

[https://teorth.github.io/equational\\_theories](https://teorth.github.io/equational_theories)[https : //github.com/teorth/equational\\_theories](https://github.com/teorth/equational_theories)[https : //teorth.github.io/equational\\_theories](https://teorth.github.io/equational_theories)

# Equational theories

Terence Tao

October 13, 2024

# Chapter 1

## Basic theory of magmas

**Definition 1.1** (Magma). A *magma* is a set  $G$  equipped with a binary operation  $\diamond : G \times G \rightarrow G$ . A *homomorphism*  $\varphi : G \rightarrow H$  between two magmas is a map such that  $\varphi(x \diamond y) = \varphi(x) \diamond \varphi(y)$  for all  $x, y \in G$ . An *isomorphism* is an invertible homomorphism.

Groups, semi-groups, and monoids are familiar examples of magmas. However, in general we do not expect magmas to have any associative properties. In some literature, magmas are also known as groupoids, although this term is also used for a slightly different object (a category with inverses).

A magma is called *empty* if it has cardinality zero, *singleton* if it has cardinality one, and *non-trivial* otherwise.

The number of magma structures on a set  $G$  of cardinality  $n$  is of course  $n^{n^2}$ , which is <sup>1</sup>

1, 1, 16, 19683, 4294967296, 298023223876953125, ...

(OEIS A002489). Up to isomorphism, the number of finite magmas of cardinality  $n$  up to isomorphism is the slightly slower growing sequence

1, 1, 10, 3330, 178981952, 2483527537094825, 14325590003318891522275680, ...

(OEIS A001329).

**Definition 1.2** (Free Magma). The *free magma*  $M_X$  generated by a set  $X$  (which we call an *alphabet*) is the set of all finite formal expressions built from elements of  $X$  and the operation  $\diamond$ . An element of  $M_X$  will be called a *word* with alphabet  $X$ . The *order* of a word is the number of  $\diamond$  symbols needed to generate the word. Thus for instance  $X$  is precisely the set of words of order 0 in  $M_X$ .

For sake of concreteness, we will take the alphabet  $X$  to default to the natural numbers  $\mathbb{N}$  if not otherwise specified.

For instance, if  $X = \{0, 1\}$ , then  $M_X$  would consist of the following words:

---

<sup>1</sup>All sequences start from  $n = 0$  unless otherwise specified.

- 0, 1 (the words of order 0);
- $0 \diamond 0, 0 \diamond 1, 1 \diamond 0, 1 \diamond 1$  (the words of order 1);
- $0 \diamond (0 \diamond 0), 0 \diamond (0 \diamond 1), 0 \diamond (1 \diamond 0), 0 \diamond (1 \diamond 1), 1 \diamond (0 \diamond 0), 1 \diamond (0 \diamond 1), 1 \diamond (1 \diamond 0), 1 \diamond (1 \diamond 1), (0 \diamond 0) \diamond 0, (0 \diamond 0) \diamond 1, (0 \diamond 1) \diamond 0, (0 \diamond 1) \diamond 1, (1 \diamond 0) \diamond 0, (1 \diamond 0) \diamond 1, (1 \diamond 1) \diamond 0, (1 \diamond 1) \diamond 1$  (the words of order 2);
- etc.

**Lemma 1.3.** *FreeMagma.elementsOfNumNodesEqCardEqCatalan<sub>m</sub>ul<sub>p</sub>owForafinitealphabetX, thenumberofw where  $C_n$  is the  $n^{\text{th}}$  Catalan number and  $X$  is the cardinality of  $X$ .*

*Proof.* Follows from standard properties of Catalan numbers.  $\square$

The first few Catalan numbers are

$$1, 1, 2, 5, 14, 42, 132, \dots$$

(OEIS A000108).

**Definition 1.4** (Induced homomorphism). free-magma-def Given a function  $f : X \rightarrow G$  from an alphabet  $X$  to a magma  $G$ , the *induced homomorphism*  $\varphi_f : M_X \rightarrow G$  is the unique extension of  $f$  to a magma homomorphism. Similarly, if  $\pi : X \rightarrow Y$  is a function, we write  $\pi_* : M_X \rightarrow M_Y$  for the unique extension of  $\pi$  to a magma homomorphism.

For instance, if  $f : \{0, 1\} \rightarrow G$  maps 0, 1 to  $x, y$  respectively, then

$$\varphi_f(0 \diamond 1) = x \diamond y$$

$$\varphi_f(1 \diamond (0 \diamond 1)) = y \diamond (x \diamond y)$$

and so forth. If  $\pi : \mathbb{N} \rightarrow \mathbb{N}$  is the map  $\pi(n) := n + 1$ , then

$$\pi_*(0 \diamond 1) = 1 \diamond 2$$

$$\pi_*(1 \diamond (0 \diamond 1)) = 2 \diamond (1 \diamond 2)$$

and so forth.

**Definition 1.5** (Law). Law.MagmaLawinduced-def Let  $X$  be a set. A *law* with alphabet  $X$  is a formal expression of the form  $w \simeq w'$ , where  $w, w' \in M_X$  are words with alphabet  $X$  (thus one can identify laws with alphabet  $X$  with elements of  $M_X \times M_X$ ). A magma  $G$  *satisfies* the law  $w \simeq w'$  if we have  $\varphi_f(w) = \varphi_f(w')$  for all  $f : X \rightarrow G$ , in which case we write  $G \models w \simeq w'$ .

Thus, for instance, the commutative law

$$0 \diamond 1 \simeq 1 \diamond 0 \tag{1.1}$$

is satisfied by a magma  $G$  if and only if

$$x \diamond y = y \diamond x \quad (1.2)$$

for all  $x, y \in G$ . We refer to ?? as the *equation* associated to the law ??. One can think of equations as the “semantic” interpretation of a “syntactic” law. However, we shall often abuse notation and identify a law with its associated equation. In particular, we shall (somewhat carelessly) also refer to ?? as “the commutative law” (rather than “the commutative equation”).

**Definition 1.6** (Models). *modelslaw-def* A *theory* is a set  $\Gamma$  of laws. Given a theory  $\Gamma$ , a magma  $G$  is a *model* of  $\Gamma$  with the (overloaded) notation  $G \models \Gamma$  if  $G \models w \simeq w'$  for every  $w \simeq w'$  in  $\Gamma$ ; we also say that  $G$  *obeys*  $\Gamma$ . Given a law  $E$ , we write  $\Gamma \models E$  if every magma  $G$  that models  $\Gamma$ , also models  $E$ .

**Definition 1.7** (Derivation). *derivelaw-def* Given a theory  $\Gamma$  and a law  $w \simeq w'$  over a fixed alphabet  $X$ , we say that  $\Gamma$  *derives*  $w \simeq w'$ , and write  $\Gamma \vdash w \simeq w'$ , if the law can be obtained using a finite number of applications of the following rules:

1. if  $w \simeq w' \in \Gamma$ , then  $\Gamma \vdash w \simeq w'$ .
2.  $\Gamma \vdash w \simeq w$  for any word  $w$ .
3. if  $\Gamma \vdash w \simeq w'$ , then  $\Gamma \vdash w' \simeq w$ .
4. if  $\Gamma \vdash w \simeq w'$  and  $\Gamma \vdash w' \simeq w''$ , then  $\Gamma \vdash w \simeq w''$ .
5. if  $\Gamma \vdash w \simeq w'$ , then  $\Gamma \vdash \varphi_f(w) \simeq \varphi_f(w')$  for every  $f : X \rightarrow M_X$ .
6. if  $\Gamma \vdash w_1 \simeq w_2$  and  $\Gamma \vdash w_3 \simeq w_4$ , then  $\Gamma \vdash w_1 \diamond w_3 \simeq w_2 \diamond w_4$ .

This definition is useful because of the following theorem:

**Theorem 1.8** (Birkhoff’s completeness theorem). *Completenessmodels-def, derivation-def* For any theory  $\Gamma$  and words  $w, w'$  over a fixed alphabet

$$\Gamma \vdash w \simeq w' \text{ iff } \Gamma \models w \simeq w'.$$

*Proof.* (Sketch) The ‘only if’ component is soundness, and follows from verifying that the rules of inference in ?? holds for  $\models$ . The ‘if’ part is completeness, and is proven by constructing the magma of words, quotiented out by the relation  $\Gamma \vdash w \simeq w'$ , which is easily seen to be an equivalence relation respecting the magma operation.  $\square$

**Corollary 1.9** (Compactness theorem). *models-def* Let  $\Gamma$  be a theory, and let  $E$  be a law. Then  $\Gamma \models E$  if and only if there exists a finite subset  $\Gamma'$  of  $\Gamma$  such that  $\Gamma' \models E$ .

*Proof.* sound-complete The claim is obvious for  $\vdash$ , and the claim then follows from ??.  $\square$

**Lemma 1.10** (Pushforward). *law-def Let  $w \simeq w'$  be a law with some alphabet  $X$ ,  $G$  be a magma, and  $\pi : X \rightarrow Y$  be a function. If  $G \models w \simeq w'$ , then  $G \models \pi_*(w) \simeq \pi_*(w')$ . In particular, if  $\pi$  is a bijection, the statements  $G \models w \simeq w'$  and  $G \models \pi_*(w) \simeq \pi_*(w')$  are equivalent.*

*Proof.* Trivial. □

If  $\pi$  is a bijection, we will call  $\pi_*(w) \simeq \pi_*(w')$  a *relabeling* of the law  $w \simeq w'$ . Thus for instance

$$5 \diamond 7 \simeq 7 \diamond 5$$

is a relabeling of the commutative law  $??$ . By the above lemma, relabeling does not affect whether a given magma satisfies a given law.

**Lemma 1.11** (Equivalence). *law-def Let  $G$  be a magma and  $X$  be an alphabet. Then the relation  $G \models w \simeq w'$  is an equivalence relation on  $M_X$ .*

*Proof.* Trivial. □

Define the total order of a law  $w \simeq w'$  to be the sum of the orders of  $w$  and  $w'$ .

**Lemma 1.12** (Counting laws up to relabeling). *push Up to relabeling, the number of laws  $w \simeq w'$  of total order  $n$  is  $C_{n+1}B_{n+2}$ .*

*Proof.* Follows from the properties of Catalan and Bell numbers. □

The first few Bell numbers are

$$1, 1, 2, 5, 15, 52, 203, \dots$$

(OEIS A000110).

The sequence in  $??$  is

$$2, 10, 75, 728, 8526, 115764, \dots$$

(OEIS A289679).

Now we would also like to count laws up to relabeling and symmetry.

**Lemma 1.13** (Counting laws up to relabeling and symmetry). *push Up to relabeling and symmetry, the number of laws  $w \simeq w'$  of total order  $n$  is*

$$C_{n+1}B_{n+2}/2$$

when  $n$  is odd, and

$$(C_{n+1}B_{n+2} + C_{n/2}(2D_{n+2} - B_{n+2}))/2$$

when  $n$  is even, where  $D_n$  is the number of partitions of  $[n]$  up to reflection.

*Proof.* Elementary counting. □

The sequence  $D_n$  is

$$1, 1, 2, 4, 11, 32, 117, \dots$$

(OEIS A103293), and the sequence in ?? is

$$2, 5, 41, 364, 4294, 57882, 888440, \dots$$

(OEIS A376620).

We can also identify all laws of the form  $w \simeq w$  with the trivial law  $0 \simeq 0$ . The number of such laws of total order  $n$  is zero if  $n$  is odd, and  $C_{n/2}B_{n/2+1}$  if  $n$  is even. We conclude:

**Lemma 1.14** (Counting laws up to relabeling, symmetry, and triviality). *law-count-sym Up to relabeling, symmetry, and triviality, the number of laws of total order  $n$  is*

$$C_{n+1}B_{n+2}/2$$

*if  $n$  is odd, 2 if  $n = 0$ , and*

$$(C_{n+1}B_{n+2} + C_{n/2}(2D_{n+2} - B_{n+2}))/2 - C_{n/2}B_{n/2+1}$$

*if  $n \geq 2$  is even.*

*Proof.* Routine counting. □

This sequence is

$$2, 5, 39, 364, 4284, 57882, 888365, \dots$$

(OEIS A376640).

In particular, up to relabeling, symmetry, and triviality, there are exactly  $2 + 5 + 39 + 364 + 4284 = 4694$  laws of total order at most 4. A list can be found here. A script for generating them may be found here. The list is sorted first by the total number of operations, then by the number of operations on the LHS. Within each such class we define an order on expressions by lexical order on variables (ordered  $x, y, z, w, u, v$ ). The equations are arranged to be minimal with respect to this sorting order, thus the LHS either has fewer operations than the RHS or else it has the same number of operations and occurs earlier in lexical order than the RHS.

## Chapter 2

### Selected laws

In this project we study the 4694 laws (up to symmetry and relabeling) of total order at most 4.

Selected laws of interest are listed below, as well as in this file.

**Definition 2.1** (Equation 1). Equation1magma-def Equation 1 is the law  $0 \simeq 0$  (or the equation  $x = x$ ).

This is the trivial law, satisfied by all magmas. It is self-dual.

**Definition 2.2** (Equation 2). Equation2magma-def Equation 2 is the law  $0 \simeq 1$  (or the equation  $x = y$ ).

This is the singleton law, satisfied only by the empty and singleton magmas. It is self-dual.

**Definition 2.3** (Equation 3). Equation3magma-def Equation 3 is the law  $0 \simeq 0 \diamond 0$  (or the equation  $x = x \diamond x$ ).

This is the idempotence law. It is self-dual.

**Definition 2.4** (Equation 4). Equation4magma-def Equation 4 is the law  $0 \simeq 0 \diamond 1$  (or the equation  $x = x \diamond y$ ).

This is the left absorption law.

**Definition 2.5** (Equation 5). Equation5magma-def Equation 5 is the law  $0 \simeq 1 \diamond 0$  (or the equation  $x = y \diamond x$ ).

This is the right absorption law (the dual of ??).

**Definition 2.6** (Equation 6). Equation6magma-def Equation 6 is the law  $0 \simeq 1 \diamond 1$  (or the equation  $x = y \diamond y$ ).

This law is equivalent to the singleton law.

**Definition 2.7** (Equation 7). Equation7magma-def Equation 7 is the law  $0 \simeq 1 \diamond 2$  (or the equation  $x = y \diamond z$ ).



This law is equivalent to the singleton law.

**Definition 2.8** (Equation 8). Equation8magma-def Equation 8 is the law  $0 \simeq 0 \diamond (0 \diamond 0)$  (or the equation  $x = x \diamond (x \diamond x)$ ).

**Definition 2.9** (Equation 14). Equation14magma-def Equation 14 is the law  $0 \simeq 1 \diamond (0 \diamond 1)$  (or the equation  $x = y \diamond (x \diamond y)$ ).

Appears in Problem A1 from Putnam 2001. See ??.

**Definition 2.10** (Equation 16). Equation16magma-def Equation 16 is the law  $0 \simeq 1 \diamond (1 \diamond 0)$  (or the equation  $x = y \diamond (y \diamond x)$ ).

**Definition 2.11** (Equation 23). Equation23magma-def Equation 23 is the law  $0 \simeq (0 \diamond 0) \diamond 0$  (or the equation  $x = (x \diamond x) \diamond x$ ).

This is the dual of ??.

**Definition 2.12** (Equation 29). Equation29magma-def Equation 29 is the law  $0 \simeq (1 \diamond 0) \diamond 1$  (or the equation  $x = (y \diamond x) \diamond y$ ).

Appears in Problem A1 from Putnam 2001. Dual to ??. See ??.

**Definition 2.13** (Equation 38). Equation38magma-def Equation 38 is the law  $0 \diamond 0 \simeq 0 \diamond 1$  (or the equation  $x \diamond x = x \diamond y$ ).

This law asserts that the magma operation is independent of the second argument.

**Definition 2.14** (Equation 39). Equation39magma-def Equation 39 is the law  $0 \diamond 0 \simeq 1 \diamond 0$  (or the equation  $x \diamond x = y \diamond x$ ).

This law asserts that the magma operation is independent of the first argument (the dual of ??).

**Definition 2.15** (Equation 40). Equation40magma-def Equation 40 is the law  $0 \diamond 0 \simeq 1 \diamond 1$  (or the equation  $x \diamond x = y \diamond y$ ).

This law asserts that all squares are constant. It is self-dual.

**Definition 2.16** (Equation 41). Equation41magma-def Equation 41 is the law  $0 \diamond 0 \simeq 1 \diamond 2$  (or the equation  $x \diamond x = y \diamond z$ ).

This law is equivalent to the constant law, ??.

**Definition 2.17** (Equation 42). Equation42magma-def Equation 42 is the law  $0 \diamond 1 \simeq 0 \diamond 2$  (or the equation  $x \diamond y = x \diamond z$ ).

Equivalent to ??.

**Definition 2.18** (Equation 43). Equation43magma-def Equation 43 is the law  $0 \diamond 1 \simeq 1 \diamond 0$  (or the equation  $x \diamond y = y \diamond x$ ).

The commutative law. It is self-dual.

**Definition 2.19** (Equation 45). Equation45magma-def Equation 45 is the law  $0 \diamond 1 \simeq 2 \diamond 1$  (or the equation  $x \diamond y = z \diamond y$ ).

This is the dual of ??.

**Definition 2.20** (Equation 46). Equation46magma-def Equation 46 is the law  $0 \diamond 1 \simeq 2 \diamond 3$  (or the equation  $x \diamond y = z \diamond w$ ).

The constant law: all products are constant. It is self-dual.

**Definition 2.21** (Equation 65). Equation65magma-def Equation 65 is the law  $0 \simeq 1 \diamond (0 \diamond (1 \diamond 0))$  (or the equation  $x = y \diamond (x \diamond (y \diamond x))$ ).

The “Asterix” law, studied further in Chapter ??.

**Definition 2.22** (Equation 168). Equation168magma-def Equation 168 is the law  $0 \simeq (1 \diamond 0) \diamond (0 \diamond 2)$  (or the equation  $x = (y \diamond x) \diamond (x \diamond z)$ ).

The law of a central groupoid. It is self-dual.

**Definition 2.23** (Equation 381). Equation381magma-def Equation 381 is the law  $0 \diamond 1 \simeq (0 \diamond 2) \diamond 1$  (or the equation  $x \diamond y = (x \diamond z) \diamond y$ ).

Appears in Putnam 1978, Problem A4, part (b).

**Definition 2.24** (Equation 387). Equation387magma-def Equation 387 is the law  $0 \diamond 1 \simeq (1 \diamond 1) \diamond 0$  (or the equation  $x \diamond y = (y \diamond y) \diamond x$ ).

Introduced in MathOverflow. See ??

**Definition 2.25** (Equation 477). Equation477magma-def Equation 477 is the law  $0 \simeq 1 \diamond (0 \diamond (1 \diamond (1 \diamond 1)))$  (or the equation  $x = y \diamond (x \diamond (y \diamond (y \diamond y)))$ ).

An example of a confluent law; see ??.

**Definition 2.26** (Equation 953). Equation953magma-def Equation 953 is the law  $0 = 1 \diamond ((2 \diamond 0) \diamond (2 \diamond 2))$  (or the equation  $x = y \diamond ((z \diamond x) \diamond (z \diamond z))$ ).

An example of a trivial law; see ??.

**Definition 2.27** (Equation 1491). Equation1491magma-def Equation 1491 is the law  $0 \simeq (1 \diamond 0) \diamond (1 \diamond (1 \diamond 0))$  (or the equation  $x = (y \diamond x) \diamond (y \diamond (y \diamond x))$ ).

The “Obelix” law, studied further in Chapter ??.

**Definition 2.28** (Equation 1571). Equation1571magma-def Equation 1571 is the law  $0 \simeq (1 \diamond 2) \diamond (1 \diamond (0 \diamond 2))$  (or the equation  $x = (y \diamond z) \diamond (y \diamond (x \diamond z))$ ).

Introduced in [?]. As shown in ??, this law characterizes abelian groups of exponent two.

**Definition 2.29** (Equation 1689). Equation1689magma-def Equation 1689 is the law  $0 \simeq (1 \diamond 0) \diamond ((0 \diamond 2) \diamond 2)$  (or the equation  $x = (y \diamond x) \diamond ((x \diamond z) \diamond z)$ ).

Mentioned in [?]. See ??.

**Definition 2.30** (Equation 2662). Equation2662magma-def Equation 2662 is the law  $0 \simeq ((0 \diamond 1) \diamond (0 \diamond 1)) \diamond 0$  (or the equation  $x = ((x \diamond y) \diamond (x \diamond y)) \diamond x$ ).

Appears in [?].

**Definition 2.31** (Equation 3167). Equation3167magma-def Equation 3167 is the law  $0 \simeq (((1 \diamond 1) \diamond 2) \diamond 2) \diamond 0$  (or the equation  $x = (((y \diamond y) \diamond z) \diamond z) \diamond x$ ).

**Definition 2.32** (Equation 3588). Equation3588magma-def Equation 3588 is the law  $0 \diamond 1 \simeq 2 \diamond ((0 \diamond 1) \diamond 2)$  (or the equation  $x \diamond y = z \diamond ((x \diamond y) \diamond z)$ ).

Our project located this law as one member of an “Austin pair”; see Chapter ??.

**Definition 2.33** (Equation 3722). Equation3722magma-def Equation 3722 is the law  $0 \diamond 1 \simeq (0 \diamond 1) \diamond (0 \diamond 1)$  (or the equation  $x \diamond y = (x \diamond y) \diamond (x \diamond y)$ ).

Appears in Putnam 1978, Problem A4, part (a). It is self-dual.

**Definition 2.34** (Equation 3744). Equation3744magma-def Equation 3744 is the law  $0 \diamond 1 \simeq (0 \diamond 2) \diamond (3 \diamond 1)$  (or the equation  $x \diamond y = (x \diamond z) \diamond (w \diamond y)$ ).

This law is called a “bypass operation” in Putnam 1978, Problem A4. It is self-dual. See ??.

**Definition 2.35** (Equation 3994). magma-def Equation 3994 is the law  $0 \diamond 1 \simeq (2 \diamond (0 \diamond 1)) \diamond 2$  (or the equation  $x \diamond y = (z \diamond (x \diamond y)) \diamond z$ ).

Our project located this law as one member of an “Austin pair”; see Chapter ??.

**Definition 2.36** (Equation 4512). Equation4512magma-def Equation 4512 is the law  $0 \diamond (1 \diamond 2) \simeq (0 \diamond 1) \diamond 2$  (or the equation  $x \diamond (y \diamond z) = (x \diamond y) \diamond z$ ).

The associative law. It is self-dual.

**Definition 2.37** (Equation 4513). Equation4513magma-def Equation 4513 is the law  $0 \diamond (1 \diamond 2) \simeq (0 \diamond 1) \diamond 3$  (or the equation  $x \diamond (y \diamond z) = (x \diamond y) \diamond w$ ).

**Definition 2.38** (Equation 4522). Equation4522magma-def Equation 4522 is the law  $0 \diamond (1 \diamond 2) \simeq (0 \diamond 3) \diamond 4$  (or the equation  $x \diamond (y \diamond z) = (x \diamond w) \diamond u$ ).

Dual to ??.

**Definition 2.39** (Equation 4564). Equation4564magma-def Equation 4564 is the law  $0 \diamond (1 \diamond 2) \simeq (3 \diamond 1) \diamond 2$  (or the equation  $x \diamond (y \diamond z) = (w \diamond y) \diamond z$ ).

Dual to ??.

**Definition 2.40** (Equation 4579). Equation 4579 magma-def Equation 4579 is the law  $0 \diamond (1 \diamond 2) \simeq (3 \diamond 4) \diamond 2$  (or the equation  $x \diamond (y \diamond z) = (w \diamond u) \diamond z$ ).

Dual to ??.

**Definition 2.41** (Equation 4582). Equation 4582 magma-def Equation 4582 is the law  $0 \diamond (1 \diamond 2) \simeq (3 \diamond 4) \diamond 5$  (or the equation  $x \diamond (y \diamond z) = (w \diamond u) \diamond v$ ).

This law asserts that all triple constants (regardless of bracketing) are constant.

## 2.1 Equations of order greater than 4

We note some selected laws of order more than 5, which are used in some later chapters of the blueprint.

**Definition 2.42** (Equation 5105). magma-def Equation 5105 Equation 5105 is the law  $0 \simeq 1 \diamond (1 \diamond (1 \diamond (0 \diamond (2 \diamond 1))))$  (or the equation  $x = y \diamond (y \diamond (y \diamond (x \diamond (z \diamond y))))$ ).

This law of order 5 was mentioned in [?]. See ??.

**Definition 2.43** (Equation 26302). magma-def Equation 26302 is the law  $0 \simeq (1 \diamond ((2 \diamond 0) \diamond 3)) \diamond (0 \diamond 3)$  (or the equation  $x = (y \diamond ((z \diamond x) \diamond w)) \diamond (x \diamond w)$ ).

A law that characterizes natural central groupoids; see ??.

**Definition 2.44** (Equation 28770). magma-def Equation 28770 Equation 28770 is the law  $0 \simeq (((1 \diamond 1) \diamond 1) \diamond 0) \diamond (1 \diamond 2)$  (or the equation  $x = (((y \diamond y) \diamond y) \diamond x) \diamond (y \diamond z)$ ).

This law of order 5 was introduced by Kisielewicz [?]. See ??.

**Definition 2.45** (Equation 345169). magma-def Equation 345169 is the law  $0 \simeq (1 \diamond ((0 \diamond 1) \diamond 1)) \diamond (0 \diamond (2 \diamond 1))$  (or the equation  $x = (y \diamond ((x \diamond y) \diamond y)) \diamond (x \diamond (z \diamond y))$ ).

This law of order 6 was shown in [?] to characterize the Sheffer stroke in a boolean algebra; see ??.

**Definition 2.46** (Equation 374794). Equation 374794 magma-def Equation 374794 is the law  $0 \simeq (((1 \diamond 1) \diamond 1) \diamond 0) \diamond ((1 \diamond 1) \diamond 2)$  (or the equation  $x = (((y \diamond y) \diamond y) \diamond x) \diamond ((y \diamond y) \diamond z)$ ).

This law of order 6 was introduced by Kisielewicz [?]; see ??.

## Chapter 3

# Infinite models

In this chapter we consider non-implications which are refuted only on infinite models, as those are more challenging to prove—they can't be proved by directly giving an operation table and checking which laws it satisfies.

The singleton or empty magma obeys all equational laws. One can ask whether an equational law admits nontrivial finite or infinite models. An *Austin law* is a law which admits infinite models, but no nontrivial finite models. Austin [?] established the first such law, namely the order 9 law

$$(((1 \diamond 1) \diamond 1) \diamond 0) \diamond (((1 \diamond 1) \diamond ((1 \diamond 1) \diamond 1)) \diamond 2) \simeq 0.$$

A shorter Austin law of order 6 was established in [?]:

**Theorem 3.1** (Kisielewicz's first Austin law). *InfModel.Finite.Equation374794 implies Equation2, InfModel.E*

*Proof.* First we show that every finite model of ?? is trivial. Write  $y^2 := y \diamond y$  and  $y^3 := y^2 \diamond y$ . For any  $y, z$ , introduce the functions  $f_y : x \mapsto y^3 \diamond x$  and  $g_{yz} : x \mapsto x \diamond (y^2 \diamond z)$ . ?? says that  $g_{yz}(f_y(x)) = x$ , hence by finiteness  $g_{yz} = f_y^{-1}$ , showing that  $g_{yz}$  does not depend on the value of  $z$ . Since

$$f_y(y^2 \diamond z) = g_{yz}(y^3),$$

it follows that  $f_y(y^2 \diamond z) = f_y(y^3)$  which by injectivity of  $f_y$  implies that  $z \mapsto y^2 \diamond z$  is a constant function (with  $y$  fixed). Substituting  $y^2$  for  $y$  shows that the same is true for  $z \mapsto (y^2 \diamond y^2) \diamond z$ , and since

$$f_y(z) = (y^2 \diamond y) \diamond z = (y^2 \diamond y^2) \diamond z$$

we conclude that  $f_y$  is also a constant function. But this function is already known to be injective, thus there do not exist distinct elements in its domain, showing that the model must be trivial.

To construct an infinite model, consider the magma of positive integers  $\mathbb{Z}^+$  with the operation  $x \diamond y$  defined by

$$x \diamond y = \begin{cases} 2^y, & x = y \\ 3^y, & x = 1, y \neq 1 \\ z, & x = 3^z, y \neq x \\ 1, & \text{else} \end{cases}.$$

Then  $y \diamond y = 2^y$  and  $(y \diamond y) \diamond y = 1$  for all  $y$ . If  $x \neq 1$  we have that

$$((y \diamond y) \diamond y) \diamond x = 3^x,$$

and since  $(y \diamond y) \diamond z$  is a power of two for all  $y, z$  it follows that

$$3^x \diamond ((y \diamond y) \diamond z) = x.$$

The case  $x = 1$  requires a further argument: observe that  $w = (y \diamond y) \diamond z$  evaluates to one unless  $z = 2^y$ , in which case it evaluates to  $2^{2^y}$  (which is greater than or equal to four). In particular,  $w$  never takes the value two. Thus

$$(((y \diamond y) \diamond y) \diamond 1) \diamond ((y \diamond y) \diamond z) = 2 \diamond w = 1,$$

concluding our proof that this magma is a model of ?? □

An even shorter law (order 5) was obtained by the same author in a follow-up paper [?]:

**Theorem 3.2** (Kisielewicz's second Austin law). *eq2, eq28770 ?? is an Austin law.*

*Proof.* Using the  $y^2$  and  $y^3$  notation as before, the law reads

$$x = (y^3 \diamond x) \diamond (y \diamond z). \tag{3.1}$$

In particular, for any  $y$ , the map  $T_y : x \mapsto y^3 \diamond x$  is injective, hence bijective in a finite model  $G$ . In particular we can find a function  $f : G \rightarrow G$  such that  $T_y f(y) = y^3$  for all  $y$ . Applying ?? with  $x = f(y)$ , we conclude

$$T_y(y \diamond z) = y^3 \diamond (y \diamond z) = f(y)$$

and thus  $y \diamond z$  is independent of  $z$  by injectivity of  $T_y$ . Thus, the left-hand side of ?? does not depend on  $x$ , and so the model is trivial. This shows there are no non-trivial finite models.

To establish an infinite model, use  $\mathbb{N}$  with  $x \diamond y$  defined by requiring

$$y \diamond y = 2^y; \quad 2^y \diamond y = 3^y$$

and

$$3^y \diamond x = 3^y 5^x$$

for  $x \neq 3^y$ , and

$$(3^y 5^x) \diamond z = x$$

for  $z \neq 3^y 5^x$ . Finally set

$$2^{3^y} \diamond z = 3^y$$

for  $z \neq 3^y, 2^{3^y}$ . All other assignments of  $\diamond$  may be made arbitrarily. It is then a routine matter to establish ?? □

In that paper a computer search was also used to show that no law of order four or less is an Austin law.

An open question is whether ?? is an Austin law. We have the following partial result from [?]:

**Theorem 3.3** (Equation 5105 has no non-trivial finite models). *InfModel.Finite.Equation5105impliesEquation5105forallnontrivialfinitemodels.*

*Proof.* From ?? we see that the map  $w \mapsto y \diamond w$  is onto, hence injective in a finite model. Using this injectivity four times in ??, we see that  $z \diamond y$  does not depend on  $z$ , hence the expression  $x \diamond (z \diamond y)$  does not depend on  $x$ . By ?? again, this means that  $x$  does not depend on  $x$ , which is absurd in a non-trivial model. □

We also have such a non-implication involving two laws of order 4:

**Theorem 3.4** (3994 implies 3588 for finite models). *eq3994,eq3588,models-def InfModel.Finite.Equation3994impliesEquation3588Allfinitemagmaswhichsatisfy??alsosatisfy??.*

*Proof.* For a finite magma  $M$ , consider the set  $S = \{x \diamond y | x, y \in M\}$ . Now  $f_z : x \mapsto z \diamond x$  and  $g_z : x \mapsto x \diamond z$ . They both map  $S$  to  $S$ , and due to the hypothesis  $g_z \diamond f_z$  is the identity on  $S$ , so because  $S$  is finite  $f_z$  and  $g_z$  must be inverse bijections on it, and therefore they commute. □

**Theorem 3.5** (3994 does not imply 3588 for infinite models). *eq3994,eq3588,models-def InfModel.Equation3994notimpliesEquation3588Thereexistsamagmawhichsatisfies??andnot??.*

*Proof.* Consider  $\mathbb{N}$ , with  $x \diamond y$  defined as  $x \oplus y$  (bitwise XOR) if  $x$  and  $y$  are even,  $y + 2$  if only  $y$  is even,  $x - 2$  if only  $x$  is even, and 0 if both are odd. Note that the range of the operation is the set of even naturals. ?? is satisfied, because for even  $z$  we get  $z \oplus (x \diamond y) \oplus z = x \diamond y$  and for odd  $z$  we get  $(x \diamond y) + 2 - 2 = x \diamond y$ . Setting  $x = y = z = 1$ , ?? isn't satisfied. □

The following result was established in [?]:

**Theorem 3.6** (Austin's finite model theorem). *models-def Any law with at most two variables has a non-trivial finite model.*

*Proof.* If neither side of the law is a single variable then the zero law  $x \diamond y = 0$  will work, so one can assume the law takes the form  $x = f(x, y)$ . Consider a finite field  $F$  with the operation  $x \diamond y := ax + by$  for some coefficients  $a, b \in F$ . Then the law becomes a pair of equations  $P(a, b) = 0$ ,  $Q(a, b) = 1$  in the coefficients for some polynomials  $P, Q$  with integer coefficients, which one can verify to not divide each other (they have the same degree, and do not have the same set of non-zero monomials). From Bezout's theorem, this equation has a solution in some field, and hence by the Lefschetz principle it has a solution in a finite field.  $\square$



## Chapter 4

# General implications

We will be interested in seeing which laws imply which other laws, in the sense that magmas obeying the former law automatically obey the latter. We will also be interested in *anti-implications* showing that one law does *not* imply another, by producing examples of magmas that obey the former law but not the latter. Here is a formal definition.

**Definition 4.1** (Implication). *models-def* A law  $E$  is said to *imply* another law  $E'$  if  $\{E\} \models E'$ , or equivalently:

$$G \models w \simeq w' \implies G \models w'' \simeq w''' \text{ for all magmas } G$$

Two laws are said to be *equivalent* if they imply each other.

**Lemma 4.2** (Pre-order). *impl* If we define  $E \leq E'$  if  $E$  implies  $E'$ , then this is a pre-order on the set of laws, and equivalence is an equivalence relation.

Note that we view the stronger law as less than or equal to the weaker law. This is because the class of magmas that obey the stronger law is a subset of the class of magmas that obey the weaker law. It is also consistent with the conventions of Lean's Mathlib.

*Proof.* Trivial. □

Implications between the laws from ?? are depicted in ??.

**Lemma 4.3** (Maximal element). *pre-orderLaw.MagmaLaw.Equation1\_maximalTheLaw0*  $\simeq 0$  is the maximal element in this pre-order.

*Proof.* Trivial. □

**Lemma 4.4** (Minimal element). *pre-orderLaw.MagmaLaw.Equation2\_minimalTheLaw0*  $\simeq 1$  is the minimal element in this pre-order.

*Proof.* Trivial. □

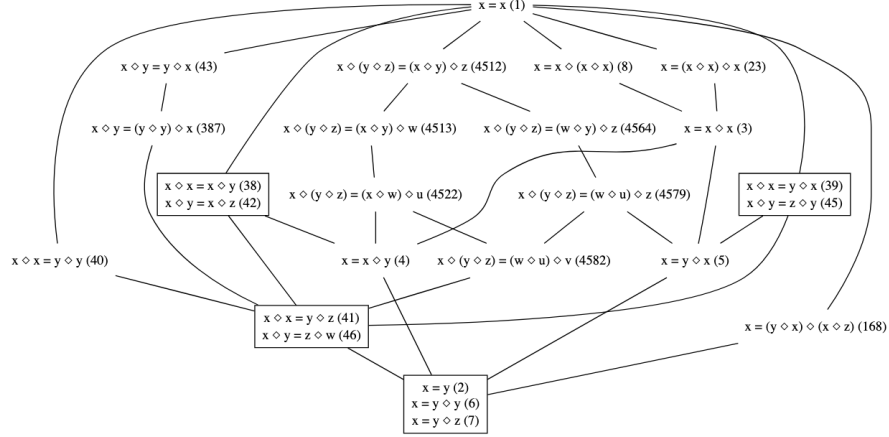


Figure 4.1: Implications between the above equations, displayed as a Hasse diagram.

Every magma  $G$  has a *reversal*  $G^{\text{op}}$ , formed by replacing the magma operation  $\diamond$  with its opposite  $\diamond^{\text{op}} : (x, y) \mapsto y \diamond x$ . There is a natural isomorphism between these magmas, which induces an involution  $w \mapsto w^{\text{op}}$  on words  $w \in M_X$ . Every law  $w \simeq w'$  then has a *dual*  $w^{\text{op}} \simeq (w')^{\text{op}}$ .

For instance, the dual of the law  $0 \diamond 1 = 0 \diamond 2$  is  $1 \diamond 0 = 2 \diamond 0$ , which after relabeling is  $0 \diamond 1 = 2 \diamond 1$ . A list of equations and their duals can be found here. Of the 4694 equations under consideration, 84 are self-dual, leaving 2305 pairs of dual equations.

The pre-ordering on laws has a duality symmetry:

**Lemma 4.5** (Duality of laws). *Law.MagmaLaw.implies<sub>if</sub> dualpre – orderThelaw  $w \simeq w'$  implies  $w'' \simeq w'''$ , if and only if  $w^{\text{op}} \simeq (w')^{\text{op}}$  implies  $w''^{\text{op}} \simeq (w''')^{\text{op}}$ .*

*Proof.* This follows from the fact that a magma  $G$  satisfies a law  $w \simeq w'$  if and only if  $G^{\text{op}}$  satisfies  $w^{\text{op}} \simeq (w')^{\text{op}}$ .  $\square$

Some equational laws can be “diagonalized”:

**Theorem 4.6** (Diagonalization). *An equational law of the form*

$$F(x_1, \dots, x_n) = G(y_1, \dots, y_m), \quad (4.1)$$

*where  $x_1, \dots, x_n$  and  $y_1, \dots, y_m$  are distinct elements of the alphabet, implies the diagonalized law*

$$F(x_1, \dots, x_n) = F(x'_1, \dots, x'_n).$$

*where  $x'_1, \dots, x'_n$  are distinct from  $x_1, \dots, x_n$ . In particular, if  $G(y_1, \dots, y_m)$  can be viewed as a specialization of  $F(x'_1, \dots, x'_n)$ , then these two laws are equivalent.*

*Proof.* From two applications of ?? one has

$$F(x_1, \dots, x_n) = G(y_1, \dots, y_m)$$

and

$$F(x'_1, \dots, x'_n) = G(y_1, \dots, y_m)$$

whence the claim.  $\square$

Thus for instance, ?? is equivalent to ??.

**Theorem 4.7** (Laws implied by the constant law). *impl,eq46 If  $w, w'$  each have order at least one, then the law  $w \simeq w'$  is implied by the constant law (?). If exactly one of  $w, w'$  has order zero, and the law  $w \simeq w'$  is not implied by the constant law.*

*Proof.* Routine.  $\square$

**Theorem 4.8** (Criterion for implication). *implLaw.MagmaLaw.SameCount.deriveIf  $w \simeq w'$  is such that every variable appears the same number of times in both  $w$  and  $w'$ , and  $w \simeq w'$  implies another law  $w'' \simeq w'''$ , then every variable appears the same number of times in both  $w''$  and  $w'''$ .*

*Proof.* Consider the magma MS of multisets over an arbitrary set  $A$  (which can be seen as finitely supported maps  $A \rightarrow \mathbb{N}$ ), with the multiset addition law  $+$ . By hypothesis, this magma obeys  $w \simeq w'$ , and hence  $w'' \simeq w'''$ , giving the claim by comparing the orders of the elements of  $A$  appearing in  $w''$  and  $w'''$  in this magma.  $\square$

## Chapter 5

# Implications between selected laws

We collect here some notable implications between the the selected laws in ??.

By ??, every implication can basically be established by a finite number of rewrites. In most cases, the sequence of rewrites is quite straightforward, and the implication is very easy, but we record some less obvious examples.

**Theorem 5.1** (387 implies 43). *eq387,eq43Subgraph.Equation387;implies\_Equation43??implies??.*

*Proof.* (From MathOverflow). By ??, one has the law

$$(x \diamond x) \diamond y = y \diamond x. \quad (5.1)$$

Specializing to  $y = x \diamond x$ , we conclude

$$(x \diamond x) \diamond (x \diamond x) = (x \diamond x) \diamond x$$

and hence by another application of ?? we see that  $x \diamond x$  is idempotent:

$$(x \diamond x) \diamond (x \diamond x) = x \diamond x. \quad (5.2)$$

Now, replacing  $x$  by  $x \diamond x$  in ?? and then using ?? we see that

$$(x \diamond x) \diamond y = y \diamond (x \diamond x)$$

so in particular  $x \diamond x$  commutes with  $y \diamond y$ :

$$(x \diamond x) \diamond (y \diamond y) = (y \diamond y) \diamond (x \diamond x). \quad (5.3)$$

Also, from two applications of ?? one has

$$(x \diamond x) \diamond (y \diamond y) = (y \diamond y) \diamond x = x \diamond y.$$

Thus ?? simplifies to  $x \diamond y = y \diamond x$ , which is ??. □

**Theorem 5.2** (29 equivalent to 14). *eq29, eq14 Subgraph. Equation29 implies Equation14 ?? is equivalent to ??.*  
This result was posed as Problem A1 from Putnam 2001.

*Proof.* duality By ?? it suffices to show that ?? implies ?. From ?? one has

$$x = ((x \diamond y) \diamond x) \diamond (x \diamond y)$$

and also

$$y = (x \diamond y) \diamond x$$

giving  $x = y \diamond (x \diamond y)$ , which is ??. □

**Theorem 5.3** (14 implies 29). *eq29, eq14 Subgraph. Equation14 implies Equation29 ?? implies ??.*  
This result was posed as Problem A1 from Putnam 2001.

*Proof.* □

The following result was Problem A4 on Putnam 1978.

**Theorem 5.4** (3744 implies 3722, 381). *eq3744, eq3722, eq381 Subgraph. Equation3744 implies Equation3722, Su*

*Proof.* By hypothesis, one has

$$x \diamond y = (x \diamond z) \diamond (w \diamond y)$$

for all  $x, y, z, w$ . Various specializations of this give

$$x \diamond y = (x \diamond z) \diamond (y \diamond y) \tag{5.4}$$

$$x \diamond z = (x \diamond z) \diamond (x \diamond z) \tag{5.5}$$

$$(x \diamond z) \diamond y = ((x \diamond z) \diamond (x \diamond z)) \diamond (y \diamond y). \tag{5.6}$$

?? gives ??, while ??, ??, ?? gives

$$x \diamond y = (x \diamond z) \diamond y$$

which is ??. □

**Theorem 5.5** (1689 is equivalent to 2). *eq1689, eq2 Subgraph. Equation1689 implies Equation2, Subgraph. Equat*

*Proof.* The implication of ?? from ?? is trivial. The converse is a surprisingly long chain of implications; see pages 326–327 of [?]. The initial law

$$x = (y \diamond x) \diamond ((x \diamond z) \diamond z)$$

is used to obtain, in turn,

$$x \diamond (((x \diamond y) \diamond y) \diamond z) \diamond z = (x \diamond y) \diamond y,$$

$$(x \diamond (y \diamond z)) \diamond (z \diamond ((z \diamond w) \diamond w)) = y \diamond z,$$

$$x \diamond (y \diamond ((y \diamond z) \diamond z)) = (x \diamond y) \diamond y,$$

$$\begin{aligned}
& ((x \diamond (y \diamond z)) \diamond z) \diamond z = y \diamond z, \\
& (x \diamond (y \diamond (z \diamond w))) \diamond (z \diamond w) = y \diamond (z \diamond w), \\
& (x \diamond (y \diamond z)) \diamond (y \diamond z) = x \diamond (y \diamond z), \\
& ((x \diamond y) \diamond ((y \diamond z) \diamond z)) \diamond ((y \diamond z) \diamond z) = y, \\
& ((x \diamond y) \diamond ((y \diamond z) \diamond z)) \diamond ((y \diamond z) \diamond z) = ((x \diamond ((x \diamond y) \diamond ((y \diamond z) \diamond z))) \diamond ((y \diamond z) \diamond z)) \diamond ((y \diamond z) \diamond z), \\
& x \diamond ((x \diamond y) \diamond y) = x, \\
& x \diamond (x \diamond (y \diamond z)) = x, \\
& (x \diamond y) \diamond y = x \diamond y, \\
& (x \diamond x) \diamond x = x, \\
& (x \diamond y) \diamond y = y, \\
& x \diamond y = y.
\end{aligned}$$

□

The following result was established in [?].

**Theorem 5.6** (Consequences of 1571). *eq1571, eq2662, eq40, eq23, eq8, eq16, eq14, eq43, eq4512Subgraph.Equation1571impliesEquation2662, Subgraph.Equation1571impliesEquation40, S*

*Proof.* Suppose that a magma  $G$  obeys ??, thus

$$x = (y \diamond z) \diamond (y \diamond (x \diamond z)). \quad (5.7)$$

$$x = ((x \diamond y) \diamond (x \diamond y)) \diamond ((x \diamond y) \diamond (x \diamond (x \diamond y)))$$

and

$$x = (x \diamond y) \diamond (x \diamond (x \diamond y))$$

whence

$$x = ((x \diamond y) \diamond (x \diamond y)) \diamond x$$

which is ?. This gives

$$y = ((y \diamond z) \diamond (y \diamond z)) \diamond y$$

while from ?? one has

$$(y \diamond z) \diamond (y \diamond z) = (x \diamond y) \diamond (x \diamond ((y \diamond z) \diamond (y \diamond z) \diamond y))$$

whence

$$(x \diamond y) \diamond (x \diamond y) = (y \diamond z) \diamond (y \diamond z).$$

This implies that  $(x \diamond y) \diamond (x \diamond y)$  does not depend on  $x$ , or on  $y$ , hence is equal to some constant  $e$ :

$$(x \diamond y) \diamond (x \diamond y) = e.$$

From ?? the magma operation is surjective, hence

$$x \diamond x = e \quad (5.8)$$

which gives ?. Applying ? with  $x = y = z$  we conclude

$$x = e \diamond (x \diamond e)$$

while if we instead take  $y = z = e$  we have

$$x = e \diamond (e \diamond (x \diamond e))$$

hence

$$x = e \diamond x$$

and then also

$$x = x \diamond e$$

from which we readily conclude ?, ?; thus  $e$  is an identity element. From ? with  $z = e$  we now have

$$x = y \diamond (y \diamond x) \quad (5.9)$$

which is ?. If instead we take  $y = e$  we have

$$x = z \diamond (x \diamond z) \quad (5.10)$$

which is ?. So if we substitute  $z = x \diamond y$  and use ? we obtain

$$x = (x \diamond y) \diamond y$$

and hence

$$y \diamond x = y \diamond ((x \diamond y) \diamond y) = x \diamond y$$

thanks to ?. This gives ?, thus  $G$  is now commutative. From ? once more one has

$$x \diamond (y \diamond z) = (y \diamond x) \diamond (z \diamond ((x \diamond (y \diamond z)) \diamond x))$$

which one can simplify using commutativity and ? (or ?) to eventually obtain

$$x \diamond (y \diamond z) = (x \diamond y) \diamond z$$

which is ?.  $G$  is now commutative and associative, and every element is its own inverse and of exponent 2, hence is an abelian group thanks to ?, so  $G$  is an abelian group of exponent 2 as claimed. The converse is easily verified.  $\square$

**Theorem 5.7** (953 is equivalent to 2). *eq953, eq2Subgraph.Equation953 implies Equation2?? is equivalent to ??.*

*Proof.* It suffices to show that ? implies ?. Pick an element 0 of  $G$  and define  $1 = 0 \diamond 0$  and  $2 = 1 \diamond 1$  (we do not require 0, 1, 2 to be distinct). From ? with  $x = z = 0$  we have

$$0 = y \diamond 2.$$

If we then apply ? with  $z = 1$  we conclude that

$$x = y \diamond 0$$

for all  $x, y$ , from which one concludes  $x = x'$  for any  $x, x' \in G$ , giving ?.  $\square$

**Theorem 5.8** (Sheffer stroke axiom). *eq345169 Definition ?? axiomatizes the Sheffer stroke operation  $x \diamond y = \overline{xy}$  in a Boolean algebra.*

*Proof.* See [?]. In fact this is the shortest law with this property.

A sketch of proof follows. One can easily verify that the Sheffer stroke operation obeys this law. Conversely, if this law holds, then automated theorem provers can show that the three Sheffer axioms

$$(x \diamond x) \diamond (x \diamond x) = x$$

$$x \diamond (y \diamond (y \diamond y)) = x \diamond x$$

$$(x \diamond (y \diamond z)) \diamond (x \diamond (y \diamond z)) = ((y \diamond y) \diamond x) \diamond ((z \diamond z) \diamond x)$$

are satisfied. A classical result of Sheffer [?] then allows one to conclude.  $\square$

A *natural central groupoid* is, up to isomorphism, a magma with carrier  $S \times S$  for some set  $S$  and operation

$$(a, b) \diamond (c, d) = (b, c).$$

These are examples of central groupoids (??).

**Theorem 5.9** (Natural central groupoid axiom). *eq26302 Definition ?? characterizes natural central groupoids.*

*Proof.* See [?, Theorem 5]. The proof is quite lengthy; a sketch is as follows. It is easy to see that natural central groupoids obey ??. Conversely, if this law holds, then

$$\begin{aligned} (y \diamond z) \diamond (z \diamond w) &= ((x \diamond ((w \diamond (y \diamond z)) \diamond w)) \diamond ((y \diamond z) \diamond w)) \diamond (z \diamond w) \\ &= z \end{aligned}$$

so we have a central groupoid. Setting  $y = (t \diamond t) \diamond t$ ,  $z = t \diamond (t \diamond t)$ ,  $w = t \diamond t$  in ?? we also obtain

$$(x \diamond t) \diamond t = (t \diamond t) \diamond t.$$

Using the notation

$$x^{(1)} := (x \diamond x) \diamond x, \quad x^{(2)} := x \diamond (x \diamond x)$$

we then have

$$\begin{aligned} x \diamond t &= ((x \diamond x) \diamond (x \diamond t)) \diamond ((x \diamond t) \diamond t) \\ &= x \diamond t^{(1)}. \end{aligned}$$

A lengthy computer-assisted argument then gave the dual identity

$$t^{(2)} \diamond x = t \diamond x$$



Together, these give

$$x^{(2)} \diamond y^{(1)} = x \diamond y.$$

Multiplying on the left by  $x = x^{(1)} \diamond x^{(2)}$ , one can conclude that

$$x^{(2)} = x \diamond (x \diamond y).$$

One then has

$$\begin{aligned} (x \diamond y)^{(1)} &= ((y \diamond x) \diamond (x \diamond y)) \diamond (x \diamond y) \\ &= x \diamond (x \diamond y) \\ &= x^{(2)} \end{aligned}$$

and a similar argument gives

$$(x \diamond y)^{(2)} = y^{(1)}.$$

Since  $(x \diamond x)^{(1)} = x^{(2)}$  and  $(x \diamond x)^{(2)} = x^{(1)}$ , we conclude that  $x^{(1)}$  and  $x^{(2)}$  are idempotent. Since  $x = x^{(1)} \diamond x^{(2)}$ , we see that every  $x$  is the product of two idempotents. One can show that this representation is unique, and gives a canonical identification with a natural central groupoid.  $\square$

## Chapter 6

# Selected magmas

Each magma can be used to establish anti-implications: if  $\Gamma$  is the set of all laws obeyed by a magma  $G$ , then we have  $\neg E \leq E'$  whenever  $E \in \Gamma$  and  $E' \notin \Gamma$ . Large numbers of implications can already be obtained from

- All magmas of order at most 4, up to isomorphism (of which there are 178, 985, 294);
- All commutative magmas of order 5, up to isomorphism **determine their count**;
- Cyclic groups  $\mathbb{Z}/N\mathbb{Z}$  with  $2 \leq N \leq 12$  and  $x \circ y = ax^2 + bxy + cy^2 + dx + ey$  for randomly chosen  $a, b, c, d, e$ .
- There are only 1410 distinct cancellative magmas of order 5 (up to isomorphism), and Mace4 can generate all of them in under 20 seconds. A shell script to do this is available [here](#). A magma is cancellative if  $xy = xz$  implies  $y = z$  and  $yx = zx$  implies  $y = z$ .

Some other magmas have been used to establish counterexamples:

- The cyclic group  $\mathbb{Z}/6\mathbb{Z}$  with the addition law.
- The natural numbers with law  $x \circ y = x + 1$ .
- The natural numbers with law  $x \circ y = xy + 1$ .
- The reals with  $x \circ y = (x + y)/2$ .
- The natural numbers with  $x \circ x$  equal to  $x$  when  $x = y$  and  $x + 1$  otherwise.
- The set of strings with  $x \circ y$  equal to  $y$  when  $x = y$  or when  $x$  ends with  $yy$ , or  $xy$  otherwise (see this [Zulip thread](#)).
- Vector spaces  $\mathbb{F}_2^n$  over  $\mathbb{F}_2$ , which obey ?? (and hence all the subsequent laws mentioned in ??).

- Knuth's construction [?] of a central groupoid (??) as follows. Let  $S$  be a (finite) set with a distinguished element  $0$ , and a binary operation  $*$  such that  $x * 0 = 0$  and  $0 * x = x$  for all  $x$ , and for each  $x, y$  there is a unique  $z$  with  $x * z = y$ . One can then define a central groupoid on  $S \times S$  by defining  $(a, b) \diamond (c, d)$  to equal  $(b, c)$  if  $c, d \neq 0$ ;  $(b, e)$  if  $b * e = c$  is non-zero and  $d = 0$ ; and  $(a * b, 0)$  if  $c = 0$ . One such example in [?] is when  $S = \{0, 1, 2\}$  with  $1 * 1 = 2 * 1 = 2$  and  $1 * 2 = 2 * 2 = 1$ .
- Cancellative magmas of orders 7 to 9, found by hand-guided search using various solvers.
- Two magmas of cardinality 8 were constructed by Z3.
- A large number of ad-hoc finite magmas were constructed using the Vampire theorem prover.
- Linear magmas  $x \diamond y = ax + by$  on various fields, such as  $\mathbb{F}_p$  for small primes  $p$ , have also been used to establish counterexamples. One such choice is  $(p, a, b) = (11, 1, 7)$ .
- A variation of the translation-invariant magma construction which resolved the Asterix / Obelix anti-implication is used to show that ?? does not imply ??.

## Chapter 7

# The Asterix equation

A *translation-invariant magma* is a magma whose carrier  $G$  is an abelian group  $G = (G, +)$ , and whose magma operation takes the form

$$y \diamond x = x + f(x - y)$$

for some function  $f : G \rightarrow G$ . Thus the translations on  $G$  become magma isomorphisms.

For translation-invariant magmas, an equational law simplifies to a univariate functional equation. For instance, writing  $x = y + h$ , we have

$$y \diamond x = x + f(h)$$

$$x \diamond (y \diamond x) = x + f(h) + f^2(h)$$

$$y \diamond (x \diamond (y \diamond x)) = x + f(h) + f^2(h) + f(h + f(h) + f^2(h))$$

where  $f^2 = f \circ f$ , so the Asterix equation (??) for such magmas simplifies to the univariate functional equation

$$f(h) + f^2(h) + f(h + f(h) + f^2(h)) = 0 \quad (7.1)$$

for  $h \in G$ .

This equation has some degenerate solutions, for instance we can take  $f(h) = c$  for any constant  $c$  of order 3 in  $G$ . It is challenging to construct more interesting solutions to this equation; however, we can do this if  $G = \mathbb{Z}$  by a greedy algorithm. We need the following technical definition.

**Definition 7.1.** A *partial solution*  $(E_0, E_1, E_2, f)$  to (??) consists of nested finite sets

$$E_0 \subset E_1 \subset E_2 \subset \mathbb{Z}$$

together with a function  $f : E_1 \rightarrow E_2$  with the following properties:

- (a) If  $h \in E_0$ , then  $f(h) \in E_1$ , so that  $f^2(h)$  is well-defined as an element of  $E_2$ ; furthermore,  $h + f(h) + f^2(h)$  lies in  $E_1$ , so that the left-hand side of (??) makes sense; and (??) holds.

(b) The function  $f$  is a bijection from  $E_1 \setminus E_0$  to  $E_2 \setminus E_1$ .

We partially order the space of partial solutions to (??) by writing  $(E_0, E_1, E_2, f) \leq (E'_0, E'_1, E'_2, f')$  if the following properties hold:

- $E_i \subset E'_i$  for  $i = 0, 1, 2$ .
- $f$  agrees with  $f'$  on  $E_0$ .

When this occurs we say that the partial solution  $(E'_0, E'_1, E'_2, f')$  *extends* the partial solution  $(E_0, E_1, E_2, f)$ .

We define the *empty partial solution*  $(E_0, E_1, E_2, f)$  by setting  $E_0 = E_1 = E_2$  to be the empty set, and  $f$  to be the empty function; it is the minimal element of the above partial order.

We have the following iterative construction, that lets us add arbitrary elements to the core domain  $E_0$ :

**Lemma 7.2** (Enlarging a partial solution). *partial-solution Let  $(E_0, E_1, E_2, f)$  be a partial solution to (??), and let  $h$  be an element of  $\mathbb{Z}$  that does not lie in  $E_0$ . Then there exists a partial solution  $(E'_0, E'_1, E'_2, f')$  to (??) that extends  $(E_0, E_1, E_2, f)$ , such that  $h \in E'_0$ .*

*Proof.* Because  $f$  maps  $E_1 \setminus E_0$  bijectively to  $E_2 \setminus E_1$ , there are three cases:

- $h$  is equal to an element  $h_0$  of  $G \setminus E_2$ .
- $h$  is equal to an element  $h_0$  of  $E_1 \setminus E_0$ .
- $h$  is equal to  $h_1 = f(h_0)$  for some  $h_0 \in E_1 \setminus E_0$ , so that  $h_1 \in E_2 \setminus E_1$ .

We deal with these three cases in turn.

First suppose that  $h = h_0 \in G \setminus E_2$ . We perform the following construction.

- Choose an element  $h_1 \in \mathbb{Z}$  that does not lie in  $E_2 \cup \{h_0\}$ ; this is possible because  $E_2$  is finite.
- Choose an element  $h_2 \in \mathbb{Z}$  such that  $h_2, h_0 + h_1 + h_2$ , and  $-h_1 - h_2$  are all distinct from each other and lie outside of  $E_2 \cup \{h_0, h_1\}$ ; this is possible because  $E_2$  is finite.
- Promote  $h_0$  to  $E_0$ , promote  $h_1, h_0 + h_1 + h_2$  to  $E_1$ , and promote  $h_2, -h_1 - h_2$  to  $E_2$ , creating new sets

$$\begin{aligned} E'_0 &:= E_0 \cup \{h_0\} \\ E'_1 &:= E_1 \cup \{h_0, h_1, h_0 + h_1 + h_2\} \\ E'_2 &:= E_2 \cup \{h_0, h_1, h_0 + h_1 + h_2, h_2, -h_1 - h_2\}. \end{aligned}$$

Clearly we still have nested finite sets  $E'_0 \subset E'_1 \subset E'_2$ .

- Extend  $f : E_1 \rightarrow E_0$  to a function  $f' : E'_1 \rightarrow E'_0$  by defining

$$\begin{aligned} f'(h_0) &:= h_1 \\ f'(h_1) &:= h_2 \\ f'(h_0 + h_1 + h_2) &:= -h_1 - h_2 \end{aligned}$$

while keeping  $f'(h) = f(h)$  for all  $h \in E_1$ .

It is then a routine matter to verify that  $(E'_0, E'_1, E'_2, f')$  is a partial solution to (??) extending  $(E_0, E_1, E_2, f)$  and that  $E'_0$  contains  $h_0$ , as required.

Now suppose that  $h = h_0 \in E_1 \setminus E_0$ , then the quantity  $h_1 := f(h_0)$  lies in  $E_2 \setminus E_1$ . We perform the following variant of the above construction:

- Choose an element  $h_2 \in \mathbb{Z}$  such that  $h_2, h_0 + h_1 + h_2$ , and  $-h_1 - h_2$  are all distinct and lie outside of  $E_2$ . This is possible because  $E_2$  is finite.
- Promote  $h_0$  to  $E_0$ , promote  $h_1$  and  $h_0 + h_1 + h_2$  to  $E_1$ , and promote  $h_2, -h_1 - h_2$  to  $E_2$ , thus creating new sets

$$\begin{aligned} E'_0 &:= E_0 \cup \{h_0\} \\ E'_1 &:= E_1 \cup \{h_1, h_0 + h_1 + h_2\} \\ E'_2 &:= E_2 \cup \{h_0 + h_1 + h_2, h_2, -h_1 - h_2\}. \end{aligned}$$

Clearly we still have nested finite sets  $E'_0 \subset E'_1 \subset E'_2$ .

- Extend  $f : E_1 \rightarrow E_0$  to a function  $f' : E'_1 \rightarrow E'_0$  by defining

$$\begin{aligned} f'(h_1) &:= h_2 \\ f'(h_0 + h_1 + h_2) &:= -h_1 - h_2 \end{aligned}$$

while keeping  $f'(h) = f(h)$  for all  $h \in E_1$ .

It is then a routine matter to verify that  $(E'_0, E'_1, E'_2, f')$  is a partial solution to (??) extending  $(E_0, E_1, E_2, f)$  and that  $E'_0$  contains  $h_0$ , as required.

Finally, suppose that  $h = h_1 = f(h_0)$  for some  $h_0 \in E_1 \setminus E_0$ , so that  $h_1 \in E_2 \setminus E_1$ . Then we perform the following algorithm.

- Choose an element  $h_2 \in \mathbb{Z}$  such that  $h_2, h_0 + h_1 + h_2$ , and  $-h_1 - h_2$  are all distinct and lie outside of  $E_2$ . This is possible because  $E_2$  is finite.
- Choose an element  $h_3 \in \mathbb{Z}$  such that  $h_3, h_1 + h_2 + h_3$ , and  $-h_2 - h_3$  are all distinct and lie outside of  $E_2 \cup \{h_2, h_0 + h_1 + h_2, -h_1 - h_2\}$ . This is possible because  $E_2$  is finite.
- Promote  $h_0, h_1$  to  $E_0$ , promote  $h_2, h_0 + h_1 + h_2, h_1 + h_2 + h_3$  to  $E_1$ , and promote  $h_3, -h_1 - h_2, -h_2 - h_3$  to  $E_2$ , creating new sets

$$\begin{aligned} E'_0 &:= E_0 \cup \{h_0, h_1\} \\ E'_1 &:= E_1 \cup \{h_1, h_2, h_0 + h_1 + h_2, h_1 + h_2 + h_3\} \\ E'_2 &:= E_2 \cup \{h_2, h_3, h_0 + h_1 + h_2, h_1 + h_2 + h_3, -h_1 - h_2, -h_2 - h_3\}. \end{aligned}$$

Clearly we still have nested finite sets  $E'_0 \subset E'_1 \subset E'_2$ .

- Extend  $f : E_1 \rightarrow E_0$  to a function  $f' : E'_1 \rightarrow E'_0$  by defining

$$\begin{aligned} f'(h_1) &:= h_2 \\ f'(h_0 + h_1 + h_2) &:= -h_1 - h_2 f'(h_2) &:= h_3 \\ f'(h_1 + h_2 + h_3) &:= -h_2 - h_3 \end{aligned}$$

while keeping  $f'(h) = f(h)$  for all  $h \in E_1$ .

It is then a routine matter to verify that  $(E'_0, E'_1, E'_2, f')$  is a partial solution to (??) extending  $(E_0, E_1, E_2, f)$  and that  $E'_0$  contains  $h_0$ , as required.  $\square$

**Corollary 7.3.** *partial-solution Every partial solution  $(E_0, E_1, E_2, f)$  to (??) can be extended to a full solution  $\tilde{f} : \mathbb{Z} \rightarrow \mathbb{Z}$ .*

*Proof.* iteration If we arbitrarily well-order the integers, and iterate ?? to add the least element of  $\mathbb{Z} \setminus E_0$  in this well-ordering to  $E_0$ , we obtain an increasing sequence  $(E_0^{(n)}, E_1^{(n)}, E_2^{(n)}, f^{(n)})$  of partial solutions to (??), where the  $E_0^{(n)}$  exhaust  $\mathbb{Z}$ :  $\bigcup_{n=1}^{\infty} E_0^{(n)} = \mathbb{Z}$ . Taking limits, we obtain a full solution  $\tilde{f}$ .  $\square$

**Corollary 7.4.** *There exists a solution  $f : \mathbb{Z} \rightarrow \mathbb{Z}$  to (??) such that the map  $h \mapsto h + f(h)$  is not injective.*

*Proof.* Select integers  $h_0, h_1, h_2, h'_0, h'_1, h'_2$  such that the quantities

$$h_0, h_1, h_2, h_0 + h_1 + h_2, -h_1 - h_2, h'_0, h'_1, h'_2, h'_0 + h'_1 + h'_2, -h'_1 - h'_2$$

are all distinct, but such that

$$h_0 + h_1 = h'_0 + h'_1$$

(there are many assignments of variables that accomplish this). Then set

$$\begin{aligned} E_0 &:= \{h_0, h'_0\} \\ E_1 &:= E_0 \cup \{h_1, h'_1, h_0 + h_1 + h_2, h'_0 + h'_1 + h'_2\} \\ E_2 &:= E_2 \cup \{-h_1 - h_2, -h'_1 - h'_2\} \end{aligned}$$

and define  $f : E_1 \rightarrow E_2$  by the formulae

$$\begin{aligned} f(h_0) &:= h_1 \\ f(h_1) &:= h_2 \\ f(h_0 + h_1 + h_2) &:= -h_1 - h_2 \\ f(h'_0) &:= h'_1 \\ f(h'_1) &:= h'_2 \\ f(h'_0 + h'_1 + h'_2) &:= -h'_1 - h'_2. \end{aligned}$$

One can then check that  $(E_0, E_1, E_2, f)$  is a partial solution to (??), and by construction  $h \mapsto h + f(h)$  is not injective on  $E_1$ . Using ?? to extend this partial solution to a full solution, we obtain the claim.  $\square$

**Corollary 7.5** (Asterix does not imply Obelix). *eq65,eq1491 There exists a magma obeying the Asterix law (??) with carrier  $\mathbb{Z}$  such that the left-multiplication maps  $L_y : x \mapsto y \diamond x$  are not injective for any  $y \in \mathbb{Z}$ . In particular, it does not obey the Obelix law (??).*

*Proof.* no-inject Note that  $L_y(y+h) = y+h+f(h)$ , so the injectivity of the left-multiplication maps is equivalent to the injectivity of the map  $h \mapsto h+f(h)$ . The non-injectivity then follows from ??. Note that the Obelix law clearly expresses  $x$  as a function of  $y$  and  $L_y x = y \diamond x$ , forcing injectivity of left-multiplication, so the Obelix law fails.  $\square$

On the other hand, for finite magmas the situation is different:

**Proposition 7.6** (Asterix implies Obelix for finite magmas). *eq65,eq1491 Any finite magma obeying the Asterix law (??) also is left-cancellative and obeys the Obelix law (??).*

*Proof.* From ?? we see the map  $z \mapsto y \diamond z$  is surjective, hence injective on a finite magma; thus the magma is left-cancellative. Replacing  $x$  by  $y \diamond x$  in this law, we see that

$$y \diamond x = y \diamond ((y \diamond x) \diamond (y \diamond (y \diamond x)));$$

using injectivity, we conclude

$$x = (y \diamond x) \diamond (y \diamond (y \diamond x))$$

which is ??.  $\square$

We remark that a very similar argument shows that a finite magma that obeys the Obelix law has  $z \mapsto y \diamond z$  injective, hence surjective, and then obeys the Asterix law.



## Chapter 8

# Greedy algorithm constructions

One way to construct magmas obeying specific laws (such as the Asterix law, ??) is via a greedy algorithm construction. Here, it is best to work with *partially defined* magma operations on some carrier  $G$ . This can be interpreted as a ternary relation  $R(x, y, z)$  on three variables  $x, y, z \in G$  which obeys the “vertical line test”:

(VLT) If  $R(x, y, z)$  and  $R(x, y, z')$  both hold for some  $x, y, z, z' \in G$ , then  $z = z'$ .

Such an operation is then associated (via a one-to-one correspondence) to a partially defined operation  $\diamond : StoG$  for some  $S \subset G \times G$ , with  $R(x, y, z)$  holding if and only if  $x \diamond y$  is well-defined (i.e.,  $(x, y) \in S$ ) and equal to  $z$ . By abuse of notation, we shall also refer to  $R$  as a partially defined magma operation. Genuine magmas then correspond to the special case where  $S = G \times G$ , that is to say  $x \diamond y$  is well-defined for all  $x, y \in G$ .

Given a word  $w(x_1, \dots, x_n)$  in variables  $x_1, \dots, x_n$  (so  $w$  is an element of the free magma on  $n$  generators), we can say that  $w(x_1, \dots, x_n)$  is *well-defined* with respect to a partially defined magma operation  $R$  if it can be fully evaluated using  $R$ . For instance, the word  $(x \diamond y) \diamond z$  is well-defined if there exists  $w, u \in G$  such that  $R(x, y, w)$  and  $R(w, z, u)$  both hold, in which case  $(x \diamond y) \diamond z$  evaluates to  $u$ . Note from the axiom (VLT) that this evaluation is unique, if it exists. Of course, in a genuine magma, all expressions are well-defined. We say that an expression  $w(x_1, \dots, x_n)$  is *almost well-defined* if all strict subexpressions of  $w$  are well-defined. For instance,  $(x \diamond y) \diamond z$  is almost well-defined if there exists  $w \in G$  such that  $R(x, y, w)$  holds.

An equational law  $w_1 \simeq w_2$  involving some variables  $x_1, \dots, x_n$  is said to be *locally obeyed* by  $R$  if, whenever  $w_1(x_1, \dots, x_n)$ ,  $w_2(x_1, \dots, x_n)$  are almost well-defined, and one of the two expressions is well-defined and evaluates to some output  $y$ , then the other expression is also well-defined and evaluates to

the same output  $y$ . For instance, in order for  $R$  to locally obey the associative law  $(x \diamond y) \diamond z = x \diamond (y \diamond z)$  (??), we require the following two axioms:

(4512-1) If  $R(x, y, w)$ ,  $R(w, z, u)$ , and  $R(y, z, v)$ , then  $R(x, v, u)$ .

(4512-2) If  $R(y, z, w)$ ,  $R(x, w, u)$ , and  $R(x, y, v)$ , then  $R(v, z, u)$ .

If a law involves a single variable on one side, then we only need one axiom. For instance, the Asterix law (??) is locally obeyed by  $R$  if and only if the following axiom holds:

(65) If  $R(y, x, z)$  and  $R(x, z, u)$ , then  $R(y, u, x)$ .

Note that if the relation  $R$  is associated to a genuine magma operation  $\diamond$ , then it locally obeys a law  $w_1 \simeq w_2$  if and only if the magma operation  $\diamond$  obeys the law  $w_1 \simeq w_2$ . For instance, the relation  $R$  associated to a globally defined magma operation  $\diamond$  obeys (4512-1) and (4512-2) if and only if the magma is associative.

More generally, one can ask for a ternary relation  $R$  to obey some theory  $\Gamma$  of universal laws, using the language of one ternary relation  $R$ , the equality symbol  $=$ , and possibly some constants (we will shortly introduce three constants  $a, b, c$  for this purpose).

Suppose we have a relation  $R$  obeying some theory  $\Gamma$  (for instance, (VLT) together with (65)), but which is only finitely supported (there are only finitely many triples  $(x, y, z)$  for which  $R(x, y, z)$  holds). Then one can find  $a, b \in G$  such that  $a \diamond b$  is currently undefined. If the carrier  $G$  is infinite (e.g., if  $G = \mathbb{N}$ ), one can then find another element  $c$  which is *novel*: it is not equal to  $a, b$ , or any of the  $x, y, z$  for which  $R(x, y, z)$  hold. In other words, the relation  $R$  and the constants  $a, b, c$  obey the following additional axioms:

(novel-1)  $c \neq a$  and  $c \neq b$ .

(novel-2) If  $R(x, y, z)$ , then  $c \neq x$ ,  $c \neq y$ , and  $c \neq z$ .

(undefined)  $R(a, b, x)$  does not hold for any  $x$ .

Let us say that a theory  $\Gamma$  is *greedily extensible* if, whenever  $R$  is a finitely supported ternary relation obeying  $\Gamma$ , and  $a, b, c$  are constants obeying (novel-1), (novel-2), (undefined), then there exists an extension  $R'$  of  $R$ , thus

(extend)  $R(x, y, z) \implies R'(x, y, z)$  for all  $x, y, z$ ,

which is also finitely supported and obeys  $\Gamma$ , and which also obeys the additional axiom

(define)  $R'(a, b, c)$ .

Informally,  $R'$  is formed from  $R$  by “forcing”  $a \diamond b = c$  and then adding other axioms as needed. (Indeed, our construction here can be viewed as a simple analogue of the forcing construction in set theory.)

Observe that if a theory  $\Gamma$  containing (VLT) is greedily extensible, then any finitely supported ternary relation  $R$  obeying  $\Gamma$  on a countably infinite carrier  $G$

can be extended to a globally defined relation obeying  $\Gamma$ , by iteratively selecting the first  $(a, b)$  (in some fixed enumeration of  $G \times G$ ) for which  $a \diamond b$  is undefined, and then selecting a novel element  $c$  to define as  $a \diamond b$ , and applying the greedily extensible property, and then taking a direct limit of the countable sequence of relations thus produced. This gives a flexible way to construct magmas that obey a given theory  $\Gamma$ , but which violate some other law  $w_1 \simeq w_2$ , as the task then reduces to just finding a partial solution  $R$  to  $\Gamma$  and some constants  $x_1, \dots, x_n$  for which the expressions  $w_1(x_1, \dots, x_n), w_2(x_1, \dots, x_n)$  are already well-defined, but not equal to each other.

Unfortunately, most theories are not greedily extensible without further modification. Consider for instance the theory  $\Gamma$  consisting of (VLT) and the Asterix law (65). Given  $a, b, c$  and a finitely supported  $R$  obeying  $\Gamma$  as well as (novel-1), (novel-2), (undefined), we would like to construct a finitely supported  $R'$  obeying  $\Gamma$ , (extend), (define). The naive guess would just be to take the minimal construction

$$R'(x, y, z) \text{ iff } R(x, y, z) \text{ or } (x, y, z) = (a, b, c).$$

This can work, but there is an obstruction: if  $R(w, a, b)$  for some  $w$ , then (65) forces  $R'(w, c, a)$ . So one would have to enlarge the definition of  $R'(x, y, z)$ , to hold true if one of the following statements holds:

- $R(x, y, z)$  holds.
- $(x, y, z) = (a, b, c)$ .
- $(x, y, z) = (w, c, a)$  for some  $w$  with  $R(w, a, b)$ .

This works more often, but there is then a second obstruction: if  $R(b, a, b)$ , then we now have  $R'(b, c, a)$ , and (65) then forces  $R'(a, a, b)$ . So we need to add a fourth item to the above list defining  $R'$ :

- $(x, y, z) = (a, a, b)$ , assuming  $R(b, a, b)$  holds.

But now if we had  $R(a, a, z)$  for some  $z \neq b$ , this would then create a violation of (VLT). To fix this, we need to extend  $\Gamma$  by adding an additional axiom:

- (65') If  $R(y, x, y)$ , then  $R(x, x, y)$ .

With this modification to  $\Gamma$ , if we run the above analysis, we now see that if  $R(b, a, b)$  hold (so that  $R(a, a, b)$  also holds), then (65) will force  $R'(a, c, a)$ ,  $R'(b, c, a)$ , and  $R'(c, c, a)$ . So now the modified definition of  $R'$  is that  $R'(x, y, z)$  holds if one of the following statements holds:

- $R(x, y, z)$  holds.
- $(x, y, z) = (a, b, c)$ .
- $(x, y, z) = (w, c, a)$  for some  $w$  with  $R(w, a, b)$ .
- $(x, y, z) = (a, c, a), (b, c, a),$  or  $(c, c, a)$ , assuming  $R(b, a, b)$  holds.

One can then finally (for instance, with the assistance of a automated theorem prover) verify that if  $R$  is finitely supported and obeys  $\Gamma = (VLT) + (65) + (65')$  and  $a, b, c$  obey (novel-1), (novel-2), (undefined), then the  $R'$  defined above is also finitely supported and obeys  $\Gamma$ , (extend), (define). This shows that the theory  $\Gamma$  is greedily extensible.

Using this, one can for instance find a magma obeying ?? that fails the left-cancellative property

$$y \diamond x = y \diamond x' \implies x = x'$$

or in terms of ternary relations

$$R(y, x, z) \text{ and } R(y, x', z) \implies x = x' \tag{8.1}$$

simply by starting with a partial solution, say on the natural numbers in which (say)  $R(1, 2, 0)$ ,  $R(1, 3, 0)$  are the only situations in which  $R$  holds. One can easily verify that this obeys (VLT) and (65), (65'), but not (??), and so any magma constructed by the above greedy construction will not be left-cancellative. Since the Obelix law ?? forces left-cancellativity, this gives an alternate proof of ??

## Chapter 9

# Equivalence with the constant and singleton laws

85 laws have been shown to be equivalent to the constant law (??), and 815 laws have been shown to be equivalent to the singleton law (??).

These are the laws up to 4 operations that follow from diagonalization of ?? and ??.

To formalize these in Lean, a search was run on the list of equations to discover diagonalizations of these two specific laws: equations of the form  $x = R$  where  $R$  doesn't include  $x$ , and equations of the form  $x \circ y = R$  where  $R$  doesn't include  $x$  or  $y$ .

The proofs themselves all look alike, and correspond exactly to the two steps described in the proof of ??. The Lean proofs were generated semi-manually, using search-and-replace starting from the output of `grep` that found the diagonalized laws.

In the case of the constant law, ?? ( $x \circ x = y \circ z$ ) wasn't detected using this method. It was added manually to the file with the existing proof from the sub-graph project.

## Chapter 10

# Metatheorems from Invariants

For the purposes of this chapter, a *theorem* is a (true) statement about particular equations, for example ‘(387 implies 43)’ is a theorem. A *metatheorem* is a general statement about implications; one can usually get many theorems from a single metatheorem. This chapter is all about generating many interesting metatheorems using a *meta-metatheorem*, called the fundamental property of invariants. If all this is making your head spin, don’t worry. Look at the sections below for examples of metatheorems you can probably agree are both concrete and interesting. Once you have done that, come back here and we will show you how to prove these and other metatheorems using *invariants*.

### 10.1 Invariants

Let  $E, E_1$ , and  $E_2$  be equations. If  $E \Rightarrow E_1$  and  $E_1 \Rightarrow E_2$ , then  $E \Rightarrow E_2$ . Very trivial. Rephrasing this, we see that if  $E \Rightarrow E_1$  and  $E \not\Rightarrow E_2$ , then  $E_1 \not\Rightarrow E_2$ .

Extending this idea, suppose we compute the set of all equations which are implied by  $E$ ; we will call this set  $\mathcal{Y}(E)$  (we use  $\mathcal{Y}$  because this is an example of a Yoneda embedding). Then  $\mathcal{Y}(E)$  is upwards closed, or closed under forward implication: no equation in  $\mathcal{Y}(E)$  can imply an equation not in  $\mathcal{Y}(E)$ . If we know  $\mathcal{Y}(E)$  well, this already settles a potentially large number of implications in the negative.

While computing  $\mathcal{Y}(E)$  for an arbitrary equation  $E$  may seem daunting, for some nice equations we can find *invariants*, which makes the task manageable. An *invariant* for  $E$  is some sort of data associated with expressions  $w$  so that

$$\mathcal{Y}(E) = \{w = w' \mid \text{Invariant}(w) = \text{Invariant}(w')\}$$

If we can find an invariant which is computable for each term  $w$ , then we can easily describe  $\mathcal{Y}(E)$ . The fact that  $\mathcal{Y}(E)$  is upwards closed is rephrased as

follows; this is called **the fundamental property of invariants**. Remember that an invariant is a function taking expressions and outputting some data.

**Meta-metatheorem 10.1** (Fundamental property of invariants). *Let  $I$  be an invariant of  $E$ . If  $w = w'$  implies  $w'' = w'''$  and  $I(w) = I(w')$  (that is,  $E$  implies  $w = w'$ ), then  $I(w'') = I(w''')$ .*

More succinctly, for an invariant  $I$  of  $E$  we must have

$$(w = w' \Rightarrow w'' = w''') \implies (I(w) = I(w') \Rightarrow I(w'') = I(w''')).$$

When using this result, we commonly take the contrapositive: if  $I(w) = I(w')$  and  $I(w'') \neq I(w''')$ , then  $w = w'$  cannot imply  $w'' = w'''$ . Note that the conclusion is independent of the equation  $E$ ; all we need to know is that  $I$  is an invariant.

*Note for category theorists.* Let  $\Pi$  denote the preorder of magma equations ordered by implication. If  $I$  is an invariant then define

$$I(w = w') \begin{cases} \mathbf{true} & \text{if } I(w) = I(w') \\ \mathbf{false} & \text{otherwise} \end{cases}.$$

(In programming languages we would say  $I(w = w')I(w) == I(w')$ ). Let  $\mathbf{Bool} = \{\mathbf{true}, \mathbf{false}\}$  be the poset where  $\mathbf{false} \leq \mathbf{true}$ . Then  $I$  becomes a function  $\Pi \rightarrow \mathbf{Bool}$ , and the fundamental property of invariants just says that this function is monotone, i.e. functorial. Thus for every invariant  $I$  we obtain a functor  $\Pi \rightarrow \mathbf{Bool}$ .

Question 1: Does every functor  $\Pi \rightarrow \mathbf{Bool}$  come from an invariant?

Question 2: What can we say about the category of functors  $\Pi \rightarrow \mathbf{Bool}$ ? Give a nice interpretation of natural transformations between invariants.  $\square$

The fundamental property of invariants is not a theorem, nor a metatheorem: it is a meta-metatheorem, in the sense that it will allow us to get a metatheorem for every invariant we find.

### Example: absorption law

Let  $E$  be the equation  $x \diamond y = x$ . Intuitively, we must have

$$\mathcal{V}(E) = \{w = w' \mid \text{the leftmost variable is the same for } w \text{ and } w'\}.$$

We will talk about proving statements like this one (say in Lean) later on; take it as given for now. The invariant is clear: we define  $I(w)$  to be the leftmost variable of  $w$ . Instantiating this invariant in the fundamental property of invariants, we get the following metatheorem.

**Metatheorem 10.2.** *Let  $w = w'$  be an equation such that the leftmost variable of  $w$  is the same as the leftmost variable of  $w'$ . Then  $w = w'$  cannot imply an equation that does not have the property from the last sentence.*

**Example: associativity**

For a more complicated example, let  $E$  be the associativity equation  $x \diamond (y \diamond z) = (x \diamond y) \diamond z$ . Intuitively, we must have

$$\mathcal{Y}(E) = \{\text{equations that, when we remove all parentheses, are of the form } w = w\}.$$

There is an invariant lurking behind: it is the (ordered) list of variables appearing in an expression, counting repetitions. More formally, we define  $I(w)$  to be the tuple of variables appearing in  $w$ , listed from left to right, say. Again, from the fundamental property of invariants we get the following.

**Metatheorem 10.3.** *Let  $w = w'$  be an equation such that the variables appearing in  $w$ , taking into account order and repetitions, are the same ones that appear in  $w'$ . Then  $w = w'$  cannot imply an equation that does not have the property from the last sentence.*

If we were coding a computer program that computes  $I(w)$  given  $w$ , one could take the string of symbols that is  $w$ , ignore all parentheses, replace all symbols  $\diamond$  by commas, and surround with an appropriate delimiter. (I imagine one could easily do this using regular expressions.)

We can compute other examples, but the invariant can get complicated even for simple equations. Exercise: what is the invariant for commutativity? Answer: To compute  $I(w)$  from  $w$  replace all parentheses with curly braces and all symbols  $\diamond$  with commas, and interpret the result as nested sets.

## 10.2 Expanding the language

The method of invariants really shines when we expand our formal language. Right now our language consists of variables, parentheses, the equal sign, and  $\diamond$  (there is also an implicit use of  $\forall$  but let's ignore that for now). Let  $\Pi$  denote the preorder of equations (built from the language described) ordered by implication.

We will add the symbol  $\wedge$  ('and') to our language. Then we consider a bigger preorder  $\Pi' \supseteq \Pi$  which includes equations and also conjunctions of equations. Even if we only care about  $\Pi$  it will be apparent that studying invariants in  $\Pi'$  gives us useful metatheorems about  $\Pi$ . Equations and conjunctions of equations are examples of *formulas* (or formulae, according to taste).

If  $\varphi$  is a formula, we can define  $\mathcal{Y}(\varphi)$  to be the set of all formulae implied by  $\varphi$ ; this agrees with our previous definition. Now define an invariant of  $\varphi$  to be a function  $I$  on terms such that

$$\mathcal{Y}(\varphi) \cap \Pi = \{w = w' \mid I(w) = I(w')\}.$$

Again, this clearly agrees with our previous definition. Although  $\mathcal{Y}(\varphi) \cap \Pi$  might not be upwards closed in  $\Pi'$ , it is upwards closed in  $\Pi$ , which is enough to get the fundamental property of invariants *verbatim*. This leads to more metatheorems we didn't have access to before.



### Example: associativity and idempotency

Let  $\varphi$  be the conjunction of the associative law and the idempotency law ( $x \diamond x = x$ ). Again, we will rely on our intuition, which says that an invariant  $I$  defined by taking  $I(w)$  to be the set of all variables appearing in  $w$ , works. The corresponding metatheorem is the following

**Metatheorem 10.4.** *Let  $w = w'$  be an equation such that the set of variables appearing in  $w$  is equal to the set of variables appearing on  $w'$ . Then  $w = w'$  cannot imply an equation that does not have the property from the last sentence.*

### Example: associativity and commutativity

For a similar example, we can let  $\varphi$  be the conjunction of the associative and the commutative laws. Here we can define  $I(w)$  to be the multiset of variables appearing in  $w$ . We obtain the following metatheorem.

**Metatheorem 10.5.** *Let  $w = w'$  be an equation such that the variables appearing in  $w$ , taking into account multiplicity, are the same ones that appear in  $w'$ . Then  $w = w'$  cannot imply an equation that does not have the property from the last sentence.*

Trivia: this was the first example of a metatheorem obtained by use of an invariant.

### Example: associativity and commutativity with a twist

We can keep expanding our language if it helps us express more intricate invariants. For instance, we can add the symbol ‘1’ to our language. Let  $\varphi$  be the conjunction of associativity, commutativity, the equations  $1 \diamond x = x$ , and

$$\underbrace{x \diamond x \diamond \cdots \diamond x}_{m \text{ times}} = 1,$$

for some fixed positive integer  $m$ . Pause to guess the invariant before we move on.

The invariant  $I(w)$  is the multiset of variables appearing in  $w$  but multiplicities are computed modulo  $m$ . Thus we have the pretty metatheorem:

**Metatheorem 10.6.** *Fix some positive integer  $m$ . Let  $w = w'$  be an equation such that every variable appearing in  $w$  appears the same number of times in  $w'$  modulo  $m$ . Then  $w = w'$  cannot imply an equation that does not have the property from the last sentence.*

## 10.3 Proving metatheorems from invariants in Lean

For the rest of this chapter we readopt the convention of calling ‘theorem’ an important result, not necessarily pertaining to specific equations.

An invariant is generally a *syntactic* property of an expression. However, invariants can also be described and calculated *semantically* through the notion of a *lifting magma family*, described below. The general idea is that the value of an invariant for an expression can be computed by substituting specific values for the variables in the expression and evaluating the result in a certain magma in the lifting magma family; additional requirements ensure that the fundamental property of invariants is satisfied.

**Definition 10.7** (Lifting Magma Family). *LiftingMagmaFamily* A *lifting magma family* is a family of magmas  $\{G_\alpha\}$ , one for each type  $\alpha$ , satisfying the following properties:

- For each type  $\alpha$ , there is a function  $\iota_\alpha : \alpha \rightarrow G_\alpha$ .
- Given a function  $f : \alpha \rightarrow G_\alpha$ , there is a magma homomorphism  $\text{lift } f : G_\alpha \rightarrow G_\alpha$  such that  $\text{lift } f(\iota_\alpha(x)) = f(x)$  for all  $x$  in  $\alpha$ .

**Example 1.** *The free Abelian groups form a lifting magma family. When the underlying set is finite, the groups are isomorphic to  $\mathbb{Z}^n$ .*

**Example 2.** *Lists form a lifting magma family.*

The key consequence of the ?? is that it is significantly easier to check whether an equation is satisfied in a lifting magma family.

**Theorem 10.8** (Evaluation theorem for lifting magma families). *MagmaLaw.models<sub>i</sub> f f<sub>s</sub> satisfies Suppose Eisan*  
*Determining whether  $E$  is satisfied by  $G_X$  is equivalent to checking that  $E$  is true with the specific substitution  $\iota_X$ .*

*Proof.* For the forward direction, suppose  $E$  is satisfied by  $G_X$ . Then, by definition, any substitution of the variables in  $E$  with elements of  $G_X$  will yield a true equation. In particular, substituting according to  $\iota_X$  will yield a true equation.

For the reverse direction, suppose that  $E$  is true when evaluated with the substitution  $\iota_X$ . Now, consider an arbitrary substitution of variables  $f : X \rightarrow G_X$ . By the lifting magma family property, there is a magma homomorphism  $\text{lift } f : G_X \rightarrow G_X$  such that  $\text{lift } f(\iota_X(x)) = f(x)$  for all  $x$  in  $X$ . In other words, applying the substitution  $f$  is equivalent to first applying to substitution  $\iota_X$  and then applying the homomorphism  $\text{lift } f$ . Since  $E$  is true when evaluated with the substitution  $\iota_X$ , it is also true after applying the homomorphism  $\text{lift } f$ . Thus,  $E$  is satisfied by  $G_X$ .  $\square$

**Theorem 10.9** (The fundamental property of invariants). *Let  $E$  and  $E'$  be equations involving a set of variables  $X$ , and let  $G$  be a lifting magma family.*

*If  $E$  is true with the substitution  $\iota_X$ , and  $E$  implies  $E'$ , then so is  $E'$ .*

*Proof.* Applying the evaluation ??, we see that  $E$  is satisfied by  $G_X$ . Since  $E$  implies  $E'$ ,  $E'$  is also satisfied by  $G_X$ , and in particular,  $E'$  is true with the substitution  $\iota_X$ .  $\square$

**Remark 1.** *The result of evaluating an expression along the function  $\iota_X : X \rightarrow G_X$  is the invariant.*

*In the case of Abelian groups, the result of evaluation is the variables in the expression with multiplicity. In the case of lists, the result of evaluation is the variables in the expression in the order they appear.*

*When the lifting magma family has good computational properties, calculating the invariant becomes easy.*

**Remark 2.** *Given an equation  $\phi$  in the language of magmas (possibly involving logical operations other than equality and universal quantification), the initial (i.e., most general) magmas satisfying  $\phi$  (provided they exist) form a lifting magma family.*

*However, for the purpose of generating invariants, we are interested in lifting magma families with convenient descriptions that are computationally tractable.*

**Remark 3.** *Suppose  $S$  is a finite set of equations in the language of magmas that is a confluent term rewriting system under a certain ordering of the terms (in the sense of the Knuth-Bendix algorithm). Then the initial magmas satisfying  $S$  form a lifting magma family where equality of elements in the magma is decidable.*

*This offers a way of generating examples of lifting magma families with good computational properties for computing invariants of expressions.*

## 10.4 Generating laws from equations

The invariants defined in this chapter are properties of the *syntax* of the equations being considered. In other words, they are properties of the laws associated with the equations, rather than of the equations themselves. Proving non-implications using invariants requires a way to operate on the syntax of the equations and then translate the reasoning back to results about the original equations.

A magma law can be generated from an equation by accessing the syntax used in its definition and converting it to a declaration representing a magma law through metaprogramming. There is a choice in the variable set of the magma law – one one hand, it can be a finite set whose size matches the number of variables, and on the other hand, it can be the set of natural numbers. The advantage of the former is that one can generate proofs that the satisfiability of the magma law is equivalent to the satisfiability of the original equation (this only needs to be done for variable sets of size up to six, since that is the maximum size currently being considered in the project; it’s convenient to prove individual lemmata for each variable set size establishing this equivalence). The advantage of the latter is that it bypasses the need to cast between various finite sets while constructing a model as a counter-example.

One approach is to generate both forms of the law, using the first to establish the equisatisfiability of the law and the equation and then transporting this

result to the second form of the law. The conversion from the first form to the second is summarised in the lemma below.

**Lemma 10.10.** *Law.satisfies<sub>f</sub>in<sub>s</sub>atisfies<sub>n</sub>at[Compatibilitybetweenmagmalawsoverfinitesetsandthenatural*

*Proof.* In the forward direction, suppose  $\phi : \mathbb{N} \rightarrow M$  is a substitution. Then the restriction of  $\phi$  to the first  $n$  natural numbers is a substitution for the variables of  $E$ , and since  $M$  satisfies  $E$ , the law  $E$  is true in  $M$  under this substitution. Since  $\tilde{E}$  is the same as  $E$  under the substitution  $\phi$ ,  $M$  satisfies  $\tilde{E}$ .

In the reverse direction, suppose  $\phi : \{0, 2, \dots, n-1\} \rightarrow M$  is a substitution. Then  $\phi$  can be extended to a substitution  $\tilde{\phi} : \mathbb{N} \rightarrow M$  by setting  $\tilde{\phi}(i) = \phi(i)$  for  $i \leq n$  and  $\tilde{\phi}(i) = 0$  for  $i \geq n$ . Since  $M$  satisfies  $\tilde{E}$  under the substitution  $\tilde{\phi}$ , it satisfies  $E$  under the restriction of  $\tilde{\phi}$  to the first  $n$  natural numbers, which is precisely  $\phi$ . The special case where  $n = 0$  is in fact impossible, since there cannot be an expression with no variables.  $\square$

## 10.5 Conclusion: Beyond Invariants

We are still lacking:

- A large collection of invariants.
- An estimate for how many implications the resulting metatheorems will settle.
- Algorithms (in Lean, Python, or otherwise) to compute known invariants.
- General results about lifting magmas.
- Formalization of the method of invariants and resulting metatheorems.
- Knowledge about the category-theoretic interpretation of invariants (see the questions in the note for category theorists).

Related to the last bullet point, we note the following. If all that matters about invariants is the fundamental property, we can apply the old French trick of turning a (meta-meta)theorem into a definition.

Q: If we were to define invariants as any functions satisfying the fundamental property, would anything change? (For those who read the note for category theorists: an equivalent redefinition is to consider invariants as functors  $\Pi \rightarrow \mathbf{Bool}$ ).

# Chapter 11

## Some abstract nonsense

This is an alternate presentation of the material of the previous section, where we use the “abstract nonsense” of free magmas in the presence of a theory as the conceptual foundation.

**Definition 11.1** (Free magma relative to a theory). *models-def* Let  $\Gamma$  be a theory with an alphabet  $X$ . A *free magma* with alphabet  $X$  subject to the theory  $\Gamma$  is a magma  $M_{X,\Gamma}$  together with a function  $\iota_{X,\Gamma} : X \rightarrow M_{X,\Gamma}$ , with the following properties:

- (i)  $M_{X,\Gamma}$  obeys the theory  $\Gamma$ :  $M_{X,\Gamma} \models \Gamma$ .
- (ii) For any magma  $M$  obeying the theory  $\Gamma$  and any function  $f : X \rightarrow M$ , there exists a unique magma homomorphism  $\tilde{f} : M_{X,\Gamma} \rightarrow M$  such that  $\tilde{f} \circ \iota_{X,\Gamma} = f$ .

Such magmas exist and are unique up to a suitable isomorphism:

**Theorem 11.2** (Existence and uniqueness of free magmas). *free-theory* Let  $\Gamma$  be a theory with alphabet  $X$ .

- (i) *There exists a free magma  $M_{X,\Gamma}$  with alphabet  $X$  subject to the theory  $\Gamma$ .*
- (ii) *If  $M_{X,\Gamma}$  and  $M'_{X,\Gamma}$  are two free magmas with alphabet  $X$  subject to the theory  $\Gamma$ , then there exists a unique magma isomorphism  $\phi : M_{X,\Gamma} \rightarrow M'_{X,\Gamma}$  such that  $\phi \circ \iota_{X,\Gamma} = \iota'_{X,\Gamma}$ .*

We remark that the ordinary free magma  $M_X$  corresponds to the case when  $\Gamma$  is the empty theory.

*Proof.* sound-complete, equiv For (i), we define  $M_{X,\Gamma} = M_X / \sim$ , where the equivalence relation  $\sim$  is defined by requiring  $w \sim w'$  if and only if  $\Gamma \models w \simeq w'$ ; this is an equivalence relation thanks to ??, and from ?? we see that this relation respects the magma operations, so that  $M_{X,\Gamma}$  is a magma. The map  $\iota_{X,\Gamma} : X \rightarrow M_{X,\Gamma}$  is defined by setting  $\iota_{X,\Gamma}(x)$  to be the equivalence class of  $x$  in  $M_{X,\Gamma}$  for each  $x \in X$ .

We first check that  $M_{X,\Gamma}$  obeys  $\Gamma$ . Let  $w \simeq w'$  be a law in  $\Gamma$ , and let  $f : X \rightarrow M_{X,\Gamma}$  be a function. We may lift this function to a function  $\tilde{f} : X \rightarrow M_X$ . From ??, we have  $\Gamma \vdash w \simeq w'$  and hence  $\Gamma \vdash \varphi_{\tilde{f}}(w) \simeq \varphi_{\tilde{f}}(w')$ . By ??, we conclude  $\Gamma \models \varphi_{\tilde{f}}(w) \simeq \varphi_{\tilde{f}}(w')$ . Quotienting by  $\sim$ , we conclude that  $\varphi_f(w) = \varphi_f(w')$ , giving the claim by ??.

Now we check the universal property (ii). Let  $M$  be a magma obeying the theory  $\Gamma$ , and let  $f : X \rightarrow M$  be a function, then we have a magma homomorphism  $\varphi_f : M_X \rightarrow M$ . If  $w, w' \in M_X$  are such that  $w \sim w'$ , then  $\Gamma \models w \simeq w'$  and hence  $\varphi_f(w) = \varphi_f(w')$ . Hence  $\varphi_f$  descends to a map  $\tilde{f} : M_{X,\Gamma} \rightarrow M$ , which one can check to be a magma homomorphism with  $\tilde{f} \circ \iota_{X,\Gamma} = f$ . By construction,  $M_{X,\Gamma}$  is generated by  $\iota_{X,\Gamma}(X)$ , and so this homomorphism is unique.  $\square$

**Example 3** (Free associative magma). *Let  $\Gamma$  consist solely of the associative law, ?? (so  $X$  contains 0,1,2). Then one can take  $M_{X,\Gamma}$  to be the set of nonempty strings with alphabet  $X$ , with magma operation given by concatenation, and  $\iota_{X,\Gamma}(x)$  being the length one string  $x$ . It is a routine matter to verify that this obeys the axioms of a free magma subject to  $\Gamma$ .*

**Example 4** (Free associative commutative magma). *Let  $\Gamma$  consist of the associative law (??) and the commutative law (??). Then one can take  $M_{X,\Gamma}$  to be the free abelian monoid  $\mathbb{N}_0^X \setminus 0$  of tuples  $(n_x)_{x \in X}$  with the  $n_x$  natural numbers, not all zero, with all but finitely many of the  $n_x$  vanishing, with the magma operation given by vector addition, and with  $\iota_{X,\Gamma}(x)$  being the standard generator of  $\mathbb{N}^X$  associated to  $x \in X$ ; one can think of this space as the space of formal non-empty commuting associating sums of  $X$ . It is a routine matter to verify that this obeys the axioms of a free magma subject to  $\Gamma$ .*

**Example 5** (Free left absorptive magma). *Let  $\Gamma$  consist of the left absorptive law (??). Then one can take  $M_{X,\Gamma}$  to be  $X$  with the law  $x \diamond y = x$ , and  $\iota_{X,\Gamma}$  to be the identity map. It is easy to see that this is indeed a free magma subject to  $\Gamma$ .*

**Example 6** (Free constant magma). *Let  $\Gamma$  consist of the constant law (??). Then one can take  $M_{X,\Gamma}$  to be the disjoint union  $X \uplus \{0\}$  of  $X$  and another object 0, with  $\iota_{X,\Gamma}$  being the identity embedding, and with the zero magma law  $x \diamond y = 0$  for all  $x, y \in X \uplus \{0\}$ .*

Free magmas can be used to characterize entailment by  $\Gamma$  in terms of a canonical invariant.

**Theorem 11.3** (Canonical invariant). *free-theory Let  $\Gamma$  be a theory with some alphabet  $X$ , and let  $M_{X,\Gamma}$  be a free magma with alphabet  $X$  subject to  $\Gamma$ , with associated map  $\iota_{X,\Gamma} : X \rightarrow M_{X,\Gamma}$ . Then for any  $w, w' \in M_X$ , we have*

$$\Gamma \models w \simeq w' \text{ if and only if } \varphi_{\iota_{X,\Gamma}}(w) = \varphi_{\iota_{X,\Gamma}}(w').$$

*Proof.* freemag-exist By ?? we may take  $M_{X,\Gamma}$  to be the canonical free magma constructed in the proof of that theorem. The claim is then clear from expanding out definitions.  $\square$

Every theory  $\Gamma$  then gives a metatheorem about anti-implication:

**Corollary 11.4** (Criterion for anti-implication). *Let  $\Gamma$  be a theory with some alphabet  $X$ , and let  $M_{X,\Gamma}$  be a free magma with alphabet  $X$  subject to  $\Gamma$ , with associated map  $\iota_{X,\Gamma} : X \rightarrow M_{X,\Gamma}$ . Let  $w \simeq w'$  and  $w'' \simeq w'''$  be laws with alphabet  $X$ . If one has*

$$\varphi_{\iota_{X,\Gamma}}(w) = \varphi_{\iota_{X,\Gamma}}(w')$$

but

$$\varphi_{\iota_{X,\Gamma}}(w'') \neq \varphi_{\iota_{X,\Gamma}}(w'''),$$

then the law  $w \simeq w'$  cannot imply the law  $w'' \simeq w'''$ .

*Proof.* canonical-invariant By ??, the hypothesis  $\iota_{X,\Gamma}(w) = \iota_{X,\Gamma}(w')$  is equivalent to  $\Gamma \models w \simeq w'$ , and the hypothesis  $\iota_{X,\Gamma}(w'') \neq \iota_{X,\Gamma}(w''')$  is equivalent to  $\Gamma \not\models w'' \simeq w'''$ . The claim follows.  $\square$

**Example 7.** Let  $\Gamma$  be the associative and commutative law, so that we can take  $M_{X,\Gamma} = \mathbb{N}_0^X \setminus 0$  as in ??. One can then check that for any word  $w \in M_X$ , that  $\varphi_{\iota_{X,\Gamma}}(w)$  is the tuple that assigns to each letter  $x$  of the alphabet, the number of times  $x$  appears in  $w$ . We conclude that if  $w, w'$  have the same number of occurrences of each letter of the alphabet, but  $w'', w'''$  do not, then  $w \simeq w'$  cannot imply  $w'' \simeq w'''$ . This recovers ??.

**Example 8.** Let  $\Gamma$  consist of the left absorption law, so we can take  $M_{X,\Gamma} = X$  as in ??. Then  $\varphi_{\iota_{X,\Gamma}}(w)$  is the first letter of  $w$ . We conclude that if  $w, w'$  have the same first letter, but  $w'', w'''$  do not, then  $w \simeq w'$  cannot imply  $w'' \simeq w'''$ .

**Example 9.** Let  $\Gamma$  consist of the constant law, so we can take  $M_{X,\Gamma} = X \uplus \{0\}$  as in ??. Then  $\varphi_{\iota_{X,\Gamma}}(w)$  is  $x$  if  $w$  is just a letter  $x$  of the alphabet, and 0 otherwise. We conclude that if  $w, w', w'''$  have order at least one, but  $w''$  has order zero, then  $w \simeq w'$  cannot imply  $w'' \simeq w'''$ ; this is basically ??.

**Example 10.** Let  $\Gamma$  be the theory consisting of the commutative and associative laws, and an additional law  $x^n \simeq y^n$  for a fixed  $n$ , where  $x^n$  denotes the magma operation applied to  $n$  copies of  $x$  (the order is irrelevant thanks to associativity), then one can check (for finite  $X$ ) that the free magma  $M_{X,\Gamma}$  can be taken to be  $(\mathbb{Z}/n\mathbb{Z})^X$  with the addition operation, and  $\iota_{X,\Gamma}(x)$  being the standard generator associated to  $x$ . Then for any word  $w$ ,  $\varphi_{\iota_{X,\Gamma}}(w)$  corresponds to a tuple that assigns to each letter  $x$  of the alphabet, the number of times  $x$  occurs in  $w$  modulo  $n$ . We conclude that if  $w, w'$  have the same number of occurrences modulo  $n$  of each letter of the alphabet, but  $w'', w'''$  do not, then  $w \simeq w'$  cannot imply  $w'' \simeq w'''$ . This is a stronger version of ??.

## 11.1 Confluent theories

One promising source of theories  $\Gamma$  for which the free magma  $M_{X,\Gamma}$  can be understood are the *confluent theories*.

**Definition 11.5** (Confluent theory). *free-magma-def* Let  $\Gamma$  be a theory. A word  $w$  can be *reduced* to another  $w'$  if one can get from  $w$  to  $w'$  by a series of substitutions of laws in  $\Gamma$ , where no substitution increases the length of the word **this is a working definition, might not be the best one to keep..** A theory  $\Gamma$  is *confluent* if whenever a word  $w$  can be reduced to both  $w'$  and  $w''$ , then both  $w'$  and  $w''$  can be reduced further to a common reduction  $\tilde{w}$ . As such, each word  $w \in M_X$  should have a *unique simplification* to a reduced word  $w_\Gamma$  in some normal form, for instance the shortest reduction that is minimal with respect to some suitable ordering such as lexicographical ordering.

**Example 11.** *The associative law, ??, appears to be confluent check this.*

**Example 12.** *The theory consisting of both the associative and commutative laws, ??, ??, appears to be confluent check this.*

**Example 13.** *The idempotent law, ??, appears to be confluent check this.*

The significance of confluent theories lies in

**Theorem 11.6** (Free magma of a confluent theory). *confluent-theory* Let  $\Gamma$  be a confluent theory. Then the free magma  $M_{X,\Gamma}$  subject to this theory can be described as the space of reduced words in  $M_X$  in normal form, where the operation  $w \diamond_\Gamma w'$  on this magma is defined as the normal form reduction of  $w \diamond w'$ , and  $\iota_{X,\Gamma}$  is the identity embedding (note that every single-letter word is already in normal form).

*Proof.* Should just be a matter of expanding definitions properly.  $\square$

**Corollary 11.7** (Criterion for anti-implication). *models-def* Let  $\Gamma$  be a confluent theory. Then a law  $w \simeq w'$  is a consequence of  $\Gamma$  if and only if  $w, w'$  have the same normal form reduction. In particular, a law with this property cannot imply a law without this property.

*Proof.* free-confluent,anti-impl Follows from ??.  $\square$

It is thus of interest to locate some confluent laws. Here is a non-trivial example:

**Theorem 11.8** (477 confluent). *confluent-theory, eq477 ?? is confluent.*

*Proof.* 477-lemma See the notes here. A sketch of proof is as follows. We induct on the length of the term. As before we consider terms of the form  $XY$ . Also, in both sequences if a simplification is applied to the whole term, then we can assume the sequence is simply final.

By ??, if any of the two sequences is final, then right before the last step, the two factors of the outermost product are both simple. This is also true for the result of the non-final sequence. By the induction hypothesis, they can be identified correspondingly, so the two sequences are either both final or both non-final, and in the first case, the same simplification is applied to give the same result.  $\square$



**Lemma 11.9** (477 lemma). *If  $Z$  and  $W$  are simple, then  $Z(W \cdots (WW))$  is simple.*

*Proof.* Assume the contrary. Then we have 2 cases.

**Case 1:**  $W \cdots (WW)$  matches the pattern  $y(x(y \cdots (yy)))$ , with  $k$  occurrences of  $W$  ( $k \leq n$ ). Since  $|x(y \cdots (yy))| > n|y|$ , but  $|\cdots(WW)| \leq (n-1)|W|$ , this is impossible.

**Case 2:**  $Z(W \cdots (WW))$  matches the pattern  $y(x(y \cdots (yy)))$ . Since  $n \geq 3$ , we have  $Z = y = W$ , so  $|W \cdots (WW)| = n|W| = n|Z|$ , contradicting  $|x(y \cdots (yy))| > n|y|$ .  $\square$

## Chapter 12

# Simple rewrites

53,905 implications were automatically generated by simple rewrites.

**describe the process of automatically generating these implications here.**

## Chapter 13

# Trivial auto-generated theorems

Approximately 4.5m transitive implications were proven by a transitive reduction of about 15k theorems. Most of these implications were derived from being the first automated run to connect the largest equivalence classes, hence creating a large set of transitively closed implications.

Scripts generated theorems to try simple combinations of equation rewrites to reach the desired goal for every unknown implication. The generated proof scripts were run with lean and the successful theorems were extracted. An example of the types of generated rewrites that were tested:

```
repeat intro
  apply

repeat intro
  try { rw [<-h] }
  try { rw [<-h, <-h] }
  try { rw [<-h, <-h, <-h] }
  try { rw [<-h, <-h, <-h, <-h] }
  try { rw [<-h, <-h, <-h, <-h, <-h] }
  repeat rw [h]

repeat intro
  try {
    nth_rewrite 1 [h]
    try { rw [h] }
    try { rw [<-h] }
  }
  try {
    nth_rewrite 2 [h]
    try { rw [h] }
```

```

    try { rw [<-h] }
  }
  try {
    nth_rewrite 3 [h]
    try { rw [h] }
    try { rw [<-h] }
  }
  try {
    nth_rewrite 4 [h]
    try { rw [h] }
    try { rw [<-h] }
  }
  try {
    nth_rewrite 1 [h]
    nth_rewrite 1 [h]
    try { rw [h] }
    try { rw [<-h] }
  }
  ...

```

## Chapter 14

# Enumerating Small Finite Magmas

describe the process of automatically generating these implications  
here.

## Chapter 15

# Equation Search

Approximately 650k transitive implications were proven by a custom tool leveraging the implication graph. After previous brute force had derived many implications expressible as a small number of rewrites, this search tool uses substitutions implied by the implication graph to search further.

An example proof illustrates the logic it uses:

```
have eq3315 (x y : G) : x * y = x * (y * (x * x)) := by
  apply Apply.Equation12_implies_Equation11 at h
  apply RewriteHypothesis.Equation11_implies_Equation3323 at h
  apply Apply.Equation3323_implies_Equation3315 at h
  apply h
have eq52 (x y : G) : x = x * (y * (x * x)) := by
  apply Apply.Equation12_implies_Equation61 at h
  apply Apply.Equation61_implies_Equation54 at h
  apply Apply.Equation54_implies_Equation52 at h
  apply h
repeat intro
nth_rewrite 1 [eq3315]
nth_rewrite 1 [← eq52]
apply h
repeat assumption
```

Using the graph of implications and refutations, it identifies equivalence classes/strongly-connected components in the implication graph and possible goals by subtracting out the refutation graph. Iterating through all equivalence classes, it can perform a meet-in-the-middle graph search where it searches outwards from both hypotheses and goals by performing equation substitutions. Depending on the number of hypotheses versus goals, it dynamically adjusts the search depth on both sides based on a configured branching factor.

Due to its naive implementation, it may only be able to perform certain substitutions in a round-about way and the graph size explodes faster than it must, so it's limited to fairly shallow search depths. Also, the tool may emit

proofs without some information Lean may require, so some generated proofs have to be fixed-up afterwards.

## Chapter 16

# E-Graphs

For proving implications, we used another technique called equality saturation [?] with the lean-egg tactic, to automatically construct proofs.

A similar approach is being pursued in the MagmaEgg tool as well, which is a standalone program that only supports magma equalities, while the lean-egg tactic supports any Lean expression.

### 16.1 lean-egg

#### 16.1.1 Methodology

The basic methodology of equality saturation is based on E-Graphs, a data structure that can store equivalence classes of terms efficiently. We used the lean-egg tactic (<https://github.com/marcusrosset/lean-egg>), based on equality saturation as a tactic, which (re)constructs a proof from the E-graph [?] in Lean. This means that we do not have to trust either the egg tool nor the tactic: if something goes wrong, Lean will not accept the constructed proof. In fact, we found issues with the proof reconstruction from the examples in this project.

The lean-egg tactic works for equational reasoning, i.e. proving equalities as consequences of other equalities (potentially universally quantified), which is exactly what we need to prove implications of laws in Magmas. In many cases, we have laws of the form  $x = y$ , where neither set of variables in the left- and right-hand-side of the law is a subset of each other. In this case the laws cannot be used as rewrite rules: it's not clear what it would be rewritten to, since there are unknowns on both sides of the equation. For these cases we used a simple heuristic, where we instantiate the variables with terms found in the (proof) context, as those are likely to be important for proving the equality.

#### 16.1.2 Results

Out of the possible implications between the 34 equations considered in ??, this method found an additional 86 implications that were not found before.



Some of these seem to be missing in the computation of the transitive closure of implications of the equalities (an investigation is in progress), but some of these are genuinely new theorems, and the `lean-egg` tactic finds good proofs of these (these can be rewritten using `calc` style with a different tactic, `calcify`: <https://github.com/nomeata/lean-calcify>). An example of this is the following proof, found by lean-egg:

**Theorem 16.1** (14 implies 23). *eq23, eq14 Subgraph. Equation 14 implies Equation 23 is equivalent to ??*.

*Proof.*

$$x = (x \diamond x) \diamond (x \diamond (x \diamond x)) = (x \diamond x) \diamond x$$

□

It was also able to (re)prove ??, albeit with a manually-provided hint (guide, in the sense of [?]).

## 16.2 MagmaEgg

This is a simple but apparently at least somewhat effective Rust theorem prover based on egg e-graph library written for this project.

It proved 5574 of the 24283 implications in the “only\_strongest.txt” file at the time.

The code was originally based on the `magma_search` pull request, but has been pretty much completely rewritten.

Currently `search` just uses the egg library in a basic fashion, except that in case there are extra variables not present in the LHS, it has code to instantiate them with all subexpressions of the original goal.

Exporting the proofs to Lean has turned out to be harder than finding the proofs, but a good solution has been implemented (modulo some issues in egg that require to sometimes turn off explanation optimization since it sometimes triggers stack overflows and assert failures) that directly produces proof terms using `let/have` and `Eq.refl`, `Eq.symm`, `Eq.trans`, `Magma.op`, a congruence lemma for `Magma.op` and variables and the hypothesis. I define one letter aliases for them to reduce verbosity.

Possible future work:

- Figure out which implications are important to prove and try it on them
- Replace the fork-based code with self-execution so that it works on Windows and is less of a hack
- Fork egg and fix the buggy and slow length optimization of explanations
- Maybe write Lean code directly instead of writing explanation sexps and converting to Lean code in a second run
- Fix the generation of extra variable values so it doesn’t take too much time in pathological cases (i.e. goals with 4-6 variables)

- Determine whether it actually has some advantages compared to Vampire and lean-egg
- Support searching for multiple goal equations at once
- Write a custom elaborator for Lean to speed up elaboration
- If the Lean kernel turns out to be too slow for some large necessary proofs and thus the custom elaborator is not enough, write a custom verified typechecker
- Support having extra rewrite rules, such as other implications that have been found implied by the hypothesis, or simple equalities found by the egraph search itself
- Run it with massive computing resources if deemed useful and someone offers those, once it's a bit more mature

## Chapter 17

# Using the Vampire theorem prover

1,775 implications were proven using the Vampire theorem prover.

The Vampire proofs were found by iteratively trying to prove some of the remaining unknown implications, then taking the transitive closure including the newly proven theorems. At the end only the transitive reduction of the implications was kept.

The Vampire proofs were converted to Lean proofs using a term elaborator implementing the deduction step of superposition calculus.