Remarks on a proof-theoretic characterization of polynomial space functions

(October 2007)

Abstract

We characterize the polynomial space operations as those which are provably total in a first order ramified system which comprises full untyped combinatory logic with extensionality, together with a principle of positive ramified induction with parameter substitution on the set $W$ of binary strings.

Parallel computations have been studied for long as a tool for classifying natural collections of sub-recursive functions: it is well known, for instance, that the class of functions computable in deterministic polynomial space coincides with the set of languages decidable in parallel alternating polynomial time.

Several machine-independent characterizations of complexity classes have been developed in the field of the so called Implicit Computational Complexity Theory, by introducing concepts like ramification and data tiering (see, for instance, [L91], [L94], [BC92], [O97], [O07]). The use of ramified data links computational complexity to levels of definitional abstraction and clarifies the correspondence between sub-recursion and complexity, by requiring that recurrence principles respect the separation between data objects which are used computationally in different guises.

Ramified recurrence with parameter substitution was introduced by Leivant and Marion in 1995 ([LM95]) as a quite general variant of ramified recurrence, where the parameters of a recursive call may be altered at each iteration using previously defined functions, thereby enabling the simulation of parallel alternating polytime computing, and thus of polyspace.

We give here a proof-theoretic and applicative characterization of poly-space operations, generalizing the result of Cantini ([C02]), which were proven for polynomial time functions. Applicative systems ([C00], [C02], [ST03]) provide a very natural framework for a proof-theoretic approach to computational complexity: all objects may be regarded as operations or rules, in the sense of combinatory logic, together with binary strings. We also assume a many-sorted structure with copies $W_0, W_1, W_2, ...$ of the algebra $W = \{0, 1\}^*$ of binary strings as our tiered universes. These ramification conditions impose a strictly predicative regime, which distinguishes between different uses of variables in induction schemas. In addition, parameter substitution allows the representation of parallel computing, by giving a tree-structure to the usual recursion on notations scheme, and leading to a branching of the computation flow.

We are concerned with a first order applicative theory of operations based on standard combinatory logic: Turing-complete untyped languages are used, while...
the intended models are just combinatory algebras extended with the type \( W \)
of binary words.

We show that polyspace functions can be extracted from quasi cut-free derivations by means of a non-standard realizability argument due to [L94] and [C02].

**References**


**C00** Cantini, A.: Feasible operations and applicative theories based on \( \lambda \eta \), *Mathematical Logic Quarterly* 46, 291-312 (2000).


