CSP lecture 16/17 winter semester – Problem Set 6

An instance of the CSP($\mathbb{A}$) with the set of variables V is called 1-*minimal* if there exists a system of subsets $P_x \subseteq A, x \in V$ such that for every constraint $R(x_1, \dots, x_k)$, the projection of R onto the j-th coordinate is equal to P_{x_j}.

Two instances of the CSP are called *equivalent* if they have the same set of solutions.

Problem 1. Devise a polynomial algorithm that transforms an instance of CSP($\mathbb{A}$) into an equivalent 1-minimal instance of CSP($\mathbb{B}$), where $\mathbb{B}$ is pp-definable from $\mathbb{A}$.

A *semilattice operation* on A is a binary operation $s : A^2 \to A$ such that for all $a, b, c \in A$

$$s(s(a,b),c) = s(a,s(b,c)), \quad s(a,b) = s(b,a), \quad s(a,a) = a$$

A *totally symmetric operation* on A of arity n is an operation $t : A^n \to A$ such that $t(a_1, \dots, a_n) = t(b_1, \dots, b_n)$ whenever $\{a_1, \dots, a_n\} = \{b_1, \dots, b_n\}$, that is, the result of t depends only on the set of its arguments.

Problem 2. Prove that every clone that contains a semilattice operation also contains, for each n, a totally symmetric operation of arity n. Observe that the binary minimum and maximum operations on $\{0, 1\}$ are semilattice operations.

Problem 3. Prove that CSP($\mathbb{A}$) is solvable in polynomial time whenever, for each n, $\mathbb{A}$ has a totally symmetric polymorphism of arity n. (Hint: Use Problems 1 and 2, reject if some P_x is empty, otherwise apply a totally symmetric operation of sufficiently large arity to each P_x and show that the resulting elements form a solution.)

An instance of the CSP is called a *simple* $(2,3)$-*minimal instance* if

- The set of variables is $V = \{x_1, \dots, x_m\}$
- For each $1 \leq i \leq m$, there is a (single) unary constraint $P_i(x_i)$
- For each pair $1 \leq i < j \leq m$, there is a (single) binary constraint $P_{i,j}(x_i, x_j)$
- There are no other constraints than those from the previous two items
- For each pair $1 \leq i < j \leq m$, the projection of $P_{i,j}$ onto the first (second, resp.) coordinate is equal to P_i (P_j, resp.).
- For each triple $1 \leq i, j, k \leq m$ of distinct integers and each $(a, b) \in P_{i,j}$, there exists $c \in P_k$ such that $(a, c) \in P_{i,k}$ and $(b, c) \in P_{j,k}$. Here, for $i > j$, we define $P_{i,j} = \{(a, b) : (b, a) \in P_{j,i}\}$.

A simple $(2,3)$-minimal instance is best visualized as a multipartite graph as follows: Each variable x_i corresponds to one partite set whose vertex set is (a disjoint copy of) P_i. Edges between P_i and P_j are given by the relation $P_{i,j}$. Interpret the last two items using this graph. Also interpret solutions of the instance.

Problem 4. Devise a polynomial time algorithm to transform an instance of CSP($\mathbb{A}$), where all relations in $\mathbb{A}$ are at most binary, to an equivalent simple $(2,3)$-minimal instance of CSP($\mathbb{B}$), where $\mathbb{B}$ is pp-definable from $\mathbb{A}$.

Adjust the algorithm to the situation when $\mathbb{A}$ has a majority polymorphism but the relations in $\mathbb{A}$ can have arbitrary arities.

Problem 5. Prove that CSP($\mathbb{A}$) is solvable in polynomial time whenever $\mathbb{A}$ has a majority polymorphism. Strategy:

- Deduce from the previous problem that it is enough to show that a simple $(2,3)$-minimal instance of CSP($\mathbb{B}$) has a solution whenever $\mathbb{B}$ has a majority polymorphism m and each P_i is nonempty.
- Gradually build a solution as follows. Take any $a_1 \in P_1$, $a_2 \in P_2$, $a_3 \in P_3$ such that $(a_1, a_2) \in P_{1,2}$, $(a_1, a_3) \in P_{1,3}$, $(a_2, a_3) \in P_{2,3}$ (a partial solution on variables x_1, x_2, x_3).
- Take $b \in P_4$ such that $(a_2, b) \in P_{2,4}$ and $(a_3, b) \in P_{3,4}$. Take $b' \in P_4$ such that $(a_1, b') \in P_{1,4}$ and $(a_3, b') \in P_{3,4}$. Take $b'' \in P_4$ such that $(a_1, b'') \in P_{1,4}$ and $(a_2, b'') \in P_{2,4}$. Define $a_4 = m(b, b', b'')$ and show that a_1, a_2, a_3, a_4 is a partial solution on variables x_1, x_2, x_3, x_4.
- Continue similarly.