From Additive to Classical Proof Search

Adam Lassiter
Department of Computer Science
University of Bath

Willem Heijltjes
Department of Computer Science
University of Bath

Introduction

Building on the work done by Heijltjes & Hughes (2015), we investigate proof search in classical logic through Additive Linear Logic (ALL). The process we investigate, called *coalescence*, is a top-down proof search from axiom links down to the conclusion. This method is promising as it boasts great efficiency for ALL proof search and has a natural transformation to sequent calculus proofs.

Additive Linear Logic

ALL is the fragment of linear logic that concerns sum $(+)$ and product $(\times)$, with their units 0 and 1. A *formula* of ALL is constructed:

$$A, B, C \quad ::= \quad 0 \mid 1 \mid a \mid \overline{a} \mid A + B \mid A \times B$$

We write $|\mathsf{A}|$ for the set of subformula occurrences of A (we distinguish both occurrences of a in $a \times a$).

A sequent for ALL is a pair $\vdash A, B$, and a sequent calculus for ALL is given by the following rules (where the symmetric rules operating on the second element of the pair $\vdash A, B$ are omitted):

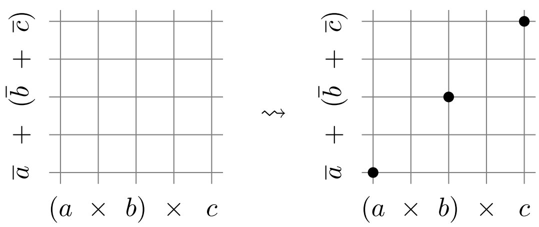
$$\frac{}{\vdash a, \overline{a}}\ \mathsf{ax} \qquad \frac{}{\vdash 1, A}\ 1 \qquad \frac{\vdash A, C}{\vdash A + B, C}\ +_1$$

$$\frac{\vdash B, C}{\vdash A + B, C}\ +_2 \qquad \frac{\vdash A, C \qquad \vdash B, C}{\vdash A \times B, C}\ \times$$

Coalescence Proof Search

Naively searching for a sequent proof, starting from a given conclusion, is exponential: the additive conjunction rule ($\times$) duplicates its context C, and reciprocated duplication between both formulas in the sequent creates exponential growth.

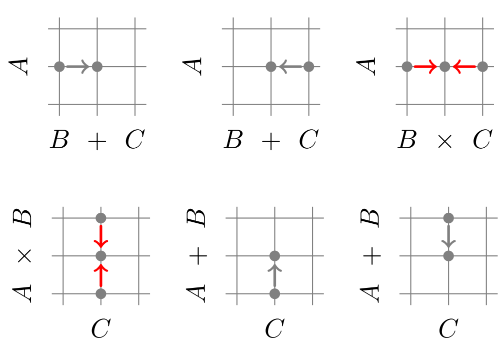
A more efficient algorithm, first observed by Galmiche & Marion **?** and later by Heijltjes & Hughes (2015), is given by placing proof search for a sequent $\vdash A, B$ in the product space $|\mathsf{A}| \times |\mathsf{B}|$. This set contains all *sub-sequents* of $\vdash A, B$ that might occur in a sequent proof of $\vdash A, B$, without redundancy. We represent $|\mathsf{A}| \times |\mathsf{B}|$ by a grid, e.g. that for $\vdash \overline{a} + (\overline{b} + \overline{c}), (a \times b) \times c$ is below:



The *coalescence proof search* algorithm for ALL is then as follows. We place tokens on this grid to indicate a sub-sequent is provable. Initially, we place tokens on each position $(\overline{a}, a)$, $(1, A)$, and $(A, 1)$, as shown above right for our example. Then, we apply the following local rules (and the symmetric variants).

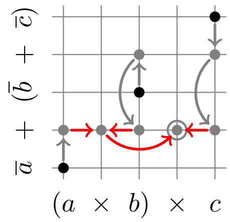
- Place a token on $(A + B, C)$ if (A, C) has a token.
- Place a token on $(A + B, C)$ if (B, C) has a token.
- Place a token on $(A \times B, C)$ if (A, C) and (B, C) have a token.

We illustrate these graphically as follows; note, though, that the *neighbours* of a position in the grid are given by the subformula relation, and not by adjacency in the plane.



Observe also that the initial token placement and the propagation rules corresponds directly to the axioms respectively the rules of the sequent rules for ALL.

Finally, if the root position (A, B) for $\vdash A, B$ has a token, we succeed; otherwise, if no more tokens can be placed we fail. For our example, we get the following trace:



Classical Logic and Coalescence

A *formula* of ALL within Classical Logic (CL) is constructed:

$$A, B, C \quad ::= \quad \top \mid \bot \mid a \mid \overline{a} \mid A \vee B \mid A \wedge B$$

A sequent calculus for CL is then given by the following rules:

$$\frac{}{\vdash \top}\,\top \qquad \frac{\vdash \Gamma, A}{\vdash \Gamma, A \vee B}\,\vee \qquad \frac{\vdash \Gamma}{\vdash \Gamma, A}\,w$$

$$\frac{}{\vdash a, \overline{a}}\,ax \qquad \frac{\vdash \Gamma, A \qquad \vdash \Gamma, B}{\vdash \Gamma, A \wedge B}\,\wedge \qquad \frac{\vdash \Gamma, A, A}{\vdash \Gamma, A}\,c$$

where A, B, C are formulae and Γ, Δ, Σ are sequents. Notice that both conjunction $\wedge$ and disjunction $\vee$ rules preserve the number of terms in a sequent.

For a formula P, the algorithm runs as follows:

1. Set $n := 2$
2. Construct the n-dimensional net of possible transitions on n-term sequents $\vdash A_1 \ldots A_n$
3. Spawn tokens in the net at all instances of axiom links $\vdash \Gamma, a, \overline{a}$
4. Exhaustively fire the net
5. Does there exist a token at $(P, P \ldots P) \equiv \vdash P, P \ldots P \equiv \vdash P$?

(a) Yes — Halt with a proof for P

(b) No — Increment $n := n + 1$ and goto 2

The dimensionality of a proof is then the dimensionality of our grid when the root is reached, equivalent to the number of contractions required in an equivalent sequent proof. We thus say a classical formula can be proved by an n-dimensional additive proof, where n is the number of terms in a sequent.

We prove that this is exactly proof search through *additive stratification* of the sequent calculus — that is, any sequent proof may be 'rearranged' up to the order of rules applied. A proof tree is said to be *additively stratified* if it is structured as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash A_1}\,\top, ax}{\vdash \Gamma_1}\,w \qquad \dots \qquad \dfrac{\dfrac{}{\vdash A_n}\,\top, ax}{\vdash \Gamma_n}\,w}{\vdash P \dots P}\,\wedge, \vee}{\vdash P}\,c$$

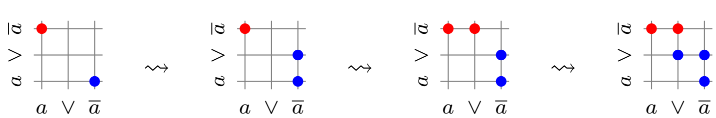
Coalescence is then equivalent to (additively stratified) proof search, with implicit weakening and contraction of all sequents up to n terms.

Motivation

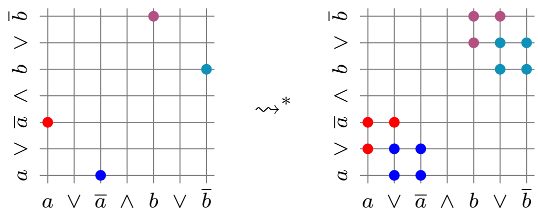
Clearly complexity scales with dimensionality and our motivation is then: *'What dimension is sufficient for a given formula?'*. In essence, this gives an upper bound for proof search.

Some Examples

Consider a simple proof requiring disjunction through $A ::= a \vee \overline{a}$ as follows:



The root position (A, A) is reached in 2 dimensions so we describe the associated proof as having dimensionality 2. Consider next a simple proof requiring conjunction through $B ::= (a \vee \overline{a}) \wedge (b \vee \overline{b})$ as follows (note that the intermediate steps are skipped and the net is saturated before a proof is reached):



We do not reach the root (B, B) for $n = 2$ despite having two proven 'subproofs' and need only apply a simple conjunction. Instead, a solution is reached for $n = 3$ (visualisation omitted due to lack of clarity of 3D diagrams). For a similar term in 3 variables a, b, c, a solution is reached for $n = 4$ and growth continues linearly. This is deemend unsatisfying and we readdress the mechanics of coalescence to fix this.

Solution

To solve this issue, we then investigate liberating the search algorithm and generalising over the properties of sequents — namely, idempotency and commutativity. This includes: some notion of applying conjunctions 'diagonally' (from $(a \vee \overline{a}, b \vee \overline{b})$ to (B, B) in one step in the above) and switching from *tuple* or *multiset* links to *set* links. The latter takes us into more familiar/obvious proof search territory:

$$
\begin{tikzcd}
& \{a, \overline{a}\} \arrow[dl] \arrow[dr] & & & \{b, \overline{b}\} \arrow[dl] \arrow[dr] & \\
\{a \vee \overline{a}, \overline{a}\} \arrow[dr] & & \{a, a \vee \overline{a}\} \arrow[dl] & \{b \vee \overline{b}, \overline{b}\} \arrow[dr] & & \{b, b \vee \overline{b}\} \arrow[dl] \\
& \{a \vee \overline{a}\} \arrow[drr] & & & \{b \vee \overline{b}\} \arrow[dll] & \\
& & & \{(a \vee \overline{a}) \wedge (b \vee \overline{b})\} & &
\end{tikzcd}
$$

Beyond these two points, there exist various other implementation trade-offs: dense/sparse representation, high-performance data structures and assorted other ad-hoc optimisations.

References

Heijltjes, W. & Hughes, D. J. (2015), Complexity bounds for sum-product logic via additive proof nets and petri nets, *in* '2015 30th Annual ACM/IEEE

Symposium on Logic in Computer Science', IEEE, pp. 80–91.