The project concerns higher-order recursion schemes. It is a formalism modeling programs that use higher-order recursion, that is, those in which procedures may take other procedures as parameters. Given a recursion scheme, we can talk about the tree generated by the scheme; particular branches of the tree describe all possible runs of our program.

Roughly speaking, our main objective is to develop mathematical tools allowing estimation of some quantities in trees generated by recursion schemes. Using these tools, we aim in achieving the following objectives of the research:

1. We want to obtain algorithms that take a recursion scheme and a description of some property (given, for example, as a logical formula) and check whether the tree generated by the scheme satisfies the given property. We are interested here in properties that talk about boundedness or unboundedness of some quantities; for example: "there is a common bound on the number of events x on every branch of the tree"; we thus want to consider logics that allow to express properties of this kind. On the one hand, we expect algorithms for as expressible formalisms (logics) as possible, allowing to supply as complicated properties as possible; these algorithms, however, will be probably very slow. On the other hand, we want to propose appropriately restricted formalisms, perhaps less expressive, for which it will be possible to develop faster algorithms—working fast enough for using them in practice.

2. We also want to create a new algorithm computing the downward closure of the language recognized by a recursion scheme (and algorithms solving similar problems). The downward closure is obtained by considering all words formed by removing letters from words of the original language. Such downward closure is always a regular language and can therefore be described by a finite automaton, which is a much simpler object than a recursion scheme. Thus, by calculating the downward closure of the language recognized by a recursion scheme we get an approximation of the recursion scheme by a simpler object, which can be then easily analyzed. It turns out that a key difficulty in this issue is the ability to estimate whether certain events may occur unboundedly many times, or not.

3. Additionally, we want to better understand theoretical properties of the classes of languages recognized by higher-order recursion schemes. In particular, we want: to obtain a pumping lemma for these languages; to say whether they are context-sensitive; to say whether every recursion scheme is equivalent to a so-called safe recursion scheme.

4. As a side effect, we also want to develop formalisms talking about boundedness of some quantities, without direct relation to recursion schemes.

Research carried out in the project will be of theoretical nature, it will mainly consist of inventing and proving theorems. In addition, we also want to write programs that implement some of our algorithms.

The subject of the proposed research has close relations with algorithms for automatic software verification; the point is that, having a program and a description of its desired properties, one wishes to determine whether the program satisfies these properties. In the strict sense, this problem cannot be solved by computers; it is even impossible to say whether a given program will terminate (this is an undecidable problem). We can, however, ignore some specific calculations performed by the program, and values held in its variables, and examine only the overall structure of the program. Most often one verifies in this way only a single procedure, in which there is no recursion; such procedures can be modeled using finite automata (finite-state systems). However, many programs are written in a recursive manner, and most modern programming languages allow the use of higher-order recursion. As already mentioned, recursion schemes are simplified models of programs using higher-order recursion. As a result of our research we want to widen the class of allowed properties, for which this type of automatic verification is possible, to properties saying that a quantity is bounded or unbounded.

Simultaneously, a large part of the proposed research has a theoretical character, and is not directly motivated by practical issues. We just consider a natural notion, like higher-order recursion schemes, and we want to discover as much as possible about its properties. Questions and issues that we plan to investigate, are important open problems in automata theory; some of them are mentioned in numerous publications.