RESOLUTION IN THE CALCULUS OF STRUCTURES

Alessio Guglielmi (TU Dresden)
29.9.2003

It is possible to search for cut-free proofs by the resolution strategy in the calculus of structures [WS]; the sequent calculus does not allow the same freedom.

The relation between the sequent calculus and resolution has been mainly studied by Mints in [GM]; the textbook [TS] presents an overview. The correspondence is sufficiently clear and clean to allow several applications, especially in proof complexity. I don't know of any application in proof search, intended as the proof theoretical framework for designing logic programming languages advocated by Miller *et al.* in [MNPS].

The resolution rule receives a natural interpretation in the sequent calculus as a cut rule, but there is no natural way, as far as I know, of expressing resolution in a cut-free system. As I'll argue briefly below, one reason is that branching in the sequent calculus is too big an obstacle for resolution to work naturally (see also [AG]).

Of course, in the case of SLD-resolution, which is a very special case of resolution, branching in the sequent calculus is no impediment to develop the interesting theory of uniform provability [MNPS]. Perhaps the observation in this note, for unrestricted resolution, could lead to some developments in proof search.

To illustrate the idea, a simple example suffices. The reader should look in [KB] for system SKSg of classical propositional logic.

Resolution in the Calculus of Structures

A formula in disjunctive normal form is represented as a disjunctive structure

$$[C_1,\dots,C_h] \ ,$$

where C_i is a clause, i.e., a conjunction of atoms

$$(a_1,\dots,a_{ki}) \ .$$

The resolution rule

$$r\downarrow \ \frac{[(R,T),(a,R),(\neg a,T),U]}{[(a,R),(\neg a,T),U]} \ ,$$

where R and T are clauses and U is in disjunctive normal form, is obtained in SKSg as

```
              [(R,T),(a,R),(¬a,T),U]
    i↓  -----------------------------
        [([a¬a],R,T),(a,R),(¬a,T),U]
   s;s  -----------------------------
        [(a,R),(¬a,T),(a,R),(¬a,T),U]
 c↓;c↓  -----------------------------.
        [(a,R),(¬a,T),U]
```

Proving the structure Q_0 by h resolution steps means finding a proof

```
     t
 w↓ -----
     .
     .
     .
 w↓ -----
    [t,P]
 r↓ -----
     Q_h-1
     .
     .
     .
     Q_1
  r↓ ---,
     Q_0
```

Where P is in disjunctive normal form.

A *refutation* by resolution is simply obtained by top-down flipping the derivation above. This means that both styles of resolution are directly supported by the calculus of structures.

By flipping the derivation above one introduces cuts (in correspondence to i↓ rules). Please notice that these cuts are *finitary*, since the atoms introduced by them are present in the conclusion as well (thanks to Kai Brünnler for this observation, see the paper [BG] about finitary cuts). This is not so important anyway, because in a refutation one builds the derivation top-down, so the presence of cuts, finitary or otherwise, doesn't really concern.

Observations

I believe that the sequent calculus, in every reasonable variation and understanding of the notion, is probably not so natural for resolution, in a cut-free proof-search perspective. Observe the resolution rule:

```
     [(R,T),(a,R),(¬a,T),U]
 r↓ ----------------------.
        [(a,R),(¬a,T),U]
```

Looking at it bottom-up, one sees 1) the disruption of two conjunctions, one between a and R and another between ¬a and T; and 2) the *construction* of a conjunction between R and T.

This is where the sequent calculus fails: (R,T) is not a subformula of [(a,R),(¬a,T)].

The more general notion of subformula property of the calculus of structures, instead, deals graciously with the situation, and we know that its generality doesn't impede the finitariness of the calculus of structures, as [BG,KB] show.

Of course, everything I showed above is trivial. One shouldn't expect anything spectacular out of these observations. The reason I illustrate them is that, despite their total simplicity, they do make the connection between resolution and a formalism for which a very rich proof theory is developed.

For example, one can think of studying the resolution rule in a deep inference setting, which is naturally provided by the calculus of structures and which could perhaps lead to improvements of the resolution technique.

References

[AG] Alessio Guglielmi. Mismatch. Manuscript, 2003, URL: http://www.ki.inf.tu-dresden.de/~guglielm/res/notes/AG9.pdf.

[BG] Kai Brünnler and Alessio Guglielmi. A Finitary System for First Order Logic. Technical Report WV-03-09, Technische Universität Dresden, 2003. Accepted at *First Order Logic 75*. URL: http://www.ki.inf.tu-dresden.de/~kai/FinitaryFOL.pdf.

[GM] G. Mints. Gentzen-type systems and resolution rules Part I Propositional logic. In *COLOG 1*, volume 417 of *Lecture Notes in Computer Science*, pages 198-231. Springer-Verlag, 1990.

[KB] Kai Brünnler. Deep inference and symmetry in classical proofs. PhD thesis, Technische Universität Dresden, 2003. URL: http://www.ki.inf.tu-dresden.de/~kai/phd.pdf.

[MNPS] D. Miller, G. Nadathur, F. Pfenning and A. Scedrov. Uniform proofs as a foundation for logic programming. *Annals of Pure and Applied Logic*. 51:125-157, 1991.

[TS] A. S. Troelstra and H. Schwichtenberg. *Basic Proof Theory*, volume 43 of Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 1996.

Web Site

[WS] http://alessio.guglielmi.name/res/cos.