Constraint Satisfaction Problem – Lecture Notes

Alexandr Kazda, Miklós Maróti

June 20, 2017

Chapter 1

Preface

This is the collection of notes on the Constraint Satisfaction Problem as it was taught at MFF UK in Autumn 2007 by Miklós Maróti. It is still work in progress. The authors would be grateful for any pointers regarding the origins of various theorems and lemmas.

While the constraint satisfaction problem (CSP) can be stated entirely in the language of graph theory, it turns out that algebraic approach has numerous advantages. The goal of these notes is to present the basics of the algebraic view of CSP, enabling the reader to understand modern articles on CSP and independently work in this area of mathematics. However, the ideas and constructions presented here can be useful even if the reader does not intend to become a CSP specialist: CSP brings together graph theory, complexity theory and universal algebra and can be used as a good motivation for study of either of these disciplines.

We assume that the reader has a general mathematical background, but there are no explicit prerequisites, as the text aims at self-containment. We will often be using algebraic tools and objects, mostly from universal algebra. Previous knowledge of this subject is beneficial, although not necessary.

1.1 Basics of complexity theory

In this section we shall formally define what a *problem* is, introduce the P and NP classes of problems and give examples of NP-complete problems.

Our universe shall be the set I of all binary words $\{0,1\}^* = \bigcup_{n=0}^{\infty} \{0,1\}^n$. For our purposes will be sufficient to use the intuitive meaning of the word "algorithm"[1].

Definition 1.1.1. A function $f : I \to I$ is computable in polynomial time (we say that f is in the class P) if there exists an algorithm A and constants c and

[1] If we wanted to be precise we can say that an algorithm is a computer program (that can use infinite amount of memory) or a Turing machine, but any reasonable formal definition will do.

d such that for any $x \in I$ the algorithm A stops in at most $|x|^d + c$ steps and computes $f(x)$.

There are many ways how to encode various objects or finite tuples of objects as words from $\{0,1\}^*$. As a most obvious example, we can use binary representation to encode numbers.

Remark 1.1.2. Thanks to the polynomial boundary, we don't have to concern ourselves with the details about encodings – even suboptimal encodings are fine, as long as they run in polynomial time.

Examples 1.1.3. The following functions and algorithms are in P:

- Basic arithmetic functions
- Euclid's algorithm
- Primality testing (algorithm known since 2004)
- Factoring polynomials in $\mathbb{Q}[x]$
- Hereditary graph properties (i.e. the ones closed under vertex removal and edge contraction)

In a *classification problem* C we are given an object x and are to decide whether $x \in C$. Here $C \subset I$ describes the problem. In complexity theory, our aim is to measure the complexity of the characteristic function of the set C for various C's. Obviously, the set of all classification problems is $P(I) = 2^I$.

Definition 1.1.4. A set $C \subset I$ of objects is in P if its characteristic function is in P.

Definition 1.1.5. A set $C \subset I$ is in NP (i.e. decidable in nondeterministic polynomial time) if there exists a function f in P and a constants k, l such that:

- If $x \in C$ then there exits a $y \in I, |y| \leq |x|^k + l$ such that $f(x,y) = 1$.
- If $x \notin C$ then $\forall y \in I$ it is $f(x,y) = 0$.

Here y is called a *short proof* and f is the *verifier*.

This definition says that C is in NP if for any $x \in C$ there is a witness for a short proof of $x \in C$ and that such witness never lies. Obviously, $P \subset NP$.

Examples 1.1.6. The following problems belong to NP (along with a great number of others):

- Composite number test (also in P because primality testing is in P).
- Solving $Ax^2 + By + C = 0$ in $\mathbb{N}$. (This equation has a solution iff it has a solution whose length is small in the lengths of A, B and C.)
- 3-colourability of a given graph.

- Graph isomorphism problem: are two given graphs isomorphic?
- The set of all theorems whose proofs (in an appropriate formal notation) are at most ten thousand times as long as the theorem itself.

The following conjecture corresponds to experience of several generations of computer scientists and makes study of the relationship between P and NP classes meaningful. However, no proof of this statement has been found yet.

Conjecture 1.1.7. It is $P \neq NP$.

Definition 1.1.8. Let $C, D \subset I$. Then C is *poly-time reducible* to D (denoted by $C \leq D$) iff there exists a poly-time computable function $f : I \to I$ such that $x \in C \iff f(x) \in D$. Two problems C, D are *poly-time equivalent* (denoted by $C \equiv D$) if $C \leq D, D \leq C$.

Proposition 1.1.9. *The relation $\leq$ is a quasi-order on $P(I)$. It is transitive and reflexive but it is possible that $C \leq D, D \leq C$ and $C \neq D$.*

Example 1.1.10. In P there are three classes of poly-time equivalent problems $\emptyset, I$ and $P \setminus \{\emptyset, I\}$.

Definition 1.1.11. Let $\mathcal{C} \subset P(I)$ be a class of problems and $D \subset I$ a problem. Then D is $\mathcal{C}$-hard if $\forall C \in \mathcal{C}\, C \leq D$.

Obviously, if D is $\mathcal{C}$-hard, then D is at least as hard as any problem in $\mathcal{C}$. We say that a problem D is *NP-complete* if it is NP-hard and in NP. There exist numerous NP-complete problems, we present here the famous SAT.

Definition 1.1.12. The SAT problem consists of all satisfiable Boolean formulas (i.e. formulas using the language $\wedge, \vee, \neg$ and variable symbols).

Theorem 1.1.13 (Cook, Levin 1971–73)**.** *SAT is NP-complete.*

Proof. (sketch of) In the proof we will have to use the Turing machine model of computation. Turing machine is an automaton with finitely many internal states that is equipped with a head. This head can read and write numbers on an (infinite) input tape. In each step a Turing machine reads the symbol under its head and then decides what to write on the tape, into which internal state to change and whether it should move the head to the left or right. While this model might seem simple, it is powerful enough to emulate any computer program.

Let $C \in NP$, let f be its verifier and x be an object. We want some g such that $g(x) \in SAT$ iff $x \in C$. When $f(x, y)$ is computed, the Turing machine M for $f(x, y)$ stops in less than $|x|^k + l$ steps, where l, k depend only on f. Thus we can assume that there are at most $m = |x|^k + l$ states of the system "M plus tape". We can encode these states by binary words $s_1, \ldots, s_m$.

Now s_{i+1} clearly depends only on s_i and x, y. If we encode states in a suitable way then there exists a (poly-length) Boolean formula $g(z)$ that, given z encoding of y (where $|y| \leq m$) and $s_1, \ldots, s_m$, checks whether $s_1, \ldots, s_m$ is a valid computation and $f(x,y)=1$ was reached. Thus $g(z)$ is satisfiable iff there exists y such that $f(x,y)=1$ for some $|y|\leq m$. But this is precisely the condition for $x\in C$ and the reduction is complete. □

Examples 1.1.14. Other NP-complete problems.

- *3-SAT* the set of all satisfiable formulas of the form $\bigwedge_{i=1}^{k} C_i$ where C_i is a disjunction of three variables or negations of variables (e.g. $C_i = \neg x_\alpha \vee x_\beta \vee \neg x_\gamma$).
- The set of all 3-colourable graphs.

Definition 1.1.15. SysEq($\mathcal{L}$) is the class of systems of equations over some fixed language $\mathcal{L}$ that are simultaneously satisfied.

Proposition 1.1.16. *Every SysEq($\mathcal{L}$) is poly-time equivalent to a SysEq($\mathcal{L}'$) in which every equation contains exactly one operation symbol.*

Proof. Let us first provide an example showing the idea of the proof.

Example 1.1.17. Consider the equation $(x\wedge y)\vee z = u$. Putting $x\wedge y = t$, we can rewrite the equation as:

$$\begin{aligned} x\wedge y &= t \\ t\vee z &= u. \end{aligned}$$

In the general case, each equation is of the form $l = r$ where l, r are *terms* – expressions formed by applying (finitely many times) operation symbols from $\mathcal{L}$ on some set of variables. For example x, $\neg x$, $(y\wedge x)\vee\neg z$ are terms in the language of Boolean algebras.

The equation $l = r$ can be rewritten as a system of two equations $l = y, r = y$, where y is a new variable. We can thus assume that the equation system contains only equations of the form $l = y$ where l is a term and y a variable.

For each equation, we use induction to simplify the term. If l itself is an operation or a variable then we are done. Otherwise, $l = t(s_1,\ldots,s_m)$ where t is an m-ary operation and $s_1,\ldots,s_m$ are terms. But then we can rewrite our equation as

$$\begin{aligned} l &= t(x_1,\ldots,x_m) \\ x_1 &= s_1 \\ x_2 &= s_2 \\ &\vdots \\ x_m &= s_m, \end{aligned}$$

where x_i's are new variables. We can now simplify the terms $s_1,\ldots,s_m$ in the same way and because each term consists of only finitely many operation symbols, we will end up with a system of equations that each contain at most one operation symbol. □

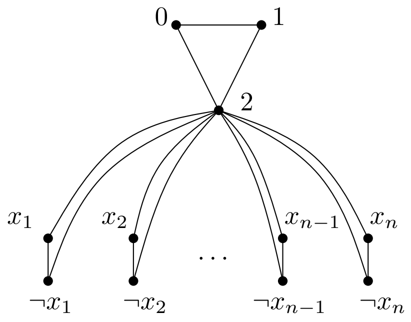

Figure 1.1: Step 1 of the reduction to 3-colouring

As a corollary, we get that SAT can be reduced to a system of equations of the form $x \vee y = z$ or $x \wedge y = z$ resp. $\neg x = y$. The first equation can be represented by a set of disjunctions

$$\begin{array}{rcl} x \vee y & \vee & \neg z \\ \neg x & \vee & z \\ \neg y & \vee & z, \end{array}$$

the second one by

$$\begin{array}{rcl} \neg x \vee \neg y & \vee & z \\ x & \vee & \neg z \\ y & \vee & \neg z \end{array}$$

and the third one as

$$\begin{array}{rcl} x & \vee & y \\ \neg x & \vee & \neg y. \end{array}$$

We obtain that SAT can be reduced (in polynomial time, as the algorithms used are quite fast) to 3-SAT[2] and 3-SAT is thus NP-complete.

Theorem 1.1.18. *3-colourability of graphs is NP-complete.*

Proof. We reduce 3-SAT to 3-colouring of a suitable graph. Let us have a 3-SAT formula with variables $x_1, \dots, x_n$. First draw the graph in Figure 1.1: Now for each $x \vee y \vee z$ glue to our graph the graph of the “gadget” seen in Figure 1.1: Here the inner five vertices are new and we glue the vertices marked $x, y, z, 0$ to the vertices $x, y, z, 0$ of the original graph (notice that it can be $x = \neg x_i$). If we do this for every $x \vee y \vee z$, we obtain a graph that is 3-colourable iff our formula is satisfiable. □

[2] To satisfy the formal requirement, we can rewrite two variable disjunction to three variables as for example $x \vee x \vee y$.

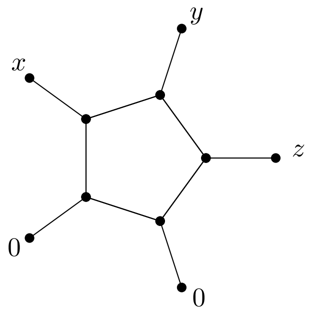

Figure 1.2: Step 2 of the reduction to 3-colouring

Exercise 1.1.19. Verify the last sentence of the above proof.

Theorem 1.1.20 (Ladner, 1975)**.** *If $P \neq NP$ then the NP problem class factored by poly-time equivalence has infinitely many blocks between P and NP.*

1.2 CSP for relational structures

A *relation* R on the set A is a subset $R \subset A^n$. We call n the *arity* of R.

Definition 1.2.1. Let $\mathcal{R}$ be a finite set of relation symbols with associated arities of these symbols. Then call $\mathcal{R}$ a *similarity type* and $\mathbb{A} = (A; \mathcal{R})$ is a *relational structure* of type $\mathcal{R}$ if A is a set and for each $R \in \mathcal{R}$ symbol of arity $n \in \mathbb{N}$ there exists an associated relation $R^{\mathbb{A}} \subseteq A^n$. We shall sometimes use the notation $\mathbb{A} = (A; \mathcal{R}^{\mathbb{A}})$ to avoid confusion as to which relations are we talking about.

Examples 1.2.2.

- Directed graphs $(V; E)$ with $E \subseteq V \times V$.
- 4-coloured set $(A; B, Y)$ with $B, Y \subseteq A$.

Definition 1.2.3. Two relational structures $\mathbb{A}, \mathbb{B}$ are *similar* if they have the same set of symbols and arities (ie. the same type). If $\mathbb{A}, \mathbb{B}$ are similar relational structures then $f : A \to B$ is a homomorphism if $\forall R \in \mathcal{R}$, $(a_1, \ldots, a_n) \in R^{\mathbb{A}} \Rightarrow (f(a_1), \ldots, f(a_n)) \in R^{\mathbb{B}}$.

If $\mathbb{A} = (A, \mathcal{R}^{\mathbb{A}})$ is a relational structure, then $\mathbb{B} = (B, \mathcal{R}^{\mathbb{B}})$ is called a *substructure* of $\mathbb{A}$ if $B \subseteq A$ and the inclusion mapping $B \to A$ is a homomorphism. If for all k-ary R's it is $R^{\mathbb{B}} = R^{\mathbb{A}} \cap B^k$ then we call $\mathbb{B}$ the *substructure of* $\mathbb{A}$ *induced by the set* B. In the case of graphs, we obtain the familiar notions of subgraph and subgraph induced by a set of vertices.

Definition 1.2.4. A homomorphism $f : \mathbb{A} \to \mathbb{B}$ is called

- *isomorphism* if there exists an inverse homomorphism $\mathbb{B} \to \mathbb{A}$

- *endomorphism* if $\mathbb{A} = \mathbb{B}$
- *automorphism* if it is an endomorphism and isomorphism.

Definition 1.2.5. Let $\mathbb{B} = (B, \mathcal{R})$ be a relational structure. Then define the *constraint satisfaction problem* of $\mathbb{B}$ as the set of relational structures

$$\mathrm{CSP}(\mathbb{B}) = \{\mathbb{A} | \mathbb{A} \text{ is similar to } \mathbb{B} \text{ and there exists a homomorphism } f : \mathbb{A} \to \mathbb{B}\}.$$

In usual encodings, it is easy to check whether given string encodes a relational structure similar to $\mathbb{B}$. The hard question is whether there exists a homomorphism $\mathbb{A} \to \mathbb{B}$.

Proposition 1.2.6. CSP($\mathbb{B}$) *is in NP for all* $\mathbb{B}$.

Proof. Any mapping $\mathbb{A} \to \mathbb{B}$ can be encoded by a string whose length is linear in $|\mathbb{A}|$ and verifying that given $f : \mathbb{A} \to \mathbb{B}$ is a homomorphism can be done in polynomial time. Thus CSP($\mathbb{B}$) is in NP. □

One of main topics of this course will be various approaches used to prove (or disprove) the following conjecture about *dichotomy* of CSP:

Conjecture 1.2.7 (Feder, Vardi, 1998)**.** For every $\mathbb{B}$ relational structure, the problem CSP($\mathbb{B}$) either lies in P or is NP-complete.

Example 1.2.8. CSP($\mathbb{B}$) for $\mathbb{B}$ being a triangle ($B = \{1, 2, 3\}, E = B^2 \setminus \{(i, i) : i \in V\}$) is precisely the 3-colouring problem and thus is NP-complete.

Example 1.2.9. The 3-SAT can be refolmulated in the language of CSP, just let $\mathbb{B} = \{0, 1\}$ and $\mathcal{R} = \{S_{\alpha\beta\gamma} | \alpha, \beta, \gamma \in \{0, 1\}\}$ where $S_{\alpha\beta\gamma} = \{0, 1\}^3 \setminus \{\alpha, \beta, \gamma\}$.

Definition 1.2.10. Define a partial ordering "$\to$" on the class of all similar relational structures by $\mathbb{A} \to \mathbb{B}$ iff there exists a homomorphism from $\mathbb{A}$ to $\mathbb{B}$.

Obviously, "$\to$" induces an equivalence on the set of all relational structures. Denote this equivalence by $\leftrightarrow$.

Theorem 1.2.11. *Let* $\mathbb{C}, \mathbb{D}$ *be similar relational structures and* $\mathbb{C} \leftrightarrow \mathbb{D}$*. Then* CSP($\mathbb{C}$) = CSP($\mathbb{D}$)*. Moreover* $\mathbb{B}_1 = (A, \textit{full relations})$ *is the maximal and* $\mathbb{B}_0 = (A, \emptyset)$ *is the minimal element in this ordering (see Figure 1.2).*

Proof. If we have a homomorphism $\mathbb{A} \to \mathbb{C}$ then we can compose it with a homomorphism $\mathbb{C} \to \mathbb{D}$ to obtain a homomorphism $\mathbb{A} \to \mathbb{D}$ and vice versa.

The second part of the theorem is an easy exercise. □

Remark 1.2.12. If there is a homomorphism $f : \mathbb{B} \to \mathbb{C}$ such that $\mathbb{C} \subset \mathbb{B}$ (we call such $\mathbb{C}$ a *retract* of $\mathbb{B}$) then CSP($\mathbb{B}$) = CSP($\mathbb{C}$). This follows from the previous theorem because inclusion is a homomorphism.

It is natural to ask what is the smallest substructure of $\mathbb{B}$ that still has interesting CSP. In the following, we introduce the notion of a core that is precisely such structure.

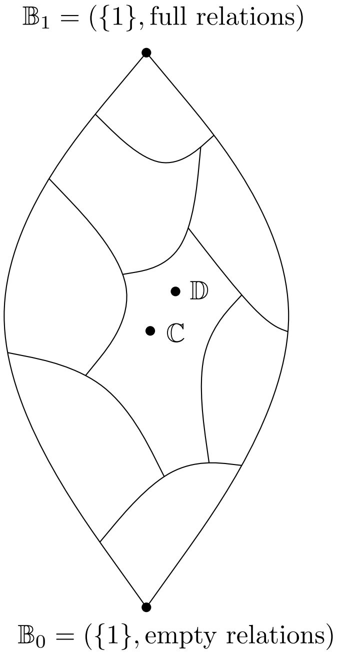
$\mathbb{B}_1 = (\{1\}, \text{full relations})$
$\mathbb{D}$
$\mathbb{C}$
$\mathbb{B}_0 = (\{1\}, \text{empty relations})$

Figure 1.3: A sketch of the lattice of relational structures on 0 and 1

Definition 1.2.13. A relational structure $\mathbb{B}$ is a *core* if all of is endomorphisms are automorphisms.

Theorem 1.2.14. *Every $\leftrightarrow$ block in the set F of all finite structures of the same similarity type contains (up to isomorphism) a uniquely determined core.*

Proof. Take a structure $\mathbb{B}$ in F of minimal $|\mathbb{B}| = |B|$. We claim that $\mathbb{B}$ is a core. Let $f : \mathbb{B} \to \mathbb{B}$ be an endomorphism, denote by $\mathbb{B}'$ the image $f(\mathbb{B})$ of $\mathbb{B}$. Because $\mathbb{B}' \subset \mathbb{B}$, we have $\mathbb{B} \leftrightarrow \mathbb{B}'$ as in the above remark and so $\mathbb{B}' \in F$. Thus $|\mathbb{B}'| = |\mathbb{B}|$ and f must be both injective and surjective. As we have only a finite number of tuples $(b_1, b_2, \ldots, b_n)$ and f induces a bijection of tuples, f^{-1} must be a homomorphism and so f is an isomorphism.

If now $\mathbb{B}, \mathbb{B}'$ are both cores, then obviously $|\mathbb{B}| = |\mathbb{B}'|$ are minimal. And thus any $f : \mathbb{B} \to \mathbb{B}'$ is a bijection. By the same argument as above, f must be also an isomorphism. □

1.2.1 Basic constructions

Lemma 1.2.15. *Let A be a finite set. Then there exists a $k \in \mathbb{N}$ such that for all f mappings $A \to A$ it is $f^{2k} = f^k$.*

Proof. Fix $x \in A$. Then the elements of the sequence $x, f(x), f^2(x), \ldots$ will eventually start to repeat themselves: It is $f^t(x) = f^s(x)$ for some $0 \le t < s \le |A|$ and thus $f^{t+v}(x) = f^{s+v}(x)$ for all $v \in \mathbb{N}$. Denote by p the period $s - t$ and by q the preperiod t. Notice that it is $0 \le p, q \le |A|$.

Let now $k = |A|!$. Obviously, $k \ge q$ and $p|k$. Putting $l = \frac{k}{p}$ we can write

$$f^{2k}(x) = f^{k+k}(x) = f^{k+lp}(x) = f^k(x),$$

proving the lemma. □

Theorem 1.2.16 ("We can add equalities."). *Let* $\mathbb{B} = (B, \mathcal{R})$ *and* $\mathbb{B}' = (B; \mathcal{R} \cup \{=_B\})$ *where* $=_B$ *is the relation* $\{(b, b) | b \in B\}$. *Then* CSP($\mathbb{B}$) *and* CSP($\mathbb{B}'$) *are poly-time equivalent.*

Proof. We need to find two reductions. The easy one is CSP($\mathbb{B}$) to CSP($\mathbb{B}'$): Given $\mathbb{A}$ similar to $\mathbb{B}$, we produce $\mathbb{A}' = (A; \mathcal{R}^{\mathbb{A}} \cup \{=_A\})$ where $=_A$ is empty. Then $\mathbb{A} \to \mathbb{B}$ iff $\mathbb{A}' \to \mathbb{B}'$ and so we have a reduction.

The other direction takes a little bit of effort: Take $\mathbb{A}' = (A', \mathcal{R}^{\mathbb{A}} \cup =_{A'})$ where $=_{A'}$ is any binary relation on A' (it need not be an equivalence). Let S be the equivalence relation generated by $=_{A'}$ (the transitive, reflexive and symmetric closure of $=_{A'}$).

For each equivalence class $[c]$ of S take one representative element c and let $A \subseteq A'$ be the set of these representative elements. Put $(c_1, \ldots, c_n) \in R^{\mathbb{A}}$ iff there exists $(a_1, \ldots, a_n) \in R^{\mathbb{A}'}$ such that $\forall i, a_i \in [c_i]$. We claim that then there exists a homomorphism $f : \mathbb{A}' \to \mathbb{B}'$ iff there exists a homomorphism $g : \mathbb{A} \to \mathbb{B}$. We prove both implications separately:

- ""$\Rightarrow$"" It is easy to see that if $a_i \in [c_i]$ then $(f(a_i), f(c_i)) \in =_B$ and so $f(a_i) = f(c_i)$. So the set $f([c])$ has one element and it makes sense to let $g(c) = f([c])$. If now $(a_1, \dots, a_n) \in R^{\mathbb{A}'}$ then $(f(c_1), \dots, f(c_n)) = (f(a_1), \dots, f(a_n)) \in R^{\mathbb{B}'} = R^{\mathbb{B}}$ and so g is a homomorphism $\mathbb{A} \to \mathbb{B}$.

- ""$\Leftarrow$"" If there exists a $g : \mathbb{A} \to \mathbb{B}$ homomorphism we can extend g to A' as $f(a) = g(c)$ for all $a \in [c]$. If $(a_1, \dots, a_n) \in R^{\mathbb{A}}$ then $(c_1, \dots, c_n) \in R^{\mathbb{A}'}$ and so $(f(a_1), \dots, f(a_n)) = (g(c_1), \dots, g(c_n)) \in R^{\mathbb{B}}$.

□

In the following we will use the notion of *primitive positive formula.* Given a class Ω of formulas we can produce a formula $\exists \alpha_1 \dots \exists \alpha_n,\ \phi_1(*, \dots, *) \wedge \dots \wedge \phi_n(*, \dots, *)$ where the stars are either free variables (say, $x_1, \dots, x_m$) or the variables $\alpha_1, \dots, \alpha_n$. The formulas ϕ_i all belong to the class Ω.

Example 1.2.17. An example of a primitive positive formula using $\Omega = \{<\}$ is the formula $\psi(y, z) = \exists x,\ x < y \wedge z < x$.

Motivation 1.2.18. When reducing SAT to 3-colourability we have used a certain subgraph as a tool for expressing logical statements in terms of colours. In that case, we have used the relation "u and v have different colours" to produce more complicated relations, such as $x \vee y \vee z$. We shall now use primitive positive formulas to perform similar feats with general relational structures.

Definition 1.2.19. For a set Γ of finitary[3] relations on a set A, define $\langle \Gamma \rangle$ as the set of all relations that can be expressed by primitive positive formulas using only Γ and "=". We say that Γ is a *relational clone* if $\Gamma = \langle \Gamma \rangle$.

Given a set X, a *closure operator on* X is a mapping $c : 2^X \to 2^X$ such that:

- $Y \subset c(Y)$ for all $Y \subset X$.

- $c^2 = c$.

- $Y \subset Z \Rightarrow c(Y) \subset c(Z)$ for all $Y, Z \subset X$.

A set Y such that $Y = c(Y)$ is called *closed.*

A *lattice* is any partially ordered set L such that for each $x, y \in L$ there exists $x \wedge y$ infimum and $x \vee y$ supremum of the pair. A lattice is *complete* if every set of its elements has both the supremum and the infimum.

Exercise 1.2.20. Let X be a set and c a closure operator on X. Then the set of closed subsets of X ordered by inclusion is a complete lattice. The supremum of a set $\mathcal{Y}$ can be computed as $c(\bigcup_{Y \in \mathcal{Y}} Y)$ and the infimum as $\bigcap_{Y \in \mathcal{Y}} c(Y)$.

Proposition 1.2.21. *$\langle \cdot \rangle$ is a closure operator on the set of sets of relations. Thus the set of relational clones is a complete lattice (with ordering given by inclusion).*

[3] A relation is finitary if it has finite arity.

Proof. It is an easy exercise to check that the conditions (i)–(iii) all hold for $\langle\cdot\rangle$. □

The following theorem allows us to easily find reductions between many kinds ofproblems.

Theorem 1.2.22. *If* $\langle\mathcal{R}_1\rangle \subset \langle\mathcal{R}_2\rangle$ *then* $\mathrm{CSP}(B,\mathcal{R}_1)$ *is poly-time reducible to* $\mathrm{CSP}(B,\mathcal{R}_2)$.

Proof. All we actually need is that $\mathcal{R}_1 \subset \langle\mathcal{R}_2\rangle$. That means that for every $R \in \mathcal{R}_1$ we have a primitive positive formula ψ_R using $\mathcal{R}_2$ equivalent to R. Each ψ_R can be written in the form:

$$\psi_R(x_1,\dots,x_n) = \exists u_1,\dots u_k, S_1(x_{\pi_{1,1}},\dots,x_{\pi_{1,n_1}},u_{\rho_{1,1}},\dots,u_{\rho_{1,k_1}}) \wedge S_2(\cdots) \wedge \dots \wedge S_m(\cdots)$$

for some $S_i \in \mathcal{R}_2$ and $n_i \le n, k_i \le k$ (this models the fact that all variables need not be present in all relations) and a suitable set of numbers $\pi_{i,j} \in \{1,\dots,n\}$. Do not fear triple indices; they are here only to show how we choose of variables from the set $\{x_1,\dots,x_n\}$. Assume for now that all S_i's are mutually different.

Consider the relational structure $\mathbb{G}$ of signature $\mathcal{R}$ with the underlying set $G = \{a_1,\dots,a_n,b_1,\dots,b_k\}$. The relations of $\mathbb{G}$ are trivial iff they are not one of S_i's, otherwise it is $S_i^{\mathbb{G}} = \{(a_{\pi_{i,1}},\dots,a_{\pi_{i,n_i}},b_{\rho_{i,1}},\dots,b_{\rho_{i,k_i}})\}$.

Observe now that any mapping $f : G \to B$ maps $(a_1,\dots,a_n)$ to an element in R iff $f : \mathbb{G} \to (B,\mathcal{R}_2)$ is a homomorphism: The right side is true iff for all $i = 1,\dots,m$ it is $(f(a_{\pi_{i,1}}),\dots,f(a_{\pi_{i,n_i}}),f(b_{\rho_{i,1}}),\dots,f(b_{\rho_{i,n_i}})) \in S_i$ which is precisely the condition $\psi_R(f(a_1),\dots,f(a_n))$.

Now return to our assumption that S_i's are mutually different. If this is not true, we can recover by taking $S_i^{\mathbb{G}}$ as the set of all tuples $(a_{\pi_{j,1}},\dots,a_{\pi_{j,n_j}},b_{\rho_{j,1}},\dots,b_{\rho_{j,k_j}})$ for j such that $S_j = S_i$. Now again f will be a homomorphism iff $(f(a_1),\dots,f(a_n)) \in R$.

For all $R \in \mathcal{R}_1$ we construct such structures $\mathbb{G}_R$. While such construction might in general take a long time, this time is not dependant upon the size of the input and thus adds only a constant to the time complexity of the algorithm.

Let us now have an instance $\mathbb{A} = (A,\mathcal{R}_1^{\mathbb{A}})$ of $\mathrm{CSP}(B,\mathcal{R}_1)$. We want to produce a reduction to an instance $\mathbb{A}'$ of $\mathrm{CSP}(B,\mathcal{R}_2)$.

The underlying set of $\mathbb{A}'$ will be $A \cup B$ where B is the set of additional elements. For every $(a_1,\dots,a_n) \in R^{\mathbb{A}} \in \mathcal{R}_1^{\mathbb{A}}$, we add the corresponding $b_1,\dots,b_k$ and use the elements $\{a_1,\dots,a_n,b_1,\dots,b_k\}$ to embed a copy of $\mathbb{G}_R$ into $\mathbb{A}'$. While elemnts of A might be shared among differenct $\mathbb{G}_R$'s, each time we find a new tuple $(a_1,\dots,a_n) \in R^{\mathbb{A}}$, we add brand new $b_1,\dots,b_k$. All this can be done in polynomial time, as we have only polynomially many n-tuples (and finitely many relations in $\mathcal{R}_1$).

It remains to observe that $f : \mathbb{A}' \to (B,\mathcal{R}_2)$ iff f restricted to each copy of $\mathbb{G}_R$ is a homomorphism iff $\forall R^{\mathbb{A}} \in \mathcal{R}_1^{\mathbb{A}}, \forall(a_1,\dots,a_n) \in R$ it is $(f(a_1),\dots,f(a_n)) \in R$ iff f restricted to A is a homomorphism $\mathbb{A} \to (B,\mathcal{R}_1)$. □

Example 1.2.23. We shall clarify the main points of the above construction by performing it for one concrete case. Consider two structures on $B = \{1,2,3\}$

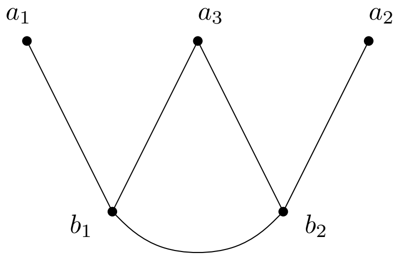

Figure 1.4: Gadget for R

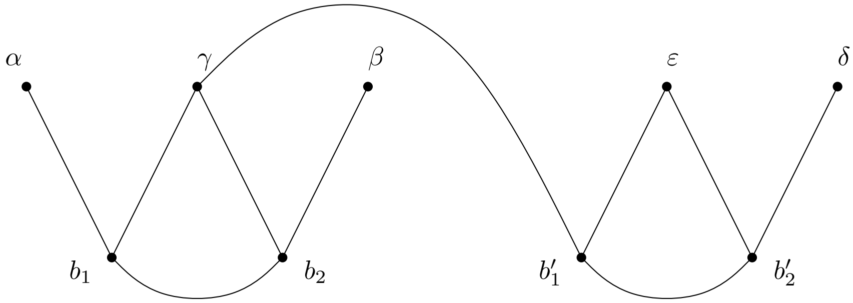

Figure 1.5: Instance of $\mathrm{CSP}(B, \mathcal{R}_2)$

given by $\mathcal{R}_1 = \{R\}$ and $\mathcal{R}_2 = \{\neq\}$ with $R = \{(x, y, z) : x = y \Rightarrow z = x\}$. Obviously, $\mathrm{CSP}(B, \mathcal{R}_2)$ is the problem of existence of a graph homomorphism to K_3. We want to show that $\mathrm{CSP}(B, \mathcal{R}_1)$ is poly-time reducible to $\mathrm{CSP}(B, \mathcal{R}_2)$. First notice that R can be rewritten as $\psi_R(x, y, z) = \exists u, v, x \neq u \wedge u \neq v \wedge y \neq v \wedge z \neq u \wedge z \neq v$ which is in $\langle\neq\rangle$. So, by the above theorem, there is a reduction. The graph $\mathbb{G}_R$ from the proof is depicted in Figure 1.2.23.

To show how the reduction works, we shall transform the instance $A = (\{\alpha, \beta, \gamma, \delta, \epsilon\}, \{(\alpha, \beta, \gamma), (\gamma, \delta, \epsilon)\})$ of $\mathrm{CSP}(B, \mathcal{R}_1)$ to an instance of $\mathrm{CSP}(B, \mathcal{R}_2)$. Consider the graph created by joining together two copies of $\mathbb{G}_R$ shown in Figure 1.2.23. It is not hard to see that homomorphisms from this graph to K_3 correspond to homomorphisms $A \to (B, \mathcal{R}_1)$ so we have the desired reduction.

Problem 1.2.24. Given two finite sets $\mathcal{R}_1, \mathcal{R}_2$, is it decidable whether $\langle\mathcal{R}_1\rangle \subset \langle\mathcal{R}_2\rangle$?

It is obviously enough to decide for all $R \in \mathcal{R}_1$ whether R belongs to $\mathcal{R}_2$. Let n be the arity of R. We shall show that $R \in \mathcal{R}_1$ can checked in finite (albeit long) time by checking all the primitive positive formulas from $\mathcal{R}_2$ that contain at most A^n new variables. Before we do that, however, let us dwelve a bit deeper into universal algebra.

Definition 1.2.25. Let R be a k-ary relation on the set A and f an n-ary operation on A. We say that R is an *invariant relation* under f or that f is a *polymorphism* of R if for every n-tuple of k-tuples $\{r_1, \dots, r_n\}$ (where $r_i = (a_{1i}, \dots, a_{ki})) \in R$ we have

$$(f(a_{11}, \dots, a_{n1}), \dots, f(a_{k1}, \dots, a_{kn})) \in R.$$

In the above situation, we will often use shorthand notation:

$$(f(r_1), \dots, f(r_n)) = (f(a_{11}, \dots, a_{n1}), \dots, f(a_{k1}, \dots, a_{kn}))$$

While this notation might be slightly unclear at first, it helps us avoid drowning in variables and indices. We will also sometimes write the above condition in the form of a table such a this one:

$$\begin{array}{llll} (a_{11}, & \dots, & a_{1k}) & \in R \\ \vdots & \vdots & \vdots & \\ (a_{n1}, & \dots, & a_{nk}) & \in R \\ \hline \multicolumn{4}{l}{(f(a_{11}, \dots, a_{n1}), \dots, f(a_{k1}, \dots, a_{kn})) \in R} \end{array}$$

Remember that in universal algebra, an *algebra* consists of the nonempty support set A together with a set $\mathcal{F}$ of finitary (ie. of finite arity) operations on A. We say that an algebra is *nontrivial* if $|A| \geq 2$. Groups, fields, boolean algebras and lattices are all examples of algebras. The *k-th power of algebra* $\underline{A}$ (denoted by $\underline{A}^n$) is an algebra with the support set A^k and all operations retained from $\underline{A}$, only acting coordinatewise.

Observation 1.2.26. *The k-ary relation R is a subalgebra of $(A, f)^k$ iff f is a polymorphism of R.*

Exercise 1.2.27. Show that any unary mapping f is a polymorphism of (A, R) iff it is an endomorphism of (A, R).

Exercise 1.2.28. What are the polymorphisms of the structure $(\{1, 2, 3\}, \{\neq\})$?

Definition 1.2.29. Let Γ be a set of relations on A, let

$$\mathrm{Pol}(\Gamma) = \{f : A^n \to A | f \text{ is a polymorphism of all } R \in \Gamma\}.$$

Definition 1.2.30. If Φ is a set of operations on A, let

$$\mathrm{Inv}(\Phi) = \{R \in A^n | R \text{ is invariant under all operations } f \in \Phi\}.$$

Remark 1.2.31. We know that an n-ary relation R is an invariant relation of an algebra $\underline{A}$ iff R is a subalgebra of $\underline{A}^n$. Using a notation common in universal algebra tools, we can write $\mathrm{Inv}(A) = \mathrm{S}\,\mathrm{P}_{fin}\,\underline{A}$ where the operator S stands for "subalgebras" and P_{fin} for "finite powers" of the given algebra or a set of algebras.

The operations Inv, Pol provide a connection between the lattice of sets of relations on A and the lattice of sets of finitary functions from A to A. The following observation formalises this notion of connection.

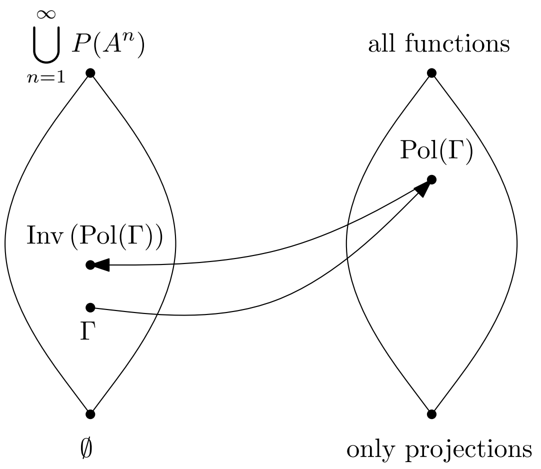

Figure 1.6: Galois correspondence

Observation 1.2.32. *The operations* Pol, Inv *form a* Galois connection*, that is:*

(i) $\Gamma \subset \Gamma' \Rightarrow \mathrm{Pol}(\Gamma') \subset \mathrm{Pol}(\Gamma)$

(ii) $\Phi \subset \Phi' \Rightarrow \mathrm{Inv}(\Phi') \subset \mathrm{Inv}(\Phi)$

(iii) $\Gamma \subset \mathrm{Inv}(\mathrm{Pol}(\Gamma))$

(iv) $\Phi \subset \mathrm{Pol}(\mathrm{Inv}(\Phi))$

Exercise 1.2.33. Prove using (i)–(iv) that $\mathrm{Pol}(\Gamma) = \mathrm{Pol}(\mathrm{Inv}(\mathrm{Pol}(\Gamma)))$ and $\mathrm{Inv}(\Phi) = \mathrm{Inv}(\mathrm{Pol}(\mathrm{Inv}(\Phi)))$.

Definition 1.2.34. A mapping $f : A^n \to A$ is a *projection* if $\exists i$ such that $f(a_1, \ldots, a_n) = a_i$ for all tuples $(a_1, \ldots, a_n) \in A^n$.

Definition 1.2.35. A set Φ of finitary operations on A is a *functional clone* iff it is closed under composition of operations and contains all projections $\pi : A^n \to A$. Any function formed by composing operations from Φ is called a *term*.

Theorem 1.2.36. *Let A, Γ be finite. Then* $\mathrm{Pol}(\Gamma)$ *is always a functional clone and* $\mathrm{Inv}(\Phi)$ *is always a relational clone. Moreover,* $\mathrm{Inv}\mathrm{Pol}(\Gamma)$ *is exactly the set of relations pp-defined from* Γ *and* $\mathrm{Pol}(\mathrm{Inv}(\Phi))$ *is exactly the clone of operations of* Φ*.*

Proof. It is easy to verify that $\mathrm{Pol}(\Gamma)$ is a functional clone. It is elementary to show that $\mathrm{Inv}(\Phi)$ is closed under pp-definitions, but we will skip this part and prove directly that $\mathrm{Inv}(\mathrm{Pol}(\Gamma)) = \langle\Gamma\rangle$. This means that $\mathrm{Inv}(\Phi) = \mathrm{Inv}(\mathrm{Pol}(\mathrm{Inv}(\Phi))) = \langle\mathrm{Inv}(\Phi)\rangle$, proving our proposition.

First notice that $\mathrm{Pol}(\Gamma) = \mathrm{Pol}(\langle\Gamma\rangle)$. The $\supseteq$ inclusion is obvious. To prove $\subseteq$, consider $f \in \mathrm{Pol}(\Gamma)$ and a relation $\psi(a_1, \ldots, a_k) = \exists b_1, \ldots, \exists b_m, \phi(a_1, a_2, \ldots, a_k, b_1, \ldots, b_m)$ where ϕ is a conjuction of relations from Γ. If now $r_1, \ldots, r_n$ are k-tuples such that $\psi(r_i)$ holds for each i then there exist $s_1, \ldots, s_n$ m-tuples such that $\phi(r_i, s_i)$ holds for each i. But because ϕ is a simple conjunction of relations from Γ and f is a polymorphism in Γ, we see that $\phi(f(r_1, \ldots, r_n), f(s_1, \ldots, s_n))$ holds. But then also $\psi(f(r_1, \ldots, r_n))$ and so f is a polymorphism in $\langle\Gamma\rangle$. This means that $\mathrm{Inv}(\mathrm{Pol}(\Gamma)) = \mathrm{Inv}(\mathrm{Pol}(\langle\Gamma\rangle) \supseteq \langle\Gamma\rangle$

Using the fact that $|A|$ is finite we prove that $\mathrm{Inv}(\mathrm{Pol}(\Gamma)) \subset \langle\Gamma\rangle$.

Consider $\mathbb{F}_n$ the set of functions $\{f(\pi_1, \ldots, \pi_n) : f \text{ polymorphism of } \Gamma\}$. Here π_i are projections $A^n \to A$ to the i-th element. Universal algebra students notice that this is the n-generated free algebra in the variety generated by $\mathbb{A}$, although we will not need this fact in our proof. Obviously, $\mathbb{F}_n \subset A^{A^n}$ so we can fix the order of elements in A^n and understand $\mathbb{F}_n$ as an $|A|^n$-ary relation. It is $(x_1, \ldots, x_{|A|^n}) \in \mathbb{F}_n$ iff there exists $f \in \mathbb{F}_n$ such that x_i is the i-th value of f. We will now prove that in this sense it is $\mathbb{F}_n \in \langle\Gamma\rangle$.

The condition "f is an n-ary polymorphism of Γ" can be rewritten as: "For all $R \in \Gamma$ k-ary relations and for all $M \in A^{n\times k}$ with rows in R it is $f(M) \in R$." (Notice the abuse of notation.) When A, Γ are finite, there exists a finite primitive positive formula that checks whether the function given by $(x_1, \ldots, x_{|A|^n})$ is a polymorphism: If n-tuples number $i_1, i_2, \ldots, i_k$ form a matrix M from the above condition, we ask whether $f(M) = (x_{i_1}, \ldots, x_{i_k}) \in R$. There are only finitely many such relations and sets of n-tuples so we can conjunct all these conditions into a primitve positive formula $\psi(x_1, \ldots, x_{|A|^n})$.

Let now $R \in \mathrm{Inv}(\mathrm{Pol}(\Gamma))$ be a k-ary relation. Then we can write $R = \{r_1, r_2, \ldots, r_m\}$ where r_i are k-tuples. We now claim that R is basically just $\mathbb{F}_m$ restricted to certain coordinates.

Let us now for each i write $r_i = (r_{i,1}, \ldots, r_{i,k})$ and let $j_1, j_2, \ldots, j_k$ be indices such that for each $f = (x_1, \ldots, x_{|A|^m})$ it is $f(r_{1,l}, \ldots, r_{m,l}) = x_{j_l}$. Then it is also $f(r_1, \ldots, r_m) = (x_{j_1}, x_{j_2}, \ldots, x_{j_k})$. Observe that for $f = \pi_i$ this means $(x_{j_1}, \ldots, x_{j_k}) = r_i$. If now $f = (x_1, \ldots, x_{|A|^m}) \in \mathbb{F}_m$ then because f is an R-polymorphism it is $(x_{j_1}, x_{j_2}, \ldots, x_{j_k}) = f(r_1, \ldots, r_m) \in R$. We conclude that the set of homomorphisms $\mathbb{F}_m$ limited to coordinates $j_1, \ldots, j_k$ is precisely R (because limited $\mathbb{F}_m$ contains all the elements r_i and nothing else). We can easily describe the coordinate limitation using a primitve positive formula: Let ψ be a formula for $\mathbb{F}_m$. For the sake of readability (and without loss of generality) let $j_l = l$ for $l = 1, 2, \ldots, k$. Then the formula for R will be

$$\phi(a_1, \ldots, a_k) = \exists u_1, \ldots, \exists u_{m-k}\psi(a_1, \ldots, a_k, u_1, \ldots, u_{m-k}).$$

This means that $R \in \langle\Gamma\rangle$ and so the proof is complete. See Figure 1.2.1 for an illustration of the last step of the proof.

Finally, to show that $\mathrm{Pol}(\mathrm{Inv}(\Phi))$ is exactly the clone of operations of Φ, note that $\mathrm{Inv}(\Phi)$ contains for each k the $|A|^k$-ary relation

$$F_k = \{(t(\overline{a}))_{\overline{a}\in A^k} : t \text{ is in the clone of } \Phi\}.$$

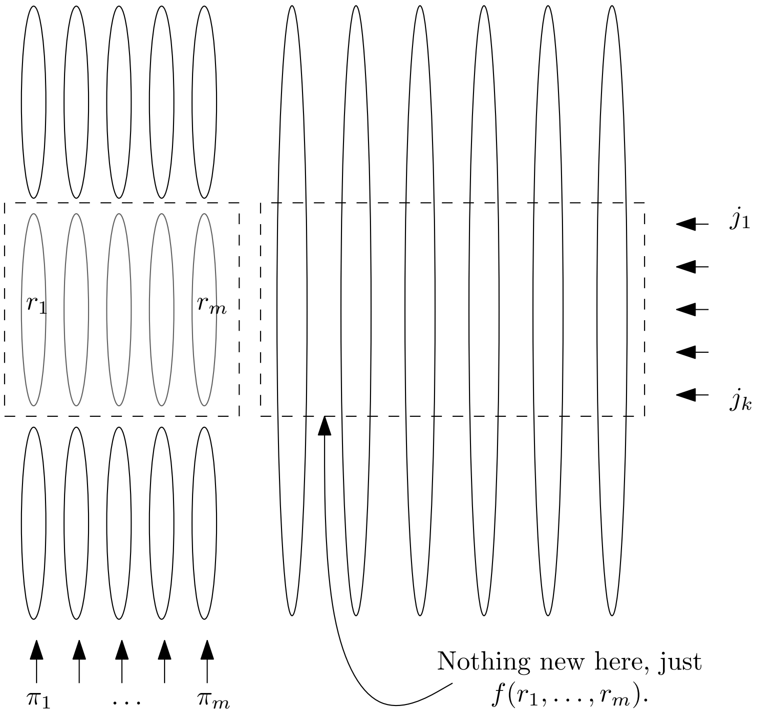

Figure 1.7: The main trick in describing the Galois correspondence Inv-Pol

Therefore, each k-ary member f of $\mathrm{Pol}(\mathrm{Inv}(\Phi))$ preserves F_k. Now it is an easy exercise that the $|A|^k$ tuple corresponding to $f(\pi_1, \pi_2, \ldots, \pi_k)$ lies in F_k and describes f (here π_i is the $|A|^k$-tuple that encodes the i-th projection). □

Remark 1.2.37. The proof of the previous theorem also shows that we need at most $|A|^m$ new variables to rewrite a k-ary relation satisfied by m tuples.

Theorem 1.2.38 ("We can add constants."). *Let $\mathbb{B} = (B, \mathcal{R})$ be a core. Then* $\mathrm{CSP}(\mathbb{B})$ *is polynomial-time equivalent to* $\mathrm{CSP}(B, \mathcal{R}^{\mathbb{B}} \cup \{\mathrm{const}(b)^{\mathbb{B}} : b \in B\})$ *where "*$\mathrm{const}(b)^{\mathbb{B}}$*" is the unary relation $\{(b)\}$.*

Proof. Call the second relational structure $\mathbb{B}'$. We need reductions $\mathrm{CSP}(\mathbb{B}) \to \mathrm{CSP}(\mathbb{B}')$ and $\mathrm{CSP}(\mathbb{B}') \to \mathrm{CSP}(\mathbb{B})$.

The first reduction is easy: We just need to add empty relations, turning $\mathbb{A} = (A, \mathcal{R}^{\mathbb{A}})$ into $\mathbb{A}' = (A, \mathcal{R}^{\mathbb{A}} \cup \{\mathrm{const}(b)^{\mathbb{A}} : b \in B\})$ where $\mathrm{const}(b)^{\mathbb{A}} = \emptyset$. Now any homomorphism $\mathbb{A}' \to \mathbb{B}'$ need only satisfy the relations from $\mathcal{R}$ and thus exists iff exists a homomorphism $\mathbb{A} \to \mathbb{B}$.

The other reduction is slightly more difficult. Let $B = \{b_1, \ldots, b_n\}$ and let $R = \{(f(b_1), \ldots, f(b_n)) | f : \mathbb{B} \to \mathbb{B}$ is an automorphism of $\mathbb{B}\}$. Then $R \in \langle \mathcal{R} \rangle$ because we can check whether $(f(b_1), \ldots, f(b_n))$ is a set of values of an automorphism using a finite set of conditions like in the proof of Theorem 1.2.36. For example, if S is a binary relation and $(b_1, b_2) \in S$ then we add the condition $(f(b_1), f(b_2)) \in S$ into the description of R. Denote $\mathbb{B}'' = (B, \mathcal{R} \cup \{R, =\})$. We know that $R \in \langle \mathcal{R} \rangle$ and so, using theorems 1.2.16 and 1.2.22 we can reduce $\mathrm{CSP}(B, \mathcal{R} \cup \{R, =\}\}$ to $\mathrm{CSP}(B, \mathcal{R})$ and to complete the proof we only need the reduction of $\mathrm{CSP}(\mathbb{B}')$ to $\mathrm{CSP}(\mathbb{B}'')$.

Given an input $\mathbb{A} = (A, \mathcal{R} \cup \{\mathrm{const}(b)^{\mathbb{A}} : b \in B\})$, consider $\mathbb{A}' = (A \dot\cup \{b_1, \ldots, b_n\}, \mathcal{R} \cup \{R, =_{\mathbb{A}'}\})$ where $\dot\cup$ denotes disjoint union. For all $s \in \mathcal{R}$ we let $s^{\mathbb{A}'} = s^{\mathbb{A}}$. We define the equivalence relation $=_{\mathbb{A}'}$ so that $=_{\mathbb{A}'}$ is identity relation on A and for $b \in B, a \in A$ it is $b =_{\mathbb{A}'} a$ iff $a \in \mathrm{const}(b)$.

We want to show that this is a reduction from $\mathrm{CSP}(\mathbb{B}')$ to $\mathrm{CSP}(\mathbb{B}'')$. If there exists a homomorphism $f : \mathbb{A} \to \mathbb{B}'$ then we obtain a homomorphism $g : \mathbb{A}' \to \mathbb{B}''$ by letting $g_{|A} = f$ and $g(b_i) = b_i$ for $i = 1, \ldots, n$. On the other hand, if $g : \mathbb{A}' \to \mathbb{B}''$ is a homomorphism then $h = g_{|B} : B \to B$ is an automorphism of $\mathbb{B}''$ because $(g(b_1), \ldots, g(b_n)) \in R$. Then $f = h^{-1}g$ is a homomorphism $\mathbb{A}' \to \mathbb{B}''$ such that $f(b_i) = b_i$ and $f_{|A}$ is the needed homomorphism $\mathbb{A} \to \mathbb{B}'$. □

Theorem 1.2.39. *If $\mathrm{CSP}(\mathbb{B})$ is in P then there exists a polynomial algoritm that, for a given $\mathbb{A}$, finds a homomorphism $f : \mathbb{A} \to \mathbb{B}$ or proves that no such homomorphism exists.*

Proof. The idea of the proof is quite simple: We keep adding constraints and use our poly-time oracle (CSP algorithm) to check that these constraints still allow a homomorphism to exist. In the end, we either run out of possibilities or our constraints specify an unique homomorphism. The following proof formalises this idea.

First of all, we show that it is enough to prove the theorem for cores with constants. Let $\mathbb{B}$ be a general relational structure. Because the size of $\mathbb{B}$ is not a part of the input, we can find the core $\mathbb{C} = (C, \mathcal{R})$ of $\mathbb{B}$ in constant time. Let $\mathbb{D} = (C, \mathcal{R} \cup \{\mathrm{const}(c) : c \in C\})$ be the core with constants, $\mathrm{const}(c) = \{(c)\}$ for all $c \in C$.

We know that CSP($\mathbb{B}$) is poly-time equivalent to CSP($\mathbb{D}$). Because CSP($\mathbb{B}$) is in P, there is a polynomial time algorithm p deciding whether a given $\mathbb{A}$ is in CSP($\mathbb{D}$). Moreover, if $f : \mathbb{A} \to \mathbb{D}$ is a homomorphism and we let $\mathbb{A}'$ be $\mathbb{A}$ stripped of all const(c) relations then f is also a homomorphism from $\mathbb{A}'$ to $\mathbb{B}$.

Take now any instance $\mathbb{A} = (A, \mathcal{R} \cup \{\mathrm{const}^{\mathbb{A}}(c) : c \in C\})$ of CSP($\mathbb{D}$). (If $\mathbb{A}$ is similar to $\mathbb{B}$, we can let $\mathrm{const}^{\mathbb{A}}(c)$ be empty for all c.) Let $A = \{a_1, a_2, \dots, a_n\}$. If $p(\mathbb{A})$ outputs that $\mathbb{A} \notin$ CSP($\mathbb{D}$) we are done and answer in negative. We shall define by induction a sequence $f(a_1), f(a_2), \dots, f(a_n) \in B$ describing a homomorphism $f : \mathbb{A} \to \mathbb{D}$.

Start with $i = 1$. Assume that $f(a_1), \dots, f(a_{i-1})$ are defined already and that they are the values of some homomorphism $f : \mathbb{A} \to \mathbb{D}$. First, we guess the value of $f(a_i)$ (there are only $|C|$ possibilities). Then define $\mathbb{A}_i$ as a relational structure similar to $\mathbb{A}$ such that $R^{\mathbb{A}_i} = R^{\mathbb{A}}$ for all $R \in \mathcal{R}$ and $\mathrm{const}^{\mathbb{A}_i}(c) = \mathrm{const}^{\mathbb{A}} \cup \{(a_j) : j \le i, f(a_j) = c\}$. Notice that in this notation it is $\mathbb{A}^0 = \mathbb{A}$ and that homomorphisms $g : \mathbb{A}_i \to \mathbb{D}$ are precisely the homomorphisms $\mathbb{A} \to \mathbb{D}$ that satisfy $g(a_j) = f(a_j)$ for $j = 1, 2, \dots, i$. In particular, our hypothetical homomorphism f is also a homomorphism $\mathbb{A}^i \to \mathbb{D}$.

There are $|C|$ possible values of $f(a_i)$ and thus $|C|$ possible candidates for the structure $\mathbb{A}_i$. By running the algorithm p on each of these candidates, we obtain some $c \in C$ such that there exists a homomorphism $\mathbb{A} \to \mathbb{D}$ with first i values $f(a_1), \dots, f(a_{i-1}), c$. But that is precisely what we want, so we fix these values, increase i by one and continue. In the end, we obtain a complete homomorphism $f(a_1), \dots, f(a_n)$. Notice that f is the unique homomorphism $\mathbb{A}_n \to \mathbb{D}$. □

1.2.2 Binary relational structures

In this section we shall discuss the case where the relational structure $\mathbb{B}$ is binary, i.e. $|B| = 2$. In the binary case, we not only have dichotomy, but we can actually decide whether CSP($\mathbb{B}$) is NP-complete or in P by looking at the set Pol($\mathbb{B}$).

Let $B = \{0, 1\}$. We begin by defining the following operations on B:

- Unary 0 and 1 *constant operations* given by $\forall x, 0(x) = 0, 1(x) = 1$.
- Unary *negation* defined as $\neg x = x + 1 \pmod 2$.
- *Binary and* defined by $x \wedge y = 1$ iff $x = y = 1$.
- *Binary or* defined as $x \vee y = 1$ iff $x = 1$ or $y = 1$.
- The *plus operation* $\mathrm{p}(x, y, z) = x + y + z \pmod 2$.
- The *majority operation* $\mathrm{m}(x, y, z) = 1$ iff at least two of the numbers x, y, z are 1.

Definition 1.2.40. An operation $f : A^n \to A$ is *idempotent* if $f(a, a, \dots, a) = a$ for all $a \in A$.

Definition 1.2.41. An operation $f : A^n \to A$ is a *projection* (to the i-th coordinate) if there exists an i such that for all $a_1, \dots, a_n \in A$ it is $f(a_1, \dots, a_n) = a_i$. When $A = \{0, 1\}$, we say that an operation f *negates projection* if there exists i such that $f(a_1, \dots, a_n) = \neg a_i$.

Definition 1.2.42. An operation $f : A^n \to A$ is a *near unanimity* (often abbreviated as "nu") if

$$f(x, \dots, x, x, y) = f(x, \dots, x, y, x) = f(x, \dots, y, x, x) = \cdots = f(y, x, \dots, x) = x.$$

Theorem 1.2.43. *Let $\mathbb{B} = (\{0, 1\}, \mathcal{R})$. Then either $\mathbb{B}$ admits at least one of the polymorphisms $0, 1, \wedge, \vee, \mathrm{p}, \mathrm{m}$ or all polymorphisms of $\mathbb{B}$ are projections or negate projections.*

Proof. If $\mathbb{B}$ is not a core then it admits by definition an unary polymorphism 0 or 1. Assume that $\mathbb{B}$ is a core. Then each unary polymorphism of $\mathbb{B}$ is an automorphism and the group of automorphisms of $\mathbb{B}$ is a subgroup of S_2.

For $f \in \mathrm{Pol}\mathbb{B}$, define $\sigma(x) = f(x, \dots, x)$. It must be $\sigma^2 = \mathrm{id}$ because σ is either the identity or the negation. We claim that $\sigma \circ f$ is idempotent. This means precisely that $\sigma \circ f(x, x, \dots, x) = x$ and we have just shown that $x = \sigma^2(x) = \sigma(f(x, \dots, x))$. This construction gives us a tool to turn any polymorphism f into an idempotent polymorphism $\sigma \circ f$. Notice that $\sigma \circ f = f$ or $\sigma \circ f = \neg f$.

It is therefore sufficient to prove that if a core $\mathbb{B}$ does not admit $\wedge, \vee, \mathrm{p}$ and m then any idempotent polymorphism f of $\mathbb{B}$ is a projection. We shall prove this by an unusual induction on the arity q of f idempotent polymorphism of $\mathbb{B}$ – there will be several first induction steps, as we have to do the the first few values of q by hand, getting some elbow room to handle the general case.

- "$q = 1$" This case is trivial, as $f = \mathrm{id}$.

- "$q = 2$" Write the values of f into a table:

f	0	1
0	0	?
1	?	1

  There are only four possible ways of writing 0 or 1 in place of the question marks, and the four resulting maps are the projection to first or second coordinate, $\vee$ and $\wedge$, respectively. We conclude that the theorem holds for $q = 2$.

- "$q = 3$" First notice that $f(\neg x, \neg y, \neg z) = \neg f(x, y, z)$. We can see this by noticing that at least two of the variables x, y, z must be equal, without loss of generality let $x = y$. Then $g(x, z) = f(x, x, z)$ is, by induction assumption, a projection and so $f(\neg x, \neg x, \neg y) = \neg f(x, x, y)$.

Because we know that $f(0,0,0) = 0, f(1,1,1) = 1$, all we need to determine f is to know the values of $f(0,0,1) = a, f(0,1,0) = b$ and $f(1,0,0) = c$. If $a = b = c = 1$ resp. $a = b = c = 0$ we get $f = \mathrm{p}$ resp. $f = \mathrm{m}$. There are only two more cases left (up to permutation of variables):

  - Let $a = 1$, $b = c = 0$. Here, it is $f(x,y,z) = x$ and f is a projection.
  - Let $a = b = 1$, $c = 0$. In this case, let $t(x,y,z) = f(x,y,f(x,y,z))$. Then t is also an idempotent polymorphism and $t(\neg x, \neg y, \neg z) = \neg t(x,y,z)$. Directly calculating the values, we see that it is $t(1,0,0) = t(0,1,0) = t(0,0,1) = 0$ and so $t = \mathrm{m}$.

- "$q \geq 4$" We shall first prove that if f is not a projection then it is a near unanimity (nu) operation.

  Assume that $f(0,0,\dots,0,1) = 1$. Then, by induction assumption, we have a set of projections: $\pi_1 = f(x,x,x_3,x_4,\dots,x_q)$, $\pi_2 = f(x,x_3,x,x_4,\dots,x_k)$, $\pi_3 = f(x_3,x,x,x_4,\dots,x_q)$. We know that these are projections on the last coordinate because $f(0,\dots,0,1) = 1$. Again, in the general case at least two of the first three variables must be equal, so it is $f(x_1,\dots,x_q) = x_q$, a projection. By a similar argument, it must be $f(1,\dots,1,0) = 1$ and by premuting the variables, we get that $f(x,\dots,x,y,x,\dots,x) = x$.

  The nu property is quite powerful and quickly brings us to a contradiction. We know that it is $f(0,1,\dots,1) = 1$, so $f(0,x_2,\dots,x_q)$ is, by the induction hypothesis, a projection to the i-th coordinate for some $i \in \{2,\dots,q\}$. But then it would be $f(0,\dots,0,1,0,\dots,0) = 1$ (with one in the i-th place), contradicting the nu property.

□

Let us visualise the previous theorem: We have proven that anything strictly above $\langle\neg\rangle$ in the (functional) clone lattice of $\{0,1\}$ is also above at least one of $\langle 0\rangle$, $\langle 1\rangle$, $\langle\wedge\rangle$, $\langle\vee\rangle$, $\langle \mathrm{p}\rangle$, $\langle \mathrm{m}\rangle$. The lattice of functional clones is sometimes called the *Post's lattice of clones.*

Let us say a few words about the properties of this lattice. First of all, notice that all the above clones are atomic, ie. there is no nontrivial element of the Post's lattice below, say $\langle 0\rangle$. In case of $\langle 0\rangle$, it is an easy observation that $\langle 0\rangle$ is the set of all projections and all the maps $f(x_1,\dots,x_n) = 0$ (n-ary zeroes) and any n-ary zero generates the unary zero, so any subclone of $\langle 0\rangle$ is either trivial or $\langle 0\rangle$.

Exercise 1.2.44. Prove that $\langle 1\rangle, \langle\wedge\rangle, \langle\vee\rangle, \langle \mathrm{p}\rangle, \langle \mathrm{m}\rangle$ and $\langle\neg\rangle$ are also atomic clones.

From the Theorem 1.2.43 we obtain that any functional clone that does not contain any of the clones $\langle 0\rangle, \langle 1\rangle, \langle\wedge\rangle, \langle\vee\rangle, \langle \mathrm{m}\rangle, \langle \mathrm{p}\rangle$ must be contained in $\langle\neg\rangle$.

Our goal now is to prove that CSP($\mathbb{B}$) is in P iff $\mathbb{B}$ admits at least one of the operations 0, 1, $\wedge$, $\vee$, p, m and is NP-complete otherwise. The original proof of this result is due to Shaefer from 1978.

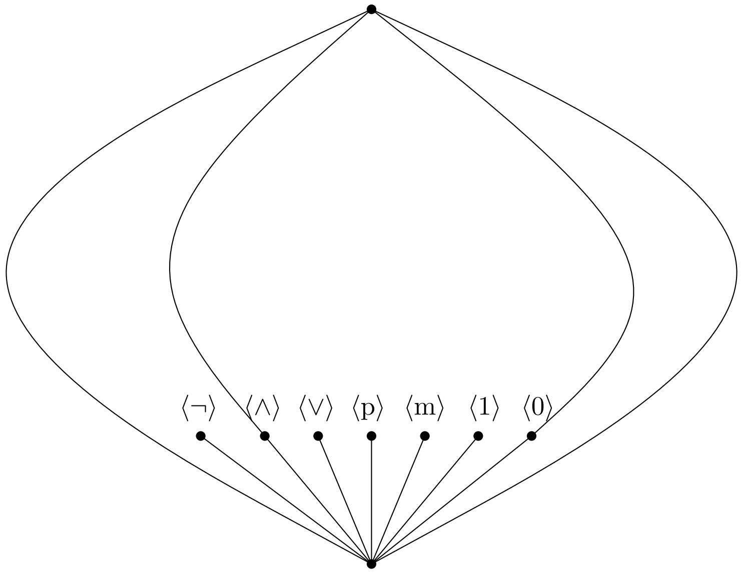

Figure 1.8: The bottom of Post's lattice

For $\mathbb{A}$ general algebra, f is a *permutation of a projection* if $f(x_1, \ldots, x_n) = \sigma(x_i)$ for a fixed i and $\sigma \in \mathrm{Aut}(\mathbb{B})$. For example, if $\mathbb{B}$ is a binary algebra admitting the automorphism $\neg$ then negation of a projection is a permutation of a projection.

Lemma 1.2.45. *Let $\mathbb{A} = (A, \mathcal{R})$ be a relational structure such that $|A| \geq 2$ and all polymorphisms of $\mathbb{A}$ are projections or permutations of projections. Then* $\mathrm{CSP}(\mathbb{A})$ *is NP-complete.*

Proof. First notice that because all unary polymorphisms of $\mathbb{A}$ are automorphisms, $\mathbb{A}$ is a core. Using Theorem 1.2.38 we obtain that $\mathrm{CSP}(A, \mathcal{R})$ is polytime equivalent with $\mathrm{CSP}(A, \mathcal{R} \cup \{\mathrm{const}(a) | a \in A\}) = \mathrm{CSP}(\mathbb{C})$ where $\mathrm{const}(a)$ are the unary constants. Obviously, $\mathrm{Aut}(\mathbb{C})$ is trivial, as every automorphism has to preserve unary constants. Thus $\mathbb{C}$ has only projections as its polymorphism. Intuitively, this means that the set of relations of $\mathbb{C}$ is very rich.

We shall now reduce 3-SAT to $\mathrm{CSP}(\mathbb{C})$. Begin by choosing two distinct elements of A and labeling them 0 and 1. Recall that the relations in 3-SAT can be written as $S_{\alpha\beta\gamma} = \{0,1\}^3 \setminus \{(\alpha, \beta, \gamma)\}$ where α, β, γ are one or zero. But because $\mathrm{Pol}(\mathbb{C})$ is the smallest possible functional clone, using the Galois correspondence we obtain that $\langle \mathcal{R}^{\mathbb{C}} \rangle = \mathrm{Inv}(\mathrm{Pol}(\mathbb{C})) = 2^A$ and so $S_{\alpha\beta\gamma} \in \langle \mathcal{R}^{\mathbb{C}} \rangle$. Due to Theorem 1.2.22 we have that 3-SAT can be poly-time reduced to $\mathrm{CSP}(\mathbb{C})$, concluding our proof. □

Remark 1.2.46. The previous theorem implies that if $\mathbb{B}$ does not admit 0, 1, $\wedge$, $\vee$, p, m then $\mathrm{CSP}(\mathbb{B})$ is NP-complete.

Remark 1.2.47. Notice that the above theorem does not requie that $\mathbb{B}$ be binary, it works for every nontrivial relational structure.

Lemma 1.2.48. *If $\mathbb{B}$ admits a constant polymorphism (i.e. 0 or 1) then $\mathrm{CSP}(\mathbb{B})$ is in P.*

Proof. In this case, $\mathbb{B}$ is very simple indeed. Denote by $\mathbb{B}_0$ the image of $\mathbb{B}$ under 0. Then $\mathbb{B}_0$ is a retract of $\mathbb{B}$ and so $\mathrm{CSP}(\mathbb{B}_0) = \mathrm{CSP}(\mathbb{B})$. Now if $\mathbb{C}$ is a relational structure, there is only one candidate for a homomorphism; namely the map $f(c) = 0$ for each $c \in C$ which is a homomorphism iff $R^{\mathbb{B}} = \emptyset \Rightarrow R^{\mathbb{C}} = \emptyset$. □

Lemma 1.2.49. *If $\mathbb{B}$ admits $\wedge$ or $\vee$ then $\mathrm{CSP}(\mathbb{B})$ is in P.*

Proof. First note that by switching one and zero, we interchange $\wedge$ with $\vee$, as it is $\neg(\neg x \wedge \neg y) = x \vee y$. Thus it is enough to prove that whenever $\mathbb{B}$ admits $\wedge$ then $\mathbb{B}$ is in P.

Without loss of generality assume that $\mathbb{B}$ is a core (if not, we use the previous lemma). Obviously, $\mathrm{CSP}(B, \mathcal{R})$ can be reduced to $\mathrm{CSP}(\mathbb{C}) = \mathrm{CSP}(B, \mathcal{R} \cup \mathrm{const}(1))$ where $\mathrm{const}(1) = \{(1)\}$ is the unary constant 1. In the following, the letter R can stand for any relation $R \in \mathcal{R}^{\mathbb{C}}$ with the exception of $\mathrm{const}(1)$.

Let now $\mathbb{A}$ be a relational structure similar to $\mathbb{C}$. First note that by slightly abusing notation, we can consider $\mathrm{const}(1)^{\mathbb{A}}$ to be identical with the set $\{a \in A : (a) \in \mathrm{const}(1)^{\mathbb{A}}\}$. We want to iteratively construct a homomorphism $f : \mathbb{A} \to \mathbb{C}$. Obviously, if $x \in \mathrm{const}(1)^{\mathbb{A}}$ then $f(x) = 1$. This statement defines a map $f_1 : \mathrm{const}(1)^{\mathbb{A}} \to \mathbb{C}$. If this is not a homomorphism then there obviously can be no $f : \mathbb{A} \to \mathbb{C}$ and so we are done. Assume thus that f_1 is a homomorphism from the substructure of $\mathbb{A}$ induced by the set $\mathrm{const}(1)^{\mathbb{A}}$. Such mappings are called *partial homomorphisms* and will play an important role later in the bounded width theory. If $\mathrm{const}(1) = A$ then we are done, otherwise we want to extend f_1.

This extension has two steps. In the first step, we transform the problem so that there is no $(a_1, \ldots, a_k) \in R^{\mathbb{A}}$ such that for some i it is $f_1(a_i) = 1$. Assume that $a_1, \ldots, a_l \in \mathrm{const}(1)^{\mathbb{A}}$ for $0 < l < k$ (we can permute relations). The condition is $(1, \ldots, 1, f(a_{l+1}), \ldots, f(a_k)) \in R^{\mathbb{C}}$. Consider the relation $R^{\mathbb{C}}_{l+1,\ldots,k} = \{(c_{l+1}, \ldots, c_k) : (1, \ldots, 1, c_{l+1}, \ldots, c_k) \in R^{\mathbb{C}}\}$ Our condition is equivalent with $(f(a_{l+1}), \ldots, f(a_k)) \in R^{\mathbb{C}}_{l+1,\ldots,k}$. Therefore, we can remove $(a_1, \ldots, a_k)$ from $R^{\mathbb{A}}$ and add to the signature a new relation $R_{l+1,\ldots,k}$ defined in $\mathbb{A}$ as $R^{\mathbb{A}}_{l+1,\ldots,k} = \{(a_l, \ldots, a_k)\}$ and in $\mathbb{C}$ as the above $R^{\mathbb{C}}_{l+1,\ldots,k}$. We can do this for every tuple $(a_1, \ldots, a_k) \in R^{\mathbb{A}}$, finally obtaining (in polynomial time) a situation where no $(a_1, \ldots, a_k) \in R^{\mathbb{A}}$ contains $a_i, f_1(a_i) = 1$. We have modified both structures $\mathbb{A}, \mathbb{C}$ but it is easy to see that these changes can not turn a non-homomorphism $f : \mathbb{A} \to \mathbb{C}$ to a homomorphism or vice versa.

In the second step of our extension procedure, we eliminate all nonempty $R^{\mathbb{C}}$ such that $(0, \ldots, 0) \notin R^{\mathbb{C}}$. Let us have one such $R^{\mathbb{C}}$. Then we take $r = \bigwedge R^{\mathbb{C}} = r_1 \wedge r_2 \wedge \cdots \wedge r_m$ where $\{r_1, \ldots, r_m\} = R^{\mathbb{C}}$. Because $R^{\mathbb{C}}$ is $\wedge$-invariant (and $\wedge$ is associative), we know that $r \in R^{\mathbb{C}}$. As $r \neq (0, \ldots, 0)$, it is for some index i true that $(s_1, \ldots, s_k) \in R^{\mathbb{C}} \Rightarrow s_i = 1$. Observe now that R-satisfaction can be emulated by putting all $s \in A$ such that $A^{i-1} \times \{s\} \times A^{k-i} \cap R^{\mathbb{A}} \neq \emptyset$ into $\mathrm{const}(1)^{\mathbb{A}}$, replacing $R^{\mathbb{C}}$ with $R^{\mathbb{C}}_{1,\dots,i-1,i+1,\dots,k}$, and replacing the relation $R^{\mathbb{A}}$ with $\{(s_1, \dots, s_{i-1}, s_{i+1}, \dots, s_k) : (s_1, \dots, s_k) \in R^{\mathbb{A}}\}$. Thus we have enlarged the set $\mathrm{const}(1)^{\mathbb{A}}$ and obtained a mapping $f_2 : \mathrm{const}(1)^{\mathbb{A}} \to C$. After checking that f_2 is indeed a partial homomorphism, we can again extend $\mathrm{const}(1)^{\mathbb{A}}$, obtaining f_3, and so on.

Assuming that $f_1, f_2, \dots$ are all partial homomorphisms, when can we no longer extend $\mathrm{const}(1)^{\mathbb{A}}$? This only happens if for all nonempty $R^{\mathbb{C}}$ it is $(0, \dots, 0) \in R^{\mathbb{C}}$. But thanks to the first step we can assume that $(a_1, \dots, a_k) \in R^{\mathbb{A}} \Rightarrow a_1, \dots, a_k \notin \mathrm{const}(1)^{\mathbb{A}}$. Thus we can define $f : A \to B$ as $f(x) = 1$ for $x \in \mathrm{const}(1)^{\mathbb{A}}$ and $f(x) = 0$ otherwise. Because for all nonempty $R^{\mathbb{C}}$ it is $(0, \dots, 0) \in R^{\mathbb{C}}$, f is a homomorphism iff for all R it is $R^{\mathbb{C}} = \emptyset \Rightarrow R^{\mathbb{A}} = \emptyset$. But that is a necessary condition for the existence of *any* homomoprhim $\mathbb{A} \to \mathbb{B}$. So $\mathbb{A} \in \mathrm{CSP}(\mathbb{B})$ iff f is a homomorphism and the problem is solved.

We shall leave to the reader to verify that the running time of all extension procedures can be limited by some polynomial of $|\mathbb{A}|$. Because we have done at most $|A|$ extensions, the whole algorithm is polynomial-time. □

Remark 1.2.50. The preceding proof might seem too complicated to the reader. This is because we wanted to make sure that we know what is going on when solving $\mathrm{CSP}(\mathbb{B})$. As we shall see, there is also a more general proof of this lemma stemming from the bounded width theory. This later proof will have the advantage of being less technical while using the same intuitive ideas.

Lemma 1.2.51. *If* $\mathbb{B}$ *admits* p *then* $\mathrm{CSP}(\mathbb{B})$ *is in P.*

Proof. Here, each nonempty R is an affine space over $\mathbb{Z}_2$. To see this, take a nonempty R and fix $r \in R$. Then for any $s, t \in R$ it is $r + s + t = \mathrm{p}(r, s, t) \in R$ where the addition is the addition in $\mathbb{Z}_2^k$. Because we are operating over a field of characteristic two, we have

$$r + (s - r) + (t - r) = r + s + t \in R,$$

so $R - r = \{s - r : s \in R\}$ is a subspace of $\mathbb{Z}_2^k$. This subspace can be described using standard linear algebra methods: There exists a set of vectors $u_1, \dots, u_m$ (basis of the space perpendicular to $R - r$) such that for $x \in \mathbb{Z}_2^k$ it is $x \in R - r$ iff the product $\langle x, u_i \rangle$ is zero for all i.

Now all homomorphisms $f : \mathbb{A} \to \mathbb{B}$ have to satisfy the condition $(a_1, \dots, a_k) \in R^{\mathbb{A}} \Rightarrow (f(a_1), \dots, f(a_k)) \in R^{\mathbb{B}}$. Every tuple $(a_1, \dots, a_k) \in R^{\mathbb{A}}$ then, by the above paragraph, translates into a set of linear equations (over $\mathbb{Z}^2$) of the form $\langle (f(a_1), \dots, f(a_k)) - r, u_i \rangle = 0$. The CSP problem is then equivalent to solving a set of linear equations given by all such tuples $(a_1, \dots, a_k) \in R^{\mathbb{A}}$ for all R. There are numerous methods (most basic being the Gauss elimination) for solving such a set in polynomial time. □

We will solve the case when $\mathbb{B}$ admits a majority operation by building the general theory of bounded width. For now, we just claim that if $\mathbb{B}$ admits m then indeed $\mathrm{CSP}(\mathbb{B})$ is in P, finishing the dichotomy proof.

Theorem 1.2.52. *If $\mathbb{B}$ is a binary relational structure then* CSP($\mathbb{B}$) *is either NP-complete or in P.*

1.3 Bounded width theory

The basic idea of the bounded width theory is to transform CSP($\mathbb{B}$) into a simpler question: Instead of one complete homomorphism we want a nice set of partial homomorphisms, called a (j,k)-strategy. If $\mathbb{A} \in$ CSP($\mathbb{B}$) then there always exists a (j,k)-strategy, but the converse implication is not true in general. However, for some $\mathbb{B}$'s, these partial homomorphisms are all that is needed to produce a full homomorphism.

Recall from the previous section that for $\mathbb{A}, \mathbb{B}$ relational structures and $C \subset A$, $f : C \to B$ is a *partial homomorphism* if it is a homomorphism $\mathbb{C} \to \mathbb{B}$ where $\mathbb{C}$ is the substructure of $\mathbb{A}$ induced by the set C. If f, g are partial homomorphisms then we say that g is an *extension* of f and f is a *subfunction* of g, writing $f \subset g$, if $\mathrm{dom} f \subset \mathrm{dom} g$ and $g_{|\mathrm{dom} f} = f$. We shall also often use the notation $R_{|K}$, where R is an n-ary relation and $K \subset \{1, 2, \ldots, n\}, |K| = k$ to mean the relation $R_{|K} = \{(r_{i_1}, r_{i_2}, \ldots, r_{i_k}) | (r_1, \ldots, r_n) \in R\} \subset A^k$, where $i_1 < i_2 < \cdots < i_k$ are the elements of K. This is consistent with treating tuples as functions $\{1, \ldots, n\} \to A$.

Definition 1.3.1. Let $\mathbb{A}, \mathbb{B}$ be similar relational structures, $0 \leq j < k$ integers. A nonempty set H of partial homomorphisms $\mathbb{A} \to \mathbb{B}$ is a (j,k)-*strategy* if:

- H is closed under taking subfunctions
- H has the (j,k)-*forth property*, that is $\forall f \in H$ such that $|\mathrm{dom}(f)| \leq j$ and for all $K \subset A$ such that $\mathrm{dom}(f) \subset K$ and $|K| \leq k$ there exists $g \in H, \mathrm{dom}(g) = K, f \subset g$. That is, all "small" f's in H can be extended to $g, |\mathrm{dom}(g)| \leq k$.

Observation 1.3.2. *If $\mathbb{A} \in$ CSP($\mathbb{B}$) then for any j, k there exists a (j,k) strategy for $\mathbb{A}$ and $\mathbb{B}$.*

Proof. Let $f : \mathbb{A} \to \mathbb{B}$ be a homomorphism. Then all we have to do is take $H = \{f_{|K} | K \subset A\}$ and verify that (i) and (ii) hold. □

Definition 1.3.3. We say that $\mathbb{B}$ has *relational width* (j,k) if

$$\mathrm{CSP}(\mathbb{B}) = \{\mathbb{A} | \text{There exists a } (j,k)\text{-strategy for } \mathbb{A} \text{ and } \mathbb{B}.\}.$$

Remark 1.3.4. If $\mathbb{B}$ has relational width (j,k) and k is smaller than the arity of the relation $R \in \mathcal{R}^{\mathbb{B}}$, then we can effectively ignore this relation when making an (j,k) strategy. This means that R is not very important (for example, it can be the full relation), because the condition $A \in$ CSP($\mathbb{B}$) does not depend on what $R^{\mathbb{A}}$ looks like.

Definition 1.3.5. $\mathbb{B}$ has *relational width* j if $\exists k$ such that $\mathbb{B}$ has relational width (as defined above) (j,k). We say that $\mathbb{B}$ has *bounded width* if there exists a finite j such that $\mathbb{B}$ has relational width j.

The above definition introduces a slight inconsistency in terminology because we now have two meanings for the term relational width. Fortunately, the two are usually easy to tell apart.

Algorithm 1.3.6 (Local Consistency)**.** For any $\mathbb{A}$, $\mathbb{B}$ relational structures and $0 \leq j < k$ integers we can in polynomial time (measured in the size of $\mathbb{A}$, we consider $j, k, \mathbb{B}$ fixed) construct a (j,k)-strategy or show that no such strategy exists (i.e. $\mathbb{A} \notin \mathrm{CSP}(\mathbb{B})$).

Proof. Let $H = \{f : \mathbb{A} \to \mathbb{B} | f \text{ partial homomorphism}, |\mathrm{dom}(f)| \leq k\}$. This set can be constructed by brute-force methods, as it has cardinality polynomial in the size of $\mathbb{A}$ (an easy upper bound would be $|A|^{k+1}|B|^{k+1}$). It is easy to see that this H must contain a (j,k)-strategy, if such a thing exists.

We shall now remove homomorphisms from H until it becomes a (j,k)-strategy or there is nothing left: We search through H and for each $f \in H$ check whether H satisfies first and second condition for (j,k)-strategy when checked at f. If not then we remove this f and start searching anew.

The maximum run-time of all such checks can be bounded by a polynomial and as we run at most polynomially many checks, the whole algorithm is polynomial. If at the end it is $H = \emptyset$ then the program has shown that there can not be any (j,k)-strategy (If $S \subset H$ is a (j,k) strategy, then our program will never delete any member of S.), otherwise H is a valid (j,k)-strategy. □

Remark 1.3.7. The existence of the above algorithm implies that if $\mathbb{B}$ has bounded width, then $\mathrm{CSP}(\mathbb{B})$ is in P, because we just have to check for (j,k) strategies for some j, k.

Having established the general theory, let us look at the case when $\mathbb{B}$ admits a near-unanimity operation. We want to show that then $\mathbb{B}$ has bounded width.

Lemma 1.3.8. *Let R be an n-ary relation invariant under a k-ary near-unanimity operation t. Then for all $r \in A^n$ we have $r \in R$ iff for all $|K| < k, K \subset \{1, 2, \dots, n\}$, it is $r_{|K} \in R_{|K}$.*

Proof. Thorough the proof we will assume that $K \subset \{1, 2, \dots, n\}$. We shall proceed by induction and show that if for a given r and for all $|K| < l$ it is $r_{|K} \in R_{|K}$ then $\forall |K| \leq l$ it is $r_{|K} \in R_{|K}$, starting with $l = k$. When $l = n$, we will be done.

Assume that we have r such that $\forall |K| < l, r_{|K} \in R_{|K}$, let (without loss of generality) $K = \{1, 2, \dots, l\}$. We need to show that $r_{|K} \in R_{|K}$. Because $r_{|\{2,\dots,l\}} \in R_{|\{2,\dots,l\}}$, we know that there exists $s^1 \in R$ such that $s^1 = (?, r_2, r_3, \dots, r_l, ?, ?, \dots, ?) \in R$. Here the question marks denote unknown elements of A. In general, there exists $s^i = (r_1, \dots, r_{i-1}, ?, r_{i+1}, \dots, r_l, ?, ?, \dots, ?) \in R$ for each $i \leq k$. Let us take $t(s^1, \dots, s^k)$:

$$\begin{array}{l}
(?, r_2, r_3, \dots, r_l, ?, ?, \dots, ?) \in R \\
(r_1, ?, r_3, \dots, r_l, ?, ?, \dots, ?) \in R \\
\vdots \quad \vdots \quad \vdots \\
(r_1, r_2, r_3, \dots, ?, ?, ?, \dots, ?) \in R \\
\hline
(r_1, r_2, r_3, \dots, r_l, ?, ?, \dots, ?) \in R
\end{array}$$

Notice that $t(s^1, \dots, s^k)_{|K} = r_{|K}$ and so $r_{|K} \in R_{|K}$, concluding our proof. □

Corollary 1.3.9. *Let Γ be a relational clone admitting a k-ary near-unanimity polymorphism. Then $\Gamma = \langle \Gamma_{|<k} \rangle$, where $\Gamma_{|<k}$ consist of all relations of Γ whose arity is less than k.*

Proof. One inclusion is obvious. To see that $\Gamma \subset \langle \Gamma_{|<k} \rangle$, use the previous lemma. First of all, for any n-ary relation $R \in \Gamma$ we can rewrite the relation $(a_1, \dots, a_{k-1}) \in R_{|\{1,\dots,k-1\}}$ using a primitive positive formula as $\exists a_k, \dots, a_n, (a_1, \dots, a_n) \in R$ and the same can be done for any $R_{|K}$. Thus $R_K \in \Gamma_{|<k}$ for all $|K| < k$. We also have that $(a_1, \dots, a_n) \in R$ iff for all $|K| < k$ it is $(a_1, \dots, a_n)_{|K} \in R_{|K}$, that there are only finitely many such K's and that $R_{|K}$ are in $\langle \Gamma_{|<k} \rangle$. Thus R is a conjunction of finitely many terms from $\langle \Gamma_{|<k} \rangle$ and so $R \in \langle \Gamma_{|<k} \rangle$. □

Lemma 1.3.10. *Let $\mathbb{B}$ be a relational structure with an r-ary near-unanimity polymorphism t. Then $\mathbb{B}$ has relational width $r - 1$.*

Proof. We want to proceed by induction, producing a $(j+1, j+2)$-strategy from $(j, j+1)$-strategy. At the beginning, let $j = r - 1$ and let H be a $(j, j+1)$-strategy.

We shall now make two observations. First, we can assume that $\mathcal{R}^{\mathbb{B}}$ is a relational clone and so, thanks to the above corollary, $\mathcal{R}^{\mathbb{B}} = \langle \mathcal{R}^{\mathbb{B}}_{|<r} \rangle$. Thus all we have to worry about are relations of arity less than r.

Our second observation is that instead of H we can take a closure $\overline{H}$ of H under t. Formally, we define $\overline{H}$ as the union of all sets H_i such that $H_0 = H$ and

$$H_i = \{t(f_1, \dots, f_r) | \forall i, f_i \in H, \forall i, j, \mathrm{dom}(f_i) = \mathrm{dom}(f_j)\}.$$

By idempotency of t, we get $H_0 \subseteq H_1 \subseteq H_2 \subseteq \dots$

Because t is a near-unanimity polymorphism, $H \subset \overline{H}$ and $\overline{H}$ is a set of partial homomorphisms. It is also a (j, k)-strategy: If for all i it is $g_i \subset f_i$ and $\mathrm{dom}(g_i) = I, \mathrm{dom}(f_i) = J$ then $t(g_1, \dots, g_r) \subset t(f_1, \dots, f_r)$ and the domain of the first function is I and domain of the second is J. This means that we can take subfunctions. Similarly, to get an extension, it is enough to extend each of the functions g_i. Notice that this construction can be generalised: If we wished, we could construct the closure of H under all polymorphisms of $\mathbb{B}$ in similar fashion.

For each $f \in \overline{H}$ and each a_{j+2} such that $a_{j+2} \notin \mathrm{dom} f = \{a_1, \dots, a_{j+1}\}$ we want to add to $\overline{H}$ a function $h \supset f$ such that $\mathrm{dom} h = \{a_1, \dots, a_{j+2}\}$. Denote

$f(a_i) = b_i$. From the $(j, j+1)$-forth property, we know that in $\overline{H}$ are also the following functions (the "–" symbol means "undefined" and c_i's are unknown values):

$$\begin{array}{ll}
f: & b_1\ b_2\ \ldots\ b_r\ \ldots\ b_{j+1}\ -\\
g_1: & -\ b_2\ \ldots\ b_r\ \ldots\ b_{j+1}\ c_1\\
g_2: & b_1\ -\ \ldots\ b_r\ \ldots\ b_{j+1}\ c_2\\
\vdots & \qquad\qquad \vdots\\
g_r: & b_1\ b_2\ \ldots\ -\ \ldots\ b_{j+1}\ c_r
\end{array}$$

Let now h be a mapping defined on $\{a_1, \ldots, a_{j+2}\}$ by $h(a_i) = b_i$ for $i \leq j+1$ and $h(a_{j+2}) = t(c_1, \ldots, c_r)$. Obviously, $f \subset h$. We claim that all subfunctions of h are in $\overline{H}$ and that h is a partial homomorphism. It is $h_{|\{a_1,\ldots,a_{j+1}\}} = f$ so the interesting case is removing $a_i, i \leq j+1$. Without loss of generality, let $i = 1$. From each g_i we can obtain a function g_i' by removing a_1 from the domain and replacing it with a_i using the $(j, j+1)$-forth property. We then have the following functions in $\overline{H}$:

$$\begin{array}{lll}
g_1: & -\ b_2\ \ldots\ b_r\ \ldots\ b_{j+1} & c_1\\
g_2': & -\ ?\ \ldots\ b_r\ \ldots\ b_{j+1} & c_2\\
\vdots & & \\
g_r': & -\ b_2\ \ldots\ ?\ \ldots\ b_{j+1} & c_r\\
\hline
t(g_1, g_2', \ldots, g_r'): & -\ b_2\ \ldots\ b_r\ \ldots\ b_{j+1} & t(c_1, \ldots, c_r)
\end{array}$$

Again, we don't have to care about the question marks because t is a near unanimity operation. We see that $t(g_1, g_2', \ldots, g_r') = h_{|\{a_2,\ldots,a_{j+2}\}}$ and because $\overline{H}$ is closed under t we have $h_{|\{a_2,\ldots,a_{j+2}\}} \in \overline{H}$.

Why is h a homomorphism? As we wrote above, it is enough to check relations of arity at most $r-1 \leq j$. Then h is a partial homomorphism iff all its restrictions to $r-1$ elements are partial homomorphisms. But $|\mathrm{dom} h| = j+2$ and all its restrictions are in $\overline{H}$, so h must be a homomorphism.

After adding h's for all $f \in \overline{H}$ such that $|\mathrm{dom}(f)| = j+1$ we obtain a set H' of partial homomorphisms that is a $(j+1, j+2)$-strategy. We can close this H' under t to obtain $\overline{H'}$ and continue. In the end, we get a $(|A|-1, |A|)$-strategy that contains a full homomorphism $\mathbb{A} \to \mathbb{B}$. □

Let us now return to the case of $\mathbb{B}$ binary relational structure. Obviously, m is a near-unanimity operation and so $\mathbb{B}$ admitting m has width 2 and CSP($\mathbb{B}$) is in P. We can also again consider the case of binary $\wedge$ and provide a more compact proof of the fact that CSP($\mathbb{B}$) is in P.

Theorem 1.3.11. *(Binary $\wedge$ revisited) If $\mathbb{B}$ is a binary relational structure that admits the polymorphism $\wedge$ then $\mathbb{B}$ has relational width 1.*

Proof. Let k be the maximum arity of $\mathcal{R}^{\mathbb{B}}$ or 2, whichever is greater. Let H be a $(1, k)$-strategy. As before, we can assume that H is closed under $\wedge$. Denote $H_{\{a\}} = \{f \in H : \mathrm{dom}(f) = \{a\}\}$ where $a \in A$. Now let $f(a) = \bigwedge H_{\{a\}}$ for each a. We claim that f is a homomorphism.

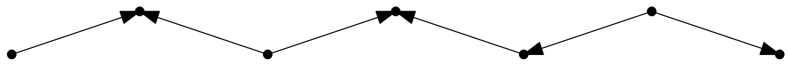

Figure 1.9: Oriented path

Let R be an n-ary relation. When $(a_1, \dots, a_n) \in R^{\mathbb{A}}$ we want $(f(a_1), \dots, f(a_n)) \in R^{\mathbb{B}}$. Obviously, the map defined only on $\{a_i\}$ as $f(a_i)$ is in H. Thus, using the $(1, k)$-forth property, we can for each i find the following set of maps (as usual, question marks are unknown elements):

value at	a_1	a_2	$\dots$	a_{n-1}	a_n
g_1 :	$f(a_1)$	?	$\dots$	?	?
g_2 :	?	$f(a_2)$	$\dots$	?	?
g_n :	?	?	$\dots$	?	$f(a_n)$
$\bigwedge_{i=1}^{n} g_i$:	$f(a_1)$	$f(a_2)$	$\dots$	$f(a_{n-1})$	$f(a_n)$

Here we have used the fact that g_i's restrictions are all in H_{a_j} and $f(a) = \bigwedge H_i\{a\}$. Because H is closed under $\wedge$, we have $\bigwedge_{i=1}^{n} g_i \in H$ and so $(f(a_1), \dots, f(a_n)) \in R^{\mathbb{B}}$, concluding the proof. □

As a side note, it is not known whether there is a structure with relational width strictly 3.

Exercise 1.3.12. Show that any binary $\mathbb{B}$ admitting one of $0, 1, \vee$ has bounded width.

Exercise 1.3.13. Show that there is a binary $\mathbb{B}$ that admits p yet does not have bounded width.

Exercise 1.3.14. Show that oriented paths (graphs as in Figure 1.3) have bounded width 1.

Exercise 1.3.15. Show that directed cycles (see Figure 1.3) have bounded width strictly 2.

1.4 CSP for algebras

In this section, we shall generalise the CSP, our goal is to strenghten the connection with universal algebra. Let us begin be giving another definition of an instance of CSP. Our new definiton shows where the name "constraint satisfaction" came from:

Definition 1.4.1. An *instance of CSP* is a triple $(V, A, \mathcal{C})$, where:

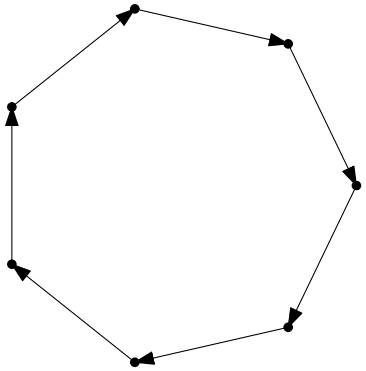

Figure 1.10: Directed cycle

- V is the set of *variables*
- A is the *domain set*
- $\mathcal{C}$ is the set of *constraints*: Each $C \in \mathcal{C}$ is a pair $C = (S, R)$ such that $S \subset V$ is the *scope of* C and $R \subset A^S$ is the *constrain relation.*

We also demand that all sets are finite. The *solution of an instance of CSP* is a map $f : V \to A$ such that $\forall (S, R) \in \mathcal{C}, f_{|S} \in R$.

As we shall see, we can straightforwardly translate this definition into the language of relational structures and back. We shall give the precise proof in a moment, for now just notice that the constraints correspond to tuples than must be mapped in a suitable relation.

This definition opens the way for another approach to CSP: Let Γ be a set of relations on the set A. Then an instance of CSP(Γ) is any instance of CSP (from the above definition) such that for all $(S, R) \in \mathcal{C}$ it is $R \in \Gamma$ after a suitable ordering of elements of S (ordering defines a bijection $R^S \to R^{|S|}$). The ordering part can be confusing, but it is merely a technical problem. We usually demand that Γ be finitely defined, otherwise CSP(Γ) is a *relative decision problem*, that is, we must trust that the input is acutally a valid instance of CSP(Γ).

Proposition 1.4.2. *If* Γ *is a finite set of relations on the set* A *then* CSP(Γ) *is poly-time equivalent to* CSP((A, Γ))*, the latter being a CSP problem for relational structures.*

Proof. Let $\mathbb{A} = (A, \Gamma)$. We show how translate instance of one problem to an instance of another in polynomial time to the size of the instance.

Let us have an instance $(V, A, \mathcal{C})$ of CSP(Γ). We want to find the corresponding instance of CSP((A, Γ)). Let in the beginning $\mathbb{B} = (V, \Gamma)$, and all the relations in $\Gamma^{\mathbb{B}}$ be empty relations. Notice that every scope S of a constraint (S, R) has by definition an ordering (such that after this ordering it is $R \in \Gamma$)

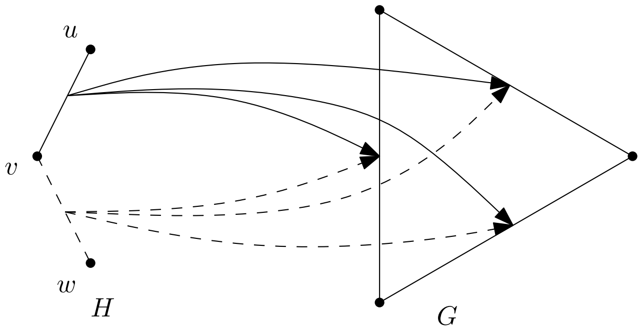

Figure 1.11: Going from one CSP to another

associated to it and we can view the ordered set $S \subset V$ as a tuple. For every $(S, R) \in \mathcal{C}$ we add this tuple S into $R^{\mathbb{B}}$.

After we add all the constraints, we obtain some $\mathbb{B}$ such that $f : \mathbb{B} \to \mathbb{A}$ is a homomorphism iff for every S scope of (S, R) it is $f(S) \in R$ (we again view S as a tuple and R as a subset of $A^{|S|}$). This condition is precisely the same as $f_{|S} \in R$, so we have a polynomial-time reduction of CSP(Γ) to CSP$((A, \Gamma))$.

On the other hand, if $\mathbb{B} = (B, \Gamma^{\mathbb{B}})$ is a relational structure, then for every $(b_1, \dots, b_k) \in R^{\mathbb{B}}$ we create a new constraint $(\{b_1, \dots, b_k\}, R)$ where R is the subset of $A^{\{b_1,\dots,b_k\}}$ corresponding in the natural way to $R^{\mathbb{A}} \subset A^k$. It is easy to see that the resulting instance $(V, A, \mathcal{C})$ has a solution iff $\mathbb{B} \in$ CSP$((A, \Gamma))$. □

To make the above proof a bit easier to swallow, we now present an example of turning an instance of CSP$((A, \Gamma))$ to an instance of CSP(Γ)

Example 1.4.3. Let Γ contain only one relation: the set E^G of all edges of a graph G on the vertex set A. Then $G = (A, \Gamma)$ is a relational structure. Given a graph $H = (V, E^H)$, we produce for each $(u, v) \in E^H$ the new constraint C_{uv} with scope $S_{uv} = \{u, v\}$ and the constraint relation $(f(u), f(v)) \in E^G$, obtaining an instance of CSP $(V, A, \mathcal{C})$.

We will now define CSP for a finite algebra $\underline{\mathrm{A}}$ in two ways: For relational structures and for CSP(Γ). Both notions are quite similar, only the language is different. Also, the first set of definitions is more elementary and thus perhaps easier to understand..

Definition 1.4.4. Let $\underline{\mathrm{A}} = (A, \mathcal{F})$ be an algebra. We say that a relational structure $(A, \mathcal{R})$ is compatible with $\underline{\mathrm{A}}$ if it is $\mathcal{R} \subset \mathrm{Inv}(\mathcal{F})$, i.e. $\mathcal{F} \subset \mathrm{Pol}(\mathcal{R})$.

Definition 1.4.5. Let $\underline{\mathrm{A}}$ be an algebra. Then

$$\mathrm{CSP}(\underline{\mathrm{A}}) = \{(\mathbb{A}, \mathbb{B}) | \mathbb{B} \text{ is compatible with } \underline{\mathrm{A}} \text{ and } \mathbb{A} \in \mathrm{CSP}(\mathbb{B})\}.$$

We say that $\underline{\mathrm{A}}$ is *globally tractable* if CSP$(\underline{\mathrm{A}})$ is in P and that $\underline{\mathrm{A}}$ is *locally tractable* if for every $\mathbb{B}$ compatible with $\underline{\mathrm{A}}$, CSP$(\mathbb{B})$ is in P. It is not known whether there is an algebra that is locally tractable but not globally tractable.

We now give the second definition of CSP for algebras. Strictly speaking, this problem is different from the CSP($\underline{A}$) given above (it is not a subset of pairs of relational structures), but both notions are poly-time equivalent and we shall use them interchangingly.

Definition 1.4.6. If $\underline{A}$ is an algebra then $\mathrm{CSP}(\underline{A}) = \mathrm{CSP}(\mathrm{Inv}(\underline{A}))$.

If Γ is a set of relations then we say that Γ is *globally tractable* if $\mathrm{CSP}(\Gamma)$ is in P. We say that Γ is *locally tractable* if for every finite $\Gamma_0 \subset \Gamma$, $\mathrm{CSP}(\Gamma_0)$ is in P. An algebra $\underline{A}$ is globally resp. locally tractable iff $\mathrm{Inv}(\underline{A})$ is globally resp. locally tractable. Thanks to Proposition 1.4.2 we have the following observation:

Observation 1.4.7. *An algebra $\underline{A}$ is locally resp. globally tractable according to the first definition iff it is locally resp. globally tractable according to the second definition.*

We shall often reduce CSP of one algebra to CSP of another algebra. It is important to notice that there are actually two kinds of reductions:

Definition 1.4.8. If for every $\mathbb{A}$ compatible with an algebra $\underline{A}$ exists a $\mathbb{B}$ compatible with an algebra $\underline{B}$ such that $\mathrm{CSP}(\mathbb{A})$ is poly-time reducible to $\mathrm{CSP}(\mathbb{B})$, then we say that $\mathrm{CSP}(\underline{A})$ is *locally reducible* to $\mathrm{CSP}(\underline{B})$.

Notice that the reducing algorithm need not be the same for each $\mathrm{CSP}(\mathbb{A})$.

Definition 1.4.9. If there exists a polynomial algorithm that for every instance of $\mathrm{CSP}(\underline{A})$ produces an instance of $\mathrm{CSP}(\underline{B})$ such that the first instance has a solution iff the other has a solution, then we say that $\mathrm{CSP}(\mathbb{A})$ is *globally reducible* to $\mathrm{CSP}(\mathbb{B})$.

We shall use local reduction in situations where the globall reduction algorithm provides us with a suitable $\mathbb{B}$ for any $\mathbb{A}$ but it is not polynomial. Notice that if there is $\mathbb{A}$ compatible with $\underline{A}$ such that $\mathrm{CSP}(\mathbb{A})$ is hard (say, NP-complete) and $\underline{A}$ is locally reducible to $\underline{B}$ then there is a $\mathbb{B}$ compatible with $\underline{B}$ then $\mathrm{CSP}(\mathbb{B})$ is also hard and so $\mathrm{CSP}(\underline{B})$ is at least as hard as $\mathrm{CSP}(\mathbb{A})$.

Lemma 1.4.10. *Let $\mathbb{B} = (B, \mathcal{R})$ be a relational structure, $\underline{B} = (B, \mathrm{Pol}(\mathcal{R}))$ its algebra. Then whenever $\mathbb{A}$ is compatible with $\underline{B}$, $\mathrm{CSP}(\mathbb{A})$ is poly-time reducible to $\mathrm{CSP}(\mathbb{B})$.*

Proof. Let $\mathbb{A} = (B, \mathcal{S})$. We know that $\mathcal{S} \subset \mathrm{Inv}(\mathrm{Pol}(\mathcal{R})) = \langle \mathcal{R} \rangle$. Thus $\langle \mathcal{S} \rangle \subset \langle \mathcal{R} \rangle$ and the result follows from Theorem 1.2.22. □

One advantage of considering CSP for algebras is that it goes well together with our previous results about structures admiting certain operations. For example, if $\underline{A}$ has a k-ary near unanimity operation then for all $\mathbb{A}$ compatible with $\underline{A}$, $\mathrm{CSP}(\mathbb{A})$ has width $k-1$ (due to Lemma 1.3.10) and so $\underline{A}$ is locally tractable.

We are going to generalise the theory of bounded width for algebras and also show that some algebras' CSP is NP-complete. We shall also show that $\underline{A}$ from the above paragraph is even globally tractable.

Definition 1.4.11. An instance $(V, A, \mathcal{C})$ is k-minimal if

- $\forall K \subset V, |K| \leq k \Rightarrow \exists (S, R) \in \mathcal{C}$ such that $K \subset S$.
- $\forall (S_1, R_1), (S_2, R_2) \in \mathcal{C}, K \subset S_1 \cap S_2, |K| \leq k \Rightarrow R_{1|K} = R_{2|K}$.

The notion of k-minimal instance, while slightly more complicated, is quite similar to the notion of $(j;k)$-strategy.

Definition 1.4.12. An algebra $\underline{\mathrm{A}}$ has relational width k if every k-minimal instance of $\underline{\mathrm{A}}$ in which all constraint relations are non-empty has a solution.

Algorithm 1.4.13 (Local Consistency)**.** Every instance of CSP can be reduced to a k-minimal instance in polynomial time (for k fixed, i.e. not part of the input) so that the original has a solution iff the reduced instance has a solution. Also, if our original instance's relations were in $\mathrm{Inv}(\underline{\mathrm{A}})$ for some algebra $\underline{\mathrm{A}}$ then so are the relations of the produced k-minimal instance.

Proof. First, for every $K \subset V, |K| = k$ take (K, A^k) as new constraints. This is an easy way to ensure that the first condition is met without changing the solution. Now we must remove tuples from constraint relation so that we satisfy the second condition – we do this by brute force checking all the possible $(S_1, R_1), (S_2, R_2), K$ and removing from R_1 all the tuples r such that $r_{|K} \in R_{1|K} \setminus R_{2|K}$. The number of checks necessary is polynomial in the size of $(V, A, \mathcal{C})$.

Observe that the added constraints do not limit the solution in any way and that when we remove a tuple from a constraint relation then the tuple can not be used by a solution anyway. So our new instance has a solution iff the original instance has a solution.

It remains to see that the new relations are all in $\mathrm{Inv}(\underline{\mathrm{A}})$. The newly added constraints (K, A^k) are certainly $\underline{\mathrm{A}}$-invariant, so it remains to check that we did not break anything by removing tuples. Assume that l-th removal of tuples has violated the $\underline{\mathrm{A}}$-invariance and that l is the smallest such number. Let, as above, (S_1, R_1), (S_2, R_2), K be the witness for the removal. Then for some $r_1, \ldots, r_n \in R_1$ and some n-ary operation f of the algebra we have removed $r = f(r_1, \ldots, r_n)$ and kept r_i's. But that can only happen if $r_{|K}$ is superfluous (i.e. $r_{|K} \notin R_{2|K}$) and $r_{i|K}$'s are not. But this means that $r_{i|K} \in R_{2|K}$ and because of f-invariance of R_2, it is $r_{|K} = f(r_{1|K}, \ldots, r_{n|K}) \in R_{2|K}$, a contradiction. $\square$

Corollary 1.4.14. *If $\underline{\mathrm{A}}$ has relational width k then $\underline{\mathrm{A}}$ is globally tractable (*$\mathrm{CSP}(\underline{\mathrm{A}})$ *is in P).*

The above algorithm has strong similarity to the local consistency algorithm from previous section, the one that produced a (j, k)-strategy for a given relational structure. For example, in the proof of the next theorem we will produce a $(j-1, j)$-strategy using a j-minimal presentation. However, the precise relation between the relational width of structures and algebras is as yet unclear:

Open problem 1.4.15. Let $\underline{\mathrm{A}}$ be a finite algebra such that every $\mathbb{A}$ compatible with $\underline{\mathrm{A}}$ has relational width k. Does it follow that $\underline{\mathrm{A}}$ has relational width k?

Theorem 1.4.16. *If $\underline{A}$ admits r-ary local unanimity operation t then $\underline{A}$ has relational width r.*

Proof. Due to proof of Lemma 1.3.10 it is enough to show that any r-minimal nonempty instance of CSP($\underline{\mathrm{A}}$) admits a $(r-1,r)$-strategy. But that is easy: Take the set $\bigcup\{R_{|K}|K \subset S, (S,R) \in \mathcal{C}, |K| \leq r\}$. It is easy to verify that this is a $(r-1,r)$-strategy which can be, using the method from the proof of Theorem 1.3.10, extended to a $(|V|-1,|V|)$-strategy. □

Remark 1.4.17. Inspired by the case of relational width for relational structures, one might hope to prove that $\underline{\mathrm{A}}$ from the above theorem has relational width $r-1$. That is, however, not true: Consider the algebra $A = \{a,b,c,d,e,f\}$ with $R_1 = \{(a,b)(c,d)\}, R_2 = \{(e,b),(f,d)\}$ and $R_3 = \{(f,a),(e,c)\}$. These relations are invariant under any $t(x,y,z)$ near-unanimity. Taking $V = \{1,2,3\}$ and constraints $((2,3),R_1),((1,3),R_2),((1,2),R_3)$, we have a nonempty 2-minimal instance that does not have a solution.

Our previous theorems about binary relational structures can be extended to binary algebras as well:

Theorem 1.4.18. *Let $\underline{A}$ be an algebra, $|A| = 2$. Then* CSP($\underline{A}$) *is NP-complete iff all terms of $\underline{A}$ are projections or permutations of projections and* CSP($\underline{A}$) *is in P (globally tractable) otherwise.*

Proof. The first case is a direct consequence of Lemma 1.2.45. We also know that in the other case CSP($\underline{\mathrm{A}}$) is globally tractable: Depending on which of the terms $0, 1, \vee, \wedge, \mathrm{p}, \mathrm{m}$ are contained in $\underline{\mathrm{A}}$, we just run the correct (polynomial-time) algorithm to solve all the instances of CSP($\underline{\mathrm{A}}$). [something is missing here] □

Recall that in universal algebra, if $\mathcal{C}$ is a class of algebras then S$\mathcal{C}$ is the class of all subalgebras of algebras of $\mathcal{C}$, P$\mathcal{C}$ the class of all the powers of algebras of $\mathcal{C}$, $\mathrm{P}_{fin}\mathcal{C}$ the class of all the *finite* powers of algebras of $\mathcal{C}$ and H$\mathcal{C}$ is the class of all homomorphic images of algebras of $\mathcal{C}$. Then the *variety* of $\mathcal{C}$ is the class HSP$\mathcal{C}$. It is the smallest class of algebras containing $\mathcal{C}$ closed under homomorphic images, subalgebras and powers. This is a very important construction in universal algebra.[something is missing here]A variety is called *locally finite* if every its finitely generated algebra is finite.[something is missing here]

Now we want to show that if $\underline{\mathrm{B}}$ is an algebra and $\underline{\mathrm{A}} \in \mathrm{H}\underline{\mathrm{B}} \cup \mathrm{S}\underline{\mathrm{B}} \cup \mathrm{P}_{fin}\underline{\mathrm{B}}$ then CSP($\underline{\mathrm{A}}$) is locally reducible to CSP($\underline{\mathrm{B}}$). This will mean that CSP of all the algebras in $\mathrm{HSP}_{fin}\underline{\mathrm{B}}$ is locally reducible to CSP of $\underline{\mathrm{B}}$. We shall do so in three lemmas, some of which offer even a global reduction.

Lemma 1.4.19. *If $\underline{A}$ is a subalgebra of $\underline{B}$ then* CSP($\underline{A}$) *is globally poly-time reducible to* CSP($\underline{B}$).

Proof. Let us have $\mathbb{A} = (A, \mathcal{R}^{\mathbb{A}})$ compatible with $\underline{\mathrm{A}}$. By adding more elements to A, we obtain the structure $\mathbb{B} = (B, \mathcal{R}^{\mathbb{A}})$. We claim that $\mathbb{C} \in \mathrm{CSP}(\mathbb{A})$ iff $\mathbb{C} \in \mathrm{CSP}(\mathbb{B})$.

Let $f : \mathbb{C} \to \mathbb{A}$ be a homomorphism. Then by extending the range set we obtain a homomorphism $\mathbb{C} \to \mathbb{B}$. If, on the other hand, $f : \mathbb{C} \to \mathbb{B}$ is a homomorphism and $f(c) \in B \setminus A$ then c can not lie in any relation (because $f(c)$ does not lie in any relation). Choose an element $a \in A$ (remember that algebra can not be empty) and consider $g : \mathbb{C} \to \mathbb{A}$ defined as $g(c) = f(c)$ for $f(c) \in A$ and $g(c) = a$ otherwise. Then g is a homomorphism and we are done. □

Lemma 1.4.20. *For any $n \in \mathbb{N}$, $\mathrm{CSP}(\underline{A}^n)$ is globally poly-time reducible to $\mathrm{CSP}(\underline{A})$*

Proof. Let us have $I = (V, A^n, \mathcal{C})$ instance of $\mathrm{CSP}(\underline{\mathrm{A}}^n)$. Take any $(S, R) \in \mathcal{C}$. For each n, we have an isomorphism $(\underline{\mathrm{A}}^n)^S \simeq \underline{\mathrm{A}}^{[n]\times S}$ where $[n]$ is the set $\{1, 2, \dots, n\}$; instead of $r(s) = (r_1(s), \dots, r_n(s))$ we can take $r'(i, s) = r_i(s)$. For each $R \subset (\underline{\mathrm{A}}^n)^S$, let R' denote the image of R under this isomorphism. Consider the set $\mathcal{C}' = \{([n] \times S, R')|(S, R) \in \mathcal{C}\}$ and the instance $I' = ([n] \times V, A, \mathcal{C}')$. This is an instance of $\mathrm{CSP}(\underline{\mathrm{A}})$, because if R is $\underline{\mathrm{A}}^n$-invariant then R' is $\underline{\mathrm{A}}$-invariant.

It remains to show that I has a solution iff I' has a solution. But that is easy: A function $f \in (A^n)^V$ is a solution of the first problem iff the corresponding function $f' \in A^{[n]\times V}$ is a solution of the second problem. □

Before stating the third lemma, let us remember that $\underline{\mathrm{B}}$ is a homomorphic image of $\underline{\mathrm{A}}$ iff $\underline{\mathrm{B}} \simeq \underline{\mathrm{A}}/\theta$ where θ is a *congruence* on $\mathbb{A}$, that is a relation invariant under all the operations of $\mathbb{A}$. (We could also write $\theta \in \mathrm{Inv}(\mathbb{A})$ but the term congruence is used much more often in this context.)

Lemma 1.4.21. *Let $\underline{A}$ be an algebra, θ a congruence on $\underline{A}$. Then $\mathrm{CSP}(\underline{A}/\theta)$ is locally reducible to $\mathrm{CSP}(\underline{A})$*

Proof. Here, the degree of the polynomial bounding the run-time of the reduction will depend on the arity of the relations involved, so we do not construct a global reduction.

Let $\mathbb{A} = (A/\theta, \mathcal{R}^{\mathbb{A}})$ be a relational structure compatible with $\underline{\mathrm{A}}/\theta$. We shall define $\mathbb{A}' = (A, \mathcal{R}^{\mathbb{A}'})$ in the following way: For each n-ary $R \in \mathcal{R}$ let

$$R^{\mathbb{A}'} = \bigcup \left\{ a_1/\theta \times \cdots \times a_n/\theta | (a_1/\theta, \dots, a_n/\theta) \in R^{\mathbb{A}} \right\}.$$

Note that we understand a/θ as a set here, so $R^{\mathbb{A}'}$ is an n-ary relation on A. This is a set of "θ-blocks" (see figure). Because $R^{\mathbb{A}}$ was invariant under $\underline{\mathrm{A}}/\theta$, each $R^{\mathbb{A}'}$ is a subalgebra of $\underline{\mathrm{A}}^n$.

We now claim that $\mathbb{C} \in \mathrm{CSP}(\mathbb{A})$ iff $\mathbb{C} \in \mathrm{CSP}(\mathbb{A}')$. Let first $f : \mathbb{C} \to \mathbb{A}$ be a homomorphism. Then we take a representative for each θ-block and construct a mapping $g : \mathbb{C} \to \mathbb{A}'$ such that $g(c) = a$ iff $f(c) = a/\theta$. This is a homomorphism because when $(c_1, \dots, c_n) \in R^{\mathbb{C}}$ then $(f(c_1), \dots, f(c_n)) = ([a_1], \dots, [a_n]) \in R^{\mathbb{A}}$ and so $(a_1, \dots, a_n) \in R^{\mathbb{A}'}$. On the other hand, if $g : \mathbb{C} \to \mathbb{A}'$ is a homomorphism then for each $(c_1, \dots, c_n) \in R^{\mathbb{C}}$ we have $([g(c_1)], \dots, [g(c_n)]) \in R^{\mathbb{A}}$, so $f(c) = [g(c)]$ defines a homomorphism $f : \mathbb{C} \to \mathbb{A}$ and the proof is complete.

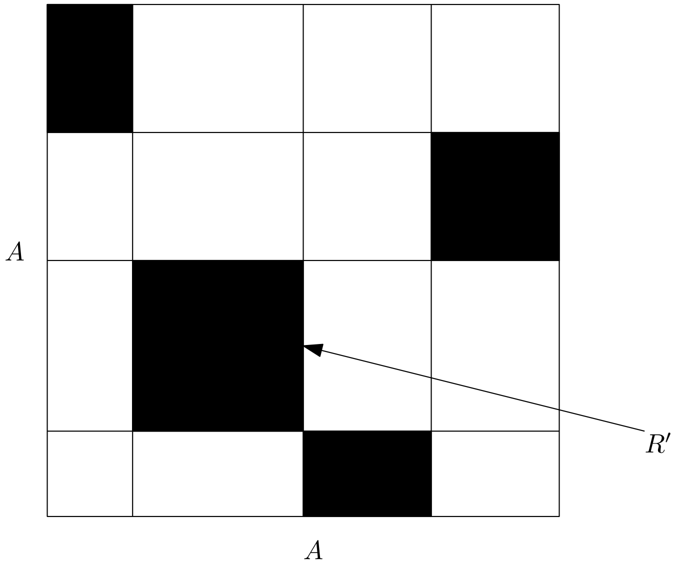

Figure 1.12: Blocks of θ

Notice that this correspondence is only local because "unpacking" of the θ-blocks has time complexity $O(|A|^n)$ (where n is the maximum arity of $\mathcal{R}$) in the worst case. □

Exercise 1.4.22. Prove in detail that $R^{\mathbb{A}'}$ are $\underline{\mathrm{A}}$-invariant.

We now mention one more lemma here, because it is of a kind similar to the three previous ones. An *unary polynomial* of an algebra $\underline{\mathrm{A}}$ is any map $f : \underline{\mathrm{A}} \to \underline{\mathrm{A}}$ such that there exists t in the functional clone $\langle \underline{\mathrm{A}} \rangle$ of $\underline{\mathrm{A}}$ (i.e. t is a composition of projections and operations from $\underline{\mathrm{A}}$) such that $p(x) = t(x, a_1, \ldots, a_n)$ for some constants $a_1, \ldots, a_n \in \underline{\mathrm{A}}$. We can then define the algebra $p(\underline{\mathrm{A}})$ as $(p(A), \{p(f(x_1, \ldots, x_n)) | f \in \langle \underline{\mathrm{A}} \rangle\})$ and the following lemma gives us a tool how to go down from $\underline{\mathrm{A}}$ to $p(\underline{\mathrm{A}})$ and still maintain an upper bound on complexity.

Lemma 1.4.23. CSP($p(\underline{A})$) *is locally poly-time reducible to* CSP($\underline{A}$)

Proof. Let us take an instance $I = (V, \mathcal{C})$ of CSP($p(\underline{\mathrm{A}})$). We want to produce an instance $I' = (V, \mathcal{C}')$ of CSP($\underline{\mathrm{A}}$).

For each $C = (S, R) \in \mathcal{C}$ we define $C' = (S, R')$ where R' is the subalgebra generated by R in $\underline{\mathrm{A}}^S$ (this is the place where a global reduction would fail, however if the arity of all R's is bounded, this step still has polynomial time complexity). Obviously, $R \subset R'$. We claim that $p(R') \subset R$. Let $r \in R'$. Then $r = t(s_1, \ldots, s_k)$ where t is an $\underline{\mathrm{A}}$-term and s_i belong to R. But then $p(r) = p(t(s_1, \ldots, s_k))$. We know that $p \circ t$ is a term from $p(\underline{\mathrm{A}})$ and so $p(r) \in R$.

Now we want to show that I has a solution iff I' has a solution. Assume we have a solution of I. Then because $R \subset R'$ it is also a solution of I'. If now f

is a solution of I', we take $p \circ f$ and claim that this is a solution of I. When $C = (S, R)$ is a constraint in I then $f_{|S} \in R'$ and so $p \circ f_{|S} \in p(R') \subset R$ meaning that we indeed have a solution.

If we are given a relational structure $\mathbb{A} = (p(A), R^{\mathbb{A}})$ compatible with $p(\underline{A})$ then we obtain a reducion by taking $\mathbb{A}' = (A, \{R' | R \in R^{\mathbb{A}}\})$. □

Recall that idempotent operations helped us to solve the case of binary relational structures. We say that an algebra is *idempotent* if all its operations are idempotent. We shall now use theorems from universal algebra to find a class of idempotent algebras whose CSP is NP-complete.

Theorem 1.4.24. *If every term of the nontrivial algebra $\underline{A}$ is a projection then there exists $\mathbb{A}$ compatible with $\underline{A}$ such that* CSP($\mathbb{A}$) *is NP-complete.*

Proof. We are more or less proving Lemma 1.2.45 again (under slightly different conditions): $\underline{A}$ allows many relations so we can construct $\mathbb{A}$ so that CSP($\mathbb{A}$) is one of known NP-complete problems.

Assume first that $|A| = 2$. Then we produce $\mathbb{A}$ such that CSP($\mathbb{A}$) is exactly 3-SAT, that is $\mathbb{A} = (A, \mathcal{R})$ where $\mathcal{R}$ consists of all the binary relations of arity three.

If now $|A| \geq 3$ then we produce $\mathbb{A} = (A, \mathcal{R})$ with $\mathcal{R}$ containing only one relation $R = \{(a, b) | a, b \in A, a \neq b\}$, giving us the $|A|$-coloring problem which is also NP-complete. □

Definition 1.4.25. We say that $t(x_1, \ldots, x_n)$ is a weak near-unanimity operation if:

- $n \geq 2$
- $t(x, \ldots, x) = x$
- $t(y, x, \ldots, x) = t(x, y, x, \ldots, x) = \cdots = t(x, \ldots, x, y)$

Observe that every near-unanimity operation is also a weak near-unanimity operation but the converse need not be true.

We shall now without proof introduce and use two universal algebra theorems.

We say that a variety μ has a term t of certain properties (for example, a weak near-unanimity term) iff every algebra $\underline{A} \in \mu$ has such a term. For the purposes of this lecture, *Taylor term* will be a little black box that either is present in a variety or not. A *type 1* will be a different kind of black box.

Theorem 1.4.26. *If μ is an idempotent locally finite variety without a Taylor term, then $\exists \underline{A} \in \mu$ nontrivial algebra such that every term operation of $\underline{A}$ is a projection.*

Theorem 1.4.27. *Let μ be a locally finite variety. Then the following are equivalent:*

- *μ has a Taylor term.*

- *μ has a weak near-unanimity operation.*
- *μ omits type 1.*

Putting theorems 1.4.26 and 1.4.27 together we obtain the following theorem whose formulation does not requie any universal algebra at all:

Theorem 1.4.28. *If $\underline{A}$ is an idempotent algebra that has no weak near-unanimity term then there exists $\mathbb{A}$ compatible with $\underline{A}$ such that* CSP($\mathbb{A}$) *is NP-complete.*

Proof. If $\underline{A}$ has no weak near-unanimity term then $\underline{A}$ generates a locally finite idepotent variety $\mu = \mathrm{HSP}_{fin}(\underline{A})$ that has no weak near-unanimity term. Now Theorem 1.4.27 gives us that μ has no Taylor term and using Theorem 1.4.26 we see that in μ there exists a nontrivial algebra $\underline{B}$ such that every term operation of $\underline{B}$ is a projection. Using Theorem 1.4.24 we get that there exists $\mathbb{B}$ compatible with $\underline{B}$ such that CSP($\mathbb{B}$) is NP-complete. Now we use the fact that CSP($\underline{B}$) is locally reducible to CSP($\underline{A}$) to see that there exists $\mathbb{A}$ compatible with $\underline{A}$ such that CSP($\mathbb{A}$) is NP-complete and the proof is done. □

Note that it is not enough to demand that $\underline{A}$ does not contain a weak near-unanimity *operation*, because operations can be composed to form a term.

The following conjecture would give us dichotomy for all algebras. Unfortunately, the proof is unknown.

Conjecture 1.4.29. If the algebra $\underline{A}$ is not idempotent or it contains a weak near-unanimity term then CSP($\underline{A}$) is in P.

1.5 Further topics

In this section we shall mention some more advanced results about CSP. It will be mostly an overview with pointers to articles that dissect various problems in more detail. Most of these articles are freely available online.

1.5.1 Mal'tsev term

Definition 1.5.1. We say that $p(x, y, z)$ is a *Mal'tsev term* if

- $p(x, x, y) = y$
- $p(x, y, y) = x.$

An example of a Mal'tsev term would be the p operation on binary algebras, in a general group we could take $p(x, y, z) = xy^{-1}z$.

Theorem 1.5.2. *If $\underline{A}$ has a Mal'tsev term then* CSP($\underline{A}$) *is in P.*

The construction of an algorithm that solves such CSP($\underline{A}$) can be found in the article [3]. The main ingredient there is the use of *compact representation* of a relation – instead of full R, we consider only its subset that in a certain sense generates R.

1.5.2 Congruence distributivity

A lattice is *distributive* if for all $x, y, z \in L$ it is $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$.

Exercise 1.5.3. Show that a lattice is distributive iff $\forall x, y, z \in L, x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$.

We say that a variety μ is *congruence distributive* if for each $A \in \mu$ the lattice of congruences on A is distributive. Notice that for congruences α, β, it is $\alpha \wedge \beta = \alpha \cap \beta$ and $(x, y) \in \alpha \vee \beta$ if there exists a chain of pairs $(x, z_1), (z_1, z_2), \ldots, (z_{n-1}, y)$ such that each pair is in α or β. This observation allows us to prove another characterisation of congruence distributive varieties.

Theorem 1.5.4 (Jónsson)**.** *A variety μ is congruence distributive iff there exist ternary terms $p_0, p_1, \ldots, p_n$, so-called* Jónsson terms*, satisfying the identities:*

- $p_0(x, y, z) = x$
- $p_i(x, y, x) = x$ *for all i.*
- $p_i(x, x, y) = p_{i+1}(x, x, y)$ *for i even*
- $p_i(x, y, y) = p_{i+1}(x, y, y)$ *for i odd*
- $p_n(x, y, z) = z.$

Proof. Assume first that μ is congruence distributive. Let $\underline{\mathrm{A}}$ be the 3-generated *free algebra* in μ. That is, the set of all terms using three variables (say, x, y, z) such that the only identities in $\underline{\mathrm{A}}$ are the ones that hold in the whole variety μ.

We now define three congruences $\alpha = \mathrm{Cg}(x, y), \beta = \mathrm{Cg}(y, z), \gamma = \mathrm{Cg}(x, z)$ as the congruences defined by identifying the respective pairs of variables. For example, $\mathrm{Cg}(x, y)$ is the congruence obtained by assuming $x = y$. From the congruence distributivity condition, we obtain that

$$(\alpha \vee \beta) \wedge \gamma = (\alpha \wedge \gamma) \vee (\beta \wedge \gamma)$$

Because $(x, z) \in \gamma$ and $(x, z) \in \alpha \vee \beta$ (because from $x = y$ and $y = z$ follows that $x = z$), we have that $(x, z) \in (\alpha \wedge \gamma) \vee (\beta \wedge \gamma)$. This means that there exists a chain of pairs $(x, p_1), (p_1, p_2), \ldots, (p_{n-1}, z)$ such that without loss of generality $(p_{2i}, p_{2i+1}) \in \alpha \wedge \gamma$ and $(p_{2i+1}, p_{2i+2}) \in \beta \wedge \gamma$, see figure.

As noted above, the elements of $\underline{\mathrm{A}}$ are actually terms on three variables, so we can write each $p_i(x, z, y)$ as a composition of operations in μ. Note that p_i is not yet a function, just a sequence of operation symbols. However, we claim that for each $\underline{\mathrm{B}} \in \mu$ and each choice of $x, y, z \in \underline{\mathrm{B}}$, the terms $x, p_1(x, y, z), \ldots, p_{n-1}(x, y, z), z$ as evaluated in $\underline{\mathrm{B}}$ are Jónsson. (Note that we can identify x with the projection $p_0(x, y, z) = x$.)

At this point, we should note that by selecting $x, y, z \in \underline{\mathrm{B}}$, we obtain the obvious homomorphism $\underline{\mathrm{A}} \to \underline{\mathrm{B}}$, where every $p(x, y, z) \in \underline{\mathrm{A}}$ gets mapped to its evaluation in $\underline{\mathrm{B}}$. This means that all the identities in $\underline{\mathrm{A}}$ are true in $\underline{\mathrm{B}}$ and,

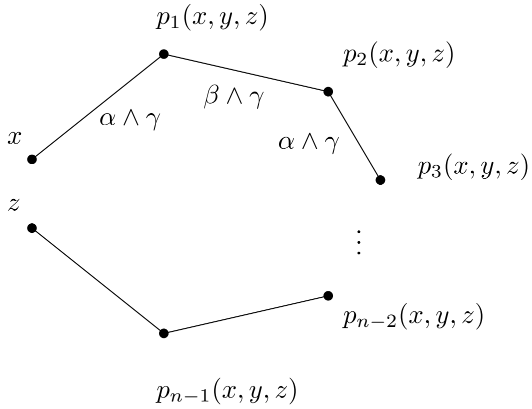

Figure 1.13: Chain of relations

moreover, if $x = y$ and $(p,q) \in \mathrm{Cg}(x,y)$, it is $p(x,y,z) = q(x,y,z)$ in $\underline{\mathrm{B}}$. Details of this construction can be found in [1].

Conditions (i) and (v) are trivial. Also, all the pairs (p_i, p_{i+1}) are in the congruence γ and so, by transitivity, for each i it is $(p_i, x) \in \gamma$, meaning that after identifying x and z it is $p_i(x,y,x) = x$, giving us (ii). When i is even, it is $(p_i, p_{i+1}) \in \alpha$ and so $p_i(x,x,z) = p_{i+1}(x,x,z)$, giving us (iii). Equality (iv) follows in similar way from $(p_i, p_{i+1}) \in \beta$ for i odd.

Conversely, let μ contain Jónsson terms $p_0, p_1, \ldots, p_n$ for some n. Assume that we have congruences τ, κ, σ on algebra $\underline{\mathrm{A}}$. We want to prove that then

$$(\tau \vee \kappa) \wedge \sigma = (\tau \wedge \sigma) \vee (\kappa \wedge \sigma).$$

We shall prove two inclusions, one of which is an easy exercise: Whenever $(a,b) \in (\tau \wedge \sigma) \vee (\kappa \wedge \sigma)$, it is $(a,b) \in \sigma$ and $(a,b) \in \tau \vee \kappa$, meaning that $(a,b) \in (\tau \vee \kappa) \wedge \sigma$.

The other inclusion is more difficult. First, we will prove a lemma:

Lemma 1.5.5. *$(\tau \circ \kappa) \wedge \sigma \subset (\tau \wedge \sigma) \vee (\kappa \wedge \sigma)$, where $\tau \circ \kappa$ is the relation defined by $(x,y) \in \tau \circ \kappa$ iff $\exists z, (x,z) \in \tau, (z,y) \in \kappa$.*

Proof. Let $(a,c) \in \sigma$ and let there exist b such that $(a,b) \in \tau, (b,c) \in \kappa$. Because σ is p_i-invariant relation and $(a,a),(b,b),(a,c) \in \sigma$, for each i we have $(p_i(a,b,a), p_i(a,b,c)) \in \sigma$. But due to (ii), it is $p_i(a,b,a) = a$, so $(a, p_i(a,b,c)) \in \sigma$. Because σ is a congruence, it is symmetric and transitive and so $(p_i(a,b,c), p_{i+1}(a,b,c)) \in \sigma$. Similarly, for all i it is $(p_i(a,b,c), p_{i+1}(a,a,c)) \in \theta$ and $(p_i(a,b,c), p_{i+1}(a,c,c)) \in \kappa$.

If i is odd, it is $p_i(a,c,c) = p_{i+1}(a,c,c)$ and so from transitivity we obtain that $(p_i(a,b,c), p_{i+1}(a,b,c)) \in \kappa$. Similarly, for i even, it is $(p_i(a,b,c), p_{i+1}(a,b,c)) \in \theta$. All in all, we have a chain of elements $p_i(a,b,c)$ from a to $p_n(a,b,c) = c$ (see figure), proving that $(a,c) \in (\tau \wedge \sigma) \vee (\kappa \wedge \sigma)$.

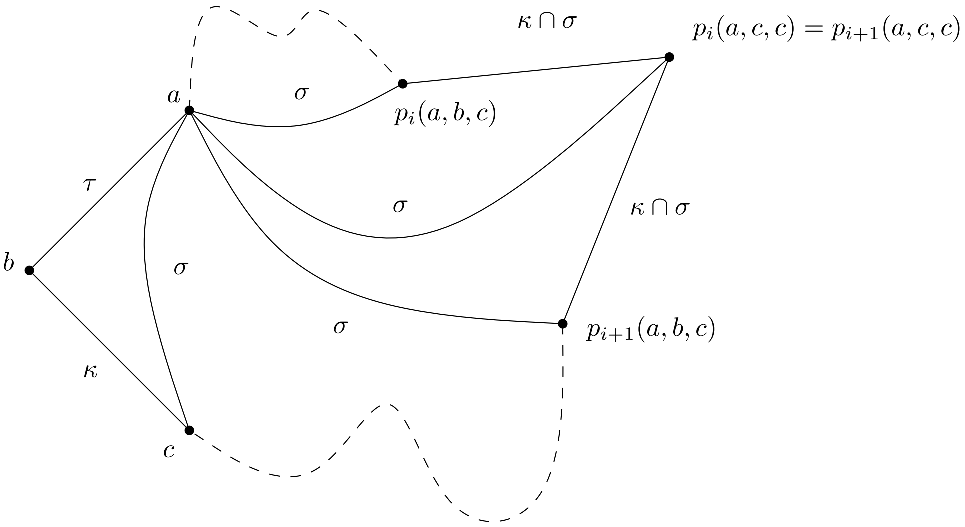

Figure 1.14: Congruences

□

Observe now that

$$\tau \vee \kappa = \tau \circ \kappa \cup \tau \circ \kappa \circ \tau \circ \kappa \cup \dots$$

and so it is enough to show that for each n it is $(\tau \circ \kappa)^n \wedge \sigma \subset (\tau \wedge \sigma) \vee (\kappa \wedge \sigma)$. We will prove this by induction on n.

Let for ω^n denote the relation $\omega \circ \cdots \circ \omega$, where the number of ω's is n. To prove the theorem, we need to show that for each n positive integer it is

$$(\kappa \circ \tau)^n \wedge \sigma \subset (\kappa \circ \tau) \wedge \sigma \subset (\tau \wedge \sigma) \vee (\kappa \wedge \sigma).$$

Notice that the second inclusion is precisely Lemma 1.5.5, so we only have to prove the first inclusion.

The statement clearly holds for $n = 1$. Assume that it holds for some n. Then for $n+1$ it is:

$$(\kappa \circ \tau)^{n+1} \wedge \sigma = ((\kappa \circ \tau)^n \circ (\kappa \circ \tau)) \wedge \sigma \subset ((\kappa \circ \tau)^n \wedge \sigma) \vee ((\kappa \circ \tau) \wedge \sigma) \subset (\kappa \circ \tau) \wedge \sigma,$$

where the first inclusion follows from Lemma 1.5.5 and the second one follows from the induction assumption. □

Exercise 1.5.6. Prove in detail that for each τ, κ congruences it is

$$\tau \vee \kappa = \tau \circ \kappa \cup \tau \circ \kappa \circ \tau \circ \kappa \cup \dots$$

Theorem 1.5.4 allows us to say that a variety is CD(n) if it is congruence distributive and the n is the minimum number of Jónsson terms from the above theorem.

Exercise 1.5.7. We say that a variety μ is *congruence permutive* if for all algebras $\underline{A} \in \mu$ and all α, β congruences on $\underline{A}$ it is $\alpha \circ \beta = \beta \circ \alpha$. Prove that $\underline{A}$ is congruence permutive iff $\underline{A}$ contains a Mal'tsev term.

Observation 1.5.8. *If μ has a near-unanimity term then μ is congruence distributive.*

Proof. Let t be a near unanimity term of algebra $\underline{A} \in \mu$. Let

$$\begin{aligned}
p_0(x,y,z) &= t(z,x,x,x,\dots,x) = x \\
p_1(x,y,z) &= t(z,y,x,x,\dots,x) \\
p_2(x,y,z) &= t(z,z,x,x,\dots,x) \\
p_3(x,y,z) &= t(z,z,y,x,\dots,x) \\
&\vdots \\
p_n(x,y,z) &= z.
\end{aligned}$$

We claim that this is a set of Jónsson terms. It is obvious that (i), (ii) and (v) hold. Furthermore, it is

$$p_{2i}(x,x,y) = t(y,\dots,y,x,\dots,x) = p_{2i+1}(x,x,y),$$

proving (iii), and

$$p_{2i+1}(x,y,y) = t(y,\dots,y,x,\dots,x) = p_{2i+2}(x,y,y),$$

proving (iv). The reader can easily verify that the number of x and y variables is indeed the same on both sides. □

Theorem 1.5.9. *If $\underline{B}$ is an algebra in a CD(4) variety then every $\mathbb{B}$ compatible with $\underline{B}$ has relational width at most the maximum arity of $\mathbb{B}$.*

For proof of this theorem, see article [2].

1.5.3 CSP for graphs

The graph homomorphism problem is for some time a point of interest of combinatorians. For an overview of combinatorial results about graph homomorphisms and their properties, see [4].

It turns out that there is not much more to CSP than graph homomorphisms, as the following theorem shows:

Theorem 1.5.10. *Balanced digraph homomorphism problem is poly-time equivalent to CSP.*

For proof, see [something is missing here]

For some classes of graphs, we know precisely which CSP problems are in P and which are NP-complete.

Theorem 1.5.11 (Nešetřil, Hell, 1990)**.** *For G symmetric graph, it is* $\mathrm{CSP}(G)$ *in P iff G is bipartite. Otherwise,* $\mathrm{CSP}(G)$ *is NP-complete.*

For proof, see [something is missing here]. Note that the original proof of this theorem uses the combinatorial approach to CSP.

Theorem 1.5.12 (Bang-Jensen,1990)**.** *If G is an oriented graph without sources and sinks then* $\mathrm{CSP}(G)$ *is in P iff G retracts to a disjoint union of directed cycles. Otherwise,* $\mathrm{CSP}(G)$ *is NP-complete.*

For proof, see [something is missing here]. This theorem was proved using algebraic methods.