8$ types of possible transformation e.

If $k = 1$, then $\delta_L + \delta_R \leq 1$. Also assume that $\delta_L = 0$. This means to test $q(8) = 7$ permutations e.

If $k = 2$, then $\delta_L + \delta_R \leq 2$. It may be assumed that the defect of e is at most one. This gives $q(7) + t_2(8) = 27$ cases to test, by Lemma 1.3.

If $k = 3$, then the situation is similar since $\delta_L + \delta_R \leq 3$. By Lemma 1.3, there are $q(6) + t_2(7) = 18$ cases to consider.

If $k = 4$, then $\delta_L + \delta_R \leq 4$. The situations with $\delta_L \leq 1$ or $\delta_R \leq 1$ are covered by choosing e with defect at most one. This yields $q(5) + t_2(6) = 11$ cases. Other 15 cases are those from Lemma 1.6. As for the order 8 observe that no other cases are needed. Suppose that $\delta_L = \delta_R = 2$ and put $a_i = |e^{-1}(i)|$, $1 \leq i \leq 9$. By Claim 3.6 it may be assumed that: $1, \ldots, 4$ are idempotents, $a_1 \geq a_2 \geq a_3 \geq a_4 \geq 1$, $a_5 \geq a_6 \geq a_7 \geq a_8 \geq a_8$, $a_1 - a_4 \leq 1$, and $a_5 - a_7 \leq 1$. A case not yet covered has to have i such that $a_i \geq 3$. Observe that this is not possible. If $a_1 \geq 3$, then $a_4 \geq 2$ and $a_1 + a_2 + a_3 + a_4 \geq 9$. Since $\sum_{i=1}^{9} a_i = 9$, hence, $a_i = 0$ for $i > 4$ and $\delta_L = 5$ which is a contradiction. Since $a_1 + a_2 + a_3 + a_4 \geq 4$, it is necessary that $a_5 + a_6 + a_7 + a_8 + a_9 \leq 5$. If $a_5 \geq 3$, then $a_7 \geq 2$ and $a_5 + a_6 + a_7 \geq 7$. Thus, $a_i \leq 2$ for every $i \in \{1, \ldots, 9\}$.

If $k = 5$, then $\delta_L + \delta_R \leq 5$. We can assume that $\delta_L \leq 2$. The situations with $\delta_L \leq 1$ or are covered by choosing e with defect at most one. This yields $q(4) + t_2(5) = 7$ cases. Other 7 cases are those from Lemma 1.7. The argument that this covers all cases follows the same steps as before.

If $k \geq 6$ consider diagonal mapping $d : x \mapsto xx$. By Lemma 1.4 there are 16 cases to be considered for filling the diagonal.

Together, there are $8 + 7 + 27 + 18 + 26 + 14 + 16 = \mathbf{116}$ cases that cover the whole search.

After enumerating all cases that cover the search of highly nonassociative quasigroups of orders eight and nine we can continue by presenting the algorithm that performs the search but more importantly counts the associative triples at the time when they arise.

4. The time for an associative triple

In this chapter, we are going to describe the process of counting associative triples in a partially constructed quasigroup. These computations are one of the main results of this thesis. The quasigroups are constructed by natural approach - row by row, top to bottom, left to right. The number of associative triples is updated in each step. If it exceeds a predetermined threshold, the partial quasigroup is no longer considered.

Throughout this chapter we will assume that quasigroup Q is defined on set $\{0, \dots, n-1\}$. Let us fix the vocabulary for dealing with associative triples in partially constructed quasigroups.

Definition. The set $Q \times Q$ is ordered linearly so that $(a, b) < (c, d)$ if and only if $a < c$ or $a = c \wedge b < d$.

Definition. For a partially constructed quasigroup Q define the *time* $(a, b) \in Q^2$ to be that step in the construction when all products $c \cdot d; (c, d) \leq (a, b)$ are determined but no product $g \cdot h; (g, h) > (a, b)$ is.

Definition. Let Q be a quasigroup. For a triple $(x, y, z) \in Q^3$ denote by $t(x, y, z)$ the least time $(a, b) \in Q^2$ when both $x \cdot yz$ and $xy \cdot z$ can be computed. Call $t(x, y, z)$ the *decisive pair* of the triple (x, y, z).

	0	1	2	3	4	5
0	0	2	1	3	4	5
1	2	1	5	0	3	4
2	3	0	2	4	5	1
3	4	5	3	1	0	2
4	5	3	$\boxed{4}$	·	·	·
5	·	·	·	·	·	·

Table 4.1: A partially constructed quasigroup of size 6 at time $(4, 2)$.

Since $(a, b) = t(x, y, z) = \max\{(x, y), (x, yz), (xy, z), (y, z)\}$, (a, b) can appear in only those four *roles*. Denote them by (A), (B), (C), (D).

$$\begin{aligned} &\text{(A) } (a, b) = (x, y);\\ &\text{(B) } (a, b) = (x, yz);\\ &\text{(C) } (a, b) = (xy, z); \text{ or}\\ &\text{(D) } (a, b) = (y, z). \end{aligned} \tag{4.1}$$

Remark. (A) and (D), and (B) and (C) are mirror symmetric.

Lemma 4.1. *Let Q be a quasigroup, $(a, b) \in Q^2$ and $(x, y, z) \in A(Q)$ be an associative triple. If* (A) *and* (C) *hold, then* (B) *holds as well. If* (B) *and* (D) *hold, then* (C) *holds as well.*

Proof. If (A) and (C) hold, then $x = a = xy$ and $y = b = z$. Substituting those values into equation $x(yz) = (xy)z$ we get $a(bb) = ab$, thus $yz = b$ and (B) holds as well. If (B) and (D) hold, then $x = a = y$ and $yz = b = z$. Also substitute those values into equation $x(yz) = (xy)z$ and observe that $ab = (aa)b$ and therefore (C) holds as well. □

Definition. Call a triple $(x, y, z) \in Q^3$ *diagonal* if $x = y = z$ and call it *diagonally idempotent* if $x = y = z \in I(Q)$.

Lemma 4.2. *Let Q be a quasigroup, $(x, y, z) \in Q^3$, $(a, b) \in Q^2$. At least two out of four equations in* (4.1) *are true if and only if at least one of the following holds:*

(i) Both (A) *and* (B) *are true if and only if x=a, y=b, and (x, y, z) is left elementary.*

(ii) Both (A) *and* (D) *are true if and only if $x = y = z = a = b$.*

(iii) Both (B) *and* (C) *are true if and only id x=a, z=b, and (x, y, z) is middle elementary.*

(iv) Both (C) *and* (D) *are true if and only if y=a, z=b, and (x, y, z) is right elementary.*

Proof. From Lemma 4.1 we know that options (i)-(iv) are sufficient to cover all distinct cases. If (A) and (B) hold, then $x = a$ and $y = yz = b$. Furthermore, $ab \cdot z = a \cdot bz = ab$ holds because (x, y, z) is an associative triple and therefore (x, y, z) is left elementary. The opposite direction is easy. The situation of the case (ii) is clear. In case (iii) observe that $x = xy = a$ and $yz = y = b$ if and only if $x = a$, $z = b$, and $f_a = y = e_b$. That is true if and only if (x, y, z) is middle elementary according to Lemma 3.1. If (C) and (D) hold, then $y = xy = a$ and $z = b$. Furthermore, $x \cdot ab = xa \cdot b = ab$ holds. Thus, (x, y, z) is right elementary. □

Corollary. In a quasigroup Q let $(x, y, z) \in A(Q)$ and $(a, b) \in Q^2$ such that at least two of the four equations in (4.1) are true. Then (x, y, z) is elementary or diagonal.

Corollary. Nonelementary and nondiagonal associative triples (x, y, z) appear in exactly one set $\mathcal{A}_{(a,b)}$, $\mathcal{B}_{(a,b)}$, $\mathcal{C}_{(a,b)}$, or $\mathcal{D}_{(a,b)}$ defined bellow.

$$\begin{aligned}
\mathcal{A}_{(a,b)} &= \{(x, y, z) \in A(Q);\ t(x, y, z) = (a, b) = (x, y)\},\\
\mathcal{B}_{(a,b)} &= \{(x, y, z) \in A(Q);\ t(x, y, z) = (a, b) = (x, yz)\},\\
\mathcal{C}_{(a,b)} &= \{(x, y, z) \in A(Q);\ t(x, y, z) = (a, b) = (xy, z)\}, \text{ and}\\
\mathcal{D}_{(a,b)} &= \{(x, y, z) \in A(Q);\ t(x, y, z) = (a, b) = (y, z)\}.
\end{aligned}$$

This provides a useful distinction between types of nonelementary associative triples. Size of these sets can be computed at each time (a, b) during construction of quasigroup and add up to a total. Diagonal and elementary associative triples are being counted separately.

The following Lemma 4.3 gives a criterion how to recognize elementary associative triples.

Lemma 4.3. *Let Q be a quasigroup and (x, y, z) is an associative triple. It is elementary if and only if $xy \in \{x, y\}$ or $yz \in \{y, z\}$.*

Proof. Direct consequence of Lemma 4.2. □

Finally, in the next chapter we will present an algorithm that counts nonelementary associative triples based on their roles. It also counts elementary triples when needed according to the Lemma 4.3.

5. Algorithm

In this chapter we are going to use mathematical results from Chapter 4 to devise an algorithm for efficient counting of associative triples in partially constructed quasigroups. An associative triple is counted depending on being elementary or nonelementary. The elementary triples are counted separately and in some cases their number can be known beforehand. The nonelementary triples are counted based on their role. Denote by $\mathcal{B}^\star_{(a,b)} = \{(x,y,z) \in \mathcal{B}_{(a,b)};\ (x,y,z)\ \textit{is not elementary}\}$ the nonelementary part of $\mathcal{B}_{(a,b)}$. Similarly define $\mathcal{C}^\star_{(a,b)}$ and $\mathcal{D}^\star_{(a,b)}$. The set $\mathcal{A}^\star_{(a,b)} = \{(x,y,z) \in \mathcal{A}_{(a,b)};\ (x,y,z)\ \textit{is not elementary and not diagonal}\}$ is defined as such in order to resolve an ambiguity mentioned by point (ii) of Lemma 4.2. The diagonal triples (a,a,a) are counted as elements of the set $\mathcal{D}^\star_{(a,b)}$. Thus, the intersection of any pair of these sets is empty. The algorithm consists of procedures that enumerate those sets. Clearly, the number of nonelementary associative triples at time (a,b) is equal to $\sum_{(i,j)=(0,0)}^{(a,b)} |\mathcal{A}^\star_{(i,j)}| + |\mathcal{B}^\star_{(i,j)}| + |\mathcal{C}^\star_{(i,j)}| + |\mathcal{D}^\star_{(i,j)}|$.

For enumeration of these sets a procedure called `test` is used. It compares two elements of the quasigroup and returns 1 if they are the same and 0 if not.

```
function test($x$, $y$)
    if $x = y$
        return 1
    else
        return 0
```

In this section it is always assumed that at time (a,b) the newly filled value is $c = a \cdot b$. Note that except quasigroup operation $\cdot$ we will also use the left division operation $\backslash$.

5.1 Nonelementary triples

The set $\mathcal{A}^\star_{(a,b)}$ contains triples (a,b,z) for which $ab = c$ and $c \notin \{a,b\}$ according to the Lemma 4.3. Observe that if $a < b$, then $\mathcal{A}^\star_{(a,b)} = \emptyset$ since no product bz is known. Similarly, $\mathcal{A}^\star_{(a,b)} = \emptyset$ if $a < c$ since no product cz is known. Focus on case $a > b$. The enumeration of those triples can be done in the following way:

```
if $a > b$ and $a > c$ and $b \neq c$
    for $z = 0, \ldots, n-1$
        if $bz < b$
            test($cz$, $a \cdot bz$)
```

Note that a straightforward simplification of this procedure using the left division can be performed. Put $r = bz$ and observe that if $r > b$, then the product ar is not yet known and if $b = r$, then $\mathcal{A}_{(a,b)}$ contains only elementary triples and therefore $\mathcal{A}^\star_{(a,b)} = \emptyset$. Thus, a simplified procedure is as follows:

```
if $a > b$ and $a > c$ and $b \neq c$
    for $r = 0, \ldots, b-1$
        test($c \cdot (b \backslash r), ar$)
```

Observe that this saves on average $n/2$ steps in **for** cycle. Therefore, we will from this point only mention simplified versions of enumeration functions.

Now, in case that $a = b$ we have $\mathcal{A}^{\star}_{(a,a)} = \emptyset$ if $a \leq c$ for the similar reasons as mentioned above. Assuming $a > c$ observe that if $z > a$, then az is not known. Thus, the restrictions are $a = b$, $a > c$, and $z < a$. Put $s = az$. If $s > a$ we do not know as and if $s = a$, then (a, a, z) would be elementary. Together with the case $a > b$, we can enumerate $\mathcal{A}^{\star}_{(a,b)}$ using procedure `countA`*.

```
function countA*($a$, $b$, $c$)
    count = 0
    if $a > c$
        if $a > b$ and $b \neq c$
            for $r = 0, \dots, b-1$
                count += test($ar$, $c \cdot (b \backslash r)$)
        if $a = b$
            for $z = 0, \dots, a-1$
                $s = az$
                if $s < a$
                    count += test($as$, $cz$)
    return count
```

The set $\mathcal{B}^{\star}_{(a,b)}$ contains triples (a, y, z) for which $yz = b$, $ay \notin \{a, y\}$, and $b \notin \{y, z\}$, according to Lemma 4.3. Divide the enumeration into two cases $a \geq b$ and $a < b$. If $a \geq b$, then the conditions for y are $y \leq b$ (ay has to be known) and $y \neq b$ (otherwise the triple would be elementary). The conditions for ay are $ay \leq a$ ($(ay)z$ has to be known) and $ay \notin \{a, y\}$. Note that $z = y \backslash b$; thus, case for $a \geq b$ is as follows:

```
if $a \geq b$
    for $y = 0, \dots, b-1$
        $l = ay$
        if $l < a$ and $l \neq y$
            count += test($c, l \cdot (y \backslash b)$)
```

If $a < b$, then the condition for y is $y \leq a$ (yz has to be known). The conditions for ay are the same as in the previous case. Observe that for $y < a$ the enumeration may proceed like when $a \geq b$. If $y = a$, then $a \backslash b$ might not be defined. Hence, the test must be performed in another way. Put $z = (aa) \backslash c$, meaning $(aa)z = c$, which is known for $aa < a$. If (a, a, z) is an associative triple, then $c = (aa)z = a(az)$. Therefore we only need to test if $az = b$, provided that $z \leq b$.

The enumeration of $\mathcal{B}^\star_{(a,b)}$ can thus be done in this way:

```
function countB*(a, b, c)
    count = 0
    if $a \geq b$
        for $y = 0, \ldots, b{-}1$
            $l = ay$
            if $l < a$ and $l \neq y$
                count += test($c, l \cdot (y\backslash b)$)
    if $a < b$
        for $y = 0, \ldots, a{-}1$
            $l = ay$
            if $l < a$ and $l \neq y$
                count += test($c, l \cdot (y\backslash b)$)
        $l = aa$
        if $l < a$
            $z = l\backslash c$
            if $z \leq b$
                count += test($b, az$)
    return count
```

The set $\mathcal{C}^\star_{(a,b)}$ contains triples (x, y, b) where $xy = a$. Observe that $y = x\backslash a$. The restrictions on both x and y are $x, y \leq a$ and $x, y \neq a$. There is one more restriction on y, $y \neq yb$ in order to avoid elementary triple. The enumeration of $\mathcal{C}^\star_{(a,b)}$ is as follows:

```
function countC*(a, b, c)
    count = 0
    for $x = 0, \ldots, a{-}1$
        $y = x\backslash a$
        if $y < a$ and $y \neq yb$
            count += test($c, x \cdot yb$)
    return count
```

Finally, the set $\mathcal{D}^\star_{(a,b)}$ contains triples (x, a, b) with restrictions on $c \notin \{a, b\}$ and $xa \notin \{x, a\}$. Observe that $x \leq a$ otherwise xa would not be known. Divide the enumeration into two parts. The first part covers cases where $x < a$. The enumeration of those cases is as follows.

```
for $x = 0, \ldots, a{-}1$
    $l = xa$
    if $l < a$
        test($xc, lb$)
```

Observe, that inequality $l \neq x$ does not have to be tested since if $xa = x$, then the triple is associative if and only if $c = b$ which we assume is not true.

The second part of the enumeration covers cases where $x = a$. Therefore, $aa < a$ and $ab < b$. This also covers the testing for diagonal triple (a, a, a).

The whole enumeration function may look like this.

```
function countD*($a$, $b$, $c$)
    count = 0
    if $a \neq c$ and $b \neq c$
        for $x = 0, \ldots, a{-}1$
            $l = xa$
            if $l < a$
                count += test($xc, lb$)
        if $a \leq b$
            $l = aa$
            if $l < a$ and $c < b$
                count += test($ac, lb$)
    return count
```

At the end of this section, we would like to note that each counting procedure can return values greater than one. Table 5.1 presents a partially constructed quasigroup of order 9. The construction is at time $(a, b) = (8, 7)$ and filled value is $c = 6$. It is easy to verify that $\mathcal{A}^\star_{(8,7)} = \{(8,7,1),(8,7,2)\}$, $\mathcal{B}^\star_{(8,7)} = \{(8,0,0),(8,1,1)\}$, $\mathcal{C}^\star_{(8,7)} = \{(0,1,7),(1,0,7),(4,6,7)\}$, and $\mathcal{D}^\star_{(8,7)} = \{(0,8,7),(1,8,7)\}$.

	0	1	2	3	4	5	6	7	8
0	7	8	6	0	4	5	2	3	1
1	8	7	4	6	5	1	3	2	0
2	6	1	0	2	3	4	5	7	8
3	0	6	2	1	7	3	4	8	5
4	1	4	5	3	2	6	8	0	7
5	3	5	7	4	0	8	6	1	2
6	4	2	3	7	8	0	1	5	6
7	5	0	1	8	6	2	7	4	3
8	2	3	8	5	1	7	0	$\boxed{6}$	·

Table 5.1: An example of time in quasigroup that induces many nonelementary associative triples.

5.2 Condensed nonelementary triples

The following section introduces a modification of the algorithm where nonelementary triples are counted before $c = ab$ is filled. Modified functions return n-tuple t (instead of integer value) such that $t[c]$ is equal to number of new nonelementary triples in corresponding set provided that newly filled value is $c = ab$. This tuple is then used when filling out partially constructed quasigroup. If the construction is done recursively, then the value $c = ab$ is considered only when $t[c]$ is less than maximum targeted value. The tuple t is computed only once and then used again when backtracking.

This modification can be done efficiently using right division operation $/$. Further overhead is introduced by maintaining the special value **nil** to indicate that the result of the operation is not known.

In this version the function test is replaced by the equation that gives the value of c for which $t[c]$ is to be incremented.

Starting with the enumeration of $\mathcal{A}^\star_{(a,b)}$, it is the only one that uses right division. The modification of function countA* is straightforward:

```
function c_countA*(a, b)
    $t = [0, \ldots, 0]$
    if $a > b$
        for $r = 0, \ldots, b{-}1$
            $c = (ar)/(b\backslash r)$
            if $c \neq$ nil and $a > c$ and $b \neq c$
                $t[c]$ += 1
    if $a = b$
        for $z = 0, \ldots, a{-}1$
            if $az < a$
                $c = (a(az))/z$
                if $c \neq$ nil and $a > c$
                    $t[c]$ += 1
    return $t$
```

The modification of function countB* benefits from the introduction of **nil**. We can simplify the procedure by testing if $a\backslash b$ is defined.

```
function c_countB*(a, b)
    $t = [0, \ldots, 0]$
    if $a \geq b$
        for $y = 0, \ldots, b{-}1$
            $l = ay$
            if $l < a$ and $l \neq y$
                $t[l \cdot (y\backslash b)]$ += 1
    if $a < b$
        for $y = 0, \ldots, a{-}1$
            $l = ay$
            if $l < a$ and $l \neq y$
                $t[l \cdot (y\backslash b)]$ += 1
        $l = aa$
        if $l < a$ and $a\backslash b \neq$ nil
            $t[l \cdot (a\backslash b)]$ += 1
    return $t$
```

The modifications of functions countC* and countD* follow the similar steps in a straightforward way.

```
function c_countC*(a, b)
    $t = [0, \ldots, 0]$
    for $x = 0, \ldots, a{-}1$
        $y = x\backslash a$
        if $y < a$ and $y \neq yb$
            $t[x \cdot yb]$ += 1
    return $t$
```

```
function c_countD$^\star$($a$, $b$)
    $t = [0, \ldots, 0]$
    for $x = 0, \ldots, a{-}1$
        $l = xa$
        if $l < a$
            $c = x\backslash(lb)$
            if $c \neq a$ and $c \neq b$
                $t[c]$ += 1
    if $a \leq b$
        $l = aa$
        if $l < a$
            $c = a\backslash(lb)$
            if $c \neq$ nil  and $c < b$ and $c \neq a$
                $t[c]$ += 1
    return $t$
```

5.3 Elementary triples

The number of elementary triples is determined only by mappings e and f and is given by formula in Lemma 3.3. If the mappings are known in advance, then the search is reduced to counting the nonelementary triples and backtracking when their number surpasses the threshold.

In case that e, f, or both are not fully known we propose the following search algorithm. Suppose, as in Sections 3.2.1 and 3.2.2, that $I(Q)$ is known. Denote by c_0 the estimate for the number of elementary associative triples obtained from Theorem 3.5 and by c_n the number of nonelementary associative triples in the partially constructed quasigroup. Naturally, the number of elementary triples is denoted by c_e.

A natural strategy is to test $c_n + c_0$ against threshold as long as $c_e \leq c_0$. Then continue testing $c_n + c_e$.

The value of c_e can be computed directly from formula in Lemma 3.3 and it changes whenever $ab \in \{a, b\}$. This results in formula (5.1).

For $x \in Q = \{0, 1, \ldots n{-}1\}$ set $\varepsilon_x = |e^{-1}(x)|$ and $\eta_x = |f^{-1}(x)|$. Set $i(x) = 1$ if $x \in I(Q)$, and $i(x) = 0$ otherwise. The initial values of ε_x and η_x are equal to $i(x)$. Denote by k the number of idempotents in Q.

$$c_e = k - n + \sum_x (\varepsilon_x^2 + \eta_x^2 + \varepsilon_x \eta_x - i(x)(\varepsilon_x + \eta_x)). \tag{5.1}$$

Therefore, in the beginning $c_e = 2k - n$. From the formula (5.1) we can determine the value by which to increment c_e. For example if η_x is incremented by one, then c_e has to be incremented by $\eta_x + 1^2 + \varepsilon_x(\eta_x + 1) - i(x)(\eta_x + 1) - \eta_x^2 - \varepsilon_x \eta_x + i(x)\eta_x = 2\eta_x + \varepsilon_x + 1 - i(x)$.

Thus, when filling out partially considered quasigroup two conditions have to be checked. If $ab = a \neq b$, then $f(a) = b$, η_b += 1, and c_e += $2\eta_b + \varepsilon_b + 1 - i(b)$. If $ab = b \neq a$, then $e(b) = a$, ε_a += 1, and c_e += $2\varepsilon_a + \eta_a + 1 - i(a)$.

This concludes the chapter describing the algorithm for counting associative triples. In the next chapter we present the results that have been achieved by implementing and using this algorithm on quasigroups of orders eight and nine. A nontrivial part of the success of our effort was optimization of the implementation. Therefore, we will dedicate a whole chapter to the implementation details that proved to be crucial when implementing this algorithm. Lastly, we will compare the unoptimized implementation of this algorithm to the optimized version and to the previous version of the algorithm that was used in [5] to determine the minimum number of associative triples among quasigroups of orders up to seven.

6. Results and future work

In this chapter, we present the results of our search for the minimum value of $\mathbf{a}(n)$ for small n. For orders one through six the minimum values have been determined by Ježek and Kepka in [2]. We carried out the search and determined the value of $\mathbf{a}(7)$ in [5]. In this thesis we determined values $\mathbf{a}(8)$ (also published in [10]) and $\mathbf{a}(9)$.

This chapter also presents a classification of extremal quasigroups as well as an analysis of their automorphism groups and other properties. The analysis might provide insights into the construction of extremal quasigroups of a higher order. Lastly, we show a construction of infinite series of quasigroups for which $\mathbf{a}(Q) = |Q|$.

6.1 Order 8

We carried out the search for highly nonassociative quasigroups of order eight, those for which $\mathbf{a}(Q) \leq 16$. The extremal value is 16 and is achieved in quasigroups with zero idempotents. Thus all associative triples in extremal cases are elementary and both mappings e and f are permutations.

We have found 6 examples up to isomorphism. However, when considering the results up to isomorphism and mirroring there are only 3 unique quasigroups. Tables 6.1, 6.2, and 6.3 present all found quasigroups up to isomorphism and mirroring.

	0	1	2	3	4	5	6	7
0	2	7	1	3	5	0	4	6
1	0	3	4	2	7	6	1	5
2	3	1	0	5	6	4	7	2
3	6	0	2	1	3	7	5	4
4	4	6	5	0	2	1	3	7
5	1	5	7	6	4	3	2	0
6	7	2	6	4	1	5	0	3
7	5	4	3	7	0	2	6	1

Table 6.1: Extremely nonassociative quasigroup Q_1 with $\mathbf{a}(Q_1)$=16, e=(0 1 2 3)(4 5 6 7), f=(0 5 1 6 2 7 3 4), f^2=e, and $\mathrm{Aut}(Q_1)$=$\langle e \rangle$.

6.1.1 Classification

When searching for quasigroups with the lowest number of associative triples we are interested in the results up to isomorphism or up to main class.

To distinguish between main classes we identify several main class invariants.

Definition. For finite quasigroup Q call non-empty $R \subseteq Q$ a *subquasigroup* of Q if R is closed for multiplication.

	0	1	2	3	4	5	6	7
0	1	6	4	3	0	2	7	5
1	7	0	2	5	3	1	4	6
2	5	1	3	7	6	4	2	0
3	0	4	6	2	5	7	1	3
4	4	2	5	0	1	6	3	7
5	3	5	1	4	7	0	6	2
6	2	7	0	6	4	3	5	1
7	6	3	7	1	2	5	0	4

Table 6.2: Extremely nonassociative quasigroup Q_2 with $\mathbf{a}(Q_2)$=16, e=(0 3)(1 2)(4 6 5 7), f=(0 4)(1 5)(2 6 3 7), and Aut(Q_2)=$\langle\varphi\rangle$, φ = (0 1)(2 3)(4 5)(6 7).

	0	1	2	3	4	5	6	7
0	3	7	2	1	6	0	5	4
1	2	0	4	3	5	7	1	6
2	0	3	1	5	7	6	4	2
3	6	1	0	2	3	4	7	5
4	5	4	6	0	1	3	2	7
5	1	6	5	7	4	2	0	3
6	4	2	7	6	0	5	3	1
7	7	5	3	4	2	1	6	0

Table 6.3: Extremely nonassociative quasigroup Q_3 with $\mathbf{a}(Q_3)$=16, e=(0 2)(1 3)(4 5 6 7), f=(0 5 2 7)(1 6 3 4), and Aut(Q_3)=$\langle\psi\rangle$, ψ = (0 1 2 3)(4 5 6 7).

Definition. Let Q be a quasigroup. A permutation $\varphi \in S_Q$ is a *complete mapping* if the mapping $x \mapsto x\varphi(x)$ is also a permutation.

The number of subquasigroups of orders 2 and 3, the number of complete mappings, and the order of autotopy group are the main class invariants as shown in [11]. We will use them to specify the main class and denote them accordingly:

(1) the number of complete mappings,

(2) the number of order 2 subquasigroups,

(3) the number of order 3 subquasigroups, and

(4) the order of autotopy group.

The analysis of autotopy groups of quasigroups Q_1, Q_2, and Q_3 can be found in [10].

The quasigroup Q_3 is autotopic to Q_1 under autotopy ((1 3)(5 7), (0 2)(4 6), (1 3)(4 6)) which means that they belong to the same main class.

The Table 6.4 presents the classification of main classes of extremal quasigroups of order eight carried out by us using quasigroup classification software developed by the author of this thesis.

	(1)	**(2)**	**(3)**	**(4)**
Q_1	88	12	0	4
Q_2	112	10	0	8

Table 6.4: Classification of extremal quasigroups of order 8.

6.2 Order 9

Due to the exponential increase in the number of quasigroups we have been able to only accomplish the search for extremely nonassociative quasigroups, those for which $\mathbf{a}(Q) = 2|Q| - |I(Q)|$. Up to isomorphism and mirroring there exist only two quasigroups. They appear in Tables 6.5 and 6.6. Note that Q_4 is the first published example of a quasigroup with $\mathbf{a}(Q) = |Q|$ making it the main focus of our attention for the rest of this chapter. We have not been able to fully accomplish the search for highly nonassociative quasigroups of order 9. We finished the search in quasigroups with 0, 1, and 2 idempotents. Table 6.7 presents the only example of highly but not extremely nonassociative quasigroup we found.

6.2.1 Classification

Main class invariants of Q_4, Q_5, and Q_6 can be found in Table 6.8

The automorphism group of Q_4, $\mathrm{Aut}(Q_4)$, is equal to

$$\langle (0\ 2\ 1)(3\ 5\ 4)(6\ 8\ 7), (1\ 4\ 2\ 8)(3\ 7\ 6\ 5), (1\ 3\ 2\ 6)(4\ 5\ 8\ 7), (0\ 5\ 3\ 1)(2\ 7\ 4\ 8) \rangle$$

and it contains a normal elementary abelian subgroup N of order 9. Furthermore, $\mathrm{Aut}(Q_4)$ contains a subgroup H isomorphic to the quaternion groups. The order

	0	1	2	3	4	5	6	7	8
0	0	3	6	2	7	4	1	8	5
1	7	1	4	5	0	8	3	2	6
2	5	8	2	6	3	1	7	4	0
3	4	2	8	3	6	0	5	1	7
4	6	5	0	1	4	7	8	3	2
5	1	7	3	8	2	5	0	6	4
6	8	4	1	7	5	2	6	0	3
7	2	6	5	0	8	3	4	7	1
8	3	0	7	4	1	6	2	5	8

Table 6.5: Extremely nonassociative quasigroup Q_4 with $\mathbf{a}(Q_4)$=9, e=f= id.

	0	1	2	3	4	5	6	7	8
0	0	5	6	4	1	7	3	8	2
1	4	3	1	6	2	8	5	7	0
2	8	2	7	3	0	6	4	5	1
3	6	7	5	1	3	2	0	4	8
4	7	1	8	0	5	4	2	3	6
5	2	8	0	5	7	3	6	1	4
6	1	0	4	2	8	5	7	6	3
7	5	6	3	8	4	0	1	2	7
8	3	4	2	7	6	1	8	0	5

Table 6.6: Extremely nonassociative quasigroup Q_5 with $\mathbf{a}(Q_5)$=17, e=(1 4 7)(2 8 3)(5 6), f=(1 2)(3 4 5)(6 7 8), and Aut(Q_5)=$\langle id \rangle$.

	0	1	2	3	4	5	6	7	8
0	0	2	3	4	1	8	5	6	7
1	8	1	4	7	6	0	3	2	5
2	5	7	2	1	8	6	0	4	3
3	6	5	8	3	2	4	7	0	1
4	7	3	6	5	4	2	1	8	0
5	4	8	7	0	3	5	2	1	6
6	1	4	5	8	0	7	6	3	2
7	2	0	1	6	5	3	8	7	4
8	3	6	0	2	7	1	4	5	8

Table 6.7: Highly nonassociative quasigroup Q_6 with $\mathbf{a}(Q_6)$=17, e=f= id, and Aut(Q_6) isomorphic to quaternion group.

	(1)	**(2)**	**(3)**	**(4)**
Q_4	801	72	0	72
Q_5	225	27	0	1
Q_6	489	32	0	8

Table 6.8: Classification of extremal quasigroups of order 9.

of $\mathrm{Aut}(Q_4)$ is equal to 72. The group can be thus characterized as a semidirect product of N and H.

In the following sections we will focus on quasigroup Q_4 and its properties in order to extrapolate and find heuristics that might help in the search for bigger quasigroups with the number of associative triples equal to their order.

6.2.2 Operation decomposition

An interesting property of Q_4 is the quasigroup operation which can be fully decomposed into cycles. For example:

$$1 \cdot 2 = 4, 2 \cdot 4 = 3, 4 \cdot 3 = 1, \text{ and } 3 \cdot 1 = 2.$$

Call this cycle $(1, 2, 4, 3)$. With the exception of idempotents, all cycles are of length four and there is 18 of them in total. The list of all those cycles can be found in Appendix A1.

These cycles form a *directed 4-cycle system of order 9.*

Definition. A *directed m-cycle system of order n* is a pair (S, C), where C is an edge disjoint collection of directed m-cycles which partitions the edge set of D_n (the complete directed graph on n vertices) with set S.

In [12] we can find a quasigroup construction from these cycle systems called the **directed standard construction**.

Definition. Let (S, C) be a directed m-cycle system of order n and define a binary operation $\cdot$ on S by:

- $x \cdot x = x$, for all $x \in S$, and
- if $x \neq y$, $x \cdot y = z$ if and only if edges (x, y) and (y, z) form a directed path in C.

Observe, that Q_4 is constructed from cycles in Appendix A1 using directed standard construction.

In the same article a special property that holds for Q_4 is mentioned. For all $x \neq y$, $(xy)(y(xy)) = x$ holds.

This naturally leads to the idea of trying different cycles of four elements and constructing the quasigroups from them. In our experiments with this approach we have been able to create quasigroups isomorphic to Q_4 as well as idempotent quasigroups with 81 associative triples. One of those is shown in Appendix, Table A2.

This might also be a potential avenue for future research as these cycle constructions are also possible for higher orders. Unfortunately, we have not been able to successfully construct quasigroups with a small number of associative triples from cycle system in the higher orders.

Another interesting pattern emerges when we look at the cycles of decomposition of Q_4 as vertices of a graph labeled as in Appendix A1. If two cycles share an opposite edge, meaning $(a, b) \in C_1$ and $(b, a) \in C_2$, then connect them with an edge.

The resulting graph consists of two isomorphic components of size nine. In the classification of small graphs those components are known as $L(K_{3,3})$. Figure 6.1 shows the resulting graph.

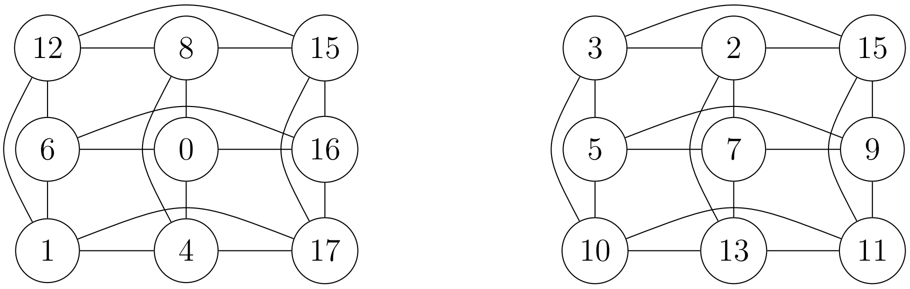


Figure 6.1: Graph constructed from cycles of decomposition of Q_4.

The graph $K_{3,3}$ is a complete bipartite graph as depicted below.

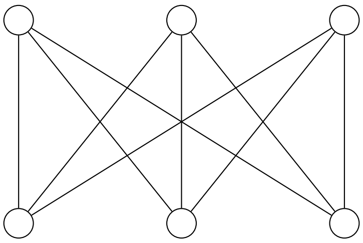


Figure 6.2

Its *line graph* $L(K_{3,3})$ named by Harary and Norman in [13] is defined below.

Definition. Given a undirected graph G, its *line graph* $L(G)$ is a graph such that each vertex of $L(G)$ represents an edge in G; and two vertices in $L(G)$ are connected with an edge if and only if their corresponding edges share a common vertex in G.

This further highlights the fact that quasigroup Q_4 is far from unstructured and possesses many interesting properties. More of them will be mentioned in the next section.

6.2.3 Sudoku property

The multiplication table of every quasigroup is a Latin square. A special case of Latin square of order 9 known as Sudoku has an additional property which can be generalized for Latin squares of order n^2.

Definition. A Latin square of order n^2 has a *Sudoku property* if any subsquare induced by rows $ni+1, \dots, ni+n$ and columns $nj+1, \dots, nj+n$ contains all values $1, \dots, n^2$, for any choice of i, j, where $0 \leq i, j < n$.

In this section we will consider Q_4 to be defined upon set $\{1, \dots 9\}$. It is easy to observe that the multiplication table of Q_4 has Sudoku property.

Since the algebraic counterparts of Latin squares are quasigroup we would like to express Sudoku property in the language of quasigroups.

Quasigroup $(Q, \cdot)$ defined upon set $N = \{1, \dots, n^2\}$ has Sudoku property if N can be partitioned into blocks $R_1, \dots, R_n$ of size n and into blocks $C_1, \dots, C_n$ of size n such that $\{x \cdot y; x \in R_i, y \in C_j\} = N$ for every $1 \leq i, j \leq n$. Standardly, blocks are chosen consecutively.

This property, however, lacks the symmetry between rows, column, and values in Latin squares that can be found in parastrophic quasigroups. To rectify this define *3-way Sudoku property*.

Definition. Let $(Q, \cdot)$ be a quasigroup defined upon set $N = \{1, \dots n^2\}$. Denote by R, C, and V partitions of N into n blocks of size n. Q has *3-way Sudoku property* if the following holds.

1. If $X \in R$, $Y \in C$, then $\{x \cdot y; x \in X, y \in Y\} = N$;
2. if $X \in R$, $Y \in V$, then $\{x \backslash y; x \in X, y \in Y\} = N$; and
3. if $X \in V$, $Y \in C$, then $\{x / y; x \in X, y \in Y\} = N$.

This 3-way condition for standard Sudoku square of order 9 means that the nine cells induced by a block of rows (say 1,2,3) and a block of symbols (say 4,5,6) cover all columns. In other words, $\{1, 2, 3\}$, $\{4, 5, 6\}$, and $\{7, 8, 9\}$ appear as transversals in each of the nine subsquares.

Observe that for quasigroup Q_4 this 3-way property holds as well.

Unfortunately, we were not successful in finding quasigroups of order 16 with this 3-way property and a small number of associative triples. However, the property can be weakened. For example, for order 12 we have found quasigroup where triples $(0, 1, 2)$, $(3, 4, 5)$, $(6, 7, 8)$, and $(9, 10, 11)$ always form a transversal if they appear in any of subsquares 3×3 formed by decomposition of rows and column into four blocks of three. This quasigroups also has 54 associative triples which is the lowest known number of associative triples for that order. Mentioned quasigroup can be found in Appendix, Table A3.

This shows that heuristics methods mentioned in this chapter can potentially lead to better understanding of quasigroups with a small number of associative triples.

6.2.4 Infinite series

Based on our results, it is easy to show that there is an infinite number of quasigroups with $\mathbf{a}(Q) = |Q|$.

Definition. For quasigroups $(Q, \cdot)$ and $(R, *)$ define their *direct product* to be quasigroup $(S, \circ)$ defined on $Q \times R$ by $(q_1, r_1) \circ (q_2, r_2) = (q_1 \cdot q_2, r_1 * r_2)$.

Lemma 6.1. *Let Q and R be quasigroups. Then $\boldsymbol{a}(Q \times R) = \boldsymbol{a}(Q)\boldsymbol{a}(R)$.*

Proof. Put $q_1, q_2, q_3 \in Q$, $r_1, r_2, r_3 \in R$ and observe that $((q_1, r_1), (q_2, r_2), (q_3, r_3))$ is an associative triple if and only if both (q_1, q_2, q_3) and (r_1, r_2, r_3) are associative triples. □

Corollary. There exists a sequence $\{Q_n\}_{n=1}^{\infty}$ of quasigroups such that the order of Q_n is 9^n and $\mathbf{a}(Q_n) = |Q_n|$ and can be constructed from quasigroup Q_4.

The existence of the infinite series of quasigroups mentioned above changes the view on the asymptotic behavior of $\mathbf{a}(n)$. In [5] it was suggested that the upper bound is at most n^2. The question of asymptotic behavior of $\mathbf{a}(n)$ is still open, however, these results suggest that it might be possible that $\mathbf{a}(n) = n$ for n big enough.

7. Implementation aspects

This chapter contains implementation details of the algorithm and lists different implementation improvements for increasing its speed. In previous chapters we introduced new approach on counting associative triples, based on mathematical results, as opposed to the more naive approach used in [5]. The improvements described in this chapter are not about mathematics but about implementation details. The magnitudes of speed improvements are compared in the next chapter.

The improvements in this chapter are separated into several groups based on the part of the algorithm they are aimed at. Those groups are:

- algorithm parallelization,
- quasigroup enumeration, and
- associative triples counting.

Throughout this chapter, we will consider quasigroups of order n defined upon the set $\{0, 1, \dots n-1\}$. We identify a quasigroup with its multiplication table represented as a two-dimensional array named Q by $Q[a, b] = a \cdot b$. We will use word *function* referring to a subroutine of a program that returns a value.

7.1 Algorithm parallelization

The parallelization has been crucial for the success of computations. The program ran on a cluster with 1000 nodes and we strived to maximize the utilization of those nodes during the time granted to us.

Call $S = \{(a_i, b_i, c_i); i{=}1, \dots, k\}$ a *starting configuration* of an algorithm if algorithm starts with partially constructed quasigroup Q for which $Q[a_i, b_i] = c_i$ for every $i{=}1, \dots, k$. The most natural exploitation of the parallelization is to consider different starting configurations at the same time.

Starting configurations are chosen in order to maximize the reduction of the search space. For orders 8 and 9 this is done upon the basis of the theory developed in Sections 3.2.1 and 3.2.2 of Chapter 3. They consist of filling out one of the mappings e, f, or diagonal mapping d in the quasigroup. However, the number of starting configurations might not be sufficient. For example, in the case of searching for highly nonassociative quasigroups of order 9 with 6 idempotents Lemma 1.4 gives 10 types of diagonal mappings to cover. The naive approach would be to run the computation on 10 CPUs. Since in our case 1000 nodes were available the following approach has been devised:

1. Denote by $\mathcal{S}$ the set of starting configurations consisting of predetermined mapping d.
2. Fix first unfilled cell $Q[x, y]$ and replace each $S \in \mathcal{S}$ by $S_j = S \cup \{(x, y, j)\}$, $j = 0, \dots n-1$ for all values j that keep the quasigroup properties.
3. Count nonelementary associative triples for time (x, y).
4. If $x = j$ or $y = j$ count elementary triples in that starting configuration.

5. Remove those starting configurations that exceed the threshold for maximum number of associative triples.

6. If $|\mathcal{S}|$ is smaller the desired number of starting configurations, go to 2.

7. Start the computation with the set of starting configurations $\mathcal{S}$.

It is easy to see that omitting steps 3, 4, and 5 may lead to the situation where starting configuration already contains more associative triples than the maximum allowed number and thus the algorithm terminates immediately. This decreases the number of CPUs that are active during computation.

Another point of view on this algorithm is to consider it as a generator of starting configurations that have a sufficiently small number of associative triples. Therefore, the implementation improvement mentioned above can be viewed as a multi-tier parallelization. In each level all resources are used to compute the starting configurations for the next level. The number of levels of parallelization depends on the overhead associated with redistributing starting configurations to the available CPUs.

7.2 Quasigroup enumeration

When enumerating quasigroups the crucial part is to be able to quickly determine which values to try for a given unfilled cell. For example, at the time (a, b) in the construction of the quasigroup (meaning that all the previous values are determined) the naive approach for determining valid value c to fill would look like a procedure below.

```
for $c = 0, \dots n-1$
    $valid$ = true
    for $col = 0, \dots b-1$
        if $Q[a, col] = c$
            $valid$ = false
    for $row = 0, \dots a-1$
        if $Q[row, a] = c$
            $valid$ = false
    if $valid$
        $Q[a, b] = c$
```

This approach, however, does not take into the consideration that starting configuration may contain filled values that are after the current position.

The better approach is to keep track of the values that are possible to fill into each row and column at each time. This can be accomplished using a two-dimensional array of boolean values where the first index determines row or column, the second index determines a value that we want to fill, and the value in the twodimensional array is **true** if it is possible. Since we are working with small orders we can use bits of integers instead of the second dimension in an array. Let *row* and *col* to be arrays of length n. Initialize them with zeros. Define functions `get`(x, i) that returns ith bit of integer x, function `seton`(x, i) that sets ith bit of integer x to be 1, and function `bmax`(x, y) that returns bitwise maximum of two integers.

At the start of the computation adjust the values in both arrays to reflect the starting configuration S.

```
for $(a, b, c) \in S$
    seton($row[a]$, $c$)
    seton($col[b]$, $c$)
```

The adjustment of both arrays is performed at every step of the computation. Observe that for row a the only valid values to fill are those c for which c-th bit of $row[a]$ is zero. It takes time and memory to keep and update the arrays, however, it still saves time overall when choosing the next value to fill with this procedure.

```
possible = bmax($row[a]$, $col[b]$)
for $c = 0, \dots n{-}1$
    if get($possible$, $c$) = 0
        $Q[a, b] = c$
        seton($row[a]$, $c$)
        seton($col[b]$, $c$)
```

Another implementation improvement that allows us to omit functions for counting nonelementary associative triples is based on the following lemmas.

Lemma 7.1. *Put $\mathcal{A}^\star_{(a,b)}$, $\mathcal{B}^\star_{(a,b)}$, $\mathcal{C}^\star_{(a,b)}$, and $\mathcal{D}^\star_{(a,b)}$ as in the Chapter 5. If $a = 0$, then $\mathcal{A}^\star_{(a,b)} = \mathcal{B}^\star_{(a,b)} = \mathcal{C}^\star_{(a,b)} = \mathcal{D}^\star_{(a,b)} = \emptyset$.*

Proof. Any associative triple (x, y, z) with decisive pair $(0, b)$ is necessary of type $(0, 0, z)$. The values x and y have to be 0 in order to compute values xy and yz. If $0 \cdot 0 \neq 0$ then $xy \cdot z$ cannot be computed thus $xy = 0$. The associative triple is elementary according to Lemma 4.3. □

Lemma 7.2. *Put $\mathcal{A}^\star_{(a,b)}$, $\mathcal{B}^\star_{(a,b)}$, $\mathcal{C}^\star_{(a,b)}$, and $\mathcal{D}^\star_{(a,b)}$ as in the Chapter 5. Assume that $a = 1$. Then $\mathcal{A}^\star_{(a,b)} = \emptyset$. If also $Q[1, 1] \neq 0$, then $\mathcal{B}^\star_{(a,b)} = \mathcal{D}^\star_{(a,b)} = \emptyset$. Lastly, if $Q[0, 0] \neq 1$, then $\mathcal{C}^\star_{(a,b)} = \emptyset$.*

Proof. Any associative triple $(x, y, z) \in \mathcal{A}^\star_{1,b}$ is of type $(1, b, z)$. Look at the procedure `countA`* in Chapter 5 and observe that $Q[1, b] = 0$. Also, if $b = 0$, then $\mathcal{A}^\star_{1,0} = \emptyset$. The last case to cover is therefore $b = 1$. This leads to $z = 0$ which would make the triple elementary.

Focus now on triple $(1, y, z)$ in $\mathcal{B}^\star_{1,b}$. If $b = 0$, then the procedure `countB`* yields no triples and if $b = 1$, then the triple is elementary. Assume that $b > a$. If $y = 0$, then $Q[1, 0]$ would have to be both less than one and not equal to zero in order to not be elementary. The only option is therefore triple of type $(1, 1, z)$ where $Q[1, 1] < 1$.

The enumeration of $\mathcal{C}^\star_{1,b}$ is simple and the set is not empty only if $x = y = 0$. By definition, $Q[0, 0]$ has to be equal 1.

Similar to the previous case, $\mathcal{D}^\star_{1,b}$ only contains triples of type $(1, 1, z)$. By necessity, $Q[1, 1]$ has to be zero otherwise $Q[Q[1, 1], z]$ could not be computed or the triple would be elementary. □

Implementation consequences of Lemmas 7.1 and 7.2 are following. Counting nonelementary triples in the first row can be omitted. In the second row, we can omit counting triples from $\mathcal{A}^{\star}_{1,b}$. Moreover, if $Q[1,1] \neq 0$, then we can skip counting triples from $\mathcal{B}^{\star}_{1,b}$ and $\mathcal{D}^{\star}_{1,b}$, and if $Q[0,0] \neq 1$, then there are no triples in the $\mathcal{C}^{\star}_{1,b}$.

The last implementation improvement in this section is based on the observation that values in the last row and the last column are fully determined by their predecessor. Thus, we can simplify and speed up the procedure that chooses the values to try out.

7.3 Associative triples counting

As mentioned in Chapter 5, when counting nonelementary associative triples it may be useful to keep the tables of left and right division as well. The memory overhead is minimal with quasigroups of small orders and the time saved is significant.

Another observation to make is that functions `countA`*, `countB`*, `countC`*, and `countD`* are usually not called separately. In fact, when the value c is filled in position a, b and $c \notin \{a, b, \}$, then all of them are called. If $c \in \{a, b\}$ then only `countB`* and `countC`* are called. By grouping those procedures into the same blocks of code several for-loops can be combined together.

The last improvement in this section focuses on the situation in the construction of the quasigroup when we achieved but not surpassed the threshold for the maximum number of nonelementary associative triples. In this situation, we are no longer interested in the number of nonelementary triples that arise when we fill the next value. The implementation improvement is to have two types of functions for counting nonelementary triples. The functions of **integer** type are those presented in Chapter 5. The other type is called **boolean** and it differs from the previous in its return value. Instead of returning the number of nonassociative triples it returns value **false** whenever it finds any associative triple. Therefore, it can break the loop and return the value faster. The algorithm thus starts in the **integer** mode using **integer** type functions and whenever the maximum allowed number of nonelementary associative triples is achieved it switches to the **boolean** mode until the quasigroup construction is completed or one of the functions returns **false**.

This concludes this chapter which details several implementation improvements that make the search more efficient. In the next chapter we are going to compare the speed of the current algorithm implementation against the algorithm that was used in [5] to determine the value of $\mathbf{a}(7)$. That algorithm used most of the implementation improvements mentioned in this chapter but still used the naive counting of associative triples.

8. Measuring the speed-up

This final chapter covers time comparison between implementations of different algorithms for finding nonassociative quasigroups. We use the algorithm from [5] as a reference point for comparisons. It will be tested against different versions of the algorithm presented in this thesis. Those versions are chosen to determine which aspects of optimization contribute the most to overall performance.

8.1 Algorithms

Throughout this chapter we will reference the implementations by letters **A** to **D**. All of these programs have been implemented in C++ language.

Program A serves as a reference. It is the implementation of the algorithm used in [5] to determine the value $\mathbf{a}(7)$.

Program B is an implementation of an algorithm described in Chapter 5 without any condensed procedures for counting nonelementary triples. It also lacks implementation optimizations mentioned in Sections 7.2 and 7.3.

Program C is similar to **Program B** but includes implementation optimizations from Sections 7.2 and 7.3. This implementation was used to determine $\mathbf{a}(8)$ and was also used in the search for $\mathbf{a}(9)$.

Program D is based on **Program C** but includes condensed procedures for counting nonelementary triples described in Section 5.2. This implementation was used in part to determine $\mathbf{a}(9)$.

8.2 Protocol

When comparing different programs we will be measuring the time of their execution on AMD A8-5600K CPU. A testing sample for each comparison is a set of 10 randomly generated starting configurations for quasigroups of order 8 with 24 cells filled. Tested program enumerates all quasigroups that contain set starting configuration and for which the number of associative triples does not surpass the predefined threshold. The upper limit on the number of associative triples is set to be 32. The comparison is based on mean, minimum, and maximum of execution times in seconds.

8.3 First comparison

Parametrization of **Program A** is possible only by filling whole rows of the multiplication table. Thus, starting configurations in the first comparison consist of filled out first three rows of the multiplication table. This does not allow programs **B**, **C**, and **D** to count elementary associative triples in advance since mappings e and f are not fully known. Therefore, elementary triples have to be counted during the enumeration of quasigroups. An example of starting configuration used in this comparison can be found in Table 8.1. The results of the testing are presented in Table 8.2.

	0	1	2	3	4	5	6	7
0	0	5	4	1	2	7	3	6
1	6	4	7	5	1	3	2	0
2	1	2	5	0	7	6	4	3
3	·	·	·	·	·	·	·	·
4	·	·	·	·	·	·	·	·
5	·	·	·	·	·	·	·	·
6	·	·	·	·	·	·	·	·
7	·	·	·	·	·	·	·	·

Table 8.1: One of the starting configuration for the first comparison.

Program	**Mean**	**Min**	**Max**
A	3039.47	2312.78	3817.46
B	88.30	21.99	133.39
C	16.36	4.22	23.68
D	15.88	3.95	23.07

Table 8.2: Results of the first comparison of implementations.

Table 8.2 shows approximately 35 fold speed increase of the **Program B** over the reference **Program A** . Moreover, programs **C** and **D** gained 5.5 fold time improvements over **Program B** . This highlights the importance of proper implementation of any algorithm. In our case, it is a result of an extended effort to optimize the program as much as possible by focusing not only on counting associative triples but also on the fast enumeration of quasigroups. That resulted in approximately 190 fold improvements over **Program A** .

Note that there is a little difference between **Program C** and **Program D** when both elementary and nonelementary triples are counted during the program execution.

8.4 Second comparison

The comparison below highlights the efficiency that can be gained from filling out mappings e and f in advance. This allows for elementary triples to be counted before enumeration. Therefore, during the enumeration only nonelementary triples are counted and their maximum allowed number is the difference between the threshold (32) and the number of elementary triples.

In this comparison 24 cells of the multiplication table are filled. It is the same number as in the previous example but we start by filling out cells that determine mappings e and f and continue by filling the multiplication table row by row starting from the top left corner until 24 cells are filled. An example of starting configuration used in this comparison can be found in Table 8.3.

	0	1	2	3	4	5	6	7
0	0	3	1	5	7	6	2	4
1	6	1	3	2	·	·	·	·
2	·	·	·	3	·	·	·	2
3	·	·	·	·	·	3	6	·
4	·	·	4	·	·	5	·	·
5	·	·	·	·	·	·	5	7
6	·	·	2	·	6	·	·	·
7	·	·	·	7	4	·	·	·

Table 8.3: One of the starting configuration for the second comparison.

Table 8.4 shows interesting results of the second comparison. Firstly, all implementations **B**, **C**, and **D** are on average 100 times faster compared to the first test despite the fact that the size of starting configurations is the same in both cases. Secondly, when counting only nonelementary triples we can see a bigger (32%) difference between **Program C** and **Program D** . This is achieved by modification of counting procedures described in Section 5.2.

Program	**Mean**	**Min**	**Max**
B	0.888	0.037	2.658
C	0.179	0.012	0.430
D	0.135	0.010	0.385

Table 8.4: Results of the second comparison of implementations.

From both comparison together we can see that an implementation of an algorithm presented in this thesis can be at least 35 times faster than our previous attempts. When combined with the observations from Chapters 1 and 3 and carefully choosing starting configurations it is possible to gain additional 2 orders of magnitude of speed improvements. Lastly, implementation aspects also play an important role as they contribute by additional 5 fold speed increase.

By combining these improvements we have been able to successfully conquer the search for highly nonassociative quasigroups of order 8 and extremely nonassociative quasigroups of order 9 with the same computational resources as in our previous work.

Conclusion

The main purpose of this thesis was to determine the minimum number of associative triples among quasigroups of orders eight and nine. We tried to achieve this goal with the same computational resources as we had available in [5], where we determined the minimum number of associative triples among quasigroups of orders up to seven. In order to accomplish this task, we had to improve our approach in several aspects.

Firstly, using combinatorics and properties of transformations and permutations we significantly reduced the search space. Secondly, we showed the distinction between two types of associative triples, called elementary and nonelementary. For elementary associative triples, we presented an estimate of their number and a strategy that brings the parallelization of search and gives the number of elementary associative triples at the same time. The strategy consists of filling out local unit mappings into multiplication table of quasigroup which fully determines the number of elementary triples. Parallelization is achieved by filling out different types of those mappings.

For nonelementary associative triples we devised an algorithm that counts their number incrementally at each step of the construction of multiplication table of the quasigroup.

By implementing and optimizing this algorithm we were able to accomplish our goal and establish that the minimum number of associative triples among quasigroups of order 8 is 16. We presented their classification and showed that extremal quasigroups of that order belong to two distinct main classes. This answers the question asked in [5] whether all extremal quasigroups belong to the same main class.

For order 9 the extremal case is even more interesting as we believe it is the first published example of quasigroup with the number of associative triples equal to its order, 9. This quasigroup has a quite large automorphism group from which we were able to construct Latin square with 3-way Sudoku property. We hope that described properties of that quasigroup, cycle decomposition, cycle construction, and 3-way Sudoku property might be useful in the future research in finding extremely nonassociative quasigroups of higher orders that are also squares.

Bibliography

[1] Aleš Drápal and Tomáš Kepka. A note on the number of associative triples in quasigroups isotopic to groups. *Commentationes Mathematicae Universitatis Carolinae*, 22(4):735–743, 1981.

[2] Jaroslav Ježek and Tomáš Kepka. Notes on the number of associative triples. *Acta Universitatis Carolinae. Mathematica et Physica*, 31(1):15–19, 1990.

[3] A. Kotzig and C. Reischer. Associativity index of finite quasigroups. *Glasnik Matematički*, 18:243–253, 1983.

[4] Otokar Grošek and Peter Horák. On quasigroups with few associative triples. *Designs, Codes and Cryptography*, 64(1):221–227, 2011.

[5] Viliam Valent. Quasigroups with few associative triples. *Bachelor thesis*, 2016.

[6] Aleš Drápal and Viliam Valent. Few associative triples, isotopisms and groups. *Designs, Codes and Cryptography*, 86(3):555–568, Mar 2018.

[7] Ivana Slaminková and Milan Vojvoda. Cryptoanalysis of a hash function based on isotopy of quasigroups. *Tatra Mt. Math*, 45(1):137–149, 2010.

[8] Milan Vojvoda. Cryptanalysis of one hash function based on quasigroup. *Tatra Mt. Math. Publ*, 29(173):173–181, 2004.

[9] J. Dvorský, E. Ochodková, and V. Snášel. Hash function based on quasigroups ("Hashovací funkce založená na kvazigrupách"). *Proceedings of Mikulášská kryptobesídka, Praha*, pages 27–36, 2001.

[10] Aleš Drápal and Viliam Valent. High nonassociativity in order 8 and an associative index estimate. *Submitted*, 2018.

[11] Charles J Colbourn and Jeffrey H Dinitz. *Handbook of combinatorial designs.* CRC press, 2006.

[12] Curt C Lindner. Quasigroups constructed from cycle systems. *Quasigroups and related systems*, 10:29–64, 2003.

[13] Frank Harary and Robert Z. Norman. Some properties of line digraphs. *Rendiconti del Circolo Matematico di Palermo*, 9(2):161–168, May 1960.

Appendix

i	cycle
0	$(0,1,3,5)$
1	$(0,2,6,7)$
2	$(0,3,2,8)$
3	$(0,4,7,3)$
4	$(0,5,4,2)$
5	$(0,6,1,4)$
6	$(0,7,8,1)$
7	$(0,8,5,6)$
8	$(1,2,4,3)$
9	$(1,5,8,4)$
10	$(1,6,3,7)$
11	$(1,7,2,5)$
12	$(1,8,6,2)$
13	$(2,3,6,5)$
14	$(2,7,4,8)$
15	$(3,4,6,8)$
16	$(3,8,7,5)$
17	$(4,5,7,6)$

Table A1: List of cycles in quasigroup Q_4.

	0	1	2	3	4	5	6	7	8
0	0	8	4	2	7	3	1	6	5
1	5	1	6	4	0	8	3	2	7
2	7	3	2	6	5	1	8	4	0
3	4	0	8	3	2	7	5	1	6
4	6	5	1	8	4	0	7	3	2
5	2	7	3	1	6	5	0	8	4
6	8	4	0	7	3	2	6	5	1
7	1	6	5	0	8	4	2	7	3
8	3	2	7	5	1	6	4	0	8

Table A2: A quasigroup reconstructed from different cycles but with the same structure as Q_4.

	0	1	2	3	4	5	6	7	8	9	10	11
0	0	6	3	7	5	11	2	8	9	10	1	4
1	4	1	7	9	8	3	10	0	6	5	11	2
2	8	5	2	4	10	6	7	11	1	0	3	9
3	1	10	6	3	11	2	0	4	7	8	9	5
4	7	2	11	0	4	9	8	1	5	3	6	10
5	9	8	0	10	1	5	3	6	2	11	4	7
6	2	11	5	8	0	10	6	9	3	4	7	1
7	3	0	9	11	6	1	4	7	10	2	5	8
8	10	4	1	2	9	7	11	5	8	6	0	3
9	11	3	8	5	7	0	1	10	4	9	2	6
10	6	9	4	1	3	8	5	2	11	7	10	0
11	5	7	10	6	2	4	9	3	0	1	8	11

Table A3: A quasigroup of size 12 with 54 associative triples and symmetries similar to Q_4.