Presentation of
Master's Thesis at
Prover Technology

Stålmarck's Method versus Resolution: A Comparative Theoretical Study

Jakob Nordström

November 7, 2001

Outline of Presentation

- Basic concepts in proof theory
- Dilemma
- Resolution
- Some results on dilemma and resolution
- Some open questions

Propositional Proof Systems

A propositional logic formula F is a **tautology** if all truth value assignments satisfy F.

$TAUT$: The set of all tautologies.

Propositional proof system: Predicate $\mathcal{P}$ computable in polynomial time such that for all F it holds that $F \in TAUT$ iff there exists a **proof** π of F such that $\mathcal{P}\Big(F, \pi\Big)$ is true.

$\mathcal{P}_1$ p-**simulates** $\mathcal{P}_2$ if there exists a polynomial-time computable function f mapping proofs in $\mathcal{P}_2$ into proofs in $\mathcal{P}_1$.

$\mathcal{P}_1$ and $\mathcal{P}_2$ are p-**equivalent** if they p-simulate each other.

Connection to Complexity Theory

$S(F)$	Size (# symbols) of formula F
$S_{\mathcal{P}}(\vdash F)$	Size of a smallest proof of tautology F in proof system $\mathcal{P}$

The **complexity** of $\mathcal{P}$ is the smallest bounding function $g : \mathbb{N} \mapsto \mathbb{N}$ for which

$$S_{\mathcal{P}}(\vdash F) \leq g\Big(S(F)\Big)$$

for all $F \in TAUT$.

A proof system of polynomial complexity is p-**bounded**.

No p-bounded proof system has been found. If none exist, it would follow that P $\neq$ NP.

Theorem (Cook and Reckhow 1979)

The equality NP = co-NP holds iff there exists a p-bounded propositional proof system.

Proof Methods

Proof method $A_{\mathcal{P}}$ for proof system $\mathcal{P}$:

- Deterministic algorithm
- Input: Propositional logic formula F
- Output: Proof π of F in $\mathcal{P}$ if F tautology, otherwise example that F is falsifiable.

Efficiency of proof method $A_{\mathcal{P}}$ measured as running time on input F relative to $S_{\mathcal{P}}(\vdash F)$.

Automatizability

Two importance properties of proof system $\mathcal{P}$:

1. What is the size of a smallest $\mathcal{P}$-proof of F (complexity)?

2. Is there an efficient way of *finding* as small as possible $\mathcal{P}$-proofs (**automatizability**)?

"Efficient" = polynomial.

A proof system $\mathcal{P}$ is **automatizable** if there is a proof method $A_{\mathcal{P}}$ that produces a $\mathcal{P}$-proof of F in time polynomial in $S_{\mathcal{P}}(\vdash F)$, i.e. if

$$Time\,(A_{\mathcal{P}}(F)) \leq S_{\mathcal{P}}(\vdash F)^{O(1)}.$$

$\mathcal{P}$ is **quasi-automatizable** if the running time of $A_{\mathcal{P}}$ is quasi-polynomial in $S_{\mathcal{P}}(\vdash F)$, i.e. if

$$Time\,(A_{\mathcal{P}}(F)) \leq \exp\left((\log S_{\mathcal{P}}(\vdash F))^{O(1)}\right).$$

Formula Relations in Dilemma

Stålmarck's method is based on the **dilemma proof system**.

Derivations are built of **formula relations**.

A formula relation R is an equivalence relation over the subformulas $Sub(F)$ of F, i.e.

- reflexive ($P \equiv P$),
- symmetric ($P \equiv Q \Rightarrow Q \equiv P$),
- transitive ($P \equiv Q$ and $Q \equiv S \Rightarrow P \equiv S$),

which in addition

- respects the semantical meaning of logical negation ($P \equiv Q \Rightarrow \neg P \equiv \neg Q$).

Formula Relation Notation

$\mathsf{R}\Big[P \equiv Q\Big]$	Formula relation R with equivalence classes of P and Q merged
$\mathsf{R}_1 \sqcap \mathsf{R}_2$	Intersection of R_1 and R_2 containing all equivalences found in both relations.
F^+	Identity relation on $Sub\,(F)$

To prove that F is a tautology, start with $F^+\Big[F \equiv \bot\Big]$ and derive a contradiction.

A contradiction is reached when P and $\neg P$ are placed in the same equivalence class for some subformula $P \in Sub\,(F)$.

The Dilemma Proof System

Propagation rules: If the formula relation R is such that some equivalence between P, Q and $P \circ Q$ ($\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$) follows from the truth table of the connective $\circ$, then there is a rule to derive this equivalence.

Composition: If $\pi_1 : \mathsf{R}_1 \Rightarrow \mathsf{R}_2$ and $\pi_2 : \mathsf{R}_2 \Rightarrow \mathsf{R}_3$ are dilemma derivations, then π_1 followed by π_2 is a derivation $\pi_1 \bullet \pi_2 : \mathsf{R}_1 \Rightarrow \mathsf{R}_3$.

Dilemma rule: If π_1 and π_2 are derivations $\pi_1 : \mathsf{R}\Big[P \equiv Q\Big] \Rightarrow \mathsf{R}_1$, $\pi_2 : \mathsf{R}\Big[P \equiv \neg Q\Big] \Rightarrow \mathsf{R}_2$, then

$$\frac{\mathsf{R}}{\begin{array}{cc} \mathsf{R}\Big[P \equiv Q\Big] & \mathsf{R}\Big[P \equiv \neg Q\Big] \\ \pi_1 & \pi_2 \\ \mathsf{R}_1 & \mathsf{R}_2 \end{array}} \over \mathsf{R}_1 \sqcap \mathsf{R}_2$$

is a dilemma rule derivation of $\mathsf{R}_1 \sqcap \mathsf{R}_2$ from R.

Dilemma Proof Hardness

Depth $D(\pi)$ of a derivation π: max # of nested dilemma rule applications.

A formula relation R is κ-**easy** if there is a derivation $\pi : \mathsf{R} \Rightarrow \bot$ with $D(\pi) \leq \kappa$.

R is κ-**hard** if there is no derivation $\pi : \mathsf{R} \Rightarrow \bot$ with $D(\pi) < \kappa$.

If R is both κ-easy and κ-hard, it is **exactly** κ-**hard** and has **hardness degree** $H(\mathsf{R}) = \kappa$.

The hardness degree of a tautology F is

$$H(F) := H\left(F^{+}\left[F \equiv \bot\right]\right).$$

Proof Hardness and Proof Length

Easy formulas have short dilemma proofs.

Hard formulas (and only hard formulas) require long dilemma proofs.

More precisely:

Theorem

Let F be a tautology with hardness $H(F)$. Then for the minimum proof length $L_{\mathcal{D}}(\vdash F)$ in dilemma it holds that

$$2^{H(F)/2} \leq L_{\mathcal{D}}(\vdash F) \leq S(F)^{H(F)+1}.$$

Dilemma Subsystems

Atomic dilemma $\mathcal{D}_A$: Dilemma rule assumptions on the form $x \equiv \bot$ or $x \equiv \top$ for atomic variables $x \in Vars(\mathsf{R})$.

Bivalent dilemma $\mathcal{D}_B$: Dilemma rule assumptions on the form $P \equiv \bot$ or $P \equiv \top$ for subformulas $P \in Sub(\mathsf{R})$.

General dilemma $\mathcal{D}$: Any dilemma rule assumptions $P \equiv Q$ for $P, Q \in Sub(\mathsf{R})$.

Reductio proof systems: Allow merging of branches only when contradiction is derived.

Corresponds to *reduction ad absurdum* rule.

Proof systems $\mathcal{RAA}_A$, $\mathcal{RAA}_B$ and $\mathcal{RAA}$.

Conjunctive Normal Form

A **literal** over x is either x itself or its negation $\overline{x}$. (In some contexts the notation x^1 for x and x^0 for $\overline{x}$ is convenient.)

A **clause** is a disjunction of literals.

A **CNF formula** is a conjunction of clauses.

A clause containing exactly k literals is called a k-**clause**.

A k-**CNF formula** is a CNF formula consisting of k-clauses.

For a k-CNF formula F with m clauses over n variables, $\Delta = m/n$ is the **density** of F.

Resolution

A **resolution derivation** of a clause A from a CNF formula F is a sequence $\pi = \{D_1, \ldots, D_s\}$ such that $D_s = A$ and each D_i, $1 \le i \le s$, is either in F or is derived from D_j, D_k in π (with $j, k < i$) by the **resolution rule**

$$\frac{B \vee x \quad C \vee \overline{x}}{B \vee C}$$

or the **weakening rule**

$$\frac{B}{B \vee C}$$

(the weakening rule can be omitted).

A **resolution refutation** of F is a resolution derivation of the empty clause 0 from F.

A resolution derivation is **tree-like** if any clause in the derivation is used at most once as a premise in the resolution rule (i.e. if the DAG corresponding to the derivation is a tree).

DLL procedures

Simple scheme for a family of algorithms for refuting a contradictory CNF formula F on n variables:

If the empty clause 0 is in F, report that F in unsatisfiable and halt.

Otherwise, pick a variable $x \in F$ and recursively try to refute $F|_{x=0}$ and $F|_{x=1}$.

Introduced by Davis, Logemann and Loveland (1962); therefore called **DLL procedures**.

Width-Length Relations

If a minimum-length resolution refutation π of a formula F is long, it seems probable that π contains clauses with many literals.

Conversely, short proofs can be expected to be narrow as well.

Making this intuition precise, Ben-Sasson and Wigderson (1999) have proved:

- If a contradictory CNF formula F has a tree-like refutation of length L_T, then it has a refutation of max width $\log_2 L_T$.

- If a contradictory CNF formula F has a general resolution refutation of length L, then it has a refutation of max width

$$\mathrm{O}\left(\sqrt{n \log L}\right)$$

(where n is the number of variables in F).

Width

The **width** $W(C)$ of a clause C is the number of literals in it.

The width of a formula (or derivation) is the max clause width in the formula (derivation).

The width of deriving a clause C from F by resolution is

$$W(F \vdash C) := \min_{\pi} \{W(\pi)\},$$

where the minimum is taken over all resolution derivation π of C from F.

$W(F \vdash \bot)$ is the min width of refuting F by resolution.

Technical Lemmas about Width

$F \vdash_w A$ denotes that A can be derived from F in width $\leq w$.

Technical lemma 1

For $\nu \in \{0, 1\}$, if it holds that $F|_{x=\nu} \vdash_w A$ then $F \vdash_{w+1} A \vee x^{1-\nu}$ (possibly by use of the weakening rule).

Technical lemma 2

For $\nu \in \{0, 1\}$, if

$$F|_{x=\nu} \vdash_{w-1} 0$$

and

$$F|_{x=1-\nu} \vdash_w 0$$

then

$$W(F \vdash \bot) \leq \max\{w, W(F)\}.$$

Width-Length for Tree Resolution

Theorem (Ben-Sasson, Wigderson 1999)

For tree-like resolution, the width of refuting a CNF formula F is bounded from above by

$$W(F \vdash \bot) \leq W(F) + \log_2 L_{\mathcal{T}}(F \vdash \bot).$$

Corollary

For tree-like resolution, the length of refuting a CNF formula F is bounded from below by

$$L_{\mathcal{T}}(F \vdash \bot) \geq 2^{(W(F \vdash \bot) - W(F))}.$$

Width-Length for Resolution

Theorem (Ben-Sasson, Wigderson 1999)

For general resolution, the width of refuting a CNF formula F is bounded from above by

$$W(F \vdash \bot) \leq W(F) + \mathrm{O}\left(\sqrt{n \log L_{\mathcal{R}}(F \vdash \bot)}\right)$$

(where n is the number of variables in F).

Corollary

For general resolution, the length of refuting a CNF formula F is bounded from below by

$$L_{\mathcal{R}}(F \vdash \bot) \geq \exp\left(\Omega\left(\frac{(W(F \vdash \bot) - W(F))^2}{n}\right)\right).$$

Proof Strategy for Length Bounds

Prove lower bounds on refutation *length* by showing lower bounds on refutation *width*. The strategy:

1. Define a complexity measure
$$\mu : \{\text{Clauses}\} \mapsto \mathbb{N}^+$$
such that $\mu\Big(C\Big) = 1$ for all $C \in F$.

2. Prove that $\mu\Big(0\Big)$ must be large.

3. Infer that in every refutation π of F there is a clause D with *medium-sized* complexity measure $\mu\Big(D\Big)$.

4. Prove that if the measure $\mu\Big(D\Big)$ of a clause $D \in \pi$ is medium then the width $W(D)$ is *large*.

Lower Bound on Refutations of Random 3-CNF Formulas

$F \sim \mathcal{F}_k^{n,\Delta}$ denotes that F is a k-CNF formula on n variables and $m = \Delta n$ independently and identically distributed random clauses from the set of all $2^k \binom{n}{k}$ k-clauses with repetitions.

Lemma (Ben-Sasson, Wigderson 1999)

For $F \sim \mathcal{F}_3^{n,\Delta}$ and any $\epsilon > 0$, with probability $1 - o(1)$ in n it holds that

$$W(F \vdash \bot) = \exp\Big(\Omega\left(n/\Delta^{2+\epsilon}\right)\Big).$$

Theorem (Beame et al. 1998)

For $F \sim \mathcal{F}_3^{n,\Delta}$ and any $\epsilon > 0$, with probability $1 - o(1)$ in n it holds that

$$L_{\mathcal{R}}(F \vdash \bot) = \exp\Big(\Omega\left(n/\Delta^{4+\epsilon}\right)\Big).$$

Results

The results in the Master's thesis can be divided into two categories:

1. Comparison of different dilemma and RAA proof systems.

2. Comparison of dilemma and resolution.

In this presentation, we concentrate on (2).

Dilemma and Tree Resolution

Atomic dilemma is exponentially stronger than tree-like resolution with respect to proof length.

That is, there exists a polynomial-size family of formulas F_n such that

$$L_{\mathcal{D}_A}(F_n \vdash \bot) = n^{\mathrm{O}(1)}$$

but

$$L_{\mathcal{T}}(F_n \vdash \bot) = \exp\Big(\Omega\,(n)\Big).$$

This shows that there are formula families for which Stålmarck's proof method beats any DLL procedure exponentially.

Depth-Width Relation of Dilemma and Resolution

Suppose that F is an unsatisfiable CNF formula in width $W(F) = k$.

Then any dilemma refutation π_D of F in depth $D(\pi_D) = d$ and length $L(\pi_D) = L$ can be translated to a resolution refutation π_R of F in width

$$W(\pi_R) \leq \mathrm{O}(kd)$$

and length

$$L(\pi_R) \leq \left(Lk^d\right)^{\mathrm{O}(1)}.$$

Intuition for Depth-Width Relation

Given a dilemma derivation π.

1. Suppose that $S_1 \equiv S_2$ is derived in π under assumptions $P_1 \equiv Q_1, \ldots, P_i \equiv Q_i$.

   Denote this

   $$P_1 \equiv Q_1 \Rightarrow \ldots \Rightarrow P_i \equiv Q_i \Rightarrow S_1 \equiv S_2.$$

2. Rewrite the above to an equivalent set of CNF clauses

   $$\mathit{CNF}\,(P_1 \equiv Q_1 \Rightarrow \ldots \Rightarrow P_i \equiv Q_i \Rightarrow S_1 \equiv S_2)\,.$$

3. Do this for each step in π.

   Show that the resulting sets of clauses form the "backbone" of a resolution derivation, the gaps of which can be completed in width and length as stated.

Stålmarck's Method and Minimum-Width Proof Search

1. Let F be a contradictory CNF formula in width $W(F) \leq k$ (for some fixed k).

   Then the minimum-width proof search algorithm in resolution refutes the formula F in time polynomial in the running time of Stålmarck's method.

2. Suppose that G is a tautological formula in propositional logic.

   Then minimum-width proof search proves G valid by refuting the Tseitin transformation to CNF G_t of G in time polynomial in the running time of Stålmarck's method on G.

Bounds on Dilemma Hardness of Random 3-CNF Formulas

Suppose that $F \sim \mathcal{F}_3^{n,\Delta}$.

Suppose also that the density Δ is sufficiently large so that F is unsatisfiable with probability $1 - \mathrm{o}(1)$ in n.

Then with probability $1 - \mathrm{o}(1)$ in n

$$\Omega\left(n/\Delta^{2+\epsilon}\right) \leq H_{\mathcal{D}}(F) \leq \mathrm{O}(n/\Delta)$$

where $\epsilon > 0$ is arbitrary.

Two Open Questions

- Bounds on depth in dilemma translates into bounds on width in resolution.

  Is this true in the opposite direction as well? That is, can resolution in width w be transformed to dilemma in depth $O(w)$?

- Minimum-width proof search in resolution is polynomial in Stålmarck's method.

  This is a purely theoretical result. How would efficient implementations of the two algorithms compare in practice?