Reg. No: 2022/45/B/ST6/02229; Principal Investigator: dr Jacek Krzaczkowski

SATISFIABILITY AND EQUIVALENCE PROBLEMS FOR FINITE ALGEBRAS

JACEK KRZACZKOWSKI

From the ancient times solving equations is one of the most important problems in mathematics and has been considered from the very beginning of what we now know as computer science. One of the most notably example is the famous Hilbert's 10th problem i.e. the question if there exists an algorithm determining if an equation of polynomials with integer coefficients has a solution with all unknowns taking integer values or not. The problem was solved by Y. Matiyasevich in 1970. It turned out that there is no algorithm checking if a given polynomial equation has a solution in integers. Nowadays, the hardness of the equations solving over finite algebraic structures ensures the security of public key cryptography. In the contrast to the Matiyasevich's results determining if the given equation over a fixed finite algebraic structure has a solution can be done by algorithm. It is enough to check all possible solutions (values of unknowns). Such the primitive algorithm to solve an equation with n unknowns over k-element structure need number of steps proportional to k^n. On the other hand, for some structures we can solve a given equation much faster e.g. if the only operation in a given equation is addition modulo some prime number p, then it is easy to determine if there is some solution from the set $\{0, 1, \ldots, p-1\}$ (in this case all what the algorithm has to do is to reduce similar terms and check if the equation after such the simplification is not trivially unsatisfiable).

The main goal of the project is to characterize for which algebraic structures several computational problems connected with solving equations can be solved by an efficient algorithm (i.e. in a so called polynomial time). The considered problems are of three types. For a given equation over some fixed algebraic structure we ask if:

- there exists a solution to the given equation,
- there exists a solution to the given equation such that every unknown has value from a given list (possibly different for different unknowns),
- equation holds for all values of unknowns.

In mathematics polynomials are the usual way of describing operations of algebraic structures. In many branches of computer science we are used to use in such a role circuits i.e. the simple computational model in which input values proceed through a sequence of gates, each of which computes a function from some fixed set. For every kind of problem listed above we consider two versions. One in which we consider equations of polynomials and the second one in which we consider equations of circuits. Circuits often enable describing functions in much more compact way than polynomials. It is because an output of one gate can be used as an input of many other gates (in case of polynomials to use value of some expression multiply times we have to put this expression in every place in which we want to use its value). Thus, solving equations of circuits (and other similar problems) can be harder than the same problem for equations of polynomials.

Automata and languages recognized by them are among widely considered problems of theoretical computer science. We are interested in Non-uniform deterministic finite automata (NUDFA). NUDFA can be defined by the following three elements: finite algebraic structures (originally some semigroup), a term/polynomial over this structures and so called program. We consider the following question:

- What is the computational complexity of the problem in which we ask if a language recognized by a given NUDFA over fixed algebraic structure is not empty?

It turns out that checking if a language recognized by a given NUDFA is not empty is closely related to the equation satisfiability problem and we plan to consider this two problems together.

We will consider the problems mentioned above for different classes of algebraic structures. We will start our investigation with finite groups. After that, we are going to consider algebraic structures from so called congruence modular varieties, a big class of nicely behaving algebraic structures containing among others groups, rings, lattices and most of popular structures with an important exception of some of semigroups. Finally, we will go outside congruence modular varieties and we will see what we are able to say about computational complexity of considered problems in a general case.