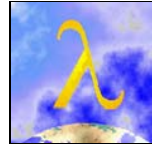

DIPARTIMENTO DI INFORMATICA
Università di Torino



Research on "Formal Methods in Computing"

Isomorphism Checker for Intersection-types

This software is a prototypal implementation of an isomorphism checker for intersection-types.

The current version (released in July 2010, based on an earlier prototype developed as part of the Master Thesis by D. Rispoli [[Rispoli, 2007](#)]):

- implements the algorithm described in [[Dezani-Ciancaglini, Di Cosmo, Giovannetti, Tatsuta 2010](#)]
- is written in [SWI-Prolog](#);
- accepts as input two types (in normal form, for now), and returns the λ -term which proves the isomorphism, if it exists, for the given pair of types.

Grammar:

- `expr ::= type , type`
- `type ::= type & type | type -> type | (type) | term`
- `term ::= term . numb | term . char | char`
- `char ::= a | ... | z | A | ... | Z`
- `numb ::= 0 | ... | 9`

Operators:

- `:- op(1200, xfy, ->).`

- `:- op(1100, xfy, &).`

Examples:

Non isomorphic types:

- `f & (s -> t & r) , f & (s -> t) & (s -> r)`
- `f & (s -> t & r) , f & (s -> t & r) & (s & t -> r)`
- `(f -> f) & ((s -> t & r) -> f) , (f -> f) & ((s -> t) & (s -> r) -> f)`

Isomorphic types:

- `(b22 -> b21 -> a2) -> b3 -> b1 -> a , b1 -> (b21 -> b22 -> a2) -> b3 -> a`
- **S1, S2** where
`S1 = ((c -> a & b -> d) & (b -> g -> c) -> s -> (e -> f) -> phi) & ((v -> u -> w) -> q & r -> (h -> k) & (p -> q) -> (a -> psi)),`
`S2 = ((e -> f) -> (a & b -> c -> d) & (g -> b -> c) -> s -> phi) & ((h -> k) & (p -> q) -> (u -> v -> w) -> q & r -> (a -> psi))`
- **T1, T2** where
`T2 = ((a111 -> a112 -> v11) -> a12 -> (a131 -> a132 -> a133 -> a134 -> v13) -> v1) -> (a21 -> (a221 -> a222 -> a223 -> v22) -> a23 -> (a241 -> a242 -> v24) -> v2) -> a3 -> v`
`T1 = (a23 -> a21 -> (a242 -> a241 -> v24) -> (a221 -> a223 -> a222 -> v22) -> v2) -> a3 -> ((a112 -> a111 -> v11) -> a12 -> (a133 -> a132 -> a131 -> a134 -> v13) -> v1) -> v`

Bibliography:

- [Dezani-Ciancaglini, Di Cosmo, Giovannetti, Tatsuta 2010]
M. Dezani-Ciancaglini, R. Di Cosmo, E. Giovannetti, M. Tatsuta. On isomorphisms of intersection types. *ACM Trans. Comput. Log.* 11(4): (2010)
- [Rispoli 2007]
D. Rispoli. A Prolog isomorphism checker for Intersection Types. *Laurea Degree Stage Final Report*, August 2007.