# Executing Extracted Programs

## FTA vs. Extraction : Round II

**Luís Cruz-Filipe**[1]
**Pierre Letouzey**[2]

[1] Center for Logic and Computation, IST Lisbon
[2] Mathematik Institut, LMU Munich

## FTA : Fundamental Theorem of Algebra

"All non-constant polynomial over $\mathbb{C}$ has at least one root"

One impressive development (40 000 lines) at Nijmegen by Geuvers, Barendregt, Wiedijk, Pollack, Zwanenburg & Niqui.

Constructive proof $\Rightarrow$ a effective root search method

But extraction wasn't planned at first.

$\Rightarrow$ Huge efforts in order to obtain a program

## Recall : FTA vs. Extraction, Round I

Cf. last TYPES talk and TPHOLs03 paper by C.F. & Spitters :

As result, a reasonable extracted source code (250kb)

But after :

- tweaking the logic of FTA : `Prop` vs. `Set`
- improving several proofs of FTA : real model, division, ...
- removing one bug in extraction : abusive constant unfolding

$\Rightarrow$ Already quite an achievement

## New : FTA vs. Extraction, Round II

Is this program usable ? No ! Not yet ?

The whole FTA program is out of understanding.

$\Rightarrow$ Study of easier subproblems :

- computations of series limits, especially $e = \sum_0^\infty \frac{1}{n!}$
- computations via the IVT, in particular $\sqrt{2}$

$\Rightarrow$ No complete solutions yet, but some progress / new clues

## Euler $e$ : first tests & improvements

Unary/binary number inefficiencies :

- in the representation of rational numbers :

  At first, $\mathbb{Q} = \mathbb{Z} \times \mathbb{N}$

  Now, $\mathbb{Q} = \mathbb{Z} \times \mathbb{N}^*$

- in the generation of real numbers :

  $$\texttt{nring} : \mathbb{N} \rightarrow \mathbb{R}$$

  vs.

  $$\texttt{zring} : \mathbb{Z} \rightarrow \mathbb{R}$$

- in the structure of proofs : at first, $1/n!$ was using

  $$\forall k : \mathbb{N}, k > 0 \rightarrow \texttt{nring}\ k \neq O$$

  Better :

  $$\forall z : \mathbb{Z}, z > 0 \rightarrow \texttt{zring}\ z \neq O$$

## Euler $e$ : why bother ?

In fact, the problems may look rather artificial :
usually, it's obvious to inject $\mathbb{Q} \rightarrow \mathbb{R}$.

But FTA is split in two parts :

- the mathematical proofs where assumptions over $\mathbb{R}$ are minimalist ($0$, $1$, $+$, ... plus a few axioms)
- a model based on Cauchy sequence implementing this minimalist interface

Modularity dilemma : rich or poor interface ?

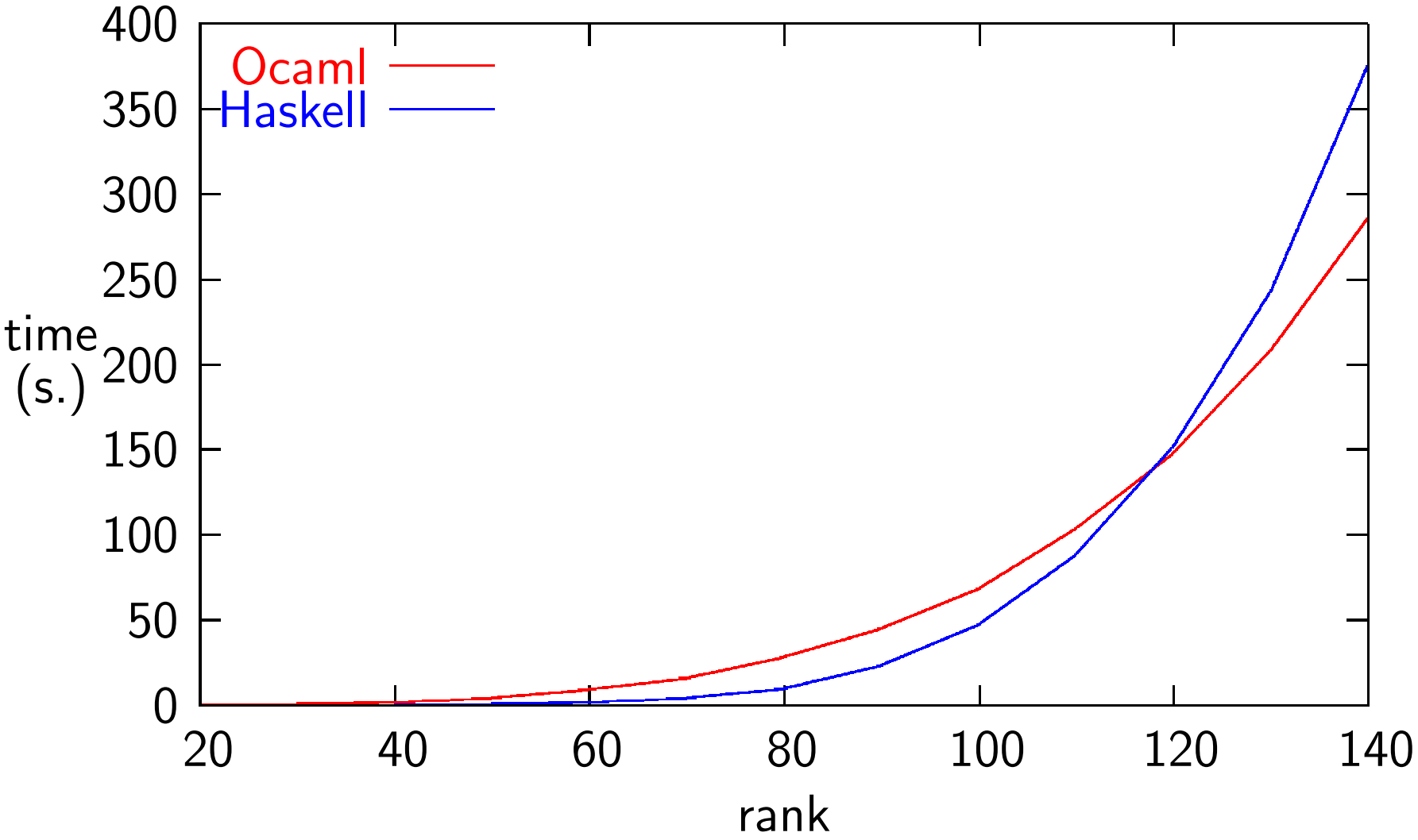
## Euler $e$ : the crucial change

We have finally shifted in the model a few critical functions, in order to access definitions of reals, of $\neq$, etc.

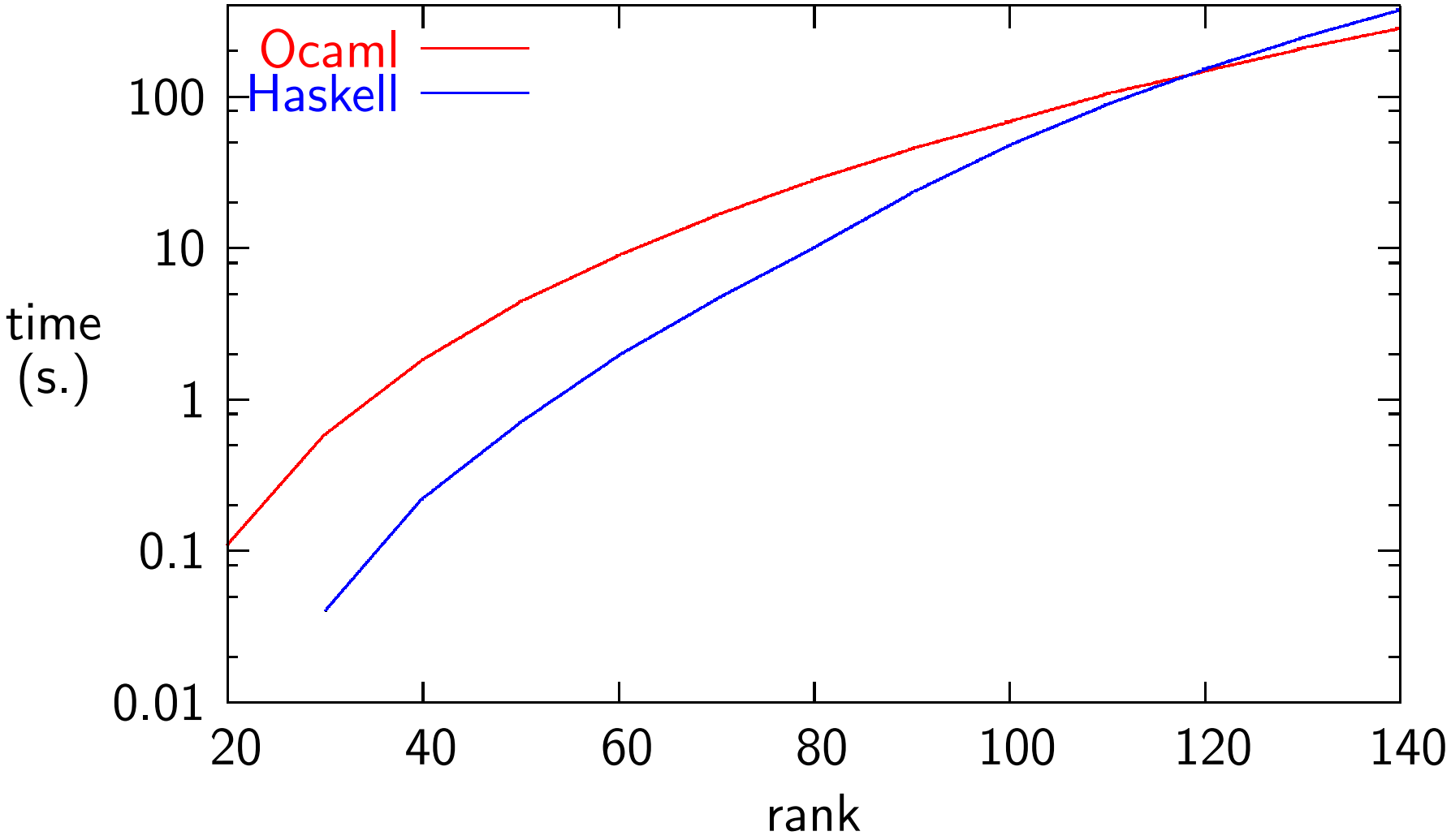
To use these optimized functions in the abstract part : axioms.

The extraction maps these axioms to the concrete functions.

## Euler $e$ : some statistics



## Euler $e$ : some statistics



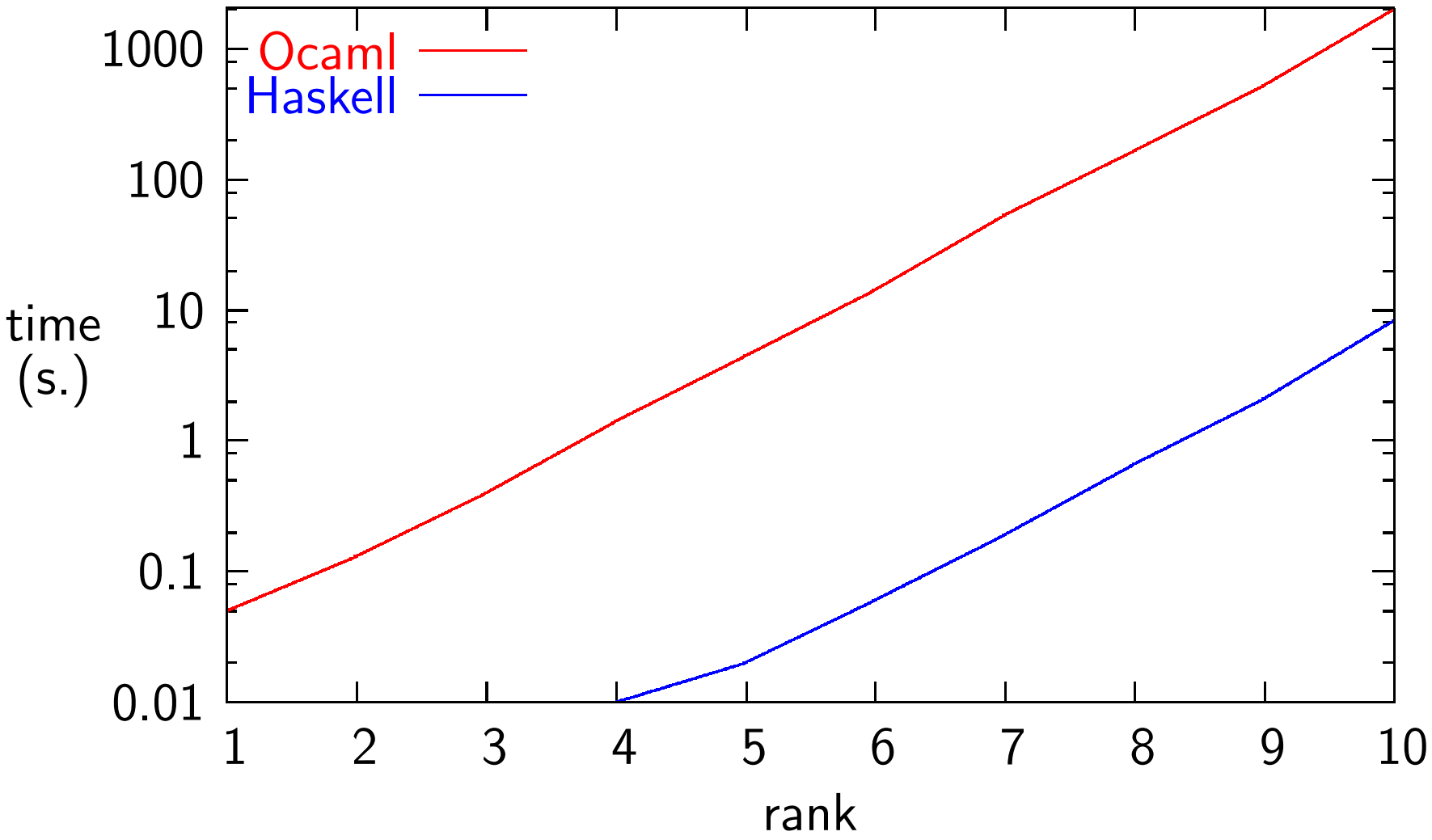
## $\sqrt{2}$

The next stage in difficulty : real functions & IVT.

- A generic but awfully inefficient proof that every non-constant polynomial is ... locally non-constant, as required by the IVT. Here, instead, a simple monotonicity argument for $X^2 - 2$.
- As for Euler's $e$, some functions moved into the real number model.

But...

# $\sqrt{2}$ : some statistics



## A new alternative $\sqrt{2}$ development

Small is ... dirty, but quick !

A short development inspired by H. Schwichtenberg : 400 lines of Coq, and 40 digits of $\sqrt{2}$ in 2 min !

Dirty :

- not finished ("sorry" !)
- really ad hoc. for example, only continuity of $X^2 - 2$.

But quite rich in lessons.

Could this approach be extended ? up to FTA ? ?

## FTA : summary of current state

- A reasonable-sized extracted program, that compiles
- Root search : unusable
- Some encouraging intermediate results : series
- Can we extend up to the whole FTA ? Or would it be quicker to start anew ?

## Conclusion : the quest for good proofs

What is a good proof ?

- efficient w.r.t extraction ?
- elegant : abstract / generic / short ... ?
- ... or just finished ?

The extraction is definitively an interesting method, but not a magic button.