Proposal for M2 Internship for MPRI for 2020

**Complexity theory with Discrete Differential Equations/Finite differences.**

Olivier Bournez

**Keywords:** Complexity Theory. Implicit Complexity. (Discrete) Differential Equations.

**Supervisor:** Olivier Bournez.

**Administrative Position:** Professor (of Computer Science) at Ecole Polytechnique.

**Location:** Laboratoire d’Informatique de l’X, LIX, Ecole Polytechnique, 91128 Palaiseau Cedex

**Phone:** +33 (0)1 77 57 80 78

**Email:** olivier.bournez@lix.polytechnique.fr.

**Www:** http://www.lix.polytechnique.fr/~bournez.

**Language:** French or English. This proposal is intentionally written in English for Non-French speaking students who may be interested.

**General Presentation**

Ordinary Differential Equations (ODEs) appear to be a natural way of expressing properties and are intensively used, in particular in applied sciences. The theory of classical (continuous) ODEs has an abundant literature (see e.g. [1, 4]) and is rather well understood under many aspects. We are mainly interested here in a discrete counterpart of classical continuous ODEs: discrete ODEs. Its analogue of the notion of derivative, called finite differences, has been widely studied in numerical optimization for function approximation [5]. It is reminiscent of discrete calculus [7, 6, 9, 10] in combinatorial analysis. Historically, similarities between discrete and continuous statements have also been observed as early as in the 19th century, under the terminology of umbral or symbolic calculus. However, even if the underlying computational content of finite differences theory is clear and has been pointed out many times, no fundamental connections with algorithms and complexity have been exhibited so far.

We recently started to develop a systematic study of discrete differential equations under a computational perspective. In a recent article [3], we demonstrated that discrete ODEs is a very natural tool for algorithm design and for proving that complexity and computability notions can be elegantly and simply captured using discrete ordinary differential equations. We illustrated this by providing a characterization of \( \text{FPTIME} \), the class of polynomial time computable functions, and of its non deterministic analog \( \text{FNP} \). To this aim, we demonstrated how some notions from the analog world such as linearity of differential equations or derivatives along some particular functions (i.e. changes of variables) are representative of a certain computational hardness and can be used to solve efficiently some (classical, digital) problems.

**Formally:** as an example, we have established that differential equations of the form:

\[
\frac{\partial f(x, y)}{\partial \log_2(x)} = h(f(x, y), x, y)
\]

where \( h \) is linear and the derivative is taken along the length of the input (hence the derivative with respect to \( \log_2(x) \)) corresponds to polynomial time: For solutions \( f \) of such type of equations it can be guaranteed that both the number of steps necessary to compute \( f(x, y) \) and the growth of \( f \) can be kept polynomial. And conversely, any polynomial time function can be shown to be a solution of such a system.

There are strong evidences that this approach can be extended to characterize other complexity measures and exhibit new connections with classical notions of analysis.

This provides a way to reinterpret classical notions with these new eyes: For example, the Master Theorem (which explains for example the complexity of a recursive algorithm) at the basis of algorithm design can be basically read as a result on (the growth of) a particular class of discrete time length ODEs.

The purpose of the current project is to develop this new perspective and to revisit logic, computability theory, complexity theory and even algorithm design from this original point of view.
Description of the work

The purpose of the work will be to provide a characterization of the main computational complexity classes using discrete ordinary differential equations: We characterized PTIME (polynomial time) in [3]. However, the approach seems flexible enough to capture a wide range of algorithmic classes by restricting time or space (e.g. NP (non-deterministic polynomial time), PSPACE (polynomial space)). Another objective is to capture parallel classes and also probabilistic classes.

One orthogonal aim can be to generalize the settings to continuous or more general dynamics: Our characterization is deeply based on encoding of configurations by integers. We propose to extend the framework to deal with more general structures (e.g. lists, stacks, arrays), using approaches based on logic. For example, using the approach of evolving algebra and abstract state machines [8]. This can in particular extend [2] where a semantic for ODEs was given.

Comments

The true topic of the work is related to complexity theory, and logic (implicit complexity). This requires only common and basic knowledge in ordinary differential equations. Most of the intuitions of our today’s constructions come from classical computability and complexity or logic.

There is no specific prerequisite for this internship. This subject can be extended to a PhD. Possibilities of funding according to the administrative situation of candidates.

References