On the Structure of Solution Sets to Regular Word Equations

Joel D. Day[1]
Loughborough University, UK
J.Day@lboro.ac.uk

Florin Manea
Georg-August Universität, Göttingen, Germany
florin.manea@informatik.uni-goettingen.de

Abstract

For quadratic word equations, there exists an algorithm based on rewriting rules which generates a directed graph describing all solutions to the equation. For regular word equations – those for which each variable occurs at most once on each side of the equation – we investigate the properties of this graph, such as bounds on its diameter, size, and DAG-width, as well as providing some insights into symmetries in its structure. As a consequence, we obtain a combinatorial proof that the problem of deciding whether a regular word equation has a solution is in NP.

2012 ACM Subject Classification Mathematics of computing → Combinatorics on words; Theory of computation → Problems, reductions and completeness

Keywords and phrases Quadratic Word Equations, Regular Word Equations, String Solving, NP

Digital Object Identifier 10.4230/LIPIcs.ICALP.2020.124

Category Track B: Automata, Logic, Semantics, and Theory of Programming

Funding Florin Manea's work was supported by the DFG grant MA 5725/2-1.

1 Introduction

A *word equation* is a tuple (α, β), usually written $\alpha \doteq \beta$, such that α and β are words comprised of letters from a *terminal alphabet* $\Sigma = \{\mathtt{a}, \mathtt{b}, \dots\}$ and *variables* from a set $X = \{x, y, z, \dots\}$. Solutions are substitutions of the variables for words in Σ^* making both sides identical. For example, one solution to the word equation $x\mathtt{ab}y \doteq y\mathtt{ba}x$ is given by $x \to \mathtt{b}$ and $y \to \mathtt{bab}$. A system of equations is a set of equations, and a solution to the system is a substitution for the variables which is a solution to all the equations in the system.

One of the most fundamental questions concerning word equations is the satisfiability problem: determining whether or not a word equation has a solution. Makanin [22] famously showed in 1977 that the satisfiability problem for word equations is decidable by giving a general algorithm. Since then, several further algorithms have been presented. Most notable among these are the algorithm given by Plandowski [25] which demonstrated that the satisfiability problem is in PSPACE, the algorithm based on Lempel-Ziv encodings by Plandowksi and Rytter [26], and the method of recompression by Jeż, which has since been shown to require only non-deterministic linear space [15, 16]. On the other hand, it is easily seen that solving word equations is NP-hard due to fact that the subcase when one side of the equation consists only of terminals is exactly the pattern matching problem which is NP-complete [3, 12]. It remains a long-standing open problem whether or not the satisfiability problem for word equations is contained in NP.

[1] Corresponding author

47th International Colloquium on Automata, Languages, and Programming (ICALP 2020).
Editors: Artur Czumaj, Anuj Dawar, and Emanuela Merelli; Article No. 124; pp. 124:1–124:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Recently, there has been elevated interest in solving more general versions of the satisfiability problem, originating from practical applications in e.g. software verification where several *string solving* tools capable of solving word equations are being developed [1, 4, 6, 18, 2] and database theory [14, 13], where one asks whether a given (system of) word equation(s) has a solution which satisfies some additional constraints. Prominent examples include requiring that the substitution for a variable x belongs to some regular language $\mathcal{L}_x$ (regular constraints), or that the lengths of the substitutions of the variables satisfy a set of given linear diophantine equations. Adding regular constraints makes the problem PSPACE complete (see [10, 25, 27], while it is another long standing open problem whether the satisfiability problem with length constraints is decidable. There are also many other kinds of constraints, however many lead to undecidable variants of the satisfiability problem [7, 19]. The main difficulty in dealing with additional constraints is that the solution-sets to word equations are often infinite sets with complex structures. For example, they are not parametrizable [24], and the set of lengths of solutions is generally not definable in Presburger arithmetic [20]. Thus, a better understanding of the solution-sets and their structures is a key aspect of improving our ability to solve problems relating to word equations both in theory and practice.

Quadratic word equations (QWEs) are equations in which each variable occurs at most twice. For QWEs, a conceptually simple and easily implemented algorithm exists which produces a representation of the set of all solutions as a graph. Despite this, however, the satisfiability problem for quadratic equations remains NP-hard, even for severely restricted subclasses [8, 11], while inclusion in NP, and whether the satisfiability problem with length constraints is decidable, have remained open for a long time, just as for the general case.

The algorithm solving QWEs is based on iteratively rewriting the equation(s) according to some simple rules called *Nielsen transformations.* If there exists a sequence of transformations from the original equation to the trivial equation $\varepsilon \doteq \varepsilon$, then the equation has a solution. Otherwise, there is no solution. Hence the satisfiability problem becomes a reachability problem for the underlying rewriting transformation relation, which we denote $\Rightarrow_{NT}$. It is natural to represent this relation as a directed graph $\mathcal{G}^{\Rightarrow_{NT}}$ in which the vertices are word equations and the edges are the rewriting transformations. This has the advantage that the set of all solutions to an equation E corresponds exactly to the set of walks in the graph starting at E and finishing at the trivial equation $\varepsilon \doteq \varepsilon$.[2] Consequently, the properties of the subgraph of $\mathcal{G}^{\Rightarrow_{NT}}$ containing all vertices reachable from E (denoted $\mathcal{G}^{\Rightarrow_{NT}}_{[E]}$) are also informative about the set of solutions to the equation. For example, in [24] a connection is made between the non-parameterisability of the solution set of E and the occurrence of combinations of cycles in the graph. Since equations with a parametrisable solution set are much easier to work with when dealing with additional constraints, this also establishes a connection between the structure of $\mathcal{G}^{\Rightarrow_{NT}}_{[E]}$ and the potential (un)decidability of variants of the satisfiability problem. Moreover, new insights into the structure and symmetries of these graphs are necessary for better understanding and optimising the practical performance of the algorithm.

[2] Each choice of edge in a walk can be seen as a decision about the corresponding solution. It is not necessarily true that different walks will result in different solutions. However, all possible decisions are accounted for, so it is guaranteed that for every solution there is a walk from E to $\varepsilon \doteq \varepsilon$ which corresponds to that solution.

Our Contribution

We consider a subclass of QWEs called regular equations (RWEs) introduced in [23]. A word equation is *regular* if each variable occurs at most once on each side of the equation. Thus, for example, $x\mathtt{ab}y \doteq y\mathtt{ba}x$ is regular while $x\mathtt{ab}x \doteq y\mathtt{ba}y$ is not. Understanding RWEs is a vital step towards understanding the quadratic case, not only because they constitute a significant and general subclass, but also because many non-regular quadratic equations can exhibit the same behaviour as regular ones (consider, e.g. $zz \doteq x\mathtt{ab}yy\mathtt{ba}x$ for which all solutions must satisfy $z = x\mathtt{ab}y = y\mathtt{ba}x$). The satisfiability problem was shown in [8] to be NP-hard for RWEs, and shown to be NP-complete in [9] for some restricted subclasses of RWEs including the classes of regular-reversed and regular-ordered equations.

For RWEs E, we investigate the structure of the graphs $\mathcal{G}^{\Rightarrow_{NT}}_{[E]}$, and as a consequence, are able to describe some of their most important properties. We achieve this by first noting that $\mathcal{G}^{\Rightarrow_{NT}}_{[E]}$ can be divided into strongly connected components $\mathcal{G}^{\Rightarrow}_{[E']}$ for which all the vertices are equations of the same length ($\Rightarrow$ shall be used to denote the restriction of $\Rightarrow_{NT}$ to length preserving transformations only). The "full" graph $\mathcal{G}^{\Rightarrow_{NT}}_{[E]}$ is comprised of these individual components $\mathcal{G}^{\Rightarrow}_{[E']}$ arranged in a DAG-like structure of linear depth (see Section 3) and therefore many properties and parameters of the "full" graph $\mathcal{G}^{\Rightarrow_{NT}}_{[E]}$ are determined by the equivalent properties and parameters of the individual components $\mathcal{G}^{\Rightarrow}_{[E']}$. We then focus on the structure of the subgraphs $\mathcal{G}^{\Rightarrow}_{[E']}$, and as a result are able to give bounds on certain parameters such as diameter, size, and DAG-width.

Our structural results come in two stages, based on whether the equation belongs to a the class of "jumbled" equations introduced in Section 4.3. In the first stage, we consider equations which are not jumbled, and we show that for all such equations E, there exists a jumbled equation $\hat{E}$ such that $\mathcal{G}^{\Rightarrow}_{[E]}$ is comprised mainly of several well-connected near-copies of $\mathcal{G}^{\Rightarrow}_{[\hat{E}]}$. For jumbled equations $\hat{E}$, we show in Section 4.4 that every vertex in $\mathcal{G}^{\Rightarrow}_{[\hat{E}]}$ is close to a vertex in a certain normal form. We show that the vertices in this normal form are determined to a large extent by a property invariant under $\Rightarrow$ introduced in Section 4.2.

With regards to the diameter of $\mathcal{G}^{\Rightarrow}_{[E']}$, we give upper bounds which are polynomial in the length of the equation. It follows that the diameter of the full graph $\mathcal{G}^{\Rightarrow_{NT}}_{[E]}$ is also polynomial, and consequently, that the satisfiability problem for RWEs is NP-complete. This can be generalised to systems of equations satisfying a natural extension of the regularity property (see Section 4.7). We also give exact upper and lower bounds on the number of vertices[3] in $\mathcal{G}^{\Rightarrow}_{[E']}$ for a subclass of RWEs called *basic* RWEs (see Section 4.1), as well as describing exactly for which equations these bounds are achieved. For RWEs which are not basic, we can infer similar bounds, at the cost of a small (linear in the length of the equation) degree of imprecision. Since in the worst case (e.g. for equations without a solution), running the algorithm will perform a full "search" of the graph, the number of vertices is integral to the running time of the algorithm, and is potentially a better indicator of difficult instances than the complexity class alone. An example of this, comes from comparing two subclasses of RWEs called regular-ordered and regular rotated equations. It follows from our results that while both classes have an NP-complete satisfiability problem, if E' is regular-ordered, then $\mathcal{G}^{\Rightarrow}_{[E']}$ will contain at most n vertices, where n is the length of the equation, while if E' is regular rotated, but not regular-ordered, then $\mathcal{G}^{\Rightarrow}_{[E']}$ will contain $\frac{n!}{2}$ vertices, indicating a vast difference in the number of vertices the algorithm would have to visit.

[3] We consider the number of vertices, rather than edges, because it is the number of vertices which is relevant to the performance of the algorithm, and by definition of $\Rightarrow_{NT}$, the out-degree of the graph is bounded by a constant so the the number of edges is linear in the number of vertices.

Motivated by generalisations of the satisfiability problem permitting additional constraints, we also consider the connectivity of the graphs $\mathcal{G}^{\Rightarrow_{NT}}_{[E]}$. To do this, we use DAG-width, a measure for directed graphs which is in several ways analogous to treewidth for undirected graphs. Intuitively, equations for which $\mathcal{G}^{\Rightarrow_{NT}}_{[E]}$ has low DAG-width are likely to be more amenable when considering additional constraints such as length constraints (see Section 3.3). We give an example class of equations for which the DAG-width is unbounded, as well as a class for which the DAG-width is at most two. The latter includes the class of regular-ordered equations which is the most general subclass of QWEs for which it is known that the satisfiability problem with length constraints is decidable [20], and we expect that both cases will be interesting classes to consider in the context of this problem.

2 Preliminaries

For a set S, we denote the cardinality of S by $\text{Card}(S)$. Let Σ be an alphabet. By Σ^*, we denote the set of all words over Σ, and by ε the empty word. By Σ^+, we denote the free semigroup $\Sigma^* \backslash \{\varepsilon\}$. A word u is a prefix (resp. suffix) of a word w if there exists v such that $w = uv$ (resp. $w = vu$). Similarly, u is a factor of w if there exist v, v' such that $w = vuv'$. A prefix/suffix/factor is *proper* if is neither the whole word w, nor ε. The length of a word w is denoted $|w|$, while for $\mathtt{a} \in \Sigma$, $|w|_{\mathtt{a}}$ denotes the number of occurrences of $\mathtt{a}$ in w. For a word $w = \mathtt{a}_1\mathtt{a}_2 \ldots \mathtt{a}_n$ with $\mathtt{a}_i \in \Sigma$ for $1 \leq i \leq n$, the notation $w[i]$ refers to the letter $\mathtt{a}_i$ in the i^{th} position. By w^R, we denote the reversal $\mathtt{a}_n\mathtt{a}_{n-1} \ldots \mathtt{a}_1$ of the word w. Two words w_1, w_2 are conjugate (written $w_1 \sim w_2$) if there exist u, v such that $w_1 = uv$ and $w_2 = vu$.

We shall generally distinguish between two types of alphabet: an infinite set $X = \{x_1, x_2, \ldots\}$ of variables, and a set $\Sigma = \{\mathtt{a}, \mathtt{b}, \ldots\}$ of terminal symbols. We shall assume that $\text{Card}(\Sigma) \geq 2$, and that there exists an order on X leading to a lexicographic order on X^*. For a word $\alpha \in (X \cup \Sigma)^*$, we shall denote by $\text{var}(\alpha)$ the set $\{x \in X \mid x \text{ is a factor of } \alpha\}$. We shall denote by $\text{qv}(\alpha)$ the set $\{x \in \text{var}(\alpha) \mid |\alpha|_x = 2\}$. A word equation is a tuple $(\alpha, \beta) \in (X \cup \Sigma)^* \times (X \cup \Sigma)^*$, usually written $\alpha \doteq \beta$. Solutions are morphisms $h : (X \cup \Sigma)^* \to \Sigma^*$ with $h(\mathtt{a}) = \mathtt{a}$ for all $\mathtt{a} \in \Sigma$ such that $h(\alpha) = h(\beta)$. The satisfiability problem is the problem of deciding algorithmically whether a given word equation has a solution. For equations E given by $\alpha \doteq \beta$, we shall often extend notations regarding words in $(X \cup \Sigma)^*$ to E for convenience, so that, e.g. $|E| = |\alpha\beta|$, $\text{var}(E) = \text{var}(\alpha\beta)$ and $\text{qv}(E) = \text{qv}(\alpha\beta)$. An equation $\alpha \doteq \beta$ is quadratic if $|\alpha\beta|_x \leq 2$ for all $x \in X$. It is regular if $|\alpha|_x \leq 1$ and $|\beta|_x \leq 1$ hold for all $x \in X$. Thus all regular equations are quadratic, but not all quadratic equations are regular. We shall usually abbreviate regular (resp. quadratic) word equation to RWE (resp. QWE). For $Y \subseteq X$, let $\pi_Y : (X \cup \Sigma^*) \to Y^*$ be the morphism such that $\pi_Y(x) = x$ if $x \in Y$ and $\pi_Y(x) = \varepsilon$ otherwise; i.e. π_Y is a projection from $(X \cup \Sigma)^*$ onto Y^*. A regular equation E given by $\alpha \doteq \beta$ is regular-ordered if $\pi_{\text{qv}(E)}(\alpha) = \pi_{\text{qv}(E)}(\beta)$, it is regular rotated if $\pi_{\text{qv}(E)}(\alpha) \sim \pi_{\text{qv}(E)}(\beta)$ and it is regular reversed if $\pi_{\text{qv}(E)}(\alpha) = \pi_{\text{qv}(E)}(\beta)^R$.

Given a set S and binary relation $\mathcal{R} \subseteq S \times S$, we denote the reflexive-transitive closure of $\mathcal{R}$ as $\mathcal{R}^*$. For each $s \in S$, we denote by $[s]_{\mathcal{R}}$ the set $\{s' \mid s\mathcal{R}^*s'\}$. The relation $\mathcal{R}$ may be represented as a directed graph, which we denote $\mathcal{G}^{\mathcal{R}}$, with vertices from S and edges from $\mathcal{R}$. Usually, we will be interested in the subgraph of $\mathcal{G}^{\mathcal{R}}$ containing vertices belonging to $[s]$ for some $s \in S$. Thus, for a subset T of S we shall denote by $\mathcal{G}^{\mathcal{R}}_T$ the subgraph of $\mathcal{G}^{\mathcal{R}}$ containing vertices from T. Given a (directed) graph $\mathcal{G}$, with vertices $V(\mathcal{G})$ and edges $E(\mathcal{G})$, a root vertex is some $v \in V(\mathcal{G})$ such that there does not exist $(u, v) \in E(\mathcal{G})$. We denote by $\text{diam}(\mathcal{G})$ the diameter: the maximum length of a shortest (directed) path between two vertices. For $W, V' \subseteq V(\mathcal{G})$, we say that W *guards* V' if for all $(u, v) \in E(\mathcal{G})$ with $u \in V'$, we have $v \in V' \cup W$. If $\mathcal{G}$ is acyclic, we write $v_1 \leq_{\mathcal{G}} v_2$ if there is a directed path from v_1 to v_2 in $\mathcal{G}$ or $v_1 = v_2$. Following [5], A DAG-decomposition of $\mathcal{G}$ is a pair (D, χ) such that D is a directed acyclic graph (DAG) with vertices $V(D)$, and $\chi = \{X_d \mid d \in V(D)\}$ is a family of subsets of $V(\mathcal{G})$ satisfying:

(D1) $V(\mathcal{G}) = \bigcup\limits_{d \in V(D)} X_d$,

(D2) if $d, d', d'' \in V(D)$ such that $d \leq_D d' \leq_D d''$, then $X_d \cap X_{d''} \subseteq X_{d'}$,

(D3) For all edges (d, d') of D, $X_d \cap X_{d'}$ guards $X_{\geq d'} \backslash X_d$, where $X_{\geq d'} = \bigcup\limits_{d'' \geq_D d'} X_{d''}$, and for all root vertices d, $X_{\geq d}$ is guarded by $\emptyset$.

The width of the DAG-decomposition is $\max\{\mathrm{Card}(X_d) \mid d \in V(D)\}$. The DAG-width of $\mathcal{G}$ is the minimum width of any possible DAG-decomposition of $\mathcal{G}$ and is denoted $\mathrm{dgw}(\mathcal{G})$.

3 Solving regular word equations

In this section we present the algorithm for solving QWEs discussed in the introduction as a rewriting system given by a relation $\Rightarrow_{NT}$. The rewriting transformations are derived from morphisms called Nielsen transformations, and we shall abuse this terminology slightly and generally also refer to the rewriting transformations themselves as Nielsen transformations. The Nielsen transformations never introduce new variables or terminal symbols, and never increase the length of the equation. They also preserve the properties of being quadratic (resp. regular). Thus, given a quadratic (resp. regular) word equation, the possible space of all equations reachable via Nielsen transformations is finite. Moreover, given an equation which has a solution h, there is always at least one Nielsen transformation which produces an equation which has a solution, such that the new equation or the new solution is shorter than the previous one. It follows that, given an equation which possesses a solution, it is possible to reach the equation $\varepsilon \doteq \varepsilon$ after finitely many rewriting steps. For a more detailed description of the algorithm, we refer the reader to e.g. Chapter 12 of [21].

3.1 Nielsen transformations

The Nielsen transformations are defined as follows: for $x \in X \cup \Sigma$ and $y \in X$, let $\psi_{x<y} : (X \cup \Sigma)^* \to (X \cup \Sigma)^*$ be the morphism given by $\psi_{x<y}(y) = xy$ and $\psi_{x<y}(z) = z$ if $z \neq y$. We define the rewriting transformations via the relations $\Rightarrow_L, \Rightarrow_R, \Rightarrow_>$ as follows. Suppose we have a QWE E of the form $x\alpha \doteq y\beta$ where $x, y \in X \cup \Sigma$ and $\alpha, \beta \in (X \cup \Sigma)^*$. Then:

1. if $x \in \mathrm{qv}(E)$ and $x \neq y$, then $x\alpha \doteq y\beta \Rightarrow_L x\psi_{y<x}(\alpha) \doteq \psi_{y<x}(\beta)$, and
2. if $y \in \mathrm{qv}(E)$ and $x \neq y$, then $x\alpha \doteq y\beta \Rightarrow_R \psi_{x<y}(\alpha) \doteq y\psi_{x<y}(\beta)$, and
3. if $x \in X \backslash \mathrm{qv}(E)$, then $x\alpha \doteq y\beta \Rightarrow_> x\alpha \doteq \beta$, and
4. if $y \in X \backslash \mathrm{qv}(E)$, then $x\alpha \doteq y\beta \Rightarrow_> \alpha \doteq y\beta$, and
5. if $x = y$, then $x\alpha \doteq y\beta \Rightarrow_> \alpha \doteq \beta$.

Moreover, for a QWE E of the form $\alpha \doteq \beta$ with $\alpha, \beta \in (X \cup \Sigma)^*$, and for each $Y \subseteq \mathrm{var}(E)$, we have the additional transformations $\alpha \doteq \beta \Rightarrow_> \pi_{X\backslash\{Y\}}(\alpha) \doteq \pi_{X\backslash\{Y\}}(\beta)$. Now, our full rewriting relation, $\Rightarrow_{NT}$, is given by $\Rightarrow_L \cup \Rightarrow_R \cup \Rightarrow_>$. For convenience, we shall define $\Rightarrow$ to be $\Rightarrow_L \cup \Rightarrow_R$. We shall call the rewriting transformations in $\Rightarrow$ *length-preserving*, since they are exactly those for which the resulting equation has the same length as the original.

▶ Remark 1. Let E, E' be QWEs such that $E \Rightarrow_{NT} E'$. If E is regular, then E' is regular. Moreover, if $E \Rightarrow E'$, then $\mathrm{var}(E) = \mathrm{var}(E')$, $\mathrm{qv}(E) = \mathrm{qv}(E')$, and $|E| = |E'|$.

If E_1, E_2 are RWEs such that $E_1 \Rightarrow_L E_2$, then it follows from the definitions that there exist $x, y \in X$ and $\alpha_1, \alpha_2, \beta_1, \beta_2, \in (X\backslash\{x, y\})^*$ such that E_1 is given by $x\alpha_1 y\alpha_2 \doteq y\beta_1 x\beta_2$ and E_2 is given by $x\alpha_1 y\alpha_2 \doteq \beta_1 yx\beta_2$. Extending this observation to multiple applications of

$\Rightarrow_L$, we may conclude that the set $\{E_2 \mid E_1 \Rightarrow_L^* E_2'\}$ is exactly the set $\{x\alpha_1 y\alpha_2 \doteq \beta_3 x\beta_2 \mid \beta_3 \sim y\beta_1\}$. A similar statement can be made for $\Rightarrow_R^*$. Consequently, $\Rightarrow_L^*$ and $\Rightarrow_R^*$ are symmetric. Since they are reflexive and transitive by definition, we get the following.

▶ Remark 2. Let E be a RWE and $Z \in \{L, R\}$. Then $\mathrm{Card}(\{E' \mid E \Rightarrow_Z^* E'\}) < |E|$ and $\Rightarrow_Z^*$ is an equivalence relation. It follows that $\Rightarrow^*$ is also an equivalence relation.

The following well-known result forms the basis for the algorithm for solving QWEs.

▶ **Theorem 3** ([21]). *Let E be a QWE. Then E has a solution if and only if $E \Rightarrow_{NT}^* \varepsilon \doteq \varepsilon$.*

3.2 The graph of all solutions

Theorem 3 provides the basis for treating the satisfiability of QWEs as a reachability problem for the rewriting relation $\Rightarrow_{NT}$. Since any relation R is naturally represented as a (directed) graph $\mathcal{G}^R$, it is also natural to interpret the resulting algorithm as a search in the graph $\mathcal{G}_{[E]}^{\Rightarrow_{NT}}$, in order to determine whether a path exists in the graph from the original equation E to the trivial equation $\varepsilon \doteq \varepsilon$. In fact, the graph $\mathcal{G}_{[E]}^{\Rightarrow_{NT}}$ can tell us significantly more than simply whether a solution to E exists: every walk from E to $\varepsilon \doteq \varepsilon$ in $\mathcal{G}_{[E]}^{\Rightarrow_{NT}}$ corresponds to a solution to E and likewise, every solution to E is represented by a walk in $\mathcal{G}_{[E]}^{\Rightarrow_{NT}}$ from E to $\varepsilon \doteq \varepsilon$. Thus the graphs $\mathcal{G}_{[E]}^{\Rightarrow}$ contain a full description of all solutions to E, and as such, their properties and structure are of inherent interest to the study of QWEs and their solutions. An immediate example of this is the diameter, which is strongly related to the complexity of the satisfiability problem, as demonstrated in the following proposition.

▶ **Proposition 4.** *Let $\mathcal{C}$ be a class of QWEs. Suppose there exists $k \in \mathbb{N}$ such that for each $E \in \mathcal{C}$, we have $\mathrm{diam}(\mathcal{G}_{[E]}^{\Rightarrow_{NT}}) \in O(|E|^k)$. Then the satisfiability problem for $\mathcal{C}$ is in NP.*

Many properties will be determined mostly (i.e. up to some small imprecision) on the subgraphs obtained by restricting our rewriting relation to length-preserving transformations only (i.e. to $\Rightarrow$). Since the rewriting relation $\Rightarrow_{NT}$ allows us to preserve or decrease the length, but never increase it again, any walk in the graph will visit a subgraph containing equations of each length only once, and in order of decreasing length.

The following proposition is an example of how we may infer a global property of $\mathcal{G}_{[E]}^{\Rightarrow_{NT}}$ from its "local" values in the individual subgraphs $\mathcal{G}_{[E']}^{\Rightarrow}$.

▶ **Proposition 5.** *Let E be a QWE. Then*

1. $\mathrm{diam}(\mathcal{G}_{[E]}^{\Rightarrow_{NT}}) \leq 1 + (|E| + 1)\max\{\mathrm{diam}(\mathcal{G}_{[E']}^{\Rightarrow}) \mid E \Rightarrow_{NT}^* E'\}$*, and*
2. $\mathrm{dgw}(\mathcal{G}_{[E]}^{\Rightarrow_{NT}}) = \max\{\mathrm{dgw}(\mathcal{G}_{[E']}^{\Rightarrow}) \mid E \Rightarrow_{NT}^* E'\}$*.*

In what follows, we shall focus predominantly on the structure of the (sub)graphs $\mathcal{G}_{[E']}^{\Rightarrow}$ corresponding to the length-preserving transformations given by $\Rightarrow$. This has the advantage of allowing us to apply further restrictions, including a reduction to the case of basic equations introduced in Section 4.1, without significantly altering the structure of the graph.

3.3 Solving equations modulo constraints

For many kinds of additional constraint, it is possible to adapt the algorithm by finding, for each Nielsen transformation, an appropriate corresponding transformation of the constraints. For example, if $x, y, z \in X$ and we have the length constraint $|x| = |z|$, when we apply the Nielsen transformation associated with $\psi_{y<x}$ to our equation, we replace each occurrence of x with yx. Thus, the updated constraint would be $|x| + |y| = |z|$. However, in some cases, including length constraints, the resulting space of possible combinations of equations and

constraints becomes infinite, meaning the algorithm is no longer guaranteed to terminate. A possible solution to this is to find finite descriptions of the potentially infinite sets of constraints which may occur alongside each equation. The task of finding such descriptions, and consequently the decidability of the corresponding extended satisfiability problems, is dependent on the structural properties of the graph, as can be seen e.g. in [20, 24].

4 Properties of the graphs $\mathcal{G}^{\Rightarrow NT}_{[E]}$ for regular equations E

The remainder of the paper concentrates on describing the structure of the graphs $\mathcal{G}_{[E]}$ for RWEs E. Our general description of $\mathcal{G}_{[E]}$ is comprised of several steps, with each one accounting for a particular aspect. The first step (Section 4.1) describes the effect of terminal symbols, single-occurrence variables, and 'decomposability' on the structure of $\mathcal{G}_{[E]}$, essentially reducing the structure of $\mathcal{G}_{[E]}$ to $\mathcal{G}_{[E']}$ for a "basic" equation E' which does not contain any of these features. The second step (Section 4.3) describes a particular symmetric structure which arises from the same factor(s) occurring on both sides of the equation once we have simplified the equations by eliminating the aforementioned features. This allows for a description of $\mathcal{G}_{[E']}$ as a combination of (near) copies of some smaller graph $\mathcal{G}_{[E'']}$ where E'' is a "jumbled equation" obtained by deleting the appropriate variables from E'. Finally, we are able to show (Section 4.4) that for jumbled equations E'', all vertices in $\mathcal{G}_{[E'']}$ are "close" to a vertex from a small subset conforming to a very particular structure called Lex Normal Form, allowing us to draw conclusions in Sections 4.5 and 4.6 about the diameter, number of vertices and connectivity (DAG-width) of $\mathcal{G}_{[E]}$. Finally, in Section 4.7 we note a generalisation of our results to systems of equations.

4.1 Basic equations: a convenient abstraction

The current section is devoted to reducing the study of the graphs $\mathcal{G}^{\Rightarrow}_{[E]}$ to the case of basic equations. This has several advantages, including a significant reduction in the size of the graphs which is useful for working with examples, as well as allowing for the simpler formulation of precise results, e.g. regarding the size of the graphs in Section 4.6, as well as avoiding unnecessary repetition in the formal statements and their proofs..

▶ **Definition 6** (Basic Equations). *Let E be a QWE given by $\alpha \doteq \beta$. Then E is* decomposable *if there exist proper prefixes α', β' of α and β such that* $\mathrm{var}(\alpha') \cap \mathrm{qv}(E) = \mathrm{var}(\beta') \cap \mathrm{qv}(E)$. *Otherwise, E is* indecomposable. *E is* basic *if it is indecomposable and $\alpha, \beta \in \mathrm{qv}(E)^*$.*

A RWE is basic only if both sides of the equation are permutations of the same set of variables, for example $x_1x_2x_3 \doteq x_3x_1x_2$ and $xywz \doteq wzxy$ are both basic and regular while $xyzw \doteq yxzw$ and $xy \doteq yz$ are not. It is easily verified that the property of being basic is preserved under $\Rightarrow^*$. In order to formally present our reduction from arbitrary RWEs to basic RWEs, we need the following notion for graphs which are structurally similar.

▶ **Definition 7** (Isolated path compression). *Let G_1, G_2 be (directed) graphs. We say that G_1 is an* isolated path compression *of order n of G_2 if G_2 may be obtained from G_1 by replacing each edge (e, e') in G_1 by a path $(e, e_1), (e_1, e_2), \ldots (e_{k-1}, e_k), (e_k, e')$ such that $k \leq n$ and $e_1, e_2, e_3, \ldots, e_k$ are new vertices unique to the edge (e, e').*

Informally, an isolated path compression of a graph is obtained simply by replacing "isolated paths" (paths whose internal vertices are not adjacent to to any vertices outside the path) of a bounded length with single edges. It is easy to see that many structural properties are thus preserved.

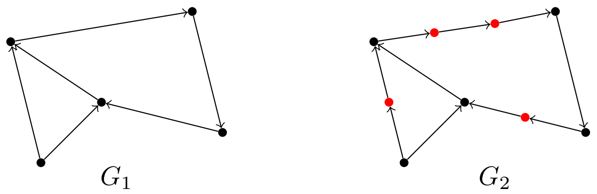

Figure 1 The graph G_1 is an isolated path compression of order two of the graph G_2.

▶ Remark 8. Consider graphs G_1, G_2 such that G_1 is an isolated path compression of order n of G_2. If $\mathrm{dgw}(G_1) = 1$, then $\mathrm{dgw}(G_2) \in \{1, 2\}$. If $\mathrm{dgw}(G_1) \geq 2$, then the $\mathrm{dgw}(G_1) = \mathrm{dgw}(G_2)$. Moreover, $\mathrm{diam}(G_2) \leq (n+1)\,\mathrm{diam}(G_1)$, and the number of vertices (resp. edges) in G_2 is at most the number of vertices in G_1 plus n times the number of edges of G_1.

Using isolated path compressions, it is possible to describe the structure of the graph $\mathcal{G}^{\Rightarrow}_{[E]}$ for any RWE E in terms of the graph $\mathcal{G}^{\Rightarrow}_{[E']}$ for a basic RWE E'.

▶ **Theorem 9.** *Let E be a RWE given by $\alpha \doteq \beta$. Let α', β' be the shortest non-empty prefixes of α, β respectively such that $\mathrm{var}(\alpha') \cap \mathrm{qv}(E) = \mathrm{var}(\beta') \cap \mathrm{qv}(E)$. Let E' be the equation given by $\pi_{\mathrm{qv}(E)}(\alpha') \doteq \pi_{\mathrm{qv}(E)}(\beta')$. Then E' is basic, and $\mathcal{G}^{\Rightarrow}_{[E']}$ is isomorphic to an isolated path compression of order $|E|$ of $\mathcal{G}^{\Rightarrow}_{[E]}$.*

4.2 A useful invariant

When reasoning about the graphs $\mathcal{G}^{\Rightarrow}_{[E]}$, we need a way to help determine whether, for two equations E_1, E_2, we have $E_1 \Rightarrow^* E_2$. Usually, showing that $E_1 \Rightarrow^* E_2$ is not a problem, since it is sufficient to simply find a sequence of length-preserving Nielsen transformations from E_1 to E_2. However, showing that $E_1 \not\Rightarrow^* E_2$ presents more of a challenge. The naive way would be to enumerate all vertices in $\mathcal{G}^{\Rightarrow}_{[E_1]}$ and show that E_2 is not among them. However, this is not suitable for generic reasoning, and, even in concrete cases, is inelegant and time-consuming. The following is a property of basic RWEs which is preserved under $\Rightarrow$ and thus provides a concise and more general means for showing that $E_1 \not\Rightarrow^* E_2$. It is an indispensable component of the proofs of our main results.

▶ **Definition 10** (The invariant Υ_E). *Let $\#$ be a new symbol not in X. Let E be a basic RWE such that $\mathrm{Card}(\mathrm{var}(E)) > 1$. Then we may write E as $x\alpha_1 y\alpha_2 \doteq y\beta_1 x\beta_2$ with $x, y \in X$ and $\alpha_1, \alpha_2, \beta_1, \beta_2 \in (X\backslash\{x,y\})^*$. Let $\mathcal{Z}_E = \mathrm{var}(\alpha_1\alpha_2\beta_1\beta_2) \cup \{\#\}$. Let the function $Q_E : \mathcal{Z}_E \to X^2$ be defined as follows: for each $z \in \mathcal{Z}_E\backslash\{\#\}$, let $Q_E(z) = (u,v)$ where uz is a factor of $x\alpha_1 y\alpha_2$ and vz is a factor of $y\beta_1 x\beta_2$. Let $Q_E(\#) = (u,v)$ where uy is a factor of $x\alpha_1 y\alpha_2$ and vx is a factor of $y\beta_1 x\beta_2$. Let $\Upsilon_E = \{Q_E(z) \mid z \in \mathcal{Z}_E\}$.*

▶ **Theorem 11.** *Let E_1, E_2 be basic RWEs such that $E_1 \Rightarrow^* E_2$. Then $\Upsilon_{E_1} = \Upsilon_{E_2}$.*

As an example, let E_1 be the basic RWE given by $xuzwy \doteq ywuxz$. Then $\mathcal{Z}_{E_1} = \{u, z, w, \#\}$ and Q_{E_1} is the function such that $Q_{E_1}(u) = (x, w)$, $Q_{E_1}(z) = (u, x)$, $Q_{E_1}(w) = (z, y)$ and $Q_{E_1}(\#) = (w, u)$. Thus, $\Upsilon_{E_1} = \{(w, u), (x, w), (u, x), (z, y)\}$. Similarly, if E_2 is the basic RWE given by $xuwzy \doteq yuxwz$, then $\Upsilon_{E_2} = \{(x, y), (u, x), (w, w), (z, u)\}$. Consequently, we may conclude that $E_1 \not\Rightarrow^* E_2$ (and symmetrically $E_2 \not\Rightarrow^* E_1$).

Since the invariant Υ_E provides a necessary condition on when two basic RWEs belong to the same equivalence class under $\Rightarrow^*$, we might also ask whether it is also sufficient, and hence characteristic. However, this is not the case. For instance, if E_1 is given by $xuvwy \doteq ywvux$ and E_2 is given by $xwvuy \doteq yuvwx$, then $\Upsilon_{E_1} = \Upsilon_{E_2} = \{(x, v), (u, w), (v, y), (w, u)\}$ but it can be verified by enumerating $[E_1]_\Rightarrow$ and $[E_2]_\Rightarrow$ that $E_1 \not\Rightarrow^* E_2$.

4.3 A special case of symmetry

The invariant property Υ_E introduced in the previous section is a set of pairs of variables. The case that $(x,x) \in \Upsilon_E$ for some $x \in \mathrm{var}(E)$ is special in the sense that it leads to a particular symmetrical structure in $\mathcal{G}^{\Rightarrow}_{[E]}$. Intuitively, $(x,x) \in \Upsilon_E$ when there exists $y \in X$ and $\alpha \doteq \beta \in [E]_{\Rightarrow}$ such that xy is a factor of both α and β. Hence the number of variables x such that $(x,x) \in \Upsilon_E$ is, in a sense, a measure of the "jumbledness" of E.

▶ **Definition 12** (Jumbled Equations and $\Delta(E)$). *Let E be a basic RWE. Let $\Delta(E) = \{x \in \mathrm{var}(E) \mid (x,x) \in \Upsilon_E\}$. If $\mathrm{Card}(\Delta(E)) = 0$, then E is* jumbled.

Note that since Υ_E is invariant under $\Rightarrow^*$, so is the property of being (not) jumbled. Any basic RWE E can be turned into a jumbled equation by simply erasing each $x \in \Delta(E)$.

▶ **Lemma 13.** *Let E be a basic RWE given by $\alpha \doteq \beta$ and let $Y = \mathrm{var}(E)\backslash\Delta(E)$. Then the equation E_Y given by $\pi_Y(\alpha) \doteq \pi_Y(\beta)$ is jumbled.*

The following theorem describes the structure of $\mathcal{G}^{\Rightarrow}_{[E]}$ for a RWE E which is not jumbled in terms of $\mathcal{G}^{\Rightarrow}_{[E_Y]}$ where E_Y is obtained from E by deleting the variables in $\Delta(E)$.

▶ **Theorem 14.** *Let E be a basic RWE given by $\alpha \doteq \beta$. Let $Y = \mathrm{var}(E)\backslash\Delta(E)$. Let E_Y be the equation $\pi_Y(\alpha) \doteq \pi_Y(\beta)$. Let $V = [E_Y]_{\Rightarrow}$. Let Φ be the set of morphisms $\varphi : Y^* \to \mathrm{var}(E)^*$ satisfying $\varphi(y) \in \Delta(E)^* y$ for all $y \in Y$, and $\sum\limits_{y \in Y} |\varphi(y)|_x = 1$ for all $x \in \Delta(E)$. For each $\varphi \in \Phi$, let $\varphi(V)$ denote the set $\{\varphi(\alpha') \doteq \varphi(\beta') \mid \alpha' \doteq \beta' \in V\}$. Then:*

1. *$\bigcup\limits_{\varphi \in \Phi} \varphi(V) \subseteq [E]_{\Rightarrow}$,*
2. *for each $E' \in [E]_{\Rightarrow}$ and $Z \in \{L,R\}$, there exists $E'' \in \bigcup\limits_{\varphi \in \Phi} \varphi(V)$ such that $E' \Rightarrow^*_Z E''$,*
3. *for each $\varphi \in \Phi$, there exists a subgraph $\mathcal{H}_\varphi$ of $\mathcal{G}^{\Rightarrow}_{[E]}$ containing $\varphi(V)$ such that $\mathcal{G}^{\Rightarrow}_{[E_Y]}$ is isomorphic to a structure-preserving contraction of order* $\mathrm{Card}(\Delta(E))$ *of $\mathcal{H}_\varphi$.*
4. *if $d = \mathrm{diam}(\mathcal{G}^{\Rightarrow}_{[E_Y]})$, then $\mathrm{diam}(\mathcal{G}^{\Rightarrow}_{[E]}) \in O(d|E|^2)$.*

Theorem 14 deserves a few remarks. Firstly, we note that, recalling Remark 2, it follows from statements 1. and 2. of the theorem that $\bigcup\limits_{\varphi \in \Phi} \varphi(V)$ is a dense subset of the vertices of $\mathcal{G}^{\Rightarrow}_{[E]}$ in the sense that every vertex is at most distance $|E|$ away from one contained in $\bigcup\limits_{\varphi \in \Phi} \varphi(V)$. Moreover, since each morphism $\varphi \in \Phi$ is injective, the sets $\varphi(V)$ are pairwise disjoint. Consequently, $\mathcal{G}^{\Rightarrow}_{[E]}$ is made up of many (one for each $\varphi \in \Phi$) slightly modified copies of the graph $\mathcal{G}^{\Rightarrow}_{[\alpha \doteq \beta]}$, with the remaining vertices creating short paths between the different copies. Due to the bound on the diameter, we see that these copies are well connected. Finally, it is worth noting that $\mathrm{Card}(\Phi)$ grows exponentially w.r.t. $\mathrm{Card}(\Delta(E))$.

4.4 Normal forms and block decompositions

Having described the structure $\mathcal{G}^{\Rightarrow}_{[E]}$ for equations E which are not jumbled in the previous section, it remains to consider equations which are jumbled. In this case, the structure of $\mathcal{G}^{\Rightarrow}_{[E]}$ is more intricate and a different approach is required. Our main insight for jumbled equations is the existence of certain normal forms, from which every vertex is polynomial distance away. By constructing these normal forms in a specific way based on reversals, we are able to take full advantage of the invariant Υ_E from Section 4.2 when reasoning about which of these normal forms may occur. The first normal form is defined as follows.

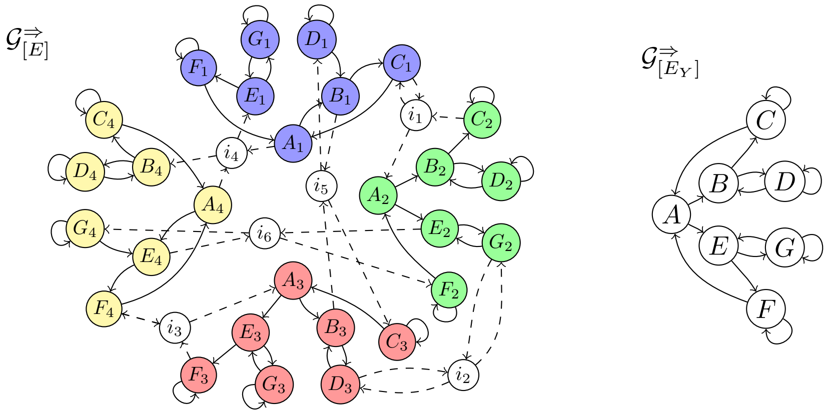

Figure 2 Example illustrating Theorem 14. On the left is $\mathcal{G}^{\Rightarrow}_{[E]}$ for the equation E given by $x_1yx_2x_3x_4 \doteq x_4x_3yx_2x_1$. Note that $\Delta(E) = \{y\}$, so $Y = \{x_1, x_2, x_3, x_4\}$ and E_Y is $x_1x_2x_3x_4 \doteq x_4x_3x_2x_1$. The graph $\mathcal{G}^{\Rightarrow}_{[E_Y]}$ is shown on the right, where the equations in $[E_Y]_{\Rightarrow}$ have been labelled A, B, C, D, E, F, G. The set Φ contains four morphisms φ_i, $1 \leq i \leq 4$, such that $\varphi_i(x_i) = yx_i$ and $\varphi_i(x_j) = x_j$ for $j \neq i$. For each $Z \in \{A, B, C, D, E, F, G\}$ given by $\alpha_Z \doteq \beta_Z$, Z_i denotes the equation $\varphi_i(\alpha_Z) \doteq \varphi_i(\beta_Z)$. The graph $\mathcal{G}^{\Rightarrow}_{[E]}$ contains a "near-copy" of $\mathcal{G}^{\Rightarrow}_{[E_Y]}$ corresponding to each of the morphisms φ_i. Each copy can be made exact by contracting length-two paths (dashed) passing through the intermediate vertices $i_1, i_2, \ldots, i_6$. For example, the subgraph containing the vertices $A_1, B_1, C_1, D_1, E_1, F_1, G_1$ can be made isomorphic to $\mathcal{G}^{\Rightarrow}_{[E_Y]}$ by contracting the paths $(A_1, i_4, E_1), (B_1, i_5, D_1)$, and (C_1, i_1, C_1) into single edges $(A_1, E_1), (B_1, D_1)$ and (C_1, C_1).

▶ **Definition 15** (Normal Form). *Let E be a basic RWE. Then E is in* normal form *if it can be written as $x\alpha_1\alpha_2, \ldots \alpha_k y \doteq y\alpha_1^R\alpha_2^R \ldots \alpha_k^R x$ where $x, y \in X$, $\alpha_i \in X^+$ for $1 \leq i \leq k$, and $|\alpha_i| \leq 3$ for $1 \leq i < k$.*

We can obtain an equation in normal form from any basic RWE by applying a polynomial number of rewriting operations.

▶ **Theorem 16.** *Let E be a jumbled basic RWE. Then there exists $\overline{E}$ which is in normal form and such that $E \Rightarrow^{n_1} \overline{E}$ and $\overline{E} \Rightarrow^{n_2} E$ for some $n_1, n_2 \in O(|E|^3)$.*

The idea behind the first normal form is to divide the RWE into pairs (α_i, α_i^R) which are regular-reversed word equations (although solutions to the full equation E are not necessarily solutions to these smaller equations), and for which all but one belong to a finite number of cases (i.e. three cases depending on the length of α_i). Forcing the sub-equations to be regular-reversed gives us the most control when working with the invariant Υ_E. Some intuition behind this fact can be derived from the observation that if we know that a (complete) basic RWE E is regular-reversed, we can uniquely reconstruct it from the leftmost two variables on the LHS and Υ_E. Indeed, any regular-reversed basic RWE E can be written in the form $x_1x_2 \ldots x_n \doteq x_nx_{n-1} \ldots x_1$, meaning that $\Upsilon_E = \{(x_{i-1}, x_{i+1} \mid 2 \leq i \leq n\} \cup \{(x_{n-1}, x_2)\}$, and if we know x_1, then we may infer from Υ_E all the odd-index variables $(x_3, x_5, \ldots)$ and if we know x_2 then we may infer all the even-index variables $(x_4, x_6, \ldots)$.

Rather than looking at the pairs (α_i, α_i^R) in isolation, in order to take full advantage of the invariant Υ_E, we actually need to consider pairs of the form $(\alpha_i\alpha_{i+1} \ldots \alpha_j, \alpha_i^R\alpha_{i+1}^R \ldots \alpha_j^R)$. We shall call such pairs *blocks*, which we define formally below. Our second normal form will be a restriction of the first, and is based on the notion of blocks.

B_0	B_1	B_2	B_3
x ¦ $z_1\ z_2$	$\mathbf{z_3}\ z_4\ z_5$ ¦ $z_6\ z_7$	$\mathbf{z_8}$ ¦ $z_9\ z_{10}$	z_{11} ¦ $z_{12}\ z_{13}\ z_{14}\ z_{15}$ ¦ y
y ¦ $z_2\ z_1$	$z_5\ z_4\ z_3$ ¦ $z_7\ z_6$	z_8 ¦ $z_{10}\ z_9$	z_{11} ¦ $z_{15}\ z_{14}\ x_{13}\ x_{12}$ ¦ x
Initial (A)	*Standard* (B)	*Standard* (A)	*End* (A)

Figure 3 A depiction of the equation E given by $xz_1z_2z_3z_4z_5z_6z_7z_8z_9z_{10}z_{11}z_{12}z_{13}z_{14}z_{15}y \doteq yz_2z_1z_5z_4z_3z_7z_6z_8z_{10}z_9z_{11}z_{15}z_{14}z_{13}z_{12}x$ where x, y and z_i for $1 \leq i \leq 15$ are variables. The LHS and RHS of the equation are aligned vertically. The block decomposition $\mathfrak{B} = (B_0, B_1, B_2, B_3)$ of E is shown with solid rectangles and with the variety and type of the block written beneath. The additional divisions into the factors α_i, α_i^R required by the definition of normal form are indicated by dashed lines (so that, i.e. $\alpha_1 = z_1z_2$, $\alpha_2 = z_3z_4z_5$, $\alpha_3 = z_6z_7$, $\alpha_4 = z_8, z_5$, $\alpha_5 = z_9z_{10}$, $\alpha_6 = z_{11}$ and $\alpha_7 = z_{12}z_{13}z_{14}z_{15}$). In order for the equation to satisfy the definition of Lex Normal Form, the variables highlighted in bold must be lexicographically minimal with respect to the appropriate sets Γ_i. Note that $\Gamma_1 = \{z_i \mid 3 \leq i \leq 15\} \backslash \{z_4\}$. In particular, Γ_1 consists of the first variable in the block B_1 (x_3) along with (nearly) all variables on the LHS of the equation occurring to the right of z_3, excluding the rightmost variable (y), and since B_1 is Type B, also excluding the second variable in the block B_1 (namely z_4). On the other hand, since B_2 is Type A, in this case we do not need to exclude the second variable in the block B_2, so $\Gamma_2 = \{z_i \mid 8 \leq i \leq 15\}$. Assuming an underlying lexicographic order for which z_{i+1} is greater than z_i, we can conclude that E is in Lex Normal Form.

▶ **Definition 17** (Blocks). *We define 3 variations of blocks which may each have up to two types.*

1. *A* standard block *is a pair* $(\alpha_1\alpha_2 \ldots \alpha_j, \alpha_1^R\alpha_2^R \ldots \alpha_j^R)$ *such that* $j \geq 1$, $\alpha_i \in X^*$ *for* $1 \leq i \leq j$, $|\alpha_1| \in \{1, 3\}$, *and for each* $i, 1 < i \leq j$, $|\alpha_i| = 2$. *It is* Type A *if* $|\alpha_1| = 1$ *and* Type B *if* $|\alpha_1| = 3$.
2. *An* initial block *is a pair* $(x\alpha_1 \ldots \alpha_j, y\alpha_1^R \ldots \alpha_j^R)$ *with* $j \geq 0$, $x, y \in X$ *with* $x \neq y$, *and* $\alpha_i \in (X\backslash\{x, y\})^*$ *for* $1 \leq i \leq j$ *such that* $|\alpha_i| = 2$ *for* $1 \leq i \leq j$. *All initial blocks are* Type A.
3. *An* end block *is a pair* $(\gamma_1\delta y, \gamma_2\delta^R x)$ *where* $x, y \in X$ *with* $x \neq y$, *and* $\gamma_1, \gamma_2, \delta \in (X\backslash\{x, y\})^*$ *with* $|\delta| \geq 1$ *such that* (γ_1, γ_2) *is a block (initial or standard). It is Type A if* (γ_1, γ_2) *is Type A, and Type B otherwise.*

Given an equation which is in normal form, we may decompose it uniquely into blocks in the following manner. The intuition behind this decomposition is that if we fix the invariant property Υ_E, then each block (with the exception of the final block) is determined entirely by the block preceding it and its first (leftmost in the first element) variable. This gives us a crucial degree of control when considering which equations in normal form may appear in $\mathcal{G}^{\Rightarrow}_{[E]}$.

▶ **Definition 18** (Block Decomposition). *Let E be a basic RWE in normal form. Then E may be written as* $x\alpha_1\alpha_2 \ldots \alpha_n y \doteq y\alpha_1^R\alpha_2^R \ldots \alpha_n^R x$ *where* $x, y \in X$, $\alpha_i \in X^+$ *for* $1 \leq i \leq n$, *and* $|\alpha_i| \leq 3$ *for* $1 \leq i < n$. *Let* $I = \{i_1, i_2, \ldots, i_k\} = \{i \mid 1 \leq i < n \text{ and } |\alpha_i| \neq 2\}$ *with* $1 \leq i_1 < i_2 < \ldots < i_k < n$. *If* $I = \emptyset$, *let* $\mathfrak{B} = (E)$. *Otherwise, let* $\mathfrak{B} = (B_0, B_1, \ldots, B_k)$ *where for* $0 \leq j \leq k$, *the* B_j *are blocks such that:*

1. $B_0 = (x\alpha_1 \ldots \alpha_{i_1-1}, y\alpha_1^R \ldots \alpha_{i_1-1}^R)$,
2. $B_k = (\alpha_{i_k} \ldots \alpha_n y, \alpha_{i_k}^R \ldots \alpha_n^R x)$, *and*
3. *for* $1 \leq j < k$, $B_j = (\alpha_{i_j} \ldots \alpha_{i_{j+1}-1}, \alpha_{i_j}^R \ldots \alpha_{i_{j+1}-1}^R)$.

Then $\mathfrak{B}$ *is the* block decomposition *of* E.

An example illustrating a block decomposition of an equation in normal form is given in Figure 3. Since the blocks are fixed by their first variable, it is natural to ask for which variables we can find an equation in our graph $\mathcal{G}^{\Rightarrow}_{[E]}$ such that the block begins with that variable. In particular, can we find an equation in normal form in $\mathcal{G}^{\Rightarrow}_{[E]}$ for which the first variable of each block is lexicographically minimal when going from left to right? The answer to the question is "nearly". In other words, if we relax the notion slightly to account for some specific cases in which we cannot guarantee minimality, then we can always guarantee the existence of such an equation. This leads to the notion of Lex Normal Form defined below.

▶ **Definition 19** (Lex Normal Form). *Let E be a basic RWE in normal form. Then there exist $x, y \in X$ and $\alpha, \beta \in (X \backslash \{x, y\})^*$ such that E has the form $x\alpha y \doteq y\beta x$. Let $(B_0, B_1, \ldots, B_n)$ be the block decomposition of E. For each i, $0 \leq i \leq n$, let $\gamma_i, \gamma'_i \in X^*$ such that $B_i = (\gamma_i, \gamma'_i)$, let $S_i = \{\gamma_i[2], y\}$ whenever B_i is Type B and $S_i = \{y\}$ otherwise, and let $\Gamma_i = \left(\bigcup\limits_{i \leq j \leq n} \mathrm{var}(\gamma_j) \right) \backslash S_i$. A block B_i is* lex-minimal *if $\gamma_i[1]$ is lexicographically minimal in Γ_i. The equation E is in Lex Normal Form (LNF) if, for each i, $0 < i < n$, B_i is lex-minimal.*

Lex Normal Form (see also Fig. 3 for an example) describes the class of equations for which the first variable of each blocks is lexicographically minimal *whenever possible*. We can, in general, guarantee the existence of an equation E' in $\mathcal{G}^{\Rightarrow}_{[E]}$ such that the first variable of each block is lexicographically minimal with the following exceptions. Firstly, we must exclude the first and last blocks (the first block is fixed completely by Υ_E). Secondly, we must only compare the first variable to other variables occurring further right in the LHS of the equation, and excluding the rightmost variable on the LHS of the equation (y in the definition above) and, for blocks of Type B, the second variable in the block. The sets Γ_i in the definition account for these exclusions. It turns out that every vertex in $\mathcal{G}^{\Rightarrow}_{[E]}$ is never more than a polynomial distance away from a vertex corresponding to an equation in LNF.

▶ **Theorem 20.** *Let E be a jumbled basic RWE. Then there exists E' such that E' is in Lex Normal Form, and such that $E \Rightarrow^{n_1} E'$ and $E \;\Rightarrow^{n_2} E$ for some $n_1, n_2 \in O(|E|^4)$.*

4.5 Diameter

It was mentioned in the previous section that the choices for the blocks in a block decomposition of an equation in normal form are restricted by the invariant Υ_E. We shall now make full use of that fact to show that the number of equations in LNF in a single graph $\mathcal{G}^{\Rightarrow}_{[E]}$ is polynomial in $|E|$, and as a consequence that the diameter of $\mathcal{G}^{\Rightarrow}_{[E]}$ is also polynomial. Since each equation in LNF has a unique block decomposition, it is sufficient to count the possible block decompositions for a given value of Υ_E for which the conditions for LNF hold. The restrictions imposed on the blocks by Υ_E are given formally in the following lemmata.

▶ **Lemma 21.** *Let E_1, E_2 be basic RWEs in normal form such that $\Upsilon_{E_1} = \Upsilon_{E_2}$. Let $(B_0, B_1, \ldots, B_k)$ and $(C_0, C_1, \ldots, C_\ell)$ be their respective block decompositions and let $k, \ell > 0$. Then $B_0 = C_0$. Moreover, suppose that $B_i = C_j$, for some $i < k - 1$, $j < \ell - 1$. Let $B_{i+1} = (\gamma_1, \gamma_2)$ and $C_{j+1} = (\delta_1, \delta_2)$ with $\gamma_1, \gamma_2, \delta_1, \delta_2 \in X^*$. If $\gamma_1[1] = \delta_1[1]$, then $B_{i+1} = C_{j+1}$.*

Lemma 21 tells us that the equations in LNF belonging to a single graph $\mathcal{G}^{\Rightarrow}_{[E]}$ are remarkably similar in that they are identical up to the last block of the shorter decomposition.

▶ **Corollary 22.** *Let E_1, E_2 be basic RWEs in LNF such that $\Upsilon_{E_1} = \Upsilon_{E_2}$. Let $(B_0, B_1, \ldots, B_k)$ and $(C_0, C_1, \ldots, C_\ell)$ be their respective block decompositions and suppose that $k, \ell > 0$. Then $B_i = C_i$ for $0 \leq i < \min(k, \ell)$.*

Consequently, two equations in LNF in the graph $\mathcal{G}^{\Rightarrow}_{[E]}$ with block decompositions containing the same number of blocks may differ only in the final block. Clearly, the number of blocks is at most $\mathrm{Card}(\mathrm{var}(E))$. Thus, in order to show that there are only polynomially many equations in LNF in $\mathcal{G}^{\Rightarrow}_{[E]}$, it remains to consider the possibilities for the final block.

▶ **Lemma 23.** *Let E_1, E_2 be basic RWEs in normal form such that $\Upsilon_{E_1} = \Upsilon_{E_2}$. Let $(B_0, B_1, \ldots, B_k)$ and $(C_0, C_1, \ldots, C_\ell)$ be their respective block decompositions and suppose that $k, \ell > 0$. Suppose moreover that $B_{k-1} = C_{\ell-1}$. Let $B_k = (\alpha_1\alpha_2 \ldots \alpha_n y, \alpha_1^R\alpha_2^R \ldots \alpha_n^R x)$ and $C_\ell = (\beta_1\beta_2 \ldots \beta_m y, \beta_1^R\beta_2^R \ldots, \beta_m^R x)$, where $x, y \in X$, $\alpha_1, \alpha_2, \ldots, \alpha_n$, $\beta_1, \beta_2, \ldots, \beta_m \in X^+$, $|\alpha_1| = |\beta_1| \in \{1, 3\}$ and $|\alpha_i|, |\beta_j| = 2$ for $2 \leq i < n$ and $2 \leq j < m$. Then if $\alpha_1[1] = \beta_1[1]$, $n = m$, and $\alpha_n[1] = \beta_m[1]$, we have $B_k = C_\ell$.*

Lemma 23 reveals that the options for last block are dependent only on the choices of three parameters: $\alpha_1[1], \alpha_n[1]$, and n. Since each of these can take at most $|E|$ possible values, there are $|E|^3$ possibilities altogether. Thus for each possible number of blocks, there are at most $|E|^3$ possible block decompositions, and therefore only $|E|^4$ possible block decompositions respecting the invariant Υ_E in total. Since every equation in LNF permits a unique block decomposition, this gives us our desired polynomial bound.

▶ **Theorem 24.** *Let E be a basic RWE. Let S be the set of basic regular equations E' in Lex Normal Form for which $\Upsilon_E = \Upsilon_{E'}$. Then* $\mathrm{Card}(S) \leq |E|^4$.

Since every vertex in $\mathcal{G}^{\Rightarrow}_{[E]}$ is at polynomial distance from a vertex in LNF, and since there are only polynomially many such vertices, it is straightforward to show that the diameter of $\mathcal{G}^{\Rightarrow}_{[E]}$ must also be polynomial: indeed if we have a sufficiently long path between two vertices, then we must have a long path between two vertices which are close to the same vertex in LNF. Since they are close to the same vertex, we can find a shortcut between them, and the initial long path is not minimal. Since the diameter of $\mathcal{G}^{\Rightarrow}_{[E]}$ is polynomial, it follows from Theorem 9 (see also Remark 8) and Proposition 5 that the diameter of $\mathcal{G}^{\Rightarrow NT}_{[E]}$ is polynomial whenever E is regular, even in the case that E is not basic.

▶ **Theorem 25.** *Let E be a basic RWE. Then* $\mathrm{diam}(\mathcal{G}^{\Rightarrow}_{[E]}) \in O(|E|^{10})$. *Consequently, for any RWE E,* $\mathrm{diam}(\mathcal{G}^{\Rightarrow NT}_{[E]}) \in O(|E|^{12})$.

Thus, by Proposition 4, we may infer that the satisfiability problem for RWEs is in NP. It was already shown in [8] that the satisfiability problem for RWEs is NP-hard, and thus we obtain matching upper and lower bounds for its complexity.

▶ **Theorem 26.** *The satisfiability problem for RWEs is NP-complete.*

4.6 Size and DAG-width

While the diameter of $\mathcal{G}^{\Rightarrow}_{[E]}$ is one important parameter, being directly related to the complexity of the satisfiability problem, it is by no means the only interesting one. The overall size of the graphs will also play a central role in the practical performance of the algorithm described in Section 3. For basic RWEs, we have the following tight upper and lower bounds on the number of vertices in the graphs $\mathcal{G}^{\Rightarrow}_{[E]}$.

▶ **Theorem 27.** *Let E be a basic RWE and let $n = \mathrm{Card}(\mathrm{var}(E))$. Suppose that $n > 1$. Let V be the number of vertices in $\mathcal{G}^{\Rightarrow}_{[E]}$. $2^{n-1} - 1 \leq V \leq \frac{n!}{2}$.*

It is worth noting that the lower bound given by Theorem 27 is already exponential in the number of variables. The interpretation of the theorem in the more general (i.e. not basic) setting therefore tells us that the number of vertices in $\mathcal{G}^{\Rightarrow}_{[E]}$ is exponential in the number of variables occurring twice in the appropriate (indecomposable) parts of the LHS and RHS. In other words, we see the rather intuitive fact here that decomposable equations are somehow easier to deal with than indecomposable equations of the same length. The following demonstrates that the bounds given by Theorem 27 are tight.

▶ **Theorem 28.** *Let E be a basic RWE and let $n = \mathrm{Card}(\mathrm{var}(\alpha))$. Suppose that $n > 1$. Let V be the number of vertices in $\mathcal{G}^{\Rightarrow}_{[E]}$. Then:*

1. *$V = 2^{n-1} - 1$ if and only if there exists $E' \in [E]_{\Rightarrow}$ such that E' is regular reversed,*
2. *$V = \frac{n!}{2}$ if and only if there exists $E' \in [E]_{\Rightarrow}$ such that E' is regular rotated.*

In addition to the size we are also able to give some insights about the connectedness of the graphs, which, as discussed in Section 3.3, are of interest when solving RWEs modulo additional constraints. We show firstly that there exist classes of equations E for which $\mathrm{dgw}(\mathcal{G}^{\Rightarrow_{NT}}_{[E]})$ may be arbitrarily large.

▶ **Theorem 29.** *Let $x, y, z_0, z_1, z_2, \ldots, z_n \in X$. Let E be the RWE given by $xz_0z_1z_2 \ldots z_ny \doteq yz_0z_nz_{n-1} \ldots z_1x$. Then $\mathrm{dgw}(\mathcal{G}^{\Rightarrow_{NT}}_{[E]}) > n$.*

Since high connectivity can be seen as an obstacle to deciding the satisfiability problem with additional constraints, it is also worth noting classes for which the DAG-width is bounded by a small constant, such as with those described in the next theorem.

▶ **Theorem 30.** *Let $\alpha_1, \alpha_2, \ldots, \alpha_n, \beta_1, \beta_2, \ldots, \beta_n \in X^*$ such that $\mathrm{var}(\alpha_i) = \mathrm{var}(\beta_i)$ for $1 \leq i \leq n$ and $\mathrm{var}(\alpha_i) \cap \mathrm{var}(\alpha_j) = \emptyset$ for $1 \leq i, j \leq n$ with $i \neq j$. Let E be the RWE $\alpha_1\alpha_2 \ldots \alpha_n \doteq \beta_1\beta_2 \ldots \beta_n$. Then $\mathrm{dgw}(\mathcal{G}^{\Rightarrow_{NT}}_{[E]}) = 2$.*

4.7 Extension to systems of equations

So far, we have considered individual equations. However, it is often the case in practice that there is not just one equation to be solved, but a system of several concurrent equations. However, while constructions exist which transform a system of equations into a single equation (see e.g. [17]), the resulting equation will generally not be quadratic/regular. We extend the definition of regular equations to regular systems as follows.

▶ **Definition 31** (Regular systems). *Let $\Theta = \{\alpha_1 \doteq \beta_1, \alpha_2 \doteq \beta_2, \ldots, \alpha_n \doteq \beta_n\}$ be a system of word equations. An* orientation *of Θ is any element of $\{\alpha_1 \doteq \beta_1, \beta_1 \doteq \alpha_1\} \times \{\alpha_2 \doteq \beta_2, \beta_2 \doteq \alpha_2\} \times \ldots \times \{\alpha_n \doteq \beta_n, \beta_n \doteq \alpha_n\}$. We say that Θ is regular if it has an orientation for which each variable occurs at most once across all LHSs and at most once across all RHSs.*

▶ **Theorem 32.** *The satisfiability problem for regular systems of equations is NP-complete. Moreover, whether a system of word equations is regular can be decided in polynomial time.*

5 Conclusions

A famous algorithm for solving quadratic word equations can be used to produce a (directed) graph containing all solutions to the equation. In the case of regular equations, we have described some underlying structures of these graphs with the intention of better understanding their solution sets. We give bounds on their diameter and number of vertices, as well as provide classes with bounded (resp. unbounded) DAG-width. Probably the most significant result arising from our analysis is that the satisfiability problem for regular word equations is in NP (and thus NP-complete), which we also extend to regular systems of equations.

We leave open many interesting problems, the most obvious of which is to generalise our results to the (full) quadratic case. We also believe that our analysis and techniques open up the possibility to investigate in far more detail the graphs $\mathcal{G}^{\Rightarrow}_{[E]}$, even in the case of regular equations. For example, in light of our results, it seems reasonable to suggest that determining whether $E_1 \Rightarrow^* E_2$ for two regular equations E_1 and E_2 may be done in polynomial time. A particularly nice characterisation of E_1 and E_2 such that $E_1 \Rightarrow^* E_2$ might yield a much quicker algorithm than the one resulting from our bound on the diameter of $\mathcal{G}^{\Rightarrow NT}_{[E]}$ by significantly reducing the degree of the polynomial. We also expect that a detailed analysis of the length-reducing transformations and symmetries which may be found there would be particularly helpful in understanding further the structure of solution sets and the performance of algorithms solving regular equations in practice.

Finally, we mention the task of investigating the decidability of the satisfiability problem for regular equations with additional constraints, in particular length constraints, with the hope that having identified cases where the DAG-width is particularly high/low, along with improved means to describe precisely the structure of the solution-graphs, might provide some useful hints with how to proceed in this direction.

References

1 P. A. Abdulla, M. F. Atig, Y. Chen, L. Holík, A. Rezine, P. Rümmer, and J. Stenman. Norn: An SMT solver for string constraints. In *Proc. Computer Aided Verification (CAV)*, volume 9206 of *Lecture Notes in Computer Science (LNCS)*, pages 462–469, 2015.

2 M. Alkhalaf, T. Bultan, and F. Yu. STRANGER: An automata-based string analysis tool for PHP. In *Proc. Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, volume 6015 of *Lecture Notes in Computer Science (LNCS)*, 2010.

3 D. Angluin. Finding patterns common to a set of strings. *Journal of Computer and System Sciences*, 21:46–62, 1980.

4 C. Barrett, C. L. Conway, M. Deters, L. Hadarean, D. Jovanović, T. King, A. Reynolds, and C. Tinelli. CVC4. In *Proc. Computer Aided Verification (CAV)*, volume 6806 of *Lecture Notes in Computer Science (LNCS)*, pages 171–177, 2011.

5 D. Berwanger, A. Dawar, P. Hunter, S. Kreutzer, and J. Obdrzálek. The DAG-width of directed graphs. *Journal of Combinatorial Theory, Series B*, 102(4):900–923, 2012.

6 M. Berzish, V. Ganesh, and Y. Zheng. Z3str3: A string solver with theory-aware heuristics. In *Proc. Formal Methods in Computer-Aided Design (FMCAD)*, pages 55–59. IEEE, 2017.

7 J. D. Day, V. Ganesh, P.l He, F. Manea, and D. Nowotka. The satisfiability of word equations: Decidable and undecidable theories. In I. Potapov and P. Reynier, editors, *In Proc. 12th International Conference on Reachability Problems, RP 2018*, volume 11123 of *Lecture Notes in Computer Science (LNCS)*, pages 15–29, 2018.

8 J. D. Day, F. Manea, and D. Nowotka. The hardness of solving simple word equations. In *Proc. Mathematical Foundations of Computer Science (MFCS)*, volume 83 of *LIPIcs*, pages 18:1–18:14, 2017.

9 J. D. Day, F. Manea, and D. Nowotka. Upper bounds on the length of minimal solutions to certain quadratic word equations. In *Proc. Mathematical Foundations of Computer Science (MFCS)*, volume 138 of *LIPIcs*, pages 44:1–44:15, 2019.

10 V. Diekert, A. Jeż, and W. Plandowski. Finding all solutions of equations in free groups and monoids with involution. *Information and Computation*, 251:263–286, 2016.

11 V. Diekert and J. M. Robson. On quadratic word equations. In *Proc. 16th Annual Symposium on Theoretical Aspects of Computer Science, STACS*, volume 1563 of *Lecture Notes in Computer Science (LNCS)*, pages 217–226, 1999.

12 A. Ehrenfeucht and G. Rozenberg. Finding a homomorphism between two words is NP-complete. *Information Processing Letters*, 9:86–88, 1979.

13 D. D. Freydenberger. A logic for document spanners. *Theory of Computing Systems*, 63(7):1679–1754, 2019.

14 D. D. Freydenberger and M. Holldack. Document spanners: From expressive power to decision problems. *Theory of Computing Systems*, 62(4):854–898, 2018.

15 A. Jeż. Recompression: A simple and powerful technique for word equations. *Journal of the ACM*, 63, 2016.

16 A. Jeż. Word equations in nondeterministic linear space. In *Proc. International Colloquium on Automata, Languages and Programming (ICALP)*, volume 80 of *LIPIcs*, pages 95:1–95:13, 2017.

17 J. Karhumäki, F. Mignosi, and W. Plandowski. The expressibility of languages and relations by word equations. *Journal of the ACM*, 47:483–505, 2000.

18 A. Kiezun, V. Ganesh, P. J. Guo, P. Hooimeijer, and M. D. Ernst. HAMPI: a solver for string constraints. In *Proc. ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA)*, pages 105–116. ACM, 2009.

19 A. W. Lin and P. Barceló. String solving with word equations and transducers: towards a logic for analysing mutation xss. In *ACM SIGPLAN Notices*, volume 51, pages 123–136. ACM, 2016.

20 A. W. Lin and R. Majumdar. Quadratic word equations with length constraints, counter systems, and Presburger arithmetic with divisibility. In S. K. Lahiri and C. Wang, editors, *In Proc. 16th International Symposium on Automated Technology for Verification and Analysis (ATVA)*, volume 11138 of *Lecture Notes in Computer Science (LNCS)*, pages 352–369. Springer, 2018.

21 M. Lothaire. *Algebraic Combinatorics on Words.* Cambridge University Press, Cambridge, New York, 2002.

22 G. S. Makanin. The problem of solvability of equations in a free semigroup. *Sbornik: Mathematics*, 32(2):129–198, 1977.

23 F. Manea, D. Nowotka, and M. L. Schmid. On the complexity of solving restricted word equations. *International Journal of Foundations of Computer Science*, 29(5):893–909, 2018.

24 E. Petre. An elementary proof for the non-parametrizability of the equation $xyz = zvx$. In *Proc. 29th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, volume 3153 of *Lecture Notes in Computer Science (LNCS)*, pages 807–817, 2004.

25 W. Plandowski. Satisfiability of word equations with constants is in PSPACE. In *Proc. Foundations of Computer Science (FOCS)*, pages 495–500. IEEE, 1999.

26 W. Plandowski and W. Rytter. Application of Lempel-Ziv encodings to the solution of words equations. In *Proc. International Colloquium on Automata, Languages and Programming (ICALP)*, volume 1443 of *Lecture Notes in Computer Science (LNCS)*, pages 731–742, 1998.

27 K. U. Schulz. Makanin's algorithm for word equations-two improvements and a generalization. In *International Workshop on Word Equations and Related Topics*, pages 85–150. Springer, 1990.