Popular science abstract

We study questions related to number sequences and power series arising in different fields of science, such as computing, logic, combinatorics, and algebra. The most popular example is the *Fibonacci sequence*:

$$f_{n+1} = f_n - f_{n-1}, \quad f_0 = 0, f_1 = 1. \tag{1}$$

Given two such sequences f_n and g_n can we decide (with an algorithm / program) whether $f_n = g_n$ for all $n \in \mathbb{N}$? This is called the *equivalence problem* and it is one of the central topics of this project. It has applications in automata theory (do two automata recognise the same language?), control theory (are the trajectories of two dynamical systems the same?), and in computer algebra (do two complicated expressions represent the same quantity?).

Another problem which is important for us is that of *invariants*. For instance the sequence $f_n = (-1)^n$ satisfies the invariant $f_n^2 = 1$. Can we find (a finite representation of) *all* the invariants satisfied by sequences like these? (Invariants generalise equivalence since $f_n = g_n$ is an invariant.) Invariants are central in physics since they provide deep insights into the system, often in the form of *conservation laws*: quantities that do not change during the computation can have deep physical significance.

There are many ways in which number sequences can be described. We have shown the simplest way where the next value f_{n+1} is defined according to some rule depending on the previous ones f_n, f_{n-1}, etc... By changing the form of the dependency we can represent sequences which are very significant, e.g., in combinatorics, such as the *Catalan numbers* $C_{n+1} = C_nC_0 + C_{n-1}C_1 + \cdots + C_nC_0$, the *Bell numbers* $B_{n+1} = \binom{n}{0}B_0 + \binom{n}{1}B_1 + \cdots + \binom{n}{n}B_n$, and many others. Once we have different classes of sequences, we can ask questions such as: Is every sequence defined like Fibonacci also definable like Catalan (yes)? And conversely (no)? In general, this is an instance of the *membership problem*, which asks whether a given sequence can also defined in some other way. This problem is important since being able to solve it algorithmically entails a deeper understanding of what is necessary to recursively define a sequence.

Number sequences arise also in logic. For instance, a formula of logic φ can describe properties of finite structures, such as "the structure consists of a sequence of elements and their positions are related by an equivalence relation", and from φ we can construct a corresponding sequence f_n^φ (*Specker sequence*) which counts structures of n elements satisfying φ. We can now ask whether $f_n^\varphi = B_n$ (yes), which is a very nontrivial question in general. In fact, equivalence for Specker sequences is in general *undecidable* (i.e., there is no algorithm that can solve this problem), so in the project we will focus on restrictions under which this obstacle can be avoided.

Number sequences f_n can be generalised by *weighted languages* f_w, where now w is a string of symbols such as $w = aaba$. Weighted languages arise as the meaning of computational models such as weighted automata (generalising Fibonacci), grammars (generalising Catalan), and others. The problems of equivalence, invariants, membership, and other analyses all generalise to weighted languages. Sometimes, they give rise to hard, long-standing open problems.

We are not only interested in whether (or not) a given problem is decidable but also what is its *computational complexity*, i.e., the resources necessary (for the problem) and sufficient (for an algorithm) in terms of the elapsed time and utilised space.

Summarising, this is a project of chiefly theoretical nature. Our general objective is to understand the decidability and computational complexity of the equivalence problem for classes of number sequences and power series arising in computer science, logic, combinatorics, physics, dynamical systems, and algebra. This gives rise to many open problems which we plan to investigate.