Proof nets and the instantiation overflow property

Paolo Pistone
paolo.pistone2@unibo.it

Introduction A main feature of System F, responsible for its huge expressive power, is the so-called *full comprehension principle*, that is, the schema (for arbitrary types A, B)

$$\forall X.A \Rightarrow A[B/X] \tag{1}$$

In a recent series of papers [FF09, FF13] it was observed that for some types A, all instances of (1) can be deduced from a much weaker schema, the *atomic* comprehension principle

$$\forall X.A \Rightarrow A[Y/X] \tag{2}$$

This property was baptized *instantiation overflow* (IO for short). An interesting application of IO is that variants of the usual *impredicative* encodings of conjunction and disjunction can be developed inside System F_{at}, the predicative subsystem of F with (2) replaced by (1).

In fact, the IO property holds for all universal types of the form $\forall X.A$, where A is $A_1 \Rightarrow \ldots \Rightarrow A_n \Rightarrow X$ and the variable X only occurs in *strictly positive* position in the A_i. We call such types *Russell-Prawitz types* (RP-types for short), as they correspond to the second order encoding of n-ary propositional connectives defined by generalized introduction/elimination rules.[1]

In [PT20] it was shown that the fact that IO holds for RP-types can be explained in categorical terms from the perspective of *functorial polymorphism* [BFSS90]. In particular, the usual encodings of conjunction and disjunction can be transformed into predicative ones by applying certain rule permutations expressing a *dinaturality* condition.

However, while the analysis above only applies to RP-types, the IO property is not restricted to them. For instance, IO holds for all universal types $\forall X.A_n$, where $A_0 = Y \Rightarrow X$ and $A_{n+1} = A_n \Rightarrow X$ [FD16]. In this paper we provide a proof-theoretic characterization of the simple (linear and non-linear) types A such that IO holds for $\forall X.A$.

A detour through linear logic. Our starting point is to consider a related though slightly simpler property, the *expansion* property, for simple linear/non-linear types. A simple type A is expansible (in a fixed variable X) when for all n and types $B_1, \ldots, B_n$, we can prove

$$A \Rightarrow A[B_1 \Rightarrow \ldots \Rightarrow B_n \Rightarrow X/X] \tag{3}$$

where $\Rightarrow$ is replaced by $\multimap$ in the linear case. It is easily seen that when a simple type A is expansible, the universal type $\forall X.A$ satisfies IO.

A convenient way to investigate the expansion property is by means of *proof nets*, the graphical syntax of linear logic (or, more precisely, *essential nets*, a well-known intuitionistic variant [LS04]). In fact, while the λ-terms corresponding to proofs of (3) require a complex inductive definition, the associated proof nets can be classified (up to η-equivalence) in two classes based on their shape:

[1] The second order encodings originate in [Rus06] and [Pra65], whence the name.

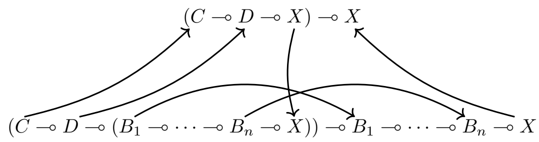

Figure 1: Simple expansion graph for $A = (C \multimap D \multimap X) \multimap X$.

1. the *simple expansion graphs*, which are made of identity links connecting each atom in A with the corresponding one in $A[B_1 \multimap \cdots \multimap B_n \multimap X/X]$, and of axiom links connecting dual occurrences of $B_1, \dots, B_n$ in $A[B/X]$ (see Fig. 1).

2. those which are obtained by a chain of compositions (with $B = B_1 \multimap \cdots \multimap B_n \multimap X$)

$$A \xrightarrow{f} A' \xrightarrow{\mathscr{G}} A'[B/X] \xrightarrow{g[B/X]} A[B/X] \tag{4}$$

   where A' is logically equivalent to A and *strictly smaller than* A (noted $|A'| < |A|$), $f : A \to A'$ and $g : A' \to A$ are proofs of this equivalence and $\mathscr{G}$ is a simple expansion graph.

The decomposition (4) follows from the *interpolation* theorem for multiplicative proof nets. Proof net interpolation algorithms are known ([BdG96, Car97]). As our results involve the implicational fragment of some intuitionistic systems, we had to consider the well-known fact that such fragments satisfy interpolation in a weaker form (see [Kan06]). To implement weak proof net interpolation for simple linear types we adapted the algorithm from [BdG96].

Using the classification above we obtain a characterization of expansible linear simple types in two steps. First, we characterize the linear simple types which admit simple expansion graphs: these are obtained by repeatedly substituting some (linear) RP-type for a variable X in a linear simple type in which X only occurs positively (we call such types *generalized RP-types*, gRP for short). Then, we prove that if an expansion graph can be decomposed as in (4), then A is equivalent to a tensor $A' = A_1 \otimes \cdots \otimes A_n$ of gRP types, where $|A'| < |A|$. We conclude then:

Theorem 1 (decidable criterion for expansible types)**.** *A linear simple type A is expansible iff it is equivalent to a tensor A' of gRP types, with $|A'| \leq |A|$.*

This result has a nice categorical corollary. It is well-known that formulas where a variable X only occurs positively act as covariant endofunctors over the syntactic category. From Theorem 1 one can deduce a converse statement:

Corollary 1. *If a linear simple type A acts as a covariant endofunctor over the syntactic category, then A is equivalent to a tensor of positive linear simple types A', with $|A'| \leq |A|$.*

From linear types to $\mathbf{F}_{at}$ By exploiting a folklore linearization argument relating simply-typed λ-terms and essential nets, and suitably adapting the definition of gRP types, the characterization of expansible types is shown to scale well from linear simple types to simple types and finally to F_{at}. This leads to the following:

Theorem 2 (criterion for IO)**.** *A simple type A has IO iff either A is provable or $\forall X.A$ is equivalent in F_{at} to $\forall X.A'$, where A' is a product of gRP types $A' = A_1 \times \cdots \times A_n$.*

Observe that the type A' is no more bounded in size by A, and in fact A' can grow exponentially in the size of A (as a consequence of the well-known fact that interpolants in intuitionistic logic cannot be polynomially bounded [SP98]). Thus, Theorem 2 does not settle whether the IO property is decidable: from the definition above it follows that it depends on derivability in F_{at}, which is undecidable. It can however be conjectured that type-equivalence within the rank-1 fragment of F_{at} (i.e. the fragment in which quantification is only admitted on quantifier-free types) is decidable.

References

[BdG96] Denis Bechet and Philippe de Groote. Constructing different phonological bracketings from a proof net. In C. Retoré, editor, *Logical Aspects of Computational Linguistics. LACL 1996*, volume 1328 of *Lecture Notes in Computer Science*, pages 118–133, Berlin, Heidelberg, 1996. Springer.

[BFSS90] E.S. Bainbridge, Peter J. Freyd, Andre Scedrov, and Philip J. Scott. Functorial polymorphism. *Theoretical Computer Science*, 70:35–64, 1990.

[Car97] Alessandra Carbone. Interpolants, cut elimination and flow graphs for the propositional calculus. *Annals of Pure and Applied Logic*, 83:249–299, 1997.

[FD16] Gilda Ferreira and Bruno Dinis. Instantiation overflow. *Reports on Mathematical Logic*, 51:15–33, 2016.

[FF09] Fernando Ferreira and Gilda Ferreira. Commuting conversions vs. the standard conversions of the "good" connectives. *Studia Logica*, 92(1):63–84, 2009.

[FF13] Fernando Ferreira and Gilda Ferreira. Atomic polymorphism. *Journal of Symbolic Logic*, 78(1):260–274, 2013.

[Kan06] Makoto Kanazawa. Computing interpolants in implicational logics. *Annals of Pure and Applied Logic*, 142(1-3):125–201, 2006.

[LS04] François Lamarche and Luz Straßburger. On proof nets for multiplicative linear logic with units. In *CSL 2004*, volume 3210 of *Lecture Notes in Computer Science*, pages 145–159, 2004.

[Pra65] Dag Prawitz. *Natural deduction, a proof-theoretical study.* Almqvist & Wiskell, 1965.

[PT20] Paolo Pistone and Luca Tranchini. Proving properties of polymorphic programs using type isomorphisms with the Yoneda lemma. `https://arxiv.org/abs/1907.03481`, 2020.

[Rus06] Bertrand Russell. On some difficulties in the theory of transfinite numbers and order types. In *Proceedings of the London Mathematical Society*, volume 4, pages 29–53, 1906.

[SP98] Uwe Schöning and Randall Pruim. The complexity of Craig interpolants. In *Gems of Theoretical Computer Science*, pages 111–113. Springer, Berlin, Heidelberg, 1998.