Simula UiB Summer internship results

Håvard Skjetne Lilleheie

26/07/2023

1 Universal definitons

Definition 1.0.1. Let $[n] = \{0, 1, \dots, n-1\}$.

Definition 1.0.2. Let $\mathrm{Sym}(n) = \{\text{Permutations from } [n] \text{ to } [n]\}$.

2 Abelian group results

2.1 Initial definitions and results

Definition 2.1.1. Let $n \in \mathbb{N}$, then $\mathrm{Ab}(n) = \{\text{All Abelian Groups over the elements } [n]\}$.

Theorem 2.1.2. *Let $H \in \mathrm{Ab}(n)$, and let $\phi \in \mathrm{Sym}(n)$.*

Let $S = ([n], +_S)$ be a set over $[n]$ with a binary operation $+_S$ defined by

$$a +_S b = \phi(\phi^{-1}(a) +_H \phi^{-1}(b)) \tag{1}$$

Then $S \in \mathrm{Ab}(n)$.

Proof. Let $e \in H$ be the identity element.

Need to check every abelian group property:

Associativity:

$$\begin{aligned}
(a +_S b) +_S c &= \phi(\phi^{-1}(a +_S b) +_H \phi^{-1}(c)) \\
&= \phi(\phi^{-1} \circ \phi(\phi^{-1}(a) +_H \phi^{-1}(b)) +_H \phi^{-1}(c)) \\
&= \phi((\phi^{-1}(a) +_H \phi^{-1}(b)) +_H \phi^{-1}(c)) \\
&= \phi(\phi^{-1}(a) +_H (\phi^{-1}(b) +_H \phi^{-1}(c))) \\
&= \phi(\phi^{-1}(a) +_H \phi^{-1} \circ \phi(\phi^{-1}(b) +_H \phi^{-1}(c))) \\
&= \phi(\phi^{-1}(a) +_H \phi^{-1}(b +_S c)) \\
&= a +_S (b +_S c)
\end{aligned}$$

Identity element:

Let $\tilde{e} = \phi(e)$. Want to show that $\tilde{e}$ is the identity element.

$$\begin{aligned}
a +_S \tilde{e} &= \phi(\phi^{-1}(a) +_H \phi^{-1}(\tilde{e})) \\
&= \phi(\phi^{-1}(a) +_H \phi^{-1} \circ \phi(e)) \\
&= \phi(\phi^{-1}(a) +_H e) \\
&= \phi \circ \phi^{-1}(a) \\
&= a
\end{aligned}$$

And similarly on the left side.

Inverse element:

Let $\tilde{a} = \phi(-\phi^{-1}(a))$. Want to show that $\tilde{a}$ is the inverse of a.

$$\begin{aligned}
a +_S \tilde{a} &= \phi(\phi^{-1}(a) +_H \phi^{-1}(\tilde{a})) \\
&= \phi(\phi^{-1}(a) +_H \phi^{-1} \circ \phi(-\phi^{-1}(a))) \\
&= \phi(\phi^{-1}(a) -_H \phi^{-1}(a)) \\
&= \phi(e) \\
&= \tilde{e}
\end{aligned}$$

And similarly on the left side.

Commutative:

$$\begin{aligned}
a +_S b &= \phi(\phi^{-1}(a) +_H \phi^{-1}(b)) \\
&= \phi(\phi^{-1}(b) +_H \phi^{-1}(a)) \\
&= b +_S a
\end{aligned}$$

□

Remark 2.1.3. The equation in Equation 1 is equivalent to:

$$a +_H b = \phi^{-1}(\phi(a) +_S \phi(b)).$$

Definition 2.1.4. Let $H \in \mathrm{Ab}(n)$, and let $\phi \in \mathrm{Sym}(n)$.

Then the *conjugacy of H with respect to ϕ* is the abelian group S as defined in Theorem 2.1.2. And it is denoted by $\phi(H)$.

Theorem 2.1.5. *Let $H \in \mathrm{Ab}(n)$ and let $\phi \in \mathrm{Sym}(n)$.*

Then ϕ is an isomorphism from H to $\phi(H)$.

Proof. Look at

$$\phi(a +_H b) = \phi(\phi^{-1}(\phi(a) +_{\phi(H)} \phi(b))) = \phi(a) +_{\phi(H)} \phi(b)$$

Which is the definition of a homomorphism. And since ϕ is a bijection, it is therefore an isomorphism. □

Remark 2.1.6. For $H \in Ab(n)$, then the set of conjugacies of H, i.e.

$$\{\phi(H) : \phi \in \mathrm{Sym}(n)\}$$

is exactly the isomorphism class of H.

Theorem 2.1.7. *Let $\phi : H \to S$ be an isomorphism.*

Then $\lambda \in \mathrm{Aut}(H) \iff \phi \circ \lambda \circ \phi^{-1} \in \mathrm{Aut}(S)$.

Proof. Show first $\Rightarrow$:

Let $\lambda \in Aut(H)$. Then

$$\begin{aligned}\phi \circ \lambda \circ \phi^{-1}(a +_S b) &= \phi \circ \lambda(\phi^{-1}(a) +_H \phi^{-1}(b)) \\ &= \phi(\lambda \circ \phi^{-1}(a) +_H \lambda \circ \phi^{-1}(b)) \\ &= \phi \circ \lambda \circ \phi^{-1}(a) +_S \phi \circ \lambda \circ \phi^{-1}(b)\end{aligned}$$

Which is the definition of $\phi \circ \lambda \circ \phi^{-1} \in \mathrm{Aut}(S)$

Show then $\Leftarrow$:

Let $\phi \circ \lambda \circ \phi^{-1} \in \mathrm{Aut}(S)$, and let $\tilde{a} = \phi(a)$. Then

$$\begin{aligned}\phi \circ \lambda \circ \phi^{-1}(\tilde{a} +_S \tilde{b}) &= \phi \circ \lambda \circ \phi^{-1}(\tilde{a}) +_S \phi \circ \lambda \circ \phi^{-1}(\tilde{b}) \\ &\Updownarrow \\ \phi^{-1} \circ \phi \circ \lambda \circ \phi^{-1}(\tilde{a} +_S \tilde{b}) &= \phi^{-1}(\phi \circ \lambda \circ \phi^{-1}(\tilde{a}) +_S \phi \circ \lambda \circ \phi^{-1}(\tilde{b})) \\ \lambda \circ \phi^{-1}(\tilde{a} +_S \tilde{b}) &= \phi^{-1} \circ \phi \circ \lambda \circ \phi^{-1}(\tilde{a}) +_H \phi^{-1} \circ \phi \circ \lambda \circ \phi^{-1}(\tilde{b}) \\ \lambda(\phi^{-1}(\tilde{a}) +_H \phi^{-1}(\tilde{b})) &= \lambda \circ \phi^{-1}(\tilde{a}) +_H \lambda \circ \phi^{-1}(\tilde{b}) \\ \lambda(\phi^{-1}(\phi(a))) +_H \phi^{-1}(\phi(b)) &= \lambda \circ \phi^{-1}(\phi(a)) +_H \lambda \circ \phi^{-1}(\phi(b)) \\ \lambda(a +_H b) &= \lambda(a) +_H \lambda(b)\end{aligned}$$

□

Remark 2.1.8. From Theorem 2.1.5 and Theorem 2.1.7 one has that taking any Abelian group G over $[n]$ with identity element $e \in [n]$, and setting $\phi = (0, e)$, one gets that any automorphism over G can be "transformed" to an automorphism over $(i, 0)(G)$ where 0 is the identity, and vice versa. Therefore one only need to look at the automorphisms of abelian groups where 0 is a fixed identity to look for classifications of permutation as automorphisms.

Definition 2.1.9. Let $\mathrm{Ab}_0(n)$ denote the subset of $\mathrm{Ab}(n)$ where 0 is the identity of the abelian groups.

Theorem 2.1.10. $\frac{\#\,\mathrm{Ab}(n)}{\#\,\mathrm{Ab}_0(n)} = n$

Proof. In order to prove this, we need to add a temporary definition:

Let $\mathrm{Ab}_i(n)$ denote the subset of $\mathrm{Ab}(n)$ where $i \in [n]$ is the identity element.

Then we have to prove three different lemmas:

First lemma: $\mathrm{Ab}_i(n) \cap \mathrm{Ab}_j(n) = \emptyset$ for $i \neq j$.

Proof of first lemma: Assume $\mathrm{Ab}_i(n) \cap \mathrm{Ab}_j(n) \neq \emptyset$. Then there exists an abelian group $G \in \mathrm{Ab}_i(n) \cap \mathrm{Ab}_j(n)$ that has both i and j as an identity element. But then $i = i *_G j = j$, which is a contradiction, and $\mathrm{Ab}_i(n) \cap \mathrm{Ab}_j(n) = \emptyset$, since the identity must be unique.

Second lemma: $\mathrm{Ab}(n) = \cup_{i=0}^{n-1} \mathrm{Ab}_i(n)$

Proof of second lemma: First show $\subseteq$: Let $G \in \mathrm{Ab}(n)$. Then, by definition, G has an identity element, let's say $e \in [n]$. But then $G \in \mathrm{Ab}_e(n)$. Therefore $\mathrm{Ab}(n) \subseteq \cup_{i=0}^{n-1} \mathrm{Ab}_i(n)$.

Then show $\supseteq$: Every $\mathrm{Ab}_j(n) \subseteq \mathrm{Ab}(n)$ by definition. Therefore it follows that $\mathrm{Ab}(n) \supseteq \cup_{i=0}^{n-1} \mathrm{Ab}_i(n)$.

Third lemma: $\# \mathrm{Ab}_i(n) = \# \mathrm{Ab}_j(n)$ for all $i, j \in [n]$.

Proof of third lemma: Let $\phi = (i, j)$. Then one can define a map:

$$\begin{aligned} \varphi : \mathrm{Ab}_i(n) &\to \mathrm{Ab}_j(n) \\ G &\mapsto \phi(G) \end{aligned}$$

Want to show that this map is a bijection, and therefore preserves the cardinality of the sets.

First, let's show that φ is injective: Let $G, H \in \mathrm{Ab}_i(n)$ and let $\tilde{a} = \phi(a)$ and $\tilde{b} = \phi(b)$.

Then

$$\begin{gathered} \varphi(G) = \varphi(H) \\ \phi(G) = \phi(H) \\ \Downarrow \\ \tilde{a} +_{\phi(G)} \tilde{b} = \tilde{a} +_{\phi(H)} \tilde{b} \\ \phi(\phi^{-1}(\tilde{a}) +_G \phi^{-1}(\tilde{b})) = \phi(\phi^{-1}(\tilde{a}) +_H \phi^{-1}(\tilde{b})) \\ \Updownarrow \\ \phi^{-1}(\phi(\phi^{-1}(\tilde{a}) +_G \phi^{-1}(\tilde{b}))) = \phi^{-1}(\phi(\phi^{-1}(\tilde{a}) +_H \phi^{-1}(\tilde{b}))) \\ \phi^{-1}(\tilde{a}) +_G \phi^{-1}(\tilde{b}) = \phi^{-1}(\tilde{a}) +_H \phi^{-1}(\tilde{b}) \\ a +_G b = a +_H b \end{gathered}$$

So G and H are groups with the same group operation over the exact same set. They are therefore equal, i.e. $G = H$.

Secondly, lets show that φ is surjective: Let $K \in \mathrm{Ab}_j(n)$. Then want to show that $\varphi(\phi^{-1}(K)) = \phi(\phi^{-1}(K)) = K$.

First, note that $a +_{\phi^{-1}(K)} b = \phi^{-1}(\phi(a) +_K \phi(b))$. Then

$$\begin{aligned} a +_{\phi(\phi^{-1}(K))} b &= \phi(\phi^{-1}(a) +_{\phi^{-1}(K)} \phi^{-1}(b)) \\ &= \phi(\phi^{-1}(\phi(\phi^{-1}(a)) +_K \phi(\phi^{-1}(b)))) \\ &= \phi(\phi^{-1}(a)) +_K \phi(\phi^{-1}(b)) \\ &= a +_K b \end{aligned}$$

So K and $\phi(\phi^{-1}(K))$ are groups with the same group operation over the exact same set. They are therefore equal, i.e. $K = \phi(\phi^{-1}(K))$.

So for the proof of the theorem: The first and the second lemma gives that $\mathrm{Ab}(n)$ is a disjoint union of $\{\mathrm{Ab}_i(n) : i \in [n]\}$. Therefore $\# \mathrm{Ab}(n) = \sum_{i=0}^{n-1} \# \mathrm{Ab}_i(n)$. But the third lemma says that $\mathrm{Ab}_i(n) = \mathrm{Ab}_0(n)$ for all i. Therefore $\# \mathrm{Ab}(n) = \sum_{i=0}^{n-1} \# \mathrm{Ab}_0(n) = n \# \mathrm{Ab}_0(n)$ which implies $\frac{\# \mathrm{Ab}(n)}{\# \mathrm{Ab}_0(n)} = n$. $\square$

Theorem 2.1.11. *Let $H \in \mathrm{Ab}(n)$.*

Then for every $a \in H$, the map $\varphi_a : x \mapsto x +_H a$ is a bijection.

Proof. Assume there exists an $a \in H$ such that φ_a is not a bijection.

Since $\varphi_a : H \to H$, and H has finite cardinality. Then φ_a is not injective. This implies that there exist to values $x \neq y$ such that $x +_H a = y +_H a$. But using the latter equation and adding $-a$ to both sides, one gets that $x = y$ which is a contradiction. □

Remark 2.1.12. The property in Theorem 2.1.11 is the *latin square property*, which will be further studied later.

2.2 Results and counterexamples

Theorem 2.2.1. *Let S be a set with a closed binary operation $*_S$ that is*

- *associative, and*
- *commutative, and*
- *where for any $a \in S$, the map $x \mapsto x *_S a$ is a bijection,*

has an identity element. And is therefore an Abelian group.

Early proof. Let a be any element in the group. Since $a + x$ is a bijection, there exist a y such that $a + y = a$. Want to prove that this y is in fact an identity element for the binary operation.

Assume $b + y = z$ for some z. adding a to both sides gives an equivalent equation: $a + (b + y) = a + z$. Using associativity, this becomes: $(a + b) + y = a + z$. Then from commutativity one gets $(b + a) + y = a + z$. Using associativity again, one gets: $b + (a + x) = a + z$. But since $a + y = a$ by assumption, this reduces to: $b + a = a + z$. But since the operation $a + x$ is a bijection, it is then injective, and hence $z = b$, which implies $b + y = b$. Since the choice of b was arbitrary, y is the identity element. □

Late proof. Follows from Theorem 3.1.4. □

Remark 2.2.2. For every $\phi \in \mathrm{Sym}(n)$, and for every $H \in \mathrm{Ab}(n)$, if $a = b +_H b$, then $\phi(a) = \phi(b) +_{\phi(H)} \phi(b)$. This implies that any element on the diagonal of the group operation table stays on the diagonal of the group operation table, but permuted.

For example, consider the case when $\phi(0) = 0$, then one has that the conjugate must have the exact same amount of zeros on the diagonal.

Counterexample 2.2.3. Let $\phi \in \mathrm{Sym}(n)$ with $\phi(0) = 0$.

Then there *does not necessarily* exist a $H \in \mathrm{Ab}_0(n)$ such that $\phi \in \mathrm{Aut}(H)$.

Special case proof. Let $\phi = (3, 4)$, there want to show that there are no groups H in $\mathrm{Ab}_0(5)$ where $\phi \in \mathrm{Aut}(H)$ with a proof by contradiction.

Assume ϕ is an automorphism for $H \in G_0(5)$.

Then $\phi(1 +_H 1) = \phi(1) +_H \phi(1) = 1 +_H 1$, and $\phi(2 +_H 2) = \phi(2) +_H \phi(2)$, and $\phi(1 +_H 2) = \phi(1) +_H \phi(2) = 1 +_H 2$. This means that $1 +_H 1, 2 +_H 2, 1 +_H 2 \in \{0, 1, 2\}$, since they are fixed points of the map ϕ. And the set $\{0, 1, 2\}$ is therefore closed under the group operation.

Furthermore, since $1 +_H 2$ can not be 1 or 2, since that would imply that either 1 or 2 is an identity, then $1 +_H 2 = 0$, and the set $\{0, 1, 2\}$ is therefore a subgroup of H, since every element in $\{0, 1, 2\}$ has an inverse. But H has order 5, which 3 does not divide which contradicts Lagrange's theorem, and therefore ϕ can not be an automorphism. $\square$

General case proof. Let $n \geq 5$, and let $\phi \in \mathrm{Sym}(n)$ with $\phi = (n-2, n-1)$. Want to do a proof by contradiction to show that there are no groups $H \in \mathrm{Ab}_0(n)$ such that $\phi \in \mathrm{Aut}(H)$.

Assume $\phi \in \mathrm{Aut}(H)$.

Then for $i, j \in [n-3]$ one has that $\phi(i +_H j) = \phi(i) +_H \phi(j) = i +_H j$. So $i +_H j \in [n-3]$ since it is a fixed point of ϕ. But then one has that the top left $(n-2) \times (n-2)$ square of the group operation table of H only contains elements in $[n-3]$. One has by the latin square property (Theorem 2.1.11) that the remaining two elements, $n-2$ and $n-1$ has to appear once for every row of the top right rectangle with the shape $(n-2) \times (2)$, since the elements doesn't appear in any of the rows in the top left $(n-2) \times (n-2)$ square of the group operation table. However, since $n \geq 5$, the top right rectangle has a height of at least 3, which is greater than it's width. One would therefore have one column where the same element would appear at least twice, which also breaks the latin square property. Therefore $\phi \notin \mathrm{Aut}(H)$. $\square$

Counterexample 2.2.4. Let $\phi \in \mathrm{Sym}(n)$.

Then there *does not neccesarily* exist a $H \in Ab(n)$ with $a \in H$ such that the map $x \mapsto x +_H a$ is equal to ϕ.

Simple proof. Let $\phi(0) = 0$, but not be equal to the identity map. Then if $x \mapsto x +_H a = a +_H x$ is equal to ϕ, it would imply that there is a row in the group operation table that is $(0, \phi(1), \phi(2), \dots, \phi(n-1))$. However since this is not the identity row, it means there must be another row in the group operation table that is $(0, 1, 2, \dots, n-1)$. But then one has two rows that start with 0, which breaks the latin square property of H. $\square$

Later proof. This is implied by Counterexample 2.2.6. $\square$

Definition 2.2.5. For $H \in \mathrm{Ab}(n)$,

then let $\mathrm{AAut}(H) := \{\phi(\_) +_H v : \phi \in \mathrm{Aut}(H) \wedge v \in H\}$.

Counterexample 2.2.6. Let $\phi \in \mathrm{Sym}(n)$.

Then there *does not necessarily* exist an $H \in \mathrm{Ab}(n)$ such that $\phi \in \mathrm{AAut}(H)$.

Proof. Let $\phi = (3, 4)$. Want to show that ϕ is not an affine automorphism for any group H in $\mathrm{Ab}(5)$.

We will do a proof by contradiction by assuming that $\phi \in \mathrm{AAut}(H)$, and getting a contradiction. We have to look at two different cases to prove this. First when the identity is $0, 1$, or 2. And then secondly when the identity is either 3 or 4.

Assume that $\phi \in \mathrm{AAut}(H)$. Denote the identity of H by "e". Furthermore let $\alpha \in \mathrm{Aut}(H)$ and $\tilde{v} \in H$ be the elements such that $\phi(\_) = \alpha(\_) +_H \tilde{v}$.

Let $v = -\tilde{v}$.

This gives that $\alpha(\_) = \phi(\_) +_H (-\tilde{v}) = \phi(\_) +_H v$.

Then one has that
$$e = \alpha(e) = \phi(e) +_H v.$$
This gives us that $v = -\phi(e)$.

Case 1: Assume that $e \in \{0, 1, 2\}$.

Then one has that $v = -e = e$. So then $\phi \in \mathrm{Aut}(H)$ by the assumption.

This gives us that $\{0, 1, 2\}$ is a subgroup of H, since for $a, b \in \{0, 1, 2\}$ then $\phi(a +_H b) = \phi(a) +_H \phi(b) = a +_H b$, which makes $a +_H b$ a fixed-point of ϕ, and therefore $a +_H b \in \{0, 1, 2\}$. Furthermore $0 +_H 1 +_H 2 = e$ since $e \in \{0, 1, 2\}$, so the identity element can be removed from the sum, and the elements left have to sum to the identity or else there would be two different identities, which is not possible in a group. Therefore every element has it's inverse in $\{0, 1, 2\}$. Along with associativity from H, then the set $\{0, 1, 2\}$ is a subgroup of order 3.

This is *not possible* since every subgroup of H, which has order 5, must divide it's order by Lagrange's theorem.

Case 2: Assume that $e \in \{3, 4\}$. Without loss of generality assume $e = 3$.

Then one has that $v = -\phi(3) = -4$. And specifically one has that
$$\alpha(4) = \phi(4) - 4 = 3 -_H 4 = -4$$
And by the homomorphism property, $\alpha(-4) = -\alpha(4) = -(-4) = 4$

In general notice that
$$\alpha(i) = \begin{cases} i -_H, 4 & i \in \{0, 1, 2\} \\ 3, & i = 3 \\ -4, & i = 4 \end{cases}$$
This gives the identity: $\alpha(\alpha(i)) = i$. So α is it's own inverse, and therefore it must be a composition of zero, one or two disjoint transmutations.

We further split this up into two cases:

Case 2.1: Assume α has a fixed-point besides 3.

Then by the fact that α is the composition of zero, one or two disjoint transmutations, it must be either identity or one transmutation. This implies there is at least one fixed-point among $\{0, 1, 2\}$. Without loss of generality, assume this fixed element is 0. Then $0 = \alpha(0) = 0 -_H 4$. But subtracting 0 on either side, on gets that $3 = -4$, which is not true, since then $3 = 4 -_H 4 = 4 +_H 3 = 4$, which is not true.

Case 2.2: Assume α has no fixed-points beside 3. Without loss of generality, assume $\alpha = (0, 4)(1, 2)$

Then
$$1 +_H 2 = 1 -_H 4 +_H 2 -_H 4 +_H 4 +_H 4 = \alpha(1) + \alpha(2) +_H 4 +_H 4 = 2 +_H 1 +_H 4 +_H 4.$$
But by adding $-(1 +_H 2)$ to both sides, one gets: $4 +_H 4 = 3$. However, this implies $4 = -4 = \alpha(4) = 0$, which is not true. $\square$

Remark 2.2.7. The above proof could most likely be rewritten to not depend on commutativity, but I have not looked into the details ("Case 2.2" may pose some issues). Should work since every group is abelian for $n = 5$ by the classification of finite groups.

But could it be extended to disprove any loops (Definition 3.2.2)? If so, that would be a big result, since it would prove that Construction 3.4.1 is the best general construction for every permutaton.

3 LatinSquare/Quasigroup

3.1 Initial definitions

Definition 3.1.1. Let Q be a set with a binary operation $\_ *_Q \_$ with the following properties:

1. $\_ *_Q \_$ is closed.
2. For every $a \in Q$, the map $x \mapsto a *_Q x$ is a bijection.
3. For every $a \in Q$, the map $x \mapsto x *_Q a$ is a bijection.

Remark 3.1.2. Item number 2 and 3 in the requirements of a quasigroup are called the "latin square property" because these rules are equivalent to that the binary operation table is a latin square, as shown in Theorem 3.1.5.

Definition 3.1.3. Let $\mathrm{Quas}(n)$ denote the set of all quasigroups over $[n]$.

Theorem 3.1.4. *An associative quasigroup is a group.*

For a proof, see anassociativequasigroupisagroup.

Theorem 3.1.5. *Any quasigroup in* $\mathrm{Quas}(n)$ *can be represented as an* $n \times n$ *latin square by their binary operation table.*

Proof. First show $\mathrm{Quas}(n) \to n \times n$ latin squares:

Assume that the binary operation table is not a latin square. Then there are two cases:

Case 1: There is a duplicate element in a row, say row a has element x occur twice. Once for column b and another time for a different column c. This implies that $a * b = x = a * c$. But then the map $\varphi : x \mapsto a * x$ is not injective, since $\varphi(b) = \varphi(c)$, but $b \neq c$.

Case 2: There is a duplicate element in a column. Then the argument is the same, but using the map $x \mapsto x * a$ instead.

Then show $n \times n$ latin squares $\to \mathrm{Quas}(n)$:

Given an $n \times n$ latin square, look at the binary operation induced by the latin square. It satisfies all the quasigroup properties, and therefore the set $[n]$ with this binary operation is a quasigroup. □

3.2 Counterexamples

Counterexample 3.2.1. A commutative quasigroup is not an abelian group.

Proof. Look at the quasigroup given by this binary operation table:

$$\begin{matrix} 2 & 1 & 0 \\ 1 & 0 & 2 \\ 0 & 2 & 1 \end{matrix}$$

But then $(0 + 0) + 1 = 2 + 1 = 2$, however $0 + (0 + 1) = 0 + 1 = 1$. □

It is also implied by Counterexample 3.2.3.

Definition 3.2.2. A loop is a quasigroup with a (unique) two-sided identity element.

Counterexample 3.2.3. A commutative loop is not an abelian group.

Proof. Look at the loop given by the binary operation table:

0	1	2	3	4	5
1	5	4	2	3	0
2	4	5	1	0	3
3	2	1	0	5	4
4	3	0	5	2	1
5	0	3	4	1	2

But then $(1+1)+4 = 5+4 = 1$, however $1+(1+4) = 1+3 = 2$. $\square$

3.3 Intermediary results

Theorem 3.3.1. *Let $H \in \mathrm{Quas}(n)$ and let $\phi \in \mathrm{Sym}(n)$.*

Then let $S = ([n], *_S)$ be a set with a binary operation where $*_S$ is given by:

$$a *_S b = \phi(\phi^{-1}(a) *_H \phi^{-1}(b)).$$

Then S is a quasigroup.

Proof. Taking the conjugacy of a quasigroup is the same as permuting the rows and columns of the binary operation table by ϕ and then permuting the elements by ϕ. And since permuting the rows or permuting the columns of the binary operation table still gives a latin square, as well as permuting the elements still keeps the latin square structure, one gets that the resulting binary operation table still is a latin square. Therefore, S is a quasigroup. $\square$

Definition 3.3.2. Let $\phi \in \mathrm{Sym}(n)$, and let $Q \in \mathrm{Quas}(n)$.

Then the *conjugacy of Q with respect to ϕ* is the quasigroup S from Theorem 3.3.1. This S is denoted as $\phi(Q)$.

Remark 3.3.3. The definition in Definition 3.3.2 reflects the definition for abelian groups in Definition 2.1.4, but for quasigroups.

Remark 3.3.4. Similar to how it is for abelian groups, Theorem 2.1.5 is also true for quasigroups by a similar proof.

This implies that one can not gain or lose any significant structure by taking conjugacy. E.g. the conjugacy preserves exactly commutativity, associativity and left/right identity.

Theorem 3.3.5. *Let $\phi \in \mathrm{Sym}(n)$, and let $Q \in \mathrm{Quas}(n)$.*

Then $\phi \in \mathrm{Aut}(Q) \iff \phi(Q) = Q$.

Proof. First look at $\Rightarrow$.

Assume $\phi \in \mathrm{Aut}(Q)$. Then $\phi(a *_Q b) = \phi(a) *_Q \phi(b)$. Look at $a *_{\phi(Q)} b = \phi(\phi^{-1}(a) *_Q \phi^{-1}(b)) = \phi(\phi^{-1}(a)) *_Q \phi(\phi^{-1}(b)) = a *_Q b$. So in fact, the binary operation table of Q and $\phi(Q)$ are identical.

Secondly, look at $\Leftarrow$.

Assume $\phi(Q) = Q$. Then $a *_Q b = a *_{\phi(Q)} b$. Look at $\phi(a) *_Q \phi(b) = \phi(a) *_{\phi(Q)} \phi(b) = \phi(\phi^{-1}(\phi(a)) *_Q \phi^{-1}(\phi(b))) = \phi(a *_Q b)$, which is the definition for $\phi \in \mathrm{Aut}(Q)$. □

Lemma 3.3.6. *Let $\alpha, \beta, \gamma \in Sym(n)$, and let Q be any quasigroup over $[n]$.*

Then there exist a unique quasigroup, denoted $(\alpha, \beta, \gamma)(Q)$, such that (α, β, γ) is an isotopy from Q to $(\alpha, \beta, \gamma)(Q)$.

Proof. Let $\tilde{Q} = \big([n], *_{\tilde{Q}}\big)$. Where the binary operation $*_{\tilde{Q}}$ is given by:

$$a *_{\tilde{Q}} b = \gamma(\alpha^{-1}(a) +_Q \beta^{-1}(b)).$$

This binary operation table corresponds to the binary operation table of Q, but where the rows are permuted by α, the columns are permuted by β, and the elements are permuted by γ. Permuting rows, columns or elements by any permutation still preserves the latin square property, so therefore $\tilde{Q}$ is still a quasigroup. □

Definition 3.3.7. Let $\alpha, \beta, \gamma \in Sym(n)$, and let Q be any quasigroup over $[n]$.

Then the unique quasigroup denoted $(\alpha, \beta, \gamma)(Q)$ from Lemma 3.3.6 is called the *isotopy image of Q with respect to* (α, β, γ).

Remark 3.3.8. The definition in Definition 3.3.7 is a more general version of the Definition 3.3.2, since given a quasigroup $Q \in \mathrm{Quas}(n)$ and a permutation $\phi \in \mathrm{Sym}(n)$ the conjugacy of Q with respect to ϕ is the same as the isotopy image of Q with respect to (ϕ, ϕ, ϕ).

Definition 3.3.9. Let $\phi \in \mathrm{Sym}(n)$.

If there exist a $\lambda \in Aut(Q)$ and $v \in Q$, such that at least one of the following is true:

- $\phi(\_) = \lambda(\_)$, or
- $\phi(\_) = \lambda(\_) *_Q v$, or
- $\phi(\_) = v *_Q \lambda(\_)$.

Then $\phi \in \mathrm{AAut}(Q)$.

Remark 3.3.10. This definition is very similar to the definition for abelian groups shown previously (Definition 2.2.5), but not assuming commutativity or even an identity element.

3.4 Construction

Construction 3.4.1. *For $\phi, \pi \in \mathrm{Sym}(n)$.*

Then for any given quasigroup $Q \in \mathrm{Quas}(n)$ with left identity. One can construct another quasigroup $\tilde{Q} = (\pi^{-1}, \phi^{-1}, \mathrm{Id})(Q)$ such that $\phi \in \mathrm{AAut}(\tilde{Q})$.

Proof. Let the left identity element of Q be e_L.

Then from the isometry property $\pi^{-1}(x) *_{\tilde{Q}} \phi^{-1}(y) = Id(x *_Q y)$, one gets:

$$\pi(x) *_Q \phi(y) = Id(\pi(x) *_Q \varphi(y)) = \pi^{-1}(\pi(x)) *_{\tilde{Q}} \varphi^{-1}(\varphi(y)) = x *_{\tilde{Q}} y$$

Using $x *_{\tilde{Q}} y = \pi(x) *_Q \phi(y)$ and taking $x = \pi^{-1}(e_L)$, one gets: $\pi^{-1}(e_L) *_{\tilde{Q}} y = e_L *_Q \phi(y) = \phi(y)$. Written another way: $\phi(y) = \pi^{-1}(e_L) *_Q \mathrm{Id}(y)$, and hence $\phi(y) \in \mathrm{AAut}(Q)$, since $\mathrm{Id} \in \mathrm{Aut}(\tilde{Q})$ and $\pi^{-1}(e) \in \tilde{Q}$. □

Remark 3.4.2. Using the construction in Construction 3.4.1 with Q as a loop with identity element e and $\pi = \mathrm{Id}$, then $\tilde{Q}$ will have a right identity, namely $\phi^{-1}(e)$, since $x *_{\tilde{Q}} \phi^{-1}(e) = \mathrm{Id}(x) *_Q e = x$ for any $x \in \tilde{Q}$.

Lemma 3.4.3. *Let $\phi \in Sym(n)$, and let $Q \in \mathrm{Quas}(n)$ be a commutative quasigroup.*

Then the quasigroup $\tilde{Q} = (\varphi, \varphi, Id)(Q)$ is also commutative.

Proof. From the isotopy property $x *_Q y = \phi^{-1}(x) *_{\tilde{Q}} \phi^{-1}(y)$.

Therefore one gets $x *_{\tilde{Q}} y = \phi(x) *_Q \phi(y) = \phi(y) *_Q \phi(x) = y *_{\tilde{Q}} x$. □

Corollary 3.4.4. *Let $\phi \in \mathrm{Sym}(n)$, and let Q be a commutative loop.*

Then using the construction in Construction 3.4.1, but setting $\pi = \phi$, one gets that $\tilde{Q} = (\phi^{-1}, \phi^{-1}, \mathrm{Id})\,(Q)$ is also commutative.

Proof. From Lemma 3.4.3, one gets that $\tilde{Q}$ is commutative. □

Theorem 3.4.5. *Let $Q \in \mathrm{Quas}(n)$ with left identity, and let $\phi \in \mathrm{Sym}(n) \setminus \{\mathrm{Id}\}$ have a fixed point.*

Then for any $\pi \in \mathrm{Sym}(n)$, every isotopy image (Definition 3.3.7) of (π, ϕ, Id) of Q does not *have a left (right) identity. In particular, it is* not *a loop.*

Proof. Let $\tilde{Q} = (\pi, \phi, \mathrm{Id})(Q)$ denote the isotopy image, and let $e_L \in Q$ be the left identity of Q.

Let a be a fixed point of ϕ from the assumption, and let b be a non-fixed point of ϕ (we know it must exist since by assumption $\phi \neq \mathrm{Id}$).

The isotopy property says that $\pi(x) *_{\tilde{Q}} \phi(y) = x *_Q y$, or equivalently, $x *_{\tilde{Q}} y = \pi^{-1}(x) *_Q \phi^{-1}(y)$.

Then $\pi(e_L) *_{\tilde{Q}} a = e_L *_Q \phi^{-1}(a) = e_L *_Q a = a$, and from the latin square property, one has that $\pi(e_L)$ has to be the unique left identity of $\tilde{Q}$ if it exists. However, $\pi(e_L) *_{\tilde{Q}} b = e_L *_Q \phi^{-1}(b) = \phi^{-1}(b) \neq b$, so $\pi(e_L)$ can not be a left identity, and $\tilde{Q}$ therefore has no left identity. □

Remark 3.4.6. By Theorem 3.4.5 it follows that the construction in Construction 3.4.1 can never create a loop, since it can not have a left identity.

This implies that if one used the construction in Corollary 3.4.4 in order to get a symmetric quasigroup, it can not have a right identity at the same time since then it would be a loop.

Furthermore, by Theorem 3.1.4 it follows that one can never get associativity for $\tilde{Q}$, since that would imply it's a loop.

Remark 3.4.7. In fact, a weaker version of Theorem 3.4.5 can be seen all the way back in Counterexample 2.2.4, where by counterexample it is shown that there are permutations where the construction in Construction 3.4.1 would never preserve the abelian group structure.

Remark 3.4.8. For every theorem and construction where we've assumed a left identity there exist a similar theorem/constrction where one assumes right identity.

4 Crypto attack ideas

I work with this security scheme: There is a Challenger that chooses a random permutation $\phi \in \mathrm{Sym}(n)$. Then the Adversary can ask for a "reasonable" amount of $\phi(x_i)$ for some given x_i. Then afterwards, the adversary can ask for a "reasonable" amount of $\phi^{-1}(y_i)$ for some given y_i. In the end, the Adversary sends a tuple $(c, \tilde{c})$. The adversary wins if $c \neq x_i \forall i$ and $\tilde{c} \neq y_i \forall i$ and $\tilde{c} = \phi(c)$.

The idea is that given a $\phi \in \mathrm{Sym}(n)$ using Construction 3.4.1, with either Remark 3.4.2 or Corollary 3.4.4 to construct a quasigroup $\tilde{Q}$, where $\phi \in \mathrm{AAut}(\tilde{Q})$. Given the current construction, it is not possible to do better due to Remark 3.4.6, so that's what we're working with.

Then using knowledge of the $Q \in \mathrm{Quas}(n)$ that is used in the construction along with symmetry or right identity, it might be possible to get knowledge of the binary operation table of $\tilde{Q}$, since that might be used in an attack. However, it turns out that the construction always gives us a known value of v in $\phi(x) = v +_{\tilde{Q}} x$, but that doesnt help out, because the only useful values in the binary operation table of $\tilde{Q}$ turns out to be those on row v, since those "model" the behaviour of $\phi(x)$ that we're looking for.

If one chooses the construction such that $\tilde{Q}$ has 0 as a right identity (choose Q to be a commutative loop with identity 0). Then one immediately knows that there is a column in the binary operation table that is equal to $(0, 1, \ldots, n-1)$. However, one only knows the column number by asking for $\phi^{-1}(0)$, which is the only useful knowldge one could have gained from the colum, which ruins the purpose.

If one choses the construction such that $\tilde{Q}$ is commutative, then for every value one checks in the binary operation table of $\tilde{Q}$, one gets two rows and two columns worth of latin square knowledge due to the symmetric binary operation table since it's commutative (if it's not on the diagonal). However, one still has to check at least $\frac{n}{2}$ different values in the table to conclude a single non-checked value of the "important" $\phi^{-1}(e)$ row, which is still too much.

If one choses the construction such that $\tilde{Q}$ is commutative and Q is the XOR abelian group table, one gets that the diagonal on the binary operation table of $\tilde{Q}$ is all zeros. However, again this is all useless since $v = \phi^{-1}(0)$, so one has already asked for the only useful information one would have gained.

I started to continue building on the idea of using a commutative construction from XOR, and trying to use the basis that XOR has as a vector space over $\mathbb{Z}_2$, but I didn't manage to find out anything, and it all seemed to be a dead end, since $\tilde{Q}$ is not associative, which makes it very hard to work with.

5 Questions moving forward

5.1 Improvements

Idea 5.1.1. The current construction as defined in Construction 3.4.1 is limited in that one can only say that for any $\phi \in \mathrm{Sym}(n)$ there is a $\tilde{Q}$ with left identity or a $\tilde{Q}$ that is commutative such that $\phi \in \mathrm{AAut}(\tilde{Q})$. One can not have both at the same time due to Remark 3.4.6.

However, the current construction is limited to looking for affine automorphisms on the form $\phi(x) = \mathrm{Id}(x) *_{\tilde{Q}} a$ for some $a \in \tilde{Q}$. If it could be extended to find affine automorphisms on the form $\phi(x) = \alpha(x) *_{\tilde{Q}} a$ for some $\alpha \in \mathrm{Aut}(\tilde{Q})$, then that would be a big improvement and could possibly avoid the consequences mentioned in Remark 3.4.6.

As a summary, here is the current limitations that have been found:

For $n = 5$, for any $\phi = (3, 4)$, it is *not possible* to find a $\tilde{Q}$ that is a group or better, such that $\phi \in \mathrm{AAut}(\tilde{Q})$, by Counterexample 2.2.6.

For any n, for any $\phi \in \mathrm{Sym}(n)$, *using the current construction*, it is *not possible* to find any $\tilde{Q}$ that is a loop or better, such that $\phi \in \mathrm{AAut}(\tilde{Q})$, by Remark 3.4.6.

Therefore I conjecture these two statements:

Firstly, a new and improved construction *could* for any n, for any $\phi \in \mathrm{Sym}(n)$, find a $\tilde{Q}$ that is at least a loop, such that $\phi \in \mathrm{AAut}(\tilde{Q})$.

Secondly, a new and improved constrocution *could* for *certain* n, for any (or most likely, only *certain*) $\phi \in \mathrm{Sym}(n)$, find a $\tilde{Q}$ that is an abelian group, such that $\phi \in \mathrm{AAut}(\tilde{Q})$.

Idea 5.1.2. Remark 2.2.7.

Idea 5.1.3. The code can currently only generate up to 5×5 latin squares. If one tries to generate 6×6 latin squares it doesn't work. In order to improve the generation algorithm I have the following two ideas:

Firstly, reduce the amount of latin squares that is needed to be generated:

By finding a certain "class" of latin squares that can be used to test and classify a result, one would only need to generate those in order to test the result. For finding a quasigroup where a permutation is an affine automorphism, perhaps only one of the representatives with a certain affine fingerprint (Idea 5.3.3) need to be generated. How to do that efficiently, I do not know.

This could further be whittled down if one for example knew that there always was a loop where a map was an affine automorphism, so one only needed to generate all the loops with different fingerprint, etc... Or maybe one only needed to generate every abelian group and then permute the rows/columns/elements in a certain way to get the rest (permuting rows/columns/elements is applying an isotopy).

Best case scenario is one only has to generate some of the abelian groups with fixed identity, and then applying permutations or some isotopies to them in order to get the rest that one need to test (perhaps the isotopies can depend on the context, i.e. what is being tested.). My code can generate every abelian group with fixed identity up to order 10 pretty quickly, that might be useful.

Secondly, optimizing the code directly:

Generating the 6×6 latin squares doesn't finish on my computer due to insufficient RAM. So trying to optimize the code so that it uses less memory may be important. A known result is that every quasigroup is isotopic to a loop (with fixed identity, and without swapping labels). So therefore, one could generate every loop and then afterwards apply every isotopy to every loop in order to get every possible quasigroup. In this way, the code might not run faster, but the memory load could probably be reduced.

The code could maybe also be multi threaded. Currently the code is only single-threaded.

The recursive generation of the latin squares could be changed to a iterative version, maybe that would save on memory.

There are also most likely other smarter and more efficient algorithms for generating the necessary latin squares. If one checks for associativity, maybe one could look at it like a property of a 3D group operation table. Maybe there is some symmetry on this 3D group operation table that implies associativity that would be faster to check?

Applying optimizations should come second to reducing the amount of latin squares that need to be generated, since the amount of latin squares explodes extremely quickly as the size increases. And although the amount of abelian groups also explodes, it does so much slower, and is therefore way more bearable to compute. Even without siginificant optimizations, my code can generate every 10×10 abelian group latin square fairly quickly.

5.2 Crazy ideas

Idea 5.2.1. One of my more crazy ideas is to try and use the Eckmann-Hilton argument to show when a permutation is an automorphism, since from Theorem 3.3.5 would imply that $*_Q$ and $*_{\phi(\tilde{Q})}$ are equal if and only if $\phi \in \mathrm{Aut}(Q)$.

However, Eckmann-Hilton would imply that $*_Q$ and $*_{\tilde{Q}}$ are both associative and commutative, which would mean they're abelian groups from our structure. This most likely makes this idea not possible in the general case, but might be possible to apply in some special cases where we know there is some $\tilde{Q}$ where the permutation is in $\mathrm{Aut}(\tilde{Q})$ or maybe $\mathrm{AAut}(\tilde{Q})$.

Idea 5.2.2. Another crazy idea relates to the attacks. Could it be possible to use the compuonding knowledge gained from asking for different values of $\phi(x_i)$. Let $\tilde{Q}$ be the commutative quasigroup gained from Corollary 3.4.4. Then it has binary operation $a *_{\tilde{Q}} b = \phi(a) *_Q \phi(b)$. If one for example has asked for k different values og $\phi^{-1}(x_i)$, then one could calculate k choose 2 different sums $x_i *_{\tilde{Q}} x_j$, which is $k(k-1) \in o(k^2)$. Could one use this compounding knowledge to help with gaining enough useful information for an attack fast enough?

Idea 5.2.3. Is it possible to use information theory code generating algorithms to make latin squares? If latin squares can make MDS-codes, then could one go the other way using Reed-Solomon codes?

5.3 Observations

Idea 5.3.1. It seems like given two abelian groups $H, G \in \mathrm{Ab}(n)$, then $\mathrm{Aut}(H) \cap \mathrm{Aut}(G)$ is either $\mathrm{Aut}(H), \mathrm{Aut}(G)$ or $\{\mathrm{Id}\}$. However, I've yet to prove this.

Idea 5.3.2. Experiments imply that for two abelian groups $H, G \in \mathrm{Ab}(n)$ where $H \simeq G$, then $\mathrm{AAut}(H) \neq \mathrm{AAut}(G)$. In the "complete_5.ods" file, s_{44915} and s_{85427} do not have the same affine automorphisms. However, it is known that there is only one ismorphism class for abelian groups of order 5, so they must be isomorphic.

However, I've yet to prove the above statement algebraically. Might be usefull for context, verification and further ideas.

Idea 5.3.3. I have looked into something that we called the "automorphism fingerprint" of different quasigroups. For a given $Q \in \mathrm{Quas}(n)$, then this is $\mathrm{Aut}(Q)$. Every generated table that I've made is sorted with regards to this "fingerprint". Have looked a bit into the relation between elements with the same automorphism fingerprint, but didn't find or test anything concrete for higher n than 3. For $n = 3$ it seems like quasigroups with the same fingerprint are related by a single flip of either row or colum or the transpose of each other, but not always.

I did not have time to look into the "affine automorphism fingerprint" of different quasigroups. They're arguably more important, since that is what the main conjecture and construction revolves around. If one can manage to find a correlation between elements with the same fingerprint, then it might be possible to only generate the latin squares that have different fingerprints, which could help a lot with Idea 5.1.3. Maybe it could also show a way to create a new construction and help with Idea 5.1.1.

Idea 5.3.4. I found no significant results or observations when working with n a prime, a composite number, or a power of a prime (i.e. a vector space), except for that every group was abelian when n was prime. Shouldn't there be some additional structure found that are lost between the different cases? Could one exploit the structure for certain n in for example Idea 5.1.3?

5.4 Future experiments

There are some experiments that I've yet to run, but that seems interesting.

Idea 5.4.1. Firstly, want to verify the first conjecture in Idea 5.1.1 by checking for every permutation if there is a loop where it's in the AAut of that loop. Maybe in the spreadsheet, one could add an additional row above the permutation where the best structure that permutation is AAut in is shown.

Idea 5.4.2. Secondly, want to improve the bounds that have been mentioned in the second conjecture in Idea 5.1.1. For $n = 2^k$, does the conjecture hold? For which $\phi \in \mathrm{Sym}(n)$ does the conjecture hold? Does it hold for any weaker statements? And so on.

5.5 Maybe irrelevant stuff

Idea 5.5.1. One result that I tried proving, but gave up halfway was the following:

For $n \geq 3$ for every $G \in \mathrm{Ab}(n)$, then $\mathrm{Aut}(G) \neq \{\mathrm{Id}\}$.

It seems very close to be true, but I didn't manage to show that and gave up, because it seemed irrelevant for what I was trying to accomplish.

6 Cool resources

Some interesting integer sequences:

- Labeled quasigroups
- Labeled loops
- Labeled groups
- Labeled abelian groups
- Labeled abelian groups with fixed identity

Helpful quasigroup wikipedia articles:

- Quasigroup
- Latin square

- Small Latin Squares and Quasigroups