TOPICS IN GRAPH BURNING AND DATALOG

by

Daniel Moghbel

Master of Science, University of West Florida, 2017

Honours Bachelor of Science, University of Toronto, 2013

A dissertation

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

in the program of

Mathematical Modelling and Methods

Toronto, Ontario, Canada, 2020

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A DISSERTATION

I hereby declare that I am the sole author of this dissertation. This is a true copy of the dissertation, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this dissertation to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this dissertation by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.

Daniel Moghbel

Topics in Graph Burning and Datalog

Doctor of Philosophy 2020

Mathematical Modelling and Methods

Ryerson University

Abstract

Graph burning studies how fast a contagion, modelled as a set of fires, spreads in a graph. The burning process takes place in discrete time-steps, or rounds. In each round, a fire breaks out at a given node (or vertex), thus burning it. Between rounds, fires from burning nodes spread to adjacent nodes. The *burning number* of a graph G, denoted $b(G)$, is the minimum number of rounds necessary for every node of G to burn. We consider $b(G_{m,n})$, where $G_{m,n}$ is the $m \times n$ Cartesian grid. For $m = \omega(\sqrt{n})$, the asymptotic value of $b(G_{m,n})$ was determined, but only the growth rate of $b(G_{m,n})$ was investigated in the case $m = O(\sqrt{n})$. Accordingly, we provide new explicit bounds on $b(G_{c\sqrt{n},n})$ for valid $c > 0$.

Graph burning is analogous to a *pebble game*, which typically involves the placement of pebbles on nodes of a graph. Burning of a node is comparable to a *pebbling step* (or *pebbling move*): the removal of two pebbles from a node, where one of the removed pebbles is placed on an adjacent node while the other is discarded. In a certain pebble game variant (discussed in Section 1.7), the existence of a winning strategy has an interesting characterization: expressibility of the relevant *constraint satisfaction problem (or CSP)* in the logic programming language *Datalog*.

If a structure with a non-empty domain is restricted to relation symbols only, then we call that structure a *template*. We show that the CSP for any finite template admitting terms of the weak Jónsson type has a property known as *bounded pathwidth duality*. This implies the expressibility of the complement CSP in linear Datalog, and places the CSP in **NL**.

Acknowledgements

First and foremost, I am eternally grateful to God, the master mathematician, for all He has done in my life. I would like to thank my mother, an educator herself, for her sacrifices, ongoing guidance, and continued academic support. I would also like to thank my other family members and friends. Finally, I am grateful for the guidance I have received from my supervisors and colleagues.

Special thanks to my thesis committee, for investing their time and taking an interest in my academic pursuits. I also gratefully acknowledge the funding sources for my doctoral work, which include a Ryerson Graduate Development Award, a Ryerson Graduate Fellowship, and a Mathematics Graduate Award.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Rumours, as the colloquial saying goes, can spread like wildfires. We can say the same for memes, viral videos, and other forms of social contagion, particularly on a social network, such as Facebook or Twitter. To model the spread of such contagion, Bonato et al. ([11, 12]) introduced the notion of graph burning, which we discuss in Section 1.6. This notion was inspired by graph searching games (or processes) such as Firefighting and graph cleaning ([11]). Identifying contagion with fire, starting at some node and spreading to all adjacent nodes, we can use graph burning to study how fast these fires spread in a given network ([13]). Formal explanations of graph-theoretic terminology used thus far can be found in Section 1.2. For further background in elementary graph theory, we refer the reader to [56].

Graph burning is similar to a pebble game played on a graph. Burning of a node is comparable to a pebbling move (see [19] and [51]). In a certain pebble game variant (described in Section 1.7), the existence of a winning strategy translates to expressibility of the relevant constraint satisfaction problem (or CSP) in Datalog ([43]). The CSP for a fixed input is the problem of assigning values to a given set of variables, subject to some constraints; we discuss this formally in Section 1.7. Many deep research problems in theoretical computer science stem from CSPs ([4]). Examples of CSPs arise not only in mathematics and computer science, but in other fields such as artificial intelligence ([4, 22, 24, 26, 42, 46]), computational linguistics ([29]), business process management ([33]), and musicology ([48]).

We begin this chapter with a review of basic mathematical notation, and an overview of graph theory. We then present some general definitions from logic and universal algebra, which are essential to our work. After providing some brief insight into the complexity classes we will encounter, we consider the process of graph burning. We then close the chapter with a detailed discussion of CSPs and Datalog.

1.1 Basic Notation

We use the notation $\mathbb{R}$, $\mathbb{N}_0$, $\mathbb{N}$, and $\mathbb{P}$ for the set of all real numbers, non-negative integers, natural numbers (or positive integers), and prime numbers, respectively. We also denote the Boolean domain $\{0,1\}$ by $\mathbb{B}$, and let $[n] = \{1, 2, \dots, n\}$ for any $n \in \mathbb{N}$.

Given two sets A and B, we denote their union by $A \cup B$, and their intersection by $A \cap B$. In case $A \cap B = \varnothing$ (that is, A and B are disjoint), we write $A \cup B$ as $A \sqcup B$. The *exponentiation* of B by A is B^A, the set of all functions $x \colon A \longrightarrow B$. If A is an index set, then $x \in B^A$ may be written $x = (x_a)_{a \in A}$, where $x_a = x(a)$. We call x a *sequence* if $A \subseteq \mathbb{N}_0$. The *complement* of B in A is $A \smallsetminus B = \{x \in A \mid x \notin B\}$. The *product* of A and B is $AB = \{ab \mid a \in A,\ b \in B\}$, which is a set of juxtaposed pairs or abstract products. The *Cartesian product* of A and B is $A \times B = \{(a,b) \mid a \in A,\ b \in B\}$. More generally, the Cartesian product of an indexed family of sets $\{A_i\}_{i \in I}$ is $\prod_{i \in I} A_i$, the set of all functions $x \colon I \longrightarrow \bigcup_{i \in I} A_i$ such that $x_i \in A_i$ for each $i \in I$. The n-fold Cartesian product of A with itself is denoted A^n.

The *power set* of a given set X is $\boldsymbol{\mathcal{P}}(X) = \{Y \mid Y \subseteq X\}$. In particular, for fixed $k \in \mathbb{N}_0$, we let $\boldsymbol{\mathcal{P}}_k(X) = \{Y \subseteq X \mid |Y| = k\}$. If α is an equivalence relation on X, then we denote the α-block (that is, α-equivalence class) of $x \in X$ by x/α. In this case, we let $X/\alpha = \{x/\alpha \mid x \in X\}$.

Finally, suppose $f(n)$ and $g(n)$ define sequences of non-negative real numbers. We write $f(n) = o(g(n))$ if f is of order less than g; that is,

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0\,.$$

The notation $f(n) = (1 + o(1))g(n)$ thus signifies that

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = 1\,.$$

We also write $f(n) = \omega(g(n))$ if $g(n) = o(f(n))$. We write $f(n) = O(g(n))$ if f is of order at most g; that is,

$$\limsup_{n\to\infty} \frac{f(n)}{g(n)} < \infty\,.$$

In case $f(n) = O(g(n))$ and $g(n) = O(f(n))$, we write $f(n) = \Theta(g(n))$.

1.2 Graph Theory

A *(simple) graph* G is a tuple (V, E) consisting of a set $V(G) = V$ of *vertices* (or *nodes*), and a set $E(G) = E$ of *edges* (which are lines or curves), where $E \subseteq \{uv \in VV \mid u \neq v\}$. We call $|V|$ and $|E|$ the *order* and *size* of G, respectively. We say that G is *finite* if V and E are both finite. In this dissertation, we only consider finite graphs of positive order.

Given a graph $G = (V, E)$, we say that ϵ is an *element* of G and write $\epsilon \in G$ if $\epsilon \in V \sqcup E$. If $e = uv \in E$, then we say e is *incident* with u and v (its *endpoints*), and vice-versa. If every $e \in E$ is directed (that is, has a direction), then G is called a *directed graph* or, more simply, a *digraph.* In this case, e is incident from u and to v, or from v and to u, depending on the direction of e. Directed edges are sometimes called *arcs.* Note that G is undirected if and only if $uv = vu$ for all $u, v \in V$. Below is an illustration of an undirected graph and a digraph.

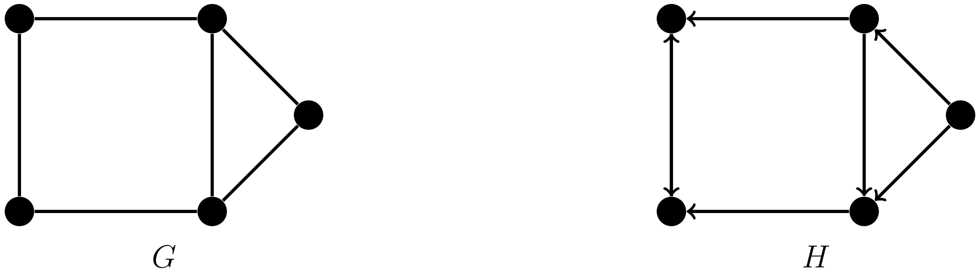


Figure 1.1: An undirected graph G and a digraph H. The double-arrow in H indicates two overlapping edges with opposing directions. Thus, although $|V(G)| = |V(H)|$, we have that $|E(H)| = |E(G)| + 1$.

Let $G = (V, E)$ be a graph with $u, v \in V$. Then u and v are *adjacent* if $uv \in E$ or $vu \in E$. This is written $u \leftrightarrow v$ in case $uv = vu$. We say that u *succeeds* v (written $u \leftarrow v$) or u *precedes* v (written $u \to v$), according as $vu \in E$ or $uv \in E$. Two edges are adjacent if they share an endpoint.

If $G = (V, E)$ is a graph, then its *complement graph* is

$$\overline{G} = \big(V, \{uv \in VV \mid u \neq v\} \smallsetminus E\big).$$

The *total graph* of G is the graph $T(G)$ with vertex set $V \sqcup E$ and an edge for each corresponding pair of adjacent or incident elements in G. A *(vertex-)colouring* of G is an assignment of colours (alternatively, symbols) to its vertices; an *edge-colouring* is similar. A colouring of G is *proper* if no two adjacent vertices have the same colour. We say G is k*-colourable* if G can be properly coloured with at most k colours. In this case, G is k*-chromatic* if G is not $(k-1)$-colourable. A *total colouring* of G is a proper colouring of $T(G)$. These concepts are illustrated in the figure below.

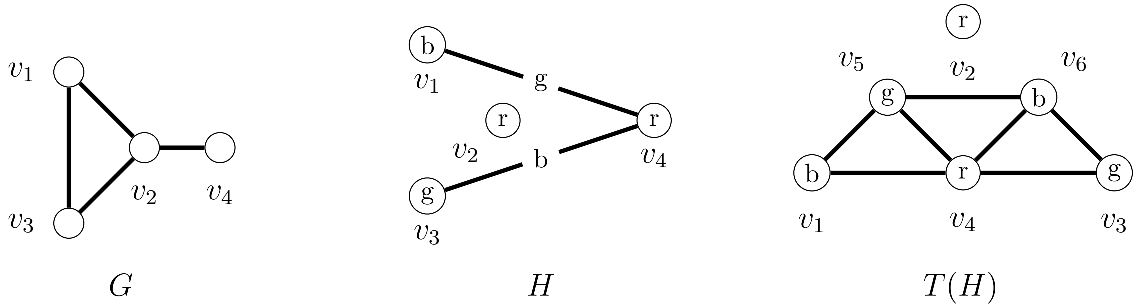


Figure 1.2: A *paw* graph G, a graph H obtained from $\overline{G}$, and $T(H)$. The elements of $\overline{G}$ are coloured by (or assigned) symbols r, g, and b so as to obtain H, which is totally coloured as seen in the proper colouring of $T(H)$. Note that $\overline{G}$ is 2-chromatic.

Let G be a graph with $v \in V(G)$. The *degree* of v in G, denoted $\deg_G v$, is the number of edges incident with v. If G is undirected, then we define the *minimum* and *maximum degree* of G to be $\delta(G) = \min\{\deg_G v \mid v \in V(G)\}$ and $\varDelta(G) = \max\{\deg_G v \mid v \in V(G)\}$, respectively. We also define in this case the *(open) neighbourhood* of v to be

$$N_G(v) = \big\{w \in V(G) \bigm| vw \in E(G)\big\}$$

(the set of all *neighbours* of v), and the *closed neighbourhood* of v to be

$$N_G[v] = N_G(v) \sqcup \{v\}\,.$$

Notice we have $\deg_G v = |N_G(v)|$ in this case. We say that v is *isolated* if $\deg_G v = 0$, and that v is a *pendant* if $\deg_G v = 1$. For example, in Figure 1.2, v_2 is isolated in H and $T(H)$, while v_4 is a pendant in G.

Let G be a graph. We call $D \subseteq V(G)$ a *dominating set* for G if for every $v \in V(G) \smallsetminus D$, there exists $w \in D$ such that $w \leftrightarrow v$. The *domination number* of G is the minimum size of a dominating set for G. The figure below gives an illustration.

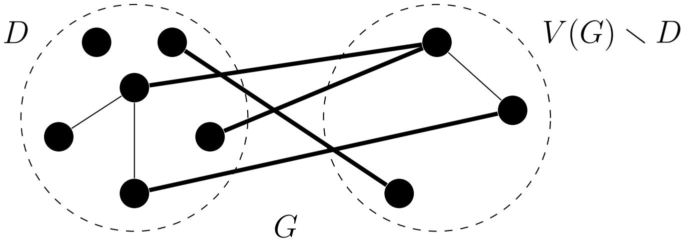


Figure 1.3: A graph G with a dominating set D. The members of D are said to *dominate* those of $V(G) \smallsetminus D$.

The following result is often called the *First Theorem of Graph Theory*:

Theorem 1.2.1 ([56])**.** If $G = (V, E)$ is an undirected graph, then

$$\sum_{v \in V} \deg_G v = 2|E| \,.$$

Proof. Each edge is counted twice upon summing the degrees. □

As an immediate corollary, we have the so-called *Handshaking Lemma*:

Corollary 1.2.1 ([56])**.** Every graph has an even number of vertices with odd degree.

Many other immediate corollaries of Theorem 1.2.1 exist (see [56]).

In any graph G, a *clique* is a set of pairwise adjacent vertices, while an *independent set* is a set of pairwise nonadjacent vertices. We say G is *complete* if $V(G)$ is a clique in G. We also say G is *k-partite* if $V(G)$ can be partitioned into k independent sets (referred to as the *partite sets* of G). The prefixes bi- (for $k = 2$), tri- (for $k = 3$), tetra- (for $k = 4$), and multi- (for unspecified k) are also used in place of prefix k-.

Two graphs G and H are *vertex-disjoint* if $V(G) \cap V(H) = \varnothing$ and *edge-disjoint* if $E(G) \cap E(H) = \varnothing$. They are simply *disjoint* if $V(G) \cap V(H) = \varnothing = E(G) \cap E(H)$. We say that H is a *subgraph* of G and write $H \subseteq G$ if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. If $H \subseteq G$, then we can also say that G is a *supergraph* of H and write $G \supseteq H$. In this case, H is a *spanning subgraph* of G if $V(H) = V(G)$, and H is the subgraph of G *induced by* $V(H)$ (written $H = G[V(H)]$) if $V(H) \neq \varnothing$ and $E(H) = \{uv \in E(G) \mid u, v \in V(H)\}$. These definitions, along with the definition of a graph, suggest that while a graph is not a set per se, on a primitive level, it actually is.

Note that in any graph $G = (V, E)$, deletion of a vertex automatically implies deletion of any edges incident with it. The subgraph of G obtained by removing a subset S from V or E is denoted $G - S$. The supergraph of G obtained by uniting a vertex set S with V or an edge set S with E is denoted $G + S$. When $S = \{s\}$, the notations can be simplified to $G - s$ and $G + s$ respectively. If G is undirected, then the graph H obtained by *duplicating* v is $G \circ v$; duplication is achieved by adding to G a new vertex v' (called a *clone* of v) such that $N_H(v') = N_G(v)$. This is illustrated in the figure below.

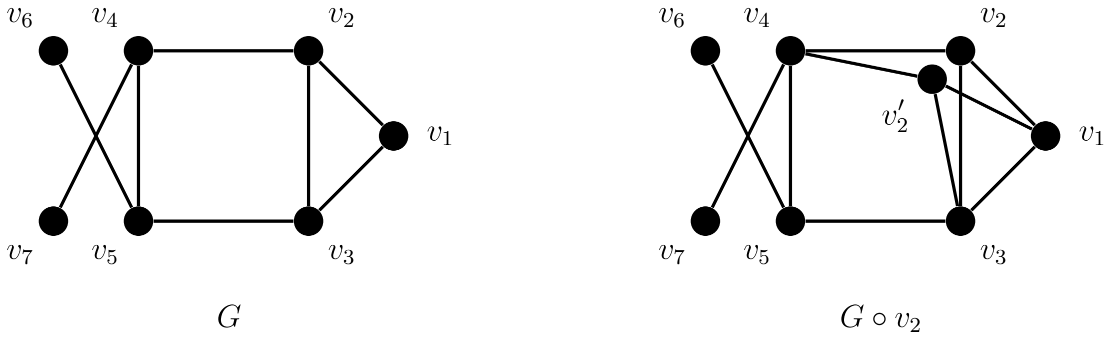
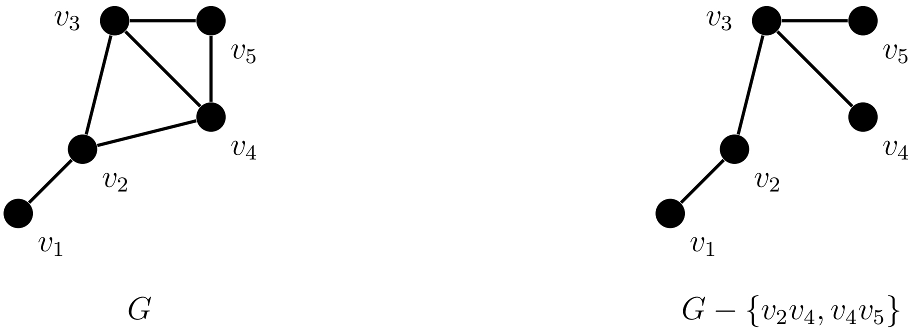


Figure 1.4: An example of vertex duplication in a graph G.

A *trail* of length k in a graph is a vertex-edge sequence $(v_0, e_1, v_1, \ldots, e_k, v_k)$, such that $e_i = v_{i-1}v_i$ for all $i \in [k]$. Such trail is *closed* (and called a *circuit*) if $v_0 = v_k$ and *open* if $v_0 \neq v_k$. It is a *path* (on $k+1$ vertices) if $v_{i-1} \neq v_j$ for all $i \neq j+1$, and a *cycle* (on k vertices) if $v_0 = v_k$ while $v_i \neq v_j$ for all $i \neq j$. A directed path (respectively, cycle) whose edges all have the same direction is called a *dipath* (respectively, *dicycle*). Any trail of the form $(s, e, \ldots, t)$ is referred to as a trail from s to t or, more simply, an s,t-trail, and is said to *join* s and t. A trail is said to be *even* or *odd* according as its length is even or odd.

A graph is *cyclic* if it contains a cycle (as a subgraph), and *acyclic* otherwise. A (di)graph is *connected* if every pair of distinct vertices is joined by some (di)path. Otherwise, the graph is *disconnected.* The *components* of a graph are its maximally connected subgraphs. An undirected, connected acyclic graph is called a *tree.* Pendants of a tree are called *leaves.* A tree with exactly one vertex of degree at least 3 is called a *spider.* A spanning subgraph that happens to be a tree is called a *spanning tree.* The figure below depicts a graph and one of its spanning trees.



G $\qquad$ $G - \{v_2v_4, v_4v_5\}$

Figure 1.5: A *kite* graph G and its spanning tree $G - \{v_2v_4, v_4v_5\}$.

A path or circuit H in a graph G is *Hamiltonian* if $V(H) = V(G)$ and *Eulerian* if $E(H) = E(G)$. For example, the paw graph G in Figure 1.2 has two Hamiltonian paths: $G - v_1v_2$ and $G - v_2v_3$.

If a graph G has a specially designated vertex r, then G is called a *rooted graph* and r is called a *root*. A *rooted tree partition* of G is a collection of rooted tree subgraphs whose vertex sets partition $V(G)$. An example of this can be seen in the figure below.

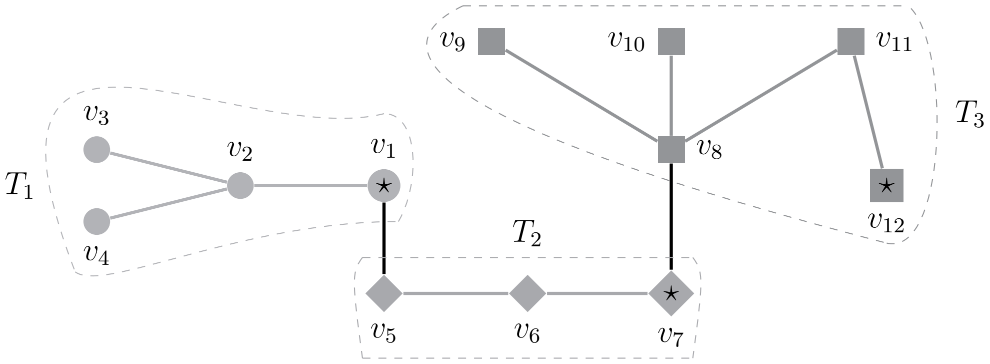


Figure 1.6: A graph with rooted tree partition $\{T_1, T_2, T_3\}$. Root nodes are labelled $\star$.

A *tree composition* of vertex-disjoint trees $T_1, \ldots, T_k$ is any tree T that can be formed from

$$T_1 \cup \cdots \cup T_k$$

by identifying some vertices among the leaves of $T_1, \ldots, T_k$. In this case, the vertices of T that arise through identification of leaves are called *composition vertices*. The *leaf components* of T are the components of T that had at most one leaf identified in the construction of T. Figure 1.7 depicts a tree composition of T_1, T_2, and T_3 from the rooted tree partition in Figure 1.6.

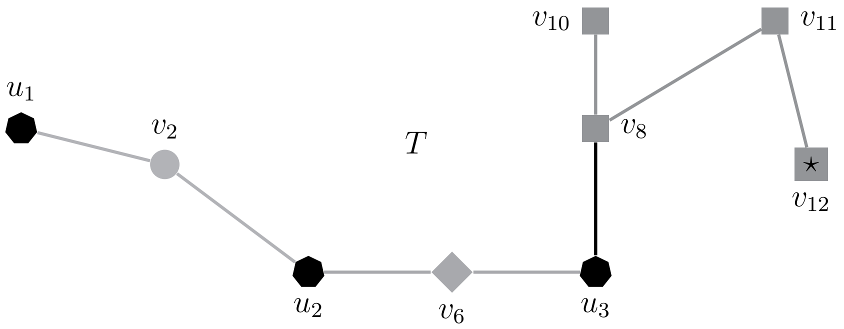


Figure 1.7: A rooted tree composition T of T_1, T_2, and T_3 from Figure 1.6. Nodes v_3 and v_4 were identified, giving u_1. Nodes v_1 and v_5 were also identified, giving u_2. Finally, nodes v_7 and v_9 were identified, giving u_3. Thus, the composition vertices of T are u_1, u_2, and u_3. The leaf components of T are $T[\{v_2\}]$, $T[\{v_6\}]$, $T[\{v_8, v_{10}, v_{11}, v_{12}\}]$, $T[\{v_2, u_1\}]$, $T[\{v_2, u_2, v_6\}]$, and $T[\{v_6, u_3, v_8, v_{10}, v_{11}, v_{12}\}]$.

The next theorem is a well-known characterization of bipartite graphs.

Theorem 1.2.2. For any graph G, the following are equivalent:

(i) G is 2-colourable;

(ii) G is bipartite;

(iii) G has no odd cycles.

Proof. See, for example, [56]. □

The odd cycle characterization of bipartite graphs is due to Dénes König ([56]), who wrote the first book on graph theory ([9]).

For any path with endpoints u and v, we say u and v are *(path-)connected* and write $u \leftrightsquigarrow v$. Given a u,v-dipath, we say that u *reaches* v and write $u \rightsquigarrow v$. The notation $u \leftsquigarrow v$ is defined analogously.

Let G be a graph. The *(graph) distance* between vertices u and v of G, denoted $\operatorname{dist}_G(u, v)$, is the length of a shortest u,v-path in G. By convention, $\operatorname{dist}_G(u, v) = \infty$ if no shortest u,v-path in G exists. Furthermore, if $X, Y \subseteq V(G)$ and $v \in V(G)$, then we set $\operatorname{dist}_G(v, Y) = \min\{\operatorname{dist}_G(v, w) \mid w \in Y\}$ and $\operatorname{dist}_G(X, Y) = \max\{\operatorname{dist}_G(u, Y) \mid u \in X\}$. We further define the *eccentricity* of v and of Y to be

$$\operatorname{ecc}_G v = \max_{u \in V(G)} \operatorname{dist}_G(u, v) \quad \text{and} \quad \operatorname{ecc}_G Y = \max_{u \in V(G)} \operatorname{dist}_G(u, Y)\,,$$

respectively. The *radius* and *diameter* of G are defined as

$$\operatorname{rad} G = \min_{v \in V(G)} \operatorname{ecc}_G v \quad \text{and} \quad \operatorname{diam} G = \max_{v \in V(G)} \operatorname{ecc}_G v\,,$$

respectively. Finally, the *centre* of G is defined to be

$$\mathrm{Z}(G) = \left\{ v \in V(G) \;\middle|\; \operatorname{ecc}_G v = \operatorname{rad} G \right\},$$

whose elements are referred to as *central* vertices. We remark that a subgraph H of G is *isometric* if

$$\operatorname{dist}_H(u, v) = \operatorname{dist}_G(u, v)$$

for all $u, v \in V(H)$. For example, a subtree of a tree is isometric ([11, 12]).

A *homomorphism* from a graph G to a graph H is a map $\phi\colon V(G) \longrightarrow V(H)$ with $E(G)$ and $E(H)$ specified; more concisely, $\phi\colon G \longrightarrow H$, having the following property: if $uv \in E(G)$, then $\phi(u)\phi(v) \in E(H)$. A bijective homomorphism $\psi\colon G \longrightarrow H$ is called an *isomorphism*, which has the property that $uv \in E(G)$ if and only if $\psi(u)\psi(v) \in E(H)$. We write $G \cong H$ if and only if such an isomorphism ψ exists. The reader can check that $\cong$ is an equivalence relation.

Up to isomorphism, the following graphs on n vertices are unique: complete graphs, paths, and cycles. Consequently, isomorphism class (or $\cong$-block) representatives for these graphs have customary notations: K_n, P_n, and C_n, respectively. In particular, a complete bipartite graph with partite sets of sizes m and n is unique in this sense; $K_{m,n}$ denotes its isomorphism class representative. Two graphs in $K_{3,3}/\cong$ are depicted in the figure below.

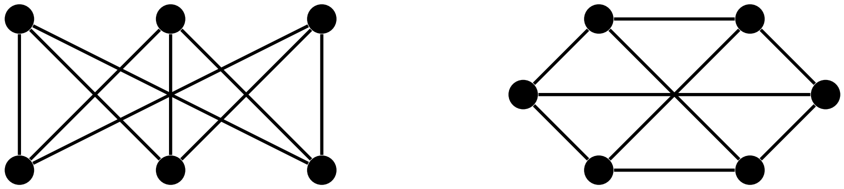


Figure 1.8: Two drawings of the *utility graph*, $K_{3,3}$.

We remark that a graph G is *chordal* if any cycle $C \subseteq G$ on 4 or more vertices has a *chord*; that is, a non-cycle edge joining two vertices of C.

We can define several graph operations for any m graphs $G_1, \ldots, G_m$, where $m \geq 2$. For example, the *union* of $G_1, \ldots, G_m$ is the graph

$$G_1 \cup \cdots \cup G_m = \big(V(G_1) \cup \cdots \cup V(G_m), E(G_1) \cup \cdots \cup E(G_m)\big)\,;$$

the *intersection* of $G_1, \ldots, G_m$ is defined similarly. If $G_1, \ldots, G_m$ are mutually disjoint, then we can define the *sum* of these graphs as their disjoint union,

$$G_1 + \cdots + G_m = \big(V(G_1) \sqcup \cdots \sqcup V(G_m), E(G_1) \sqcup \cdots \sqcup E(G_m)\big)\,.$$

When $G_1, \ldots, G_m \cong G$, we let $mG = G_1 + \cdots + G_m$. In particular, a sum of (disjoint) trees is called a *forest.* A *product* of $G_1, \ldots, G_m$ is a graph $(V(G_1) \times \cdots \times V(G_m), E)$, with E defined according to the type of product:

(i.) *Cartesian*, $G_1 \,\square \cdots \square\, G_m$, where $(u_1, \dots, u_j, \dots, u_m)(v_1, \dots, v_j, \dots, v_m) \in E$ if and only if $u_i = v_i$ for all $i \in [m] \smallsetminus \{j\}$ and $u_j v_j \in E(G_j)$;

(ii.) *weak*, $G_1 \times \cdots \times G_m$, where $(u_1, \dots, u_m)(v_1, \dots, v_m) \in E$ if and only if $u_i v_i \in E(G_i)$ for all $i \in [m]$;

(iii.) *strong*, $G_1 \boxtimes \cdots \boxtimes G_m$, where $E = E(G_1 \,\square \cdots \square\, G_m) \sqcup E(G_1 \times \cdots \times G_m)$.

Algebraic combinations of unique-up-to-isomorphism graphs yield new graphs, and new isomorphism classes. For example, the $m \times n$ *Cartesian grid* is $G_{m,n} \cong P_m \,\square\, P_n$; similarly, the $m \times n$ *strong grid* is $G^{\times}_{m,n} \cong P_m \boxtimes P_n$. These graphs are illustrated in the figure below.

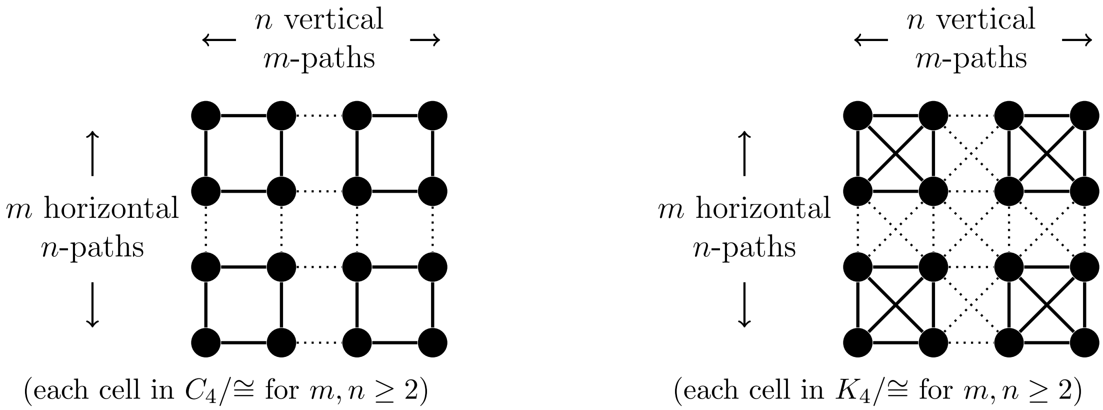

$\longleftarrow$ n vertical m-paths $\longrightarrow$
$\uparrow$ m horizontal n-paths $\downarrow$
(each cell in $C_4/\cong$ for $m, n \geq 2$)
$\longleftarrow$ n vertical m-paths $\longrightarrow$
$\uparrow$ m horizontal n-paths $\downarrow$
(each cell in $K_4/\cong$ for $m, n \geq 2$)

Figure 1.9: The graphs $G_{m,n}$ (left) and $G^{\times}_{m,n}$ (right).

Fixing $m = 2$ for $G_{m,n}$ gives the *ladder* on n vertices, denoted L_n.

More generally, the n-dimensional *Cartesian hypergrid* is $G_{m_1,\dots,m_n} \cong P_{m_1} \square \cdots \square P_{m_n}$; we can define a *strong hypergrid* analogously. The n-dimensional *hypercube* graph is $Q_n = G_{2,\dots,2}$ (where 2 appears n times); note that $|V(Q_n)| = |(V(P_2))^n| = 2^n$. The *ladder-rung* graph on $2n$ vertices is nP_2. The *prism* (or *circle-ladder*) on n vertices is

$Y_n \cong K_2 \,\Box\, C_n$. The *wheel* on n vertices, denoted by W_n, consists of one central vertex adjacent to the vertices of C_{n-1}.

Finally, an undirected graph G is said to be *planar* if it can be embedded in the plane. Intuitively, this means that G can be drawn with no edges crossing. In this case, the resulting graph divides the set of points of the plane not lying on G into regions called *faces*, one of which is unbounded. A planar graph is *outerplanar* if it can be embedded with all vertices on the outer (or unbounded) face.

1.3 Logic and Universal Algebra

We now present the main definitions we will need from logic and universal algebra. For further background on these branches of mathematics, we refer the reader to [18] and [31].

A *structure* with universe (or domain) $A \neq \varnothing$ restricted to function and relation symbols has the form

$$\mathfrak{A} = (A; \mathcal{F} \sqcup \mathcal{R}; \mathrm{ar}_{\mathcal{F}\sqcup\mathcal{R}}; \iota)\,,$$

where $\mathcal{F}$ is the set of function symbols, $\mathcal{R}$ is the set of relation symbols, $\mathrm{ar}_{\mathcal{F}\sqcup\mathcal{R}}$ is the arity function on $\mathcal{F} \sqcup \mathcal{R}$ (with omissible subscript), and

$$\iota \colon f \longmapsto f^{\mathfrak{A}} \in A^{A^{\mathrm{ar}\, f}},\ R \longmapsto R^{\mathfrak{A}} \subseteq A^{\mathrm{ar}\, R} \ \text{ for all } (f, R) \in \mathcal{F} \times \mathcal{R}$$

is the interpretation function. The indexed elements $f^{\mathfrak{A}}$ and $R^{\mathfrak{A}}$ are referred to as the *basic operations* and *basic relations* of $\mathfrak{A}$, respectively.

We call a structure $\mathfrak{A} = (A; \mathcal{F} \sqcup \mathcal{R}; \mathrm{ar}_{\mathcal{F}\sqcup\mathcal{R}})$ a *template* (or *relational structure*) if $\mathcal{F} = \varnothing$ and a *(universal) algebra* if $\mathcal{R} = \varnothing$. The tuple $(\mathcal{F} \sqcup \mathcal{R}, \mathrm{ar}_{\mathcal{F}\sqcup\mathcal{R}})$ is often called the *signature* of $\mathfrak{A}$. A graph equipped with binary relations on its vertex set is a prime example of a template. Group-like structures are among the best known examples of universal algebras.

A *homomorphism* $f\colon \mathfrak{A} \longrightarrow \mathfrak{B}$ from a structure $\mathfrak{A} = (A; \mathcal{R})$ to a structure $\mathfrak{B} = (B; \mathcal{R})$ is a map $f\colon A \longrightarrow B$ that preserves all of the basic operations and basic relations (or just the latter in the case of templates, and just the former in the case of algebras). If such a homomorphism f exists, then we say $\mathfrak{A}$ is *homomorphic* to $\mathfrak{B}$ and write $\mathfrak{A} \to \mathfrak{B}$. We let

$$\mathrm{Hom}(\mathfrak{A}, \mathfrak{B}) = \left\{ f\colon \mathfrak{A} \longrightarrow \mathfrak{B} \;\middle|\; \mathfrak{A} \to \mathfrak{B} \right\}.$$

If $g \in \mathrm{Hom}(\mathfrak{A}, \mathfrak{B})$ is bijective, then g is called an *isomorphism* and we say that $\mathfrak{A}$ is *isomorphic* to $\mathfrak{B}$; written $\mathfrak{A} \cong \mathfrak{B}$. An *endomorphism* of $\mathfrak{A}$ is any $f \in \mathrm{Hom}(\mathfrak{A}, \mathfrak{A})$. A bijective endomorphism is called an *automorphism*. A structure is a *core* if all of its endomorphisms are automorphisms. Homomorphisms between graphs and groups are typical examples of the types of maps mentioned here.

Two structures $\mathfrak{A} = (A; \mathcal{S})$ and $\mathfrak{B} = (B; \mathcal{S})$ are *elementarily equivalent* (written $\mathfrak{A} \equiv \mathfrak{B}$) if they satisfy the same first-order sentences (which are formulas with no free variables). In the case that $\mathfrak{A}$ and $\mathfrak{B}$ are templates, an *Ehrenfeucht-Fraïssé game* can determine whether $\mathfrak{A} \equiv \mathfrak{B}$. The game is played between two players, called the *Spoiler* and the *Duplicator*, and is mainly used to prove results concerning inexpressibility in first-order logic. The interested reader can refer to [35] for details. In Section 1.7, we describe a pebble game variant of this game, the same variant we alluded to earlier.

A k-ary relation R and an m-ary operation f on a given set A are *compatible* if

$$(a_{1,1}, \ldots, a_{1,k}), \ldots, (a_{m,1}, \ldots, a_{m,k}) \in R$$

implies

$$\big(f(a_{1,1},\ldots,a_{m,1}),\ldots,f(a_{1,k},\ldots,a_{m,k})\big) \in R\,.$$

In this case, we say that f is a *polymorphism* of R, and denote this fact by $f \triangleright R$ (read "*f preserves R*"). Alternatively, we say that R is *invariant* under f, and denote this fact by $R \triangleleft f$. The set of all polymorphisms for a template $\mathfrak{A} = (A;\mathcal{R})$, denoted by $\operatorname{Pol}\mathfrak{A}$, consists of all finitary operations on A that preserve (in the above sense) the basic relations of $\mathfrak{A}$. We let $\operatorname{Pol}_n\mathfrak{A} = \{f \in \operatorname{Pol}\mathfrak{A} \mid \operatorname{ar} f = n\}$. The *polymorphism algebra* of $\mathfrak{A}$ is $(A;\operatorname{Pol}\mathfrak{A})$. Dually, the set of all invariants for an algebra $\mathfrak{V} = (A;\mathcal{F})$ is denoted by $\operatorname{Inv}\mathfrak{V}$, and consists of all finitary relations on A that are invariant under the basic operations of $\mathfrak{V}$. We let $\operatorname{Inv}_n\mathfrak{V} = \{R \in \operatorname{Inv}\mathfrak{V} \mid \operatorname{ar} R = n\}$. The operators Pol and Inv form a *canonical Galois connection*; see [21] for details. We will say that $\mathfrak{V}$ is *finitely related* if there exists a finite set $\mathcal{R}$ of relation symbols such that $\mathcal{F} = \operatorname{Pol}(A;\mathcal{R})$. Polymorphisms satisfying certain algebraic identities have been used extensively in the study of CSPs. As an example, ternary polymorphisms $f_1,\ldots,f_n$ for a template $\mathfrak{A} = (A;\mathcal{R})$ form a *Hagemann-Mitschke sequence* (or, an *HM-sequence*) for $\mathfrak{A}$, provided that the following equations hold for all $x, y \in A$:

$$\begin{aligned} x &= f_1(x,y,y)\,;\\ f_i(x,x,y) &= f_{i+1}(x,y,y) \text{ for each } i \in [n-1]\,;\\ f_n(x,x,y) &= y\,. \end{aligned}$$

The relevance of HM-sequences in our work will become apparent in Chapters 4 and 5.

A structure $\mathfrak{B} = (B;\mathcal{F}\sqcup\mathcal{R})$ is a *substructure* of $\mathfrak{A} = (A;\mathcal{F}\sqcup\mathcal{R})$ if $B \subseteq A$ and the inclusion map $\psi\colon B \longrightarrow A$ induces a homomorphism from $\mathfrak{B}$ to $\mathfrak{A}$. In particular, $\mathfrak{B}$

is the *substructure of* $\mathfrak{A}$ *induced by* B, written $\mathfrak{B} = \mathfrak{A}[B]$, if the basic operations and basic relations of $\mathfrak{B}$ are those of $\mathfrak{A}$ restricted to B. A map $f\colon C \subseteq A \longrightarrow B$ is a *partial homomorphism* if f induces a homomorphism from $\mathfrak{A}[C]$ to $\mathfrak{B}$. Subgraphs and subgroups are classic examples of underlying substructures.

A template $\mathfrak{D}_1 = (D; \mathcal{R})$ *p.p.-defines* a template $\mathfrak{D}_2 = (D; \mathcal{S})$ if every $S \in \mathcal{S}$ can be defined by a *positive-primitive* (or *p.p.*) formula with respect to $\mathcal{R}$; that is, there exists a formula $\psi(x) \equiv \exists y.\phi(x, y)$ where $\phi \equiv R_1 \wedge \cdots \wedge R_{\operatorname{ar}\phi}$ and $R_1, \ldots, R_{\operatorname{ar}\phi} \in \mathcal{R} \cup \{=\}$. A useful characterization (see, for example, [5]) is that $\operatorname{Pol}\mathfrak{D}_1 \subseteq \operatorname{Pol}\mathfrak{D}_2$.

Let $\mathfrak{A} = (A; \mathcal{F})$ and $\mathfrak{B} = (B; \mathcal{F})$ be algebras, where $B \subseteq A$. If $\mathfrak{B} = \mathfrak{A}[B]$ (that is, the basic operations of $\mathfrak{B}$ are those of $\mathfrak{A}$ restricted to B), then $\mathfrak{B}$ is called a *subalgebra* of $\mathfrak{A}$; written $\mathfrak{B} \leq \mathfrak{A}$. We say B is a *subuniverse* of $\mathfrak{A}$ (written $B \leqslant \mathfrak{A}$) provided that B is closed under the basic operations of $\mathfrak{A}$. Consequently, if $\varnothing \neq B \leqslant \mathfrak{A}$, then B defines a subalgebra of $\mathfrak{A}$. The *subuniverse of* $\mathfrak{A}$ *generated by* $X \subseteq A$ is

$$\operatorname{Sg}_{\mathfrak{A}} X = \bigcap_{X \subseteq C \leqslant \mathfrak{A}} C\,.$$

An algebra $\mathfrak{A} = (A; \mathcal{F})$ is *conservative* if, for every k-ary operation $f^{\mathfrak{A}}$, and all $a_1, \ldots, a_k \in A$, we have that

$$f^{\mathfrak{A}}(a_1, \ldots, a_i, \ldots, a_k) = a_i \quad \text{for each} \quad i \in [k]\,.$$

That is, $f^{\mathfrak{A}}$ acts as an i^{th}-coordinate *projection* for each $i \in [k]$. Notice that $\mathfrak{A}$ is conservative if and only if $B \leqslant \mathfrak{A}$ for every $B \subseteq A$.

We now mention five types of operations that are relevant to the study of CSPs. Given a set A, an operation $f\colon A^n \longrightarrow A$ is:

(i.) *idempotent* if $f(a, \dots, a) = a$ for all $a \in A$;

(ii.) a *Maltsev* operation if $n = 3$ and

$$f(y, y, x) = f(x, y, y) = x \text{ for all } x, y \in A\,;$$

(iii.) a *near-unanimity (NU)* operation if $n \geq 3$ and

$$f(x, \dots, x, y) = f(x, \dots, x, y, x) = \cdots = f(y, x, \dots, x) = x \text{ for all } x, y \in A\,;$$

(iv.) a *weak near-unanimity (WNU)* operation if $n \geq 2$ and, for all $x, y \in A$,

$$f(x, \dots, x) = x \text{ and } f(x, \dots, x, y) = f(x, \dots, x, y, x) = \cdots = f(y, x, \dots, x)\,;$$

(v.) a *Taylor* operation if $n \geq 2$ and, for all $x, y \in A$,

$$f(z_1, \dots, z_{i-1}, x, z_{i+1}, \dots, z_n) = f(w_1, \dots, w_{i-1}, y, w_{i+1}, \dots, w_n)\,;\; z_j, w_j \in \{x, y\}$$

for all $i, j \in [n]$.

An algebra is *idempotent* if all of its basic operations are idempotent. Algebras with Taylor operations are called *Taylor algebras* (named after W.F. Taylor). It is useful to note (as done in [5]) that the defining property of a Taylor operation $f\colon A^n \longrightarrow A$ prevents f from being an i^{th}-coordinate projection for each $i \in [n]$, whenever $|A^n| > 1$. As indicated in [26], a finite idempotent algebra is a Taylor algebra if and only if it has an at-least ternary WNU operation.

A collection $\mathcal{O}$ of finitary operations on a set D is called a *functional clone* on D if

$\mathcal{O}$ contains all i^{th}-coordinate projections on D and is closed under generalized function composition. Similarly, a collection $\mathcal{Q}$ of finitary relations on a set D is called a *relational clone* on D if $\mathcal{Q}$ contains the equality relation $(=)$ and is closed under intersection, direct products, projections onto a subset I of coordinates (proj_I), and permutations of coordinates. Equivalently, $\mathcal{Q}$ is a relational clone if $\mathcal{Q}$ is closed under *p.p. definitions*; that is, whenever $\mathcal{R} \subseteq \mathcal{Q}$, it follows that $\mathcal{Q}$ contains all relations definable by p.p. formulas with respect to $\mathcal{R}$. Sets of polymorphisms and invariants form functional and relational clones, respectively.

If $\{\mathfrak{A}_i = (A_i; \mathcal{S}) \mid i \in I\}$ is an indexed family of structures (namely templates or algebras), then

$$\prod_{i\in I} \mathfrak{A}_i = \left(\prod_{i \in I} A_i \, ; \mathcal{S}^{|I|} \right)$$

is a structure of same kind, and is denoted by $\mathfrak{B}^{|I|}$ if $\mathfrak{A}_i = \mathfrak{B}$ for all $i \in I$. It is well-known (see, for example, [4]) that if R is a non-empty n-ary relation on the universe D of some template $\mathfrak{D}$, then R is p.p.-definable over $\mathfrak{D}$ if and only if $R \leqslant (D; \mathrm{Pol}\,\mathfrak{D})^n$.

An n-ary relation S on a given set D is *diagonal* if

$$S \supseteq \mathit{\Delta}_n(D) = \big\{(d, \ldots, d) \in D^n \;\big|\; d \in D\big\}\,;$$

that is, S contains the n-diagonal of D. If $\mathfrak{A} = (A; \mathcal{P}_A)$ is an algebra such that $\mathcal{P}_A$ is a functional clone on A containing all of the constant operations on A, then the invariants for $\mathfrak{A}$ will all be diagonal.

If $\mathfrak{A} = (A; \mathcal{F}_A)$ is an algebra where $\mathcal{F}_A$ is a functional clone on A, then its associated *template of invariants* is the dual structure

$$\mathfrak{A}^{\perp} = (A; \operatorname{Inv} \mathfrak{A})\,,$$

where $\mathfrak{A}^{\perp\perp} = \mathfrak{A}$. Thus, the restriction of $\mathfrak{A}^{\perp}$ to $U \subseteq A$ is $\mathfrak{A}^{\perp}|_U = (U; \operatorname{Inv}(U; \mathcal{F}_A))$.

A *term* of an algebra $\mathfrak{A} = (A; \mathcal{F})$ is the image of a composition of basic operations of $\mathfrak{A}$. A *term operation* $t^{\mathfrak{A}} \colon A^n \longrightarrow A$ is the interpretation of a function t defined by term $t(x_1, \ldots, x_n)$ as an n-ary operation on A. When all mappings $h_a \colon A \longrightarrow A$ defined by $h_a(x) = a$ are among the basic operations of $\mathfrak{A}$, we refer to the terms of $\mathfrak{A}$ as *polynomials.* The set of all term operations of $\mathfrak{A}$ is called the *(term) clone* of $\mathfrak{A}$, denoted by $\operatorname{Clo} \mathfrak{A}$. The set of all polynomial operations of $\mathfrak{A}$ is called the *polynomial clone* of $\mathfrak{A}$, denoted by $\operatorname{Pol} \mathfrak{A}$. (This is how Pol is defined when the input is an algebra.) We let $\operatorname{Clo}_n \mathfrak{A} = \{\phi \in \operatorname{Clo} \mathfrak{A} \mid \operatorname{ar} \phi = n\}$ and $\operatorname{Pol}_n \mathfrak{A} = \{\phi \in \operatorname{Pol} \mathfrak{A} \mid \operatorname{ar} \phi = n\}$. If $\mathfrak{B} = (B; \mathcal{G})$ is another algebra, then $\mathfrak{A}$ and $\mathfrak{B}$ are *term equivalent* if $A = B$ and $\operatorname{Clo} \mathfrak{A} = \operatorname{Clo} \mathfrak{B}$; similarly, *polynomially equivalent* if $A = B$ and $\operatorname{Pol} \mathfrak{A} = \operatorname{Pol} \mathfrak{B}$. We say that $\mathfrak{A}$ is *minimal* if $2 \leq |A| < \infty$ and every $f \in \operatorname{Pol}_1 \mathfrak{A}$ is a constant operation or a permutation on A.

The *star composition* of an m-ary term operation f and an n-ary term operation g is the mn-ary term operation $f \star g$ defined by

$$(f \star g)\big(x_1, \ldots, x_{mn}\big) = f\big(g(x_1, \ldots, x_n), g(x_{n+1}, \ldots, x_{2n}), \ldots, g(x_{mn-n+1}, \ldots, x_{mn})\big)\,.$$

Notice that $f \star g$ resembles the composition of f with the m-tuple $(g, \ldots, g)$.

Let $\mathfrak{A} = (A; \mathcal{F})$ be an algebra. Suppose there exist $d_0, d_1, \ldots, d_n \in \operatorname{Clo}_3 \mathfrak{A}$; $n \geq 2$, such that the following equations hold for all $x, y, z \in A$:

$$\begin{aligned} x &= d_0(x, y, z)\,; \\ d_i(x, y, y) &= d_{i+1}(x, y, y) \text{ for even } i < n\,; \\ d_i(x, y, x) &= d_{i+1}(x, y, x) \text{ for even } i < n\,; \\ d_i(x, x, y) &= d_{i+1}(x, x, y) \text{ for odd } i < n\,; \\ d_n(x, y, z) &= z\,. \end{aligned}$$

The terms defining $d_0, d_1, \dots, d_n$ are called *weak Jónsson terms.* These are simply called *Jónsson terms* if we also have that $d_i(x, y, x) = x$ for $0 < i < n$. If $\mathfrak{A}$ has weak Jónsson terms defining $d_0, d_1, \dots, d_n$, then $B \leqslant \mathfrak{A}$ is a *Jónsson ideal* of $\mathfrak{A}$ provided that $d_i(b, a, b') \in B$ for all $(b, a, b') \in B \times A \times B$ and $0 \leq i \leq n$. Jónsson ideals are prominent in the proofs of the main results of [1] and [3].

Given sets $C_1, \dots, C_n$ and $X \subseteq C_1 \times \cdots \times C_n$, we say that X is *subdirect* and write $X \subseteq_{\text{sd}} C_1 \times \cdots \times C_n$ if $\text{proj}_i[X] = C_i$ for each $i \in [n]$. Note that proj_i is the same as $\text{proj}_{\{i\}}$, and its domain here is $C_1 \times \cdots \times C_n$ while its range is C_i.

Let $\mathfrak{A} = (A; \mathcal{F})$ and $\mathfrak{B} = (B; \mathcal{F})$ be algebras, where $B \subseteq A$. We write $\mathfrak{B} \leq_{\text{sd}} \mathfrak{A}$ if $\mathfrak{B} \leq \mathfrak{A}$ and $B \subseteq_{\text{sd}} A$. Similarly, we write $B \leqslant_{\text{sd}} \mathfrak{A}$ if $B \leqslant \mathfrak{A}$ and $B \subseteq_{\text{sd}} A$. If $\mathfrak{B} \leq \mathfrak{A}$, then $\mathfrak{B}$ is said to be *absorbing* for $\mathfrak{A}$, if there exists $\phi \in \text{Clo}_k \mathfrak{A}$ such that $\phi(a_1, \dots, a_k) \in B$ whenever $|\{i \in [k] \mid a_i \notin B\}| \leq 1$. In this case, we say $\mathfrak{B}$ *absorbs* $\mathfrak{A}$ with respect to ϕ; written $\mathfrak{B} \trianglelefteq_\phi \mathfrak{A}$.

Let $\mathfrak{A}$ be an algebra with an n-ary operation ϕ, such that $B, C \leqslant \mathfrak{A}$. Then B *absorbs* C with respect to ϕ if, for all $i \in [n]$, all $b_1, \dots, b_{i-1}, b_{i+1}, \dots, b_n \in B$, and all $c \in C$, we have that $\phi(b_1, \dots, b_{i-1}, c, b_{i+1}, \dots, b_n) \in B$. We write $B \trianglelefteq_\phi C$ in this case, provided that $B \subseteq C$.

Suppose $\mathfrak{C} = (C; \mathcal{F})$ is an algebra with an operation ϕ. Then $c \in C$ is an *absorption constant for* $\mathfrak{C}$ *with respect to* ϕ if, for all $R \leqslant_{\mathrm{sd}} \mathfrak{C}^n$ where $n \geq 2$, the n-tuple $(c, \ldots, c)$ lies in R whenever $R \trianglelefteq_\phi \mathit{\Delta}_n(C)$. Below are two well-known facts about absorption, whose proofs are straightforward.

Lemma 1.3.1 ([4])**.** Let $\mathfrak{A} = (A; \mathcal{F})$ be an algebra with basic operation ϕ. If

$$B \trianglelefteq_\phi C \leqslant \mathfrak{A}\,,$$

then $B \cap D \trianglelefteq_\phi C \cap D$ for any $D \leqslant \mathfrak{A}$. Also, if $\mathfrak{A}$ is idempotent, then ϕ is an NU operation if and only if $\{a\} \trianglelefteq_\phi A$ for each $a \in A$.

Lemma 1.3.2 ([4])**.** Let $\mathfrak{B}_1 = (B_1; \mathcal{F})$ and $\mathfrak{C}_1 = (C_1; \mathcal{F})$ be algebras such that $\phi \in \mathcal{F}$ and $S \leqslant \mathfrak{B}_1 \times \mathfrak{C}_1$. If $\mathrm{proj}_1[S] = B_1$ and $C_0 \trianglelefteq_\phi C_1$, then $\{b \in B_1 \mid \exists c \in C_0 \ni (b, c) \in S\} \trianglelefteq_\phi B_1$.

The following theorem was discovered only in the last decade:

Theorem 1.3.1 ([4])**.** If $\mathfrak{C}$ is a finite algebra with an idempotent operation ϕ, then there exists an absorption constant for $\mathfrak{C}$ with respect to ϕ.

Note that the operation ϕ in the context of absorption as seen above need not always be specified.

An algebra $(L; f)$ where $\operatorname{ar} f = 2$ is called a *meet-* or *join-semilattice* according as f is $\wedge$ ("meet") or $\vee$ ("join"), and if f is commutative, associative, absorptive, and idempotent with respect to L. If $(L; \wedge)$ is a meet-semilattice and $(L; \vee)$ is a join-semilattice, then the algebra $(L; \wedge, \vee)$ is called a *lattice*. Such a lattice is *distributive* if $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$ and $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$ for all $x, y, z \in L$.

Given an algebra $\mathfrak{A} = (A; \mathcal{F})$, an equivalence relation α on A is called a *congruence* on $\mathfrak{A}$ if α is preserved by all basic operations of $\mathfrak{A}$. In this case, one can define the *quotient algebra* $\mathfrak{A}/\alpha = (A/\alpha; \mathcal{F})$, with basic operations induced by those of $\mathfrak{A}$. The set of all congruences on $\mathfrak{A}$ is denoted $\operatorname{Con}\mathfrak{A}$. We say that $\mathfrak{A}$ is *simple* if $\operatorname{Con}\mathfrak{A} = \{=_A, A^2\}$ where $=_A = \mathit{\Delta}_2(A)$. It is a well-known fact (see, for example, [18]) that $(\operatorname{Con}\mathfrak{A}; \wedge, \vee)$ is a lattice where, for any $\alpha, \beta \in \operatorname{Con}\mathfrak{A}$,

$$\alpha \wedge \beta = \alpha \cap \beta \quad \text{and} \quad \alpha \vee \beta = \bigcap_{\alpha, \beta \subseteq \gamma \in \operatorname{Con}\mathfrak{A}} \gamma\,.$$

If $[\alpha, A^2] = \{X \mid \alpha \subseteq X \subseteq A^2\}$ (the interval from α to A^2), then we have that $(\operatorname{Con}(\mathfrak{A}/\alpha); \wedge, \vee) \cong ([\alpha, A^2]; \wedge, \vee)$ for any $\alpha \in \operatorname{Con}\mathfrak{A}$ (see, for example, [18]). It then follows that $\mathfrak{A}/\alpha$ is simple if and only if α is maximal (that is, $|[\alpha, A^2]| = 2$) or $\alpha = A^2$.

A *variety* is a class of algebras of the same signature, closed under the formation of direct products, subalgebras, and homomorphic images; equivalently, the formation of quotient algebras by congruences. A variety $\mathcal{V}$ is *congruence distributive* if $(\operatorname{Con}\mathfrak{A}; \wedge, \vee)$ is distributive for every $\mathfrak{A} \in \mathcal{V}$. In this case, the algebras in $\mathcal{V}$ are also called congruence distributive. A variety $\mathcal{V}$ is said to be *congruence n-permutable* if for every $\mathfrak{A} \in \mathcal{V}$ and any $\alpha, \beta \in \operatorname{Con}\mathfrak{A}$, we have that $\alpha \circ_n \beta = \beta \circ_n \alpha$ (where $\alpha \circ_n \beta = \alpha \circ \beta \circ \alpha \circ \cdots \circ \delta_n$ and δ_n is the n^{th} compositional factor, either α or β according as n is odd or even). A *pseudovariety* only differs from a variety in that it need not be closed under the formation of infinite direct products of algebras.

The following result of G. Birkhoff says that every variety is an equational class, and vice-versa:

Theorem 1.3.2 ([18])**.** Every variety $\mathcal{V}$ is uniquely determined by a set Φ of identities (equalities of terms) $s = t$ such that $\mathfrak{A} \in \mathcal{V}$ if and only if $\mathfrak{A} \models \Phi$.

Note that a structure $\mathfrak{A}$ models a set of formulas Φ (written $\mathfrak{A} \models \Phi$) if $\mathfrak{A} \models \phi$ for each $\phi \in \Phi$; that is, every Φ-formula holds true in $\mathfrak{A}$.

Given an algebra $\mathfrak{A}$, the *variety generated by* $\mathfrak{A}$, which we will denote by $\mathrm{V}(\mathfrak{A})$, is the smallest variety containing $\mathfrak{A}$. We similarly define the *pseudovariety generated by* $\mathfrak{A}$, and denote this class by $\mathrm{V}_{\mathrm{fin}}(\mathfrak{A})$. If $\mathfrak{A}$ has a k-ary and a $(k+1)$-ary WNU term operation for $k \geq 3$, then $\mathrm{V}(\mathfrak{A})$ is said to be *congruence* $\wedge$*-semidistributive.*

We close this section with a useful characterization theorem of B. Jónsson, after whom Jónsson terms are named.

Theorem 1.3.3 ([1])**.** An algebra $\mathfrak{A}$ has Jónsson terms if and only if $\mathfrak{A}$ is congruence distributive.

A consequence of the preceding theorem is that every algebra with an NU term operation is congruence distributive. To see this, simply construct a chain of Jónsson terms, as done in [1].

1.4 Localization Theory

We now delve into some localization theory for algebras and templates. Broadly speaking, given a "global" problem involving structures (for example, a CSP), we can use information about certain substructures, or "local" data, to solve the problem.

Let $\mathfrak{A} = (A; \mathcal{F}_A)$ be an algebra where $\mathcal{F}_A$ is a functional clone on A. The restrictions of relations in $\mathrm{Inv}\,\mathfrak{A}$ to arbitrary $U \subseteq A$ can induce a homomorphism between algebras arising, respectively, from $\mathrm{Inv}\,\mathfrak{A}$ and $\mathrm{Inv}(U; \mathcal{F}_A)$. A criterion for this to occur is the following:

Theorem 1.4.1 ([39])**.** If $U \subseteq A$, then $\mathfrak{A}^{\perp} = (A; \mathrm{Inv}(A; \mathcal{F}_A)) \to \mathfrak{A}^{\perp}|_U$ if and only if

$U = e[A]$ for some unary operation $e \in \mathcal{F}_A$ with the property that $e(e(x)) = e(x)$ for all $x \in A$.

The preceding theorem tells us which subsets of A are suitable for localization, and so we give a special name to these. We say that $U \subseteq A$ is a *neighbourhood* of $\mathfrak{A}$ if $U = e[A]$ for some unary operation $e \in \mathcal{F}_A$ with the property that $e(e(x)) = e(x)$ for all $x \in A$. If U and V are two neighbourhoods of $\mathfrak{A}$ such that $\mathfrak{A}^{\perp}|_U \cong \mathfrak{A}^{\perp}|_V$, then we say that U and V are *isomorphic* and we write $U \equiv V$.

An immediate consequence of the above definitions is the following (see also Lemma 2.6 in [39]):

Theorem 1.4.2. Suppose U and V are two neighbourhoods of an algebra $\mathfrak{A} = (A; \mathcal{F}_A)$. Then $U \equiv V$ if and only if there exist unary operations $f, g \in \mathcal{F}_A$ such that $f\colon U \longrightarrow V$ and $g\colon V \longrightarrow U$ are inverse bijections.

If $U = e[A]$ is a neighbourhood of $\mathfrak{A}$ for some e, then the *algebra induced by* $\mathfrak{A}$ *on* U is defined to be

$$\mathfrak{A}|_U = e[\mathfrak{A}] = \mathfrak{A}^{\perp}\big|_U^{\;\perp} = \left(U; \left\{ f|_{U^{\operatorname{ar} f}} \;\middle|\; f \in \mathcal{F}_A,\ f[U^{\operatorname{ar} f}] \subseteq U \right\}\right) .$$

We thus have the following proposition (see also Lemma 2.8 in [39]):

Proposition 1.4.1. If $U = e[A]$ is a neighbourhood of $\mathfrak{A}$ for some e, then

$$\mathfrak{A}|_U = \left(U; \left\{ e \circ f|_{U^{\operatorname{ar} f}} \;\middle|\; f \in \mathcal{F}_A \right\}\right) .$$

Given $S, T \in \operatorname{Inv} \mathfrak{A}$ where $S \subseteq T$, we say that U is (S,T)-*minimal* if every function $g = e \circ f|_U$ from $\mathfrak{A}|_U$ is a permutation on U or $g[T] \subseteq S$.

Let $S, T \in \operatorname{Inv}\mathfrak{A}$ for $\mathfrak{A}$ as above. A set $\mathcal{U}$ of neighbourhoods of $\mathfrak{A}$ is a *cover* of $\mathfrak{A}$ if the condition $S|_U = T|_U$ for each $U \in \mathcal{U}$ implies that $S = T$. A set $\mathcal{V}$ of neighbourhoods of $\mathfrak{A}$ *covers* $U \in \mathcal{U}$ if the condition $S|_V = T|_V$ for each $V \in \mathcal{V}$ implies that $S|_U = T|_U$. We say that $\mathcal{V}$ *refines* $\mathcal{U}$ if every $V \in \mathcal{V}$ is contained in some $U \in \mathcal{U}$ and $\mathcal{V}$ covers every $U \in \mathcal{U}$.

The following theorem is fundamental in the theory of localization (and globalization) of algebras:

Theorem 1.4.3 ([39])**.** Let $\mathfrak{A} = (A; \mathcal{F}_A)$ be an algebra where $\mathcal{F}_A$ is a functional clone on A. The following conditions are equivalent:

(i) $\mathcal{U}$ is a cover of $\mathfrak{A}$;

(ii) $\mathfrak{A}$ satisfies an equation of the form

$$f\big(e_1(\rho_1(x)), \ldots, e_k(\rho_k(x))\big) = x\,,$$

where f is a k-ary operation in $\mathcal{F}_A$ and both e_i and ρ_i are unary operations in $\mathcal{F}_A$ such that $e_i[A] \in \mathcal{U}$ for all $i \in [k]$;

(iii) $\mathfrak{A}^{\perp}$ is a retract of a product of templates from $\big\{\mathfrak{A}^{\perp}|_U \;\big|\; U \in \mathcal{U}\big\}$.

Note that in the context of the preceding theorem, a set $\mathcal{V}$ of neighbourhoods of $\mathfrak{A}$ covers $U \in \mathcal{U}$ if and only if $\mathfrak{A}$ satisfies an equation of the form

$$f\big(e_1(\rho_1(x)), \ldots, e_k(\rho_k(x))\big) = e(x)\,,$$

where $e_i[A] \in \mathcal{V}$ for all $i \in [k]$ and $e[A] = U$.

The next result pertains to algebras $\mathfrak{A} = (A; \mathcal{F}_A)$, where $\mathcal{F}_A$ is a functional clone on A.

Theorem 1.4.4 ([39])**.** Up to isomorphism, every finite $\mathfrak{A}$ has a unique non-refinable cover.

Thus, in refining covers of $\mathfrak{A}$, we eventually find a cover not containing any neighbourhood that can be covered by the collection of its properly contained neighbourhoods. That brings us to our next definition concerning $\mathfrak{A}$, and the final concept we introduce in this section.

A neighbourhood U of $\mathfrak{A}$ is *irreducible* if each cover of $\mathfrak{A}|_U$ contains U. Alternatively (see, for example, [39]), U is irreducible if and only if U is (S, T)-minimal for some S and T.

1.5 Complexity Theory

A *decision problem* is a problem that can be posed as a yes-no question. As such, the question of whether an input has a certain property, characterized by membership in a set P, is just a set of the form $\{x \mid x \in P\}$. Any algorithm A for solving such a problem requires a certain amount of resources in order to run; this is known as the *(computational) complexity* of A. In particular, *time complexity* refers to the number of steps required by A to solve a problem, while *space complexity* refers to the amount of memory required. The worst-case time complexity for A would be the maximum number of steps required to solve a problem, which is usually expressed as a function of the size of the problem input. Some common running times are given in Table 1.1.

Table 1.1: Some common running times.

	Running Time, $T(n)$
Constant Time	$O(1)$
Logarithmic Time (or Log-time)	$O(\log n)$
Linear Time	$O(n)$
Quadratic Time	$O(n^2)$
Cubic Time	$O(n^3)$
Polynomial Time (or Poly-time)	$2^{O(\log n)} = n^{O(1)}$
Exponential Time	$2^{n^{O(1)}}$

Given a non-negative sequence $f\colon \mathbb{N} \longrightarrow \mathbb{R}$, the *time complexity class* $\textbf{TIME}(f(n))$ is the set of all decision problems with input size n that are solvable in $O(f(n))$ time. This may be viewed, upon encoding the problem inputs by strings, as the set of all languages that are decidable by an $O(f(n))$ time Turing machine (see [55]). The *non-deterministic time complexity class* $\textbf{NTIME}(f(n))$ is the set of all decision problems with input size n whose candidate solutions (or proofs of membership) are verifiable in $O(f(n))$ time. Again, upon encoding the problem inputs by strings, this may be viewed as the set of all languages that are decidable by an $O(f(n))$ time non-deterministic Turing machine. The classes $\textbf{SPACE}(f(n))$ and $\textbf{NSPACE}(f(n))$ are defined analogously. We refer the reader to [55] for more details.

The complexity classes in Table 1.2 are fundamental.

Table 1.2: Some canonical complexity classes.

Complexity Class	Definition
Exponential Time, **Exp**	$\textbf{\textsc{Time}}\left(2^{n^{O(1)}}\right)$
Interactive Polynomial Time or Polynomial Space, **IP**	$\textbf{\textsc{Space}}\big(n^{O(1)}\big)$
Non-deterministic Polynomial Time, **NP**	$\textbf{N\textsc{Time}}\big(n^{O(1)}\big)$
(Deterministic) Polynomial Time, **P**	$\textbf{\textsc{Time}}\big(n^{O(1)}\big)$
Non-deterministic Logarithmic Space, **NL**	$\textbf{N\textsc{Space}}\big(\log n\big)$
(Deterministic) Logarithmic Space, **L**	$\textbf{\textsc{Space}}\big(\log n\big)$

Moreover, *complement*-**NP**, denoted **co-NP**, is the set of all decision problems that are complements of those in **NP**.

Let $\mathcal{S}^*$ be the set of all strings over a symbol set $\mathcal{S}$. A function $f\colon \mathcal{S}^* \longrightarrow \mathcal{S}^*$ is **P**-*computable* if some poly-time Turing machine, on every input $x \in \mathcal{S}^*$, halts with only $f(x)$ on its tape. A language $\mathcal{A} \subseteq \mathcal{S}^*$ is **P**-*reducible* to a language $\mathcal{B} \subseteq \mathcal{S}^*$ (written $\mathcal{A} \leq_{\mathbf{P}} \mathcal{B}$) if there is a **P**-computable function $f\colon \mathcal{S}^* \longrightarrow \mathcal{S}^*$ such that $\mathcal{A} = f^{-1}[\mathcal{B}]$. In this case, f is called the **P**-*reduction* of $\mathcal{A}$ to $\mathcal{B}$. This formalizes the notion of a decision problem A being **P**-reducible to a decision problem B. In fact, $\leq_{\mathbf{P}}$ is a pre-order on the set of all decision problems. We say A and B are **P**-*equivalent* (written $A \equiv_{\mathbf{P}} B$) if $A \leq_{\mathbf{P}} B$ and $B \leq_{\mathbf{P}} A$. Definitions analogous to the preceding ones can be made for **L**. For any class of decision problems $\mathcal{D}$, we say that A is $\mathcal{D}$-*hard* if $D \leq_{\mathbf{P}} A$ for any $D \in \mathcal{D}$. If, in addition, we have that $A \in \mathcal{D}$, then A is said to be $\mathcal{D}$-*complete*. For example, the class of **NP**-complete problems is $\mathbf{NP\text{-}C} = \mathbf{NP} \cap \mathbf{NP\text{-}H}$, the intersection of **NP** and the class of all **NP**-hard problems. Other classes, such as **NL-C**, are similarly defined.

The figure below shows the containment hierarchy of specific complexity classes introduced thus far.

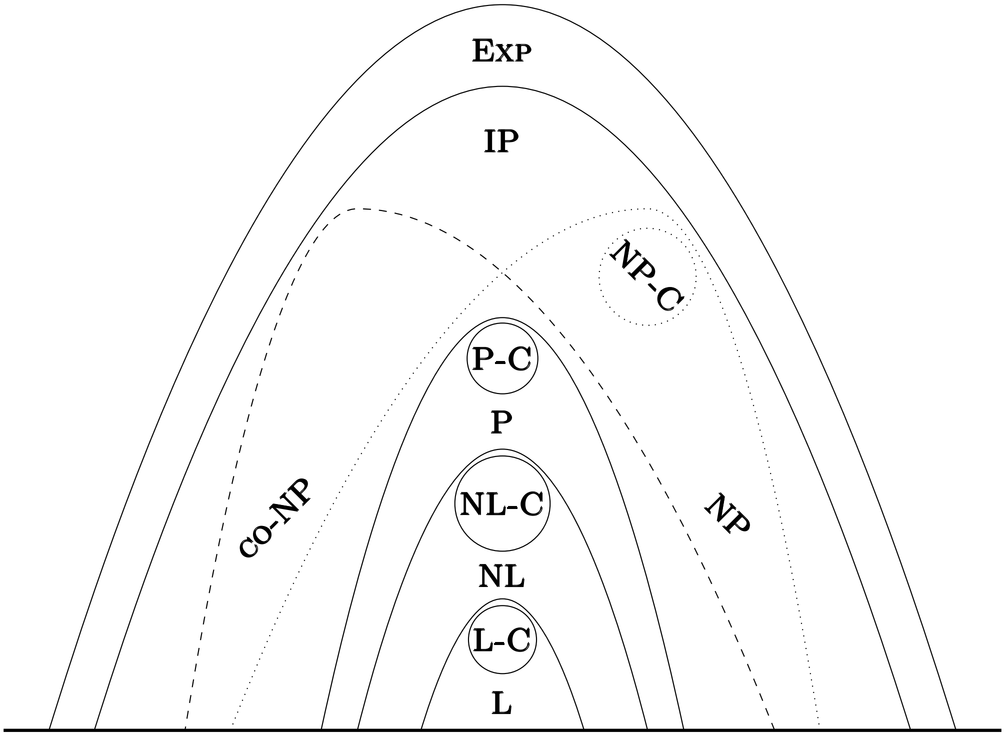


Figure 1.10: The hierarchy of various complexity classes.

Determining the exact containment relationships between such classes is a standard problem in modern complexity theory. The most famous problem of this kind is arguably the **P** versus **NP** problem, which is the open question of whether **P** is equal to **NP**.

The following result, known as the *Cook-Levin Theorem* (due to S. Cook ([20]) and L. Levin ([47])), says that the question of whether a Boolean formula is satisfiable — the problem known as Sat — is **NP**-complete:

Theorem 1.5.1 ([55])**.** Sat $\in$ **NP-C**

An interesting consequence of the Cook-Levin Theorem is that Sat $\in$ **P** if and only if **P** $=$ **NP**. R. Karp ([36]) used the theorem to show that there is a **P**-reduction from Sat to 21 different combinatorial and graph-theoretical problems, thus showing that each of them is **NP**-complete. Among these 21 problems is 3-SAT, the restriction of SAT to the case of at most 3 literals per clause. The aforementioned results are among the earliest that motivated the study of the **P** versus **NP** problem.

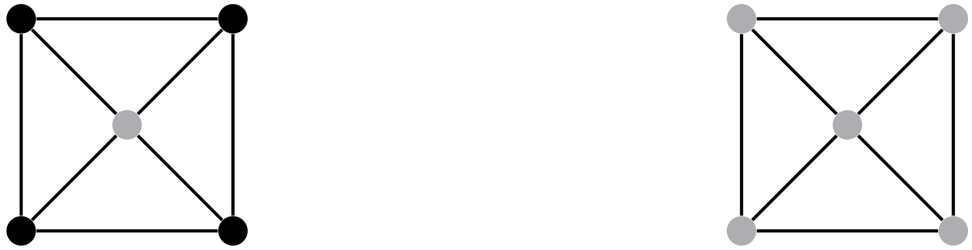
The restriction of 3-SAT to Horn clauses (that is, clauses containing at most one unnegated literal) is 3-HORN-SAT. Using an **L**-reduction, one can show, as noted in [5], that 3-HORN-SAT $\in$ **P-C**.

Descriptive complexity characterizes complexity classes by the type of logic needed to define their associated languages; that is, the languages which can be viewed as being in the classes. This provides a natural formalization, which does not depend on theoretical models of computer hardware, such as Turing machines. Natural measures of descriptive complexity include depth of quantifier nesting and number of variables in logical formulas ([35]). These correspond to traditional notions of machine-based complexity. For more on this approach to complexity theory, see [35].

We conclude this section with a formal description of CSPs, which, in general, are known to be in **NP**. An *instance* of a CSP is a finite triple $\mathcal{I} = (V, D, \mathcal{C})$, where V is a non-empty set $\{x_1, \ldots, x_n\}$ of variables, D is a domain for those variables, and $\mathcal{C}$ is a set of *constraints* of the form $C_x = (x, R_x)$ with $x \in V^k$ and $R_x = R_{x_1} \times \cdots \times R_{x_k} \subseteq D^k$ for each $k \leq n$. We call x the *scope* of C_x and R_x the *constraint relation* of C_x. An evaluation map $f \colon V \longrightarrow D$ *satisfies* $C_x \in \mathcal{C}$ if $f(x) = (f(x_1), \ldots, f(x_k)) \in R_x$. We say f is a *solution* for $\mathcal{I}$ if f satisfies $C_x \in \mathcal{C}$ for all x. We also say f is *optimal* if f satisfies the maximum possible number of constraints. If a template or polymorphism algebra is associated to $\mathcal{I}$, then the CSP is said to be *parametrized* by that structure.

1.6 Graph Burning

The process of burning a graph G takes place over discrete time-steps, or rounds. In round 1, we burn a node v_1 of G. In round $t > 1$, more nodes of G are burned: one of our choosing, v_t, and all neighbours of nodes burned in round $t-1$. Once a node is burned, it remains burned indefinitely, and the process ends when all nodes of G are burned — alternatively, when $V(G)$ is burned — say, in round k. In this way, fire spreads from burning nodes to unburned neighbours, according to a sequence $(v_1, \ldots, v_k)$ called a *burning sequence*. The terms of this sequence are called *sources of fire*. Figure 1.11 below gives a simple illustration of this process.



(a) Round 1

(b) Round 2

Figure 1.11: Typical burning of the wheel W_5 on 5 nodes. In Round 1, a fire breaks out at the central node. In Round 2, the fire spreads to neighbouring nodes, any one of which may be taken as the new source of fire.

The *burning number* of a graph G, denoted $b(G)$, is the minimum number of rounds necessary for $V(G)$ to burn; equivalently, the length of a shortest burning sequence for G. This indicates the level of ease or speed of spreading contagion in G. For example, as seen in Figure 1.11, $b(W_5) = 2$. It is also easy to see that for the complete graph K_n on n nodes, $b(K_n) = 1$ if $n = 1$ and $b(K_n) = 2$ if $n \geq 2$. Indeed, $b(G) \geq 2$ if $|V(G)| \geq 2$. The problem of computing $b(G)$ in general is **NP**-hard ([13]). For related approximation algorithms, see [13]. Many other results on the burning number have been published;

some key results are included in Section 2.1.

Two contrasting models for graph burning demonstrate how gossip can spread in a social network. In the *telephone model* ([13]), gossip spreads via traditional phone calls between neighbours, in some specific order. Thus, neighbours receive information one at a time. On the other hand, the *radio model* sees each neighbour broadcasting gossip to all other neighbours in a given round ([13]). Gossip, therefore, tends to spread more freely, although the sources of fire are pre-determined, and it is often assumed that neighbours have limited information about their environment (or network structure) ([13]).

An economic application of graph burning can be seen in viral marketing ([13]), which is a business strategy that refers to the cascading word-of-mouth effect concerning a product, in people's social networks — including, but not limited to, social media. See, for example, [41]. Influence maximization is the idea that a maximum number of people in a social network are eventually influenced by an initial group of informed individuals, who spread information simultaneously. This coincides with the notion of graph burning, when the initial group of individuals consists of just one person. Some approximation algorithms for influence maximization are given by Kempe et al. ([40]).

1.7 Datalog

As mentioned in [46], Datalog was conceived as a database query language. A Datalog program depends on a finite template $\mathfrak{B} = (B; \mathcal{R})$, and also takes a finite template $\mathfrak{A} = (A; \mathcal{R})$ as input. The program consists of finitely many *rules*, which are logical expressions of the form `φ_0 :- φ_1,...,φ_n`; that is,

$$\phi_0 \leftarrow \phi_1 \wedge \cdots \wedge \phi_n \,. \tag{1.1}$$

These are actually negation-free, function-free clauses, where ϕ_i is an atomic formula for $i = 0, 1, \ldots, n$. The formula ϕ_0 is called the *head* of the rule, and $\phi_1 \wedge \cdots \wedge \phi_n$ is called the *body.* The relation symbols in the heads of rules are not from $\mathcal{R}$, and are called *intensional database predicates (IDBs).* IDBs may also appear in the body of a rule; they are defined recursively. All of the other relation symbols belong to $\mathcal{R}$, and are known as *extensional database predicates (EDBs).* If $\phi_i = R(a)$ for some i where R is an IDB and $a = (a_1, \ldots, a_{\operatorname{ar} R}) \in A^{\operatorname{ar} R}$, then ϕ_i is called a *fact* of the program. One IDB, usually nullary, is designated as the *goal predicate*; the program accepts or rejects $\mathfrak{A}$ according as the formula containing the goal predicate is true or false in $\mathfrak{A}$. Specifically, the program checks $\mathfrak{A}$ for obstructions that prevent $\mathfrak{A}$ from being homomorphic to $\mathfrak{B}$. If obstructions are found, then the program (in accordance with the goal predicate) outputs `True`; otherwise, it outputs `False`. The program terminates polynomially in $|A|$.

A Datalog program is said to be *linear* whenever each of its rules is linear; that is, the body of each rule contains at most one IDB. If the rule (1.1) is linear and recursive such that, say, ϕ_1 is the only IDB in its body, then its *symmetrization* is the rule

$$\phi_1 \leftarrow \phi_0 \wedge \cdots \wedge \phi_n \, .$$

A linear Datalog program is said to be *symmetric* if it contains the symmetrizations of all its recursive rules. *Linear Datalog* is the restriction of Datalog to linear programs. Similarly, *symmetric Datalog* is the restriction of Datalog to symmetric programs. The term *k-Datalog* refers to Datalog programs with at most k variables in the head and in the body of each rule.

The semantics of Datalog programs are frequently defined in terms of fixed-point operators in the fixed-point logic ([26]), which is an extension of first-order logic. An equivalent notion is that of *derivation* ([23]): a Datalog program on input $\mathfrak{A} = (A; \mathcal{R})$ derives a fact $R_0(a)$ if it has a rule $R_0(x_0) \leftarrow R_1(x_1) \wedge \cdots \wedge R_n(x_n)$, and there is a map $f\colon \{x_0, x_1, \ldots, x_n\} \longrightarrow A$ defined by $f(x_0) = a$, such that $R_1(f(x_1)), \ldots, R_n(f(x_n))$ hold in $\mathfrak{A}$ or are derived by the program.

The CSP for a template $\mathfrak{H}$ is $\operatorname{CSP}\mathfrak{H} = \{\mathfrak{G} \mid \mathfrak{G} \to \mathfrak{H}\}$; that is, the *homomorphism problem* for $\mathfrak{H}$:

PROBLEM: Hom($\mathfrak{H}$)

INPUT: A template $\mathfrak{G}$.

QUESTION: Is $\mathfrak{G}$ homomorphic to $\mathfrak{H}$?

For example, let $\mathfrak{H} = (K_2; \leftrightarrow)$, where K_2 is the complete graph on 2 vertices and $\leftrightarrow$ is interpreted as adjacency. It then follows that $\operatorname{CSP}\mathfrak{H}$ is the undirected version of the *2-colouring problem* for graphs, which asks whether an undirected graph G can be properly coloured with 2 colours. By Theorem 1.2.2, G is not 2-colourable if and only if G has an odd cycle. The following linear Datalog program logically defines $\neg\operatorname{CSP}\mathfrak{H}$, detecting an odd cycle in G as an obstruction to a homomorphism between input $\mathfrak{G} = (G; \leftrightarrow)$ and $\mathfrak{H} = (K_2; \leftrightarrow)$:

```
R(x,y) :- E(x,y)

R(x,y) :- R(x,z), E(z,w), E(w,y)

S() :- R(x,x) ,
```

where `x` and `y` are vertices x and y of G, respectively; `E` is the EDB $\leftrightarrow$; `R` is an IDB interpreted as a relation asserting the existence of an odd x,y-path or an odd cycle in G; and `S` is the goal predicate. It can be shown, using first-order reductions ([5]), that

CSP $\mathfrak{H} \in \mathbf{L\text{-}C}$.

Even if a problem cannot be formulated as a CSP, it might be closely related to one. Such is the case for the *reachability problem*, which asks whether a node s reaches another node t in a graph G. The related CSP can be parametrized by the template $\mathfrak{B} = (\mathbb{B}; \{\{0\}, \{1\}, = \cup\, I\})$, where $I = \{(0,1)\}$ if G is directed and $I = \varnothing$ if G is undirected. To solve an instance $\mathcal{I}$ of CSP $\mathfrak{B}$, we form a graph G with vertices labelled 0 or 1 and edges corresponding to $= \cup\, I$. Now $\mathcal{I}$ has a solution if and only if a vertex labelled 1 does not reach a vertex labelled 0. Thus, CSP $\mathfrak{B}$ can be solved via $\neg$ CSP $\mathfrak{B}$. By the *Immerman-Szelepcsényi Theorem* ([5]), CSP $\mathfrak{B}$ is **NL**-complete if G is directed (see also [55]), and **L**-complete if G is undirected ([5]).

Another example of a CSP over a template $(F; \mathcal{S})$ is 3-Lin(p), where $p \in \mathbb{P}$. This is the problem of solving a linear system over a p-element field $(F; +, \cdot)$, where every equation has at most 3 variables; that is,

$$\mathcal{S} = \left\{ \{(x,y,z) \in F^3 \mid ax + by + cz = d\} \;\middle|\; a, b, c, d, \in F \right\}.$$

This problem is poly-time solvable, by Gaussian elimination ([5]).

The more general *list homomorphism problem* for a template $\mathfrak{H} = (H; \mathcal{S})$ is as follows:

PROBLEM: L-Hom$(\mathfrak{H})$

INPUT: A template $\mathfrak{G} = (G; \mathcal{R})$ with lists $L(\epsilon) \subseteq H$; $\epsilon \in G$.

QUESTION: Does there exist $f \in \operatorname{Hom}(\mathfrak{G}, \mathfrak{H})$ such that $f(\epsilon) \in L(\epsilon)$ for each $\epsilon \in G$?

For example, if G and H are graphs such that $\mathcal{R} = \mathcal{S} = \{\leftrightarrow\}$ and $L(\epsilon)$ is a list of colours for ϵ (a vertex or an edge), then L-Hom$(\mathfrak{H})$ is a *list colouring problem* for G. Observe that L-Hom$(\mathfrak{H})$ coincides with Hom$(\mathfrak{H})$ when $(H; \operatorname{Pol} \mathfrak{H})$ is conservative. In this case, all possible unary relations are among the basic relations of $\mathfrak{H}$, and we call both $\mathfrak{H}$ and

CSP $\mathfrak{H}$ *conservative.* The main reason for the study of conservative CSPs is that they present a test case for various conjectures on CSPs.

Feder and Vardi ([32]) showed that the CSP for any template is **P**-equivalent to the CSP for some associated digraph. Several years later, Bulín et al. ([17]) sharpened this result to **L**-equivalence. The previous example concerning reachability provides fundamental insight into the construction of the aforementioned digraphs.

The CSP for an algebra $\mathfrak{C} = (C; \mathcal{F})$, denoted by CSP $\mathfrak{C}$, is the set of all pairs $(\mathfrak{A}, \mathfrak{B})$ of templates such that $\mathfrak{A} \to \mathfrak{B}$ and Pol $\mathfrak{B}$ contains the basic operations of $\mathfrak{C}$. We say that $\mathfrak{C}$ is *globally tractable* if CSP $\mathfrak{C} \in \mathbf{P}$, and that $\mathfrak{C}$ is *locally tractable* if CSP $\mathfrak{U} \in \mathbf{P}$ for every template $\mathfrak{U}$ such that Pol $\mathfrak{U}$ contains the basic operations of $\mathfrak{C}$.

Existential k-pebble games (or *$(\exists, k)$-pebble games*) involve two players, the Spoiler and the Duplicator, who play on two templates $\mathfrak{A} = (A; \mathcal{R})$ and $\mathfrak{B} = (B; \mathcal{R})$. Each player initially has k pebbles (where $k \geq 1$), and the rules of the game are as follows ([43]): in a given round of the game, the Spoiler places a pebble on an element of A, and the Duplicator responds with a similar action on B; the Spoiler may remove some pebbles afterwards to begin a new round. The objective of the game for the Spoiler is to create a non-homomorphic mapping between the pebble-marked elements of the Spoiler and those of the Duplicator ([43]). Accordingly, the objective for the Duplicator is to permanently prevent such a mapping from being created. It follows that for a given finite template $\mathfrak{B}$, $\neg$ CSP $\mathfrak{B}$ is expressible in k-Datalog if and only if $\neg$ CSP $\mathfrak{B}$ is the set of all templates $\mathfrak{A}$ for which a $(\exists, k)$-pebble game played on $\mathfrak{A}$ and $\mathfrak{B}$ sees the Duplicator as the winner ([42, 43]). For a purely algebraic description of $(\exists, k)$-pebble games, see [43].

Given a finite template $\mathfrak{A} = (A; \mathcal{R})$, let $\mathfrak{A}^* = (A; \mathcal{R}^*)$ be its expansion by all possible relations on A. The *canonical linear k-Datalog program* (or, more simply, the k-program) on input $\mathfrak{A}$ for $k \in \mathbb{N}_0$ contains an EDB for every $R \in \mathcal{R}$ and an IDB for every $S \in \mathcal{R}^* \setminus \mathcal{R}$

where $\operatorname{ar} S \leq k$. The rules of the program express logical formulas valid in $\mathfrak{A}^*$.

We say that $\mathfrak{A} \in \operatorname{CSP} \mathfrak{B}$ passes the *k-test* if the k-program for $\mathfrak{B}$ on input $\mathfrak{A}$ outputs `True`. Accordingly, $\neg \operatorname{CSP} \mathfrak{B}$ is expressible in linear Datalog if and only if the k-program outputs `False`.

If a template $\mathfrak{A} = (A; \mathcal{R})$ is a core, then one can expand $\mathcal{R}$ by singleton unary relations to obtain a template $\mathfrak{A}_1 = (A; \mathcal{R} \cup \{\{a\} \mid a \in A\})$. This enables further restriction of $(A; \operatorname{Pol} \mathfrak{A}_1)$. Specifically, if $\phi \in \operatorname{Pol}_m \mathfrak{A}_1$, it is plain that $\phi(a, a, \ldots, a) = a$ for all $a \in A$. Additionally, $\operatorname{CSP} \mathfrak{A} \equiv_{\mathbf{L}} \operatorname{CSP} \mathfrak{A}_1$. Therefore, we may assume that the polymorphism algebra associated with any CSP under consideration is idempotent.

1.8 Dissertation Overview

In this chapter, we have reviewed the various concepts that are preliminary to our work. In Chapter 2, we will review some of the literature on graph burning; we will continue our discussion of this topic in Chapter 3. There, we will present our main result concerning a special case of the Cartesian grid graph. In Chapter 4, we will review some published results and conjectures on CSPs and Datalog. Chapter 5 will lay the necessary groundwork for proving the *Linear Datalog Conjecture*. Note that the results of Chapter 3 appear in [10], while the results of Chapter 5 appear in [27] and [28]. Finally, open problems and future directions will be considered in Chapter 6.

Chapter 2

Background on Graph Burning

We review some of the literature on graph burning, so as to provide some background on the topic and allow us to refer to results as needed. The results stated here, along with their proofs, can be found in the sources cited.

2.1 Results on Graph Burning

Let G be an undirected graph with $v \in V(G)$. For any $k \in \mathbb{N}_0$, the *(open) k-neighbourhood* of v in G is defined as

$$N_G(v;k) = \left\{ w \in V(G) \;\middle|\; \operatorname{dist}_G(v,w) = k \right\},$$

while the *closed k-neighbourhood* of v in G is defined as

$$N_G[v;k] = \left\{ w \in V(G) \;\middle|\; \operatorname{dist}_G(v,w) \leq k \right\} = N_G(v;0) \cup N_G(v;1) \cup \cdots \cup N_G(v;k).$$

Indeed, $N_G(v;0) = \{v\}$, $N_G(v;1) = N_G(v)$, and $\{v\} \cap N_G(v;i) = \varnothing$ for any $i \in [k]$. Members of $N_G[v;k]$ are referred to as *k-neighbours* of v in G. This set is essentially a ball of radius k centred at v.

If G is the $m \times n$ Cartesian grid, then for any $v \in V(G)$, we have that

$$\begin{aligned}\big|N_G[v;k]\big| &\leq \big|\{v\}\big| + \big|N_G(v)\big| + \cdots + \big|N_G(v;k)\big| = 1 + \deg_G v + \cdots + k \deg_G v \\ &\leq 1 + 4 + \cdots + 4k = 1 + 4(1 + \cdots + k) = 1 + 4 \cdot \frac{k(k+1)}{2} = 1 + 2k(k+1)\,.\end{aligned}$$

The first result we present in this chapter reduces the process of burning a graph G to a “ball decomposition” of $V(G)$.

Lemma 2.1.1 ([12]). If G is a graph, then $(x_1, \ldots, x_k)$ is a burning sequence for G if and only if for any $i, j \in [k]$ where $i < j$, we have that $\operatorname{dist}_G(x_i, x_j) \geq j - i$ and

$$N_G[x_1; k-1] \cup \cdots \cup N_G[x_k; 0] = V(G)\,.$$

The following theorem gives an upper bound on the burning number of a graph G, via coverings of $V(G)$:

Theorem 2.1.1 ([12]). Let G be a graph such that $H_1, \ldots, H_t \subseteq G$ are connected and $\operatorname{rad} H_i \leq k - i$ for $1 \leq i \leq t \leq k$. If $V(H_1) \cup \cdots \cup V(H_t) = V(G)$, then $b(G) \leq k$.

As an immediate corollary, we have:

Corollary 2.1.1 ([12]). If $(x_1, \ldots, x_k)$ is a sequence of vertices from a graph G such that $N_G[x_1; k-1] \cup \cdots \cup N_G[x_k; 0] = V(G)$, then $b(G) \leq k$.

Thus, the preceding theorem and corollary demonstrate that for any graph G, suitable decomposition of $V(G)$ into at most k subsets ensures that $b(G) \leq k$.

If H is a subgraph of G, it does not follow in general that $b(H) \leq b(G)$. The next theorem speaks to this fact.

Theorem 2.1.2 ([12])**.** If H is a spanning subgraph of G, then $b(G) \leq b(H)$.

Hence, the burning number is order-reversing for spanning subgraphs.

If T is a tree with root u, then the *depth* of $v \in V(T)$ is defined to be

$$\operatorname{depth}_T v = \operatorname{dist}_T(v, u) \,.$$

It turns out that the burning of a tree T in k rounds is equivalent to a partition of T into k suitable subtrees. We have the following theorem:

Theorem 2.1.3 ([11, 12])**.** For any graph G, we have that $b(G) \leq k$ if and only if there is a rooted tree partition $\{T_1, \ldots, T_k\}$ of G with

$$\max_{v \in V(T_i)} \operatorname{depth}_{T_i} v \leq k - i \quad \text{for all} \quad i \in [k] \,,$$

and distance between the roots of T_i and T_j at least $|i - j|$ for all $i, j \in [k]$.

For an illustration of the preceding theorem, refer back to Figure 1.6, and consider the burning sequence $(x_1, x_2, x_3, x_4) = (v_1, v_7, v_{12}, v_l)$, where $l \in \{9, 10, 11\}$.

The following corollary reduces the task of burning a graph to the task of burning its spanning trees:

Corollary 2.1.2 ([11, 12])**.** If $\mathcal{S}$ is the set of all spanning trees of a graph G, then

$$b(G) = \min_{H \in \mathcal{S}} b(H) \,.$$

The proof of Corollary 2.1.2 follows from Theorem 2.1.3 by taking $b(G) = k$, and then adding edges between $T_1, \ldots, T_k$ to form $H \in \mathcal{S}$. See [11] or [12] for details.

The next theorem gives sufficient conditions for monotonicity of the burning number on isometric subgraphs.

Theorem 2.1.4 ([12])**.** Let H be an isometric subgraph of G. Suppose that for each $v \in V(G) \smallsetminus V(H)$ and each $r \in \mathbb{N}$, there exists $v_r^* \in V(H)$ satisfying the following condition: $N_G[v; r] \cap V(H) \subseteq N_H[v_r^*; r]$. Then $b(H) \leq b(G)$.

It follows that if H is an isometric tree subgraph of G, then $b(H) \leq b(G)$. We invite the reader to consult [12] for the proof.

Let G be a graph such that $D_k \subseteq V(G)$. Suppose that for each $u \in V(G) \smallsetminus D_k$, there exists $v \in D_k$ such that $\mathrm{dist}_G(u, v) \leq k$. Then D_k is called a *k-distance dominating set* for G. The minimum size of a k-distance dominating set for G is called the *k-distance domination number* of G. Using a known bound on this number (see [8] and [11]), Bonato et al. ([11, 12]) established an upper bound on $b(G)$ which is dependent on $|V(G)|$:

Theorem 2.1.5 ([12])**.** Suppose G is a connected graph of order n. We then have that $b\big(G\big) \leq 2\big\lceil\sqrt{n}\big\rceil - 1$.

The following computational result is now used extensively:

Theorem 2.1.6 ([11, 12])**.** If G is a path or cycle on n vertices, then $b\big(G\big) = \big\lceil\sqrt{n}\big\rceil$.

As an immediate corollary of the preceding theorem, we have the following result:

Corollary 2.1.3 ([12])**.** If G is a graph of order n containing a Hamiltonian path or cycle, then $b\big(G\big) \leq \big\lceil\sqrt{n}\big\rceil$.

A more powerful assertion is the so-called *Burning Conjecture*:

Conjecture 2.1.1 ([12])**.** If G is a connected graph of order n, then $b\big(G\big) \leq \lceil \sqrt{n} \rceil$.

The next result gives the best known upper bound on the burning number of a connected graph.

Theorem 2.1.7 ([45])**.** For any connected graph G of order n,

$$b\big(G\big) \leq \left\lceil \frac{\sqrt{24n+33}-3}{4} \right\rceil .$$

The preceding upper bound is approximately $\big(\sqrt{6}/2\big)\sqrt{n}$. The previous best known upper bound ([8]) was $\big(\sqrt{32/19}+o(1)\big)\sqrt{n}$.

The next theorem gives sharp bounds on the burning number, in terms of eccentricity.

Theorem 2.1.8 ([11, 12])**.** For any graph G, we have that

$$\left\lceil \sqrt{\operatorname{diam} G+1} \right\rceil \leq b(G) \leq \operatorname{rad} G + 1 \,.$$

In particular, $b\big(P_n\big) = \left\lceil \sqrt{\operatorname{diam} P_n + 1} \right\rceil$ while $b(H) = \operatorname{rad} H + 1$ for certain spider graphs H (see [12]).

Consider the following decision problem:

PROBLEM: Burning

INSTANCE: A graph G of order n and $k \in \mathbb{N} \smallsetminus \{1\}$.

QUESTION: Is $b(G) \leq k$?

Bessy et al. ([7]) established the following:

Theorem 2.1.9 ([7])**.** The problem Burning is **NP**-complete for trees of maximum degree 3 and spider graphs.

As a corollary of the preceding theorem, Burning is **NP**-complete for path-forests and forests of maximum degree 3, as well as bipartite, chordal, and planar graphs. See [7] for details.

We conclude our review of the literature on graph burning, with a fairly recent result on the burning number of Cartesian and strong graph products:

Theorem 2.1.10 ([50])**.** If G and H are two connected graphs, then

$$\max\{b(G), b(H)\} \leq b(G \boxtimes H) \leq b(G \,\square\, H) \leq \min\{b(G) + \operatorname{rad} H, b(H) + \operatorname{rad} G\}\,.$$

In particular, $b(L_n) = b(G_{2,n}) = \max\{b(P_2), b(P_n)\}$ if $n \in \{k^2+1, k^2+2 \mid k \in \mathbb{N}\}$, and $b(G^{\times}_{2,n}) = \max\{b(P_2), b(P_n)\}$ if $n \in \{k^2 \mid k \in \mathbb{N}\}$ (as indicated in [50]). Also, $\min\{b(G) + \operatorname{rad} H, b(H) + \operatorname{rad} G\}$ is tight if $\operatorname{rad} G = 1$ and $H \cong P_{n^2}$ for some $n \in \mathbb{N}$.

Chapter 3

Burning Fence Graphs

We now focus on the burning number of specific grid graphs. The value of $b(G_{m,n})$, the burning number of the $m \times n$ Cartesian grid, was first studied in [49]. All graphs considered in this chapter are undirected.

3.1 Burning Fences

Our main result of this chapter concerns the following theorem, and the questions it left open after it was published:

Theorem 3.1.1 ([49])**.** For $m = m(n)$,

$$b\big(G_{m,n}\big) = b\big(P_m \,\square\, P_n\big) = \begin{cases} \sqrt[3]{\dfrac{3}{2}}\,\big(1+o(1)\big)\,\sqrt[3]{mn}\,, & n \geq m = \omega\big(\sqrt{n}\big) \\ \\ \Theta\big(\sqrt{n}\big)\,, & m = O\big(\sqrt{n}\big)\,. \end{cases}$$

Two specific cases worth considering in the context of the preceding theorem are $m = \sqrt[k]{n} = o(\sqrt{n})$ for any $k \in \mathbb{N} \setminus [2]$ (which produces a "skinny" grid) and

$m = n = \omega(\sqrt{n})$ (which produces a square grid). Notice that while Theorem 3.1.1 gives an asymptotically tight value for $b(G_{m,n})$ in the case where $n \geq m = \omega(\sqrt{n})$, only the growth rate is given in the remaining case where $m = O(\sqrt{n})$. For valid $c > 0$, we refer to the grid $G_{c\sqrt{n},n}$ as a *fence*, since it is by definition wider than tall. By "valid" $c > 0$, we mean all $c > 0$ such that $c\sqrt{n} \in \mathbb{N}$, as per the definition of a Cartesian grid. The figure below illustrates a burning sequence for the fence $G_{4,16}$.

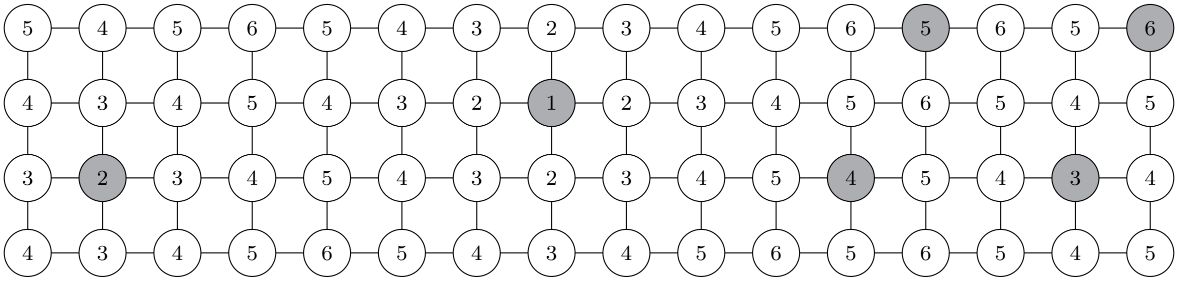


Figure 3.1: A 4×16 fence, where a vertex is labelled i if it is burned in round i. The burning sequence depicted here is $(x_1, x_2, x_3, x_4, x_5, x_6)$, where x_i is the grey vertex labelled i for $i \in [6]$.

We improve on Theorem 3.1.1, by giving explicit lower and upper bounds on the burning number of fences.

We prove the following theorem, which is our main result:

Theorem 3.1.2. For all $c > 0$ such that $c\sqrt{n} \in \mathbb{N}$, we have that

$$a \leq b\big(G_{c\sqrt{n},n}\big) \leq 2\sqrt{\lceil (c/2)^{2/3} \rceil n} + \lceil (c/2)^{2/3} \rceil - 1\,,$$

where

$$a = \begin{cases} \dfrac{1}{2}\left(c + \sqrt{4 - c^2}\right)\big(1 + o(1)\big)\sqrt{n}\,, & c < 2 \\ \\ \left\lceil \sqrt{n \max\left\{k \in \mathbb{N} \mid (k-1)\sqrt{kn} + 1 \leq c\sqrt{n}\right\}} \right\rceil\,, & c \geq 2\,. \end{cases}$$

In particular, if $c \leq 2\sqrt{2}$, then

$$b\big(G_{c\sqrt{n},n}\big) \leq \left(\frac{c}{2} + \sqrt{1 - \frac{c^2}{16}}\right)\big(1 + o(1)\big)\sqrt{n}\,.$$

The lower bound in Theorem 3.1.2 will follow immediately from Theorems 3.2.1 and 3.2.2, while the upper bound will follow from Theorems 3.3.1 and 3.3.2.

The most interesting case for fences $G_{c\sqrt{n},n}$ might just be $c = 1$. In this case, our lower bound is

$$\frac{1+\sqrt{3}}{2}\big(1 + o(1)\big)\sqrt{n} \approx 1.366\sqrt{n}\,,$$

while our upper bound is

$$\frac{2+\sqrt{15}}{4}\big(1 + o(1)\big)\sqrt{n} \approx 1.468\sqrt{n}\,.$$

Also, by Theorem 3.1.1,

$$b\big(G_{\sqrt{n},n}\big) = \sqrt[3]{\frac{3}{2}}\,\big(1 + o(1)\big)\sqrt{n} \approx 1.145\sqrt{n}\,,$$

which is well below our lower bound. This illustrates the fact that the value of $b(G_{m,n})$ given in Theorem 3.1.1 for $m = \omega(\sqrt{n})$ does not hold for $m = O(\sqrt{n})$.

3.1.1 Partial Burning

In burning a graph G, we tend to focus on certain subsets S of $V(G)$ that burn. We define the *burning number of G with respect to S* as the minimum number of rounds necessary for S to burn in G. We denote this parameter by $b(G; S)$. Observe that $b(G; V(G)) = b(G)$.

In [10], it was noted that adding edges between distinct components of a graph can increase the burning number by at most one. We now generalize and extend this fact to the setting of partial burning.

Lemma 3.1.1. If H is a subgraph of G and $X \subseteq V(H)$, then

$$b(G;X) \leq b(H;X) \leq b(G;X) + |E(G)| - |E(H)| \, .$$

Proof. It suffices to show, inductively, that:

$$b(H;X) \leq b(H+u+uv;X)+1 \ \text{ for any } \ u \in V(G) \setminus V(H) \ \text{ and } \ v \in V(H) \ ; \tag{3.1}$$

and

$$b(H;X) \leq b(H+uv;X)+1 \ \text{ for any } \ uv \notin E(H) \ \text{ where } \ u,v \in V(H) \, . \tag{3.2}$$

For (3.1), we have that $b(H;X) = b(H+u+uv;X)$, since u can only spread fire to v. Hence, any burning sequence containing u can be replaced by a burning sequence containing v and excluding u, thus giving a burning sequence for H with respect to X. For (3.2), fix a burning sequence $(v_1, v_2, \ldots, v_{b(H+uv;X)})$ that is optimal for $H+uv$ with respect to X. Assume without loss of generality that, when burning according to this sequence, u is burned in round i and v is burned in round $j \geq i$ if u and v are both burned. Otherwise, if one of these vertices is not burned after $b(H+uv;X)$ rounds, then take v to be the one not burned. Observe that $(v, v_1, v_2, \ldots, v_{b(H+uv;X)})$ is a burning sequence of length $b(H+uv;X)+1$ for H with respect to X, since the only effect uv could have is allowing the spread of fire from u to v, but here v would already be burned. $\square$

Two ideas in particular inspired the preceding lemma: firstly, connecting pairs of disjoint graphs via an arbitrary number of edges; secondly, graph sums as spanning subgraphs.

3.2 Lower Bound

Our lower bound for the burning number of fences will follow from analyzing the partial burning number of certain paths in $G_{m,n}$. Given a path P in $G_{m,n}$, we will say that P is a *horizontal path at height* h if $V(P) \subseteq \{v_{1,h}, \ldots, v_{n,h}\}$, where $v_{i,h}$ is the vertex of $G_{m,n}$ with Cartesian coordinates $(i,h) \in [n] \times [m]$. That is, for $G_{m,n}$ as depicted in Figure 1.9, we have that the bottom horizontal n-path is at height 1, while the horizontal n-path above it is at height 2, and so on, with the top horizontal n-path being at height m.

Our next result will allow us to bound the number of vertices burned via each source of fire, whenever we are burning paths that are sufficiently far apart.

Lemma 3.2.1. Let $P^{(1)}, \ldots, P^{(k)} \cong P_n$ be horizontal paths in $G = G_{m,n}$ at heights $h_1, \ldots, h_k$ respectively, where $h_1 < \cdots < h_k$ and $h_i - h_{i-2} \geq 2t + 2$ for each $i \in [k] \setminus \{1\}$. Let $V_P = V(P^{(1)}) \sqcup \cdots \sqcup V(P^{(k)})$. For $t \leq (n-1)/2$, we have that

$$\max_{v \in V(G)} \left| N_G[v;t] \cap V_P \right| = \max_{v \in V_P} \left| N_G[v;t] \cap V_P \right| .$$

Proof. Let $v \in V(G)$. By the bound on the distance between the paths stated in the hypotheses, $N_G[v;t]$ intersects at most two of the horizontal paths $P^{(i)}$, and so

$$\begin{aligned} \left| N_G[v;t] \cap V_P \right| \leq \max &\left\{ 2 \left[t - \operatorname{dist}_G \left(v, V\left(P^{(a)}\right)\right)\right] + 1, 0 \right\} \\ &+ \max \left\{ 2 \left[t - \operatorname{dist}_G \left(v, V\left(P^{(a+1)}\right)\right)\right] + 1, 0 \right\} , \end{aligned}$$

where $P^{(a)}$ and $P^{(a+1)}$ are the two horizontal paths closest to v. If

$$\max\left\{\operatorname{dist}_G\left(v, V\big(P^{(a)}\big)\right), \operatorname{dist}_G\left(v, V\big(P^{(a+1)}\big)\right)\right\} > t\,,$$

then

$$\left|N_G[v;t] \cap V_P\right| \leq 2t+1 \leq \left|N_G[v^*;t] \cap V_P\right|,$$

where v^* is a central vertex in $P^{(1)}$. Otherwise,

$$\begin{aligned}\left|N_G[v;t] \cap V_P\right| &\leq 4t - 2\left[\operatorname{dist}_G\left(v, V\big(P^{(a)}\big)\right) + \operatorname{dist}_G\left(v, V\big(P^{(a+1)}\big)\right)\right] + 2 \\ &\leq 4t - 2\operatorname{dist}_G\left(V\big(P^{(a)}\big), V\big(P^{(a+1)}\big)\right) + 2 \\ &= \left|N_G[v^{**};t] \cap V_P\right|,\end{aligned}$$

where v^{**} is the closest vertex to v in $P^{(a)}$ (sufficiently central). This completes the proof. □

The preceding lemma leads to lower and upper bounds on $b(G;V_P)$. We thus have the following lemma:

Lemma 3.2.2. Let $P^{(1)}, \ldots, P^{(k)} \cong P_n$ be horizontal paths in $G = G_{m,n}$ at heights $h_1, \ldots, h_k$ respectively, where $h_1 < \cdots < h_k$ and $h_i - h_{i-1} \geq \left\lceil \sqrt{kn} \right\rceil$ for each $i \in [k] \smallsetminus \{1\}$. We then have that

$$\left\lceil \sqrt{kn} \right\rceil \leq b\big(G; V(P^{(1)}) \sqcup \cdots \sqcup V(P^{(k)})\big) \leq \left\lceil \sqrt{kn} \right\rceil + k - 1\,.$$

Proof. Let $V_P = V(P^{(1)}) \sqcup \cdots \sqcup V(P^{(k)})$. We first prove the lower bound. Since the horizontal paths $P^{(i)}$ are far apart,

$$\max_{v\in V_P}\left|N_G[v;t]\cap V_P\right|\le 2t+1$$

for $1\le t<\left\lceil\sqrt{kn}\right\rceil$. Furthermore, by Lemma 3.2.1, $|N_G[v;t]\cap V_P|\le 2t+1$ for all $v\in V(G)$. Now, if $\left(v_1,v_2,\ldots,v_{\lceil\sqrt{kn}\rceil-1}\right)$ is a burning sequence, then

$$\sum_{i=1}^{\lceil\sqrt{kn}\rceil-1}\left|N_G\left[v_i;\left\lceil\sqrt{kn}\right\rceil-i-1\right]\cap V_P\right|\le\sum_{i=0}^{\lceil\sqrt{kn}\rceil-2}(2i+1)=\left(\left\lceil\sqrt{kn}\right\rceil-1\right)^2<kn=\left|V_P\right|.$$

Thus, V_P cannot possibly be burned according to the burning sequence, implying that $b(G;V_P)\ge\left\lceil\sqrt{kn}\right\rceil$.

For the upper bound, observe that

$$b\big(G;V_P\big)\le b\big(G[V_P]\big)=b\big(kP_n\big)\le b\big(P_{kn}\big)+k-1=\left\lceil\sqrt{kn}\right\rceil+k-1\,,$$

where the last inequality follows from Lemma 3.1.1 applied with $X=V(P_{kn})$. □

To establish a lower bound on $b(G_{c\sqrt{n},n})$ for all $c>0$, we consider the two cases separately: $0<c<2$ and $c\ge 2$. In the first case, our strategy for obtaining the lower bound is to partially burn the top and bottom horizontal paths. In the second case, we will consider the partial burning number of a collection of horizontal paths that are sufficiently far apart.

Theorem 3.2.1. If $0<c<2$ such that $c\sqrt{n}\in\mathbb{N}$, then

$$b\big(G_{c\sqrt{n},n}\big)\ge\frac{1}{2}\left(c+\sqrt{4-c^2}\right)\big(1+o(1)\big)\sqrt{n}\,.$$

Proof. Let $P_\perp, P_\top \cong P_n$ be the horizontal paths at heights 1 and $c\sqrt{n}$, (that is, the bottom and top path), respectively. Let $V_P = V(P_\perp) \sqcup V(P_\top)$. We will bound the number $k = b(G; V_P)$, where $G = G_{c\sqrt{n},n}$.

Suppose $(v_1, v_2, \ldots, v_k)$ is a shortest burning sequence, and let $a = b(G_{c\sqrt{n},n})$. Given $v^* \in V_P$ and $0 \le t < c\sqrt{n}$, we have $|N_G[v^*; t] \cap V_P| \le 2t + 1$, and so by Lemma 3.2.1, $|N_G[v_{a-t}; t] \cap V_P| \le 2t + 1$. This implies that the fire from any vertex v_{a-t} where $0 \le t < c\sqrt{n} - 1$ can spread to at most

$$\sum_{i=0}^{c\sqrt{n}-1} (2i + 1) \tag{3.3}$$

vertices in V_P.

Now, if $v^* \in V_P$ and $c\sqrt{n} \le t \le a - 1$, then

$$\left|N_G[v^*; t] \cap V_P\right| \le 2t + 2 + 2\big(t - c\sqrt{n} - 1\big)\,,$$

and so by Lemma 3.2.1, the fire from any vertex v_{a-t} where $c\sqrt{n} \le t < a$ can spread to at most

$$\sum_{i=c\sqrt{n}}^{a-1} \big[2i + 2 + 2\left(i - \left(c\sqrt{n} - 1\right)\right)\big] \tag{3.4}$$

vertices in V_P.

The sum of (3.3) and (3.4) must be greater than or equal to $2n = |V_P|$. Such inequality holds if and only if

$$a \ge \frac{1}{2}c\sqrt{n} \pm \sqrt{(c^2+4)n + 2c^2n - 4c^2n - 4c^2\sqrt{n} - 1} - \frac{3}{2}\,.$$

Grouping the dominant terms, we obtain

$$a \geq \left(\frac{1}{2}c + \frac{1}{2}\sqrt{4-c^2}\right)\big(1+o(1)\big)\sqrt{n} = \frac{1}{2}\left(c + \sqrt{4-c^2}\right)\big(1+o(1)\big)\sqrt{n}\,,$$

as desired. □

The second last inequality in the preceding proof was obtained via SageMath (computer algebra software).

We now consider the second case.

Theorem 3.2.2. If $c \geq 2$ such that $c\sqrt{n} \in \mathbb{N}$, then

$$b\big(G_{c\sqrt{n},n}\big) \geq \left\lceil \sqrt{n \max\left\{k \in \mathbb{N} \;\middle|\; (k-1)\sqrt{kn}+1 \leq c\sqrt{n}\right\}} \right\rceil .$$

Proof. Let $M = \max\left\{k \in \mathbb{N} \;\middle|\; (k-1)\sqrt{kn}+1 \leq c\sqrt{n}\right\}$. For each $i \in [M]$, let $P^{(i)}$ be the horizontal path in $G_{c\sqrt{n},n}$ at height $1 + \left\lceil \sqrt{Mn} \right\rceil (i-1)$. The conditions of Lemma 3.2.2 are then satisfied with $k = M$, so we have that

$$b\big(G_{c\sqrt{n},n}\big) \geq b\big(G_{c\sqrt{n},n}; V(P^{(1)}) \sqcup \cdots \sqcup V(P^{(M)})\big) \geq \left\lceil \sqrt{Mn} \right\rceil ,$$

and the proof follows. □

In the next section, we establish an upper bound on $b(G_{c\sqrt{n},n})$ for all $c > 0$.

3.3 Upper Bound

Before we consider an upper bound for the burning number of fences, we will show that determining the asymptotics for $b(G_{m,n})$ when $m = o(\sqrt{n})$ is straightforward. Noting that $\operatorname{rad} P_n = \lfloor n/2 \rfloor$, we have the following corollary of Theorem 2.1.10:

Corollary 3.3.1. *If $m = o\big(\sqrt{n}\big)$, then $b\big(G_{m,n}\big) = \big(1 + o(1)\big)\sqrt{n}$.*

Proof. Let $m = o\big(\sqrt{n}\big)$. Theorem 2.1.10 implies that

$$b(G_{m,n}) \leq b(P_n) + \operatorname{rad}(P_m) = \sqrt{n} + o\big(\sqrt{n}\big) = \big(1 + o(1)\big)\sqrt{n}\,.$$

We also have that $b(G_{m,n}) \geq \big(1 + o(1)\big)\sqrt{n}$, since any horizontal n-path in $G_{m,n}$ burns in at least $\lceil \sqrt{n} \rceil$ rounds. □

We will now turn our attention back to fences.

To establish an upper bound on the burning number of fences, we first present a lemma which is a useful generalization of Theorem 2.1.10.

Lemma 3.3.1. *Let G be a graph with $X \subseteq V(G)$. If $\operatorname{dist}_G(v, X) \leq l$ for each $v \in V(G)$, then*

$$b(G) \leq b(G; X) + l\,.$$

Proof. First burn X in $b(G;X)$ rounds. Regardless of what other vertices of G are burned, after at most l more rounds, G will be burned. □

Given the preceding lemma, we can bound the burning number of our fence $G_{c\sqrt{n},n}$ from above, if we can efficiently estimate $b\left(G_{c\sqrt{n},n}; X\right)$ for some suitable $X \subseteq V(G)$. For our purposes, we will always choose X to be the vertex set of a sum of horizontal paths.

If $P^{(1)}$ and $P^{(2)}$ are two horizontal paths that are sufficiently far apart, then

$$b\left(G_{c\sqrt{n},n}; V\big(P^{(1)}\big) \sqcup V\big(P^{(2)}\big)\right) = 2b\big(P_n\big)\,.$$

When c is large, we find it useful to only burn a few horizontal paths that are sufficiently far apart and equidistant. In this case, since we need not worry about interactions between the horizontal paths we burn, we can easily obtain an upper bound for $b(G_{c\sqrt{n},n})$.

Theorem 3.3.1. If $k = \lceil (c/2)^{2/3} \rceil$ where $c > 0$, then $b\big(G_{c\sqrt{n},n}\big) \le 2\sqrt{kn} + k - 1$.

Proof. For $i \in [k-1] \sqcup \{0\}$, let $P^{(i)}$ denote the horizontal path in $G = G_{c\sqrt{n},n}$ at height $(2\sqrt{kn}+1)i + \sqrt{kn} + 1$. Let $V_P = V(P^{(0)}) \sqcup V(P^{(1)}) \sqcup \cdots \sqcup V(P^{(k-1)})$. We then have that

$$b\big(G; V_P\big) \le b\big(kP_n\big) \le \sqrt{kn} + k - 1\,,$$

where the latter inequality follows from Lemma 3.1.1, and the fact that $kP_n \subset P_{kn}$ where

$$\big|E(kP_n)\big| = \big|E(P_{kn})\big| - (k-1)\,.$$

By our choice of k, the following hold true: $\mathrm{dist}_G(V(P^{(i)}), V(P^{(i+1)})) = 2\sqrt{kn}+1$ for each i, path $P^{(0)}$ is at height $\sqrt{kn}+1$, and $P^{(k-1)}$ is at height

$$(2k-1)\sqrt{kn} + k \ge 2k^{3/2}\sqrt{n} - \sqrt{kn} \ge c\sqrt{n} - \sqrt{kn}\,.$$

Consequently, every vertex in G is of distance at most $\sqrt{kn}$ from a burned vertex. Thus, by Lemma 3.3.1, after at most $\sqrt{kn}$ more rounds, G is burned. □

For small values of c, the upper bound in the preceding theorem can be improved. Just take two horizontal paths $P^{(1)}$ and $P^{(2)}$ that are close enough to each other, so that

$$b\left(G_{c\sqrt{n},n}; V\big(P^{(1)}\big) \sqcup V\big(P^{(2)}\big)\right) < 2b\big(P_n\big)\,.$$

We thus have the following lemma:

Lemma 3.3.2. For $0 < c \leq \sqrt{2}$, let $P_\perp, P_\top \cong P_n$ be the horizontal paths in $G_{c\sqrt{n},n}$ at heights 1 and $c\sqrt{n}$, respectively. We then have that

$$b\big(G_{c\sqrt{n},n}; V(P_\perp) \sqcup V(P_\top)\big) \leq \left(\frac{c}{2} + \sqrt{1 - \frac{c^2}{4}}\right)\big(1 + o(1)\big)\sqrt{n}\,.$$

Proof. Let $m = c\sqrt{n}$ and let $V(G) = \{v_{i,j} \mid (i,j) \in [m] \times [n]\}$ where $G = G_{c\sqrt{n},n}$. For any $i \in [m]$ and $t \geq m - 1$, we have that

$$N_G[v_{1,i}; t] \cap N_G[v_{m,i+2t-m+1}; t-1] = \varnothing\,,$$

while the intersection of

$$N_G[v_{1,i}; t] \sqcup N_G[v_{m,i+2t-m+1}, t-1]$$

with $V(P_\top)$ and with $V(P_\perp)$ induces a connected path in either case. Analogously, for any $i \in [m]$ and $t \geq m - 1$, we have that

$$N_G[v_{m,i}; t] \cap N_G[v_{1,i+2t-m+1}; t-1] = \varnothing\,,$$

while

$$N_G[v_{m,i}; t] \sqcup N_G[v_{1,i+2t-m+1}; t-1]$$

intersects $V(P_\top)$ and $V(P_\perp)$ in a connected path in each case. Thus, letting

$$a = \left(\frac{c}{2} + \sqrt{1 - \frac{c^2}{4}}\right)\big(1 + o(1)\big)\sqrt{n}\,,$$

we may assume that the first $a-m$ sources of fire appear in alternation in $P_\top$ and $P_\perp$, such that no vertex in $V(P_\perp) \sqcup V(P_\top)$ is burned via two sources of fire, and that the vertices in this set burned via the aforementioned sources of fire induce two paths. Moreover, if the first source of fire is $v_{a,1}$ and the next $a-m-1$ sources of fire are further to the right of $v_{a,1}$, then each source of fire among these that burns for $i \geq m$ rounds causes $4i-2m$ vertices to burn ($2i-1$ vertices on one horizontal path, and $2i-2m+1$ vertices on the other). These sources of fire cause a total of

$$k = \sum_{i=m}^{a} (4i-2m) = 2a(a-m+1)$$

vertices in $V(P_\perp) \sqcup V(P_\top)$ to burn. Furthermore, the vertices in $V(P_\perp) \sqcup V(P_\top)$ not burned via those $a-m$ sources of fire constitute at most three paths: a path containing $v_{m,1}$, another containing $v_{m,n}$, and another containing $v_{1,n}$. The orders of these three paths sum up to $2n-k$, and if we consider the sum of these three paths as a spanning subgraph of P_{2n-k}, then by Lemma 3.1.1 (applied with $X = V(P_{2n-k})$), the remaining vertices can be burned with at most $b(P_{2n-k}) + 2 \leq \sqrt{2n-k} + 2$ sources of fire. Thus, if $\sqrt{2n-k}+2 \leq m$, we are done. Indeed, it can be routinely verified (for example, via SageMath) that as long as

$$\begin{aligned} b\big(G_{c\sqrt{n},n}; V(P_\perp) \sqcup V(P_\top)\big) &\leq \frac{1}{2}\left(m + \sqrt{4n - m^2 - 2m + 1} + 1\right) \\ &= \frac{1}{2}\left(m + \sqrt{4n-m^2}\right)\big(1+o(1)\big) \\ &= \left(\frac{c}{2} + \sqrt{1 - \frac{c^2}{4}}\right)\big(1+o(1)\big)\sqrt{n}\,, \end{aligned}$$

we have that $\sqrt{2n-k}+2 \leq m$, and we are done. □

Our final theorem of this chapter gives a refined upper bound on the burning number of a fence $G_{c\sqrt{n},n}$ for small c.

Theorem 3.3.2. If $0 < c \leq 2\sqrt{2}$, then we have that

$$b\big(G_{c\sqrt{n},n}\big) \leq \left(\frac{c}{2} + \sqrt{1 - \frac{c^2}{16}}\right)\big(1 + o(1)\big)\sqrt{n}\,.$$

Proof. Let $P^{(1)}$ and $P^{(2)}$ be the horizontal paths at heights $c\sqrt{n}/4$ and $(3c/4 - 1)\sqrt{n}$, respectively, in $G = G_{c\sqrt{n},n}$. If $H \cong G_{c\sqrt{n}/2,n}$ is the subgraph of G induced by the set of vertices at heights $c\sqrt{n}/4$ to $(3c/4 - 1)\sqrt{n}$ inclusive, then

$$b\big(G_{c\sqrt{n},n}; V(P^{(1)}) \sqcup V(P^{(2)})\big) \leq b\big(H; V(P^{(1)}) \sqcup V(P^{(2)})\big) \leq \left(\frac{c}{4} + \sqrt{1 - \frac{c^2}{16}}\right)\big(1+o(1)\big)\sqrt{n}\,,$$

where the first inequality follows from Lemma 3.1.1, and the second follows from Lemma 3.3.2 with $c/2$ in place of c. Since $\mathrm{dist}_G(v, P^{(i)}) \leq c\sqrt{n}/4 + 1$ for all $v \in V(G)$ and $i \in [2]$, Lemma 3.3.1 implies that

$$b\big(G\big) \leq b\big(G; V(P^{(1)}) \sqcup V(P^{(2)})\big) + \frac{c}{4}\sqrt{n} + 1 \leq \left(\frac{c}{2} + \sqrt{1 - \frac{c^2}{16}}\right)\big(1 + o(1)\big)\sqrt{n}\,,$$

as desired. $\square$

Chapter 4

Background on CSPs and Datalog

We now review some of the literature on CSPs and Datalog. We begin with the famous *CSP Dichotomy Conjecture* of T. Feder and M. Vardi ([32]):

Conjecture 4.0.1. The CSP for any finite template is either in **P** or in **NP-C**.

Progress towards resolving this conjecture includes the use of an algebraic approach (see [14]), where templates are classified via their polymorphisms ([4]). This approach inspired the *Tractability Conjecture* of Bulatov et al. ([14]), which has now been verified (and is presented in Section 4.1 below). Interestingly, the verification is due to the joint results of Bulatov ([16]) and Zhuk ([57]), who give proofs of the CSP Dichotomy Conjecture. Delić ([26]) has provided another proof, using advanced algorithmic methods.

The aforementioned algebraic approach has an important connection with templates of *bounded width*: those finite templates $\mathfrak{B}$ whose CSP can be solved in polynomial time by a so-called local consistency algorithm ([4]). A plain obstruction to such $\mathfrak{B}$ having bounded width is basic relations of $\mathfrak{B}$ encoding linear equations over an additive Abelian group ([32]). The algebraic approach gives a full classification, in terms of descriptive complexity, of templates omitting that obstruction — effectively characterizing templates of bounded width. Other characterizations exist, including $\operatorname{CSP}\mathfrak{B}$ having *bounded treewidth duality* (see [24] and [32]), and $\neg\operatorname{CSP}\mathfrak{B}$ (the complement of $\operatorname{CSP}\mathfrak{B}$) being expressible in Datalog ([32]).

4.1 Results and Conjectures

In light of the foregoing discussion, there is a notable connection between templates of bounded width and congruence $\wedge$-semidistributivity. We have the following theorem:

Theorem 4.1.1 ([3])**.** Let $\mathfrak{B} = (B; \mathcal{S})$ be a finite template. If $\mathrm{V}(B; \operatorname{Pol}\mathfrak{B})$ is congruence $\wedge$-semidistributive, then any instance of $\neg\operatorname{CSP}\mathfrak{B}$ is expressible in Datalog. Hence, the CSPs expressible in Datalog are precisely those parametrized by finite, idempotent algebras $\mathfrak{A}$ for which $\mathrm{V}(\mathfrak{A})$ is congruence $\wedge$-semidistributive.

The results of [34] show that in the context of algebras $\mathfrak{A}$ as considered above, the existence of HM-terms implies the existence of weak Jónsson terms.

The following is the Tractability Conjecture (or *Algebraic CSP Dichotomy Conjecture*), which has now been verified:

Theorem 4.1.2 (Algebraic Dichotomy)**.** Suppose $\mathfrak{B}$ is a finite core template. If the polymorphism algebra of $\mathfrak{B}$ is a Taylor algebra, then $\operatorname{CSP}\mathfrak{B} \in \mathbf{P}$. Otherwise, we have that $\operatorname{CSP}\mathfrak{B} \in \textbf{NP-C}$.

This theorem, as we mentioned before, was originally conjectured by Bulatov et al. ([14]). They had only managed to prove the **NP**-completeness part at the time, leaving the tractability part open.

If $\mathfrak{B} = (B; \mathcal{S})$ is a finite template and $j, k \in \mathbb{N}_0$ with $0 \leq j \leq k$, then $\mathfrak{B}$ has *at-most (j,k)-pathwidth* provided there exist $I_0, I_1, \ldots, I_m \subseteq B$ such that ([4]):

(i) $I_i \cap I_j \subseteq I_l$ for all i, j, and l satisfying $0 \leq i \leq l \leq j \leq m$;

(ii) $|I_t| \leq k$ for all t and $|I_t \cap I_{t+1}| \leq j$ for all $t < m$;

(iii) for each n-ary $R \in \mathcal{S}$ and each $(b_1, \ldots, b_n) \in R^{\mathfrak{B}}$, there is some $t \leq m$ for which $\{b_1, \ldots, b_n\} \subseteq I_t$.

If $\operatorname{CSP} \mathfrak{B}$ has an obstruction set consisting entirely of templates with at-most (j,k)-pathwidth, then $\operatorname{CSP} \mathfrak{B}$ is said to have *(j,k)-pathwidth duality* ([4]). We say that $\operatorname{CSP} \mathfrak{B}$ has *bounded pathwidth duality* if $\operatorname{CSP} \mathfrak{B}$ has (j,k)-pathwidth duality for some j and k. In this case, $\operatorname{CSP} \mathfrak{B} \in \mathbf{NL}$ (see, for example, [22]). It turns out that $\operatorname{CSP} \mathfrak{B}$ has bounded pathwidth duality if and only if $\neg \operatorname{CSP} \mathfrak{B}$ is expressible in linear Datalog. See [22] for further details.

In addition to the aforementioned characterization of bounded pathwidth duality, there is another one in terms of *(j,k)-pebble-relation games* (or *(j,k)-PR games*), where $j, k \in \mathbb{N}_0$ and $0 \leq j \leq k$. These games are played between two players, the Spoiler and the Duplicator, on two templates $\mathfrak{A} = (A; \mathcal{R})$ and $\mathfrak{B} = (B; \mathcal{R})$. As described in [22], a configuration of the game consists of a relation $\mathcal{H} \subseteq \operatorname{Hom}(\mathfrak{A}|_I, \mathfrak{B})$, where $I \subseteq A$ such that $|I| \leq k$. Initially, $I = \varnothing$ and $\mathcal{H} = \{\varnothing\}$. In a given round of the game, the Spoiler places pebbles on the elements of I, and the Duplicator responds with a similar action on B so as to determine $\mathcal{H}$. If $\mathcal{H}_t$ is the configuration after round t, then the Spoiler decides what type of round $\mathcal{H}_{t+1}$ will be: either a *shrinking* round or a *blowing* round. In a shrinking round, the Spoiler sets $I_{t+1} \subseteq I_t$, and the Duplicator responds by setting $\mathcal{H}_{t+1} = \mathcal{H}_t|_{I_{t+1}}$. In a blowing round, the Spoiler sets $I_{t+1} \supseteq I_t$ if and only if $|I_t| \leq j$, in which case the Duplicator responds with $\mathcal{H}_{t+1}|_{I_t} \subseteq \mathcal{H}_t$. The Spoiler wins the game if the Duplicator sets $\mathcal{H}_{t+1} = \varnothing$; otherwise, the game resumes. An algebraic description of (j,k)-PR games is also given in [22]. Based on these games, Barto et al. ([4]) make the following definition:

1. A *solo play of the* (j,k)*-PR game* on finite templates $\mathfrak{A} = (A; \mathcal{R})$ and $\mathfrak{B} = (B; \mathcal{R})$ is a finite sequence $(I_0, I_1, \dots, I_m)$ of subsets of A satisfying two conditions:

   a) $|I_t| \leq k$ for $0 \leq t \leq m$, and

   b) either $I_{t+1} \subseteq I_t$ or $I_t \subset I_{t+1}$ for $0 \leq t < m$, where $|I_t| \leq j$ in the latter case.

2. The *resulting relations* $\mathcal{H}_0, \mathcal{H}_1, \dots, \mathcal{H}_m$ from a solo play $(I_0, I_1, \dots, I_m)$ of the (j,k)-PR game on $\mathfrak{A}$ and $\mathfrak{B}$ are defined recursively as follows:

   a) $\mathcal{H}_0 = \mathrm{Hom}(\mathfrak{A}|_{I_0}, \mathfrak{B})$, and

   b) for $0 \leq t < m$, $\mathcal{H}_{t+1} = \begin{cases} \mathcal{H}_t|_{I_{t+1}}, & I_{t+1} \subseteq I_t \\ \left\{h \in \mathrm{Hom}(\mathfrak{A}|_{I_{t+1}}, \mathfrak{B}) \;\middle|\; h|_{I_t} \in \mathcal{H}_t\right\}, & I_t \subset I_{t+1}. \end{cases}$

Solo plays and resulting relations correspond to plays of the (j,k)-PR game where, for each t, the Spoiler chooses I_t and the Duplicator responds with $\mathcal{H}_t$ at best ([4]).

In particular, for every solo play of the (j,k)-PR game on $\mathfrak{A}$ and $\mathfrak{B}$, the final resulting relation is non-empty (written $\mathfrak{A} \looparrowright_{j,k} \mathfrak{B}$) if and only if the Duplicator has a *strict winning strategy* (see [22]). We thus have the following theorem, which is Proposition 2 in [4]:

Theorem 4.1.3 ([4])**.** Suppose $\mathfrak{A}$ and $\mathfrak{B}$ are finite templates of the same signature. Then CSP $\mathfrak{B}$ has (j,k)-pathwidth duality if and only if $\mathfrak{A} \looparrowright_{j,k} \mathfrak{B}$ implies $\mathfrak{A} \to \mathfrak{B}$.

Barto et al. ([4]) used a corollary of Theorem 4.1.3 to verify a reduction of their main result to an at-most binary case. In Section 5.4, we will proceed in a similar fashion, using a fact equivalent to Theorem 4.1.3 to verify an at-most binary reduction of our own.

Two plain obstructions to bounded pathwidth duality are 3-Horn-Sat and linear equations over an Abelian group ([4]). A reasonable conjecture is that the CSP for any finite template omitting these obstructions has bounded pathwidth duality. The results below indicate significant progress towards resolving this conjecture.

The main result of [1] was proven using the following theorem:

Theorem 4.1.4 ([1])**.** If $\mathfrak{A} = (A; \mathcal{R})$ is a finite template such that $\operatorname{ar}_{\mathcal{R}} R \leq 2$ for each $R \in \mathcal{R}$ and $(A; \operatorname{Pol} \mathfrak{A})$ is congruence distributive, then $\mathfrak{A}$ admits an NU polymorphism.

The following lemma is a modification of one step in the proof of a theorem that reduces CSPs to digraphs (see Theorem 11 in [32]):

Lemma 4.1.1 ([4])**.** Fix $t \in \mathbb{N}$. Suppose $\mathfrak{A} = (A; \mathcal{R})$ is a finite template such that $\operatorname{ar}_{\mathcal{R}} R \leq 2t$ for every $R \in \mathcal{R}$. If $\mathfrak{W} = (A; \operatorname{Pol} \mathfrak{A})$, then there is a template $\mathfrak{A}^{(t)} = (A^t; \mathcal{S})$ with $\operatorname{ar}_{\mathcal{S}} S \leq 2$ for every $S \in \mathcal{S}$ such that:

(i) $\mathfrak{W}^{(t)} = \left(A^t; \operatorname{Pol} \mathfrak{A}^{(t)}\right)$ and,

(ii) for $0 \leq j \leq k$, (j,k)-pathwidth duality of $\operatorname{CSP} \mathfrak{A}^{(t)}$ implies (jt, kt)-pathwidth duality of $\operatorname{CSP} \mathfrak{A}$.

The next lemma generalizes Lemma 2 in [24].

Lemma 4.1.2 ([4])**.** Suppose $\mathfrak{A} = (A; \mathcal{R})$ is a finite template admitting a $(d+1)$-ary NU polymorphism for some $d \geq 2$. Then there is a template $\mathfrak{A}_d = (A; \mathcal{S})$ with $\operatorname{ar}_{\mathcal{S}} S \leq d$ for every $S \in \mathcal{S}$ such that:

(i) $\operatorname{Pol} \mathfrak{A} = \operatorname{Pol} \mathfrak{A}_d$ and,

(ii) (j, k)-pathwidth duality of $\operatorname{CSP} \mathfrak{A}_d$ implies $(k, k + \max(\{\operatorname{ar}_{\mathcal{R}} R\}_{R \in \mathcal{R}} \cup \{d\}) - d)$-pathwidth duality of $\operatorname{CSP} \mathfrak{A}$.

The following proposition was used by Barto et al. ([4]) to prove their main result:

Proposition 4.1.1 ([4])**.** Suppose $\mathfrak{B} = (B; \mathcal{S})$ is a finite template admitting a $(d+1)$-ary NU polymorphism for $d \geq 2$, with $\operatorname{ar}_{\mathcal{S}} S \leq 2$ for every $S \in \mathcal{S}$. Let

$$p = 2\big(\lfloor \log_3(2d - 3) \rfloor + 2\big)^{|B|} - |B| - 1\,.$$

Then $\operatorname{CSP} \mathfrak{B}$ has $(p, p+1)$-pathwidth duality. In case $d = 2$, $\operatorname{CSP} \mathfrak{B}$ has $(2|B|, 2|B|+1)$-pathwidth duality.

The following main result of Barto et al. ([4]), referred to as the d-mapping property by Feder and Vardi ([32]), is the main inspiration for our work in Chapter 5.

Theorem 4.1.5 ([4])**.** If a finite template $\mathfrak{B}$ admits a $(d+1)$-ary NU polymorphism for some $d \geq 2$, then $\operatorname{CSP} \mathfrak{B}$ has bounded pathwidth duality and is thus in **NL**.

Observe that the d-mapping property proves the aforementioned conjecture not just for finite templates admitting NU polymorphisms as hypothesized, but also for Jónsson polymorphisms, by Theorem 4.1.4 and Proposition 4.1.1.

Chapter 5

New Results on CSPs and Datalog

In this chapter, we prove the longstanding Linear Datalog Conjecture. We first give a proof in Section 5.3 for the special case of at-most binary conservative templates. In Section 5.4, we give a proof for the general case.

5.1 Consistency Checks

Let $\mathcal{K}$ be a set of candidate solutions for an instance

$$\mathcal{I} = \Big(V, D, \big\{(x, R_x) \mid x = (x_1, \ldots, x_k) \in V^k\big\}\Big)$$

of a CSP. In solving $\mathcal{I}$, checking for logical consistency would entail the removal of the obstruction set $\{f(x) \in D^k \smallsetminus R_x \mid f \in \mathcal{K}\}$. Suppose $D_l \subseteq D$ is specifically the domain for x_l. For all $i, j \in [k]$, the formula $x_i \to x_j$ is said to be *arc consistent* if, for every $a \in D_i$, there exists $b \in D_j$ such that $(a, b) \in R_{(x_i, x_j)}$. That is, $x_i x_j$ is an arc in the corresponding digraph, with D_i and D_j being acceptable lists of labels (or "colours") for

x_i and x_j, respectively. Special cases of arc consistency for Datalog will be considered in Section 5.4.2. We say that $\mathcal{I}$ is (s,t)-*consistent* if, given $f \in \mathcal{K}$ and any $S \subseteq V$ and $T \subseteq V \smallsetminus S$ with $|S| = s$ and $|T| = t$, consistency of $f[S]$ implies consistency of $g[S \sqcup T]$ for some extension g of f.

Given a template $\mathfrak{A} = (A; \mathcal{R}) \in \operatorname{CSP} \mathfrak{B}$ where $\mathfrak{B} = (B; \mathcal{R})$, we define a *walk on* $\mathfrak{A}$ as a sequence $(\bar{c}_1, \bar{a}_1, \bar{c}_2, \ldots, \bar{c}_n, \bar{a}_n)$ of tuples on A satisfying the following condition: $\bar{c}_i$ contains all coordinates of $\bar{a}_{i-1}$ and $\bar{a}_i$ for all $i \in [n]$ and $\bar{a}_0 = (\,)$. If $\bar{a}_1 = \bar{a}_n = \alpha$, then we say that the walk is *closed with base* α. The *width* of the walk is the maximal length of $\bar{c}_i$ appearing in it. In particular, the walk is *simple* if its width is 2. A *realization* of the walk in $\mathfrak{B}$ is a walk $(\bar{d}_1, \bar{b}_1, \bar{d}_2, \ldots, \bar{d}_n, \bar{b}_n)$ on $\mathfrak{B}$ such that, for each $i \in [n]$, there is a partial homomorphism $h_i \colon C \subseteq A \longrightarrow B$, where C consists of all coordinates in $\bar{c}_i$ such that $\bar{d}_i = h_i(\bar{c}_i)$, $\bar{b}_i = h_i(\bar{a}_i)$, and $\bar{b}_{i-1} = h_i(\bar{a}_{i-1})$.

We make the following claim:

Claim 5.1.1. Let $\mathfrak{B}$ be a finite template, and let $k \in \mathbb{N}$. Then for any $\mathfrak{A} \in \operatorname{CSP} \mathfrak{B}$, the following are equivalent:

(i) Input $\mathfrak{A}$ fails the k-test.

(ii) There is a walk on $\mathfrak{A}$ of width k, with no realizations in $\mathfrak{B}$.

Proof. Let $\mathfrak{A} = (A; \mathcal{R})$ and $\mathfrak{B} = (B; \mathcal{R})$. Given $\mathfrak{A} \in \operatorname{CSP} \mathfrak{B}$ and the k-program for $\mathfrak{B}$, we can generate a list of p.p.-definable subsets P_a of B, indexed by $a \in A$. (One refers to $(A; \{P_a \mid a \in A\})$ as a *list instance*.) Now obtain a system of $k+1$ such lists, in the following way:

a) Let $P_a^{(0)} = P_a$ for each $a \in A$.

b) For each $i \in [k]$ and $a \in A$, let $P_a^{(i)} = R_1^{(i)}(a) \cap \cdots \cap R_s^{(i)}(a)$, where $R_1^{(i)}(a), \ldots, R_s^{(i)}(a)$ are unary facts derived by the i-program on input $(A; \{P_a^{(i-1)} \mid a \in A\})$.

We now look to reduce $\mathfrak{A}$ to a list instance $\mathfrak{A}'$ using a consistency check, in such a way that $\mathfrak{A}$ has a solution if and only if $\mathfrak{A}'$ does. This is a fairly standard technique in the study of parametrized CSPs; we omit the technical details here, and refer the reader to [4]. However, the reduction technique will be fully demonstrated in Section 5.4.2, as it is precisely what we will use to prove the Linear Datalog Conjecture. □

The following lemma is a well-known fact from descriptive complexity:

Lemma 5.1.1. For any $j, k \in \mathbb{N}_0$ and every list instance $\mathfrak{A} = (A; \{P_a \mid a \in A\})$ of CSP $\mathfrak{B}$, there exists $r \in \mathbb{N}_0$ computable from j and k, such that $(A; \{P_a^{(j)} \mid a \in A\})$ passes the k-test whenever $\mathfrak{A}$ passes the r-test.

Notice that Claim 5.1.1 provides an alternative to Datalog for checking whether $\mathfrak{A}$ in the preceding lemma passes the r-test. The usefulness of alternative consistency notions will become especially evident in Section 5.4.

5.2 Linear Datalog

We now state the Linear Datalog Conjecture, which we prove in Section 5.4:

Conjecture 5.2.1 ([46])**.** If $\mathfrak{B}$ is a finite template, then $\mathfrak{B}$ admits a chain of weak Jónsson terms if and only if $\neg\,\mathrm{CSP}\,\mathfrak{B}$ is expressible in linear Datalog.

Upon replacing the word "linear" with the word "symmetric" in the preceding conjecture as stated, we obtain the Symmetric Datalog Conjecture. The latter can also be viewed as the *Space Dichotomy Conjecture*:

Conjecture 5.2.2 ([46])**.** The CSP for any finite, idempotent algebra is either in **L** or **NL-H**.

It was already shown in [46] that if a finite template $\mathfrak{B}$ does not admit a chain of weak Jónsson terms, then $\mathrm{CSP}\,\mathfrak{B} \in \{3\text{-}\textsc{Sat}, 3\text{-}\textsc{Horn}\text{-}\textsc{Sat}, 3\text{-}\textsc{Lin}(p)\}$, and consequently, $\mathrm{CSP}\,\mathfrak{B} \notin \mathbf{NL}$ (which means that $\mathrm{CSP}\,\mathfrak{B}$ does not have bounded pathwidth duality). We now proceed to show that, within the class of at-most binary conservative templates, the presence of weak Jónsson terms implies expressibility of $\neg\,\mathrm{CSP}\,\mathfrak{B}$ in linear Datalog.

5.2.1 Bulatov Colouring of a Taylor Algebra

If $\mathfrak{A} = (A; \mathcal{F})$ is an idempotent, conservative Taylor algebra, then we can assign colours to two-element subuniverses of $\mathfrak{A}$ in the manner set out by A. Bulatov in [15], based on E.L. Post's classification of algebraic clones on a two-element set (see [53]). According to this classification, every idempotent algebra with universe $C = \{a, b\}$ has one of the following three types of operations:

(i.) a *semilattice* operation f, satisfying

$$f(a,a) = a\,,\ f(b,b) = b\,,\text{ and } f(a,b) = f(b,a) = a\,,$$

in which case we refer to a as the "minimal element";

(ii.) a *majority* (that is, ternary NU) operation;

(iii.) a ternary *affine* operation m, which satisfies

$$m(x,x,y) = m(x,y,x) = m(y,x,x) = y$$

for all $x, y \in C$.

Appealing to graph theory, we make a rule that a and b form a directed red edge if C can be equipped with a semilattice operation, with the edge being from b to a whenever a is the minimal element. Otherwise, the edge between a and b is undirected; it is coloured yellow if C can be equipped with a majority operation, and coloured blue if it is not yellow and $\operatorname{Sg}_{\mathfrak{A}} C$ can be equipped with a ternary affine operation. This scheme is sometimes referred to as a *Bulatov colouring*.

If $\mathfrak{A}$ has weak Jónsson terms, then $\mathfrak{A}$ can only have semilattice or majority operations. Furthermore, the choice of these operations can be uniformized; there exists $f \in \operatorname{Clo}_2 \mathfrak{A}$ and $g \in \operatorname{Clo}_3 \mathfrak{A}$ such that f and g agree with semilattice and majority operations on C, respectively. Specifically, f induces the same minimal element when a and b form a red edge, while $f(a, b) = a$ when they form a yellow edge. Additionally, $f(a, f(a, b)) = f(a, b)$ for any $a, b \in C$. As for g, we have that $g(x, y, z) = f(x, f(y, z))$ for all $x, y, z \in C$ when the edge from b to a is red.

5.2.2 Taxonomy of Two-Element Subuniverses

I.G. Rosenberg's classification of functional clones on $\mathbb{B}$ (see also [6]) provides useful insight into the fine structure of finite, Boolean, idempotent algebras. The content of Subsection 5.2.1 reveals that such an algebra having weak Jónsson terms must have either a majority or semilattice operation; the latter being $\wedge$ satisfying

$$0 \wedge 0 = 0 \wedge 1 = 1 \wedge 0 = 0 \quad \text{and} \quad 1 \wedge 1 = 1\,,$$

or $\vee$ satisfying

$$0 \vee 0 = 0 \quad \text{and} \quad 0 \vee 1 = 1 \vee 0 = 1 \vee 1 = 1\,.$$

Furthermore, if an algebra on $\mathbb{B}$ has weak Jónsson terms but no majority operation, then it must have one of the following operations:

(i.) a $(d+1)$-ary NU polymorphism for some $d > 2$; or

(ii.) a Boolean operation g defined by $g(x, y, z) = x \vee (y \wedge z)$ (or, a Boolean operation $\widehat{g}$ defined by $\widehat{g}(x, y, z) = x \wedge (y \vee z)$); or

(iii.) a Boolean operation h defined by $h(x, y, z) = x \vee (y \wedge \neg z)$ (or, a Boolean operation $\widehat{h}$ defined by $\widehat{h}(x, y, z) = x \wedge (y \vee \neg z)$).

Note that if $\mathfrak{A}$ is an idempotent algebra with an NU polymorphism, then every algebra in $\mathrm{V}(\mathfrak{A})$ has Jónsson terms, since $\mathrm{V}(\mathfrak{A})$ is congruence distributive (by Theorem 1.3.3).

The results of [6] also imply the following:

Theorem 5.2.1. If $\mathfrak{A}$ is an idempotent algebra with weak Jónsson terms but with no proper subalgebras, then $\mathrm{V}_{\mathrm{fin}}(\mathfrak{A})$ is congruence distributive.

In fact, when $\mathfrak{A}$ is finite, the preceding theorem can be sharpened as follows:

Corollary 5.2.1. If $\mathfrak{A}$ is a finite, idempotent algebra with weak Jónsson terms but with no proper subalgebras, then $\mathrm{V}(\mathfrak{A}) = \mathrm{V}_{\mathrm{fin}}(\mathfrak{A})$ is congruence distributive; its elements thus have Jónsson terms.

We invite the reader to consult [6] and [38] for the proof of the above corollary.

Our final theorem of this section says that Boolean algebras with weak Jónsson terms and a majority operation are precisely those whose clones are minimal, in the sense of [54].

Theorem 5.2.2 ([38])**.** An algebra $\mathfrak{A} = (\mathbb{B}; \mathcal{F})$ with (weak) Jónsson terms and a minimal functional clone is term equivalent to a Boolean algebra with a majority operation.

We remark that if $\mathfrak{A} = (A; \mathcal{F})$ is a finite, idempotent, conservative algebra with weak Jónsson terms, then for all $a, b \in A$ such that $a \neq b$, both $\{a\}$ and $\{b\}$ are Jónsson ideals of $(\{a, b\}; \mathcal{F}) \leq \mathfrak{A}$. This will be an important observation for the proofs in the next section.

5.3 Linear Datalog Conjecture: Conservative Case

We are now ready to develop our proof of the Linear Datalog Conjecture, for the case of at-most binary conservative templates. Our proof is rooted in universal algebra, and is largely inspired by the techniques of [1].

5.3.1 Local Near-Unanimity Polymorphisms

Suppose $\mathfrak{B} = (B; \mathcal{R})$ is a template where all relation symbols in $\mathcal{R}$ are either unary or binary. For any $n \in \mathbb{N} \setminus \{1\}$, we can define an at-most binary instance

$$P(\mathfrak{B}, n) = (B^n, B, \mathcal{C}), \tag{5.1}$$

where $\mathcal{C}$ consists of the following constraint relations:

$$R^{P(\mathfrak{B},n)}_{((a_1,\ldots,a_n),(b_1,\ldots,b_n))} = \left\{ \big(p(a_1, \ldots, a_n), p(b_1, \ldots, b_n)\big) \;\middle|\; p \in \operatorname{Pol}_n \mathfrak{B} \right\}.$$

Observe that $P(\mathfrak{B}, n)$ is an instance of $\mathrm{CSP}(B; \operatorname{Pol} \mathfrak{B})$. The following is also true:

Proposition 5.3.1 ([1])**.** For every at-most binary template $\mathfrak{B}$ and any $n \in \mathbb{N} \setminus \{1\}$, the set of solutions for $P(\mathfrak{B}, n)$ is $\operatorname{Pol}_n \mathfrak{B}$.

Given a parametrizing algebra $\mathfrak{P}$, we say that an instance $\mathcal{P} = (V, B, \mathcal{C})$ of $\operatorname{CSP} \mathfrak{P}$ is *simple* if, for all $x, y \in V$ where $x \neq y$, there exists $C_{(x,y)} = ((x, y), R_{(x,y)}) \in \mathcal{C}$. This corresponds to a $|V|$-partite graph $G = (V, E)$ with partite sets $\{P_x \mid x \in V\}$, where $|P_x| = |B|$ for each x, and $xy \in E$ if and only if $(x, y) \in R_{(x,y)}$. Solutions for $\mathcal{P}$ thus correspond to cliques of size $|V|$.

Let

$$\mathcal{P} = \Big(V, B, \big\{\big((x_1, x_2), R^{\mathcal{P}}_{(x_1,x_2)}\big)\big\}_{x_1,x_2 \in V}\Big)$$

be a simple at-most binary instance of $\operatorname{CSP} \mathfrak{P}$. If $\mathcal{J} = \{J_x \mid x \in V\} \subseteq \boldsymbol{\mathcal{P}}(B)$, then the *restriction of* $\mathcal{P}$ *to* $\mathcal{J}$ is the simple at-most binary instance

$$\mathcal{P}|_{\mathcal{J}} = \left(V, B, \left\{\left((x_1, x_2), R^{\mathcal{P}|_{\mathcal{J}}}_{(x_1,x_2)}\right)\right\}_{x_1,x_2 \in V}\right)$$

where $R^{\mathcal{P}|_{\mathcal{J}}}_{(x_1,x_2)} = R^{\mathcal{P}}_{(x_1,x_2)} \cap (J_{x_1} \times J_{x_2})$. Henceforth, all simple CSP instances we consider will be at-most binary. This type of consideration is standard, and our results will not depend on more general instances.

Suppose $\mathfrak{B} = (B; \mathcal{R})$ is a template where $|\mathcal{R}| < \infty$, and let $\mathfrak{P} = (B; \mathcal{F})$ be a conservative, idempotent algebra with weak Jónsson terms. The proof of the main result of this subsection will require us to show that for every non-degenerate subset $\{a, b\}$ of B, there exists $n \geq 2$ and $\phi_{a,b} \in \operatorname{Pol}_n \mathfrak{B}$ such that $\phi_{a,b}|_{\{a,b\}^n}$ is an NU operation. To that end, we will use the results of Section 5.2.2. Treating $\{a, b\}$ as we would $\{0, 1\}$, we note that every subalgebra $\mathfrak{U} = (\{a, b\}; \mathcal{F})$ of $\mathfrak{P}$ must have one of three types of polymorphisms: operation 5.2.2(i.), 5.2.2(ii.), or 5.2.2(iii.). To prove the desired result, it suffices to consider just operations 5.2.2(ii.) and 5.2.2(iii.) as possibilities for $\mathfrak{U}$.

(1,2)-systems

Let

$$\mathcal{P} = \Big(V, B, \big\{\big((x_1, x_2), R_{(x_1,x_2)}\big)\big\}_{x_1,x_2 \in V}\Big)$$

be a simple at-most binary instance with $\{P_x \mid x \in V\} \subseteq \boldsymbol{\mathcal{P}}(B)$. We then call $\mathcal{P}$ a *(1,2)-system* if $R_{(x_1,x_2)} \subseteq_{\text{sd}} P_{x_1} \times P_{x_2}$ for all $x_1, x_2 \in V$. In case $\mathfrak{P}$ is an idempotent, conservative algebra with weak Jónsson terms and $\mathcal{P}$ is an instance of CSP $\mathfrak{P}$, we say that $\mathcal{P}$ is a *(1,2)-system over* $\mathfrak{P}$. As an example, (5.1) is a (1,2)-system over $\mathfrak{P}$, with unary constraint relations of the form

$$R_{(a_1,\ldots,a_n)} = \big\{p(a_1, \ldots, a_n) \bigm| p \in \operatorname{Pol}_n \mathfrak{B}\big\} = \big\{a_1, \ldots, a_n\big\} = \operatorname{Sg}_{\mathfrak{P}} \big\{a_1, \ldots, a_n\big\},$$

where the second equality holds since $\mathfrak{P}$ is conservative.

Let

$$\mathcal{P} = \Big(V, B, \big\{\big((x_1, x_2), R_{(x_1,x_2)}\big)\big\}_{x_1,x_2 \in V}\Big)$$

be a simple at-most binary instance. A *$\mathcal{P}$-tree* is a tree T with a labelling (or colouring) map

$\mathrm{X}\colon V(T) \longrightarrow V$. A *realization* r of a $\mathcal{P}$-tree T in $\mathcal{P}$ is a map $r\colon V(T) \longrightarrow B$ such that

$$\big(r(v_1), r(v_2)\big) \in R_{(\mathrm{X}(v_1),\mathrm{X}(v_2))} \text{ for every } v_1v_2 \in E(T)\,.$$

Let $\Gamma_T(\mathcal{P})$ be the set of all realizations of T in $\mathcal{P}$. Now suppose $\mathcal{P}$ is an instance of CSP $\mathfrak{P}$, with $\mathcal{J} = \{J_x \mid x \in V\} \subseteq \boldsymbol{\mathcal{P}}(B)$. If $\mathcal{P}$ is also a (1,2)-system over $\mathfrak{P}$, then it readily follows that every $\mathcal{P}$-tree T is realizable and $\{r(v) \mid r \in \Gamma_T(\mathcal{P})\} = J_{\mathrm{X}(v)}$. Conversely, if every $\mathcal{P}$-tree T is realizable in $\mathcal{P}$ and $\mathcal{T}(\mathcal{P})$ is the set of all $\mathcal{P}$-trees, then

$$J_x = \bigcap_{\substack{T \in \mathcal{T}(\mathcal{P}) \\ v \in V(T) \\ \mathrm{X}(v) = x}} \{r(v) \mid r \in \Gamma_T(\mathcal{P})\} \neq \varnothing$$

for each $x \in V$, and $\mathcal{P}|_{\mathcal{J}}$ is a (1,2)-system over $\mathfrak{P}$. For a proof of this fact, see Proposition 5.3 in [1].

(2,3)-systems

Let

$$\mathcal{P} = \Big(V, B, \big\{\big((x_1, x_2), R_{(x_1,x_2)}\big)\big\}_{x_1,x_2 \in V}\Big)$$

be a (1,2)-system. We then call $\mathcal{P}$ a *(2,3)-system* if, for all $x_1, x_2, x_3 \in V$ and each $(a_1, a_2) \in R_{(x_1,x_2)}$, there exists $a_3 \in B$ such that $(a_1, a_3) \in R_{(x_1,x_3)}$ and $(a_2, a_3) \in R_{(x_2,x_3)}$. Indeed, (5.1) is an example of a (2,3)-system.

If $\mathcal{P} = (V, B, \mathcal{C})$ is a (1,2)-system, then a *pattern* in $\mathcal{P}$ is any tuple

$$x = (x_1, \ldots, x_n) \in V^n;\ n \geq 2\,,$$

such that for each $i \in [n-1]$, there exists $C_{(x_i,x_{i+1})} \in \mathcal{C}$. (This is similar to a walk on a template, as defined in Section 5.1.) Such a pattern x is said to be *closed with base* u if $x_1 = x_n = u$ for $u \in V$. We say that $(b_1, \ldots, b_n) \in B^n$ is a *realization* of the pattern x in $\mathcal{P}$ if (b_i, b_{i+1}) belongs to the constraint relation $R_{(x_i,x_{i+1})}$ for each $i \in [n-1]$. We also say that $a, a' \in B$ are *connected* by x if there is a realization $(a, b_2, \ldots, b_{n-1}, a')$ of the pattern x in $\mathcal{P}$.

The *concatenation* of two patterns $x = (x_1, \ldots, x_m)$ and $x' = (x_m, \ldots, x_{2m-1})$ is

$$x^\frown x' = (x_1, \ldots, x_m, \ldots, x_{2m-1})\,.$$

We denote the k-fold concatenation of x with itself by $x^{\frown k}$.

Let $\mathcal{P} = (V, B, \mathcal{C})$ be a (1,2)-system with $\{P_v \mid v \in V\} \subseteq \boldsymbol{\mathcal{P}}(B)$. We say $\mathcal{P}$ is a *Prague strategy* if, for each $v \in V$, and any two $\mathcal{P}$-patterns $x = (v, x_2, \ldots, x_{n-1}, v)$ and $x' = (v, x_{n+1}, \ldots, x_{2n-2}, v)$ with $\{x_2, \ldots, x_{n-1}\} \subseteq \{x_{n+1}, \ldots, x_{2n-2}\}$, and any $a, a' \in P_v$ connected by x, there exists $k \in \mathbb{N}$ such that a and a' are connected by $x'^{\frown k}$.

The next theorem is an extension of Theorems 5.6 and 5.7 in [1], which were stated under the assumption that the (2,3)-system is parametrized by a congruence distributive algebra; equivalently, one that has Jónsson terms. We relax this condition by requiring only weak Jónsson terms.

Theorem 5.3.1. Let $\mathcal{P} = (V, B, \mathcal{C})$ be a (2,3)-system with $\{P_x \mid x \in V\} \subseteq \boldsymbol{\mathcal{P}}(B)$, over a conservative algebra $\mathfrak{P} = (B; \mathcal{F})$ with weak Jónsson terms. Suppose $\mathcal{J}$ is a family of Jónsson ideals of $(P_x; \mathcal{F}) \leq \mathfrak{P}$.

(1.) If all $\mathcal{P}$-trees with at most $4^{8^{|B|}}$ vertices are realizable in $\mathcal{P}|_{\mathcal{J}}$, then all $\mathcal{P}$-trees are realizable in $\mathcal{P}|_{\mathcal{J}}$.

(2.) If $\mathcal{P}|_{\mathcal{J}}$ is a (1,2)-system, then $\mathcal{P}|_{\mathcal{J}}$ is a Prague strategy, and therefore has a solution (since CSP $\mathfrak{P}$ is of bounded width).

Proof. The proof of part (1.) is based on that of Lemma 6.1 in [1], which uses the properties of Jónsson terms, without the requirement that $p_i(x, y, x) = x$ hold in the parametrizing algebra for $0 < i < n$. That proof is therefore valid under the assumption that $p_0, p_1, \ldots, p_n$ form a sequence of weak Jónsson terms. For part (2.), we simply note that the proof in [2] which shows that $\mathcal{P}|_{\mathcal{J}}$ is a Prague strategy is based on the proof of Lemma 6.7 in [2]. This also uses the properties of Jónsson terms, except for the one that sets Jónsson terms apart from weak Jónsson terms. The proof of the theorem thus follows. □

Existence of Local Near-Unanimity Polymorphisms

We shall return to the simple at-most binary instance $P(\mathfrak{B}, n) = (B^n, B, \mathcal{C})$ which, as we have seen in Proposition 5.3.1, characterizes $\operatorname{Pol}_n \mathfrak{B}$ for all $n \geq 2$. First, we state a variant of Theorem 5.3.1(1.).

Proposition 5.3.2. Let $\mathcal{P} = (V, B, \mathcal{C})$ be a (2,3)-system with $\{Q_x \mid x \in V\} \subseteq \boldsymbol{\mathcal{P}}(B)$, over a conservative algebra $\mathfrak{P} = (B; \mathcal{F})$ with weak Jónsson terms. Suppose ρ_x is a congruence on $\mathfrak{Q}_x = (Q_x; \mathcal{F})$ such that, for each $x \in V$, either $\mathfrak{Q}_x$ is an algebra with Jónsson terms or $\rho_x = Q_x \times Q_x$. Finally, let $\mathcal{J} = \{J_x \mid x \in V\} \subseteq \boldsymbol{\mathcal{P}}(B)$ such that J_x is a ρ_x-block for each $x \in V$. If all $\mathcal{P}$-trees with at most $4^{8^{|B|}}$ vertices are realizable in $\mathcal{P}|_{\mathcal{J}}$, then all $\mathcal{P}$-trees are realizable in $\mathcal{P}|_{\mathcal{J}}$.

Proof. Since J_x is a Jónsson ideal of $\mathfrak{Q}_x$ for each $x \in V$, the proof is a simple adaptation of the proof of Theorem 5.3.1(1.). More specifically, the assumption that the statement of the proposition fails in $\mathcal{P}$ implies the existence of $J_x \leqslant \mathfrak{P}$ for some $x \in V$, such that $J_x \neq Q_x$. This violates the conditions $U, L \subseteq B$, $E, F \leqslant \mathfrak{P}^2$, and $a, b \in B$ of Lemma 6.1 from [1] (apart from our choice of notation $\mathfrak{P}$), since the condition $J_x \neq Q_x$ together with the other aforementioned conditions implies that $\mathfrak{Q}_x \leq \mathfrak{P}$ has Jónsson terms. □

In light of the preceding proposition, given $\mathcal{P} = P(\mathfrak{B}, n)$, consider $n > 4^{8^{|B|}}$. Suppose that the restrictions of $p_0, p_1, \ldots, p_s \in \operatorname{Pol}_3 \mathfrak{B}$ to $\{a, b\}^3$ induce a sequence of Jónsson terms for $(\{a, b\}; \mathcal{F}) \leq (B; \mathcal{F})$. We now explore three key scenarios.

If $x \in B^n$ has only one coordinate b and all remaining coordinates are a, then let ρ_x be $=_{\{a,b\}}$ and let $J_x = \{a\}$. Similarly, if $x \in B^n$ has only one coordinate a and all remaining coordinates are b, then let ρ_x be $=_{\{a,b\}}$ and let $J_x = \{b\}$. Otherwise, for any other $x \in B^n$, let $\rho_x = Q_x \times Q_x$ with $J_x = Q_x$. Now every $\mathcal{P}$-tree T of order at most $n-1$ is realizable in $\mathcal{P}|_{\mathcal{J}}$, where $\mathcal{J} = \{J_x \mid x \in B^n\}$. Since B^n is comprised of n-tuples and $|V(T)| < n$, there exists $i \in [n]$ such that b cannot occur in the i^{th} position of any n-tuple x of the first kind considered above. The i^{th}-coordinate projection of any n-tuple x is a desired realization of T in $\mathcal{P}|_{\mathcal{J}}$. Thus, by Proposition 5.3.2, every $\mathcal{P}$-tree is realizable in $\mathcal{P}|_{\mathcal{J}}$.

Further to the comments above, the discussion at the end of the subsection on (1,2)-systems implies the existence of $\mathcal{J}' = \{J'_x \mid x \in V'\} \subseteq \mathcal{J}$, such that $(V'; \mathcal{J}')$ is a (1,2)-consistent list instance of the CSP considered. Here, V' consists of all $x \in B^n$ from the third scenario above. To show that $\mathcal{Q} = \mathcal{P}|_{\mathcal{J}'}$ has a solution, it is enough to show that $\mathcal{Q}|_{V'}$ (the restriction of $\mathcal{Q}$ to V') has a solution. This is indeed the case, since $|J'_x| = 1$ for each $x \in B^n \setminus V'$ and $\mathcal{Q}|_{V'}$ is a (1,2)-system. However, since a common n-ary operation can be defined on J'_x for each $x \in V'$ (which may be a unary projection), a solution to $\mathcal{Q}$ exists. We have thus established the following result:

Proposition 5.3.3. Suppose $\mathfrak{P} = (B; \mathcal{F})$ is a finitely related, idempotent, conservative algebra with weak Jónsson terms. If $a, b \in B$ where $a \neq b$, then $\mathfrak{P}$ has a $(d_{a,b}+1)$-ary term operation $\phi_{a,b}$ such that $\phi_{a,b}|_{\{a,b\}^{d_{a,b}+1}}$ is a $(d_{a,b}+1)$-ary NU operation, where $d_{a,b} \leq 4^{8^{|B|}}$.

We have in fact given sufficient conditions for the existence of a "local" NU polymorphism. In the next subsection, we consider "global" NU polymorphisms.

5.3.2 Near-Unanimity Polymorphisms: From Local to Global

Our objective now is to show that, under the assumption of $\mathfrak{P}$ being finitely related with weak Jónsson terms as per Proposition 5.3.3, $\mathfrak{P}$ has a stronger universal algebraic property: $\mathfrak{P}$ has an NU polymorphism of a certain computable arity. We have the following proposition:

Proposition 5.3.4. Suppose $\mathfrak{P}$ is a finite, conservative, idempotent algebra, defined by a finite set of at-most binary relation symbols, with weak Jónsson terms. Then $\mathfrak{P}$ has a conservative NU polymorphism.

Proof. Assume that $\mathfrak{P} = (B; \operatorname{Pol}\mathfrak{B})$. By Proposition 5.3.3, for any $a, b \in B$ where $a \neq b$, there exists $\phi_{a,b} \in \operatorname{Pol}_{d_{a,b}+1}\mathfrak{B}$; $d_{a,b} \geq 2$, such that $\phi_{a,b}|_{\{a,b\}^{d_{a,b}+1}}$ is a $(d_{a,b}+1)$-ary NU polymorphism. If

$$B = \{a_0, a_1, \ldots, a_{n-1}\}\,, \tag{5.2}$$

and $\phi_{a,a}$ denotes the unary identity polymorphism on B, then the star composition

$$\phi_a = \phi_{a,a_0} \star \phi_{a,a_1} \star \cdots \star \phi_{a,a_{n-1}}$$

has the following property for all $b \in B$:

$$\phi_a(b, a, \ldots, a) = \phi_a(a, b, a, \ldots, a) = \cdots = \phi_a(a, \ldots, a, b) = a\,. \tag{5.3}$$

This can be checked using the fact that $\phi_{a,b} = \phi_{a,b}|_{\{a,b\}^{d_{a,b}+1}}$ on $\{a,b\}^{d_{a,b}+1}$, and that $\{a,b\} \leqslant \mathfrak{P}$ for all $a, b \in B$.

Now, to construct an NU polymorphism for $\mathfrak{P}$, we use (5.2), and consider the star composition

$$\phi = \phi_{a_0} \star \phi_{a_1} \star \cdots \star \phi_{a_{n-1}} .$$

Using (5.3), it can be readily verified that ϕ is an NU polymorphism. $\square$

The arity of ϕ in the preceding proof can thus be computed from $\{d_{a,b} \mid a, b \in B;\ a \neq b\}$. It follows that $\operatorname{ar}\phi = d + 1$ for sufficiently large d, and that $\operatorname{ar}\phi$ is bounded above by a constant dependent on $|B|$.

Using the d-mapping property (Theorem 4.1.5), we thus obtain a proof of our main result in this section:

Theorem 5.3.2. Suppose $\mathfrak{P} = (B; \mathcal{F})$ is a finite, conservative, idempotent, algebra, defined by a finite set $\mathcal{R}$ of at-most binary relation symbols, with $\mathcal{F} = \operatorname{Pol}(B; \mathcal{R})$. If $\mathfrak{B} = (B; \mathcal{R})$, then the following are equivalent:

(i) $\mathfrak{B}$ admits a chain of weak Jónsson terms;

(ii) $\neg\, \mathrm{CSP}\, \mathfrak{B}$ is expressible in linear Datalog;

(iii) $\mathrm{CSP}\, \mathfrak{B}$ has bounded pathwidth duality.

Hence, among the templates admitting weak Jónsson terms, the conservative templates must have Jónsson terms (by Theorem 1.3.3).

Having proven our main result above, we have provided another proof of the result by Dalmau et al. ([23]), which established the space dichotomy of list homomorphism problems for at-most binary templates (by proving that the Symmetric Datalog Conjecture holds for these problems). The result of Dalmau et al. ([23]) is a corollary of the following theorem of A. Kazda:

Theorem 5.3.3 ([37])**.** Let $\mathfrak{B}$ be a template with an HM-sequence. If $\neg\,\mathrm{CSP}\,\mathfrak{B}$ is expressible in linear Datalog, then $\neg\,\mathrm{CSP}\,\mathfrak{B}$ is also expressible in symmetric Datalog.

Our proof relies almost entirely on the algebraic properties of $\mathfrak{B}$, instead of a finer combinatorial analysis of [23] or its predecessor [30]. This provides an alternative viewpoint, which lends itself to other systems of terms witnessing algebraic properties of the parametrizing algebra.

It is now, in fact, not difficult to adapt our proof to the special case when weak Jónsson terms for $\mathfrak{B}$ are actually HM-terms witnessing congruence n-permutability. We can therefore give a direct algebraic proof of the Symmetric Datalog Conjecture for at-most binary conservative templates. We sketch this proof below.

Let $\mathfrak{B} = (B; \mathcal{R})$. Since the presence of HM-terms implies the existence of (weak) Jónsson terms, using the same argument as before, we can show that there is a $(d+1)$-ary NU polymorphism for some $d \geq 2$ on P_a for each $a \in A$, where $\{P_a \mid a \in A\}$ is the list generated by the canonical symmetric 2-Datalog program for $\mathfrak{B}$, after its run on input $\mathfrak{A} = (A; \mathcal{R})$. If $\mathfrak{U}$ is a simple, idempotent, conservative algebra with HM-terms, then the cover will consist of irreducible, two-element neighbourhoods U inducing polynomially equivalent algebras, for which we have Maltsev polynomials satisfying

$$m_U(x, x, y) = m_U(y, x, x) = y\,,$$

for every U. Additionally, since all such U induce polynomially equivalent algebras, the Maltsev term can be uniformized: there is a unique term operation m acting as a Maltsev polymorphism on all U. After fixing a maximal congruence θ_a on P_a for all $a \in A$, we obtain a list instance $\mathfrak{L} = (A; \{P_a/\theta_a \mid a \in A\})$ for which the relations

$$R_{(a_1,a_2)}/(\theta_{a_1}, \theta_{a_2}) \subseteq P_{a_1}/\theta_{a_1} \times P_{a_2}/\theta_{a_2}$$

are defined as follows: $(b_1/\theta_{a_1}, b_2/\theta_{a_2}) \in R_{(a_1,a_2)}/(\theta_{a_1}, \theta_{a_2})$ if and only if there exists $c_1 \in P_{a_1}$, $c_2 \in P_{a_2}$, $(b_1, c_1) \in \theta_{a_1}$, $(b_2, c_2) \in \theta_{a_2}$, and $(b_1, b_2) \in R_{(a_1,a_2)} \subseteq P_{a_1} \times P_{a_2}$. Since $\mathfrak{B}$ has both Maltsev and NU polymorphisms, the results of [25] imply that $\neg\,\mathrm{CSP}\,\mathfrak{B}$ is expressible in symmetric Datalog.

It follows that $\mathfrak{L}$ will have a solution whenever $\mathfrak{L}$ passes the symmetric r-test for some $r \geq 2$. It can be shown that the reduced list instance $(A; \{\{p_a/\theta_a\} \mid a \in A\})$ will also have a solution. The proof parallels the one given in [4] via the authors' Claims 3 and 4, but uses blocks arising from P_a for each $a \in A$ instead of singletons, as absorbing subuniverses.

5.4 Linear Datalog Conjecture: General Case

Let $\mathfrak{B} = (B; \mathcal{S})$ be an at-most binary template, whose set of basic relations is closed under p.p. definitions. Recall that if $\mathcal{P} = (V, B, \mathcal{C})$ is a (1,2)-system with subdomains $\{J_x \mid x \in V\}$ over a parametrizing algebra $\mathfrak{P}$, then every $\mathcal{P}$-tree T is realizable and $\{r(v) \mid r \in \Gamma_T(\mathcal{P})\} = J_{\mathrm{X}(v)}$, where $\Gamma_T(\mathcal{P})$ is the set of all realizations of T in $\mathcal{P}$, and $\mathrm{X}\colon V(T) \longrightarrow V$ is a labelling map for T. If B_v is a p.p.-definable subset of B for each $v \in V(T)$ and $V(T) \supseteq \{v_1, \ldots, v_n\}$, then the following relations are p.p.-definable over $\mathfrak{B}$:

$$R = \big\{\big(r(v_1), \ldots, r(v_n)\big) \in B^n \;\big|\; r \in \Gamma_T(\mathcal{P});\ r(v) \in B_v\ \forall v \in V(T)\big\}$$

and

$$S = \left\{ b \in B \;\middle|\; (b, \ldots, b) \in R \right\}.$$

The reader can check this; see also Lemma 13 in [4]. For convenience, we will continue to use the notation $\Gamma_T(\mathcal{P})$ for the remainder of this section. We will also let $\mathcal{W}(\mathcal{P})$ denote the set of all $\mathcal{P}$-paths (that is, $\mathcal{P}$-trees with two leaves).

Barto et al. ([4]) recursively define, for fixed $d \in \mathbb{N} \setminus \{1\}$, the following sets of trees:

(i.) $\mathcal{T}_d^{(0)}$, the set of all single-edge trees;

(ii.) for each $i \in \mathbb{N}$, $\mathcal{T}_d^{(i+1)}$, the set of all tree compositions with components from $\mathcal{T}_d^{(i)}$ and at most d leaf components.

Indeed, $\mathcal{T}_d^{(0)} \subseteq \mathcal{T}_d^{(1)} \subseteq \mathcal{T}_d^{(2)} \subseteq \cdots$. For $d = 2$, this simplifies to $\mathcal{T}_2^{(1)} = \mathcal{T}_2^{(2)} = \cdots = \mathcal{L}$, where $\mathcal{L}$ is the set of all paths. This is the case in which our proof of the Linear Datalog Conjecture applies.

5.4.1 Algebraic Results

Let $\mathfrak{A} = (A; \mathcal{F}_A)$ be an algebra such that $\mathcal{F}_A$ is a functional clone on A containing a $(d+1)$-ary NU operation for some $d \geq 2$. If $\mathcal{R}_d = \{R \in \operatorname{Inv} \mathfrak{A} \mid \operatorname{ar} R = d\}$, then $(A; \mathcal{R}_d)$ p.p.-defines $\mathfrak{A}^{\perp}$. If $\mathcal{F}_A$ contains all nullary constant operations and $\operatorname{Inv} \mathfrak{A}$ consists of all compatible diagonal relations, then $\mathfrak{A}^{\perp}$ is p.p.-defined by $(A; \mathcal{B})$, where $\mathcal{B}$ is the set of all binary reflexive compatible relations from $\operatorname{Inv} \mathfrak{A}$.

The following proposition will play an important role in our characterization of finitely related, finite, simple algebras with weak Jónsson terms:

Proposition 5.4.1. Let $\mathfrak{A} = (A; \mathcal{F}_A)$ be an algebra where $\mathcal{F}_A$ is a functional clone on A. If $\mathcal{U} = \{U_1, \ldots, U_k\}$ is a cover of $\mathfrak{A}$ and $\mathfrak{A}|_{U_i}$ has a majority term for all $i \in [k]$, then

$\mathfrak{A}$ also has a majority term.

Proof. For each $i \in [k]$, let m_i be a majority term operation for $\mathfrak{A}|_{U_i}$. Since $\mathcal{U}$ is a cover of $\mathfrak{A}$, by Theorem 1.4.3, there exist $f, e_1, \rho_1, \ldots, e_k, \rho_k \in \operatorname{Clo}\mathfrak{A}$ witnessing this fact. Consider the ternary term

$$M(x, y, z) = f\big(m_1(a_1, b_1, c_1), \ldots, m_k(a_k, b_k, c_k)\big)$$

where $a_i = e_i(\rho_i(x))$, $b_i = e_i(\rho_i(y))$, and $c_i = e_i(\rho_i(z))$ for all $i \in [k]$. According to Theorem 1.4.3(ii), $M(x, y, z) = x$. Combining this with the fact that m_i satisfies

$$m_i(x, x, y) = m_i(x, y, x) = m_i(y, x, x) = x$$

for all $i \in [k]$, we see that

$$M(x, x, y) = M(x, y, x) = M(y, x, x) = x\,.$$

□

We now look to derive a result about simple algebras with weak Jónsson terms. Specifically, we aim to show that such algebras possess a compatible function, assuming that the relational clone contains diagonal relations of all arities.

Chapters 2, 3, and 8 of [34] contain results from *tame congruence theory*, which is a localization theory. We combine several of these results into one theorem (Theorem 5.4.1). While the proof of our unified theorem is highly nontrivial, the interested reader can refer to [34] for all of the individual details.

Theorem 5.4.1. Let $\mathfrak{A} = (A; \operatorname{Pol}\mathfrak{A})$ be a simple algebra such that $\operatorname{Pol}\mathfrak{A}$ contains (weak) Jónsson term operations. If $\mathcal{M}$ is the set of all minimal (under $\subseteq$) neighbourhoods of $\mathfrak{A}$, then the following hold true:

a) Every $U \in \mathcal{M}$ has cardinality 2.

b) For each $a \in A$, there exists $U \in \mathcal{M}$ such that $a \in U$.

c) The system $\mathcal{M}$ *separates points*: for any $a, b \in A$ where $a \neq b$, there exists $U = e[A] \in \mathcal{M}$ such that $e \in \operatorname{Pol}_1 \mathfrak{A}$ and $e(e(a)) = e(a) \neq e(b) = e(e(b))$.

d) For every $U \in \mathcal{M}$, there exists a majority polynomial $m_U(x, y, z)$ satisfying

$$m_U(x, x, y) = m_U(x, y, x) = m_U(y, x, x) = x \quad \text{for all} \quad x, y \in U\,.$$

As a consequence of Theorem 5.4.1, we obtain the following result:

Corollary 5.4.1. If $\mathfrak{A}$ is a finite, simple algebra with weak Jónsson terms, then $\mathfrak{A}$ has a majority polynomial.

Proof. Suppose $\mathcal{R}_A$ is the set of all binary reflexive invariants for $\mathfrak{A}$. Given the set $\mathcal{M}$ of all minimal neighbourhoods of $\mathfrak{A} = (A; \mathcal{R}_A)^{\perp}$, parts b) and c) of Theorem 5.4.1 together imply that $\mathcal{M}$ covers $\mathfrak{A}$. Since $\mathfrak{A}|_U$ has a majority polynomial for each $U \in \mathcal{M}$, Proposition 5.4.1 implies that $\mathfrak{A}$ must also have a majority polynomial. □

5.4.2 Proof of Linear Datalog Conjecture

Let $\mathfrak{A} = (A; \mathcal{R})$ be a finite, at-most binary template admitting weak Jónsson terms. Without loss of generality, we may assume that the basic relations of $\mathfrak{A}$ include every unary and binary relation preserved by the weak Jónsson term operations. If

$$\mathcal{P} = (V, A, \mathcal{C})$$

is a (1,2)-system with $\{P_x \mid x \in V\} \subseteq \boldsymbol{\mathcal{P}}(A)$ over $\mathfrak{V} = (A; \operatorname{Pol}\mathfrak{A})$, then for each $x \in V$, we can define a chain of subsets (or "levels")

$$P_x = P_x^{(0)} \supseteq P_x^{(1)} \supseteq \cdots \tag{5.4}$$

in the following way: for any $T \in \mathcal{W}(\mathcal{P})$ with labelling map $\mathrm{X}\colon V(T) \longrightarrow V$, let

$$\begin{aligned} P_x^{(i+1)}(T, \mathrm{X}) = \Bigg\{ a \in P_x^{(i)} \;\Bigg|\; & \exists r \in \Gamma_T(\mathcal{P});\ r(v) \in P_{\mathrm{X}(v)}^{(i)}\ \forall v \in V(T), \\ & r(v) = a\ \forall v \in \{u \in V(T) \mid \mathrm{X}(u) = x\} \Bigg\}, \end{aligned}$$

and let

$$P_x^{(i+1)} = \bigcap_{T \in \mathcal{W}(\mathcal{P})} P_x^{(i+1)}(T, \mathrm{X}).$$

A Datalog program verifying the *linear arc consistency* (or LAC) of instance $\mathcal{P}$ has one IDB for P_x; $x \in V$, and the rules are as described in [26]. We will appeal to a provably stronger notion, known as *singleton linear arc consistency* (or SLAC).

A fairly recent result of M. Kozik ([44]) shows that all CSPs over templates of bounded width can be solved by the SLAC algorithm (see [26]), while there are CSPs over templates of bounded width that cannot be solved by the LAC algorithm; for example, 3-Horn-Sat. Given this result, we make the following claim regarding (5.4):

Claim 5.4.1 ([4]). For fixed $k \in \mathbb{N}_0$, we must have $P_x^{(k)} \neq \varnothing$ for all $x \in V$.

Thus, for each $x \in V$, we have $A \supseteq P_x^{(0)} \supseteq P_x^{(1)} \supseteq \cdots \supseteq P_x^{(k)} \neq \varnothing$, where $|A| = k$. Consequently, there exists $n_x < k$ such that $P_x^{(n_x)} = P_x^{(n_x+1)}$. In fact, given $x \in V$, we may assume that

$$n_x = \min\left\{s \;\middle|\; P_x^{(s)} = P_x^{(s+1)}\right\}.$$

Let $y \in V$ be such that

$$n_y = \max\left\{n_x \;\middle|\; x \in V\right\}.$$

We may also assume that $n_y \geq 1$. Namely, if $n_x = 0$ for all $x \in V$, then $\mathcal{P}$ is a SLAC-instance (see [26]) and by the result of [44] mentioned above, $\mathcal{P}$ has a solution. Thus, $P_y^{(n)} = P_y^{(n+1)}$ for some $n \geq 1$.

Let α be a maximal congruence on $P_y^{(1)}$ such that $c/\alpha \in P_y^{(n_y)}/\alpha$. Given $\mathcal{P}$ as above, let $\mathcal{T}_L^*(\mathcal{P})$ denote the set of all $\mathcal{P}$-tree compositions (alternatively, *tree patterns*; see [44]) T whose labelling map $\mathrm{X}\colon V(T) \longrightarrow V$ sends all composition vertices of T into $L \subseteq V$. The following result concerning (5.4) is a restatement of Claim 4 in [4]:

Proposition 5.4.2. If $V' = V \smallsetminus \{y\}$, then there exists nonempty $D_{x'} \subseteq P_{x'}^{(n_y)}$ for each $x' \in V'$ such that $D_{x'} \leqslant \mathfrak{V}$ (where $\mathfrak{V}$ is the algebra over which $\mathcal{P}$ is a (1,2)-system). Moreover, for every $T \in \mathcal{T}_L^*(\mathcal{P})$, if $L = \{u \in V(T) \mid \deg_T u = 1,\ \mathrm{X}(u) \in V'\}$, then for each $u \in L$ and each $a \in D_{x'}$, there exists $r \in \Gamma_T(\mathcal{P})$ such that:

(i) $r(u) \in P_y^{(n_y)} \cap (c/\alpha)$ for each $u \in V(T)$ for which $\mathrm{X}(u) = y$;

(ii) $r(v) \in D_{x'}$ for each $v \in L$ for which $\mathrm{X}(v) = x' \in V'$; and

(iii) $r(u) = a$.

A proof by contradiction works well for the preceding result, as demonstrated in [4].

Our strategy for proving the Linear Datalog Conjecture will be to reduce $\mathcal{P}$ to a proper subsystem $\mathcal{Q}$ with subdomains $\{Q_x \mid x \in V\}$, satisfying the following property: if $c \in Q_x \cap P_x^{(n_y+1)}$ for all $x \in V$, then every $\mathcal{P}$-tree T with labelling map X has a realization r such that $r(v) = c$ for every $v \in V(T)$ for which $\mathrm{X}(v) = x$. By restricting to the case of $T \in \mathcal{W}(\mathcal{P})$, we obtain a consistency condition for linear Datalog programs. Iterating the reduction, we will eventually arrive at the trivial system (whose subdomains are singletons), which has the aforementioned property (thus giving a solution for $\mathcal{P}$).

Let $V = \{x_0, x_1, \ldots, x_m\}$ so that $n_{x_0} \geq n_{x_1} \geq \cdots \geq n_{x_m}$. In the first step of the reduction, we replace P_{x_i} with $P_{x_i}^{(2)}$ for every $i \in [m] \sqcup \{0\}$. By definition of a Datalog program, the subinstance $\mathcal{P}^{(2)}$ of $\mathcal{P}$ with subdomains $P_{x_0}^{(2)}, P_{x_1}^{(2)}, \ldots, P_{x_m}^{(2)}$ has a solution if and only if $\mathcal{P}$ has a solution, since restricting domains based on the program rules does not eliminate any elements which appear in some solution for $\mathcal{P}$. Since each constraint relation is either unary or binary, every compatible binary relation on $P_{x_i}^{(2)}$ for each i is defined by a $\mathcal{P}$-path pattern with labelling map X, whose terminal vertices u and v satisfy $\mathrm{X}(u) = \mathrm{X}(v) = x_i$. Now suppose $T \in \mathcal{T}_L^*(\mathcal{P})$ and $\{w_1, \ldots, w_k\}$ is the set of all leaves of T such that $\mathrm{X}(w_1) = \cdots = \mathrm{X}(w_k) = x_i$ for some i. Now for every i, let R be the relation consisting of all tuples $(c_1, \ldots, c_k) \in (P_{x_i}^{(2)})^k$ for which there exists $r \in \Gamma_T(\mathcal{P})$ such that $r(w_j) = c_j$ for all $j \in [k]$. By the preceding commentary and the remarks made at the beginning of this subsection, $R \in \mathrm{Inv}_k(P_{x_i}^{(2)}; \mathrm{Pol}\,\mathfrak{A})$ for all i. Since $\mathrm{Pol}\,\mathfrak{A}$ is a functional clone on A containing all of the constant operations on A, we have that R is diagonal.

A linear Datalog program on input $\mathcal{P}^{(2)}$ obeys a certain consistency condition, as we claim (see also Figure 5.1):

Claim 5.4.2 (L-Property)**.** For each $c \in P_{x_i}^{(l)}$ with $l \in [k] \setminus \{1\}$, and every u,v-path T such that $\mathrm{X}(u) = x_i$ and $\mathrm{X}(v) = x_j$, there exists $r \in \Gamma_T(\mathcal{P})$ such that $r(u) = c$.

Proof. Follows from (5.4) and the definition of canonical linear $|E(T)|$-Datalog program.

□

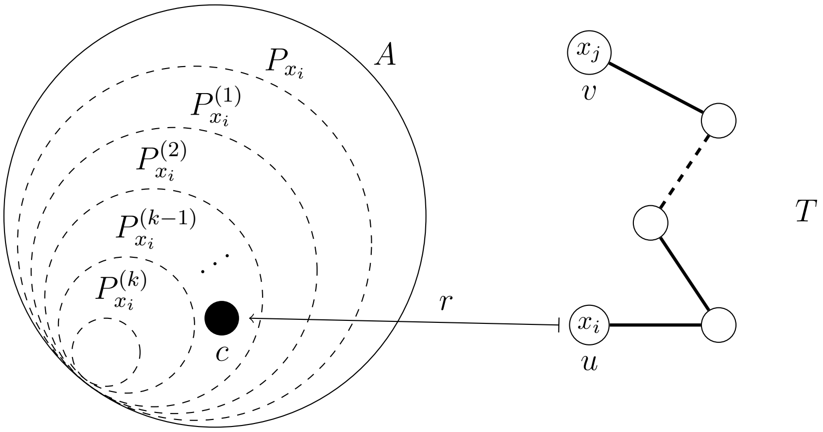


Figure 5.1: A diagram illustrating the L-property (Claim 5.4.2). The "L" not only refers to the levels $P_{x_i}^{(l)}$; $l \in [k] \smallsetminus \{1\}$, but to the linear Datalog program satisfying the property.

Note that by Claim 5.1.1, the conclusion of Claim 5.4.2 is equivalent to $\mathcal{P}$ passing the $|E(T)|$-test.

Let β be a maximal congruence on $(P_{x_0}^{(2)}; \mathcal{F}) \leq \mathfrak{A} = (A; \mathcal{F})$. As $(P_{x_0}^{(2)}/\beta; \mathcal{F})$ is a finite, simple algebra with weak Jónsson terms, this algebra has a majority polynomial operation p by Corollary 5.4.1. Hence, if $y \in P_{x_0}^{(n_{x_0})}$, then $\{y/\beta\} \trianglelefteq_p P_{x_0}^{(2)}/\beta$.

The next theorem is essentially the main device used in proving the existence of a proper subsystem satisfying the L-property. It is a simple consequence of Theorem 7.1 from [44].

Theorem 5.4.2. Let $\mathcal{P}$ be a CSP instance with subdomains $P_{x_0}, P_{x_1}, \ldots, P_{x_m}$, satisfying the L-property. For each $i \in [m] \sqcup \{0\}$, let Q_{x_i} absorb P_{x_i} such that for every $T \in \mathcal{T}_L^*(\mathcal{P})$ with labelling map X, there exists $r \in \Gamma_T(\mathcal{P})$ with the property that $r(u) \in Q_{x_i}$ whenever $\mathrm{X}(u) = x_i$. We then have that the subinstance $\mathcal{Q}$ with subdomains $Q_{x_0}, Q_{x_1}, \ldots, Q_{x_m}$ also satisfies the L-property.

We now proceed to give a formal proof of the fact that our recursive reduction to trivial subinstances is valid. We have the following theorem:

Theorem 5.4.3. Let $\mathcal{P} = (V, A, \mathcal{C})$ be a (1,2)-system over $\mathfrak{A}$ with subdomain P_x for each $x \in V$, and let $\mathcal{P}^{(2)}$ be the subinstance having subdomain $P_x^{(2)}$ for each $x \in V$. Then there is a proper subinstance of $\mathcal{P}^{(2)}$ having the L-property.

Proof. Recall that $\mathcal{P}^{(2)}$ has the L-property. Let $P'_{x_0} = \{y/\beta\}$, where $y \in P_{x_0}^{(n_{x_0})}$ and β is a maximal congruence on $P_{x_0}^{(2)}$. As mentioned earlier, $\{y/\beta\} \trianglelefteq_p P_{x_0}^{(2)}/\beta$, where p is a majority polynomial operation. Therefore, we let $P'_{x_i} = \{y/\beta\}$ if $i = 0$, and we let $P'_{x_i} = P_{x_i}^{(2)}$ if $i > 0$. The existence of p ensures that the clone generated by binary reflexive compatible relations contains all relations definable by the leaves u of $T \in \mathcal{T}_L^*(\mathcal{P})$, such that $\mathrm{X}(u) = x_0$ if $\mathrm{X}\colon V(T) \longrightarrow V$. In particular, any relation defined by some subset of the leaves of T must contain the constant tuple $(y/\beta, \ldots, y/\beta)$. In particular, for any $T \in \mathcal{T}_L^*(\mathcal{P})$ with labelling map X, there exists $r \in \Gamma_T(\mathcal{P})$ such that $r(u) \in y/\beta$ for all $u \in V(T)$ for which $\mathrm{X}(u) = x_0$. Thus, every $T \in \mathcal{T}_L^*(\mathcal{P})$ can be realized in $\mathcal{P}^{(2)}$. By Theorem 5.4.2, the L-property is also satisfied by the proper subinstance of $\mathcal{P}^{(2)}$ having subdomain P'_{x_i} for each $i \in [|V| - 1] \sqcup \{0\}$. □

By taking $V = \{x_0, x_1, \ldots, x_m\}$ so that $n_{x_0} \geq n_{x_1} \geq \cdots \geq n_{x_m}$ (as suggested earlier) and iterating the process in the preceding proof, we can reduce P_{x_i} to a singleton subset of $P_{x_i}^{(n_{x_i})}$ for each $i \in [m] \sqcup \{0\}$, while preserving the L-property. When the L-property holds for some trivial instance $\mathcal{I}$ whose subdomains are $\{a_0\}, \{a_1\}, \ldots, \{a_m\}$, we have that $\mathcal{I}$ yields a solution for $\mathcal{P}^{(2)}$. This implies the existence of a template homomorphism $h\colon (V; \mathcal{R}) \longrightarrow (A; \mathcal{R})$ such that $h(x_i) = a_i$ for each $i \in [m] \sqcup \{0\}$. It follows that $\neg\,\mathrm{CSP}\,\mathfrak{A}$ is expressible in linear Datalog, since the existence of h is due to our consistency condition being met (that is, the L-property being preserved); see [43] for more insight. We have thus shown the following:

Corollary 5.4.2. If $\mathfrak{A}$ is a finite, at-most binary template admitting weak Jónsson terms, then $\neg\,\mathrm{CSP}\,\mathfrak{A}$ is expressible in linear Datalog.

Now, relaxing the assumption that $\mathfrak{A}$ is at-most binary, we invoke Lemma 4.1.1 which assumes $\mathfrak{A}$ is at-most $2t$-ary for $t \in \mathbb{N}$. Since $\mathfrak{A}^{(t)}$ from that lemma satisfies the hypotheses of Corollary 5.4.2 as $t \to \infty$, the Linear Datalog Conjecture (Conjecture 5.2.1) has thus been validated.

Chapter 6

Conclusion and Open Problems

6.1 Summary

We found new bounds on the burning number of fence graphs $G_{c\sqrt{n},n}$. We note that our main result of Chapter 3, Theorem 3.1.2, implies that there exist $k_1, k_2 \in \mathbb{R}$, independent of both n and c, such that

$$k_1 c^{1/3}\sqrt{n} \leq b\big(G_{c\sqrt{n},n}\big) \leq k_2 c^{1/3}\sqrt{n}\,.$$

Our bounds in Theorem 3.1.2 are not asymptotically tight, so it would be interesting to determine the constant in the leading term for the burning number of fences. An adaptation of our methods might provide a means for finding tight bounds.

We have also shown that for every finite at-most binary conservative template $\mathfrak{P}$ admitting weak Jónsson terms, $\operatorname{CSP}\mathfrak{P}$ has bounded pathwidth duality and, consequently, $\neg\operatorname{CSP}\mathfrak{P}$ is expressible in linear Datalog. For that reason, $\operatorname{CSP}\mathfrak{P} \in \mathbf{NL}$. By a result of Larose and Tesson ([46]), all problems in **NL** for which $\mathfrak{P}$ does not admit HM-terms, are, in fact, **NL**-complete. From the perspective of descriptive complexity, this provides a complete classification of CSPs for at-most binary conservative templates of bounded width.

Building on our results for the case of conservative templates, we developed a proof of the Linear Datalog Conjecture for the general case. We showed, by way of consistency checks, that the CSP of any finite core template admitting weak Jónsson terms must have bounded pathwidth duality, thereby placing it in **NL**.

The reader might have noticed that pebble games are not the only connection between graph burning and Datalog. Besides complexity theory, graph theory plays a major part in bridging the two areas of research. Recall, in particular, that burning of a tree T in k rounds is equivalent to a partition of T into k suitable subtrees. Compare this to realizations of $\mathcal{P}$-trees, as seen in our reduction (or consistency checking) method of the last chapter. It is evident that meaningful parallels exist between graph burning and Datalog.

6.2 Open Problems

Our work on CSPs and Datalog has thoroughly addressed space dichotomy, which has been a rich source of open problems. Thus, for the sake of brevity, we reserve further discussion on this, and instead focus on graph burning. Open questions remain from our work on burning grids, and new questions have emerged from variants of ordinary burning. Although we have not had the time to fully explore such variants, we plan to do so in future work.

6.2.1 Burning Grids

Recall our upper bound for $b\big(G_{\sqrt{n},n}\big)$:

$$\frac{2+\sqrt{15}}{4}\big(1+o(1)\big)\sqrt{n} \approx 1.468\sqrt{n}\,.$$

A potential improvement is

$$\frac{3\big(1+\sqrt{29}\big)}{14}\big(1+o(1)\big)\sqrt{n} \approx 1.368\sqrt{n}\,,$$

personally communicated to us by P. Prałat ([52]). A sound method for computing this bound was also described; it relies on closed k-neighbourhoods (or balls), as treated in Chapters 2 and 3 of this dissertation.

An open direction worth exploring is the extension of our results to the setting of strong products. Since $G \,\square\, H$ is a spanning subgraph of $G \boxtimes H$, many of our results extend to strong products of paths, and therefore strong grids. In [50], it was found that for $m \le n$,

$$b\big(G^{\times}_{m,n}\big) = b\big(P_m \boxtimes P_n\big) = \begin{cases} \sqrt[3]{\dfrac{3}{4}}\,\big(1+o(1)\big)\sqrt[3]{mn}\,, & m = \omega\big(\sqrt{n}\big) \\[2em] \Theta\big(\sqrt{n}\big)\,, & m = O\big(\sqrt{n}\big)\,. \end{cases}$$

Another open direction is to improve the bounds on $b(G^{\times}_{m,n})$ for $m = O(\sqrt{n})$. This also begs the question: what is the burning number of the $m \times n$ *toroidal grid*, which is isomorphic to $C_m \,\square\, C_n$? By Theorems 2.1.2 and 2.1.10,

$$b(G_{m,n}) = b(P_m \,\square\, P_n) \ge b(C_m \,\square\, C_n) \ge \max\{b(C_m), b(C_n)\}\,.$$

Such open questions naturally extend to higher dimensional grids as well.

6.2.2 Total Burning

For any graph G, *total graph burning* amounts to burning $T(G)$. Thus, nodes or edges may be burned in G, with fires spreading across edges. We define the *total burning number* of G as $b_{\mathrm{t}}(G) = b(T(G))$; that is, the minimum number of rounds necessary for all elements of G to burn. The figure below illustrates total burning for the case of the 6-wheel, W_6.

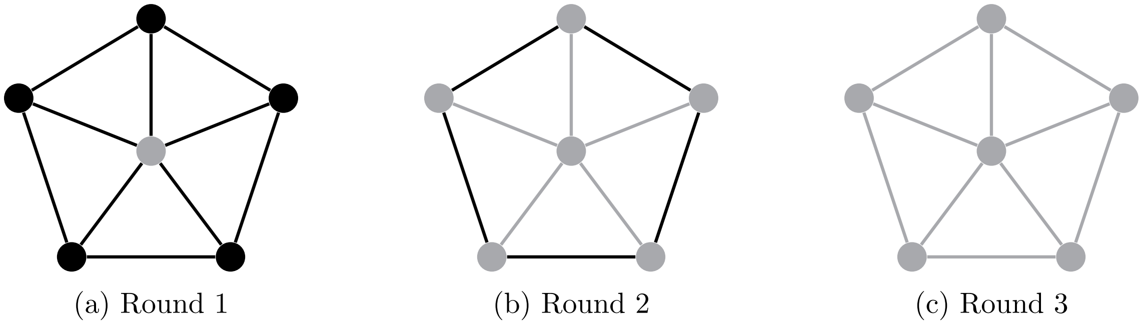

(a) Round 1 (b) Round 2 (c) Round 3

Figure 6.1: Sample total burning of W_6. In Round 1, a fire breaks out at the central node of W_6 (and of $T(W_6)$). In Round 2, the fire spreads to adjacent nodes in W_6 and in $T(W_6)$, thereby burning the edges incident with the central node in W_6. Finally, in Round 3, the remaining elements are burned. Hence, $b_{\mathrm{t}}(W_6) = b(T(W_6)) = 3$.

A reasonable conjecture is that for any undirected graph G, we have that

$$b(G) \leq b_{\mathrm{t}}(G) \leq b(G) + 1\,.$$

However, it would be useful to have a characterization, perhaps analogous to that for rooted tree partitions, of graphs G with $b_{\mathrm{t}}(G) = k$. The behaviour of total burning on induced or isometric subgraphs is also worth considering. Determining the total burning numbers of various graph products, including grids, is also an open problem.

6.2.3 Fast and Slow Burning

Given a graph G and $k \in \mathbb{N}$, suppose that burned nodes of G spread fire to all their k-neighbours. We call this *k-fast burning*, and the associated graph parameter $b_{\mathrm{f}}(G; k)$. For example, $b_{\mathrm{f}}(C_4; 2) = 3$. This reduces to ordinary graph burning when $k = 1$.

Now suppose every burned node of G spreads fire to k neighbours of our choosing. This coincides with ordinary graph burning when $k = \mathit{\Delta}(G)$. We refer to this process as *k-slow burning*, and denote the associated graph parameter by $b_{\mathrm{s}}(G; k)$. Note that $b_{\mathrm{s}}(G; k) \geq b(G)$, as seen in the case of complete graphs G. For example,

$$b_{\mathrm{s}}(K_7; 1) = 4 > 2 = b(K_7)\,.$$

A characterization for graphs G with $b_{\mathrm{f}}(G; k)$ or $b_{\mathrm{s}}(G; k)$ equal to a given natural number would be helpful. Perhaps the k-fast and k-slow burning numbers of planar graphs, trees, and hypercube graphs are also worth investigating.

References

[1] Libor Barto. Finitely related algebras in congruence distributive varieties have near unanimity terms. *Canad. J. Math.*, 65(1):3–21, 2013.

[2] Libor Barto and Marcin Kozik. Congruence distributivity implies bounded width. *SIAM Journal on Computing*, 39(4):1531–1542, 2009.

[3] Libor Barto and Marcin Kozik. Constraint satisfaction problems of bounded width. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2009)*, pages 595–603. IEEE Computer Soc., Los Alamitos, CA, 2009.

[4] Libor Barto, Marcin Kozik, and Ross Willard. Near unanimity constraints have bounded pathwidth duality. In *LICS*, pages 125–134, 2012.

[5] Libor Barto, Andrei Krokhin, and Ross Willard. Polymorphisms, and how to use them. In *Dagstuhl Follow-Ups*, volume 7. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.

[6] Joel Berman. A proof of Lyndon's finite basis theorem. *Discrete Mathematics*, 29(3):229–233, 1980.

[7] Stéphane Bessy, Anthony Bonato, Jeannette Janssen, Dieter Rautenbach, and Elham Roshanbin. Burning a graph is hard. *Discrete Applied Mathematics*, 232:73–87, 2017.

[8] Stéphane Bessy, Anthony Bonato, Jeannette Janssen, Dieter Rautenbach, and Elham Roshanbin. Bounds on the burning number. *Discrete Applied Mathematics*, 235:16–22, 2018.

[9] Norman Biggs, E. Keith Lloyd, and Robin J. Wilson. *Graph Theory, 1736-1936*. Oxford University Press, 1986.

[10] Anthony Bonato, Sean English, Bill Kay, and Daniel Moghbel. Improved bounds for burning fence graphs. *Preprint 2020, arXiv:1911.01342*, 2019.

[11] Anthony Bonato, Jeannette Janssen, and Elham Roshanbin. Burning a graph as a model of social contagion. In *International Workshop on Algorithms and Models for the Web-Graph*, pages 13–22. Springer, 2014.

[12] Anthony Bonato, Jeannette Janssen, and Elham Roshanbin. How to burn a graph. *Internet Mathematics*, 12(1-2):85–100, 2016.

[13] Anthony Bonato and Shahin Kamali. Approximation algorithms for graph burning. In *International Conference on Theory and Applications of Models of Computation*, pages 74–92. Springer, 2019.

[14] Andrei Bulatov, Peter Jeavons, and Andrei Krokhin. Classifying the complexity of constraints using finite algebras. *SIAM J. Comput.*, 34(3):720–742, 2005.

[15] Andrei A. Bulatov. Complexity of conservative constraint satisfaction problems. *ACM Trans. Comput. Log.*, 12(4):Art. 24, 66, 2011.

[16] Andrei A. Bulatov. A dichotomy theorem for nonuniform CSPs. *Preprint 2020, arXiv:1703.03021*, 2017.

[17] Jakub Bulín, Dejan Delić, Marcel Jackson, and Todd Niven. A finer reduction of constraint problems to digraphs. *Logical Methods in Computer Science*, 11(4):1–33, 2015.

[18] Stanley Burris and H.P. Sankappanavar. *A Course in Universal Algebra.* 2006.

[19] Fan R.K. Chung. Pebbling in hypercubes. *SIAM Journal on Discrete Mathematics*, 2(4):467–472, 1989.

[20] Stephen A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, pages 151–158, 1971.

[21] Nadia Creignou, Phokion G. Kolaitis, and Heribert Vollmer, editors. *Complexity of Constraints - An Overview of Current Research Themes [Result of a Dagstuhl Seminar]*, volume 5250 of *Lecture Notes in Computer Science.* Springer, 2008.

[22] Víctor Dalmau. Linear Datalog and bounded path duality of relational structures. *Preprint 2020, cs/0504027*, 2005.

[23] Víctor Dalmau, Lászlő Egri, Pavol Hell, Benoît Larose, and Arash Rafiey. Descriptive complexity of list **H**-coloring problems in logspace: a refined dichotomy. In *Proceedings of the 2015 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 487–498. IEEE Computer Society, 2015.

[24] Víctor Dalmau and Andrei Krokhin. Majority constraints have bounded pathwidth duality. *European Journal of Combinatorics*, 29(4):821–837, 2008.

[25] Víctor Dalmau and Benoît Larose. Maltsev + Datalog $\Longrightarrow$ Symmetric Datalog. In *Proceedings of the 23rd IEEE Symposium on Logic in Computer Science*, LICS '08, pages 297–306, 2008.

[26] Dejan Delić. Constraint satisfaction problem dichotomy for finite templates: a proof via consistency checks. *Preprint 2020, arXiv:1706.03451*, 2017.

[27] Dejan Delić and Daniel Moghbel. Linear Datalog conjecture. *Preprint 2020*, 2020.

[28] Dejan Delić and Daniel Moghbel. Linear Datalog conjecture for conservative templates. *Preprint 2020*, 2020.

[29] George Demetriou. Lexical disambiguation using constraint handling in prolog (chip). In *Sixth Conference of the European Chapter of the Association for Computational Linguistics*, 1993.

[30] László Egri, Pavol Hell, Benoît Larose, and Arash Rafiey. Space complexity of list **H**-colouring: a dichotomy. In *Proceedings of the Twenty-fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 349–365. SIAM, 2014.

[31] Herbert B. Enderton. *A Mathematical Introduction to Logic, 2^{nd} edition.* Elsevier, 2001.

[32] Tomás Feder and Moshe Y. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: a study through Datalog and group theory. *SIAM J. Comput.*, 28(1):57–104 (electronic), 1999.

[33] Lisa Grumbach and Ralph Bergmann. Workflow flexibility by deviation by means of constraint satisfaction problem solving. In *LWDA*, page 212, 2017.

[34] David Hobby and Ralph McKenzie. *The Structure of Finite Algebras*, volume 76 of *Contemporary Mathematics*. American Mathematical Society, Providence, RI, 1988.

[35] Neil Immerman. *Descriptive Complexity.* Springer Science & Business Media, 2012.

[36] Richard M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85–103. Springer, 1972.

[37] Alexandr Kazda. n-permutability and linear Datalog implies symmetric Datalog. *Preprint 2020, arXiv:1508.05766*, 2015.

[38] Keith Kearnes. Idempotent simple algebras. *Lecture Notes in Pure and Applied Mathematics*, pages 529–572, 1996.

[39] Keith A. Kearnes. Tame congruence theory is a localization theory. In *Lecture Notes from "A Course in Tame Congruence Theory" Workshop, Budapest*, 2001.

[40] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 137–146, 2003.

[41] Jon Kleinberg. Cascading behavior in social and economic networks. In *Proceedings of the Fourteenth ACM Conference on Electronic Commerce*, pages 1–4, 2013.

[42] Phokion G. Kolaitis. Constraint satisfaction, databases, and logic. In *IJCAI*, pages 1587–1595, 2003.

[43] Phokion G. Kolaitis and Moshe Y. Vardi. A game-theoretic approach to constraint satisfaction. In *AAAI/IAAI*, pages 175–181, 2000.

[44] Marcin Kozik. Weak consistency notions for all the CSPs of bounded width. In *2016 31st Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–9. IEEE, 2016.

[45] Max R. Land and Linyuan Lu. An upper bound on the burning number of graphs. In *International Workshop on Algorithms and Models for the Web-Graph*, pages 1–8. Springer, 2016.

[46] Benoît Larose and Pascal Tesson. Universal algebra and hardness results for constraint satisfaction problems. *Theoret. Comput. Sci.*, 410(18):1629–1647, 2009.

[47] Leonid A. Levin. Universal sequential search problems. *Problemy Peredachi Informatsii*, 9(3):115–116, 1973.

[48] Toro Mauricio, Camilo Rueda, Carlos Agón, and Gérard Assayag. Gelisp: A framework to represent musical constraint satisfaction problems and search strategies. 2016.

[49] Dieter Mitsche, Paweł Prałat, and Elham Roshanbin. Burning graphs: a probabilistic perspective. *Graphs and Combinatorics*, 33(2):449–471, 2017.

[50] Dieter Mitsche, Paweł Prałat, and Elham Roshanbin. Burning number of graph products. *Theoretical Computer Science*, 746:124–135, 2018.

[51] Lior Pachter, Hunter S. Snevily, and Bill Voxman. On pebbling graphs. *Congressus Numerantium*, 107:65–80, 1995.

[52] Paweł Prałat. Personal communication, 2020.

[53] Emil L. Post. *The Two-Valued Iterative Systems of Mathematical Logic.(AM-5)*, volume 5. Princeton University Press, 2016.

[54] Ivo G. Rosenberg. Minimal clones I: the five types. In *Lectures in Universal Algebra*, pages 405–427. Elsevier, 1986.

[55] Michael Sipser. *Introduction to the Theory of Computation, 3rd edition.* Cengage Learning, 2012.

[56] Douglas B. West. *Introduction to Graph Theory, 2nd edition.* Prentice Hall Upper Saddle River, 2001.

[57] Dmitriy Zhuk. A proof of CSP dichotomy conjecture. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 331–342. IEEE, 2017.