Electronic Colloquium on Computational Complexity, Report No. 5 (2005)

Constraint satisfaction on finite groups with near subgroups

Tomás Feder*

November 10, 2004

Abstract

Constraint satisfaction on finite groups, with subgroups and their cosets described by generators, has a polynomial time algorithm. For any given group, a single additional constraint type that is not a coset of a near subgroup makes the problem NP-complete. We consider constraint satisfaction on groups with subgroups, near subgroups, and their cosets. We give two polynomial time algorithms for the case of solvable groups. We then give a polynomial time algorithm for general groups with subgroups, near subgroups, and their cosets. Bulatov has shown that Mal'tsev constraints have a polynomial time algorithm; we finally show that subgroups, near subgroups, and their cosets are Mal'tsev constraints. Our results generalize the results of Bulatov on Mal'tsev in the special case of near subgroups and some cases of twisted subgroups by only requiring the subgroups and their cosets to be given by generators describing possibly a constraint of exponential size, and allowing different variables to have different domains of varying size, with corresponding group operations.

*268 Waverley St., Palo Alto, CA 94301. E-mail: tomas@theory.stanford.edu.

ISSN 1433-8092

1 Introduction

Feder and Vardi [10] considered the class CSP of constraint satisfaction problems. The main question addressed is: Which problems in CSP are polynomial and which are NP-complete? The classification of the computational complexity of problems in Boolean CSP had been settled earlier by Schaefer [17].

The first tool considered in [10] for showing that a problem in CSP is polynomial is the language Datalog. It is shown that problems whose complement is expressible in Datalog have *width* k for some constant k. A simple canonical program solves such problems, essentially in time $O(n^k)$. The program depends only on k and not on the actual CSP problem.

The next step proposed in [10] is to determine which problems cannot be approached with Datalog. A property of CSP problems called the *ability to count* is defined. It is shown that problems with this property cannot even be solved by means of Datalog with successor. The proof is an adaptation of Razborov's super-polynomial lower bound on the size of monotone circuits for graph matching [16], with the connection to Datalog with successor given by Afrati, Cosmadakis, and Yannakakis [1].

The only problems in CSP known to have the ability to count acquire it by simulating linear equations modulo p, which has a polynomial algorithm. In the context of the original classification project, it is natural to ask which extensions of this problem are polynomial and which are NP-complete.

Linear equations modulo p are essentially the same as the CSP problems whose constraints are defined by subgroups and their cosets in an abelian group. For such an abelian group, it is shown in [10] that any additional constraint type that is not a coset of a subgroup makes the problem NP-complete.

The problem with subgroups and their cosets remains polynomial even if the group is not abelian, based on the approach of Furst, Hopcroft and Luks [12]. For groups that are not abelian, additional constraints that are not cosets of subgroups resist attempts at showing NP-completeness, with one exception. If for a constraint type, the group has an abelian section where a coset of the constraint containing 1 does not define a subgroup, then the problem is NP-complete by the result on abelian groups.

One of the definitions of a *near subgroup* of a group is a set containing 1 whose every coset containing 1, when restricted to an abelian section, defines a subgroup. Thus sets that are not cosets of near subgroups make the problem for a given group NP-complete. The proof is a direct simulation of one-in-three SAT. On the other hand, problems whose constraints are subgroups, near subgroups, and their cosets, cannot simulate one-in-three SAT directly, provided that the intersection of near subgroups is a near subgroup [10]. This property of near subgroups is later shown by Aschbacher [3].

This suggests that CSP problems with near subgroups may not be NP-complete, and starts the project of finding polynomial time algorithms for these problems, of which this paper is a continuation. It is shown in [10] that the problem with near subgroups is polynomial for groups of odd order. For arbitrary groups, the problem decomposes into an *odd order problem*, for elements of odd order, and a *2-element problem*, for elements of order a power of two. In [10], the odd order problem is shown polynomial, while the 2-element problem reduces to the *involution problem*, where a certain element has order two.

The involution problem is polynomial when the near subgroups satisfy a certain *2-element* *property.* This property holds, for instance, for groups with a normal Sylow 2-subgroup, which are solvable. Aschbacher shows that the 2-element property holds for groups without elements of order four, which are not necessarily solvable, but fails for certain groups that are not solvable. Eventually, he constructs examples of solvable groups for which the 2-element property does not hold [3].

In this paper, we continue the project by attempting to solve the general case of groups with subgroups, near subgroups, and their cosets. We consider a more general model where subgroups and their cosets may not be explicitly described but given by generators, and the problem may involve several groups which are given as part of the input, rather than a single fixed group as in CSP. The effect of this is that some polynomial pre-processing of the groups, to determine their structure in relation to the given near subgroups, may be required.

The first aim is to handle the case of solvable groups which was left open in [10]. Aschbacher [3] showed that solvable groups generated by the 2-elements in a near subgroup have a very special structure. Since for the 2-element or involution problem we are only interested in 2-elements, this characterization is of interest here. It lends itself to simple algorithms for recognizing and handling such solvable groups. Two algorithms for the case of solvable groups with near subgroups are given here. The first one depends only on the basic structure of the solvable group, while the second one depends on a property of a certain automorphism associated with near subgroups, which is known to hold in the case of solvable groups, as suggested by Aschbacher [4].

We then give an algorithm for the case of general groups with near subgroups. The algorithm reduces the involution problem on arbitrary groups to the problem for certain solvable groups defined from the given groups. This result depends on a structural characterization of groups generated by the 2-elements in a near subgroup obtained by Aschbacher [5].

Bulatov [7, 8] has shown that Mal'tsev constraints have a polynomial time algorithm; we finally show that subgroups, near subgroups, and their cosets are Mal'tsev constraints.

In the light of the results presented here and related work, one may conjecture that the following approach can lead to a full classification of the problems in CSP as polynomial or NP-complete. (1) It is decidable whether a CSP problem simulates one-in-three SAT directly or not; if it does, the problem is NP-complete. Otherwise, the following two-step approach may constitute a polynomial time algorithm: (2) Perform the width k inference procedure for some appropriate constant k, essentially in time $O(n^k)$; (3) If the constraints thus obtained are nonempty, then interpret them for some appropriately chosen group G as describing subgroups, near subgroups, and their cosets, in the direct product G^n, or more generally as Mal'tsev constraints; solve this problem. All currently known polynomial cases of CSP yield to this approch: in brief, they are group theoretic or Mal'tsev problems underneath a Datalog, bounded width layer.

While the work of Bulatov on Mal'tsev constraints superseeds the case of near subgroups in what concerns CSP, the approach taken here has several advantages. First of all, the proof presented here that near subgroups are Mal'tsev constraints is nontrivial. Furthermore, near subgroups lead to a dichotomy result for extensions of problems with subgroups and cosets, since for such problems we may only add near subgroups and their cosets to remain polynomial, any other kind of subset of a group leads to NP-completeness.

A deeper advantage of the approach presented here is that we do not require each variable to range over the same group, or that the subgroups and cosets be defined for a bounded number of variables at the same time. This allows for subgroups and cosets that may be of exponential size, and therefore must be described succintly by means of generators. Most if not all the polynomial results for CSP depend on closure properties of the subsets used as constraints. As far as we know, this is the first result that shows that if the subsets are not described explicitly, but are given as the closure under a closure function of a smaller set of elements, the problem remains polynomial. It would be in particular interesting to know if Mal'tsev problems remain polynomial when a subset is defined as the closure under a Mal'tsev function, possibly with a different Mal'tsev function for each variable, of an explicitly given set of elements. This is the case in this paper for subgroups and their cosets, as they may be of exponential size but given by a small number of generatoes.

Near subgroups were defined by Feder and Vardi [10], and studied in greater depth by Aschbacher [3, 5]. The theory of near subgroups acquires greater interest in the context of the recent work of Feder [9] and of Aschbacher [6], were strong near subgroups are considered, with important links both to the theory of loops via order dividing properties as studied by Glauberman [13, 14], and to the theory of gyrogroups as studied by Foguel and Ungar [11, 18, 19]. The results presented here indicate that near subgroups have both mathematical and computational interest.

2 Near subgroups of finite groups

The reader is directed to [2] for notation, terminology, and basic results on finite groups.

Let G be a finite group. A subset K of G is a *near subgroup* of G if $1 \in K$ and for all $b \in G$ such that $1 \in bK$, for all subgroups M of G, and for all normal subgroups N of M such that $M^* = M/N$ is abelian, the set $(bK)^* = \{aN \subseteq M : bK \cap aN \neq \emptyset\}$ is a subgroup of M^*. In brief, the intersection of K with the abelian sections of G forms subgroups.

Of course, ordinary subgroups are near subgroups. For the purpose of our study, a different but equivalent definition is more useful. A subset K of G is a *twisted subgroup* of G if it satisfies the following two conditions:

(1) $1 \in K$.

(2) If $x, y \in K$ then $xyx \in K$.

Define a twisted subgroup K of G to be a *near subgroup* of G if it also satisfies the following condition:

(3) If N is a normal subgroup of $M \leq G$ with M/N isomorphic to E_4, then there is no $b \in G$ such that $bK \cap M$ meets exactly three of the four cosets of N in M.

The following properties of twisted and near subgroups were known in [10] and appear with proof in [3].

Lemma 1 *(1) If K is a twisted subgroup of G, then $\langle x\rangle \subseteq K$ for each $x \in K$.*

(2) If $H \leq G$ and K is a twisted subgroup (near subgroup) of G then $H \cap K$ is a twisted subgroup (resp. near subgroup) of G.

(3) If $a, b \in G$ and K is a twisted subgroup (near subgroup) of G with $1 \in aKb$, then aKb is a twisted subgroup (resp. near subgroup) of G.

(4) If N is a normal subgroup of G and K is a twisted subgroup (near subgroup) of G then K^* is a twisted subgroup (resp. near subgroup) of $G^* = G/N$.

(5) If N is a normal subgroup of G with $bN \subseteq K$ for all $b \in K$, and K^* is a twisted subgroup (near subgroup) of $G^* = G/N$, then K is a twisted subgroup (resp. near subgroup) of G.

(6) If K is a near subgroup of $G = \langle K \rangle$, and G is either abelian or a 2-group, then $G = K$.

The following result is also shown in [10]. Let K be a twisted subgroup of G. In $G \times G$, consider the subgroup $H_K = \langle \{(x, x^{-1}) : x \in K\} \rangle$.

Lemma 2 *(1) If $(x, y^{-1}) \in H_K$, then $xy \in K$.*

(2) If $z \in K$ has order $2l - 1$, and we let $x = y = z^l$, then $z = xy$ and $(x, y^{-1}) \in H_K$.

Proof: We first prove (1). If $(x, y^{-1}) \in H_K$, then $x = t_1 t_2 \cdots t_n$ and $y^{-1} = t_1^{-1} t_2^{-1} \cdots t_n^{-1}$ with $t_i \in K$. Then $xy = t_1 t_2 \cdots t_n t_n \cdots t_2 t_1 \in K$ follows from condition (2) in the definition of a twisted subgroup.

For (2), $x = y = z^l \in K$ follows from (1) of Lemma 1, with $xy = z^{2l} = z$, and (x, y^{-1}) is a generator for H_K. ■

The following is from [3]. Let K be a twisted subgroup of G such that $G = \langle K \rangle$. In $G \times G$, let $H_K = \langle \{(x, x^{-1}) : x \in K\} \rangle$ as before, and let $N_K = \{x \in G : (x, 1) \in H_K\}$.

Lemma 3 *(1) N_K is a normal subgroup of G, and $bN_K \subseteq K$ for each $b \in K$.*

(2) If $N_K = 1$, then $H_K = \{(x, x^\tau) : x \in G\}$, where $\tau = \tau_K$ is an automorphism of G with $\tau^2 = 1$, and $K \subseteq K(\tau) = \{x \in G : x^\tau = x^{-1}\}$.

G is said to be K-*reduced* if $N_K = 1$. In general $\hat{G} = G/N_K$ is $\hat{K}$-reduced.

Theorem 1 *[3] If K and L are near subgroups of G, then $K \cap L$ is a near subgroup of G.*

Let S denote the set of 2-elements of G, and let $O(G)$ denote the largest normal subgroup of G of odd order. Suppose K is a near subgroup of G such that G is K-reduced and $G = \langle S \cap K \rangle$.

Theorem 2 *[3] If G is solvable, then $G = O(G)T$ for some Sylow 2-subgroup T of G with T abelian and $T \subseteq K = K(\tau)$.*

Let $Z(G)$ denote the center of G.

Theorem 3 *[5] Suppose K is a near subgroup of $G = \langle S \cap K \rangle$ and G is K-reduced with corresponding automorphism $\tau = \tau_K$. Let $G^* = G/O(G)$, and let $\tilde{G} = G^*/Z(G^*)$. Then*

(1) $K = K(\tau)$.

(2) $Z(G^*)$ is an abelian 2-group and $Z(G^*) \subseteq K^* = K(\tau^*)$.

(3) $\tilde{G}$ is a direct product $\tilde{G} = G_1 G_2 \cdots G_n$, where the G_i are nonabelian simple groups.

(4) $\tilde{K} = K_1 K_2 \cdots K_n$, where each $K_i = K(\tau_i)$ is a near subgroup of G_i.

The proof in [5] essentially shows that the characterization whose proof was sketched in [3] goes through. Aschbacher's result further contains a complete list of the possible G_i, K_i, τ_i.

3 Group constraint satisfaction

We now introduce the basic model for group constraint satisfaction. An instance of the problem has groups $G_1, G_2, \ldots, G_n$. The candidate solutions to the problem are from $G = G_1 \times G_2 \times \cdots \times G_n$. For $I \subseteq [n]$, let G_I denote the direct product of the G_i with $i \in I$; in particular, $G = G_{[n]}$. An instance also has a collection of subsets $A_I \subseteq G_I$ called *constraints*. The aim of the problem is to find an element $x \in G$ such that the projection x_I if x into G_I satisfies $x_I \in A_I$ for all given $A_I \subseteq G_I$. The usual model for the class CSP corresponds to the case where all G_i are the same and fixed independently of the instance of the problem; furthermore the possible $A_I \subset G_I$ are also fixed ahead of time, independently of the instance of the problem.

If $A_I = a_I B_I$ with $1 \in B_I$, and B_I is not a subgroup of G_I, then we shall assume that the instance has an additional $G_m = \langle B_I \rangle$ with $m > n$. The set B_I can then be described by some $B_m \subseteq G_m$, and the correspondence between G_m and each of the G_i with $i \in I$ needed to describe $A_I = a_I B_I \subseteq G_I$ is given by a coset in $G_m \times G_i$. After this transformation, all the A_I that are not cosets of subgroups have $|I| = 1$, and we denote them by A_i, where $I = \{i\}$.

The following is from [10].

Lemma 4 *Let G_0 be a group and $A_0 \subseteq G_0$ be a subset that is not a coset of a near subgroup. Consider the problem where each G_i is isomorphic to G_0 and has the constraint A_i corresponding to A_0; the remaining $A_I \subseteq G_I$ are cosets of subgroups and have $|I| \leq 3$. This problem is NP-complete.*

From now on, we shall assume that all $A_i \subseteq G_i$ are cosets of near subgroups. The remaining constraints are cosets $b_j H_j$ of subgroups $H_j \leq G$. We generalize the model. Instead of describing $b_j H_j$ by a subset $A_I \subseteq G_I$, we are given generators for $H_j \leq G$. If $|G_i| \leq m$ for each $i \in [n]$, then at most nm generators are needed to describe each H_j.

If the intersection of the $b_j H_j$ is nonempty, then it is given by $bH = \cap_j b_j H_j$ with $H \leq G$. The following follows as in [10], based on the approach of [12]; see also Theorem II.12 in [15].

Theorem 4 *One can find $b \in bH$ and generators for H from the given b_j and generators for the H_j, in polynomial time.*

Proof: The main observation is that, given a group J with known generators and a chain of subgroups $J = J_0 > J_1 > \cdots > J_r = \{1\}$, one can obtain distinct representatives from each coset of each J_j in J_{j-1}, namely one element from each coset, as follows. Select two elements x, x' among the generators of J_0 that belong to the same coset of J_1, say $x' = xy$ with $y \in J_1$; then discard x' and add y to the list of generators. Iterate until there is only one generator in each coset of each J_j in J_{j-1}, and carry out the process for products xy of two current generators as well. The fact that only products of pairs of generators are needed to obtain representatives for all cosets of each J_j in J_{j-1} requires proof; see Theorem II.8 in [15].

In our application, we are looking for a solution in $G = G_1 \times G_2 \times \cdots \times G_n$. We have cosets $b_j H_j$ for $1 \leq j \leq s$. Let $J_j = G \cap H_1 \cap H_2 \cap \cdots \cap H_j$ for $0 \leq j \leq s$. From J_s, let J_{s+1} consist of those elements of J_s whose projection into G_1 is 1, let J_{s+2} consist of those elements of J_{s+1} whose projection into G_2 is 1, and so on until $J_r = J_{s+n} = 1$ is obtained. Now obtain representatives for all cosets of each J_j in J_{j-1} using the above algorithm.

To solve the constraint satisfaction problem that obtains an element $b \in bH$, observe that the first coset b_1H_1 is a coset of $J_1 = H_1$ in $J_0 = G$, so we may select a representative b for this coset from the above representation, and then look for a solution of the form bx with b fixed and x in J_1. Having fixed b, a condition $bx \in b_jH_j$ now becomes $x \in b^{-1}b_jH_j = c_jH_j$. Now we proceed with c_2H_2 and J_2 as we did before for b_1H_1 and J_1. Here it might be that no coset representative c for J_2 in J_1 is in c_2H_2, in which case the problem has no solution. If such a representative c exists, we may again look for a solution of the form cy with y in J_2. We proceed similarly to $J_3, J_4, \ldots, J_s$. In the end, the element 1 will be a solution if a solution exists, and all the solutions will be described by the subgroup $H = J_s$.

This gives a polynomial time algorithm, provided we have a polynomial membership test for each H_j, and we also have n, r, $|G_i|$, and $|J_j|/|J_{j-1}|$ all polynomially bounded. To enforce this last condition, for each coset b_jH_j, we also include larger cosets b_jH_{ij}, where $H_{ij} = G_{[i]}H_j$ and $G_{[i]}$ consists of those elements of G whose projection into $G_{[n]\setminus[i]}$ equals 1, for $0 \leq i \leq n$. Note that $H_{0j} = H_j$, $H_{nj} = G$, and $|H_{ij}|/|H_{(i-1)j}|$ is polynomially bounded. A membership test for H_j can be obtained from the generators of H_j by considering a chain of subgroups H'_{ij} as before, where $H'_{0j} = H_j$ and H'_{ij} consists of the elements of $H'_{(i-1)j}$ whose projection into G_i equals 1. ■

From now on, we assume that we are given $A_i \subseteq G_i$ that are cosets of near subgroups, and a single constraint defined by a coset bH with a subgroup $H \leq G$ given by generators. In fact, we can also assume $b = 1$ so that this constraint is a subgroup $H \leq G$; this is achieved by replacing each A_i with $b_i^{-1}A_i$.

Consider an instance given by a subgroup $H \leq G$ and cosets of near subgroups $A_i \subseteq G_i$. Let k be such that $1 \in A_i$ for all $i < k$. Define the *single element problem* as follows. Replace all A_i for $i > k$ with $A'_i = G_i$. Replace A_k with $\{a_k\}$, where $a_k \in A_k$.

Suppose we can solve the single element problem in polynomial time. Then we can try all $a_k \in A_k$ in succession for the single element problem. If a solution exists, we will succeed for some such a_k. Once a solution is found, we can transform the problem so that the solution is 1, as we did before when we replaced bH with H. We can then move on to the problem for $k' = k+1$. If $k' = n+1$, then we are done. Thus the problem with cosets of near subgroups is polynomially reducible to the single element problem.

Let the *odd order problem* be the single element problem for a_k of odd order. Let the *2-element problem* be the single element problem for a_k of order a power of two. Let the *involution problem* be the single element problem for a_k of order two. The following is as in [10].

Theorem 5 *(1) The group constraint satifaction problem with subgroups, twisted subgroups, and their cosets reduces to the odd order problem and the involution problem.*

(2) If the odd order problem has a solution, then it has a solution of odd order.

(3) If the involution problem has a solution, then it has a solution of order a power of two.

(4) The odd order problem is polynomial even for twisted subgroups (that are not necessarily near subgroups).

Proof: We start with the single element problem for a_k of order $(2l-1)2^r$, and reduce it to the odd order problem for $a_k^{2^r}$ and the 2-element problem for a_k^{2l-1}. If x is a solution for a_k, then x^{2^r} is a solution for $a_k^{2^r}$ and x^{2l-1} is a solution for a_k^{2l-1}. For the converse, note that 1 is a solution for $a_k^0 = 1$. Furthermore, if x is a solution for a_k^s and y is a solution for a_k^{s+t}, then $yx^{-1}y$ is a solution for a_k^{s+2t}, by the definition of twisted subgroups and (1) from Lemma 1. This process gives solutions for all of $\langle a_k \rangle$.

We then reduce the 2-element problem for a_k to the involution problem for some b with $b^2 = 1$. Let $G_{n+1} = \langle b \rangle$, and introduce a new subgroup $H' \leq G_k \times G_{n+1}$ with $H' = \langle (a_k, b) \rangle$. A solution x to the involution problem for b is a solution to the 2-element problem for some a_k^{2l-1}, and (1) follows from the fact that $\langle a_k^{2l-1} \rangle = \langle a_k \rangle$.

For (2), note that if a_k is of order $2l-1$, then $a_k^{2l} = a_k$, so we can repeatedly replace a solution x with $x' = x^{2l}$ until a solution of odd order is obtained. For (3), if x is a solution to the involution problem for b, with x of order $(2l-1)2^r$, then we can just take $x' = x^{2l-1}$ since $b^{2l-1} = b$.

We finally prove (4). We use Lemma 2 and define a problem in $G \times G$ where $H \times H$ is the subgroup constraint corresponding to the single subgroup $H \leq G$. For a twisted subgroup A_i of G_i, we use the subgroup $H_{A_i} \leq G_i \times G_i$. For the element a_k of order $2l-1$, we use $(a_k^l, a_k^{-l}) \in G_k \times G_k$. We solve the problem in $G \times G$ with subgroups and a single coset by Theorem 4; the fact that this solves the original problem with twisted subgroups follows from (2) of this theorem and Lemma 2. ∎

Corollary 1 *The problem with a subgroup $H \leq G$ given by generators and cosets of twisted subgroups, for a group of odd order, can be solved in polynomial time.*

Corollary 2 *The problem with a subgroup $H \leq G$ given by generators and cosets of near subgroups reduces to the involution problem.*

4 Solvable groups

We begin by describing the basic approach that is common to all algorithms, whether the groups G_i are solvable or not. By Corollary 2, it is sufficient to give an algorithm for the involution problem.

An instance of the involution problem consists of $G = G_1 \times \cdots \times G_n$, where $G_n = \langle b \rangle$ with $b^2 = 1$, a subgroup $H \leq G$ given by generators, and near subgroups $A_i \subseteq G_i$ for all i. The aim is to find $x \in H$ such that $x_i \in A_i$ for $i < n$, and $x_n = b$.

If A_i is a near subgroup of G_i, then $B_i = A_i \times G_{[n]-i}$ is a near subgroup of G, by (5) of Lemma 1. Let K be the intersection of all B_i. Then $K = A_1 \times \cdots \times A_n$ is a near subgroup of G by Theorem 1. The aim in the involution problem is to find $x \in K \cap H$ with $x_n = b$.

By (3) of Theorem 5, it is sufficient to look for solutions in the set S of 2-elements. Say that an instance of the involution problem is in *proper form* if $G_i = \langle S \cap A_i \rangle$ for all $i < n$, and the projection of H into each G_i is G_i.

It is easy to put an instance in proper form. If $G_i' = \langle S \cap A_i \rangle$ is smaller than G_i, then replace G_i with G_i' and restrict H accordingly by intersecting it with $G_i' \times G_{[n]-i}$, using Theorem 4. The generators for H give generators for the projection H_i of H into G_i, and we can use these generators to find all elements of H_i. If H_i is smaller than G_i, then again replace G_i with H_i and repeat the process.

Suppose the involution problem is in proper form. Obtain the normal subgroup N_{A_i} of G_i so that $\hat{G}_i = G_i/N_{A_i}$ is $\hat{A}_i$-reduced. Let $\hat{G} = G/N = \hat{G}_1 \times \cdots \times \hat{G}_n$. We can the consider the image subgroup $\hat{H}$ of the subgroup H, and thus we have reduced the problem to an instance in $\hat{G}$, since $a_i N_{A_i} \subseteq A_i$ for each $a_i \in A_i$ by (1) of Lemma 3.

We assume from now on that the instance of the involution problem is in proper form and each G_i is A_i-reduced.

We define $G_i^* = G_i/O(G_i)$ and let $G^* = G/O(G) = G_1^* \times \cdots \times G_n^*$. We now show:

Theorem 6 *The involution problem has a polynomial time algorithm for solvable groups.*

By Theorem 2, $(K \cap H)^*$ is an abelian 2-group, since $A_i^* = G_i^*$ is an abelian 2-group. We give two algorithms. The first one uses a direct approach but is more complicated. The second one is simpler and uses the fact that $K = K(\tau)$ for solvable groups from Theorem 2.

For the first algorithm, let $K_{(k)}^*$ denote the abelian 2-group obtain when A_i is replaced by G_i for all $i \geq k$. We have $H^* = K_{(1)}^* \geq K_{(2)}^* \geq \cdots \geq K_{(n)}^* = (K \cap H)^*$. We find generators for the successive $K_{(k)}^*$. In the end we obtain $(K \cap H)^*$ and test whether the projection into $G_n^* = \langle b \rangle$ meets b, by testing each generator. If it does, we have an x^* that corresponds to a solution in $xO(G)$. Such a solution can be found by solving the resulting problem for the group of odd order $O(G)$, using Corollary 1.

Suppose we have found generators up to stage $k-1$, i.e., for $K_{(k-1)}^*$. We can assume that each element of G_i^* corresponds to an element of $K_{(k-1)}^*$, by restricting G_i^* accordingly. Clearly $K_{(k)}^* \leq K_{(k-1)}^*$. Let $J^* \leq G^*$ be the elements of G^* corresponding to $1 \in G_k^*$. For each $x^* \in J^* \cap K_{(k-1)}^*$, the corresponding representative $x \in J \cap K_{(k-1)}$ can be taken to have $x_k = 1$, by considering $x' = x^l$ for some appropriate odd l. Therefore $J_{(k)} = J^* \cap K_{(k)}^* = J^* \cap K_{(k-1)}^*$, and we can find generators for $J_{(k)}$.

For each $b^* \in G_k^*$, there is an $x^* \in K_{(k-1)}^*$ such that $x_k^* = b_k$. Such an x^* can be found from the generators of $K_{(k-1)}^*$. The cosets of $J_{(k)}$ in $K_{(k-1)}^*$ are then the corresponding $x^* J_{(k)}$. There are then two possibilities: either $x^* \in K_{(k)}^*$, in which case $x^* J_{(k)} \leq K_{(k)}^*$ and we add x^* to the generators for $K_{(k)}^*$, or $x^* \notin K_{(k)}^*$, in which case $x^* J_{(k)} \cap K_{(k)}^* = \emptyset$.

This will give all the generators for $K_{(k)}^*$. It remains to show how to determine whether $x^* \in K_{(k)}^*$, i.e., whether $x^* O(G) \cap K_{(k)} \neq \emptyset$. This is a problem with cosets of subgroups and near subgroups on the group $O(G)$ of odd order, and we can apply the algorithm from Corollary 1 to this group. This completes the first algorithm.

For the second algorithm, as well as the algorithm of the next section, we introduce a new notion. We say that an instance of the involution problem when the G_i are A_i-reduced is in *τ-invariant proper form* if it is in proper form and $H^\tau = H$, that is H is τ-invariant. Here τ is the automorphism of G with $\tau^2 = 1$ induced by the automorphisms τ_i for the various G_i, from (2) of Lemma 3. We can always ensure that H is τ-invariant, otherwise replace H with $H \cap H^\tau$, since τ maps each element x_i in the near subgroup A_i to x^{-1}. The next lemma is due to Aschbacher [4].

Lemma 5 *If H is τ-invariant, $s^* \in H^*$ is a 2-element inverted by τ^*, and $K \cap S = K(\tau) \cap S$, then there is a 2-element $r \in sO(G) \cap S \cap H \cap K$.*

Proof: Since $s^* \in H^*$ is a 2-element, there is a $t \in H \cap S$ such that $s^* = t^*$. Note that $\langle s^* \rangle$ is τ^*-invariant with s^* inverted by τ^*. Since H is τ invariant, $U = \langle t(H \cap O(G)) \rangle$ is τ-invariant. By Sylow's theorem, U has a τ-invariant 2-Sylow subgroup, which is of the form $\langle r \rangle$ with $r \in t(H \cap O(G))$. Since s^* is inverted by τ^*, we have that r is inverted by τ, so $r \in K(\tau) \cap S = K \cap S$. ■

From this lemma, it follows that $H^* = (K \cap H)^*$, since H it τ-invariant and all elements of H^* are inverted by τ^*. To finish the algorithm, we just need to test whether the projection of H^* into $G_n^* = \langle b \rangle$ meets b, and once an appropriate element $s^* \in H^*$ is found, obtain a corresponding $r \in sO(G) \cap H \cap K$ by solving a problem in the odd order group $O(G)$ using Corollary 1.

5 General groups

As for the second algorithm of the last section, we assume that $G_i = \langle S \cap A_i \rangle$ is A_i-reduced, and the involution problem is in τ-invariant proper form.

Theorem 7 *There is a polynomial time algorithm that solves the involution problem for general groups.*

Proof: Recall that $G_n = \langle b \rangle$ with $b^2 = 1$. If there is no $x \in H$ with $x_n = b$, then the involution problem does not have a solution. Otherwise, we find a solution as follows. Note that the subgroup R of H of elements x with $x_n = 1$ has index two in H.

Let $G^* = G/O(G) = G_1^* \times \cdots \times G_n^*$, where $G_i^* = G_i/O(G_i)$. Let $\tilde{G} = G^*/Z(G^*) = \tilde{G}_1 \times \cdots \times \tilde{G}_n$, where $\tilde{G}_i = G_i^*/Z(G_i^*)$. It is easy to find these groups in polynomial time, since $z \in O(G_i)$ if and only if $\langle \{z^t = t^{-1}zt : t \in G_i\} \rangle$ is of odd order.

Consider $\tilde{H}$ and $\tilde{R}$. We shall show that $\tilde{H}$ does not have two normal subgroups N, M with $N \leq M \leq \tilde{H}$ such that N has index two in M. In particular, $\tilde{H}$ does not have a subgroup of index two, and therefore $\tilde{R} = \tilde{H}$.

It follows that $R^* \cap Z(G^*)$ has index two in $H^* \cap Z(G^*)$. Find any element $s^* \in H^* \cap Z(G^*)$ with $s_n^* = b$. Let $K = A_1 \times \cdots \times A_n$. By (2) of Theorem 3 applied to the A_i, we have $s^* \in K^* = K(\tau^*)$. By (1) of Theorem 3 applied to the A_i, we have $K = K(\tau)$. Therefore, by Lemma 5, there is an element $r \in sO(G) \cap S \cap H \cap K$. Obtain such an r by solving a problem on the odd order group $O(G)$, using Corollary 1. This completes the algorithm.

We show that $\tilde{H} \leq \tilde{H}_1 \times \cdots \times \tilde{H}_n$ does not have normal subgroups N, M with $N \leq M \leq \tilde{H}$ and N of index two in M. The proof is by induction. The base case for $\tilde{H}_i = \tilde{G}_i$ follows from (3) of Theorem 3. For the inductive step, we take $\tilde{H} \leq \tilde{H}_1 \times \tilde{H}'$, where $\tilde{H}'$ is the projection of $\tilde{H}$ into $\tilde{H}_2 \times \cdots \times \tilde{H}_n$. Suppose $\tilde{H}$ has such N, M. If the projection N_1 has index two in M_1 inside $\tilde{H}_1$ then we are done by the base case above. Otherwise $N_1 = M_1$, and we consider $1 \in N_1 = M_1$; the subgroup of elements $(1, x') \in N$ is then a subgoup of index two in the subgroup of elements $(1, x') \in M$, thus giving a normal N' of index two in a normal M' inside $\tilde{H}'$, completing the inductive step and the proof. ■

6 Near subgroups are Mal'tsev

Let G be a finite domain. A ternary function f on G is said to be *Mal'tsev* if it satisfies $f(x,y,y) = f(y,y,x) = x$. A subset $R \subseteq G^k$ is *f-closed* if for all $x = (x_1,\ldots,x_k)$, $y = (y_1,\ldots,y_k)$, $z = (z_1,\ldots,z_k)$ in R, the tuple $(f(x_1,y_1,z_1),\ldots,f(x_k,y_k,z_k))$ is also in R. In particular, cosets of subgroups have the Mal'tsev operation $f(x,y,z) = xy^{-1}z$.

Bulatov has recently shown that constraint satisfaction problems on a finite domain G, whose constraints are f-closed for a Mal'tsev function f on G, have a polynomial time algorithm [7, 8]. The following shows that this should yield alternative polynomial time algorithms for the problems solved in this paper, provided that the results of Bulatov can be extended to the cases of possibly different domains of arbitrary sizes with corresponding closure functions for different variables, and constraints of possibly exponential size involving all variables given by a smaller set of generators, which remains open in cases other than those studied in this paper. The two problems here that are special cases of the result on Malt'sev functions, when the domain is fixed and has a single closure function, are the case of near subgroups and their cosets (shown in Theorem 7 in the more general context), and the case of twisted subgroups plus constants of odd order (the odd order problem of Theorem 5(2), 5(4) in the more general context).

Theorem 8 *(1) Every finite group G has a Mal'tsev function f such that subgroups, near subgroups, and their cosets in G^k are f-closed.*

(2) There exists a Mal'tsev function f such that the odd order elements of any twisted subgroup of G^k for the odd order problem form an f-closed subset.

Proof: We first define a binary function g on G. Let r be the order of G, and consider the direct product $G^{r^2} = G_1 \times G_2 \times \cdots \times G_{r^2}$, where each G_i is isomorphic to G. Choose two elements $x = (x_1, x_2, \ldots, x_{r^2})$, $y = (y_1, y_2, \ldots, y_{r^2})$ of G^{r^2} such that each pair of elements from G occurs as a pair (x_i, y_i) in one of the G_i.

Let K be the smallest near subgroup of G^{r^2} containing both x and y, which we know exists since the intersection of near subgroups is a near subgroup. By the definition of near subgroup, there exists an element z in the commutator group N of the group generated by x, y such that xyz is in K, since $\langle x,y\rangle/N$ is abelian.

We define $g(x_i,y_i) = x_i y_i z_i$. Note that $g(x_i,1) = x_i$ and $g(1,y_i) = y_i$, since all z in the commutator group N have $z_i = 1$ in these cases. We define now $f(ax_i, a, ay_i) = ag(x_i,y_i)$. We then have $f(ax_i,a,a) = ax_i$ and $f(a,a,ay_i) = ay_i$, so f is indeed Mal'tsev, proving part (1).

For part (2), if x is of odd order, let $\sqrt{x}$ denote the element $u \in \langle x\rangle$ with $u^2 = 1$. If G has order ot with o odd and t a power of 2, choose q, r such that $qo + rt = -1$, and let $l = qo + 1 = rt$. Define, for x, y, z of odd order, $f(x,y,z) = (\sqrt{x}\sqrt{y^{-1}}z\sqrt{y^{-1}}\sqrt{x})^l$ of odd order. Then $f(x,y,y) = f(y,y,x) = x^l = x$, so f is Mal'tsev. ■

References

[1] F. Afrati, S. S. Cosmadakis, and M. Yannakakis. “On Datalog vs. polynomial time”. Proceedings of the 10th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (1991) 13–25.

[2] M. Aschbacher. “Finite group theory”. Cambridge University Press (1986).

[3] M. Aschbacher. “Near subgroups of finite groups”. Journal of Group Theory 1 (1998) 113–119.

[4] M. Aschbacher. Personal communication.

[5] M. Aschbacher. Manuscript.

[6] M. Aschbacher. Manuscript 2.

[7] A. A. Bulatov. “Tractable constraint satisfaction problems on a three-element set”. Electronic Colloquium on Computational Complexity Report TR02-032 (2002).

[8] A. A. Bulatov. “Mal’tsev constraints are tractable”. Electronic Colloquium on Computational Complexity Report TR02-034 (2002).

[9] T. Feder, “Strong near subgroups and left gyrogroups”. Journal of Algebra, to appear.

[10] T. Feder and M. Vardi. “The computational structure of monotone monadic SNP and constraint satisfaction: A study through Datalog and group theory”. SIAM Journal on Computing 28 (1998) 57–104.

[11] T. Foguel and A.A. Ungar. “Involutory decomposition of groups into twisted subgroups and subgroups”. J. Group Theory 3 (2000) 27–46.

[12] M. Furst, J. E. Hopcroft, and E. Luks. “Polynomial time algorithms for permutation groups”. Proceedings of the 21st IEEE Symposium on Foundations of Computer Science (1980) 36–41.

[13] G. Glauberman. “On loops of odd order ”. Journal of Algebra 1 (1964) 374–396.

[14] G. Glauberman. “On loops of odd order II”. Journal of Algebra 8 (1968) 393–414.

[15] C. M. Hoffman. “Group-theoretic algorithms and graph isomorphism”. Lecture Notes in Computer Science 136, Springer-Verlag, New York (1982).

[16] A. A. Razborov. “Lower bounds on monotone complexity of the logical permanent”. Mathematical Notes of the Academy of Science of the USSR 37 (1995) 485–493.

[17] T. J. Schaefer. “The complexity of satisfiability problems”. Proceedings of the 10th ACM Symposium on Theory of Computing (1978) 216–226.

[18] A.A. Ungar. “Thomas precession: its underlying gyrogroup axioms and their use in hyperbolic geometry and relativistic physics”. Found. Phys. 27 (1997) 881–951.

[19] A.A. Ungar. “Beyond the Einstein Addition Law and its Gyroscopic Thomas Precession: The Theory of Gyrogroups and Gyrovector Spaces”. Dordrecht, Boston, London: Kluwer Acad. Publ. (2001).

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'