FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

MASTER THESIS

Maryia Kapytka

Minion Cores of Clones

Department of Algebra

Supervisor of the master thesis: doc. Mgr. Libor Barto, Ph.D.
Study programme: Mathematical Structures
Study branch: Mathematical Structures

Prague 2023

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

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

In .... date ....

Author's signature

I would like to express my gratitude to my advisor, Libor Barto, for his unwavering support, guidance, and belief in me. I would also like to thank my family and friends for their constant encouragement and love.

Title: Minion Cores of Clones

Author: Maryia Kapytka

Department: Department of Algebra

Supervisor: doc. Mgr. Libor Barto, Ph.D., Department of Algebra

Abstract: This thesis provides a classification of the minion homomorphism preordering and minion cores within a class of multi-sorted Boolean clones. These clones can be described as those clones defined on the set $\{0,1\}^k = \{0,1\} \times \{0,1\} \times \cdots \times \{0,1\}$, where the clone operations act component-wise on the k-tuples, which are determined by multi-sorted unary or binary relations.

The second chapter of this thesis focuses on presenting the key findings. We introduce specific minion cores and establish the preordering among them. Furthermore, we prove that each clone falling under the aforementioned type is equivalent to one of these minion cores.

Keywords: universal algebra, minion, multi-sorted Boolean clone, minion homomorphism

Contents

Introduction

In the field of universal algebra, a *clone* $\mathcal{C}$ on a finite set A is a collection of finitary operations that satisfy two conditions: $\mathcal{C}$ contains all the projections and $\mathcal{C}$ is closed under finite compositions. Clones are central objects in Universal Algebra, where they enable us to study algebras up to term equivalence, and also in Computational Complexity, where they provide complexity invariants, e.g., for the fixed-template finite-domain Constraint Satisfaction Problems (CSPs): to every such a CSP one associates a certain clone and the complexity of the CSP is fully captured by the associated clone.

Clones on a fixed domain A are naturally ordered by inclusion. Larger clones in this ordering correspond to more structured algebras and to easier Constraint Satisfaction Problems. However, full classification of this ordering of clones seems currently out of reach already for three-element domains.

The focus of this thesis is on studying a preordering of clones that is coarser than inclusion. This preordering is defined by the existence of a minion homomorphism, which is a mapping from clone $\mathcal{A}$ to clone $\mathcal{B}$ that preserves arities and compositions with projections.

The motivation to study clones up to minion homomorphisms is that the position of a clone in this order still fully captures the complexity of CSPs [1]. Moreover, describing clones up to this preordering seems to be a much easier (albeit still challenging) task than up to inclusion. The description can be also much nicer, as illustrated by comparing Post's ordering of Boolean clones with respect to inclusion [2] and Bodirky's and Vucaj's [3] ordering of the same class of clones up to minion homomorphisms.

In accordance with standard practice for preorders, we consider two clones to be equivalent if there exists a minion homomorphism from one clone to the other, and vice versa. Describing the original preordering then amounts to describing the equivalence and the ordering of clones factorized by the equivalence. It turns out that two clones are equivalent if they have the same minion core. Here, a minion is a weaker "version" of a clone: it contains all the projections and is closed under finite compositions with projections. A minion is called a *minion core* if every minion homomorphism to itself is an automorphism.

The thesis classifies the minion homomorphism preordering and minion cores in a class of clones that we now introduce.

Clones can be in general described using relations, which in our case are binary. An operation f is said to preserve a binary relation if the relation holds among the results of applying f to the arguments if it holds among the arguments, i.e. the following implication holds:

$$x_1Ry_1 \wedge x_2Ry_2 \wedge \cdots \wedge x_nRy_n \Rightarrow f(x_1,\ldots,x_n)Rf(y_1,\ldots,y_n)$$

We say that a clone $\mathcal{C}$ is *determined* by a set of relations if it contains exactly those operations that preserve all the relations in the set.

The class of clones I study in the thesis is the class of clones on finite sets that are determined by unary or binary relations whose both projections are at most two-element. This class comes from an ongoing project of classifying all clones on a three-element domain determined by binary arbitrary relations.

An equivalent viewpoint on this class of clones is via multi-sorted Boolean clones[1]: clones on $\{0,1\}^k = \{0,1\} \times \{0,1\} \times \cdots \times \{0,1\}$ whose operations act component-wise on the k-tuples, i.e., an n-ary operation on the clone is determined by a k-tuple of operations on $\{0,1\}$. The class of clones above correspond to the class of multi-sorted Boolean clones determined by multi-sorted unary or binary relations. Note that the description of all multi-sorted Boolean clones ordered by inclusion is widely open; the thesis' result can be regarded as a step toward describing multi-sorted Boolean clones up to minion equivalence.

The diagram below represents the poset of equivalence classes of clones. In the following chapters and sections of the thesis, we provide a proof of the diagram's validity.

The first chapter, titled "Clones and Minions," serves as an introductory chapter where we present the fundamental concepts utilized throughout the thesis.

Building upon these concepts, the second chapter delves into their application, demonstrating how they can be used to establish the validity of the diagram. The results in this chapter are original. Within the section titled "Collapse," we provide a proof that establishes the equivalence between each multi-sorted Boolean clone determined by multi-sorted binary relations, and one of specific minions we introduce. In the section labeled "Cores," we present a proof validating that the introduced minions indeed are minion cores. In the section titled "Ordering," we establish that the relations between the introduced minions align precisely with those depicted on the diagram. Lastly, in the section labeled "Summary" we summarize our results and present the final diagram.

Also we note that we used ChatGPT in order to improve sentences and formulations.

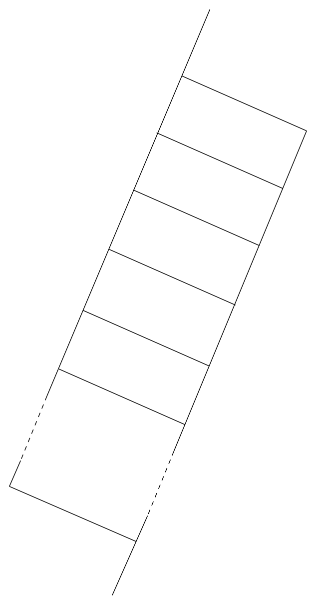


Figure 1: Preliminary diagram illustrating the preordering of multisorted Boolean clones determined by binary relations.

[1]The thesis does not contain a proof of this equivalence. It will appear in a forthcoming paper.

1. Clones and minions

1.1 Clones

For a nonnegative integer n we use the notation

$$[n] = \{1, 2, \dots, n\}.$$

Let A be a set and let n be a positive integer. We define A^n as the set of all n-tuples of elements of A. An n-ary *operation on* A is a function from A^n to A. The number n is referred to as the *arity*. Operations of arity 1 and 2 are also called *unary* and *binary* operations, respectively. We denote the set of n-ary operations on A as $\mathrm{Op}^{(n)}(A)$.

For any positive integer n, $i \in [n]$, and a set A, we define the *n-ary projection to the ith coordinate* as follows.

$$\pi_i^n(a_1, \dots, a_n) = a_i \text{ for every } a_1, \dots, a_n \in A$$

The set A should be clear from the context.

Now, suppose we have an n-ary operation f on A and m-ary operations $g_1, \dots, g_n$ on A. We define the *composition* of f with $g_1, \dots, g_n$, denoted as $f \circ (g_1, \dots, g_n)$, as the m-ary operation on A defined as follows.

$$(f \circ (g_1, \dots, g_n))(a_1, \dots, a_m) = f(g_1(a_1, \dots, a_m), \dots, g_n(a_1, \dots, a_m))$$

An *m-ary relation* on a set A is defined as a subset of A^m.

Let f be an n-ary operation and R be an m-ary relation on the same set A. We say that f *preserves* R if for all m-tuples $\boldsymbol{r}_1, \dots, \boldsymbol{r}_n \in R$, we have $f(\boldsymbol{r}_1, \dots, \boldsymbol{r}_n) \in R$. Here, $f(\boldsymbol{r}_1, \dots, \boldsymbol{r}_n)$ is computed component-wise:

$$\begin{pmatrix} r_{11} \\ r_{12} \\ \vdots \\ r_{1m} \end{pmatrix}, \begin{pmatrix} r_{21} \\ r_{22} \\ \vdots \\ r_{2m} \end{pmatrix}, \dots, \begin{pmatrix} r_{n1} \\ r_{n2} \\ \vdots \\ r_{nm} \end{pmatrix} \in R \implies \begin{pmatrix} f(r_{11}, r_{21}, \dots, r_{n1}) \\ f(r_{12}, r_{22}, \dots, r_{n2}) \\ \vdots \\ f(r_{1m}, r_{2m}, \dots, r_{nm}) \end{pmatrix} \in R$$

Now, we introduce the concept of a clone on a set A. Let $\mathrm{Op}(A) = \bigcup_{n=1}^{\infty} \mathrm{Op}^{(n)}(A)$ be the set of all operations on A. A *clone on* A is a subset $\mathcal{C}$ of $\mathrm{Op}(A)$ that contains all the projections and is closed under composition. We denote the set of n-ary operations in $\mathcal{C}$ as $\mathcal{C}^{(n)}$.

Given a set θ of relations on the same set A, the set $\mathrm{Pol}(\theta)$ is defined as the set of all operations on A that preserve all the relations in θ. It is not hard to show that this set is always a clone as stated in the following theorem. In fact, conversely, every clone on a finite set is of this form [4], but we will not need this fact in this thesis.

Theorem 1.1 (see [4]). *Let A be a set and θ be a set of relations on A. Then* $\mathrm{Pol}(\theta)$ *is a clone on A.*

We will work with a generalization of clones to multi-sorted sets. We define a *k-sorted set* as a k-tuple $\boldsymbol{A} = (A_1, A_2, \dots, A_k)$ comprising k sets. A *k-sorted n-ary operation* on $(A_1, A_2, \dots, A_k)$ is defined as a k-tuple $\boldsymbol{f} = (f_1, f_2, \dots, f_k)$, where f_i is an n-ary operation on A_i. The *n-ary k-sorted projection to the ith coordinate* (where $i \in [n]$), denoted by $\boldsymbol{\pi}_i^n$, is the k-sorted operation defined by

$$\boldsymbol{\pi}_i^n = (\underbrace{\pi_i^n, \dots, \pi_i^n}_{k \text{ times}}),$$

where the jth projection in the tuple is on the set A_j.

Let $\boldsymbol{f} = (f_1, f_2, \dots, f_k)$ be an n-ary k-sorted operation on a k-sorted set $\boldsymbol{A}$ and $\boldsymbol{g}^1 = (g_1^1, g_2^1, \dots, g_k^1)$, $\boldsymbol{g}^2 = (g_1^2, g_2^2, \dots, g_k^2)$, ..., $\boldsymbol{g}^n = (g_1^n, g_2^n, \dots, g_k^n)$ be m-ary k-sorted operations on the same k-sorted set $\boldsymbol{A}$. Then their composition, denoted as $\boldsymbol{f} \circ (\boldsymbol{g}^1, \boldsymbol{g}^2, \dots, \boldsymbol{g}^n)$, is the k-sorted m-ary operation defined as follows:

$$\begin{aligned}&\boldsymbol{f}\circ(\boldsymbol{g}^1, \boldsymbol{g}^2, \dots, \boldsymbol{g}^n)\\ &\quad = (f_1 \circ (g_1^1, g_1^2, \dots, g_1^n), f_2 \circ (g_2^1, g_2^2, \dots, g_2^n), \dots, f_k \circ (g_k^1, g_k^2, \dots, g_k^n))\end{aligned}$$

A *k-sorted clone* on k-sorted set $\boldsymbol{A}$ is a set of k-sorted operations on $\boldsymbol{A}$ that contains all the k-sorted projections on $\boldsymbol{A}$ and is closed under composition. A *multi-sorted clone* is a k-sorted clone for some k.

An m-ary *k-sorted relation* on a k-sorted set $\boldsymbol{A} = (A_1, A_2, \dots, A_k)$ of type $(i_1, \dots, i_m)$, where $i_1, \dots, i_m \in [k]$, is a subset $R \subseteq A_{i_1} \times A_{i_2} \times \dots \times A_{i_m}$. We say that an n-ary k-sorted operation $\boldsymbol{f} = (f_1, \dots, f_k)$ on $\boldsymbol{A}$ *preserves* such a k-sorted relation R if the following implication holds:

$$\begin{pmatrix} r_{11} \\ r_{12} \\ \vdots \\ r_{1m} \end{pmatrix}, \begin{pmatrix} r_{21} \\ r_{22} \\ \vdots \\ r_{2m} \end{pmatrix}, \dots, \begin{pmatrix} r_{n1} \\ r_{n2} \\ \vdots \\ r_{nm} \end{pmatrix} \in R \implies \begin{pmatrix} f_{i_1}(r_{11}, r_{21}, \dots, r_{n1}) \\ f_{i_2}(r_{12}, r_{22}, \dots, r_{n2}) \\ \vdots \\ f_{i_m}(r_{1m}, r_{2m}, \dots, r_{nm}) \end{pmatrix} \in R$$

We remark that the type of R is regarded as a part of the definition of R, so, formally, a k-sorted relation is a tuple $(R, i_1, \dots, i_m)$.

For a set θ of k-sorted relation on a k-sorted set $\boldsymbol{A}$, we define $\mathrm{Pol}(\theta)$ in a completely analogous way as before, i.e., as the set of all k-sorted operations on $\boldsymbol{A}$ that preserve all the k-sorted relations in θ. A multi-sorted analogue of Theorem 1.1 is also easily seen.

Theorem 1.2. *Let $\boldsymbol{A}$ be a k-sorted set and θ a set of k-sorted relations on $\boldsymbol{A}$. Then* $\mathrm{Pol}(\theta)$ *is a k-sorted clone on $\boldsymbol{A}$.*

1.2 Minions

In this section, we will define the concepts of minions and multi-sorted minions, which generalize clones and multi-sorted clones. The concept of a minion comes from [5].

Let A and B be any sets and let n be a positive integer. An n-ary operation from A to B is a function from A^n to B. We denote the set of all such n-ary operations as $\mathrm{Op}^{(n)}(A,B)$. Let $\mathrm{Op}(A,B) = \bigcup_{n=1}^{\infty} \mathrm{Op}^{(n)}(A,B)$ be the set of all operations from A to B.

Note that, in general, composition as defined above does not make sense for operations from A to B. However, it makes sense to compose such operations with projections on A. Explicitly, for an n-ary operation f from A to B, a positive integer m, and $i_1, \dots, i_n \in [m]$, the composition $f \circ (\pi^m_{i_1}, \pi^m_{i_2}, \dots, \pi^m_{i_n})$ is the following m-ary operation from A to B:

$$(f \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n}))(a_1, \dots, a_m) = f(a_{i_1}, \dots, a_{i_n}).$$

The operation $f \circ (\pi^m_{i_1}, \pi^m_{i_2}, \dots, \pi^m_{i_n})$ is also called a *minor* of f.

A *minion* on (A, B) is a nonempty subset $\mathcal{M}$ of $\mathrm{Op}(A,B)$ that is closed under composition with projections; in other words, it is closed under taking minors.

A suitable analogue of a relation in this context is a pair of relations of the same arity, one on A and the other one on B. Although we will not need a generalized concept of preservation, we state it for comparison. We say that an n-ary operation f from A to B *preserves* a pair of relations (R, S), where R is an m-ary relation on A and S is an m-ary relation on S, if the following implication holds:

$$\begin{pmatrix} r_{11} \\ r_{12} \\ \vdots \\ r_{1m} \end{pmatrix}, \begin{pmatrix} r_{21} \\ r_{22} \\ \vdots \\ r_{2m} \end{pmatrix}, \dots, \begin{pmatrix} r_{n1} \\ r_{n2} \\ \vdots \\ r_{nm} \end{pmatrix} \in R \implies \begin{pmatrix} f(r_{11}, r_{21}, \dots, r_{n1}) \\ f(r_{12}, r_{22}, \dots, r_{n2}) \\ \vdots \\ f(r_{1m}, r_{2m}, \dots, r_{nm}) \end{pmatrix} \in S$$

The set of all operations of A to B preserving a set of pairs of relations is always a minion.

Finally, we generalize minions to multi-sorted minions in a similar way as we generalized clones to multi-sorted clones. Let $\boldsymbol{A} = (A_1, A_2, \dots, A_k)$ and $\boldsymbol{B} = (B_1, B_2, \dots, B_k)$ be k-sorted sets A k-sorted n-ary operation from $\boldsymbol{A}$ to $\boldsymbol{B}$ is a k-tuple $\boldsymbol{f} = (f_1, f_2, \dots, f_k)$, where f_i is an n-ary operation from A_i to B_i. Note that k-sorted operations from $\boldsymbol{A}$ to $\boldsymbol{B}$ can still be composed with k-sorted projections. Explicitly, if $\boldsymbol{f} = (f_1, f_2, \dots, f_k)$ is a k-sorted n-ary operation from $\boldsymbol{A}$ to $\boldsymbol{B}$, m is a positive integer, and $i_1, \dots, i_n \in [m]$, then the j-th component ($j \in [k]$) of $\boldsymbol{f} \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_m})$ is $f_j \circ (\pi^m_{i_1}, \dots, \pi^m_{i_m})$, which is an m-ary operation from A_j to B_j. As above, the obtained k-sorted operation from $\boldsymbol{A}$ to $\boldsymbol{B}$ is called a *minor* of $\boldsymbol{f}$.

Definition 1.3. Let $\boldsymbol{A}$ and $\boldsymbol{B}$ be k-sorted sets. *A k-sorted minion* $\mathcal{M}$ on $(\boldsymbol{A}, \boldsymbol{B})$ is a nonempty set of k-sorted operations from $\boldsymbol{A}$ to $\boldsymbol{B}$ which is closed under taking minors, that is, for every $\boldsymbol{f} \in \mathcal{M}$, every positive integer m, and every $i_1, \dots, i_n \in [m]$ (where n is the arity of $\boldsymbol{f}$), we have

$$\boldsymbol{f} \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_m}) \in \mathcal{M}.$$

For a positive integer n, we denote by $\mathcal{M}^{(n)}$ the set of all n-ary members of $\mathcal{M}$ and call it the *n-ary part of* $\mathcal{M}$.

Note that every multi-sorted clone on $\boldsymbol{A}$ is a multi-sorted minion on $(\boldsymbol{A}, \boldsymbol{A})$.

1.3 Minion homomorphism

When ordering clones, there are various methods to consider. In this section, we will focus on a preordering that is determined by the existence of a minion homomorphism.

Definition 1.4 (Minion homomorphism)**.** Let $\mathcal{M}$, $\mathcal{N}$ be multi-sorted minions. A map $\xi : \mathcal{M} \to \mathcal{N}$ is called *minion homomorphism* if

- it preserves arities
- it *preserves minors*, that is, for all n-ary multi-sorted operations $\boldsymbol{f} \in \mathcal{M}$ and all $m \in \mathbb{N}$, $i_1, i_2, \ldots, i_n \in [m]$,

$$\xi(\boldsymbol{f} \circ (\boldsymbol{\pi}^m_{i_1}, \ldots, \boldsymbol{\pi}^m_{i_n})) = \xi(\boldsymbol{f}) \circ (\boldsymbol{\pi}^m_{i_1}, \ldots, \boldsymbol{\pi}^m_{i_n})$$

For a positive integer n we denote by $\xi^{(n)}$ the n-ary part of ξ, that is,

$$\xi^{(n)} = \xi|_{\mathcal{M}^{(n)}} : \mathcal{M}^{(n)} \to \mathcal{N}^{(n)}.$$

We remark that the projections on the left hand side of the above equations can have different number of sorts and be on different multi-sorted sets. The notation is abused in this way for simplicity and hopefully will not cause confusion.

Using minion homomorphisms, we define a preordering on the class of all multi-sorted minions.

Definition 1.5 (Preordering)**.** Let $\mathcal{M}$ and $\mathcal{N}$ be two multi-sorted minions. We say that $\mathcal{M}$ is less than or equal to $\mathcal{N}$, denoted by $\mathcal{M} \leq \mathcal{N}$, if there exists a minion homomorphism from $\mathcal{M}$ to $\mathcal{N}$.

The relation $\leq$ is a preordering on the class of all minions, that is, a reflexive and transitive relation. Reflexivity is clear as the identity mapping from a minion to itself is a minion homomorphism. Moreover, if there exist minion homomorphisms from $\mathcal{M}$ to $\mathcal{N}$ and from $\mathcal{N}$ to $\mathcal{L}$, then their composition is a minion homomorphism from $\mathcal{M}$ to $\mathcal{L}$.

The preodering induces an equivalence in the standard way as follows.

Definition 1.6 (Equivalence)**.** Let $\mathcal{M}$ and $\mathcal{N}$ be two multi-sorted minion. We say that $\mathcal{M}$ is *equivalent* to $\mathcal{N}$, denoted by $\mathcal{M} \sim \mathcal{N}$, if there exist minion homomorphisms from $\mathcal{M}$ to $\mathcal{N}$ and from $\mathcal{N}$ to $\mathcal{M}$.

The thesis studies the partially ordered class whose elements are $\sim$-equivalence classes of multi-sorted clones (taken from some collection of multi-sorted clones) ordered by $\leq$. We informally say that we order multi-sorted clones by minion homomorphisms, although the elements of the partially ordered class are $\sim$-equivalence classes rather than single clones.

The largest multi-sorted minion is the unique one-sorted clone $\mathcal{T}$ on the set $\{1\}$:

$$\mathcal{T} = \text{ the clone on } \{1\}$$

Indeed, the unique mapping from a multi-sorted minion to $\mathcal{T}$ is clearly a minion homomorphism.

The smallest multi-sorted minion is the one-sorted clone $\mathcal{P}$ on the set $\{0, 1\}$ containing only projections.

$$\mathcal{P} = \text{ the clone of projections on } \{0,1\}$$

1.4 Idempotent clones and minion cores

An operation $f\colon A^n \to A$ is *idempotent* if $f(a, a, \dots, a) = a$ for every $a \in A$. In other words, f is idempotent if it preserves all the singleton unary relations $\{a\}$.

Similarly, a multi-sorted operation $\boldsymbol{f} = (f_1, \dots, f_k)$ is *idempotent* if so are all the f_i. For a fixed k-sorted set $\boldsymbol{A} = (A_1, \dots, A_k)$, we denote by $\mathcal{I}(\boldsymbol{A})$ the k-sorted clone of all idempotent operations on $\boldsymbol{A}$:

$$\mathcal{I}(\boldsymbol{A}) = \{(f_1, \dots, f_k) \mid (f_1, \dots, f_k) \text{ is a } k\text{-sorted idempotent operation on } \boldsymbol{A}\}.$$

For a set θ of multi-sorted relations on $\boldsymbol{A}$, we write $\operatorname{IdPol}(\theta)$ for the idempotent part of $\operatorname{Pol}(\theta)$:

$$\operatorname{IdPol}(\theta) = \operatorname{Pol}(\theta) \cap \mathcal{I}$$

A well-known fact [1] is that each clone is $\sim$-equivalent to an idempotent clone, so-called idempotent core of that clone. This fact extends to multi-sorted clones in a straightfoward way. The following formulation will be convenient for our purposes.

Theorem 1.7. *Let $\boldsymbol{A} = (A_1, \dots, A_k)$ be a k-sorted set with each A_i finite and let θ be a set of k-sorted relations on $\boldsymbol{A}$ of arity at most m such that $\operatorname{Pol}(\theta) \not\sim \mathcal{T}$. Then there exists a k'-sorted set $\boldsymbol{B} = (B_1, B_2, \dots, B_{k'})$ and a set θ' of k'-sorted relations of arity at most m such that*

- $1 < |B_i| \leq \max_{j \in \{1,\dots,k\}} |A_j|$ *for every* $i \in [k']$
- $\operatorname{Pol}(\theta) \sim \operatorname{IdPol}(\theta')$

Proof. We prove this theorem in two steps. Firstly, we find a set of relations θ' such that $\operatorname{Pol}(\theta)$ is equivalent to $\operatorname{Pol}(\theta')$ for some set of multi-sorted relations θ'. Then, we show that $\operatorname{Pol}(\theta')$ is equivalent to $\operatorname{IdPol}(\theta')$. Finally we "remove" one-element sets.

Let $\boldsymbol{f} = (f_1, \dots, f_k)$ be a unary k-sorted operation in $\operatorname{Pol}(\theta)$ such that its image $(f_1(A_1), \dots, f_k(A_k))$ is minimal with respect to componentwise inclusion. Define the set $\boldsymbol{B}$ as

$$\boldsymbol{B} = (B_1, \dots, B_k) = (f_1(A_1), \dots, f_k(A_k))$$

Define θ' in a natural way using restrictions of relations in θ, i.e.,

$$\begin{aligned}\theta' = \{&R \cap (B_{i_1} \times \dots \times B_{i_m}) \mid R \in \theta,\\ &R \text{ is a } k\text{-sorted relation of type } (i_1, \dots, i_m),\ m \in \mathbb{N}\}\end{aligned}$$

Let us observe a significant property of the set $\boldsymbol{B}$. For any unary k-sorted operation $\boldsymbol{g} = (g_1, \dots, g_k) \in \operatorname{Pol}(\theta')$ and any $i \in [k]$, the function g_i is a bijection on B_i. Suppose, for the sake of contradiction, that there exists some $j \in [k]$ for which g_j is not a bijection. In such a case, consider the composition $\boldsymbol{g} \circ \boldsymbol{f}$. It is a unary polymorphism of θ and, moreover, the image of $\boldsymbol{g} \circ \boldsymbol{f}$ is smaller, a contradiction to the minimality of $\boldsymbol{B}$.

Now we show that $\operatorname{Pol}(\theta) \sim \operatorname{Pol}(\theta')$.

In the left-to-right direction, we define a minion homomorphism $\xi : \mathrm{Pol}(\theta) \to \mathrm{Pol}(\theta')$ as follows:

$$\xi(\boldsymbol{h}) = (f_1 \circ h_1 \restriction_{B_1}, \dots, f_k \circ h_k \restriction_{B_k}),$$

where $\boldsymbol{h} = (h_1, \dots, h_k)$ is a k-sorted n-ary operation $\boldsymbol{A} \to \boldsymbol{A}$.

Firstly, we show that this map is well-defined. Assume that $S \in \theta'$ is an m-ary k-sorted relation of type $(i_1, \dots, i_m)$, and $(r_{11}, \dots, r_{1m}), (r_{21}, \dots, r_{2m}), \dots, (r_{n1}, \dots, r_{nm}) \in S$. Then, from the definition of θ', it follows that $(r_{11}, \dots, r_{1m}), (r_{21}, \dots, r_{2m}), \dots, (r_{n1}, \dots, r_{nm}) \in R$ for some $R \in \theta$. Therefore, as both $\boldsymbol{h}$ and $\boldsymbol{f}$ are polymorphisms of θ, we obtain: $(f_{i_1} \circ h_{i_1}(r_{11}, r_{21}, \dots, r_{n1}), \dots, f_{i_m} \circ h_{i_m}(r_{1m}, r_{2m}, \dots, r_{nm}) \in R$. Moreover, it follows from the definition of $\boldsymbol{f}$ that $(f_{i_1} \circ h_{i_1}(r_{11}, r_{21}, \dots, r_{n1}), \dots, f_{i_m} \circ h_{i_m}(r_{1m}, r_{2m}, \dots, r_{nm})) \in B$. Therefore, $(f_1 \circ h_1 \restriction_{B_1}, \dots, f_k \circ h_k \restriction_{B_k})$ is a polymorphism of θ', and the map ξ is well-defined

Furthermore, ξ is a minion homomorphism, as shown in the following computation:

$$\begin{aligned}
&\xi(\boldsymbol{h} \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_n})) \\
&\qquad = \xi((h_1(\pi^m_{i_1}, \dots, \pi^m_{i_n}), \dots, h_k(\pi^m_{i_1}, \dots, \pi^m_{i_n}))) \\
&\qquad = (f_1 \circ h_1(\pi^m_{i_1}, \dots, \pi^m_{i_n}), \dots, f_k \circ h_k(\pi^m_{i_1}, \dots, \pi^m_{i_n})) \\
&\qquad = (f_1 \circ h_1, \dots, f_k \circ h_k) \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_n}) \\
&\qquad = \xi(\boldsymbol{h}) \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_n})
\end{aligned}$$

In the other direction, we define a map $\zeta : \mathrm{Pol}(\theta') \to \mathrm{Pol}(\theta)$ as follows:

$$\zeta(\boldsymbol{h}) = \boldsymbol{h}' = (h'_1, \dots, h'_k),$$

where for each $i \in [k]$ and for each n-tuple $\boldsymbol{a} = (a_1, \dots, a_n) \in A_i^n$:

$$h'_i(\boldsymbol{a}) = h_i(f_i(a_1), \dots, f_i(a_n))$$

Assume R is an m-ary k-sorted relation of type $(i_1, \dots, i_m)$ in θ, and $(a_{11}, \dots, a_{1m}), (a_{21}, \dots, a_{2m}), \dots, (a_{n1}, \dots, a_{nm}) \in R$. Then, as $\boldsymbol{f}$ is a polymorphism of θ, $(f_{i_1}(a_{11}), f_{i_2}(a_{12}), \dots, f_{i_m}(a_{1m})), \dots, (f_{i_1}(a_{n1}), f_{i_2}(a_{n2}), \dots, f_{i_m}(a_{nm})) \in R$. Now, as the image of $\boldsymbol{f}$ is the domain of $\boldsymbol{h}$, we can apply $\boldsymbol{h}$ and obtain $h_{i_1}(f_{i_1}(a_{11}), \dots, f_{i_1}(a_{n1})), \dots, h_{i_m}(f_{i_m}(a_{1m}), \dots, f_{i_m}(a_{nm}))$. Moreover, as $(f_{i_1}(a_{11}), f_{i_2}(a_{12}), \dots, f_{i_m}(a_{1m})), \dots, (f_{i_1}(a_{n1}), f_{i_2}(a_{n2}), \dots, f_{i_m}(a_{nm})) \in R \cap (B_{i_1} \times \dots, \times B_{i_m}) \in \theta'$ and $\boldsymbol{h}$ is a polymorphism of θ', we obtain $h_{i_1}(f_{i_1}(a_{11}), \dots, f_{i_1}(a_{n1})), \dots, h_{i_m}(f_{i_m}(a_{1m}), \dots, f_{i_m}(a_{nm})) \in R \cap (B_{i_1} \times \dots, \times B_{i_m}) \in \theta'$, which implies that $h_{i_1}(f_{i_1}(a_{11}), \dots, f_{i_1}(a_{n1})), \dots, h_{i_m}(f_{i_m}(a_{1m}), \dots, f_{i_m}(a_{nm})) \in R$. Thus we have shown that ζ is well-defined.

Moreover, $\boldsymbol{h}'$ is indeed a minion homomorphism as shown in the following computation. Assume $\boldsymbol{a}_1 = (a_{11}, \dots, a_{1m}) \in A_1^m, \dots, \boldsymbol{a}_k = (a_{k1}, \dots, a_{km}) \in A_k^m$. Then for each $i \in [k]$:

$$h'_i \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n})(\boldsymbol{a}_i) = h'_i(a_{ii_1}, \dots, a_{ii_n}) = h_i(f_i(a_{ii_1}), \dots, f_i(a_{ii_n}))$$

Therefore

$$
\begin{aligned}
&\zeta(\boldsymbol{h}) \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_n})(\boldsymbol{a}_1, \dots, \boldsymbol{a}_k) = \boldsymbol{h}' \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_n})(\boldsymbol{a}_1, \dots, \boldsymbol{a}_k) \\
&= (h'_1 \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n})(\boldsymbol{a}_1), \dots, h'_k \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n})(\boldsymbol{a}_k)) \\
&= (h_1(f_1(a_{1i_1}), \dots, f_1(a_{1i_n})), \dots, h_k(f_k(a_{ki_1}), \dots, f_k(a_{ki_1}))) \\
&= (h_1 \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n})(f_1(a_{11}), \dots, f_1(a_{1m})), \\
&\dots, h_k \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n})(f_k(a_{k1}), \dots, f_k(a_{km}))) \\
&= \zeta(h_1 \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n}), \dots, h_k \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n}))(\boldsymbol{a}_1, \dots, \boldsymbol{a}_k) \\
&= \zeta(\boldsymbol{h} \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_n}))(\boldsymbol{a}_1, \dots, \boldsymbol{a}_k)
\end{aligned}
$$

As this holds for an arbitrary $\boldsymbol{a}_1, \dots, \boldsymbol{a}_k$, we conclude that ζ is a minion homomorphism.

Thus far, we have shown that $\mathrm{Pol}(\theta) \sim \mathrm{Pol}(\theta')$.

Next, we prove that $\mathrm{Pol}(\theta') \sim \mathrm{IdPol}(\theta')$. The existence of a right-to-left minion homomorphism is obvious (it is just inclusion).

For the reverse direction, consider a polymorphism $\boldsymbol{h} = (h_1, \dots, h_k)$ of θ'. We define a map $\boldsymbol{z}$ as follows:

$$
\boldsymbol{z} = (z_1, \dots, z_k) = \boldsymbol{h} \circ (\underbrace{\boldsymbol{\pi}^1_1, \dots, \boldsymbol{\pi}^1_1}_{n \text{ times}})
$$

In other words, for each $i \in [k]$ and for each $a \in A_i$, we have $z_i(a) = h_i(\underbrace{a, \dots, a}_{n \text{ times}})$.

The map $\boldsymbol{z}$ is a polymorphism of θ, therefore it's a bijection and there exists an inverse $\boldsymbol{z}^{-1}$:

$$
\boldsymbol{z}^{-1} = (z_1^{-1}, \dots, z_k^{-1})
$$

We define a map ψ in the following way:

$$
\psi(\boldsymbol{h}) = \boldsymbol{z}^{-1} \circ \boldsymbol{h}
$$

Notice that $\boldsymbol{z}^{-1} \circ \boldsymbol{h}$ is idempotent, as for each $\boldsymbol{a} = (a_1, \dots, a_k) \in A_1 \times \dots \times A_k$:

$$
\begin{aligned}
&\boldsymbol{z}^{-1} \circ \boldsymbol{h}(\underbrace{\boldsymbol{a}, \dots, \boldsymbol{a}}_{n \text{ times}}) = \boldsymbol{z}^{-1}(h_1(\underbrace{a_1, \dots, a_1}_{n \text{ times}}), \dots, h_k(\underbrace{a_k, \dots, a_k}_{n \text{ times}})) \\
&= (z_1^{-1}(h_1(\underbrace{a_1, \dots, a_1}_{n \text{ times}})), \dots, z_k^{-1}(h_k(\underbrace{a_k, \dots, a_k}_{n \text{ times}})) = (a_1, \dots, a_k)
\end{aligned}
$$

Therefore the map ψ is well-defined.

Moreover, ψ is a minion homomorphism:

$$
\begin{aligned}
&\psi(\boldsymbol{h} \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_n})) = \psi(h_1(\pi^m_{i_1}, \dots, \pi^m_{i_n}), \dots, h_k(\pi^m_{i_1}, \dots, \pi^m_{i_n})) \\
&= \boldsymbol{z}^{-1} \circ (h_1(\pi^m_{i_1}, \dots, \pi^m_{i_n}), \dots, h_k(\pi^m_{i_1}, \dots, \pi^m_{i_n})) \\
&= (z_1^{-1} \circ h_1(\pi^m_{i_1}, \dots, \pi^m_{i_n}), \dots, z_k^{-1} \circ h_k(\pi^m_{i_1}, \dots, \pi^m_{i_n})) \\
&= (\boldsymbol{z}^{-1} \circ \boldsymbol{h}) \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_n}) = \psi(\boldsymbol{h}) \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_n})
\end{aligned}
$$

The only thing we have to ensure is that $|B_i| > 1$. Without the loss of generality we assume that $|B_1| = 1$, i.e. there exist some b such that $B_1 = \{b\}$. We show that $\mathrm{IdPol}(\theta') \sim \mathrm{IdPol}(\theta'')$, where θ'' is the set of $(k-1)$-sorted relations on $B_2 \times \cdots \times B_k$, obtained in a way that we remove the first coordinate from the relations in θ'. In left-to-right direction we define minion homomorphism τ in a natural way by removing the first coordinate of $\boldsymbol{h} = (h_1, \ldots, h_k)$ i.e.:

$$\tau(h_1, \ldots, h_k) = (h_2, \ldots, h_k)$$

In the other direction we define a minion homomorphism μ as

$$\mu((h_2, \ldots, h_k)) = (h_1, h_2, \ldots, h_k),$$

where $h_1(\underbrace{b, \ldots, b}_{n \text{ times}}) = b$.

In both cases it is easy to see that the maps are correctly defined minion homomorphisms.

In this way we iteratively "remove" one-element sets from $\boldsymbol{B}$. Consequently, we obtain the set of k'-sorted relations on $B_1 \times \cdots \times B_{k'}$, where $k' \leq k$, and $|B_i| > 1$ for each $i \in [k']$. Also note that if all the B_i are one-element, then $\mathrm{Pol}(\theta) \sim \mathcal{T}$, which we assumed is not the case. □

We will use a different version of a core. To distinguish them from idempotent cores we call them minion cores.

Definition 1.8 (Minion core)**.** A multi-sorted minion $\mathcal{A}$ is called a *minion core* if every minion homomorphism from $\mathcal{A}$ to itself is a minion automorphism, i.e., a bijective minion homomorphism.

It turns out that every (multisorted) minion has a unique (multisorted) minion core; we will not prove this result in general in this thesis. The concept of minion cores is important in the study of clones, as it provides a canonical representation for each clone, and allows us to compare clones by comparing their corresponding minion cores.

2. Boolean multi-sorted clones

In this chapter we work with the k-sorted set

$$\boldsymbol{A} = (A_1, A_2, \ldots, A_k), \quad A_1 = A_2 = \cdots = A_k = \{0, 1\},$$

where k is a positive integer. The k-sorted operations on $\boldsymbol{A}$ are called *k-sorted Boolean operations*, they are k-tuples of Boolean operations $\{0,1\}^n \to \{0,1\}$. The k-sorted clones on $\boldsymbol{A}$ are called *k-sorted Boolean clones*.

The goal is to describe the ordering by minion homomorphisms in the class of all multi-sorted Boolean clones of the form $\mathrm{Pol}(\theta)$, where θ is a set of multi-sorted at most binary relations. This will be done by computing all possible minion cores of these clones and the ordering between them.

Theorem 1.7 will allow us to concentrate on idempotent multi-sorted clones. We use the notation $\mathcal{I}_k$ instead of $\mathcal{I}(\boldsymbol{A})$.

$$\mathcal{I}_k = \text{ all idempotent } k\text{-sorted Boolean operations}$$

We denote by $\leq$ the natural ordering of $\{0, 1\}$, i.e., $0 \leq 1$. This ordering is extended to tuples and Boolean operations: for $\boldsymbol{x} = (x_1, \ldots, x_n)$, $\boldsymbol{y} = (y_1, \ldots, y_n) \in \{0,1\}^n$, and $f, g \in \{0,1\}^n \to \{0,1\}$ we define

- $\boldsymbol{x} \leq \boldsymbol{y}$ if $x_i \leq y_i$ for every $i \in [n]$, and
- $f \leq g$ if $f(\boldsymbol{x}) \leq g(\boldsymbol{x})$ for every $\boldsymbol{x} \in \{0,1\}^n$.

We denote by $\wedge$ and $\vee$ the binary minimum and maximum operation on $\{0, 1\}$, respectively, and extend it to Boolean operations.

- $x \wedge y = \min\{x, y\}$, $x \vee y = \max\{x, y\}$
- $(f \wedge g)(\boldsymbol{x}) = f(\boldsymbol{x}) \wedge g(\boldsymbol{x})$, $(f \vee g)(\boldsymbol{x}) = f(\boldsymbol{x}) \vee g(\boldsymbol{x})$ for every $\boldsymbol{x} \in \{0,1\}^n$

Finally, we denote by $\overline{x}$ the "negation" of x and extend the notation to tuples:

- $\overline{x} = 1 - x$,
- $\overline{(x_1, x_2, \ldots, x_n)} = (\overline{x_1}, \overline{x_2}, \ldots, \overline{x_n})$.

2.1 Boolean operations

We will see (cf. Example 2.7) that a $(f_1, \ldots, f_n)$ is in $\mathrm{Pol}(\theta)$ for a set binary multi-sorted relations θ if certain relationship hold between the f_i. We now introduce two of them and prove some basic properties.

Definition 2.1. ($\triangleleft$) Let f and g be two n-ary Boolean operations. We use the notation $f \triangleleft g$ to indicate that for every pair of n-tuples $(\boldsymbol{x}, \boldsymbol{y})$ such that $\boldsymbol{x} \leq \boldsymbol{y}$, it holds that $f(\boldsymbol{x}) \leq g(\boldsymbol{y})$.

Note that while $f \leq f$ holds for every Boolean operation, the relationship $f \triangleleft f$ is nontrivial. Operations satisfying $f \triangleleft f$ are called *monotone*.

Definition 2.2. (Dual) Let f be an n-ary Boolean operation. The dual of f, denoted by f^d, is defined as follows:

$$f^d(\boldsymbol{x}) = 1 - f(\overline{\boldsymbol{x}})$$

where $\overline{\boldsymbol{x}}$ denotes the n-tuple obtained by negating each component of $\boldsymbol{x}$, i.e., if $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, then $\overline{\boldsymbol{x}} = (1 - x_1, 1 - x_2, \ldots, 1 - x_n)$.

Lemma 2.3. *Let f, g be two n-ary Boolean operations. Then the following statements are equivalent:*

(i) $f \lhd g$

(ii) there exists a monotone function g' such that $f \leq g' \leq g$

Proof. To prove the left-to-right implication assume that for each pair of n-tuples $(\boldsymbol{x}, \boldsymbol{y})$ such that $\boldsymbol{x} \leq \boldsymbol{y}$, we have $f(\boldsymbol{x}) \leq g(\boldsymbol{y})$. We will show that $f \lhd g$, i.e. that there exists a monotone Boolean operation g' such that $f \leq g' \leq g$.

Define $g'(\boldsymbol{x})$ to be the maximum of all values of $f(\boldsymbol{y})$ such that $\boldsymbol{y} \leq \boldsymbol{x}$. In other words, $g'(\boldsymbol{x}) = \max\{f(\boldsymbol{y}) \mid \boldsymbol{y} \leq \boldsymbol{x}\}$. Note that g' is monotone because if $\boldsymbol{x} \leq \boldsymbol{y}$, then for any $\boldsymbol{z} \leq \boldsymbol{x}$ also $\boldsymbol{z} \leq \boldsymbol{x} \leq \boldsymbol{y}$, so $\{f(\boldsymbol{z}) \mid \boldsymbol{z} \leq \boldsymbol{x}\}$ is a subset of $\{f(\boldsymbol{z}) \mid \boldsymbol{z} \leq \boldsymbol{y}\}$ and therefore $g'(\boldsymbol{x}) = \max\{f(\boldsymbol{z}) \mid \boldsymbol{z} \leq \boldsymbol{x}\} \leq \max\{f(\boldsymbol{z}) \mid \boldsymbol{z} \leq \boldsymbol{y}\} = g'(\boldsymbol{y})$.

We claim that $f \leq g' \leq g$. To see this, note that $f \leq g'$ by construction of g'. Moreover, for $\boldsymbol{x} \leq \boldsymbol{y}$ we have $f(\boldsymbol{x}) \leq g(\boldsymbol{y})$ by assumption, and therefore $g'(\boldsymbol{x}) = \max\{f(\boldsymbol{y}) \mid \boldsymbol{y} \leq \boldsymbol{x}\} \leq g(\boldsymbol{y})$ for any $\boldsymbol{y}$ such that $\boldsymbol{x} \leq \boldsymbol{y}$. Hence, $g'(\boldsymbol{x}) \leq g(\boldsymbol{y})$ for all $\boldsymbol{x}$ and $\boldsymbol{y}$ such that $\boldsymbol{x} \leq \boldsymbol{y}$, which implies $g' \leq g$.

Therefore, we have shown that $f \leq g' \leq g$ for some monotone Boolean operation g'.

To prove the right-to-left implication we assume that there exists a monotone Boolean operation g' such that $f \leq g' \leq g$. Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be n-tuples such that $\boldsymbol{x} \leq \boldsymbol{y}$. Since g' is monotone, we have $g'(\boldsymbol{x}) \leq g'(\boldsymbol{y})$. Therefore, using the fact that $f \leq g' \leq g$, we obtain:

$$f(\boldsymbol{x}) \leq g'(\boldsymbol{x}) \leq g'(\boldsymbol{y}) \leq g(\boldsymbol{y})$$

This completes the proof of the right-to-left implication. □

Lemma 2.4 (Properties of Duality)**.** *Let f, g, and h be n-ary Boolean operations. The following statements hold:*

(i) Double Duality: $(f^d)^d = f$.

(ii) Duality of Joins: $(f \vee g)^d = f^d \wedge g^d$ and Duality of Meets: $(f \wedge g)^d = f^d \vee g^d$.

Proof. (i) We prove that f is dual to f^d. Let $\boldsymbol{x}$ be an arbitrary n-tuple. Then:

$$(f^d)^d(\boldsymbol{x}) = \overline{f^d(\overline{\boldsymbol{x}})} = \overline{\overline{f(\boldsymbol{x})}} = f(\boldsymbol{x})$$

Here, the first two equalities follow from the definition of duality. As this holds for any n-tuple $\boldsymbol{x}$, we conclude that f is indeed the dual of f^d, and therefore $(f^d)^d = f$.

(ii) Similarly, we prove that $f^d \wedge g^d$ is the dual of $f \vee g$. Let $\boldsymbol{x}$ be an arbitrary n-tuple. Then:

$$(f \vee g)^d(\boldsymbol{x}) = \overline{(f \vee g)(\overline{\boldsymbol{x}})} = \overline{f(\overline{\boldsymbol{x}}) \vee g(\overline{\boldsymbol{x}})} = \overline{f(\overline{\boldsymbol{x}})} \wedge \overline{g(\overline{\boldsymbol{x}})} = f^d(\boldsymbol{x}) \wedge g^d(\boldsymbol{x})$$

Here, the first equality follows from the definition of duality, the second one from the definition of join, the third one is De Morgan's law, and the last one is again the definition of duality.

Duality of Meets is proven analogously.

□

Lemma 2.5 (Properties of $\lhd$)**.** *Let f, g and h be n-ary Boolean operations. Then the following statements hold:*

(i) $f \lhd g^d$ *if and only if* $g \lhd f^d$

(ii) If either $f \leq g$ *and* $g \lhd h$*, or* $f \lhd g$ *and* $g \leq h$*, then* $f \lhd h$*.*

(iii) If $f \lhd g$ *and* $h \lhd r$*, then* $f \wedge h \lhd g \wedge r$ *and* $f \vee h \lhd g \vee r$*.*

(iv) If $f \lhd g$*, then* $f \leq g$*, i.e., the* $\lhd$ *relation is stronger than the* $\leq$ *relation.*

(v) If f *is monotone, then* f^d*,* $f \wedge f^d$*, and* $f \vee f^d$ *are also monotone. Moreover, for an arbitrary function* g*, the following holds:* $f \wedge g \lhd f \lhd f \vee g$*.*

(vi) If $f \lhd g \lhd f$*, then* $f = g$

Proof. (i) Suppose $f \lhd g^d$. We will assume for the sake of contradiction that $g \not\lhd f^d \in \mathtt{D}$, i.e. there exist n-tuples $\boldsymbol{x}$ and $\boldsymbol{y}$ such that $\boldsymbol{x} < \boldsymbol{y}$, but $g(\boldsymbol{x}) = 1$ and $f^d(\boldsymbol{y}) = 0$. By the definition of duality, this implies that $g^d(\overline{\boldsymbol{x}}) = 0$ and $f(\overline{\boldsymbol{y}}) = 1$. This means according to the definition of dual that $g^d(\overline{\boldsymbol{x}}) = 0$ and $f(\overline{\boldsymbol{y}}) = 1$. If $\boldsymbol{x} < \boldsymbol{y}$, then $\overline{\boldsymbol{y}} < \overline{\boldsymbol{x}}$. Therefore $f \not\lhd g^d$ and we have come to a contradiction with the assumption.

(ii) Suppose $f \lhd g$ and $g \leq h$. Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be n-tuples such that $\boldsymbol{x} \leq \boldsymbol{y}$. Since $f \lhd g$ and $\boldsymbol{x} \leq \boldsymbol{y}$, we have $f(\boldsymbol{x}) \leq g(\boldsymbol{y})$, and since $g \leq h$, we have $g(\boldsymbol{y}) \leq h(\boldsymbol{y})$. Using the fact that $\leq$ is transitive, we can combine these inequalities to get $f(\boldsymbol{x}) \leq g(\boldsymbol{y}) \leq h(\boldsymbol{y})$. Therefore, we have $f(\boldsymbol{x}) \leq h(\boldsymbol{y})$, and since this holds for all n-tuples $\boldsymbol{x} \leq \boldsymbol{y}$, we can conclude that $f \lhd h$.

Now suppose $f \leq g$ and $g \lhd h$. Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be n-tuples such that $\boldsymbol{x} \leq \boldsymbol{y}$. Since $f \leq g$, we have $f(\boldsymbol{x}) \leq g(\boldsymbol{x})$, and since $g \lhd h$, we have $g(\boldsymbol{x}) \leq h(\boldsymbol{y})$. We can combine these inequalities to get $f(\boldsymbol{x}) \leq g(\boldsymbol{x}) \leq h(\boldsymbol{y})$. Therefore, we have $f(\boldsymbol{x}) \leq h(\boldsymbol{y})$, and since this holds for all n-tuples $\boldsymbol{x} \leq \boldsymbol{y}$, we can conclude that $f \lhd h$.

(iii) We will use definition 2.1 Assume that $\boldsymbol{x} \leq \boldsymbol{y}$. Then $f(\boldsymbol{x}) \leq g(\boldsymbol{y})$ and $h(\boldsymbol{x}) \leq r(\boldsymbol{y})$. Therefore from the definition of $\wedge$: $f(\boldsymbol{x}) \wedge h(\boldsymbol{x}) \leq g(\boldsymbol{y}) \wedge r(\boldsymbol{y})$. This means that $(f \wedge h)(\boldsymbol{x}) \leq (g \wedge r)(\boldsymbol{y})$, therefore by 2.1 $f \wedge h \lhd g \wedge r$.

The proof for $\vee$ is analogous.

(iv) Suppose that f and g are n-ary functions and that $f \lhd g$. Then according to Theorem 2.3 there exists a monotone function such that $f \leq g' \leq g$. Transitivity of $\leq$ implies that $f \leq g$.

(v) Monotonicity of f^d follows from (i) as $f \lhd f$.

For monotonicity of $f \wedge f^d$ we use (iii), as $f \lhd f$ and $f^d \lhd f^d$. Analogously we prove that $f \vee f^d$ is monotone.

For the last inequality we use Theorem 2.3. The following chain of inequalitites holds trivially according to definitions of $\wedge$ and $\vee$:

$$f \wedge g \leq f \lhd f \leq f \vee g$$

Using transitivity from (ii) we obtain:

$$f \wedge g \lhd f \lhd f \vee g,$$

which is what we wanted to prove.

(vi) Assume $f \lhd g \lhd f$. By the property stated in (iv), this implies $f \leq g \leq f$. Since the relation $\leq$ is antisymmetric, it follows that $f = g$.

□

The following lemma will enable us to project a self-dual operation g to an interval $f \leq f^d$.

Lemma 2.6. *Let f and g be functions such that $g = g^d$ and $f \leq f^d$. Define $h = (g \vee f) \wedge f^d$. Then the following statements hold:*

(i) $h = h^d$

(ii) $f \leq h \leq f^d$

(iii) If f is monotone, then $f \lhd h \lhd f^d$

Proof. (i) We have $h^d = ((g \vee f) \wedge f^d)^d = ((g \vee f)^d \vee f) = ((g^d \wedge f^d) \vee f) = ((g \vee f) \wedge (f^d \vee f)) = ((g \vee f) \wedge f^d) = h$. In the first three equalities, we expanded the brackets using Lemma 2.4(ii). Then, in the fourth equality, we applied distributivity and the fact that $g = g^d$. Finally, in the last equality, we utilized the fact that $f \leq f^d$.

(ii) It is clear that $h \leq f^d$ from the definition of $\wedge$. Moreover, from (i), we have $h = h^d = (g \wedge f^d) \vee f$. Since $f \leq f^d$, we know that $f \leq (g \wedge f^d) \vee f$ from the definition of $\vee$. Therefore, we conclude that if $f \leq f^d$, then $f \leq h \leq f^d$.

(iii) The relation $f \leq h \leq f^d$ can be proven in the same way as in (ii). Moreover, since $f \lhd f$ and $f \leq h$, we can apply Lemma 2.5(ii) to conclude that $f \lhd h$. Similarly, we have $h \lhd f^d$.

□

2.2 Description

In this section we show that the clones of our interest can be described by means of $\lhd$, $=$, and duals (Lemma 2.10) and then we simplify the description (Theorem 2.15).

The following example illustrates the idea of the first part.

Example 2.7. *Let $k = 3$ and $\theta = \{S_{11}, N\}$, where $S_{11} = \{(0,0),(0,1),(1,0)\}$ of type $(1,2)$ and $N = \{(0,1),(1,0)\}$ of type $(2,3)$. We claim that*

$$\mathrm{IdPol}(\theta) = \{(f_1, f_2, f_3) \in \mathcal{I} \mid f_1 \lhd f_2^d = f_3\}.$$

(Here $f_1 \lhd f_2^d = f_3$ of course means $f_1 \lhd f_2^d$ and $f_2^d = f_3$.)

Specifically, we show that (f_1, f_2, f_3) preserves S_{11} iff $f_1 \lhd f_2^d$ and that (f_1, f_2, f_3) preserves N iff $f_2^d = f_3$, from which the claim immediately follows.

To prove left to right implication we assume (f_1, f_2, f_3) of type $(1,2,3)$ preserves the binary relation S_{11} of type $(1,2)$. Let $\boldsymbol{x} = (x_1, \ldots, x_n)$ and $\boldsymbol{y} = (y_1, \ldots, y_n)$ be n-tuples such that $\boldsymbol{x} \leq \boldsymbol{y}$. This means that for each $i \in [n]$, $(x_i, y_i) \neq (1,0)$, which is equivalent to $(x_i, \overline{y}_i) \neq (1,1)$, i.e., $(x_i, \overline{y}_i) \in S_{11}$. Based on our assumption, $(f_1(\boldsymbol{x}), f_2(\overline{\boldsymbol{y}})) \in S_{11}$. Since $f_2^d(\boldsymbol{y}) = \overline{f_2(\overline{\boldsymbol{y}})}$, we can conclude that $(f_1(\boldsymbol{x}), f_2^d(\boldsymbol{y})) \in S_{10}$, i.e., $f_1(\boldsymbol{x}) \leq f_2^d(\boldsymbol{y})$, which is what we wanted to prove.

Similarly, we can prove it for the second relation: If $\boldsymbol{x} = \boldsymbol{y}$, then for each $i \in [n]$, $(x_i, y_i) \in \{(1,1),(0,0)\}$, and therefore $(x_i, \overline{y}_i) \notin \{(1,1),(0,0)\}$. This is equivalent to $(x_i, \overline{y}_i) \in S_{00} \cap S_{11}$. According to our assumption, $(f_2(\boldsymbol{x}), f_3(\overline{\boldsymbol{y}})) \in S_{00} \cap S_{11}$. This implies that $(f_2(\boldsymbol{x}), f_3^d(\boldsymbol{y})) \notin S_{00} \cap S_{11}$, i.e., $f_2^d(\boldsymbol{x}) = f_3(\boldsymbol{y})$, which is what we wanted to prove.

For the right-to-left implication, we assume that f_1, f_2, and f_3 are n-ary operations such that $f_1 \lhd f_2^d = f_3$. We also assume that $\boldsymbol{x} = (x_1, \ldots, x_n)$ and $\boldsymbol{y} = (y_1, \ldots, y_n)$ are n-tuples such that for each $i \in [n]$, $(x_i, y_i) \in S_{11}$. Therefore, $(x_i, \overline{y}_i) \in S_{10}$, i.e. $\boldsymbol{x} \leq \overline{\boldsymbol{y}}$. By applying the assumption $f_1 \lhd f_2^d$ on tuples $\boldsymbol{x}$ and $\overline{\boldsymbol{y}}$, we find that $f_1(\boldsymbol{x}) \leq f_2^d(\overline{\boldsymbol{y}})$. This inequality is equivalent to $(f_1(\boldsymbol{x}), f_2^d(\overline{\boldsymbol{y}})) \in S_{10}$. Consequently, we have $(f_1(\boldsymbol{x}), \overline{f_2(\boldsymbol{y})}) \in S_{10}$, and finally, $(f_1(\boldsymbol{x}), f_2(\boldsymbol{y})) \in S_{11}$, which is what we wanted to prove.

Similarly, we can prove it for the second relation: If $(x_i, y_i) \in S_{00} \cap S_{11}$ for each $i \in [n]$, this means that $x_i \neq y_i$, and therefore $x_i = \overline{y}_i$ and $\overline{\boldsymbol{x}} = \boldsymbol{y}$. Applying the assumption yields $f_2^d(\overline{\boldsymbol{x}}) = f_3(\boldsymbol{y})$, i.e., $\overline{f_2(\boldsymbol{x})} = f_3(\boldsymbol{y})$. This means that $(f_2(\boldsymbol{x}), f_3(\boldsymbol{y})) \in S_{00} \cap S_{11}$.

We have shown that $\{(f_1, f_2, f_3) \in \mathcal{I} \mid f_1 \lhd f_2^d = f_3\} = \mathrm{IdPol}(\{S_{11}, N\})$.

We formalize descriptions of clones similar to the example and show that we can work only with such clones.

Definition 2.8 (Description)**.** A description over a sequence of symbols $(\mathtt{f_1}, \mathtt{f_2}, \ldots, \mathtt{f_k})$ is a set of formal expressions, called *constraints*, that can take one of the following forms

$$\mathtt{f_i} \lhd \mathtt{f_j}, \quad \mathtt{f_i} \lhd \mathtt{f_j^d}, \quad \mathtt{f_i^d} \lhd \mathtt{f_j}, \quad \mathtt{f_i^d} \lhd \mathtt{f_j^d}, \quad \mathtt{f_i} = \mathtt{f_j}, \quad \mathtt{f_i} = \mathtt{f_j^d}, \quad \mathtt{f_i^d} = \mathtt{f_j}, \quad \mathtt{f_i^d} = \mathtt{f_j^d},$$

where $i, j \in [k]$ (not necessarily distinct).

If $\mathtt{D}$ is a description over $(\mathtt{f_1}, \ldots, \mathtt{f_k})$, then we define $\mathrm{Clo}(\mathtt{D})$ as the set of all idempotent k-sorted Boolean operations $(f_1, \ldots, f_k) \in \mathcal{I}_k$ which satisfy all the constraints in $\mathtt{D}$ (in the natural sense). We also say that $\mathtt{D}$ *describes* $\mathrm{Clo}(\mathtt{D})$.

Example 2.9. *The clone from Example 2.7 is*

$$\mathrm{Clo}(\{\mathtt{f_1} \lhd \mathtt{f_2^d}, \mathtt{f_3} = \mathtt{f_2^d}\}) = \{(f_1, f_2, f_3) \in \mathcal{I} \mid f_1 \lhd f_2^d = f_3\}.$$

We will also use a shorter notation such as, e.g.,

$$\mathrm{Clo}(\mathtt{f_1} \lhd \mathtt{f_2^d} = \mathtt{f_3}).$$

Lemma 2.10. *For each set θ of at most binary k-sorted relations on the k-sorted $\boldsymbol{A}$, the clone $\mathrm{Pol}(\theta)$ is equivalent to $\mathcal{T}$ or to $\mathrm{Clo}(\mathtt{D})$ for some description $\mathtt{D}$ (possible over less than k-element sequence of symbols).*

Proof. From Theorem 1.7, we deduce the existence of a k'-sorted set $\boldsymbol{B} = (B_1, B_2, \ldots, B_{k'})$ and a set θ' containing k'-sorted relations with arity at most 2. It is ensured that $1 < |B_i| \leq \max_{j \in [k]} |A_j|$ for each $i \in [k']$, and $\mathrm{Pol}(\theta) \sim \mathrm{IdPol}(\theta')$. Since $|A_j| = 2$ for every $j \in [k]$, we conclude that $|B_i| = 2$ for each $i \in [k']$ and we can assume that each B_i is $\{0, 1\}$.

Now we construct a description $\mathtt{D}$ over $(f_1, \ldots, f_{k'}) \in \mathcal{I}$ such that $\mathrm{IdPol}(\theta') = \mathrm{Clo}(\mathtt{D})$.

- For each relation $\{0,1\}^2 \setminus \{(1,1)\} = S_{11}$ in θ' of type (i, j), where $i, j \in [k]$, we add $\mathtt{f_i} \lhd \mathtt{f_j^d}$ to $\mathtt{D}$.
- For each relation $\{0,1\}^2 \setminus \{(0,0)\} = S_{00}$ in θ' of type (i, j), where $i, j \in [k]$, we add $\mathtt{f_i^d} \lhd \mathtt{f_j}$ to $\mathtt{D}$.
- For each relation $\{0,1\}^2 \setminus \{(1,0)\} = S_{10}$ in θ' of type (i, j), where $i, j \in [k]$, we add $\mathtt{f_i} \lhd \mathtt{f_j}$ to $\mathtt{D}$.
- For each relation $\{0,1\}^2 \setminus \{(0,1)\} = S_{01}$ in θ' of type (i, j), where $i, j \in [k]$, we add $\mathtt{f_i^d} \lhd \mathtt{f_j^d}$ to $\mathtt{D}$.
- For each relation $\{(0,0), (1,1)\} = \mathrm{Eq}$ in θ' of type (i, j), where $i, j \in [k]$, we add $\mathtt{f_i} = \mathtt{f_j}$ to $\mathtt{D}$.
- For each relation $\{(0,1), (1,0)\} = \mathrm{Ineq}$ in θ' of type (i, j), where $i, j \in [k]$, we add $\mathtt{f_i} = \mathtt{f_j^d}$ to $\mathtt{D}$.
- Any relation in θ' that is not mentioned above is ignored.

To prove that $\mathrm{IdPol}(\theta') = \mathrm{Clo}(\mathtt{D})$, we consider the inclusion in both directions.

$\subseteq$: Let $\boldsymbol{f} \in \mathrm{IdPol}(\theta')$. Since $\boldsymbol{f}$ is idempotent, it is enough to show that $\boldsymbol{f}$ satisfies all the constraints in $\mathrm{Clo}(\mathtt{D})$. For the constraints of the form $\mathtt{f_i} \lhd \mathtt{f_j^d}$ and $\mathtt{f_i} = \mathtt{f_j^d}$ this was demonstrated in Example 2.7: For instance, if $\mathtt{f_i} \lhd \mathtt{f_j^d}$ is in $\mathtt{D}$, then θ' contains S_{11}, so $\boldsymbol{f}$ preserves S_{11} (since $\boldsymbol{f} \in \mathrm{IdPol}(\theta')$), which implies $f_i \lhd f_j^d$ by that example. The proof for the other relations is completely analogous.

$\supseteq$: Let $\boldsymbol{f} \in \mathrm{Clo}(\mathtt{D})$. Again, $\boldsymbol{f}$ is idempotent. We need to show that $\boldsymbol{f}$ preserves every relation R in θ'. For relations S_{11} and Ineq this was again proved in Example 2.7 (e.g., if $f_i \lhd f_j^d$, then $\boldsymbol{f}$ preserves S_{11}). For relations S_{01}, S_{10}, S_{00} and Eq, the proof is analogous. The remaining relations, which were ignored while defining $\mathtt{D}$, are the following.

- Trivial: $\emptyset, \{0,1\}^2$

- One-element: $\{(0,0)\}\{(0,1)\},\{(1,0)\},\{(1,1)\}$,
- Two-element: $\{(0,0),(0,1)\},\{(0,0),(1,0)\},\{(0,1),(1,1)\},\{(1,0),(1,1)\}$

It is evident that all multi-sorted Boolean operations preserve $\{0,1\}^2$ (of any type) and $\emptyset$. That the one-element and two-element relations are preserved by $\boldsymbol{f}$ follows from idempotency. For example, for $R=\{(0,1)\}$, the implication

$$\underbrace{\begin{pmatrix}0\\1\end{pmatrix},\begin{pmatrix}0\\1\end{pmatrix},\dots,\begin{pmatrix}0\\1\end{pmatrix}}_{n\text{ times}} \implies \begin{pmatrix} f_i(\underbrace{0,0,\dots,0}_{n\text{ times}}) \\ f_j(\underbrace{1,1,\dots,1}_{n\text{ times}}) \end{pmatrix} = \begin{pmatrix}0\\1\end{pmatrix},$$

is satisfied because $f_i(0,0,\dots,0)=0$ and $f_j(1,1,\dots,1)=1$; for $R=\{(0,1),(0,0)\}$, the implication

$$\begin{pmatrix}0\\a_1\end{pmatrix},\begin{pmatrix}0\\a_2\end{pmatrix},\dots,\begin{pmatrix}0\\a_n\end{pmatrix} \implies \begin{pmatrix} f_i(\underbrace{0,0,\dots,0}_{n\text{ times}}) \\ f_j(a_1,a_2,\dots,a_n) \end{pmatrix} \in \left\{\begin{pmatrix}0\\1\end{pmatrix},\begin{pmatrix}0\\0\end{pmatrix}\right\},$$

is satisfied because $f_i(0,0,\dots,0)=0$.

□

Now we move on to the task of making the description simpler. Namely, we elimate expressions of the form $\mathtt{f_i^d} \lhd \mathtt{f_j}$ from $\mathtt{D}$ and make $=$ appear only in a restricted way.

Definition 2.11 (Reduced form)**.** A *description in reduced form* is a description over $(\mathtt{f_1},\dots,\mathtt{f_n},\mathtt{g_1},\dots,\mathtt{g_m})$, where m and n are nonnegative integers such that $m+n\geq 1$, such that

(i) $\mathtt{g_i} = \mathtt{g_i^d}$ is in $\mathtt{D}$ for each $i\in[m]$,

(ii) all the remaining constraints in $\mathtt{D}$ are of the form $\mathtt{f_i}\lhd\mathtt{f_j}$, $\mathtt{f_i}\lhd\mathtt{f_j^d}$, $\mathtt{f_i}\lhd\mathtt{g_j}$, or $\mathtt{g_i}\lhd\mathtt{g_i}$, and

(iii) there are no $\lhd$-cycles of length more than 1, that is, there are no chains $\mathtt{f_{i_1}}\lhd\mathtt{f_{i_2}}\lhd\dots\lhd\mathtt{f_{i_l}}\lhd\mathtt{f_{i_1}}$ in $\mathtt{D}$ with $l>1$. (Here, again, the membership in $\mathtt{D}$ means that $\mathtt{f_{i_1}}\lhd\mathtt{f_{i_2}}$ is in $\mathtt{D}$, $\mathtt{f_{i_2}}\lhd\mathtt{f_{i_3}}$ is in $\mathtt{D}$, etc.)

We aim to show that each Clo($\mathtt{D}$) is equivalent to Clo($\mathtt{D}'$), where the description $\mathtt{D}'$ is reduced.

Minion homomorphisms will be defined by formulas such as

$$\xi(f_1,f_2,f_3) = (f_3, f_2^d, f_1, f_1)$$

and later by more complex formulas, e.g.,

$$\xi(f_1,f_2,f_3) = (f_2^d, (f_3\wedge f_2)\vee f_3^d).$$

We show that each such a formula defines a minor preserving map.

Lemma 2.12. *Let $k, k' \in \mathbb{N}$. Let t_1, t_2, ..., $t_{k'}$ be terms over the set of symbols $\{\mathtt{f_1}, \dots, \mathtt{f}_k\}$ in the signature $\{\wedge, \vee, {}^d\}$. For a k-sorted Boolean n-ary operation $(f_1, \dots, f_k)$ and $j \in [k']$ we define $t_j(f_1, \dots, f_k) : \{0,1\}^n \to \{0,1\}$ in the natural way (replace $\mathtt{f_i}$ by f_i and compute the expression). Then the mapping $\xi : \mathcal{I}_k \to \mathcal{I}_{k'}$ defined by*

$$\xi(f_1, \dots, f_k) = (t_1(f_1, \dots, f_k), t_2(f_1, \dots, f_k), \dots, t_{k'}(f_1, \dots, f_k))$$

is a minion homomorphism.

Proof. First we claim that for every term s over $\{\mathtt{f_1}, \dots, \mathtt{f}_k\}$ in the signature $\{\wedge, \vee, {}^d\}$ there exists a Boolean operation $\tilde{s}$ of arity $2k$ such that for every $n \in \mathbb{N}$, every k-tuple of n-ary Boolean functions $\boldsymbol{f} = (f_1, \dots, f_k)$, and every $\boldsymbol{a} \in \{0,1\}^n$, we have

$$(s(\boldsymbol{f}))(\boldsymbol{a}) = \tilde{s}(f_1(\boldsymbol{a}), \dots, f_k(\boldsymbol{a}), f_1(\overline{\boldsymbol{a}}), \dots, f_k(\overline{\boldsymbol{a}}))).$$

The claim is proved by induction of the depth of s. The base case when $s = \mathtt{f_j}$ for some j is clear. If $s = s_1 \wedge s_2$, we have

$$\begin{aligned}(s(\boldsymbol{f}))(\boldsymbol{a}) &= (s_1(\boldsymbol{f}) \wedge s_2(\boldsymbol{f}))(\boldsymbol{a}) = (s_1(\boldsymbol{f}))(\boldsymbol{a}) \wedge (s_2(\boldsymbol{f}))(\boldsymbol{a}) \\ &= \tilde{s_1}(f_1(\boldsymbol{a}), \dots, f_k(\boldsymbol{a}), f_1(\overline{\boldsymbol{a}}), \dots, f_k(\overline{\boldsymbol{a}})) \\ &\quad \wedge \tilde{s_2}(f_1(\boldsymbol{a}), \dots, f_k(\boldsymbol{a}), f_1(\overline{\boldsymbol{a}}), \dots, f_k(\overline{\boldsymbol{a}})),\end{aligned}$$

so we can define

$$\tilde{s}(a_1, \dots, a_k, b_1, \dots, b_k) = \tilde{s_1}(a_1, \dots, a_k, b_1, \dots, b_k) \wedge \tilde{s}_2(a_1, \dots, a_k, b_1, \dots, b_k).$$

The proof for $s = s_1 \vee s_2$ is completely analogous. Finally, if $s = s_1^d$, then we have

$$\begin{aligned}(s(\boldsymbol{f}))(\boldsymbol{a}) &= (s_1^d(\boldsymbol{f}))(\boldsymbol{a}) = 1 - (s_1(\boldsymbol{f}))(\overline{\boldsymbol{a}}) \\ &= 1 - \tilde{s_1}(f_1(\overline{\boldsymbol{a}}), \dots, f_k(\overline{\boldsymbol{a}}), f_1(\boldsymbol{a}), \dots, f_k(\boldsymbol{a}))\end{aligned}$$

so we can define

$$\tilde{s}(a_1, \dots, a_k, b_1, \dots, b_k) = 1 - \tilde{s_1}(b_1, \dots, b_k, a_1, \dots, a_k).$$

The mapping ξ clearly preserves arities. It also easy to see that $t_j(f_1, \dots, f_k)$ is idempotent, so ξ is correctly defined. It remains to verify that that for every $n \in \mathbb{N}$, every n-ary k-sorted $\boldsymbol{f} = (f_1, \dots, f_k) \in \mathcal{I}_k$, every $m \in \mathbb{N}$, and every i_1, ..., $i_n \in [m]$, we have

$$\xi(\boldsymbol{f}) \circ (\boldsymbol{\pi}_{i_1}^m, \boldsymbol{\pi}_{i_2}^m, \dots, \boldsymbol{\pi}_{i_n}^m) = \xi(\boldsymbol{f} \circ (\boldsymbol{\pi}_{i_1}^m, \boldsymbol{\pi}_{i_2}^m, \dots, \boldsymbol{\pi}_{i_n}^m)).$$

Both sides are k'-tuples of m-ary Boolean operations. We need to show that for each $j \in [k']$, the jth Boolean operations are the same on both sides. That is, we need to prove

$$t_j(\boldsymbol{f}) \circ (\pi_{i_1}^m, \pi_{i_2}^m, \dots, \pi_{i_n}^m) = t_j(\boldsymbol{f} \circ (\boldsymbol{\pi}_{i_1}^m, \boldsymbol{\pi}_{i_2}^m, \dots, \boldsymbol{\pi}_{i_n}^m)).$$

For any tuple $\boldsymbol{a} = (a_1, \dots, a_m) \in \{0,1\}^m$ we have

$$\begin{aligned}
&(t_j(\boldsymbol{f})\circ(\pi^m_{i_1}, \dots, \pi^m_{i_n}))(\boldsymbol{a}) \\
&\qquad = t_j(\boldsymbol{f})(\pi^m_{i_1}(\boldsymbol{a}), \dots \pi^m_{i_n}(\boldsymbol{a})) \\
&\qquad = t_j(\boldsymbol{f})(a_{i_1}, \dots, a_{i_n}) \\
&\qquad = \tilde{t}_j(f_1(a_{i_1}, \dots, a_{i_n}), \dots, f_k(a_{i_1}, \dots, a_{i_n}) \\
&\qquad\qquad f_1(\overline{a_{i_1}}, \dots, \overline{a_{i_n}}), \dots, f_k(\overline{a_{i_1}}, \dots, \overline{a_{i_n}})) \\
&\qquad = \tilde{t_j}(f_1(\pi^m_{i_1}(\boldsymbol{a}), \dots, \pi^m_{i_n}(\boldsymbol{a})), \dots, f_k(\pi^m_{i_1}(\boldsymbol{a}), \dots, \pi^m_{i_n}(\boldsymbol{a})), \\
&\qquad\qquad f_1(\pi^m_{i_1}(\overline{\boldsymbol{a}}), \dots, \pi^m_{i_n}(\overline{\boldsymbol{a}})), \dots, f_k(\pi^m_{i_1}(\overline{\boldsymbol{a}}), \dots, \pi^m_{i_n}(\overline{\boldsymbol{a}}))) \\
&\qquad = \tilde{t_j}((f_1 \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n}))(\boldsymbol{a}), \dots, (f_k \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n}))(\boldsymbol{a}), \\
&\qquad\qquad (f_1 \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n}))(\overline{\boldsymbol{a}}), \dots, (f_k \circ (\pi^m_{i_1}, \dots, \pi^m_{i_n}))(\overline{\boldsymbol{a}})) \\
&\qquad = (t_j(\boldsymbol{f} \circ (\boldsymbol{\pi}^m_{i_1}, \dots, \boldsymbol{\pi}^m_{i_n})))(\boldsymbol{a}),
\end{aligned}$$

as required. □

The second ingredient is a criterion for satisfiability of a 2-CNF from [6], which we now state. *2-CNF* is a formula of propositional logic of a specific form: it is a conjunction of clauses, each clause is a disjunction of two literals, and each literal is either a variable or a negated variable. Such a formula is e.g.

$$(x \text{ or } y) \text{ and } (\neg y \text{ or } z) \text{ and } (y \text{ or } y).$$

(We use a nonstandard notation for disjunction and conjunction so that we do not overload the symbols $\wedge$ and $\vee$ too much.) A 2-CNF is *satisfiable* if there exists an assignment ϕ : variables $\to \{\text{true}, \text{false}\}$ making the formula true.

The *implication graph* of a 2-CNF is the following directed graph. Vertices are the variables and their negations. For each clause, the graph contains the two edges corresponding to the two implications that are logically equivalent to the clause; and there are no other edges. In the example above, the implication graph has edges

$$\neg x \to y,\ \neg y \to x,\ y \to z,\ \neg z \to \neg y,\ \neg y \to y,\ y \to \neg y.$$

The criterion for satisfiability is as follows.

Theorem 2.13 ([6]). *A 2-CNF formula is satisfiable if and only if there is no variable x such that there exists a directed walk $x \to \cdots \to \neg x$ and a directed walk $\neg x \to \cdots \to x$.*

We are ready to reduce the descriptions. At first we give a simple example and then prove the general theorem.

Example 2.14. *Consider the description $\mathtt{D} = \{\mathtt{f}_1^{\mathtt{d}} \lhd \mathtt{f}_2, \mathtt{f}_3 \lhd \mathtt{f}_4 \lhd \mathtt{f}_3^{\mathtt{d}} \lhd \mathtt{f}_3\}$. This description is not in reduced form since it contains the constraint $\mathtt{f}_1^{\mathtt{d}} \lhd \mathtt{f}_2$ and a cycle $\mathtt{f}_3 \lhd \mathtt{f}_4 \lhd \mathtt{f}_3^{\mathtt{d}} \lhd \mathtt{f}_3$. We claim that $\mathrm{Clo}(\mathtt{D})$ is equivalent to $\mathrm{Clo}(\mathtt{D}')$, where $\mathtt{D}' = \{f_1 \lhd f_2^d, g_1 = g_1^d, g_1 \lhd g_1\}$. Note that $\mathtt{D}'$ is in reduced form.*

To demonstrate the equivalence, we define a minion homomorphism $\xi : \mathrm{Clo}(\mathtt{D}) \to \mathrm{Clo}(\mathtt{D}')$ as follows:

$$\xi((f_1, f_2, f_3, f_4)) = (f_1^d, f_2^d, f_3)$$

We need to verify that the constraints in $\mathtt{D}'$ *hold. Since* $f_1^d \lhd f_2$*, and according to Lemma 2.4(i),* $(f_2^d)^d = f_2$*, we can conclude that* $f_1^d \lhd (f_2^d)^d$*. Furthermore, from* $f_3 \lhd f_4 \lhd f_3^d \lhd f_3$*, using Lemma 2.5(vi) and 2.5(ii), we find that* $f_3 = f_4 = f_3^d$*. By replacing* f_4 *with* f_3 *in* $f_3 \lhd f_4$*, we also obtain that* f_3 *is monotone. Therefore, the map* ξ *is well-defined since all the constraints in* $\mathtt{D}'$ *are satisfied. Additionally, this map preserves minors by virtue of Lemma 2.12.*

Conversely, we define a minion homomorphism $\zeta : \mathrm{Clo}(\mathtt{D}') \to \mathrm{Clo}(\mathtt{D})$ *as follows:*

$$\zeta((f_1, f_2, g_1)) = (f_1^d, f_2^d, g_1, g_1)$$

Once again, we verify the constraints in $\mathtt{D}$*. Similarly to before, since* $f_1 \lhd f_2^d$*, Lemma 2.4(i) implies* $(f_1^d)^d \lhd f_2^d$*. Additionally, since* $g_1 = g_1^d$*, replacing the second occurrence of* g_1 *with* g_1^d *in* $g_1 \lhd g_1$ *yields* $g_1 \lhd g_1^d$*. If we replace the first occurrence instead, we obtain* $g_1^d \lhd g_1$*. Thus, we have* $g_1 \lhd g_1 \lhd g_1^d \lhd g_1$*. Similar to the previous case,* ζ *preserves minors as stated in Lemma 2.12.*

Theorem 2.15. *Let* $\mathtt{D}$ *be a description over* $\{\mathtt{h_1}, \ldots, \mathtt{h_k}\}$*. Then there exists a description* $\mathtt{D}'$ *in a reduced form such that the clones* $\mathrm{Clo}(\mathtt{D})$ *and* $\mathrm{Clo}(\mathtt{D}')$ *are equivalent.*

Proof. Let $\mathtt{D}$ be a description over $\{\mathtt{h}_1, \ldots, \mathtt{h}_k\}$. Without loss of generality, assume that there is no description over smaller set of symbols which describes a multi-sorted clone equivalent to $\mathrm{Clo}(\mathtt{D})$ (if there is such, we can replace the original description with one which has the smallest number of symbols and describes an equivalent clone).

A consequence of this *minimality assumption* is that there are no $i, j \in [k]$ with $i \neq j$ such that every $(h_1, \ldots, h_k) \in \mathrm{Clo}(\mathtt{D})$ satisfies $h_i = h_j$. Indeed, if there are such i and j, say $i = 1$ and $j = 2$, then we define $\mathtt{D}'$ over the smaller set of symbols $\{\mathtt{h}_2, \ldots, \mathtt{h_k}\}$ by replacing every occurrence of $\mathtt{h}_1$ by $\mathtt{h}_2$ and every occurrence of $\mathtt{h}_1^d$ by $\mathtt{h}_2^d$. The mapping $\xi : \mathrm{Clo}(\mathtt{D}) \to \mathrm{Clo}(\mathtt{D}')$ defined by $\xi(h_1, \ldots, h_k) = (h_2, \ldots h_k)$ is correctly defined: the constraints in $\mathtt{D}'$ are satisfied since h_1 is always equal to h_2 when $(h_1, \ldots, h_k) \in \mathrm{Clo}(\mathtt{D})$. Also ξ preserves minors by Lemma 2.12, therefore it is a minion homomorphism. On the other hand, it is easy to see that $\xi : \mathrm{Clo}(\mathtt{D}') \to \mathrm{Clo}(\mathtt{D})$ defined by $\xi(h_2, h_3, \ldots, h_k) = (h_2, h_2, h_3, \ldots, h_k)$ is a minion homomorphism. Therefore $\mathrm{Clo}(\mathtt{D})$ is equivalent to $\mathrm{Clo}(\mathtt{D}')$, a contradiction to the minimality assumption.

Similarly, there are no $i, j \in [k]$ with $i \neq j$ such that every $(h_1, \ldots, h_k) \in \mathrm{Clo}(\mathtt{D})$ satisfies $h_i = h_j^d$. The differences in the above argument is that while creating $\mathtt{D}'$ we replace $\mathtt{h}_1$ by $\mathtt{h}_2^\mathtt{d}$ (instead of $\mathtt{h}_2$) and $\mathtt{h}_1^\mathtt{d}$ by $\mathtt{h}_2$, and we define the second homomorphism by $\xi(h_2, h_3, \ldots, h_k) = (h_2^d, h_2, h_3, \ldots, h_k)$. The homomorphisms are correctly defined since $(h_1^d)^d = h_1$ by Lemma 2.4(i).

Now we change the set of symbols $\{\mathtt{h}_1, \ldots, \mathtt{h_k}\}$ of $\mathtt{D}$ (and change the constraints accordingly) to $\{\mathtt{f}_1, \ldots, \mathtt{f_n}, \mathtt{g}_1, \ldots, \mathtt{g_m}\}$ $(n + m = k)$ so that for each $(f_1, \ldots, f_n, g_1, \ldots, g_m) \in \mathrm{Clo}(\mathtt{D})$ we have $g_1 = g_1^d, \ldots, g_m = g_m^d$ and m is the largest with this property, i.e., for any $i \in [m]$ we have $f_i \neq f_i^d$ for some $(f_1, \ldots, f_n, g_1, \ldots, g_m) \in \mathrm{Clo}(\mathtt{D})$.

We make several adjustments to $\mathtt{D}$, none of which changes $\mathrm{Clo}(\mathtt{D})$ because of trivial reasons or We add to $\mathtt{D}$ all the constraints $\mathtt{g_i} = \mathtt{g_i^d}$ $(i \in [m])$, remove redundant constraints of the form $\mathtt{f_i} = \mathtt{f_i}$, $\mathtt{f_i^d} = \mathtt{f_i^d}$, $\mathtt{g_i} = \mathtt{f_i}$, $\mathtt{g_i^{(d)}} = \mathtt{g_i}$, and $\mathtt{g_i^d} = \mathtt{g_i^d}$, replace constraints $\mathtt{f}_\mathtt{i}^\mathtt{d} \lhd \mathtt{f}_\mathtt{j}^\mathtt{d}$ by $\mathtt{f}_\mathtt{j} \lhd \mathtt{f}_\mathtt{i}$ (due to Lemma 2.5(i)), replace $\mathtt{g}_\mathtt{i}^\mathtt{d}$ by $\mathtt{g}_\mathtt{i}$ in every $\lhd$-constraint, and replace $\mathtt{g}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}$ by $\mathtt{f}_\mathtt{j}^\mathtt{d} \lhd \mathtt{g}_\mathtt{i}$ and $\mathtt{g}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}^\mathtt{d}$ by $\mathtt{f}_\mathtt{j} \lhd \mathtt{g}_\mathtt{i}$ (again due to Lemma 2.5(i)). We refer to these adjustments of $\mathtt{D}$ as *simple adjustments*.

The only $=$-constraints left are $\mathtt{g}_\mathtt{i} = \mathtt{g}_\mathtt{i}^\mathtt{d}$ since $\mathtt{f}_\mathtt{i} = \mathtt{f}_\mathtt{i}^\mathtt{d}$ is not in $\mathtt{D}$ by the choice of m and all the other $=$-constraints were either removed or are impossible by the minimality assumption. Note also that $\mathtt{g}_\mathtt{i} \lhd \mathtt{g}_\mathtt{j}$ with $i \neq j$ is not in $\mathtt{D}$, otherwise every $(f_1, \dots, f_n, g_1, \dots, g_m) \in \mathrm{Clo}(\mathtt{D})$ satisfies $g_i \lhd g_j = g_j^d \lhd g_i^d = g_i$, therefore $g_i = g_j$ (due to Lemma 2.5(vi)) , a contradiction to the minimality assumption. The remaining $\lhd$-constraints of $\mathtt{D}$ are thus of the form $\mathtt{f}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}$, $\mathtt{f}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}^\mathtt{d}$, $\mathtt{f}_\mathtt{i}^\mathtt{d} \lhd \mathtt{f}_\mathtt{j}$, $\mathtt{f}_\mathtt{i} \lhd \mathtt{g}_\mathtt{j}$, $\mathtt{f}_\mathtt{i}^\mathtt{d} \lhd \mathtt{g}_\mathtt{j}$, or $\mathtt{g}_\mathtt{i} \lhd \mathtt{g}_\mathtt{i}$. We need to get rid of the cases $\mathtt{f}_\mathtt{i}^\mathtt{d} \lhd \mathtt{f}_\mathtt{j}$ and $\mathtt{f}_\mathtt{i}^\mathtt{d} \lhd \mathtt{g}_\mathtt{j}$.

The description $\mathtt{D}'$ will be obtained from $\mathtt{D}$ by selecting some numbers $j \in \mathbb{N}$ and replacing $\mathtt{f}_\mathtt{j}$ by $\mathtt{f}_\mathtt{j}^\mathtt{d}$ and vice versa. Note that whatever set of numbers we take, the obtained clone $\mathrm{Clo}(\mathtt{D}')$ is equivalent to $\mathrm{Clo}(\mathtt{D})$ via the minion homomorphisms defined in both directions as $\xi(f_1, \dots, f_k, g_1, \dots, g_k) = \xi(f_1', \dots, f_k', g_1, \dots, g_k)$, where $f_j' = f_j^d$ if j was selected and $f_j' = f_j$ otherwise. The numbers will be selected according to a satisfying assignment of a 2-CNF defined as follows. The set of variables of the 2-CNF is $\{\mathtt{f}_1, \dots, \mathtt{f}_\mathtt{n}\}$ and

- for each constraint $\mathtt{f}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}$ in $\mathtt{D}$ we add to the 2-CNF the clause $(\neg \mathtt{f}_\mathtt{i}$ or $\mathtt{f}_\mathtt{j})$,
- for each $\mathtt{f}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}^\mathtt{d}$ in $\mathtt{D}$ we add $(\neg \mathtt{f}_\mathtt{i}$ or $\neg \mathtt{f}_\mathtt{j})$,
- for each $\mathtt{f}_\mathtt{i}^\mathtt{d} \lhd \mathtt{f}_\mathtt{j}$ in $\mathtt{D}$ we add $(\mathtt{f}_\mathtt{i}$ or $\mathtt{f}_\mathtt{j})$,
- for each $\mathtt{f}_\mathtt{i} \lhd \mathtt{g}_\mathtt{j}$ in $\mathtt{D}$ we add $(\neg \mathtt{f}_\mathtt{i}$ or $\neg \mathtt{f}_\mathtt{i})$, and
- for each $\mathtt{f}_\mathtt{i}^\mathtt{d} \lhd \mathtt{g}_\mathtt{j}$ in $\mathtt{D}$ we add $(\mathtt{f}_\mathtt{i}$ or $\mathtt{f}_\mathtt{i})$.

We claim that if $(\neg)\mathtt{f}_\mathtt{i} \to (\neg)\mathtt{f}_\mathtt{j}$ in the implication graph of the 2-CNF formula, then the corresponding relation $f_i^{(d)} \lhd f_j^{(d)}$ holds for every $(f_1, \dots, f_n, g_1, \dots, g_m) \in \mathrm{Clo}(\mathtt{D})$. This follows from Lemma 2.5(i) as follows. In the first item (i.e., $\mathtt{f}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}$ in $\mathtt{D}$) we added $(\neg \mathtt{f}_\mathtt{i}$ or $\mathtt{f}_\mathtt{j})$ to the 2-CNF which gives $\mathtt{f}_\mathtt{i} \to \mathtt{f}_\mathtt{j}$ and $\mathtt{f}_\mathtt{j}^\mathtt{d} \to \mathtt{f}_\mathtt{i}^\mathtt{d}$ in the implication graph; we have $f_i \lhd f_j$ (since $\mathtt{f}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}$ in $\mathtt{D}$) and $f_j^d \lhd f_i^d$ (by that lemma) for every $(f_1, \dots, g_m) \in \mathrm{Clo}(\mathtt{D})$. The second and third items are similar. The fourth item $(\mathtt{f}_\mathtt{i} \lhd \mathtt{g}_\mathtt{j})$ gives $\mathtt{f}_\mathtt{i} \to (\neg \mathtt{f}_\mathtt{i})$ in the implication graph and for any $(f_1, \dots, g_m) \in \mathrm{Clo}(\mathtt{D})$ we have $f_i \lhd g_j = g_j^d \lhd f_i^d$. The fifth item is similar.

It follows that there is no symbol $\mathtt{f}_i$ such that $\mathtt{f}_\mathtt{i} \to \cdots \to \neg \mathtt{f}_\mathtt{i}$ and $\neg \mathtt{f}_\mathtt{i} \to \cdots \to \mathtt{f}_\mathtt{i}$ in the implication graph. Indeed, otherwise $f_i \lhd \cdots \lhd f_i^d \lhd \cdots \lhd f_i$ for every $(f_1, \dots, g_m) \in \mathrm{Clo}(\mathtt{D})$, so $f_i = f_i^d$, a contradiction to the choice of m. Let ϕ be a satisfying assignment to our 2-CNF guaranteed by Theorem 2.13.

We create $\mathtt{D}'$ by replacing $\mathtt{f}_\mathtt{j}$ by $\mathtt{f}_\mathtt{j}^\mathtt{d}$ and $\mathtt{f}_\mathtt{j}^\mathtt{d}$ by $\mathtt{f}_\mathtt{j}$ for every j such that $\phi(\mathtt{f}_\mathtt{j}) = \text{true}$, and making the simple adjustments. Observe that the only $=$-constraints in $\mathtt{D}'$ are still $\mathtt{g}_\mathtt{i} = \mathtt{g}_\mathtt{i}^\mathtt{d}$ (and we have all such in $\mathtt{D}'$) and that all $\lhd$-constraints in $\mathtt{D}'$ are of the form $\mathtt{f}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}$, $\mathtt{f}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}^\mathtt{d}$, $\mathtt{f}_\mathtt{i} \lhd \mathtt{g}_\mathtt{j}$, or $\mathtt{g}_\mathtt{i} \lhd \mathtt{g}_\mathtt{i}$ by the choice of the 2-CNF and the adjustments. It remains to observe that there are no cycles $\mathtt{f}_{\mathtt{i}_1} \lhd \mathtt{f}_{\mathtt{i}_2} \lhd \cdots \lhd \mathtt{f}_{\mathtt{i}_1} \lhd \mathtt{f}_{\mathtt{i}_1}$ in $\mathtt{D}'$ with $l > 1$ since otherwise $f_{i_1} \lhd f_{i_2} \lhd \cdots \lhd f_{i_l} \lhd f_{i_1}$ and thus $f_{i_1} = f_{i_2} = \dots$ for every $(f_1, \dots, g_m) \in \mathrm{Clo}(\mathtt{D}')$, which is impossible by the minimality assumption. Now the description $\mathtt{D}'$ is reduced and $\mathrm{Clo}(\mathtt{D}')$ is equivalent to $\mathrm{Clo}(\mathtt{D})$. The proof is concluded. ☐

2.3 Collapse

In this section we show that every multi-sorted Boolean clone of the form Clo($\mathtt{D}$), where $\mathtt{D}$ is in reduced form, is equivalent to a multi-sorted minion from an explicit collection.

We start by introducing the specific multi-sorted minions. For $k \in \mathbb{N}$ we define:

i. $\mathcal{A}_k = \{(h_1, h_1, \dots, h_k) \in \mathcal{I}_k \mid h_1 \lhd h_2 \lhd \cdots \lhd h_k \leq h_k^d \lhd \cdots \lhd h_2^d \lhd h_1^d\}$

ii. $\mathcal{B}_k = \{(h_1, h_2, \dots, h_k) \in \mathcal{I}_k \mid h_1 \lhd h_2 \lhd \cdots \lhd h_k \lhd h_k^d \lhd \cdots \lhd h_2^d \lhd h_1^d\}$

iii. $\mathcal{C}_k = \{(h_1, h_2, \dots, h_k) \in \mathcal{I}_n \mid h_1 \lhd h_2 \lhd \cdots \lhd h_k = h_k^d \lhd \cdots \lhd h_2^d \lhd h_1^d\}$

iv. $\mathcal{D}_k = \{(h_1, h_2, \dots, h_k, h_{k+1}) \in \mathcal{I}_{k+1} \mid h_1 \lhd h_2 \lhd \cdots \lhd h_k \leq h_{k+1} = h_{k+1} \leq h_k^d \lhd \cdots \lhd h_2^d \lhd h_1^d,\ h_k \lhd h_k^d\}$

v. $\mathcal{X} = \{(h) \in \mathcal{I}_1 \mid h \lhd h \lhd h^d \lhd h^d\}$

vi. $\mathcal{Y} = \{(h_1, h_2) \in \mathcal{I}_2 \mid h_1 \lhd h_1 \lhd h_2 = h_2^d \lhd h_1^d \lhd h_1^d\}$

vii. $\mathcal{W} = \{(h) \in \mathcal{I}_1 \mid h \lhd h = h^d \lhd h^d\}$

Note that all these multi-sorted minions are on $(\boldsymbol{A}, \boldsymbol{A})$, where $\boldsymbol{A} = (\{0,1\}, \dots, \{0,1\})$ is 1-sorted for $\mathcal{X}$ and $\mathcal{W}$, 2-sorted for $\mathcal{Y}$, k-sorted for $\mathcal{A}_k$, $\mathcal{B}_k$, $\mathcal{C}_k$, and $(k+1)$-sorted for $\mathcal{D}_k$. Not all of these multi-sorted minions are multi-sorted clones. This is caused by $\leq$ in the "descriptions". We remark that $f \leq g$ iff (f, g) preserves the pair of relations (Eq, S_{10}) of type $(1, 2)$.

Also notice that the "descriptions" of these minions contain redundant information. For instance, consider $\mathcal{B}_k = \{(h_1, h_2, \dots, h_k) \in \mathcal{I} \mid h_1 \lhd h_2 \lhd \cdots \lhd h_k \lhd h_k^d \lhd \cdots \lhd h_2^d \lhd h_1^d\}$. In this case, the $\lhd$ relations following h_k^d can be inferred from the initial relation $h_1 \lhd h_2 \lhd \cdots \lhd h_k$, as per Lemma 2.5(i).

For each of these specific minions we give a sufficient condition under which a multi-sorted Boolean clone described by a reduced description is equivalent to it. It will follow from the subsequent result that this condition is also necessary, because the above multi-sorted minions are pairwise inequivalent.

For the rest of this section, we fix a description $\mathtt{D}$ over $(\mathtt{f_1}, \dots, \mathtt{f_n}, \mathtt{g_1}, \dots, \mathtt{g_m})$ in a reduced form. We define monotone symbols, ranks of the $\mathtt{f}_i$, and chain rank of $\mathtt{D}$ as follows.

Definition 2.16. We say that a function symbol $\mathtt{f}$ is *monotone* if $\mathtt{f} \lhd \mathtt{f}$ is in $\mathtt{D}$. We say that $\mathtt{D}$ is *monotone-free* if no symbol $\mathtt{f_i}$ or $\mathtt{g_i}$ is monotone.

A *path of length* l to $\mathtt{f_i}$ is a sequence $\mathtt{f_{i_1}} \lhd \mathtt{f_{i_2}} \lhd \cdots \lhd \mathtt{f_{i_{l-1}}} \lhd \mathtt{f_{i_l}}$ in $\mathtt{D}$, where $i_l = i$ and the i_j are pairwise distinct.

We define the *rank* of $\mathtt{f_i}$, denoted as $\mathrm{rank}(\mathtt{f_i})$, to be the length of the longest such path, i.e.,

$$\mathrm{rank}(\mathtt{f_i}) = \max\{l \in \mathbb{N} \mid \mathtt{f_{i_1}} \lhd \mathtt{f_{i_2}} \lhd \cdots \lhd \mathtt{f_{i_{l-1}}} \lhd \mathtt{f_{i_l}} \text{ is a path to } \mathtt{f_i}\}.$$

Finally, the *chain rank* of $\mathtt{D}$ is defined as

$$\begin{aligned}\mathrm{chr}(\mathtt{D}) = \max(&\{\mathrm{rank}(\mathtt{f_i}) \mid i \in [n]\} \cup \\ &\{\mathrm{rank}(\mathtt{f_i}) + \mathrm{rank}(\mathtt{f_j}) \mid i, j \in [n],\ \mathtt{f_i} \lhd \mathtt{f_j^d} \text{ is in } \mathtt{D}\} \cup \\ &\{2\,\mathrm{rank}(\mathtt{f_i}) + 1 \mid i \in [n],\ \mathtt{f_i} \lhd \mathtt{g_j} \text{ is in } \mathtt{D} \text{ for some } j \in [m]\})\end{aligned}$$

Note that if $\mathtt{f_i} \lhd \mathtt{f_j}$ and $i \neq j$, then $\mathrm{rank}(f_i) < \mathrm{rank}(f_j)$. Indeed, by appending $\mathtt{f_j}$ to a longest path to $\mathtt{f_i}$ we get a longer path to $\mathtt{f_j}$ (note that there are no repeated symbols in the new path since $\mathtt{D}$ does not contain cycles of length greater than one).

2.3.1 $\mathcal{X}$

$$\mathcal{X} = \{(h) \in \mathcal{I}_1 \mid h \lhd h \lhd h^d \lhd h^d\}$$

Lemma 2.17. *Assume that*

- $m = 0$, *and*
- *there exists* $i \in [n]$ *such that* $\mathtt{f_i}$ *is monotone.*

Then $\mathrm{Clo}(\mathtt{D})$ *is equivalent to* $\mathcal{X}$.

Proof. We define a mapping $\xi : \mathrm{Clo}(\mathtt{D}) \to \mathcal{X}$ as follows:

$$\xi((f_1, f_2, \ldots, f_i, \ldots, f_n)) = f_i \wedge f_i^d$$

This mapping preserves minors due to Lemma 2.12. Furthermore, we can observe that $(f_i \wedge f_i^d)^d = f_i^d \vee f_i$ due to Lemma 2.4(ii) and $f_i \wedge f_i^d \lhd f_i \wedge f_i^d \lhd f_i \vee f_i^d$ due to the last inequality in Lemma 2.5(v) and transitivity in Lemma 2.5(ii).

Therefore, we have shown that the following chain of inequalities holds:

$$f_i \wedge f_i^d \lhd f_i \wedge f_i^d \lhd (f_i \wedge f_i^d)^d,$$

which proves that $(f_i \wedge f_i^d) \in \mathcal{X}$. Thus, we can conclude that ξ is well-defined.

In the other direction we define a mapping $\zeta : \mathcal{X} \to \mathrm{Clo}(\mathtt{D})$ as follows

$$\zeta((h)) = (f_1, \ldots, f_n) = (\underbrace{h, \ldots, h}_{n \text{ times}})$$

This mapping preserves minors due to Lemma 2.12. Moreover, ζ makes sense: if $\mathtt{f_i} \lhd \mathtt{f_j}$ is in $\mathtt{D}$, we need to verify that $f_i \lhd f_j$, which holds true because h is monotone. Similarly, if $\mathtt{f_i} \lhd \mathtt{f_j^d}$ is in $\mathtt{D}$, we need to confirm that $f_i \lhd f_j^d$, which is also true because $h \lhd h^d$. □

2.3.2 $\mathcal{Y}$

$$\mathcal{Y} = \{(h_1, h_2) \in \mathcal{I}_2 \mid h_1 \lhd h_1 \lhd h_2 = h_2^d \lhd h_1^d \lhd h_1^d\}$$

Lemma 2.18. *Assume that*

- $m > 0$,
- *there exists* $i \in [n]$ *such that* $\mathtt{f_i}$ *is monotone, and*
- *there is no* $j \in [m]$ *such that* $\mathtt{g_j}$ *is monotone.*

Then $\mathrm{Clo}(\mathtt{D})$ *is equivalent to* $\mathcal{Y}$.

Proof. In this case we define minion homomorphism $\xi : \text{Clo}(\mathtt{D}) \to \mathcal{Y}$ as follows.

$$\xi((f_1, f_2, \dots, f_n, g_1, \dots, g_m)) = (f_i \wedge f_i^d, (g_1 \vee (f_i \wedge f_i^d)) \wedge (f_i \vee f_i^d))$$

Here, due to Lemma 2.6(i) and (iii) as $g_1 = g_1^d$, $(f_i \wedge f_i^d)^d = f_i \vee f_i^d$ and f_i is monotone, then $(g_1 \vee (f_i \wedge f_i^d)) \wedge (f_i \vee f_i^d)) = ((g_1 \vee (f_i \wedge f_i^d)) \wedge (f_i \vee f_i^d))^d$ and $(f_i \wedge f_i^d) \lhd (g_1 \vee (f_i \wedge f_i^d)) \wedge (f_i \vee f_i^d))$.

Furthermore, $f_i \wedge f_i^d \lhd f_i \wedge f_i^d$ due to Lemma 2.5(v).

Thus, we have shown that ξ preserves the relations in $\mathcal{Y}$, which completes the proof that the map ξ is well-defined. The fact that the map is a minion homomorphism is shown analogously as in the previous case.

In the other direction, minion homomorphism $\zeta : \mathcal{Y} \to \text{Clo}(\mathtt{D})$ is defined as follows.

$$\begin{aligned}\xi((h_1, h_2)) &= (f_1, f_2, \dots, f_n, g_1, \dots, g_m) \\ &= (\underbrace{h_1, \dots, h_1}_{n \text{ times}}, \underbrace{h_2, \dots, h_2}_{m \text{ times}})\end{aligned}$$

This map is well-defined:

- If $\mathtt{f_i} \lhd \mathtt{f_j}$ is in $\mathtt{D}$ for $i, j \in [n]$, then we need to check that $f_i \lhd f_j$, which is true since $f_i = h_1 \lhd h_1 = f_j$.
- If $\mathtt{f_i} \lhd \mathtt{f_j^d}$ is in $\mathtt{D}$ for $i, j \in [n]$, then $h_1 \lhd h_1^d$.
- If $\mathtt{f_i} \lhd \mathtt{g_j}$ is in $\mathtt{D}$ for $i \in [n]$ and $j \in [m]$, then $h_1 \lhd h_2$.

□

2.3.3 $\mathcal{W}$

$$\mathcal{W} = \{(h) \in \mathcal{I}_1 \mid h \lhd h = h^d \lhd h^d\}$$

Lemma 2.19. *Assume that*

- $m > 0$, *and*
- *there exists* $j \in [m]$ *such that* $\mathtt{g_j}$ *is monotone.*

Then $\text{Clo}(\mathtt{D})$ *is equivalent to* $\mathcal{W}$.

Proof. We define a mapping $\xi : \text{Clo}(\mathtt{D}) \to \mathcal{W}$ by

$$\xi((f_1, f_2, \dots, f_n, g_1, \dots, g_m)) = (g_j).$$

This mapping preserves minors as in previous case and makes sense since $(g_j) \in \mathcal{W}$.

In the other direction, we define a mapping $\zeta : \mathcal{W} \to \text{Clo}(\mathtt{D})$ by

$$\zeta((h)) = (\underbrace{h, \dots, h}_{n+m \text{ times}})$$

This mapping makes sense since h satisfies all the relations in $\mathtt{D}$, and therefore $(\underbrace{h, \dots, h}_{n+m \text{ times}}) \in \text{Clo}(\mathtt{D})$. □

In all the previous three cases description D contained a monotone function symbol and we considered all the cases which could happen. Now in all the following cases we assume that there are no monotone functions in the description.

2.3.4 $\mathcal{A}_k$

$$\mathcal{A}_k = \{(h_1, h_1, \ldots, h_k) \in \mathcal{I}_k \mid h_1 \lhd h_2 \lhd \cdots \lhd h_k \leq h_k^d \lhd \cdots \lhd h_2^d \lhd h_1^d\}$$

Lemma 2.20. *Assume that*

- $m = 0$*,*
- $\mathtt{D}$ *is monotone-free, and*
- $\mathrm{chr}(\mathtt{D})$ *is odd.*

Then $\mathrm{Clo}(\mathtt{D})$ *is equivalent to* $\mathcal{A}_k$ *where* $k = (\mathrm{chr}(\mathtt{D}) + 1)/2$*.*

Proof. We have $\mathrm{chr}(\mathtt{D}) = \mathrm{rank}(\mathtt{f_i})$ for some $i \in [n]$ or $\mathrm{chr}(\mathtt{D}) = \mathrm{rank}(\mathtt{f_i}) + \mathrm{rank}(\mathtt{f_j})$ for some $i, j \in [n]$ such that $\mathtt{f_i} \lhd \mathtt{f_j^d}$ is in $\mathtt{D}$.

Consider first the second case. Assume that $\mathrm{rank}(\mathtt{f_i}) \geq \mathrm{rank}(\mathtt{f_j})$ (the proof for the converse inequality is completely analogous). Let $\mathtt{f_{i_1}} \lhd \mathtt{f_{i_2}} \lhd \cdots \lhd \mathtt{f_{i_r}}$ be a path to $\mathtt{f_i}$, where $i_1, i_2, \ldots, i_r = i \in [n]$, $r = \mathrm{rank}(\mathtt{f_i})$. Similarly, let $\mathtt{f_{j_1}} \lhd \mathtt{f_{j_2}} \lhd \ldots \mathtt{f_{j_s}}$ be a path to $\mathtt{f_j}$, where $j_1, j_2, \ldots, j_s = j \in [n]$, $s = \mathrm{rank}(\mathtt{f_j})$. Note that $k = (r + s + 1)/2$. We define $\xi : \mathrm{Clo}(\mathtt{D}) \to \mathcal{A}_k$ in the following way:

$$\begin{aligned}\xi((f_1, f_2, \ldots, f_n)) &= (h_1, h_2, \ldots, h_k)\\ &= (\underbrace{f_{i_1} \wedge f_{j_1},\ f_{i_2} \wedge f_{j_2},\ \ldots,\ f_{i_s} \wedge f_{j_s},}_{s \text{ operations}}\\ &\qquad \underbrace{f_{i_{s+1}} \wedge f_{i_r}^d,\ f_{i_{s+2}} \wedge f_{i_{r-1}}^d,\ \ldots,\ f_{i_k} \wedge f_{i_k}^d)}_{k-s \text{ operations}}\end{aligned}$$

Note that $h_k \leq h_k^d$. Indeed, $h_k = f_{i_k} \wedge f_{i_k}^d \leq f_{i_k} \vee f_{i_k}^d = (f_{i_k}^d \wedge f_{i_k})^d = h_k^d$, where we applied Lemma 2.4 (i) and (ii).

We have

$$f_{i_1} \lhd f_{i_2} \lhd \cdots \lhd f_{i_s} \lhd f_{i_{s+1}} \lhd f_{i_{s+2}} \lhd \cdots \lhd f_{i_k}$$

and

$$f_{j_1} \lhd f_{j_2} \lhd \cdots \lhd f_{j_s} \lhd f_{i_r}^d \lhd f_{i_{r-1}}^d \lhd \cdots \lhd f_{i_k}^d,$$

where we applied Lemma 2.5(i). Applying 2.5(iii) we obtain $h_1 \lhd h_2 \lhd \cdots \lhd h_k$. We have shown that ξ is well-defined and is thus a minion homomorphism, which finishes the second case.

The first case is simpler and we only give the definition of ξ. Let $\mathtt{f_{i_1}} \lhd \mathtt{f_{i_2}} \lhd \ldots \lhd \mathtt{f_{i_r}}$ be a path to $\mathtt{f_i}$, where $i_1, i_2, \ldots, i_r = i \in [n]$, $r = \mathrm{rank}(\mathtt{f_i})$. Note that $k = (r + 1)/2$. We define $\xi : \mathrm{Clo}(\mathtt{D}) \to \mathcal{A}_k$ in the following way:

$$\xi((f_1, f_2, \ldots, f_n)) = (f_{i_1} \wedge f_{i_r}^d,\ f_{i_2} \wedge f_{i_{r-1}}^d,\ \ldots,\ f_{i_k} \wedge f_{i_k}^d)$$

It remains to construct a minion homomorphism $\zeta : \mathcal{A}_k \to \mathrm{Clo}(\mathtt{D})$. For $(h_1, h_2, \ldots, h_k) \in \mathcal{A}_k$ we define

$$(h'_1, h'_2, \ldots, h'_{2k-1}) = (h_1, h_2, \ldots, h_k, h_{k-1}^d, h_{k-2}^d, \ldots, h_1^d)$$

and define ζ in the following way:

$$\zeta((h_1, h_2, \ldots, h_k)) = (h'_{\mathrm{rank}(\mathtt{f}_1)}, h'_{\mathrm{rank}(\mathtt{f}_2)}, \ldots, h'_{\mathrm{rank}(\mathtt{f}_n)})$$

Note that $\mathrm{rank}(\mathtt{f_i}) \leq 2k-1$ (since $2k-1 = \mathrm{chr}(\mathtt{D}) \geq \mathrm{rank}(\mathtt{f_i})$), so the indices of h' are in $[2k-1]$. Also note that $h'_1 \lhd h'_2 \lhd \cdots \lhd h'_{2k-1}$ by Lemma 2.5(ii) and (i) and that $h'_r \lhd (h'_s)^d$ whenever $r+s \leq 2k-1$ by the same lemma.

The map ζ is well-defined:

- If $\mathtt{f_i} \lhd \mathtt{f_j}$ is in $\mathtt{D}$ for $i, j \in [n]$, then $\mathrm{rank}(\mathtt{f_i}) < \mathrm{rank}(\mathtt{f_j})$, therefore $h'_{\mathrm{rank}(\mathtt{f_i})} \lhd h'_{\mathrm{rank}(\mathtt{f_j})}$.
- If $\mathtt{f_i} \lhd \mathtt{f_j^d}$ is in $\mathtt{D}$ for $i, j \in [n]$, then $\mathrm{rank}(\mathtt{f_i}) + \mathrm{rank}(\mathtt{f_j}) \leq \mathrm{chr}(\mathtt{D}) = 2k-1$, therefore $h'_{\mathrm{rank}(\mathtt{f_i})} \lhd (h'_{\mathrm{rank}(\mathtt{f_j})})^d$.

□

2.3.5 $\mathcal{B}_k$

$$\mathcal{B}_k = \{(h_1, h_2, \ldots, h_k) \in \mathcal{I}_k \mid h_1 \lhd h_2 \lhd \cdots \lhd h_k \lhd h_k^d \lhd \cdots \lhd h_2^d \lhd h_1^d\}$$

Lemma 2.21. *Assume that*

- $m = 0$*,*
- $\mathtt{D}$ *is monotone-free, and*
- $\mathrm{chr}(\mathtt{D})$ *is even.*

Then $\mathrm{Clo}(\mathtt{D})$ *is equivalent to* $\mathcal{B}_k$ *where* $k = \mathrm{chr}(\mathtt{D})/2$.

Proof. The proof is similar to the previous one. We have $\mathrm{chr}(\mathtt{D}) = \mathrm{rank}(\mathtt{f_i})$ for some $i \in [n]$ or $\mathrm{chr}(\mathtt{D}) = \mathrm{rank}(\mathtt{f_i}) + \mathrm{rank}(\mathtt{f_j})$ for some $i, j \in [n]$ such that $\mathtt{f_i} \lhd \mathtt{f_j^d}$ is in $\mathtt{D}$.

In the second case, assume that $\mathrm{rank}(\mathtt{f_i}) \geq \mathrm{rank}(\mathtt{f_j})$. Let $\mathtt{f}_{\mathtt{i}_1} \lhd \mathtt{f}_{\mathtt{i}_2} \lhd \cdots \lhd \mathtt{f}_{\mathtt{i}_r}$ be a path to $\mathtt{f_i}$, where $i_1, i_2, \ldots, i_r = i \in [n]$, $r = \mathrm{rank}(\mathtt{f_i})$. Let $\mathtt{f}_{\mathtt{j}_1} \lhd \mathtt{f}_{\mathtt{j}_2} \lhd \ldots \mathtt{f}_{\mathtt{j}_s}$ be a path to $\mathtt{f_j}$, where $j_1, j_2, \ldots, j_s = j \in [n]$, $s = \mathrm{rank}(\mathtt{f_j})$. Note that $k = (r+s)/2$. We define $\xi : \mathrm{Clo}(\mathtt{D}) \to \mathcal{B}_k$ in the following way:

$$\begin{aligned}\xi((f_1, f_2, \ldots, f_n)) &= (h_1, h_2, \ldots, h_k) \\ &= (\underbrace{f_{i_1} \wedge f_{j_1},\ f_{i_2} \wedge f_{j_2},\ \ldots,\ f_{i_s} \wedge f_{j_s}}_{s \text{ operations}}, \\ &\quad \underbrace{f_{i_{s+1}} \wedge f^d_{i_r},\ f_{i_{s+2}} \wedge f^d_{i_{r-1}},\ \ldots,\ f_{i_k} \wedge f^d_{i_{k+1}})}_{k-s \text{ operations}}\end{aligned}$$

Note that $h_k \lhd h_k^d$. Indeed, if $r > s$, then $h_k = f_{i_k} \wedge f^d_{i_{k+1}} \lhd f_{i_{k+1}} \wedge f^d_{i_k} \leq f_{i_{k+1}} \vee f^d_{i_k} = (f^d_{i_{k+1}} \wedge f_{i_k})^d = h_k^d$; if $r = s = k$, then $h_k = f_{i_k} \wedge f_{j_k} \lhd f^d_{j_k} \wedge f^d_{i_k} \leq f^d_{j_k} \vee f^d_{i_k} = (f_{j_k} \wedge f_{i_k})^d = h_k^d$. The rest of the argument for this case is as in the previous lemma.

The first case is also similar. Let $\mathtt{f}_{\mathtt{i}_1} \lhd \mathtt{f}_{\mathtt{i}_2} \lhd \cdots \lhd \mathtt{f}_{\mathtt{i}_r}$ be a path to $\mathtt{f}_\mathtt{i}$, where $i_1, i_2, \ldots, i_r = i \in [n]$, $r = \mathrm{rank}(\mathtt{f}_\mathtt{i})$. Note that $k = r/2$. We define $\xi : \mathrm{Clo}(\mathtt{D}) \to \mathcal{A}_k$ in the following way:

$$\xi((f_1, f_2, \ldots, f_n)) = (f_{i_1} \wedge f_{i_r}^d,\ f_{i_2} \wedge f_{i_{r-1}}^d,\ \ldots,\ f_{i_k} \wedge f_{i_{k+1}}^d)$$

In the other direction, for $(h_1, h_2, \ldots, h_k) \in \mathcal{B}_k$ we define

$$(h'_1, h'_2, \ldots, h'_{2k}) = (h_1, h_2, \ldots, h_k, h_k^d, h_{k-1}^d, \ldots, h_1^d)$$

and define ζ in the following way:

$$\zeta((h_1, h_2, \ldots, h_k)) = (h'_{\mathrm{rank}(\mathtt{f}_1)}, h'_{\mathrm{rank}(\mathtt{f}_2)}, \ldots, h'_{\mathrm{rank}(\mathtt{f}_\mathtt{n})})$$

Note that $\mathrm{rank}(\mathtt{f}_\mathtt{i}) \leq 2k$ (since $2k = \mathrm{chr}(\mathtt{D}) \geq \mathrm{rank}(\mathtt{f}_\mathtt{i})$), so the indices of h' are in $[2k]$. Also note that $h'_1 \lhd h'_2 \lhd \cdots \lhd h'_{2k}$ and that $h'_r \lhd (h'_s)^d$ whenever $r + s \leq 2k$. This implies that ζ is well-defined as in the previous proof. □

2.3.6 $\mathcal{C}_k$

$$\mathcal{C}_k = \{(h_1, h_2, \ldots, h_k) \in \mathcal{I}_k \mid h_1 \lhd h_2 \lhd \cdots \lhd h_k = h_k^d \lhd \cdots \lhd h_2^d \lhd h_1^d\}$$

Lemma 2.22. *Assume that*

- $m > 0$,
- $\mathtt{D}$ *is monotone-free, and*
- $\mathrm{chr}(\mathtt{D})$ *is odd.*

Then $\mathrm{Clo}(\mathtt{D})$ *is equivalent to* $\mathcal{C}_k$ *where* $k = (\mathrm{chr}(\mathtt{D}) + 1)/2$.

Proof. We have $\mathrm{chr}(\mathtt{D}) = \mathrm{rank}(\mathtt{f}_\mathtt{i})$ for some $i \in [n]$, or $\mathrm{chr}(\mathtt{D}) = \mathrm{rank}(\mathtt{f}_\mathtt{i}) + \mathrm{rank}(\mathtt{f}_\mathtt{j})$ for some $i, j \in [n]$ such that $\mathtt{f}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}^\mathtt{d}$ is in $\mathtt{D}$, or $\mathrm{chr}(\mathtt{D}) = 2\,\mathrm{rank}(\mathtt{f}_\mathtt{i}) + 1$ for some $i \in [n]$, $j \in [m]$ such that $\mathtt{f}_\mathtt{i} \lhd \mathtt{g}_\mathtt{j}$ is in $\mathtt{D}$.

As usual, consider first the second case. Assume that $\mathrm{rank}(\mathtt{f}_\mathtt{i}) \geq \mathrm{rank}(\mathtt{f}_\mathtt{j})$. Let $\mathtt{f}_{\mathtt{i}_1} \lhd \mathtt{f}_{\mathtt{i}_2} \lhd \cdots \lhd \mathtt{f}_{\mathtt{i}_r}$ be a path to $\mathtt{f}_\mathtt{i}$, where $i_1, i_2, \ldots, i_r = i \in [n]$, $r = \mathrm{rank}(\mathtt{f}_\mathtt{i})$. Similarly, let $\mathtt{f}_{\mathtt{j}_1} \lhd \mathtt{f}_{\mathtt{j}_2} \lhd \ldots \mathtt{f}_{\mathtt{j}_s}$ be a path to $\mathtt{f}_\mathtt{j}$, where $j_1, j_2, \ldots, j_s = j \in [n]$, $s = \mathrm{rank}(\mathtt{f}_\mathtt{j})$. Note that $k = (r + s + 1)/2$. We define $\xi : \mathrm{Clo}(\mathtt{D}) \to \mathcal{C}_k$ in the following way:

$$\begin{aligned}\xi((f_1, f_2, \ldots, f_n)) &= (h_1, h_2, \ldots, h_k)\\ &= (\underbrace{f_{i_1} \wedge f_{j_1},\ f_{i_2} \wedge f_{j_2},\ \ldots,\ f_{i_s} \wedge f_{j_s}}_{s \text{ operations}},\\ &\quad \underbrace{f_{i_{s+1}} \wedge f_{i_r}^d,\ f_{i_{s+2}} \wedge f_{i_{r-1}}^d,\ \ldots,\ f_{i_{k-1}} \wedge f_{i_{k+1}}^d}_{k-s-1 \text{ operations}},\\ &\quad (g_1 \vee (f_{i_k} \wedge f_{i_k}^d)) \wedge (f_{i_k} \wedge f_{i_k}^d)^d)\end{aligned}$$

Denote $h'_k = f_{i_k} \wedge f_{i'_k}$. Similarly as in the previous two lemmas we obtain $h_1 \lhd h_2 \lhd h_{k-1} \lhd h'_k$. Note that $h'_k \leq (h'_k)^d$ and $h_k = (g_1 \vee h'_k) \wedge (h'_k)^d$. By Lemma 2.6(i) we have $h_k = h_k^d$. From Lemma 2.6(ii) we obtain $h'_k \leq h_k$. Combining this with $h_{k-1} \lhd h'_k$ we get $h_{k-1} \lhd h_k$. This finishes the second case.

As for the first case, let $\mathtt{f}_{\mathtt{i}_1} \lhd \mathtt{f}_{\mathtt{i}_2} \lhd \cdots \lhd \mathtt{f}_{\mathtt{i}_r}$ be a path to $\mathtt{f}_\mathtt{i}$, where $i_1, i_2, \ldots, i_r = i \in [n]$, $r = \mathrm{rank}(\mathtt{f}_\mathtt{i})$. Note that $k = (r+1)/2$. We define $\xi : \mathrm{Clo}(\mathtt{D}) \to \mathcal{C}_k$ in the following way:

$$\xi((f_1, f_2, \ldots, f_n)) = (f_{i_1} \wedge f_{i_r}^d,\ f_{i_2} \wedge f_{i_{r-1}}^d,\ \ldots,\ f_{i_{k-1}} \wedge f_{i_{k+1}}^d,\\ (g_1 \vee (f_{i_k} \wedge f_{i_k}^d)) \wedge (f_{i_k} \wedge f_{i_k}^d)^d)$$

The third case is the simplest. Let $\mathtt{f}_{\mathtt{i}_1} \lhd \mathtt{f}_{\mathtt{i}_2} \lhd \cdots \lhd \mathtt{f}_{\mathtt{i}_r}$ be a path to $\mathtt{f}_\mathtt{i}$, where $i_1, i_2, \ldots, i_r = i \in [n]$, $r = \mathrm{rank}(f_i)$. Recall that $\mathtt{f}_\mathtt{i} \lhd \mathtt{g}_\mathtt{j}$ is in $\mathtt{D}$ and note that $k = r+1$. We define $\xi : \mathrm{Clo}(\mathtt{D}) \to \mathcal{C}_k$ in the following way:

$$\xi((f_1, f_2, \ldots, f_n)) = (f_{i_1}, f_{i_2}, \ldots, f_{i_r}, g_j)$$

It remains to construct a minion homomorphism $\zeta : \mathcal{C}_k \to \mathrm{Clo}(\mathtt{D})$. For $(h_1, h_2, \ldots, h_k) \in \mathcal{C}_k$ we define

$$(h'_1, h'_2, \ldots, h'_{2k-1}) = (h_1, h_2, \ldots, h_k, h_{k-1}^d, h_{k-2}^d, \ldots, h_1^d)$$

and define ζ in the following way:

$$\zeta((h_1, h_2, \ldots, h_k)) = (\underbrace{h'_{\mathrm{rank}(\mathtt{f}_1)}, h'_{\mathrm{rank}(\mathtt{f}_2)}, \ldots, h'_{\mathrm{rank}(\mathtt{f}_\mathtt{n})}}_{n \text{ operations}}, \underbrace{h_k, h_k, \ldots, h_k}_{m \text{ operations}})$$

Note

- that $\mathrm{rank}(\mathtt{f}_\mathtt{i}) \leq 2k-1$ (since $2k-1 = \mathrm{chr}(\mathtt{D}) \geq \mathrm{rank}(\mathtt{f}_\mathtt{i})$), so the indices of h' are in $[2k-1]$,
- that $h'_1 \lhd h'_2 \lhd \cdots \lhd h'_{2k-1}$,
- that $h'_r \lhd (h'_s)^d$ whenever $r + s \leq 2k-1$, and
- that $h'_r \lhd h_k$ whenever $r \leq k-1$.

The map ζ is therefore well-defined:

- If $\mathtt{f}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}$ is in $\mathtt{D}$ for $i, j \in [n]$, then $\mathrm{rank}(\mathtt{f}_\mathtt{i}) < \mathrm{rank}(\mathtt{f}_\mathtt{j})$, therefore $h'_{\mathrm{rank}(\mathtt{f}_\mathtt{i})} \lhd h'_{\mathrm{rank}(\mathtt{f}_\mathtt{j})}$.
- If $\mathtt{f}_\mathtt{i} \lhd \mathtt{f}_\mathtt{j}^\mathtt{d}$ is in $\mathtt{D}$ for $i, j \in [n]$, then $\mathrm{rank}(\mathtt{f}_\mathtt{i}) + \mathrm{rank}(\mathtt{f}_\mathtt{j}) \leq \mathrm{chr}(\mathtt{D}) = 2k-1$, therefore $h'_{\mathrm{rank}(\mathtt{f}_\mathtt{i})} \lhd (h'_{\mathrm{rank}(\mathtt{f}_\mathtt{j})})^d$.
- If $\mathtt{f}_\mathtt{i} \lhd \mathtt{g}_\mathtt{j}^\mathtt{d}$ is in $\mathtt{D}$ for $i \in [n]$ and $j \in [m]$, then $\mathrm{rank}(\mathtt{f}_\mathtt{i}) \leq (\mathrm{chr}(\mathtt{D}) - 1)/2 = k-1$, therefore $h'_{\mathrm{rank}((f_i))} \lhd h_k$.

□

2.3.7 $\mathcal{D}_k$

$$\begin{aligned}\mathcal{D}_k =&\{(h_1, h_2, \ldots, h_k, h_{k+1}) \in \mathcal{I}_{k+1} \mid \\ &h_1 \lhd h_2 \lhd \cdots \lhd h_k \leq h_{k+1} = h_{k+1} \leq h_k^d \lhd \cdots \lhd h_2^d, h_k \lhd h_k^d\}\end{aligned}$$

Lemma 2.23. *Assume that*

- $m > 0$,
- `D` *is monotone-free, and*
- $\mathrm{chr}(\mathtt{D})$ *is even.*

Then $\mathrm{Clo}(\mathtt{D})$ *is equivalent to* $\mathcal{D}_k$ *where* $k = \mathrm{chr}(\mathtt{D})/2$.

Proof. We have $\mathrm{chr}(\mathtt{D}) = \mathrm{rank}(\mathtt{f_i})$ for some $i \in [n]$ or $\mathrm{chr}(\mathtt{D}) = \mathrm{rank}(\mathtt{f_i}) + \mathrm{rank}(\mathtt{f_j})$ for some $i, j \in [n]$ such that $\mathtt{f_i} \lhd \mathtt{f_j^d}$ is in `D` (we do not have the third case here unlike for $\mathcal{C}_k$).

In the second case we define $\xi : \mathrm{Clo}(\mathtt{D}) \to \mathcal{D}_k$ with the usual notation (noting $k = (r+s)/2$) as follows.

$$\begin{aligned}\xi((f_1, f_2, \ldots, f_n)) &= (h_1, h_2, \ldots, h_k, h_{k+1}) \\ &= (\underbrace{f_{i_1} \wedge f_{j_1},\ f_{i_2} \wedge f_{j_2},\ \ldots,\ f_{i_s} \wedge f_{j_s}}_{s \text{ operations}}, \\ &\quad \underbrace{f_{i_{s+1}} \wedge f_{i_r}^d,\ f_{i_{s+2}} \wedge f_{i_{r-1}}^d,\ \ldots,\ f_{i_k} \wedge f_{i_{k+1}}^d}_{k-s \text{ operations}}, \\ &\quad (g_1 \vee (f_{i_k} \wedge f_{i_{k+1}}^d)) \wedge (f_{i_k} \wedge f_{i_{k+1}}^d)^d))\end{aligned}$$

As in the proof for $\mathcal{B}_k$ we get $h_k \lhd h_k^d$. As in the proof for $\mathcal{C}_k$ we get $h_1 \lhd \cdots \lhd h_k \leq h_{k+1}$ and $h_{k+1} = h_{k+1}^d$.

The definition of ξ for the first case is as follows.

$$\begin{aligned}\xi((f_1, f_2, \ldots, f_n)) =&(f_{i_1} \wedge f_{i_r}^d,\ f_{i_2} \wedge f_{i_{r-1}}^d,\ \ldots,\ f_{i_k} \wedge f_{i_{k+1}}^d, \\ &(g_1 \vee (f_{i_k} \wedge f_{i_{k+1}}^d)) \wedge (f_{i_k} \wedge f_{i_{k+1}}^d)^d)\end{aligned}$$

The construction of $\zeta : \mathcal{D}_k \to \mathrm{Clo}(\mathtt{D})$ is again analogous to the proofs for $\mathcal{B}_k$ and $\mathcal{C}_k$. For $(h_1, h_2, \ldots, h_k) \in \mathcal{C}_k$ we define

$$(h'_1, h'_2, \ldots, h'_{2k}) = (h_1, h_2, \ldots, h_k, h_k^d, h_{k-1}^d, \ldots, h_1^d)$$

and define ζ in the following way:

$$\zeta((h_1, h_2, \ldots, h_{k+1})) = (\underbrace{h'_{\mathrm{rank}(\mathtt{f_1})}, h'_{\mathrm{rank}(\mathtt{f_2})}, \ldots, h'_{\mathrm{rank}(\mathtt{f_n})}}_{n \text{ operations}}, \underbrace{h_{k+1}, h_{k+1}, \ldots, h_{k+1}}_{m \text{ operations}})$$

□

2.3.8 Putting it together

Theorem 2.24. *Let θ be a set of at most binary l-sorted relations on a l-sorted $\boldsymbol{A} = (\{0,1\},\dots,\{0,1\})$ for some positive integer l. The clone* $\mathrm{Pol}(\theta)$ *is equivalent to at least one of the multi-sorted minions $\mathcal{A}_k$, $\mathcal{B}_k$, $\mathcal{C}_k$, $\mathcal{D}_k$, $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{W}$, $\mathcal{T}$, $k \in \mathbb{N}$.*

Proof. By Lemma 2.10, $\mathrm{Pol}(\theta)$ is equivalent to $\mathcal{T}$ or is equivalent to $\mathrm{Clo}(\mathtt{D})$ for a description $\mathtt{D}$. In the latter case, by Theorem 2.15 we can assume that $\mathtt{D}$ is in a reduced form. As in the beginning of the section, let $\mathtt{D}$ be in a reduced form over $\mathtt{f}_1,\dots,\mathtt{f}_\mathtt{n},\mathtt{g}_1,\dots,\mathtt{g}_\mathtt{m}$. The following case analysis shows that our lemmas covered all the cases.

- $\mathtt{D}$ contains a monotone symbol.
  - $m = 0$. Then $\mathrm{Clo}(\mathtt{D})$ is equivalent to $\mathcal{X}$ by Lemma 2.17.
  - $m > 0$.
    - some symbol $\mathtt{g}_j$ is monotone. Then $\mathrm{Clo}(\mathtt{D})$ is equivalent to $\mathcal{W}$ by Lemma 2.19.
    - no symbol $\mathtt{g}_j$ is monotone. Then $\mathrm{Clo}(\mathtt{D})$ is equivalent to $\mathcal{Y}$ by Lemma 2.18.
- $\mathtt{D}$ is monotone-free.
  - $m = 0$.
    - $\mathrm{chr}(\mathtt{D})$ is odd. Then $\mathrm{Clo}(\mathtt{D})$ is equivalent to $\mathcal{A}_{(\mathrm{rank}(\mathtt{D})+1)/2}$ by Lemma 2.20.
    - $\mathrm{chr}(\mathtt{D})$ is even. Then $\mathrm{Clo}(\mathtt{D})$ is equivalent to $\mathcal{B}_{\mathrm{rank}(\mathtt{D})/2}$ by Lemma 2.21.
  - $m > 0$.
    - $\mathrm{chr}(\mathtt{D})$ is odd. Then $\mathrm{Clo}(\mathtt{D})$ is equivalent to $\mathcal{C}_{(\mathrm{rank}(\mathtt{D})+1)/2}$ by Lemma 2.22.
    - $\mathrm{chr}(\mathtt{D})$ is even. Then $\mathrm{Clo}(\mathtt{D})$ is equivalent to $\mathcal{D}_{\mathrm{rank}(\mathtt{D})/2}$ by Lemma 2.23.

□

2.4 Cores

In this section we show that all the multi-sorted minions from the last section are minion cores. That is, we prove the following theorem.

Theorem 2.25. *The multi-sorted minions $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{W}$, $\mathcal{A}_n$, $\mathcal{B}_n$, $\mathcal{C}_n$, and $\mathcal{D}_n$ are minion cores for every $n \in \mathbb{N}$.*

Recall that all these multi-sorted minions (apart from the trivial $\mathcal{T}$ which we do not need to consider) are on $(\boldsymbol{A},\boldsymbol{A})$, where $\boldsymbol{A} = (\{0,1\},\dots,\{0,1\})$.

The following lemma follows from the fact that the sorts are two-element sets. The second part shows that it is enough to show that the binary parts of minion homomorphisms $\mathcal{M} \to \mathcal{M}$ are bijections.

Lemma 2.26. *Let $\mathcal{M}$ be a multi-sorted minion on $(\boldsymbol{A}, \boldsymbol{A})$.*

(i) If $\xi, \nu : \mathcal{M} \to \mathcal{M}$ are minion homomorphisms such that $\xi^{(2)} = \nu^{(2)}$, then $\xi = \nu$.

(ii) If for every minion homomorphism $\mathcal{M} \to \mathcal{M}$ its binary part $\xi^{(2)} : \mathcal{M}^{(2)} \to \mathcal{M}^{(2)}$ is a bijection, then $\mathcal{M}$ is a minion core.

Proof. (i) Let $\boldsymbol{f} = (f_1, \dots, f_k) \in \mathcal{M}$, say m-ary, and denote $\xi(\boldsymbol{f}) = (g_1, \dots, g_k)$ and $\nu(\boldsymbol{f}) = (h_1, \dots, h_k)$. We need to verify that $g_i = h_i$ for every $i \in [k]$. To prove it we check that $g_i(\boldsymbol{a}) = h_i(\boldsymbol{a})$ for every $\boldsymbol{a} = (a_1, \dots, a_m) \in \{0,1\}^m$ as follows.

For any $i_1, \dots, i_m \in [2]$ we have

$$\begin{aligned}\xi(\boldsymbol{f}) \circ (\boldsymbol{\pi}^2_{i_1}, \dots, \boldsymbol{\pi}^2_{i_m}) &= \xi^{(2)}(\boldsymbol{f} \circ (\boldsymbol{\pi}^2_{i_1}, \dots, \boldsymbol{\pi}^2_{i_m}))\\ &= \nu^{(2)}(\boldsymbol{f} \circ (\boldsymbol{\pi}^2_{i_1}, \dots, \boldsymbol{\pi}^2_{i_m}))\\ &= \nu(\boldsymbol{f}) \circ (\boldsymbol{\pi}^2_{i_1}, \dots, \boldsymbol{\pi}^2_{i_m})\end{aligned}$$

By comparing the ith components of these tuples of Boolean operations we obtain

$$g_i \circ (\pi^2_{i_1}, \dots, \pi^2_{i_m}) = h_i \circ (\pi^2_{i_1}, \dots, \pi^2_{i_m}).$$

Now we have

$$\begin{aligned}g_i(\boldsymbol{a}) = g_i(a_1, \dots, a_m) &= g_i(\pi^2_{a_1+1}(0,1), \dots, \pi^2_{a_m+1}(0,1))\\ &= (g_i \circ (\pi^2_{a_1+1}, \dots, \pi^2_{a_m+1}))(0,1)\\ &= (h_i \circ (\pi^2_{a_1+1}, \dots, \pi^2_{a_m+1}))(0,1)\\ &= h_i(a_1, \dots, a_m) = h_i(\boldsymbol{a})\end{aligned}$$

(ii) We need to prove that every minion homomorphism $\xi : \mathcal{M} \to \mathcal{M}$ is a bijection. Consider a homomorphism ξ. By the assumption, the mapping $\xi^{(2)}$ is a bijection on $\mathcal{M}^{(2)}$. Therefore, there exists n such that

$$(\xi^{(2)})^n = \underbrace{\xi^{(2)} \circ \xi^{(2)} \circ \dots \circ \xi^{(2)}}_{n \text{ times}} = \mathrm{id}_{\mathcal{M}^{(2)}}$$

(one can take e.g. $n = |\mathcal{M}^{(2)}|!$). It follows from the first part of the lemma that $\mu = \xi^{n-1}$ is a both-sided inverse to ξ: Indeed, we have $(\mu \circ \xi)^{(2)} = \mu^{(2)} \circ \xi^{(2)} = (\xi^{(2)})^{n-1} \circ \xi^{(2)} = (\xi^{(2)})^n = \mathrm{id}^{(2)}_{\mathcal{M}}$, so $\mu \circ \xi = \mathrm{id}_{\mathcal{M}}$, and similarly $(\xi \circ \mu)^{(2)} = \mathrm{id}^{(2)}_{\mathcal{M}}$, so $\xi \circ \mu = \mathrm{id}_{\mathcal{M}}$.

□

Binary (multi-sorted) operations will therefore play an important role. To simplify notation, we denote

$$x = \pi^2_1, \quad y = \pi^2_2, \quad \boldsymbol{x} = \boldsymbol{\pi}^2_1 = (x, x, \dots, x), \quad \boldsymbol{y} = \boldsymbol{\pi}^2_2 = (y, y, \dots, y).$$

There are exactly four Boolean idempotent binary operations, namely

$$x, y, \wedge, \vee.$$

Observe that

$$\wedge \leq x, y \leq \vee, \quad \wedge \lhd\ x, y \lhd \vee, \quad x^d = y,\ y^d = x,\ \wedge^d = \vee, \vee^d = \wedge.$$

A multi-sorted idempotent Boolean operation is e.g. the 3-sorted operation $(y, \wedge, \vee)$.

Our multi-sorted minions are all idempotent. This has the following consequence.

Lemma 2.27. *Let $\mathcal{M}$ be a multi-sorted minion on $(\boldsymbol{A}, \boldsymbol{A})$ whose all members are idempotent and let $\xi : \mathcal{M} \to \mathcal{M}$ be a minion homomoprhism. Then $\mathcal{M}$ contains all the projections on $\boldsymbol{A}$ and for any $m \in \mathbb{N}$ and $i \in [m]$ we have*

$$\xi(\boldsymbol{\pi}_i^m) = \boldsymbol{\pi}_i^m.$$

In particular, $\xi(\boldsymbol{x}) = \boldsymbol{x}$ and $\xi(\boldsymbol{y}) = \boldsymbol{y}$.

Proof. Since $\mathcal{M}$ is nonempty, also $\mathcal{M}^{(1)}$ is nonempty (take a unary minor of any member of $\mathcal{M}$). The only idempotent unary multi-sorted operation on $\boldsymbol{A}$ is $\boldsymbol{\pi}_1^1$. One if its minors is $\boldsymbol{\pi}_1^1 \circ (\boldsymbol{\pi}_i^m) = \boldsymbol{\pi}_i^m$, so it is in $\mathcal{M}$ and we have

$$\xi(\boldsymbol{\pi}_i^m) = \xi(\boldsymbol{\pi}_1^1 \circ (\boldsymbol{\pi}_i^m)) = \xi(\boldsymbol{\pi}_1^1) \circ (\boldsymbol{\pi}_i^m) = \boldsymbol{\pi}_1^1 \circ (\boldsymbol{\pi}_i^m) = \boldsymbol{\pi}_i^m.$$

□

The two lemmata combined already give the result for $\mathcal{W}$, see Lemma 2.32.

For the remaining multi-sorted minions, we will use the fact that minion homomorphisms "preserve identities" in the following sense. Let $\mathcal{M}$ and ξ be as in the last lemma. If $\boldsymbol{t} \in \mathcal{M}^{(n)}$ and $\boldsymbol{s} \in \mathcal{M}^{(n')}$ satisfy

$$\boldsymbol{t} \circ (\boldsymbol{\pi}_{i_1}^m, \boldsymbol{\pi}_{i_2}^m, \dots, \boldsymbol{\pi}_{i_n}^m) = \boldsymbol{s} \circ (\boldsymbol{\pi}_{i'_1}^m, \boldsymbol{\pi}_{i'_2}^m, \dots, \boldsymbol{\pi}_{i'_n}^m),$$

then by applying ξ to both sides and using that ξ is a minion homomorphism, we get

$$\begin{aligned} \xi(\boldsymbol{t} \circ (\boldsymbol{\pi}_{i_1}^m, \boldsymbol{\pi}_{i_2}^m, \dots, \boldsymbol{\pi}_{i_n}^m)) &= \xi(\boldsymbol{s} \circ (\boldsymbol{\pi}_{i'_1}^m, \boldsymbol{\pi}_{i'_2}^m, \dots, \boldsymbol{\pi}_{i'_n}^m)) \\ \xi(\boldsymbol{t}) \circ (\boldsymbol{\pi}_{i_1}^m, \boldsymbol{\pi}_{i_2}^m, \dots, \boldsymbol{\pi}_{i_n}^m) &= \xi(\boldsymbol{s}) \circ (\boldsymbol{\pi}_{i'_1}^m, \boldsymbol{\pi}_{i'_2}^m, \dots, \boldsymbol{\pi}_{i'_n}^m). \end{aligned}$$

In this sense $\xi(\boldsymbol{s})$ and $\xi(\boldsymbol{t})$ satisfy "the same identity".

Because of the last lemma, ξ also preserves identities where one of the sides is just a projection:

$$\boldsymbol{t} \circ (\boldsymbol{\pi}_{i_1}^m, \boldsymbol{\pi}_{i_2}^m, \dots, \boldsymbol{\pi}_{i_n}^m) = \boldsymbol{\pi}_i^m \quad \Rightarrow \quad \xi(\boldsymbol{t}) \circ (\boldsymbol{\pi}_{i_1}^m, \boldsymbol{\pi}_{i_2}^m, \dots, \boldsymbol{\pi}_{i_n}^m) = \boldsymbol{\pi}_i^m.$$

We will mostly use identities where $m = 2$. To increase readability, we write identities in a simplified form: we use the convention $\boldsymbol{x} = \boldsymbol{\pi}_1^2$ etc. above, we use the plain font, and skip "$\circ$" and ",". For example, the identities

$$\boldsymbol{t} \circ (\boldsymbol{\pi}_1^2, \boldsymbol{\pi}_1^2, \boldsymbol{\pi}_1^2, \boldsymbol{\pi}_2^2, \boldsymbol{\pi}_2^2) = \boldsymbol{s} \circ (\boldsymbol{\pi}_1^2, \boldsymbol{\pi}_2^2, \boldsymbol{\pi}_2^2, \boldsymbol{\pi}_2^2, \boldsymbol{\pi}_2^2), \quad \boldsymbol{s} \circ (\boldsymbol{\pi}_1^2, \boldsymbol{\pi}_1^2, \boldsymbol{\pi}_2^2, \boldsymbol{\pi}_2^2, \boldsymbol{\pi}_2^2) = \boldsymbol{\pi}_2^2$$

will be written as

$$t(xxxyy) = s(xyyyy), \quad s(xxyyy) = y.$$

The only other identities that will be used are identities for 5-ary multi-sorted operations of the form

$$\boldsymbol{t} \circ (\boldsymbol{\pi}_1^5, \boldsymbol{\pi}_2^5, \boldsymbol{\pi}_3^5, \boldsymbol{\pi}_4^5, \boldsymbol{\pi}_5^5) = \boldsymbol{t} \circ (\boldsymbol{\pi}_{\sigma(1)}^5, \boldsymbol{\pi}_{\sigma(2)}^5, \boldsymbol{\pi}_{\sigma(3)}^5, \boldsymbol{\pi}_{\sigma(4)}^5, \boldsymbol{\pi}_{\sigma(5)}^5),$$

where $\sigma \in S_5$ is a permutation on [5]. We write them as

$$t(x_1, x_2, x_3, x_4, x_5) = t(x_{\sigma(1)}, x_{\sigma(2)}, x_{\sigma(3)}, x_{\sigma(4)}, x_{\sigma(5)}).$$

The common strategy for the proof that the considered minion $\mathcal{M}$ is a core is as follows.

- We consider a set of identities.
- We show that $\mathcal{M}$ has multi-sorted operations satisfying those identities and we prove that the identities (almost) uniquely determine the operations. Moreover, binary minors of these operations will contain (almost) all binary members of $\mathcal{M}$.
- Because minion homomorphisms preserves identities, each minion homomorphism $\xi : \mathcal{M} \to \mathcal{M}$ maps these operation (almost) to themselves. The last property from the previous item will guarantee that $\xi^{(2)}$ is a bijection.
- We conclude the proof by applying the second part of Lemma 2.26.

Remark 2.28. *We will often use 5-ary idempotent symmetric multi-sorted operations, i.e., idempotent multi-sorted operations* $\boldsymbol{t}$ *satisfying the identities*

$$t(x_1, x_2, x_3, x_4, x_5) = t(x_{\sigma(1)}, x_{\sigma(2)}, x_{\sigma(3)}, x_{\sigma(4)}, x_{\sigma(5)}) \quad \forall \sigma \in S_5$$

Such a $\boldsymbol{t} = (h_1, \ldots, h_n)$ *will be represented by a table like the following.*

$\boldsymbol{t}$	h_1	h_2		h_{n-1}	h_n
10000	*0*	*0*	...	*0*	*?*
11000	*0*	*0*	...	*0*	*0*
11100	*1*	*1*	...	*1*	*1*
11110	*?*	*1*	...	*1*	*1*

The four rows correspond to the 5-tuples $(1,0,0,0,0)$, $(1,1,0,0,0)$, $(1,1,1,0,0)$, $(1,1,1,1,0)$ *and columns correspond to the components of* $\boldsymbol{t}$. *The entries are values of the operations. For example, the one in position* $(3,2)$ *(third row, second column) indicates that* $h_2(1,1,1,0,0) = 1$. *Note that such a table uniquely determines* $\boldsymbol{t}$ *because of the symmetry and idempotency. The question mark indicates that the value is unknown at that point.*

We utilize the following observations, where we use $[i,j]$ *to denote the value at position* (i,j).

- $h_{j-1} \leq h_j$ *iff* $[i, j-1] \leq [i, j]$ *for every* $i \in [4]$. *In particular, if* $[i,j] = 0$, *then* $[i, j-1] = 0$, *and if* $[i, j-1] = 1$, *then* $[i,j] = 1$.
- $h_{j-1} = h_j$ *iff the columns* $j-1$ *and* j *are identical.*

- $h_{j-1} \lhd h_j$ *iff* $[i, j-1] \leq [i', j]$ *for every* $i, i' \in [4]$ *such that* $i' \geq i$. *E.g., if* $[i, j] = 0$, *then* $[i, j-1] = [i-1, j-1] = \ldots, = 0$.

- $h_j = h_j^d$ *iff* $[i, j] = 1 - [5-i, j]$ *for every* $i \in [4]$.

Remark 2.29. *In some cases, we utilize ternary idempotent multi-sorted operations instead of 5-ary symmetric ones. multi-sorted ternary operation* $\boldsymbol{t} = (h_1, \ldots, h_n)$ *will be represented as follows.*

110

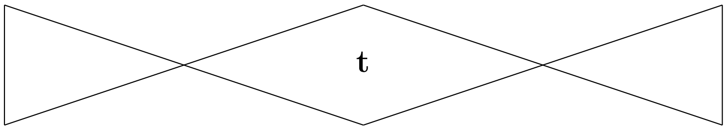
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
?	?	1	$\ldots$	1	1	1

101

h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0

011

h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
?	?	1	$\ldots$	1	1	1



100

h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	?

010

h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
?	1	1	$\ldots$	1	1	1

001

h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	?

The six tables correspond to the triples $(1,1,0)$, $(1,0,1)$, *etc. and their columns correspond to the components of* $\boldsymbol{T}$. *The entries are, again, values of the operations. For example, the zero in position* $(101, 2)$ *(table labeled* 101, *second column) indicates that* $h_2(1,0,1) = 0$.

In the following observations, we use $[\boldsymbol{i}, j]$ *to denote the value in the* j*th column in the table labeled* $\boldsymbol{i}$.

We write $\boldsymbol{i} \leq \boldsymbol{i}'$ *for triples* $\boldsymbol{i}, \boldsymbol{i}' \in I$ *if the inequalities hold component-wise. This order is depicted in the figure by lines.*

- $h_{j-1} \leq h_j$ *iff* $[i, j-1] \leq [i, j]$ *for every* $i \in I = \{110, 101, 011, 100, 010, 001\}$. *In particular, if* $[i, j] = 0$, *then* $[i, j-1] = 0$, *and if* $[i, j-1] = 1$, *then* $[i, j] = 1$.

- $h_{j-1} = h_j$ *iff the columns* $j-1$ *and* j *are identical.*

- $h_{j-1} \lhd h_j$ *iff* $[i, j-1] \leq [i', j]$ *for every* $i, i' \in I$ *such that* $i' \geq i$. *E.g., if* $[i, j] = 0$, *then* $[i, j-1] = [i-1, j-1] = \ldots, = 0$.

- $h_j = h_j^d$ *iff* $[i, j] = 1 - [111 - i, j]$ *for every* $i \in I$.

2.4.1 $\mathcal{X}$

$$\mathcal{X} = \{(h) \in \mathcal{I}_1 \mid h \lhd h \lhd h^d \lhd h^d\}$$

Lemma 2.30. $\mathcal{X}$ *is a minion core.*

Proof. Let $\xi : \mathcal{X} \to \mathcal{X}$ be a minion homomorphism. We aim to show that $\xi^{(2)}$ is a bijection. Note that the binary part of $\mathcal{X}$ is $\mathcal{X}^{(2)} = \{(x), (y), (\wedge)\}$. By Lemma 2.27, $\xi((x)) = (x)$ and $\xi((y)) = y$.

We use the following identity.

$$t(xy) = t(yx)$$

Note that $(\wedge) \in \mathcal{X}^{(2)}$ satisfies this identity and it is the only member of $\mathcal{X}$ that satisfies it. Since ξ preserves identities, we have $\xi((\wedge)) = (\wedge)$.

We have shown that $\xi^{(2)}$ is the identity on $\mathcal{X}^{(2)}$. By Lemma 2.26, $\mathcal{X}$ is a minion core. □

2.4.2 $\mathcal{Y}$

$$\mathcal{Y} = \{(h_1, h_2) \in \mathcal{I}_2 \mid h_1 \lhd h_1 \lhd h_2 = h_2^d \lhd h_1^d \lhd h_1^d\}$$

Lemma 2.31. *$\mathcal{Y}$ is a minion core.*

Proof. First note that the binary part is

$$\mathcal{Y}^{(2)} = \{\boldsymbol{x} = (x, x), \boldsymbol{y} = (y, y), (\wedge, x), (\wedge, y)\}.$$

We use the following identities

$$t(xxxxy) = t(xxyyy)$$
$$t(x_1, x_2, x_3, x_4, x_5) = t(x_{\sigma(1)}x_{\sigma(2)}x_{\sigma(3)}x_{\sigma(4)}x_{\sigma(5)}) \quad \forall \sigma \in S_5$$

We show that there are exactly two members of $\mathcal{Y}$ that satisfy those identities.

Assume that $\boldsymbol{t} = (h_1, h_2)$ satisfies the identities.

To begin, we establish that $h_1(11100)$ must be equal to 0. Let's assume, for contradiction, that $h_1(11100) = 1$. Since $h_1 \lhd h_2$, according to 2.28, this implies that $h_2(11100) = 1$ and $h_2(11110) = 1$. Consequently, as h_2 cannot be $\vee$, we conclude that $h_2(xxxyy) = x$ and $h_2(xxxxy) = x$. However, this contradicts the assumption that $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}, \boldsymbol{y})$.

Next, we aim to demonstrate that $h_1(11000)$ is also equal to 0. Let's assume, once again for contradiction, that $h_1(11000) = 1$. By employing the same property mentioned in 2.28, we find that $h_2(11000) = 1$ and $h_2(11100) = 1$. This, however, again contradicts the binary minors of $\mathcal{Y}$ since none of them contain a $\vee$ in the second position.

Therefore, by combining these two results, we can conclude that the first operation in $\boldsymbol{t}(\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y})$ is necessarily $\wedge$. This implies that the first operation in $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y})$ is also $\wedge$ since $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}, \boldsymbol{y})$ and $\boldsymbol{t}$ exhibits symmetry.

Moving on, we consider the second operation in $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y})$, which can either be x or y. If the second operation is x, denoted as $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}) = (\wedge, x)$, then due to symmetry, we can infer that $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}, \boldsymbol{y}) = (\wedge, y)$ and $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}) = (\wedge, y)$ based on the assumption. Similarly, if $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}) = (\wedge, y)$, then applying symmetry leads to the conclusion that $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}, \boldsymbol{y}) = (\wedge, x)$ and $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}) = (\wedge, x)$ according to the assumption.

We have shown that if a multi-sorted operation satisfies the identities, then it is either $\boldsymbol{t}$ or $\boldsymbol{t}'$ defined by the following tables.

$\boldsymbol{t}$	h_1	h_2
10000	0	0
11000	0	1
11100	0	0
11110	0	1

$\boldsymbol{t}'$	h_1	h_2
10000	0	1
11000	0	0
11100	0	1
11110	0	0

On the other hand, both of these multi-sorted operations are in $\mathcal{Y}$ (see Remark 2.28).

Now consider a minion homomorphism $\xi : \mathcal{Y} \to \mathcal{Y}$. Since ξ preserve identities, we have $\xi(\boldsymbol{t}) = \boldsymbol{t}$ or $\xi(\boldsymbol{t}) = \boldsymbol{t}'$.

Consider the second case. By Lemma 2.27 and since ξ preserves minors, we get

$$\begin{aligned}
\xi^{(2)}(\boldsymbol{x}) &= \boldsymbol{x}, \\
\xi^{(2)}(\boldsymbol{y}) &= \boldsymbol{y}, \\
\xi^{(2)}((\wedge, x)) &= \xi(\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}, \boldsymbol{y})) = \xi(\boldsymbol{t}) \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}, \boldsymbol{y}) = (\wedge, y) \\
\xi^{(2)}((\wedge, y)) &= \xi(\boldsymbol{t} \circ (\boldsymbol{y}, \boldsymbol{y}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x})) = \xi(\boldsymbol{t}) \circ (\boldsymbol{y}, \boldsymbol{y}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}) = (\wedge, x)
\end{aligned}$$

Therefore $\xi^{(2)}$ is a bijection and then $\mathcal{Y}$ is a minion core by Lemma 2.26.

The first case when $\xi(\boldsymbol{t}) = \boldsymbol{t}$ is similar.

□

2.4.3 $\mathcal{W}$

$$\mathcal{W} = \{(h) \in \mathcal{I}_1 \mid h \lhd h = h^d \lhd h^d\}$$

Lemma 2.32. *$\mathcal{W}$ is a minion core.*

Proof. Let $\xi : \mathcal{W} \to \mathcal{W}$ be a minion homomorphism. We aim to show that $\xi^{(2)}$ is a bijection. The binary part of $\mathcal{W}$ is $\mathcal{W}^{(2)} = \{(x), (y)\}$. By Lemma 2.27, $\xi((x)) = (x)$ and $\xi((y)) = y$.

Therefore $\xi^{(2)}$ is the identity on $\mathcal{X}^{(2)}$. By Lemma 2.26, $\mathcal{X}$ is a minion core. □

2.4.4 $\mathcal{A}_n$

$$\mathcal{A}_n = \{(h_1, h_1, \ldots, h_n) \in \mathcal{I}_n \mid h_1 \lhd h_2 \lhd \cdots \lhd h_n \leq h_n^d \lhd \cdots \lhd h_2^d \lhd h_1^d\}$$

Lemma 2.33. *The only binary n-sorted operations $(h_1, h_2, \ldots, h_n)$, which satisfy the relations $h_1 \lhd h_2 \lhd \cdots \lhd h_n \leq h_n^d \lhd \cdots \lhd h_2^d \lhd h_1^d$ are*

- $(\underbrace{\wedge, \wedge, \ldots, \wedge}_{n \text{ times}})$
- $(\underbrace{\wedge, \wedge, \ldots, \wedge, x}_{n \text{ times}})$, $(\underbrace{\wedge, \wedge, \ldots, x, x}_{n \text{ times}})$, $\ldots$, $(\underbrace{\wedge, x, \ldots, x, x}_{n \text{ times}})$, $(\underbrace{x, x, \ldots, x, x}_{n \text{ times}}) = x$
- $(\underbrace{\wedge, \wedge, \ldots, \wedge, y}_{n \text{ times}})$, $(\underbrace{\wedge, \wedge, \ldots, y, y}_{n \text{ times}})$, $\ldots$, $(\underbrace{\wedge, y, \ldots, y, y}_{n \text{ times}})$, $(\underbrace{y, y, \ldots, y, y}_{n \text{ times}}) = y$

Proof. For $n = 1$, the only binary operations h_2 that satisfy the relation $h_1 \leq h_1^d$ are $\wedge$, x, and y. Thus, the lemma holds for $n = 1$.

Now assume that $(h_1, h_2, \ldots, h_{n-1}, h_n)$ is an n-sorted operation satisfying $h_1 \lhd h_2 \lhd \cdots \lhd h_n \leq h_n^d \lhd \cdots \lhd h_2^d \lhd h_1^d$. We will use the inductive assumption on the $(n-1)$-sorted operation $(h_2, \ldots, h_{n-1}, h_n)$.

By the inductive assumption, the only binary $(n-1)$-sorted operations that satisfy the relations $h_2 \lhd \cdots \lhd h_n \leq h_n^d \lhd \cdots \lhd h_2^d$ are:

- $(\underbrace{\wedge, \wedge, \dots, \wedge}_{n-1 \text{ times}})$
- $(\underbrace{\wedge, \wedge, \dots, \wedge, x}_{n-1 \text{ times}})$, $(\underbrace{\wedge, \wedge, \dots, x, x}_{n-1 \text{ times}})$, $\dots$, $(\underbrace{\wedge, x, \dots, x, x}_{n-1 \text{ times}})$, $(\underbrace{x, x, \dots, x, x}_{n-1 \text{ times}}) = x$
- $(\underbrace{\wedge, \wedge, \dots, \wedge, y}_{n-1 \text{ times}})$, $(\underbrace{\wedge, \wedge, \dots, y, y}_{n-1 \text{ times}})$, $\dots$, $(\underbrace{\wedge, y, \dots, y, y}_{n-1 \text{ times}})$, $(\underbrace{y, y, \dots, y, y}_{n-1 \text{ times}}) = y$

Now consider the possible values of h_1. If $h_2 = \wedge$, then necessarily $h_1 = \wedge$. If $h_2 = x$, there are two possibilities: either $h_1 = \wedge$, or $h_1 = x$. Similarly, if $h_2 = y$, there are two possibilities: either $h_1 = \wedge$, or $h_1 = y$.

By combining these values with the operations obtained from the inductive assumption, we have exhausted all possible cases, and we have shown that the only binary n-sorted operations satisfying the given relations are as stated in the lemma. □

Lemma 2.34. *$\mathcal{A}_n$ is a minion core for every $n \in \mathbb{N}$.*

Proof. We use the following identities.

$$\begin{aligned}
t_1(xxxyy) &= x\\
t_2(xxxyy) &= t_1(xxxxy)\\
t_3(xxxyy) &= t_2(xxxxy)\\
&\vdots\\
t_n(xxxyy) &= t_{n-1}(xxxxy)\\
t_n(yyyyx) &= t_n(xxxxy)\\
t_i(x_1x_2x_3x_4x_5) &= t_i(x_{\sigma(1)}x_{\sigma(2)}x_{\sigma(3)}x_{\sigma(4)}x_{\sigma(5)}) \quad \forall i \in [n]\ \forall \sigma \in S_5
\end{aligned}$$

We show that there exist unique $\boldsymbol{t}_1, \dots, \boldsymbol{t}_n$ in $\mathcal{A}_n$ satisfying those identities. We start by proving uniqueness. Let $\boldsymbol{t}_1, \dots, \boldsymbol{t}_n$ by such multi-sorted operations in $\mathcal{A}_n$.

We begin by applying the first identity $t_1(xxxyy) = x$, which allows us to fill the second and the third rows in the table for $\boldsymbol{t}_1$:

$\boldsymbol{t}_1$	h_1	h_2		h_{n-1}	h_n
10000	?	?	...	?	?
11000	0	0	...	0	0
11100	1	1	...	1	1
11110	?	?	...	?	?

We fill in ones in the last row and zeros in the first row according to Remark 2.28:

$\boldsymbol{t}_1$	h_1	h_2		h_{n-1}	h_n
10000	0	0	...	0	?
11000	0	0	...	0	0
11100	1	1	...	1	1
11110	?	1	...	1	1

Now we use the second identity to fill in the table for $\boldsymbol{t}_2$ (the identity says that we should "copy" the first and the fourth rows of the table for $\boldsymbol{t}_1$ and "paste" them into the second and the third row of the table for $\boldsymbol{t}_2$ respectively):

$\boldsymbol{t}_2$	h_1	h_2		h_{n-1}	h_n
10000	?	?	...	?	?
11000	0	0	...	0	?
11100	?	1	...	1	1
11110	?	?	...	?	?

Again, we use Remark 2.28 to fill in zeroes in the first row and ones in the last one:

$\boldsymbol{t}_2$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	?	?
11000	0	0	0	...	0	0	?
11100	?	1	1	...	1	1	1
11110	?	?	1	...	1	1	1

We repeat the same until we reach the final operations. Therefore, in $\boldsymbol{t}_{n-1}$ we have:

$\boldsymbol{t}_{n-1}$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	?	?	...	?	?	?
11000	0	0	?	...	?	?	?
11100	?	?	?	...	?	1	1
11110	?	?	?	...	?	?	1

And for $\boldsymbol{t}_n$:

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	?	?	?	...	?	?	?
11000	0	?	?	...	?	?	?
11100	?	?	?	...	?	?	1
11110	?	?	?	...	?	?	?

According to Lemma 2.33 the only binary n-sorted operation which belongs to $\mathcal{A}_n$ and satisfies the last identity $t_n(yyyyx) = t_n(xxxxy)$ is $\underbrace{(\wedge, \dots, \wedge)}_{\text{n times}}$, therefore we can also fill the first and the last rows in the table for $\boldsymbol{t}_n$:

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	?	?	...	?	?	?
11100	?	?	?	...	?	?	1
11110	0	0	0	...	0	0	0

Moreover, we can fill zeroes in the third, and consequently, second row:

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	?	?	...	?	?	?
11100	0	0	0	...	0	0	1
11110	0	0	0	...	0	0	0

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	?	?
11100	0	0	0	...	0	0	1
11110	0	0	0	...	0	0	0

The only possibility for the last element in the second row is 0, otherwise $f(xxxyy)$ would be $\vee$, which contradicts binary n-sorted functions in $\mathcal{A}_n$ as proven in Lemma 2.33. This zero then necessarily implies 0 in the previous position in this row:

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	0	0	0	...	0	0	1
11110	0	0	0	...	0	0	0

Now if we use the n-th identity, we "copy" the second and the third rows of the table for $\boldsymbol{t}_n$ into the first and the fourth rows of $\boldsymbol{t}_{n-1}$:

$\boldsymbol{t}_{n-1}$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	?	...	?	?	?
11100	?	?	?	...	?	1	1
11110	0	0	0	...	0	0	1

Again, we use Remark 2.28 to fill zeroes in the second and the third rows:

$\boldsymbol{t}_{n-1}$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	?	?	?
11100	0	0	0	...	0	1	1
11110	0	0	0	...	0	0	1

As before, the only possibilities for the last two question marks are zeroes as there are no $\vee$. The remaining question mark is 0, because otherwise the minor wouldn't be in 2.37:

$\boldsymbol{t}_{n-1}$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	0	0	0	...	0	1	1
11110	0	0	0	...	0	0	1

We use the same approach until we reach $\boldsymbol{t}_1$.

$\boldsymbol{t}_1$	h_1	h_2		h_{n-1}	h_n
10000	0	0	...	0	0
11000	0	0	...	0	0
11100	1	1	...	1	1
11110	0	1	...	1	1

We have shown that if $\boldsymbol{t}_1, \dots \boldsymbol{t}_n$ are in $\mathcal{A}_n$ and satisfy the identities, then they necessarily have the following tables:

$\boldsymbol{t}_1$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	1	1	1	...	1	1	1
11110	0	1	1	...	1	1	1

$\boldsymbol{t}_2$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	0	1	1	...	1	1	1
11110	0	0	1	...	1	1	1

$$\vdots$$

$\boldsymbol{t}_{n-1}$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	0	0	0	...	0	1	1
11110	0	0	0	...	0	0	1

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	0	0	0	...	0	0	1
11110	0	0	0	...	0	0	0

On the other hand, these multi-sorted operations satisfy the identities and belong to $\mathcal{A}_n$, as is easily checked using Remark 2.28.

Every minion homomorphism $\xi : \mathcal{A}_n \to \mathcal{A}_n$ therefore satisfies $\xi(\boldsymbol{t}_i) = \boldsymbol{t}_i$ for every $i \in [n]$. Note that every binary member of $\mathcal{A}_n$ is a minor of some $\boldsymbol{t}_i$:

$$\begin{aligned}
\boldsymbol{t}_1 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) &= \boldsymbol{x} \\
\boldsymbol{t}_2 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) &= (\wedge, \underbrace{x,\dots,x,x}_{n-1\text{ times}}) \\
\boldsymbol{t}_3 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) &= (\wedge,\wedge, \underbrace{x,\dots,x,x}_{n-2\text{ times}}) \\
&\vdots \\
\boldsymbol{t}_{n-1} \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) &= (\underbrace{\wedge,\wedge,\dots,\wedge,\wedge}_{n-1\text{ times}}, x) \\
\boldsymbol{t}_n \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) &= (\underbrace{\wedge,\wedge,\dots,\wedge,\wedge}_{n\text{ times}})
\end{aligned}$$

Therefore the binary part $\xi^{(2)}$ is the identity (see the argument in the proof of Lemma 2.31) and $\mathcal{A}_n$ is a minion core by Lemma 2.26.

□

Alternatively, in order to prove that A_n is a minion core, we can use the following ternary identities:

$$
\begin{aligned}
t_1(yxy) &= x \\
t_2(yxy) &= t_1(yxx) \\
t_3(yxy) &= t_2(yxx) \\
&\vdots \\
t_n(yxy) &= t_{n-1}(yxx) \\
t_n(xxy) &= t_n(yyx)
\end{aligned}
$$

We start filling the table of the first multi-sorted operation $\boldsymbol{t}_1$ from the first identity:

110

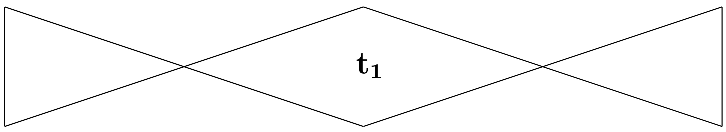
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

101

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			



100

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

010

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
1	1	1	…	1	1	1

001

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

As explained in Remark 2.29, ones in the row below for 100 imply ones in the rows above for $h_2, \dots, h_n$ for the elements 110 and 011 connected by vertical lines with 010. Similarly, zeroes in the table above for 101 imply zeroes in the tables in the row below for the elements 100 and 001 for the elements $h_1, \dots, h_{n-1}$:

110

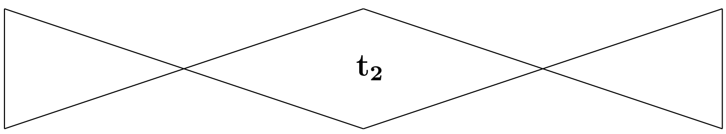
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

101

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1



100

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	?

010

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
1	1	1	…	1	1	1

001

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	?

According to Lemma 2.33, $h_n(xyx)$ cannot be equal to $\vee$. Consequently, we are able to fill the final zero in the tables for 100 and 001.

110

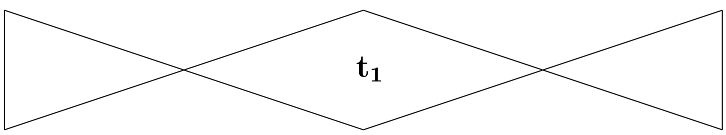
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

101

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1



100

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

010

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
1	1	1	…	1	1	1

001

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

Utilizing the second identity, we replicate the tables corresponding to 011 and 100 for $\boldsymbol{t}_1$ within the tables for $\boldsymbol{t}_2$ representing the tuples 010 and 101 respectively.

110

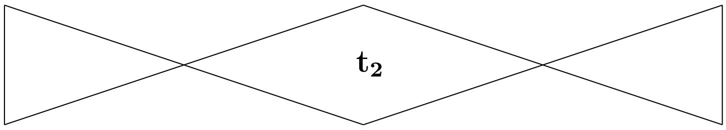
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

101

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			



100

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

010

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

001

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

Again zeroes in the table for 101, as it is located in the row above, imply zeroes for operations $h_1, \ldots, h_{n-1}$ in the tables for 100 and 001 according to Remark 2.29. Similarly, ones in the table for 010 imply ones for operations $h_3, \ldots, h_{n-1}$ in the tables for 110 and 011:

110

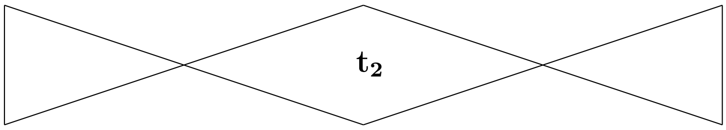
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	1	…	1	1	1

101

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	1	…	1	1	1



100

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	?

010

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

001

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	?

Moreover, again according to Lemma 2.29 $h_n(xyx)$ cannot be equal to $\vee$. Consequently we are able to fill final zeroes in the tables for 100 and 001.

110

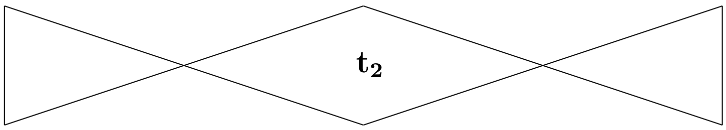
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	1	…	1	1	1

101

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	1	…	1	1	1



100

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

010

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

001

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

Now using the third inequality, we "copy" and "paste" the tables for 011 and 100 into the tables for the tuples 010 and 101 respectively for the multi-sorted operation $\boldsymbol{t}_3$:

110

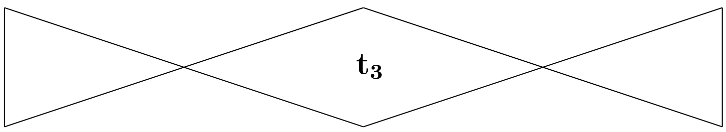
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

101

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			



100

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

010

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	1	…	1	1	1

001

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

And, consequently, utilizing Remark 2.29 and then Lemma 2.33:

110

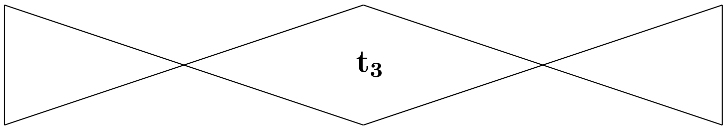
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	?	…	1	1	1

101

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	?	…	1	1	1



100

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

010

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	1	…	1	1	1

001

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

By continuing filling the tables in the similar manner, for $\boldsymbol{t}_{n-1}$ we obtain:

110

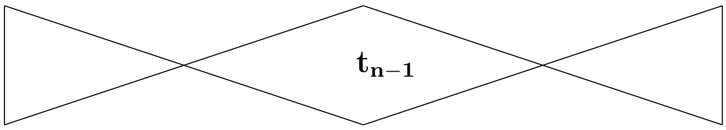
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	?	…	?	?	1

101

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	?	…	?	?	1



100

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

010

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	?	…	?	1	1

001

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

And for $\boldsymbol{t}_n$:

110

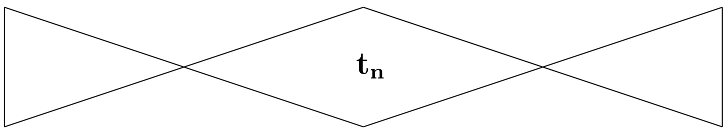
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

101

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			



100

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

010

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	?	…	?	?	1

001

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

Finally, utilizing the last equality we have:

110

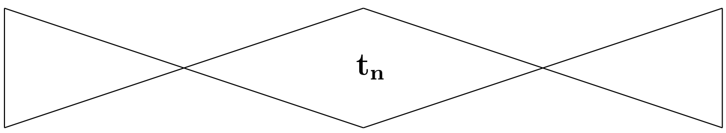
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

101

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0



100

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
1	1	1	…	1	1	1

010

h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	?	…	?	?	1

001

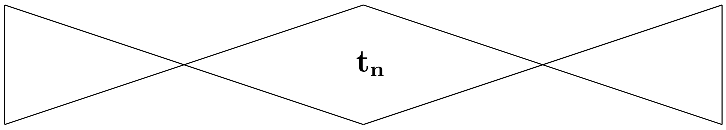
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

And again utilizing Remark 2.29:

110						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0

101						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0

011						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
			$\dots$			



100						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
			$\dots$			

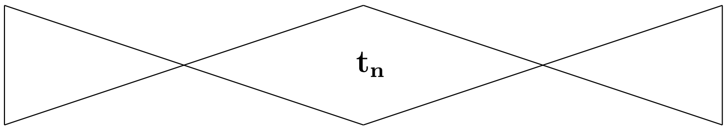
010						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
?	?	?	$\dots$	?	?	1

001						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0

110						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0

101						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0

011						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
			$\dots$			



100						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	?

010						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	1

001						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0

Now we return to the previous operations and fill the question marks. It turns out that all of them are zeroes.

Therefore, we have shown that if ternary operations $\boldsymbol{t}_1, \dots, \boldsymbol{t}_n$ satisfy the identities above, then necessarily:

$$
\begin{aligned}
\boldsymbol{t}_1(\boldsymbol{y},\boldsymbol{x},\boldsymbol{y}) &= \boldsymbol{x} \\
\boldsymbol{t}_2(\boldsymbol{y},\boldsymbol{x},\boldsymbol{y}) &= \boldsymbol{t}_1(\boldsymbol{y},\boldsymbol{x},\boldsymbol{x}) = (\underbrace{\wedge, x, \dots, x, y}_{n \text{ times}}) \\
\boldsymbol{t}_3(\boldsymbol{y},\boldsymbol{x},\boldsymbol{y}) &= \boldsymbol{t}_2(\boldsymbol{y},\boldsymbol{x},\boldsymbol{x}) = (\underbrace{\wedge, \wedge, x, \dots, x, x}_{n \text{ times}}) \\
&\vdots \\
\boldsymbol{t}_n(\boldsymbol{y},\boldsymbol{x},\boldsymbol{y}) &= \boldsymbol{t}_{n-1}(\boldsymbol{y},\boldsymbol{x},\boldsymbol{x}) = (\underbrace{\wedge, \wedge, \dots, \wedge, x}_{n \text{ times}}) \\
\boldsymbol{t}_n(\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) &= \boldsymbol{t}_n(\boldsymbol{y},\boldsymbol{y},\boldsymbol{x}) = (\underbrace{\wedge, \wedge, \dots, \wedge, \wedge}_{n \text{ times}})
\end{aligned}
$$

On the other hand, these multi-sorted operations satisfy the identities and belong to $\mathcal{A}_n$, as is easily checked using Remark 2.29.

By using a similar argument as for 5-ary operations im Lemma 2.34, if ξ is a minion homomorphism, then the binary part of $\xi^{(2)}$ is the identity (see the argument in the proof of Lemma 2.31), therefore $\mathcal{A}_n$ is a minion core by Lemma 2.26.

2.4.5 $\mathcal{B}_n$

$$\mathcal{B}_n = \{(h_1, h_2, \dots, h_n) \in \mathcal{I}_n \mid h_1 \lhd h_2 \lhd \dots \lhd h_n \lhd h_n^d \lhd \dots \lhd h_2^d \lhd h_1^d\}$$

Lemma 2.35. *The only binary n-sorted operations $(h_1, h_2, \dots, h_n)$, which satisfy the relations $h_1 \lhd h_2 \lhd \cdots \lhd h_n \lhd h_n^d \lhd \cdots \lhd h_2^d \lhd h_1^d$ are*

- $\underbrace{(\wedge, \wedge, \dots, \wedge)}_{n \text{ times}}$
- $\underbrace{(\wedge, \wedge, \dots, \wedge, x)}_{n \text{ times}}, \underbrace{(\wedge, \wedge, \dots, x, x)}_{n \text{ times}}, \dots, \underbrace{(\wedge, x, \dots, x, x)}_{n \text{ times}}, \underbrace{(x, x, \dots, x, x)}_{n \text{ times}} = x$
- $\underbrace{(\wedge, \wedge, \dots, \wedge, y)}_{n \text{ times}}, \underbrace{(\wedge, \wedge, \dots, y, y)}_{n \text{ times}}, \dots, \underbrace{(\wedge, y, \dots, y, y)}_{n \text{ times}}, \underbrace{(y, y, \dots, y, y)}_{n \text{ times}} = y$

Proof. The proof is analogous as for Lemma 2.33, because for binary n-sorted operations relations $h_1 \lhd h_2 \lhd \cdots \lhd h_n \leq h_n^d \lhd \cdots \lhd h_2^d \lhd h_1^d$ and $h_1 \lhd h_2 \lhd \cdots \lhd h_n \lhd h_n^d \lhd \cdots \lhd h_2^d \lhd h_1^d$ are equivalent. □

Lemma 2.36. *Minion $\mathcal{B}_n$ is a minion core for $n \in \mathbb{N}$.*

Proof. For minion $\mathcal{B}_n$ we use the same identities as in $\mathcal{A}_n$:

$$\begin{aligned}
t_1(xxxyy) &= x \\
t_2(xxxyy) &= t_1(xxxxy) \\
t_3(xxxyy) &= t_2(xxxxy) \\
&\vdots \\
t_n(xxxyy) &= t_{n-1}(xxxxy) \\
t_n(yyyyx) &= t_n(xxxxy) \\
t_i(x_1x_2x_3x_4x_5) &= t_i(x_{\sigma(1)}x_{\sigma(2)}x_{\sigma(3)}x_{\sigma(4)}x_{\sigma(5)}) \quad \forall i \in [n]\ \forall \sigma \in S_5
\end{aligned}$$

The tables are filled exactly as in the previous case, as they are derived from the given inequalities $h_1 \lhd \cdots \lhd h_n$. Thus, we can conclude the following:

$\boldsymbol{t}_1$	h_1	h_2		h_{n-1}	h_n
10000	0	0	...	0	0
11000	0	0	...	0	0
11100	1	1	...	1	1
11110	0	1	...	1	1

$\boldsymbol{t}_2$	h_1	h_2	h_3		h_{n-1}	h_n
10000	0	0	0	...	0	0
11000	0	0	0	...	0	0
11100	0	1	1	...	1	1
11110	0	0	1	...	1	1

$\vdots$

$\boldsymbol{t}_{n-1}$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	0	0	0	...	0	1	1
11110	0	0	0	...	0	0	1

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	0	0	0	...	0	0	1
11110	0	0	0	...	0	0	0

We only need to check that these operations indeed are in $\mathcal{B}_n$.

The inequalities $h_1 \lhd h_2 \lhd \cdots \lhd h_n$ hold by construction.

We only need to verify that $h_n \lhd h_n^d$ for all operations $\boldsymbol{t}_i$, $i \in [n]$. Consider the last column in the table for operations $h_1, \ldots, h_{n-1}$:

$\boldsymbol{t}_i$	h_n
10000	0
11000	0
11100	1
11110	1

It can be observed that h_n^d has the same values as h_n:

$\boldsymbol{t}_i$	h_n^d
10000	0
11000	0
11100	1
11110	1

Both h_n and h_n^d are monotone, confirming the relation $h_n \lhd h_n^d$.

Now, let's consider the last column for the multi-sorted operation $\boldsymbol{t}_n$:

$\boldsymbol{t}_n$	h_n
10000	0
11000	0
11100	1
11110	0

The corresponding values for h_n^d are:

$\boldsymbol{t}_n^d$	h_n
10000	1
11000	1
11100	0
11110	1

We can observe that $\boldsymbol{x}$ is a monotone binary operation that satisfies $h_n \leq x \leq h_n^d$, hence $h_n \lhd h_n^d$ holds.

Therefore, in both cases, the relation $h_n \lhd h_n^d$ is satisfied.

Similar to the previous Lemma 2.34, we have demonstrated the existence of unique operations $\boldsymbol{t}_1, \ldots, \boldsymbol{t}_n$ in $\mathcal{B}_n$ that fulfill the given identities. These operations yield the following binary minors:

$$
\begin{aligned}
\boldsymbol{t}_1 \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}) &= \boldsymbol{x} \\
\boldsymbol{t}_2 \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}) &= (\wedge, \underbrace{x, \dots, x, x}_{n-1 \text{ times}}) \\
\boldsymbol{t}_3 \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}) &= (\wedge, \wedge, \underbrace{x, \dots, x, x}_{n-2 \text{ times}}) \\
&\vdots \\
\boldsymbol{t}_{n-1} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}) &= (\underbrace{\wedge, \wedge, \dots, \wedge, \wedge}_{n-1 \text{ times}}, x) \\
\boldsymbol{t}_n \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}) &= (\underbrace{\wedge, \wedge, \dots, \wedge, \wedge}_{n \text{ times}})
\end{aligned}
$$

Therefore, each minion homomorphism ξ would map them to themselves.

□

2.4.6 $\mathcal{C}_n$

$$\mathcal{C}_n = \{(h_1, h_2, \dots, h_n) \in \mathcal{I}_n \mid h_1 \lhd h_2 \lhd \dots \lhd h_n = h_n^d \lhd \dots \lhd h_2^d \lhd h_1^d\}$$

Lemma 2.37. *The only binary n-sorted operations $(h_1, h_2, \dots, h_n)$, which satisfy the relations $h_1 \lhd h_2 \lhd \dots \lhd h_n = h_n^d \lhd \dots \lhd h_2^d \lhd h_1^d$ are*

- $\underbrace{(\wedge, \wedge, \dots, \wedge, x)}_{n \textit{ times}}, \underbrace{(\wedge, \wedge, \dots, x, x)}_{n \textit{ times}}, \dots, \underbrace{(\wedge, x, \dots, x, x)}_{n \textit{ times}}, \underbrace{(x, x, \dots, x, x)}_{n \textit{ times}} = x$
- $\underbrace{(\wedge, \wedge, \dots, \wedge, y)}_{n \textit{ times}}, \underbrace{(\wedge, \wedge, \dots, y, y)}_{n \textit{ times}}, \dots, \underbrace{(\wedge, y, \dots, y, y)}_{n \textit{ times}}, \underbrace{(y, y, \dots, y, y)}_{n \textit{ times}} = y$

Proof. For $n = 1$, the only binary operations h_1 that satisfy the relation $h_1 = h_1^d$ are x and y. Thus, the lemma holds for $n = 1$.

Now assume that $(h_1, h_2, \dots, h_{n-1}, h_n)$ is an n-sorted operation satisfying $h_1 \lhd h_2 \lhd \dots \lhd h_n \leq h_n^d \lhd \dots \lhd h_2^d \lhd h_1^d$. We will use the inductive assumption on the $(n-1)$-sorted operation $(h_2, \dots, h_{n-1}, h_n)$.

By the inductive assumption, the only binary $(n-1)$-sorted operations that satisfy the relations $h_2 \lhd \dots \lhd h_{n-2} = h_{n-2}^d \lhd \dots \lhd h_2^d$ are:

- $\underbrace{(\wedge, \wedge, \dots, \wedge, x)}_{n-1 \text{ times}}, \underbrace{(\wedge, \wedge, \dots, x, x)}_{n-1 \text{ times}}, \dots, \underbrace{(\wedge, x, \dots, x, x)}_{n-1 \text{ times}}, \underbrace{(x, x, \dots, x, x)}_{n-1 \text{ times}} = x$
- $\underbrace{(\wedge, \wedge, \dots, \wedge, y)}_{n-1 \text{ times}}, \underbrace{(\wedge, \wedge, \dots, y, y)}_{n-1 \text{ times}}, \dots, \underbrace{(\wedge, y, \dots, y, y)}_{n-1 \text{ times}}, \underbrace{(y, y, \dots, y, y)}_{n-1 \text{ times}} = y$

Now consider the possible values of h_2. If $h_2 = \wedge$, then necessarily $h_1 = \wedge$. If $h_2 = x$, there are two possibilities: either $h_1 = \wedge$, or $h_1 = x$. Similarly, if $h_2 = y$, there are two possibilities: either $h_1 = \wedge$, or $h_1 = y$.

By combining these values with the operations obtained from the inductive assumption, we have exhausted all possible cases, and we have shown that the only binary n-sorted operations satisfying the given relations are as stated in the lemma. □

Lemma 2.38. *Minion $\mathcal{C}_n$ is a minion core for $n \in \mathbb{N}$.*

Proof. For the minion $\mathcal{C}_n$, we use the following identities:

$$
\begin{aligned}
t_1(xxxyy) &= x \\
t_2(xxxyy) &= t_1(xxxxy) \\
t_3(xxxyy) &= t_2(xxxxy) \\
&\vdots \\
t_{n-1}(xxxyy) &= t_{n-2}(xxxxy) \\
t_n(xxxyy) &= t_{n-1}(xxxxy) = t_n(yyyyx) \\
t_i(x_1x_2x_3x_4x_5) &= t_i(x_{\sigma(1)}x_{\sigma(2)}x_{\sigma(3)}x_{\sigma(4)}x_{\sigma(5)}) \quad \forall i \in [n]\ \forall \sigma \in S_5
\end{aligned}
$$

It is worth noting that these identities closely resemble those of $\mathcal{B}_n$, with the only distinction being the identity $t_n(xxxyy) = t_n(yyyyx)$. Therefore the first $n-1$ tables are filled exactly as in the previous case.

$\boldsymbol{t}_1$	h_1	h_2		h_{n-1}	h_n
10000	0	0	...	0	?
11000	0	0	...	0	0
11100	1	1	...	1	1
11110	?	1	...	1	1

$\boldsymbol{t}_2$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	?	?
11000	0	0	0	...	0	0	?
11100	?	1	1	...	1	1	1
11110	?	?	1	...	1	1	1

$\vdots$

$\boldsymbol{t}_{n-1}$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	?	?	...	?	?	?
11000	0	0	?	...	?	?	?
11100	?	?	?	...	?	1	1
11110	?	?	?	...	?	?	1

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	?	?	?	...	?	?	?
11000	0	?	?	...	?	?	?
11100	?	?	?	...	?	?	1
11110	?	?	?	...	?	?	?

Let's analyze the last table for $\boldsymbol{t}_n$. We $h_n(11100) = 1$. Also according to Lemma 2.37 h_n cannot be $\wedge$, therefore due to symmetry $h_n(11000) = 0$, which consequently implies that all the elements in the second row are zeroes. Moreover, this implies that all the elements in the first row, except for the last one, should contain zeroes.

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	?
11000	0	0	0	...	0	0	0
11100	?	?	?	...	?	?	1
11110	?	?	?	...	?	?	?

Now we apply the identity $t_n(xxxyy) = t_n(yyyyx)$:

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	1
11000	0	0	0	...	0	0	0
11100	0	0	0	...	0	0	1
11110	0	0	0	...	0	0	0

Therefore, we have shown that $\boldsymbol{t}_n \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}) = \underbrace{(\wedge, \ldots, \wedge, x)}_{n \text{ times}}$ and $\boldsymbol{t}_n \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}) = \underbrace{(\wedge, \ldots, \wedge, y)}_{n \text{ times}}$.

Similarly as before, by "copying and pasting" the rows of the tables we obtain the following tables:

$\boldsymbol{t}_1$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	1	1	1	...	1	1	1
11110	0	1	1	...	1	1	1

$\boldsymbol{t}_2$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	0	1	1	...	1	1	1
11110	0	0	1	...	1	1	1

$\vdots$

$\boldsymbol{t}_{n-1}$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	0	0	0	...	0	1	1
11110	0	0	0	...	0	0	1

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n
10000	0	0	0	...	0	0	1
11000	0	0	0	...	0	0	0
11100	0	0	0	...	0	0	1
11110	0	0	0	...	0	0	0

Binary minors of $\boldsymbol{t}_1, \ldots, \boldsymbol{t}_n$ are the following:

$$
\begin{aligned}
\boldsymbol{t}_1 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) &= \boldsymbol{x} \\
\boldsymbol{t}_2 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) &= \boldsymbol{t}_1 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = \underbrace{(\wedge, x, \dots, x)}_{n \text{ times}} \\
\boldsymbol{t}_3 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) &= \boldsymbol{t}_2 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = \underbrace{(\wedge, \wedge, x, \dots, x)}_{n \text{ times}} \\
&\vdots \\
\boldsymbol{t}_{n-1} \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) &= \boldsymbol{t}_{n-2} \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = \underbrace{(\wedge, \dots, \wedge, x, x)}_{n \text{ times}} \\
\boldsymbol{t}_n \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) &= \boldsymbol{t}_{n-1} \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{t}_n \circ (\boldsymbol{y},\boldsymbol{y},\boldsymbol{y},\boldsymbol{y},\boldsymbol{x}) = \underbrace{(\wedge, \dots, \wedge, x)}_{n \text{ times}}
\end{aligned}
$$

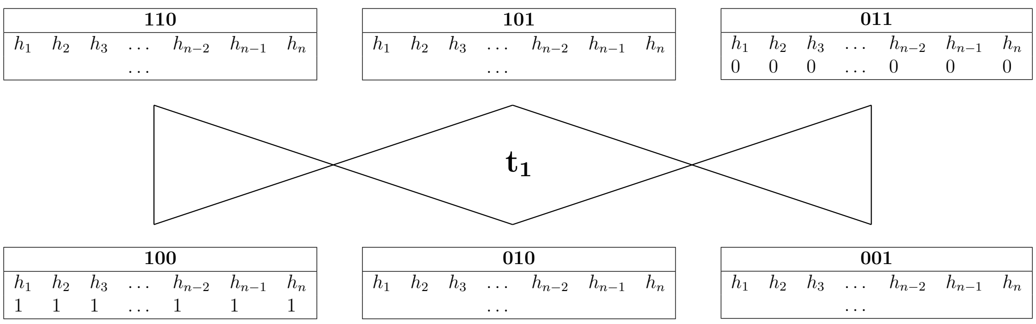
Similarly as in the previous case, we have shown that there are unique operations $\boldsymbol{t}_1, \dots, \boldsymbol{t}_n$ satisfying the identities. Clearly, all of them are in in $\mathcal{C}_n$. Therefore, each minion homomorphism ξ would map them to themselves.

□

Alternatively we could use the following ternary identities:

$$
\begin{aligned}
t_1(yxx) &= y \\
t_2(yxx) &= t_1(yxy) \\
t_3(yxx) &= t_2(yxy) \\
&\vdots \\
t_{n-1}(yxx) &= t_{n-2}(yxy) \\
t_n(yxx) &= t_{n-1}(yxy) = t_n(xyx)
\end{aligned}
$$

Again, we start filling the table of the first multi-sorted operation $\boldsymbol{t}_1$ from the first equality:

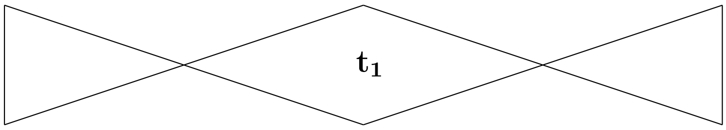


As explained in Remark 2.29, ones in the row below for 100 imply ones in the rows above for $h_2, \dots, h_n$ for the elements 110 and 101 connected by vertical lines with 100. Similarly, zeroes in the table above for 011 imply zeroes in the tables in the row below for the elements 010 and 001 for the elements $h_1, \dots, h_{n-2}$ (as $h_1 \lhd h_2 \lhd \dots \lhd h_{n-2} \lhd h_{n-1}$:

110						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

101						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

011						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0



100						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
1	1	1	…	1	1	1

010						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	?	?

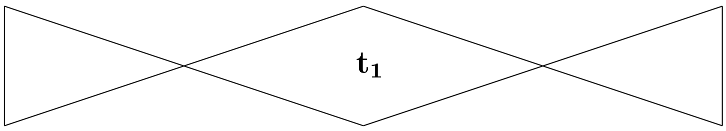
001						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	?	?

Moreover, as proven in Lemma 2.39, there can't be $\vee$ minor. Therefore, we can fill the last question marks in the tables for 010 and 001.

110						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

101						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

011						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0



100						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
1	1	1	…	1	1	1

010						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

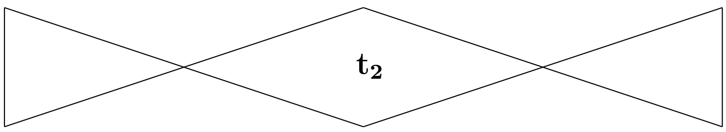
001						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

Now using the second equality we "copy" the tables for 010 and 101 for $\boldsymbol{t}_1$ into the tables for 011 and 100 for $\boldsymbol{t}_2$ respectively:

110						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

101						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

011						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0



100						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

010						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

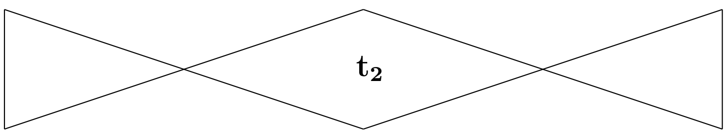
001						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

Similar to the case of $\boldsymbol{t}_1$, when examining the table for 100, the ones in that table imply the ones in the tables above for functions $h_3, \ldots, h_n$, which are connected to 100 through lines. Likewise, if there are zeroes in the table for 011, it signifies zeroes in the tables below for functions $h_1, \ldots, h_{n-2}$, and, due to Lemma 2.39, also for functions h_{n-1} and h_n:

110						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	1	…	1	1	1

101						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	1	…	1	1	1

011						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0



100						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

010						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	?

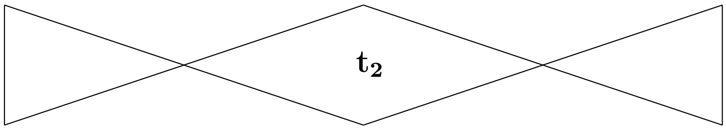
001						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	?

Once again, due to the absence of a $\vee$ minor, we can determine the final values for the tables corresponding to 010 and 001:

110						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
?	?	1	$\dots$	1	1	1

101						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
?	?	1	$\dots$	1	1	1

011						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0



100						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
?	1	1	$\dots$	1	1	1

010						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0

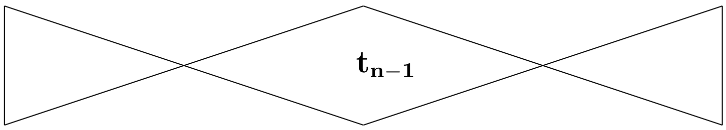
001						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0

Continuing this process, for the multi-sorted operation $\boldsymbol{t}_{n-1}$, we obtain the following corresponding tables:

110						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
			$\dots$			

101						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
			$\dots$			

011						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0



100						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
?	?	?	$\dots$	?	1	1

010						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
			$\dots$			

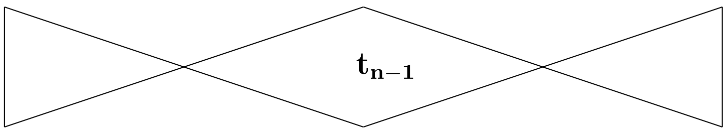
001						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
			$\dots$			

And, consequently:

110						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
?	?	?	$\dots$	?	?	1

101						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
?	?	?	$\dots$	?	?	1

011						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0



100						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
?	?	?	$\dots$	?	1	1

010						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0

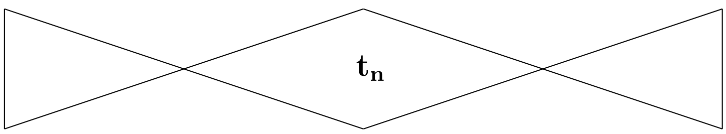
001						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0

Therefore for $\boldsymbol{t}_n$ we obtain using the last equality:

110						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
			$\dots$			

101						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0

011						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\dots$	0	0	0



100						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
?	?	?	$\dots$	?	?	1

010						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
?	?	?	$\dots$	?	?	1

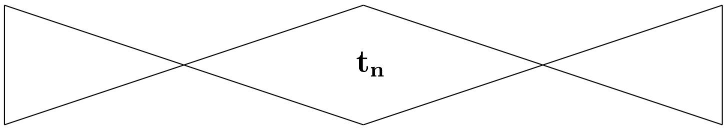
001						
h_1	h_2	h_3	$\dots$	h_{n-2}	h_{n-1}	h_n
			$\dots$			

Again, zero in the row above imply zeroes in the row below:

110						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

101						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

011						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0



100						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	1

010						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	1

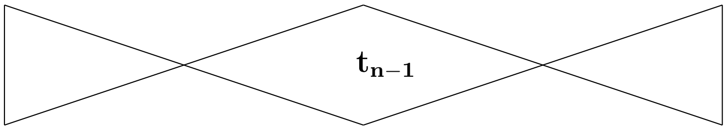
001						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
			…			

Now, as in previous cases, we go back to the previous operations and fill their tables. For $\boldsymbol{t}_{n-1}$:

110						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	?	…	?	?	1

101						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	1

011						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0



100						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	?	…	?	1	1

010						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

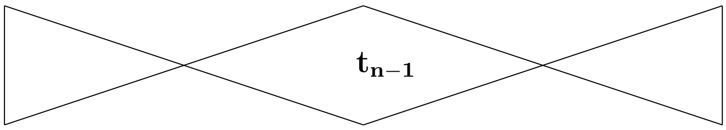
001						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

And, zeros in the row above imply zeroes for $h_1, \ldots, h_{n-2}$ for the element 100.

110						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	?	…	?	?	1

101						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	1

011						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0



100						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	1	1

010						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

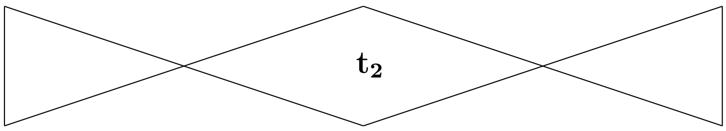
001						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

Finally for $\boldsymbol{t}_2$ and $\boldsymbol{t}_1$ we have:

110						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	?	1	…	1	1	1

101						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	1	…	1	1	1

011						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0



100						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	1	1	…	1	1	1

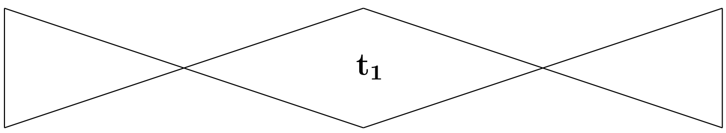
010						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

001						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

110						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
?	1	1	…	1	1	1

101						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	1	1	…	1	1	1

011						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0



100						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
1	1	1	…	1	1	1

010						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

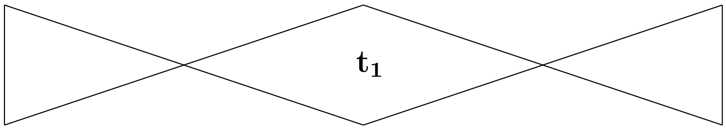
001						
h_1	h_2	h_3	…	h_{n-2}	h_{n-1}	h_n
0	0	0	…	0	0	0

Therefore we have the following tables for operations $\boldsymbol{t}_1, \ldots, \boldsymbol{t}_n$:

110						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
?	1	1	$\ldots$	1	1	1

101						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	1	1	$\ldots$	1	1	1

011						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0



100						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
1	1	1	$\ldots$	1	1	1

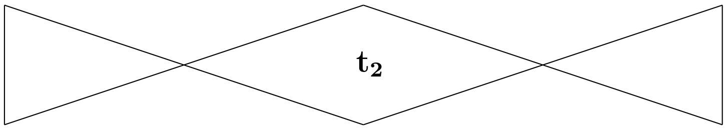
010						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0

001						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0

110						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
?	?	1	$\ldots$	1	1	1

101						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	1	$\ldots$	1	1	1

011						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0



100						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	1	1	$\ldots$	1	1	1

010						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0

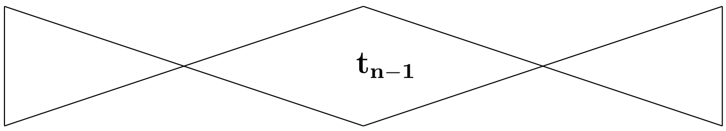
001						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0

$\vdots$

110						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
?	?	?	$\ldots$	?	?	1

101						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	1

011						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0



100						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	1	1

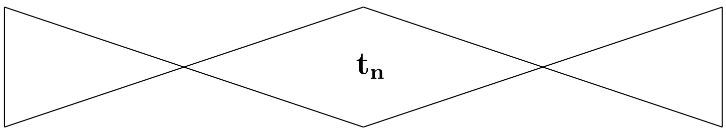
010						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0

001						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0

110						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
			$\ldots$			

101						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0

011						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	0



100						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	1

010						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
0	0	0	$\ldots$	0	0	1

001						
h_1	h_2	h_3	$\ldots$	h_{n-2}	h_{n-1}	h_n
			$\ldots$			

Here after filling the tables it turns out that

$$
\begin{aligned}
\boldsymbol{t}_1 \circ (\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{x}) &= \boldsymbol{y} \\
\boldsymbol{t}_2 \circ (\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{x}) &= \boldsymbol{t}_1 \circ (\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{y}) = \underbrace{(\wedge, y, \dots, y)}_{n \text{ times}} \\
&\vdots \\
\boldsymbol{t}_{n-1} \circ (\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{x}) &= \boldsymbol{t}_{n-2} \circ (\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{y}) = \underbrace{(\wedge, \dots, \wedge, y, y)}_{n \text{ times}} \\
\boldsymbol{t}_n \circ (\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{x}) &= \boldsymbol{t}_{n-1} \circ (\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{t}_n \circ (\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{x}) = \underbrace{(\wedge, \dots, \wedge, y)}_{n \text{ times}}
\end{aligned}
$$

2.4.7 $\mathcal{D}_n$

$$
\begin{aligned}
\mathcal{D}_{n-1} =& \{(h_1, h_2, \dots, h_n) \in \mathcal{I}_n \mid \\
& h_1 \lhd h_2 \lhd \dots \lhd h_{n-1} \leq h_n = h_n^d \leq h_{n-1}^d \lhd \dots \lhd h_2^d \lhd h_1^d, h_{n-1} \lhd h_{n-1}^d\}
\end{aligned}
$$

Lemma 2.39. *The only binary n-sorted operations $(h_1, h_2, \dots, h_n)$, which satisfy the relations $h_1 \lhd h_2 \lhd \dots \lhd h_{n-1} \leq h_n = h_n^d \leq h_{n-1}^d \lhd \dots \lhd h_2^d \lhd h_1^d$ and $h_{n-1} \lhd h_{n-1}^d$ are*

- $\underbrace{(\wedge, \wedge, \dots, \wedge, x)}_{n \text{ times}}, \underbrace{(\wedge, \wedge, \dots, x, x)}_{n \text{ times}}, \dots, \underbrace{(\wedge, x, \dots, x, x)}_{n \text{ times}}, \underbrace{(x, x, \dots, x, x)}_{n \text{ times}} = x$
- $\underbrace{(\wedge, \wedge, \dots, \wedge, y)}_{n \text{ times}}, \underbrace{(\wedge, \wedge, \dots, y, y)}_{n \text{ times}}, \dots, \underbrace{(\wedge, y, \dots, y, y)}_{n \text{ times}}, \underbrace{(y, y, \dots, y, y)}_{n \text{ times}} = y$

Proof. Again similar induction as before. □

Lemma 2.40. *Minion $\mathcal{D}_{n-1}$ is a minion core for $n \in \mathbb{N}$, $n > 1$.*

Proof. We will utilize the following identities:

$$
\begin{aligned}
t_1(xxxyy) &= x \\
t_2(xxxyy) &= t_1(xxxxy) \\
t_3(xxxyy) &= t_2(xxxxy) \\
&\vdots \\
t_{n-1}(xxxyy) &= t_{n-2}(xxxxy) \\
t_n(xxxyy) &= t_{n-1}(xxxxy) = t_n(yyyyx) \\
t_i(x_1x_2x_3x_4x_5) &= t_i(x_{\sigma(1)}x_{\sigma(2)}x_{\sigma(3)}x_{\sigma(4)}x_{\sigma(5)}) \quad \forall i \in [n]\ \forall \sigma \in S_5
\end{aligned}
$$

Notice that these operations are exactly the same as in Lemma 2.38. We start filling the tables in the same way as before. The only difference is that for convenience as $h_{n-1} \lhd h_{n-1}^d$ we have an additional column corresponding to h_{n-1}^d.

$\boldsymbol{t}_1$	h_1	h_2		h_{n-1}	h_n	h_{n-1}^d
10000	0	0	...	0	?	0
11000	0	0	...	0	0	0
11100	1	1	...	1	1	1
11110	?	1	...	1	1	1

Clearly $h_n(10000)$ is equal to 0.

$\boldsymbol{t}_1$	h_1	h_2		h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	0
11000	0	0	...	0	0	0
11100	1	1	...	1	1	1
11110	?	1	...	1	1	1

Again, we "copy" the first and last rows and "paste" them into the second and third rows of the next table.

$\boldsymbol{t}_2$	h_1	h_2	h_3		h_{n-1}	h_n	h^d_{n-1}
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	?	1	1	...	1	1	1
11110	?	?	1	...	1	1	1

Then in $\boldsymbol{t}_{n-1}$ we obtain:

$\boldsymbol{t}_{n-1}$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	?	?
11000	0	0	...	0	0	0	0
11100	?	?	...	?	1	1	1
11110	?	?	...	?	?	?	1

And finally for $\boldsymbol{t}_n$:

$\boldsymbol{t}_n$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000			...				
11000	0	0	...	0	0	?	?
11100	?	?	...	?	?	?	1
11110			...				

Again we fill zeroes which are implied:

$\boldsymbol{t}_n$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	?	?	?
11000	0	0	...	0	0	?	?
11100	?	?	...	?	?	?	1
11110			...				

And apply the last identity:

$\boldsymbol{t}_n$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	?	?	1
11000	0	0	...	0	0	?	?
11100	0	0	...	0	?	?	1
11110	0	0	...	0	0	?	?

Moreover, we can notice that $h^d_{n-1}(11110)$ is equal to 1, otherwise $h_{n-1}(10000)$ would be equal to 1 and the inequality $h_{n-1} \lhd h^d_{n-1}$ wouldn't be satisfied. This means that $h_{n-1}(11000)$ is equal to 0.

$\boldsymbol{t}_n$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	?	1
11000	0	0	...	0	0	?	?
11100	0	0	...	0	?	?	1
11110	0	0	...	0	0	?	1

Again we use the last identity and obtain:

$\boldsymbol{t}_n$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	?	1
11000	0	0	...	0	0	?	1
11100	0	0	...	0	0	?	1
11110	0	0	...	0	0	?	1

There are two possibilities for the column for h_n:

$\boldsymbol{t}_n$	h_n
10000	1
11000	0
11100	1
11110	0

$\boldsymbol{t}_n$	h_n
10000	0
11000	1
11100	0
11110	1

Consider the first case. Again, we fill the table for the last multi-sorted operation $\boldsymbol{t}_n$ and then go back to the previous operations:

$\boldsymbol{t}_n$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	1	1
11000	0	0	...	0	0	0	1
11100	0	0	...	0	0	1	1
11110	0	0	...	0	0	0	1

$\boldsymbol{t}_{n-1}$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	0	1
11000	0	0	...	0	0	0	0
11100	?	?	...	?	1	1	1
11110	0	0	...	0	0	1	1

$\boldsymbol{t}_{n-1}$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	0	1
11000	0	0	...	0	0	0	0
11100	0	0	...	0	1	1	1
11110	0	0	...	0	0	1	1

We go back to operations $\boldsymbol{t}_2$ and $\boldsymbol{t}_1$:

$\boldsymbol{t}_2$	h_1	h_2		h_{n-1}	h_n	h_{n-1}^d
10000	0	0	...	0	0	0
11000	0	0	...	0	0	0
11100	0	1	...	1	1	1
11110	0	0	...	1	1	1

$\boldsymbol{t}_1$	h_1	h_2		h_{n-1}	h_n	h_{n-1}^d
10000	0	0	...	0	0	0
11000	0	0	...	0	0	0
11100	1	1	...	1	1	1
11110	0	1	...	1	1	1

Therefore we obtain the following tables:

$\boldsymbol{t}_1$	h_1	h_2		h_{n-1}	h_n	h_{n-1}^d
10000	0	0	...	0	0	0
11000	0	0	...	0	0	0
11100	1	1	...	1	1	1
11110	0	1	...	1	1	1

$\boldsymbol{t}_2$	h_1	h_2		h_{n-1}	h_n	h_{n-1}^d
10000	0	0	...	0	0	0
11000	0	0	...	0	0	0
11100	0	1	...	1	1	1
11110	0	0	...	1	1	1

$$\vdots$$

$\boldsymbol{t}_{n-1}$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h_{n-1}^d
10000	0	0	...	0	0	0	1
11000	0	0	...	0	0	0	0
11100	0	0	...	0	1	1	1
11110	0	0	...	0	0	1	1

$\boldsymbol{t}_n$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h_{n-1}^d
10000	0	0	...	0	0	1	1
11000	0	0	...	0	0	0	1
11100	0	0	...	0	0	1	1
11110	0	0	...	0	0	0	1

And the following binary minors:

$$\boldsymbol{t}_1 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) = \boldsymbol{x}$$
$$\boldsymbol{t}_2 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) = \boldsymbol{t}_1 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = (\wedge, \underbrace{x,\dots,x}_{n-1 \text{ times}})$$
$$\boldsymbol{t}_3 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) = \boldsymbol{t}_2 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = (\wedge, \wedge, \underbrace{x,\dots,x}_{n-2 \text{ times}})$$
$$\vdots$$
$$\boldsymbol{t}_{n-1} \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) = \boldsymbol{t}_{n-2} \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = (\underbrace{\wedge,\dots,\wedge}_{n-2 \text{ times}}, x, x)$$
$$\boldsymbol{t}_n \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) = \boldsymbol{t}_{n-1} \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{t}_n \circ (\boldsymbol{y},\boldsymbol{y},\boldsymbol{y},\boldsymbol{y},\boldsymbol{x}) = (\underbrace{\wedge,\dots,\wedge}_{n-1 \text{ times}}, x)$$

In the second case:

$\boldsymbol{t}'_n$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	0	1
11000	0	0	...	0	0	1	1
11100	0	0	...	0	0	0	1
11110	0	0	...	0	0	1	1

$\boldsymbol{t}'_{n-1}$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	1	1
11000	0	0	...	0	0	0	0
11100	?	?	...	?	1	1	1
11110	0	0	...	0	0	0	1

$\boldsymbol{t}'_{n-1}$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	1	1
11000	0	0	...	0	0	0	0
11100	0	0	...	0	1	1	1
11110	0	0	...	0	0	0	1

We go back to operations $\boldsymbol{t}_2$ and $\boldsymbol{t}_1$:

$\boldsymbol{t}'_2$	h_1	h_2		h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	0
11000	0	0	...	0	0	0
11100	0	1	...	1	1	1
11110	0	0	...	1	1	1

$\boldsymbol{t}'_1$	h_1	h_2		h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	0
11000	0	0	...	0	0	0
11100	1	1	...	1	1	1
11110	0	1	...	1	1	1

Therefore we obtain:

$\boldsymbol{t}'_1$	h_1	h_2		h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	0
11000	0	0	...	0	0	0
11100	1	1	...	1	1	1
11110	0	1	...	1	1	1

$\boldsymbol{t}'_2$	h_1	h_2		h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	0
11000	0	0	...	0	0	0
11100	0	1	...	1	1	1
11110	0	0	...	1	1	1

$\vdots$

$\boldsymbol{t}'_{n-1}$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	1	1
11000	0	0	...	0	0	0	0
11100	0	0	...	0	1	1	1
11110	0	0	...	0	0	0	1

$\boldsymbol{t}'_n$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h^d_{n-1}
10000	0	0	...	0	0	0	1
11000	0	0	...	0	0	1	1
11100	0	0	...	0	0	0	1
11110	0	0	...	0	0	1	1

$$\boldsymbol{t}'_1 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) = \boldsymbol{x}$$
$$\boldsymbol{t}'_2 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) = \boldsymbol{t}'_1 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = (\wedge, \underbrace{x,\dots,x}_{n-1 \text{ times}})$$
$$\boldsymbol{t}'_3 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) = \boldsymbol{t}'_2 \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = (\wedge, \wedge, \underbrace{x,\dots,x}_{n-2 \text{ times}})$$
$$\vdots$$
$$\boldsymbol{t}'_{n-1} \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) = \boldsymbol{t}'_{n-2} \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = (\underbrace{\wedge,\dots,\wedge}_{n-2 \text{ times}}, x, x)$$
$$\boldsymbol{t}'_n \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y},\boldsymbol{y}) = \boldsymbol{t}'_{n-1} \circ (\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{x},\boldsymbol{y}) = \boldsymbol{t}'_n \circ (\boldsymbol{y},\boldsymbol{y},\boldsymbol{y},\boldsymbol{y},\boldsymbol{x}) = (\underbrace{\wedge,\dots,\wedge}_{n-1 \text{ times}}, y)$$

We have shown that there are two sets of operations satisfying the identities, namely the operations $\boldsymbol{t}_1,\dots,\boldsymbol{t}_n$ and the operations $\boldsymbol{t}'_1,\dots,\boldsymbol{t}'_n$.

On the other hand, these multi-sorted operations satisfy the identities and belong to $\mathcal{D}_n$, as is easily checked using Remark 2.28.

Every minion homomorphism $\xi : \mathcal{D}_n \to \mathcal{D}_n$ therefore satisfies $\xi(\boldsymbol{t}_i) = \boldsymbol{t}_i$ or $\xi(\boldsymbol{t}_i) = \boldsymbol{t}'_i$ for every $i \in [n]$. Since every binary member of $\mathcal{A}_n$ is a minor of some $\boldsymbol{t}_i$, the binary part $\xi^{(2)}$ is either the identity (see the argument in the proof of Lemma 2.31), or the identity for all the binary minors apart from $(\underbrace{\wedge,\dots,\wedge}_{n-1 \text{ times}}, y)$: it maps $(\underbrace{\wedge,\dots,\wedge}_{n-1 \text{ times}}, y)$ to $(\underbrace{\wedge,\dots,\wedge}_{n-1 \text{ times}}, x)$ and $(\underbrace{\wedge,\dots,\wedge}_{n-1 \text{ times}}, x)$ to $(\underbrace{\wedge,\dots,\wedge}_{n-1 \text{ times}}, y)$. In any case, the binary part of the minion homomorphism is a bijection. Therefore $\mathcal{D}_n$ is a minion core by Lemma 2.26.

□

2.5 Ordering

In this section, we will establish the mutual relations between the minion cores from the previous section, as depicted in the diagram 2.1. We will accomplish this by constructing minion homomorphisms between some of the multi-sorted minions and demonstrating that there are no homomorphisms between the remaining pairs of multi-sorted minions. Specifically, we prove the following theorem.

Theorem 2.41. *The following inequalities are satisfied.*

$$\begin{aligned}
&\mathcal{W} \leq \mathcal{Y} \leq \mathcal{X}\\
&\mathcal{X} \leq \cdots \leq \mathcal{B}_3 \leq \mathcal{A}_3 \leq \mathcal{B}_2 \leq \mathcal{A}_2 \leq \mathcal{B}_1 \leq \mathcal{A}_1 \leq \mathcal{T}\\
&\mathcal{Y} \leq \cdots \leq \mathcal{D}_3 \leq \mathcal{C}_3 \leq \mathcal{D}_2 \leq \mathcal{C}_2 \leq \mathcal{D}_1 \leq \mathcal{C}_1\\
&\mathcal{C}_n \leq \mathcal{A}_n, \quad \mathcal{D}_n \leq \mathcal{B}_n \quad \forall n \in \mathbb{N}.
\end{aligned}$$

There are no inequalities among the multi-sorted minions $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{W}$, $\mathcal{A}_n$, $\mathcal{B}_n$, $\mathcal{C}_n$, $\mathcal{D}_n$ and $\mathcal{T}$ other than those that follow from the above inequalities by reflexivity and transitivity of $\leq$.

2.5.1 Inequalities

The inequalities are witnessed by the following mappings. That each of them is a homomorphism follows immediately from the definitions of the multi-sorted minions.

$$\begin{aligned}
\zeta_{\mathcal{B}A} : \mathcal{B}_n &\to \mathcal{A}_n\\
(h_1, \ldots, h_n) &\mapsto (h_1, \ldots, h_n)
\end{aligned}$$

$$\begin{aligned}
\zeta_{\mathcal{D}B} : \mathcal{D}_n &\to \mathcal{B}_n\\
(h_1, \ldots, h_n, h_{n+1}) &\mapsto (h_1, \ldots, h_n)
\end{aligned}$$

$$\begin{aligned}
\zeta_{\mathcal{C}A} : \mathcal{C}_n &\to \mathcal{A}_n\\
(h_1, \ldots, h_n) &\mapsto (h_1, \ldots, h_n)
\end{aligned}$$

$$\begin{aligned}
\zeta_{\mathcal{D}C} : \mathcal{D}_n &\to \mathcal{C}_n\\
(h_1, \ldots, h_n, h_{n+1}) &\mapsto (h_1, \ldots, h_{n-1}, h_{n+1})
\end{aligned}$$

$$\begin{aligned}
\zeta_{\mathcal{A}B} : \mathcal{A}_{n+1} &\to \mathcal{B}_n\\
(h_1, \ldots, h_n, h_{n+1}) &\mapsto (h_1, \ldots, h_n)
\end{aligned}$$

$$
\begin{aligned}
\zeta_{\mathcal{C}D} : \mathcal{C}_{n+1} &\to \mathcal{D}_n \\
(h_1, \dots, h_n, h_{n+1}) &\mapsto (h_1, \dots, h_n, h_{n+1})
\end{aligned}
$$

$$
\begin{aligned}
\zeta_{\mathcal{Y}D} : \mathcal{Y} &\to \mathcal{D}_n \\
(h_1, h_2) &\mapsto (\underbrace{h_1, \dots, h_1}_{\text{n times}}, h_2)
\end{aligned}
$$

$$
\begin{aligned}
\zeta_{\mathcal{X}B} : \mathcal{X} &\to \mathcal{B}_n \\
(h) &\mapsto (\underbrace{h, \dots, h}_{\text{n times}})
\end{aligned}
$$

$$
\begin{aligned}
\zeta_{\mathcal{Y}X} : \mathcal{Y} &\to \mathcal{X} \\
(h_1, h_2) &\mapsto (h_1)
\end{aligned}
$$

$$
\begin{aligned}
\zeta_{\mathcal{W}Y} : \mathcal{W} &\to \mathcal{Y} \\
(h) &\mapsto (h, h)
\end{aligned}
$$

2.5.2 Non-inequalities

Lemma 2.42. $\mathcal{X} \not\leq \mathcal{C}_1$.

Proof. There exists a binary multi-sorted operation $\boldsymbol{t} \in \mathcal{X}$ satisfying $t(x, y) = t(y, x)$ (the same multi-sorted operation we used in Lemma 2.30 to prove that $\mathcal{X}$ is a core). If $\xi : \mathcal{X} \to \mathcal{C}_1$ is a minion homomorphism, then $\xi(\boldsymbol{t}) \circ (\boldsymbol{x}, \boldsymbol{y}) = \xi(\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{y})) = \xi(\boldsymbol{t} \circ (\boldsymbol{y}, \boldsymbol{x})) = \xi(\boldsymbol{t}) \circ (\boldsymbol{y}, \boldsymbol{x})$, but there is no such multi-sorted operation $\xi(\boldsymbol{t})$ in $\mathcal{C}_1$ (as there are no symmetric binary 1-sorted operations due to Lemma 2.37). □

Note that Lemma 2.42 implies the absence of any minion homomorphism from $\mathcal{X}$ to either $\mathcal{C}_n$ or $\mathcal{D}_n$ for all $n \in \mathbb{N}$. The assumption of such a homomorphism would inevitably lead to a contradiction. Suppose, for the sake of contradiction, that a minion homomorphism exists from $\mathcal{X}$ to $\mathcal{C}_n$ for $n > 1$. In that case, if we compose this homomorphism with a sequence of homomorphisms $\mathcal{C}_n \to \mathcal{D}_{n-1} \to \mathcal{C}_{n-1} \to \cdots \to \mathcal{D}_1 \to \mathcal{C}_1$, we would obtain a minion homomorphism from $\mathcal{X}$ to $\mathcal{C}_1$. However, this contradicts the statement made in Lemma 2.42. Similar reasoning applies to the case of $\mathcal{D}_n$.

Lemma 2.43. *$\mathcal{B}_n \not\leq \mathcal{C}_n$ for $n \in \mathbb{N}$.*

Proof. Similarly as in the previous lemma we will utilize the symmetry argument. There exists a binary multi-sorted operation $\boldsymbol{t} \in \mathcal{B}_n$ satisfying $t(yyyyx) = t(xxxxy)$ (the same multi-sorted operation we used in Lemma 2.30 to prove that $\mathcal{B}_n$ is a core). If $\xi : \mathcal{B}_n \to \mathcal{C}_n$ is a minion homomorphism, then $\xi(\boldsymbol{t}) \circ (\boldsymbol{y}, \boldsymbol{y}, \boldsymbol{y}, \boldsymbol{y}, \boldsymbol{x}) = \xi(\boldsymbol{t} \circ (\boldsymbol{y}, \boldsymbol{y}, \boldsymbol{y}, \boldsymbol{y}, \boldsymbol{x})) = \xi(\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y})) = \xi \circ (\boldsymbol{t})(\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y})$, but there is no such multi-sorted operation $\xi(\boldsymbol{t})$ in $\mathcal{C}_n$ (as there are no symmetric binary 1-sorted operations due to Lemma 2.37). □

Lemma 2.44. *$\mathcal{C}_n \not\leq \mathcal{B}_n$ for $n \in \mathbb{N}$.*

Proof. In this proof, we use the same set of identities as used in Lemma 2.38 to establish the properties of $\mathcal{C}_n$. These identities are as follows:

$$\begin{aligned}
t_1(xxxyy) &= x \\
t_2(xxxyy) &= t_1(xxxxy) \\
t_3(xxxyy) &= t_2(xxxxy) \\
&\vdots \\
t_{n-1}(xxxyy) &= t_{n-2}(xxxxy) \\
t_n(xxxyy) &= t_{n-1}(xxxxy) = t_n(yyyyx) \\
t(x_1, x_2, x_3, x_4, x_5) &= t(x_{\sigma(1)}x_{\sigma(2)}x_{\sigma(3)}x_{\sigma(4)}x_{\sigma(5)}) \quad \forall \sigma \in S_5
\end{aligned}$$

There are such operations in $\mathcal{C}_n$ as we have shown in Lemma 2.38. Assume that there exist operations in $\mathcal{B}_n$ satisfying these identities. We start filling the tables:

$\boldsymbol{t}_1$	h_1	h_2		h_{n-1}	h_n
10000	0	0	...	0	?
11000	0	0	...	0	0
11100	1	1	...	1	1
11110	?	1	...	1	1

$\boldsymbol{t}_2$	h_1	h_2		h_{n-1}	h_n
10000	0	0	...	?	?
11000	0	0	...	0	?
11100	?	1	...	1	1
11110	?	?	...	1	1

$\vdots$

$\boldsymbol{t}_{n-1}$	h_1	h_2		h_{n-1}	h_n
10000	0	?	...	?	?
11000	0	0	...	?	?
11100	?	?	...	1	1
11110	?	?	...	?	1

$\boldsymbol{t}_n$	h_1	h_2		h_{n-1}	h_n
10000	?	?	...	?	?
11000	0	?	...	?	?
11100	?	?	...	?	1
11110	?	?	...	?	?

Also, due to Lemma 2.35 we know that h_n cannot be $\wedge$, therefore necessarily $h_n(11000) = 0$, which implies that all the previous elements in this row are also zeroes.

$\boldsymbol{t}_n$	h_1	h_2		h_{n-1}	h_n
10000	?	?	...	?	?
11000	0	0	...	0	0
11100	?	?	...	?	1
11110	?	?	...	?	?

Therefore $h_n(xxxyy) = x$. Identity $t_n(xxxyy) = t_n(yyyyx)$ and symmetry then implies that $h_n(xxxxy) = y$:

$\boldsymbol{t}_n$	h_1	h_2		h_{n-1}	h_n
10000	?	?	...	?	1
11000	0	0	...	0	0
11100	?	?	...	?	1
11110	?	?	...	?	0

That is a contradiction with the assumption that $h_n \lhd h_n^d$, because, for example $h_n^d(11000) = 1 - h_n(00111) = 1 - h_n(11100) = 0$, but $h_n(10000) = 1$. Therefore there are no operations in $\mathcal{B}_n$ satisfying the identities

$$\begin{aligned}
t_1(xxxyy) &= x \\
t_2(xxxyy) &= t_1(xxxxy) \\
t_3(xxxyy) &= t_2(xxxxy) \\
&\vdots \\
t_{n-1}(xxxyy) &= t_{n-2}(xxxxy) \\
t_n(xxxyy) &= t_{n-1}(xxxxy) = t_n(yyyyx) \\
t(x_1, x_2, x_3, x_4, x_5) &= t(x_{\sigma(1)}x_{\sigma(2)}x_{\sigma(3)}x_{\sigma(4)}x_{\sigma(5)}) \quad \forall \sigma \in S_5,
\end{aligned}$$

and, consequently, there is no minion homomorphism from $\mathcal{C}_n$ to $\mathcal{B}_n$.

□

Lemma 2.45. $\mathcal{A}_{n+1} \not\leq \mathcal{D}_n$.

Proof. Once again, we employ the symmetry argument. In $\mathcal{A}_{n+1}$, there exists a 5-ary symmetric multi-sorted operation $\boldsymbol{t}$ that satisfies $t(yyyyx) = t(xxxxy)$ (e.g., the multi-sorted operation $\boldsymbol{t}_n$ mentioned in Lemma 2.34). However, no such multi-sorted operation exists in $\mathcal{D}_n$ since $\mathcal{D}_n$ lacks symmetric binary minors. Consequently, there is no minion homomorphism from $\mathcal{A}_{n+1}$ to $\mathcal{D}_n$.

□

Lemma 2.46. $\mathcal{D}_n \not\leq \mathcal{A}_{n+1}$.

Proof. Here we use the same identities we used in Lemma 2.40 to prove $\mathcal{D}_n$ is a core:

$$\begin{aligned}
t_1(xxxyy) &= x \\
t_2(xxxyy) &= t_1(xxxxy) \\
t_3(xxxyy) &= t_2(xxxxy) \\
&\vdots \\
t_{n-1}(xxxyy) &= t_{n-2}(xxxxy) \\
t_n(xxxyy) &= t_{n-1}(xxxxy) = t_n(yyyyx) \\
t(x_1, x_2, x_3, x_4, x_5) &= t(x_{\sigma(1)}x_{\sigma(2)}x_{\sigma(3)}x_{\sigma(4)}x_{\sigma(5)}) \quad \forall \sigma \in S_5
\end{aligned}$$

As usual, we show that there are no operations satisfying these identitites in $\mathcal{A}_{n+1}$. For contradiction assume that such 5-ary operations exist in $\mathcal{A}_{n+1}$. We start filling the tables, for $\boldsymbol{t}_1$ we obtain:

$\boldsymbol{t}_1$	h_1	h_2		h_{n-1}	h_n	h_{n+1}
10000	0	0	...	0	0	0
11000	0	0	...	0	0	0
11100	1	1	...	1	1	1
11110	?	1	...	1	1	1

As usual, in order to fill the table we used Remark 2.28 and Lemma 2.33.
For $\boldsymbol{t}_2$:

$\boldsymbol{t}_2$	h_1	h_2		h_{n-1}	h_n	h_{n+1}
10000	0	0	...	0	0	0
11000	0	0	...	0	0	0
11100	?	1	...	1	1	1
11110	?	?	...	1	1	1

Filling the tables in this manner, for $\boldsymbol{t}_{n-1}$ we have:

$\boldsymbol{t}_{n-1}$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h_{n+1}
10000	0	0	...	0	0	0	0
11000	0	0	...	0	0	0	0
11100	?	?	...	?	1	1	1
11110	?	?	...	?	?	1	1

And finally for $\boldsymbol{t}_n$:

$\boldsymbol{t}_n$	h_1	h_2		h_{n-2}	h_{n-1}	h_n	h_{n+1}
10000	0	0	...	0	0	0	0
11000	0	0	...	0	0	0	0
11100	?	?	...	?	?	1	1
11110	?	?	...	?	?	?	1

Clearly, multi-sorted operation $\boldsymbol{t}_n$ doesn't satisfy the identity $t_n(xxxyy) = t_n(yyyyx)$, for example because $h_{n+1}(xxxyy) = h_{n+1}(xxxxy) = x$. Therefore there are no such operations in $\mathcal{A}_{n+1}$, and, consequently, there does not exist a minion homomorphism from $\mathcal{D}_n$ to $\mathcal{A}_{n+1}$.

□

Lemma 2.47. *$\mathcal{B}_n \not\leq \mathcal{A}_{n+1}$ for $n \in \mathbb{N}$.*

Proof. Again, we use identities which hold in $\mathcal{B}_n$, but not in $\mathcal{A}_{n+1}$. As in the previous examples, we use identities we used in Lemma 2.36 to prove that $\mathcal{B}_n$ was a core, namely the following identities:

$$\begin{aligned}
t_1(xxxyy) &= x \\
t_2(xxxyy) &= t_1(xxxxy) \\
t_3(xxxyy) &= t_2(xxxxy) \\
&\vdots \\
t_n(xxxyy) &= t_{n-1}(xxxxy) \\
t_n(yyyyx) &= t_n(xxxxy) \\
t_i(x_1x_2x_3x_4x_5) &= t_i(x_{\sigma(1)}x_{\sigma(2)}x_{\sigma(3)}x_{\sigma(4)}x_{\sigma(5)}) \quad \forall i \in [n]\ \forall \sigma \in S_5
\end{aligned}$$

We have shown in Lemma 2.36 that in $\mathcal{B}_n$ there exist multi-sorted operations which satisfy the identities. For contradiction assume that there exist such operations $\boldsymbol{t}_1, \dots, \boldsymbol{t}_n$ in $\mathcal{A}_{n+1}$. We start filling the tables for them. Using the first identity and then Remark 2.28 for $\boldsymbol{t}_1$ we obtain:

$\boldsymbol{t}_1$	h_1	h_2	h_3		h_{n-1}	h_n	h_{n+1}
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	1	1	1	...	1	1	1
11110	?	1	1	...	1	1	1

Using the second identity we get the following table for $\boldsymbol{t}_2$:

$\boldsymbol{t}_1$	h_1	h_2	h_3		h_{n-1}	h_n	h_{n+1}
10000	0	0	0	...	0	0	0
11000	0	0	0	...	0	0	0
11100	?	1	1	...	1	1	1
11110	?	?	1	...	1	1	1

Continuing in this manner, for $\boldsymbol{t}_{n-1}$ we have:

$\boldsymbol{t}_{n-1}$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n	h_{n+1}
10000	0	0	0	...	0	0	0	0
11000	0	0	0	...	0	0	0	0
11100	?	?	?	...	?	1	1	1
11110	?	?	?	...	?	?	1	1

And for $\boldsymbol{t}_n$:

$\boldsymbol{t}_n$	h_1	h_2	h_3		h_{n-2}	h_{n-1}	h_n	h_{n+1}
10000	0	0	0	...	0	0	0	0
11000	0	0	0	...	0	0	0	0
11100	?	?	?	...	?	?	1	1
11110	?	?	?	...	?	?	?	1

The table for $\boldsymbol{t}_n$ contradicts the last identity as from the table $h_{n+1}(xxxxy) = x$, which is not symmetric. Therefore there are no such terms in $\mathcal{A}_{n+1}$ and there is no minion homomorphism from $\mathcal{B}_n$ to $\mathcal{A}_{n+1}$.

□

Lemma 2.48. *$\mathcal{Y} \not\leq \mathcal{W}$.*

Proof. We use the same identities we used to prove that $\mathcal{Y}$ is a minion core, namely:

$$t(xxxxy) = t(xxyyy)$$
$$t(x_1, x_2, x_3, x_4, x_5) = t(x_{\sigma(1)}x_{\sigma(2)}x_{\sigma(3)}x_{\sigma(4)}x_{\sigma(5)}) \quad \forall \sigma \in S_5$$

Assume that $\boldsymbol{t}$ is a 1-sorted operation in $\mathcal{W}$ which satisfies these identities. The only monotone operation h which satisfies $h = h^d$ has the following table:

$\boldsymbol{t}$	h
10000	0
11000	0
11100	1
11110	1

Clearly, it doesn't satisfy the identity $t(xxxxy) = t(xxyyy)$ as $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}) = x$ and $\boldsymbol{t} \circ (\boldsymbol{x}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}, \boldsymbol{y}) = y$. □

Proof of Theorem 2.41. The inequalities were established in the previous section.

Consider multi-sorted minions $\mathcal{M}$ and $\mathcal{N}$ appearing in the theorem such that $\mathcal{M} \leq \mathcal{N}$ does not follow from the established inequalities by the reflexivity or transitivity of $\leq$. We need to show that $\mathcal{M} \not\leq \mathcal{N}$. Assume, for a contradiction, that $\mathcal{M} \leq \mathcal{N}$. We distinguish cases according to $\mathcal{M}$.

- $\mathcal{M} = \mathcal{A}_n$. Since the inequality $\mathcal{A}_n \leq \mathcal{N}$ does not follow from the established ones, we have $\mathcal{N} \in \{\mathcal{X}, \mathcal{Y}, \mathcal{W}, \mathcal{A}_{n+1}, \mathcal{A}_{n+2}, \dots, \mathcal{B}_n, \mathcal{B}_{n+1}, \dots, \mathcal{C}_1, \dots, \mathcal{D}_1, \dots\}$. From the established inequalities and $\mathcal{A}_n \leq \mathcal{N}$ it then follows that $\mathcal{A}_n \leq \mathcal{B}_n$ or $\mathcal{A}_n \leq \mathcal{C}_1$. If $\mathcal{A}_n \leq \mathcal{B}_n$, then if we compose minion homomorphisms from $\mathcal{C}_n$ to $\mathcal{A}_n$ and from $\mathcal{A}_n$ to $\mathcal{B}_n$, we obtain a minion homomorphism from $\mathcal{C}_n$ to $\mathcal{B}_n$, which is a contradiction with Lemma 2.44. If $\mathcal{A}_n \leq \mathcal{C}_1$, then if we compose minion homorphisms from $\mathcal{X}$ to $\mathcal{A}_n$ and from $\mathcal{A}_n$ to $\mathcal{C}_1$, we obtain a minion homomorphism from $\mathcal{X}$ to $\mathcal{C}_1$, which contradicts Lemma 2.42.
- $\mathcal{M} = \mathcal{B}_n$. We have $\mathcal{N} \in \{\mathcal{X}, \mathcal{Y}, \mathcal{W}, \mathcal{A}_{n+1}, \mathcal{A}_{n+2}, \dots, \mathcal{B}_{n+1}, \mathcal{B}_{n+2}, \dots, \mathcal{C}_1, \dots, \mathcal{D}_1, \dots\}$. From the established inequalities and $\mathcal{B}_n \leq \mathcal{N}$ it then follows that $\mathcal{B}_n \leq \mathcal{A}_{n+1}$ or $\mathcal{B}_n \leq \mathcal{C}_1$. The first case contradicts Lemma 2.47. The second case, if we similarly as before compose minion homorphisms from $\mathcal{X}$ to $\mathcal{B}_n$ and from $\mathcal{B}_n$ to $\mathcal{C}_1$, contradicts Lemma 2.42.
- $\mathcal{M} = \mathcal{C}_n$. We have $\mathcal{N} \in \{\mathcal{X}, \mathcal{Y}, \mathcal{W}, \mathcal{A}_{n+1}, \dots, \mathcal{B}_n, \dots, \mathcal{C}_{n+1}, \dots, \mathcal{D}_n, \dots\}$. From the established inequalities and $\mathcal{C}_n \leq \mathcal{N}$ it then follows that $\mathcal{C}_n \leq \mathcal{B}_n$, which contradicts Lemma 2.44.
- $\mathcal{M} = \mathcal{D}_n$. We have $\mathcal{N} \in \{\mathcal{X}, \mathcal{Y}, \mathcal{W}, \mathcal{A}_{n+1}, \dots, \mathcal{B}_{n+1}, \dots, \mathcal{C}_{n+1}, \dots, \mathcal{D}_{n+1}, \dots\}$. From the established inequalities and $\mathcal{D}_n \leq \mathcal{N}$ it then follows that $\mathcal{D}_n \leq \mathcal{A}_{n+1}$, which contradicts Lemma 2.46.
- $\mathcal{M} = \mathcal{X}$. We have $\mathcal{N} \in \{\mathcal{Y}, \mathcal{W}\}$, it then follows that $\mathcal{X} \leq \mathcal{Y}$, which contradicts Lemma 2.42.
- $\mathcal{M} = \mathcal{Y}$. We have $\mathcal{Y} \leq \mathcal{W}$, which contradicts Lemma 2.48.
- $\mathcal{M} = \mathcal{T}$. We have $\mathcal{T} \leq \mathcal{A}_1$. $\mathcal{T}$ contains a unary operation satisfying the identity $t(x) = t(y)$ whereas $\mathcal{A}_1$ clearly doesn't contain such an operation.

□

2.6 Summary

Theorem 2.24 shows that every multi-sorted Boolean clone of the form $\mathrm{Pol}(\theta)$, where θ is a set of multi-sorted binary relations, is equivalent to one of the multi-sorted minions $\mathcal{X}, \mathcal{Y}, \mathcal{W}, \mathcal{A}_n, \mathcal{B}_n, \mathcal{C}_n, \mathcal{D}_n$, or $\mathcal{T}$. Theorem 2.41 then shows that the ordering between these multi-sorted minions is exactly as in Figure 2.1, so the ordering of the original multi-sorted Boolean clones is the same. We have additionally proved that the multi-sorted minions are minion cores in Theorem 2.25.

The diagram below summarizes our results and illustrates the preordering of multi-sorted Boolean clones. Each rectangle in the diagram corresponds to a minion, accompanied by its respective name and description.

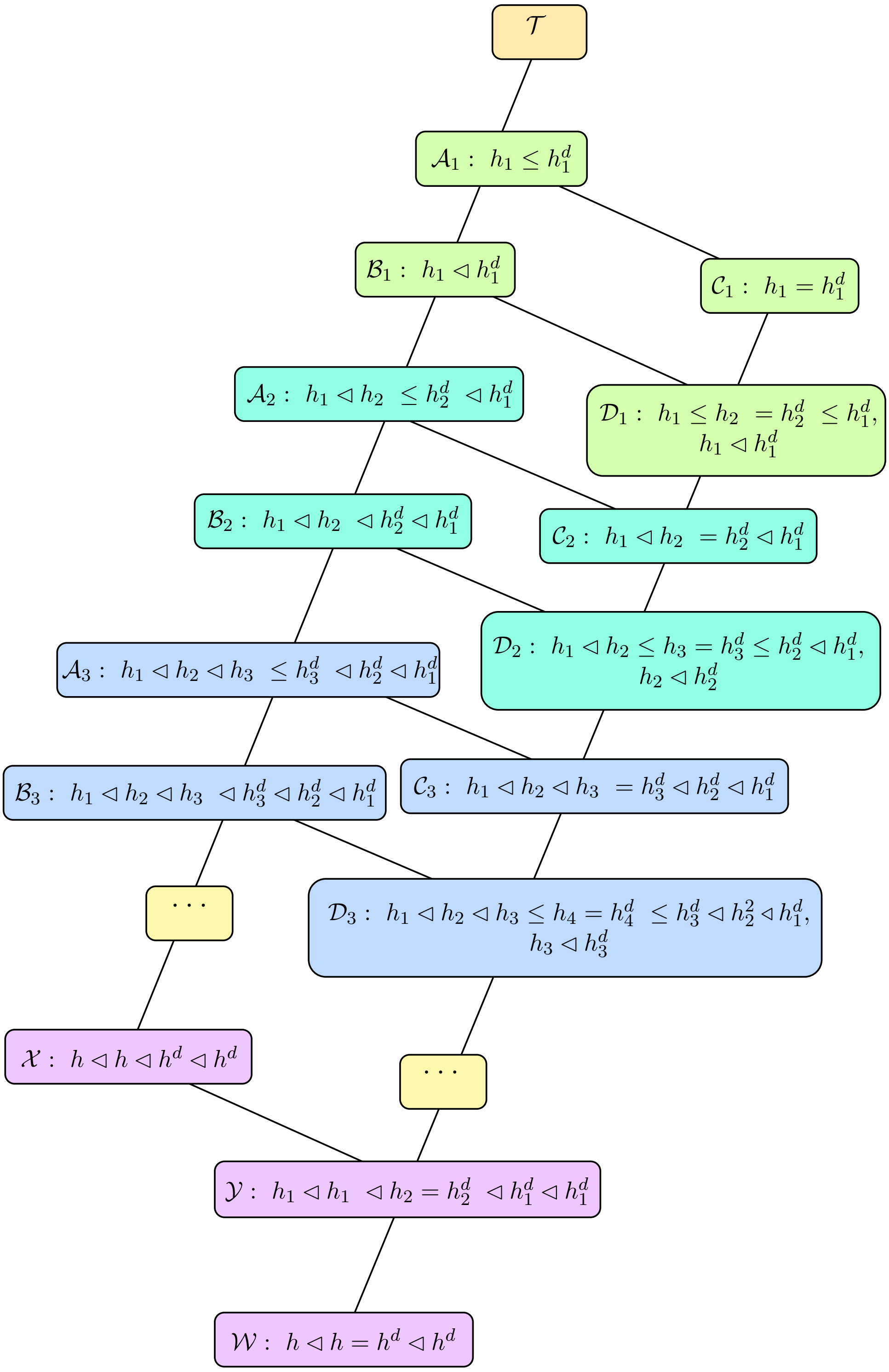


Figure 2.1: Diagram illustrating the preordering of multi-sorted Boolean clones determined by binary relations.

Conclusion

The thesis aimed to investigate the preordering of Boolean multisorted clones determined by binary multisorted relations. The most important results are presented through Theorem 2.24, Theorem 2.25, and Theorem 2.41.

In Theorem 2.24, we established that each clone falls into one of the following multisorted minions: $\mathcal{A}_n$, $\mathcal{B}_n$, $\mathcal{C}_n$, $\mathcal{D}_n$, $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{W}$, $\mathcal{T}$ (with n being a natural number). Furthermore, in Theorem 2.25, we demonstrated that these minions are minion cores. Building upon these findings, in Theorem 2.41, we ordered these minion cores. As a consequence, we obtained an ordering of equivalence classes Boolean multisorted clones determined by unary or binary multisorted relations.

The thesis presents its main result through diagram 2.1, which depicts equivalence classes of clones and their ordering.

In addition to the main result, the thesis opens up opportunities for further research in two distinct directions. The first direction is to characterize multi-sorted Boolean *minions* determined by pairs of unary or binary relations; examples of such minions are the minion cores we found. Are there any others? The second direction is to stay within multi-sorted clones but consider relations of arity greater than two.

Bibliography

[1] Libor Barto, Jakub Opršal, and Michael Pinsker. The wonderland of reflections, 2017.

[2] Emil L. Post. *The Two-Valued Iterative Systems of Mathematical Logic.* Princeton University Press, Princeton, N. J.,,, 1941.

[3] Manuel Bodirsky and Albert Vucaj. Two-element structures modulo primitive positive constructability. *Algebra Universalis*, 81(2):Paper No. 20, 17, 2020.

[4] Clifford H. Bergman. *Universal Algebra: Fundamentals and selected topics.* CRC Press, 2012.

[5] Libor Barto, Jakub Bulín, Andrei Krokhin, and Jakub Opršal. Algebraic approach to promise constraint satisfaction. *J. ACM*, 68(4):Art. 28, 66, 2021.

[6] Bengt Aspvall, Michael F. Plass, and Robert Endre Tarjan. A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Information Processing Letters*, 8(3):121–123, 1979.

List of Figures