Word Equations: Sheet 1

In context of word equations we always consider a finite alphabet Σ and finite set of variables $\mathcal{X}$, which is disjoint with Σ. Elements of Σ are usually denoted by small letters $a, b, c, \ldots$. Elements of $\mathcal{X}$ are usually denoted as $X, Y, Z, \ldots$.

A *word equation* is a pair (u, v), usually written as $u = v$, where $u, v \in (\Sigma \cup \mathcal{X})^*$. A *system of word equations* is a set of word equations, usually denoted as (u_1, v_1), (u_2, v_2), ...

A *substitution* is a morphism $S : \mathcal{X} \mapsto \Sigma^+$ (we sometimes also consider Σ^*, but then we explicitly state this). It is extended to Σ as an identity (so $S(a) = a$ for $a \in \Sigma$) and to $(\Sigma \cup \mathcal{X})^*$ as a homomorphism (so $S(\alpha\beta) = S(\alpha)S(\beta)$ for $\alpha, \beta \in (\Sigma \cup \mathcal{X})^+$.

A substitution is a *solution* of a word equation $u = v$, when $S(u) = S(v)$ (the solution of a system of equations is defined similarly). A solution S of a word equation $u = v$ is *length-minimal* (or simply *minimal*), when for any other solution S' it holds that

$$|S(u)| \leq |S'(u)| \ .$$

A *satisfiability problem* for word equations is:
"Given a system of word equations decide, whether they have a solution."

We say that a system of word equations is *quadratic*, if every variable occurs at most twice in it. It is *cubic*, when every variable occurs at most thrice.

Constraints for system of word equations are given as additional constraints of the form $X \in C$ or $X \notin C$, where $X \in \mathcal{X}$ and C comes from some specified language class (say: regular, context-free, etc.). The meaning of the constraint $X \in C$ (or $X \notin C$) is that we require from a solution S that $S(X) \in C$ (or $S(X) \notin C$).

Task 1 Show that a satisfiability of a system of word equations is NP-hard already when $\Sigma = \{a\}$.
Hint: This reduces to some other known equations.

Task 2 Show that the satisfiability of word equations is NP-hard when we consider only systems in which every v_i does not contain variables. (Note: it might be easier to show this when we allow also ϵ as a substitution for a variable).

Task 3 Show that the problem of satisfiability of a system of word equations can be reduced to the problem of satisfiability of a single word equation, when we are allowed to add letters to the alphabet. Show the same result also when adding letters is not allowed, but $|\Sigma| \geq 2$.

Task 4 Suppose that S is a length-minimal solution of an equation $u = v$ over an alphabet Σ.

- Suppose that a occurs in the word $S(u)$. Show that it occurs in u or in v. (In other words: length-minimal solution uses only letters that occur in the equation).
- Suppose that a factor ab, where $a \neq b$ and $a, b \in \Sigma$, occurs in $S(u)$. Show that there is an occurrence of ab in $S(u)$ or in $S(v)$ such that those two letters do not come from a substitution for the same occurrence of a variable (in other words: either one of them was already in the equation or they come from substitutions for two occurrences of variables).

Task 5 Reduce the satisfiability problem for word equations to the satisfiability problem of cubic word equations

Task 6 Show that the problem of word equations with context-free constraints is undecidable. (Meaning, that we allow constraints with languages from the class of context-free languages).

Task 7 Show that the problem: "given DFAs (deterministic finite automata) $D_1, D_2, \ldots, D_m$ decide, whether the intersection of their languages is non-empty" is PSPACE-hard. Deduce from this that word equations with regular constraints are PSPACE-hard.

Task 8[Long: two points] Consider a mapping from $\Sigma = \{a, b\}$ to 2×2 matrices over $\mathbb{N}$, defined as

$$\varphi(a) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \text{ and } \varphi(b) = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \ .$$

Extend this to Σ^* as a homomorphism.

Show that for any $w \in \Sigma^*$ its image is a a matrix with a determinant one.

Show that this mapping is injective; to do this, consider, what are the rows of a matrix $\varphi(a) \begin{bmatrix} n & n' \\ m & m' \end{bmatrix}$ and what are the rows of $\varphi(a) \begin{bmatrix} n & n' \\ m & m' \end{bmatrix}$. Deduce from this that looking at the matrix $M_w = \varphi(w)$ we can determine the left-most letter of w by looking at rows of M_w.

Show that if a 2×2 matrix M with determinant 1 and all natural entries can be represented as either $\varphi(a)M'$ or $\varphi(b)M'$, where M' has a determinant 1 and all natural entries. Again: compare the rows.

Deduce from this that φ is an isomorphism between Σ^* and 2×2 matrices with determinant 1 and all entries natural.

Deduce from this that satisfiability of word equation over $\Sigma = \{a, b\}$ reduces to the satisfiability of equations over natural numbers (to do this, represent a $\varphi(X)$ as a matrix of variables representing natural numbers).