# **Elaboration with First-Class Implicit Function Types**

## ANONYMOUS AUTHOR(S)

Implicit functions are dependently typed functions, such that arguments are provided (by default) by inference machinery instead of programmers of the surface language. Implicit functions in Agda are an archetypal example. In the Haskell language as implemented by the Glasgow Haskell Compiler (GHC), polymorphic types are another example. Implicit function types are *first-class* if they are treated as any other type in the surface language. This is true in Agda and partially true in GHC. Inference and elaboration in the presence of first-class implicit functions poses a challenge; in the context of GHC and ML-like languages, this has been dubbed "impredicative instantiation" or "impredicative inference". We propose a new framework for elaborating first-class implicit functions, which is applicable for full dependent type theories and compares favorably to prior solutions in terms of power, generality and conceptual simplicity. We build atop Norell's bidirectional elaboration algorithm for Agda, and note that key issue is incomplete information about insertions of implicit abstractions and applications. We make it possible to track and refine information related to such insertions, by adding a new function type to a core Martin-Löf type theory, which supports strict (definitional) currying. This allows us to represent undetermined domain arities of implicit function types, and we can decide at any point during elaboration whether implicit abstractions should be inserted.

Additional Key Words and Phrases: impredicative polymorphism, type theory, elaboration, type inference

### 1 INTRODUCTION

- Elaboration.
- Hole filling.
- Insertion of implicits.
- First-class implicit functions vs Coq.
- Unique vs non-unique solutions.

#### 2 BIDIRECTIONAL ELABORATION

- •
- Limitations in Agda
- Insertion of implicits.
- First-class implicit functions vs Coq.
- Unique vs non-unique solutions.
- On the uniqueness of implicit insertion.

#### REFERENCES