On fairness in theorem proving

Maria Paola Bonacina

Dipartimento di Informatica
Università degli Studi di Verona
Verona, Italy, EU

Talk given at Microsoft Research, Redmond, Washington, USA

26 June 2013

The gist of this talk

- Theorem proving is search, not saturation
- The relevant property is fairness
- Fairness should earn less than saturation
- Fairness should consider both expansion and contraction

Fairness in computing

- Scheduling: no starvation of processes
- Search: no neglect of “useful” moves

Automated reasoning

- Inference system or Transition system:
  set of non-deterministic rules
  defines the search space of all possible steps
- Search plan: controls rules application
  guides search for proof/model
  adds determinism: given input, unique derivation

Procedure/Strategy = Rule system + Search plan

Requirements

- System of rules: completeness
  there exist successful derivations
- Search plan: fairness
  ensure that the generated derivation succeeds

Theorem proving (TP)

- Inference system: refutational completeness
  if input set unsat
  there exist derivations yielding $\perp$ (and a proof)
- Search plan: fairness
  ensure that the generated derivation yields $\perp$
- Complete TP strategy $=$
  Refutationally complete inference system $+$ Fair search plan

Fairness?

- Exhaustive: consider eventually all applicable steps
  trivial, brute force way to be fair
- How to be fair without being exhaustive?
- Non-trivial definitions of fairness?
- Non-trivially fair search plans?
- Non-trivial fairness: reduce gap between completeness and efficiency

Fairness and redundancy

- Consider eventually all needed steps: What is needed?
- Dually: what is not needed, or: what is redundant?
- Fairness and redundancy are related

Redundancy I

- Resolution: generate resolvents by resolving complementary literals
- Subsumption: clause C eliminates less general clause D
- Subsumption ordering: $D \stackrel{\bullet}{\geq} C$ if $C\sigma \subseteq D$ (as multisets)
  $D \stackrel{\bullet}{>} C$ if $D \stackrel{\bullet}{\geq} C$ and $C \stackrel{\bullet}{\not\geq} D$
- D redundant in S ($D \in Red(S)$)
  if there exists $C \in S$ that subsumes D (strictly)
  [Michäel Rusinowitch]

Redundancy II

- Well-founded ordering $\prec$ on terms and literals
- Superposition: resolution with equality built-in: superpose maximal side of maximal equation into maximal literal/side (maximal after mgu)
- Simplification: by well-founded rewriting
- Ground D redundant in S if for ground instances $C_1 \ldots C_n$ of clauses in S, $C_1 \ldots C_n \prec D$ and $C_1 \ldots C_n \models D$;
  D redundant in S ($D \in Red(S)$) if all its ground instances are [Leo Bachmair and Harald Ganzinger]

Redundancy III

- From clauses to inferences
- Redundant inference: uses/generates redundant clause

Fairness is a global property

Derivation:

$$S_0 \vdash S_1 \vdash \ldots S_i \vdash S_{i+1} \ldots$$

Limit: set of persistent clauses

$$S_\infty = \bigcup_{j \geq 0} \bigcap_{i \geq j} S_i$$

Uniform fairness

$C \in I_E(S)$: C generated from S by expansion

$S_0 \vdash S_1 \vdash \ldots S_i \vdash S_{i+1} \ldots$

- For all $C \in I_E(S_\infty)$ exists j such that $C \in S_j \cup Red(S_j)$
- For all $C \in I_E(S_\infty \setminus Red(S_\infty))$ exists j such that $C \in S_j$
- All non-redundant expansion inferences done eventually

[Leo Bachmair and Harald Ganzinger]

A weaker notion of fairness?

- Uniform fairness is for saturation
- Fairness for theorem proving?

Proof orderings

- Well-founded proof ordering $<$

[Leo Bachmair, Nachum Dershowitz and Jieh Hsiang]

- May reduce to formula ordering if we compare proofs by their premises
- But it is more flexible: small proofs may have large premises

Proof reduction

- Justification: set of proofs P
- Comparing justifications:
  Q better than P, written $P \sqsupseteq Q$:
  $\forall p \in P. \exists q \in Q.\ p \geq q$

Comparing presentations by their proofs

- S presentation of $Th(S)$
- Proofs with premises in S: $Pf(S)$
- S' simpler than S, written $S \succsim S'$:
  $S \equiv S'$ and $Pf(S) \sqsupseteq Pf(S')$

Best proofs

- Minimal proofs in a justification: $\mu(P)$
- Normal-form proofs of S:

$$Nf(S) = \mu(Pf(Th(S)))$$

the minimal proofs in the deductively closed presentation

Saturated vs. complete presentation

- Saturated: provides all normal-form proofs
- Complete: provides a normal-form proof for every theorem
- They coincide if minimal proofs are unique
  (e.g., total proof ordering)

Example I

$\{a \simeq b, b \simeq c, a \simeq c\}$

Minimal proofs: valley proofs: $s \stackrel{*}{\rightarrow} \circ \stackrel{*}{\leftarrow} t$

- $a \succ b \succ c$
- Complete: $\{b \simeq c, a \simeq c\}$
  with $a \rightarrow c \leftarrow b$ as minimal proof of $a \simeq b$
- Saturated: $\{a \simeq b, b \simeq c, a \simeq c\}$
  with both $a \rightarrow b$ and $a \rightarrow c \leftarrow b$

Example II

$\{a \simeq b, b \simeq c, a \simeq c\}$

Minimal proofs: valley proofs: $s \xrightarrow{*} \circ \xleftarrow{*} t$

- $a \# b$, $a \succ c$, $b \succ c$
- Complete: $\{b \simeq c, a \simeq c\}$
- Saturated: $\{b \simeq c, a \simeq c\}$
  because $a \leftrightarrow b$ not minimal

Canonical presentation

- Contracted: contains all and only the premises of its minimal proofs
- Canonical ($S^\sharp$):
  - Contains all and only the premises of normal-form proofs
  - Saturated and contracted
  - Smallest saturated presentation
  - Simplest presentation

[Nachum Dershowitz and Claude Kirchner]

Equational theories

- Normal-form proof of $\forall\bar{x}\ s \simeq t$:
  valley proof $\hat{s} \stackrel{*}{\rightarrow} \circ \stackrel{*}{\leftarrow} \hat{t}$ by rewriting
  $\hat{s}$ and $\hat{t}$ are s and t with variables replaced by Skolem constants
- Saturated: convergent (confluent and terminating)
- Contracted: inter-reduced
- Canonical: convergent and inter-reduced
- Finite and canonical: decision procedure

Proof-ordering based redundancy

- C redundant in S ($C \in Red(S)$) if adding it does not improve minimal proofs:
  $\mu(Pf(S)) = \mu(Pf(S \cup \{C\}))$
- C redundant in S ($C \in Red(S)$) if removing it does not worsen proofs:
  $S \succsim S \setminus \{C\}$ or $Pf(S) \sqsupseteq Pf(S \setminus \{C\})$

Inference as proof reduction I

$S_0 \vdash S_1 \vdash \ldots S_i \vdash S_{i+1} \ldots$

- Good: $S_i \succsim S_{i+1}$ for all i
- Once redundant always redundant: $S_{i+1} \cap Red(S_i) \subseteq Red(S_{i+1})$

Inference as proof reduction II

$S_0 \vdash S_1 \vdash \ldots S_i \vdash S_{i+1} \ldots$

- Expansion: $A \vdash A \cup B$ with $B \subseteq Th(A)$
- Contraction: $A \cup B \vdash A$ with $A \cup B \succsim A$
- Expansions and contractions are good

Derivations

$S_0 \vdash S_1 \vdash \ldots S_i \vdash S_{i+1} \ldots$

- Saturating: S_∞ is saturated
- Completing: S_∞ is complete
- Contracting: S_∞ is contracted
- Canonical: saturating and contracting

Proof-ordering based fairness I

$S_0 \vdash S_1 \vdash \ldots S_i \vdash S_{i+1} \ldots$

- Whenever a minimal proof of the target theorem is reducible by inferences, it is reduced eventually
- For all $i \geq 0$ and $p \in \mu(Pf(S_i))$
  if there are inferences $S_i \vdash \ldots \vdash S'$ and $q \in \mu(Pf(S'))$
  such that $q < p$
  then there exist $j > i$ and $r \in \mu(Pf(S_j))$ such that $r \leq q$
- Applies to both expansion and contraction
- Contraction is not only deletion

Proof-ordering based fairness II

$S_0 \vdash S_1 \vdash \ldots S_i \vdash S_{i+1} \ldots$

- Critical proof: minimal proof, not in normal form, all proper subproofs in normal form
  (E.g.: peak $\hat{s} \leftarrow \circ \rightarrow \hat{t}$ yielding critical pair)
- $C(S)$: critical proofs of S
- Critical proofs with persistent premises: $C(S_\infty)$
- Fairness: All strictly reduced eventually:
  $C(S_\infty) \sqsupset Pf(\bigcup_{i \geq 0} S_i)$

Uniform fairness

- Trivial proof: made of the theorem itself
- $\widehat{S}$: trivial proofs of S
- Trivial proofs with persistent premises: $\widehat{S_\infty}$
- Uniform fairness: All strictly reduced eventually (unless canonical): $\widehat{S_\infty} \setminus \widehat{S}^\sharp \sqsupseteq Pf(\bigcup_{i \geq 0} S_i)$

Results about good derivations

- If fair then completing
- Uniformly fair iff saturating
- Fairness sufficient for theorem proving (proof search):
  no need to add all consequences of critical proofs
  only enough to provide a smaller proof for each critical proof

Properties of the search plan

- Schedule enough expansion and contraction to be fair hence completing
- Schedule enough contraction to be contracting
- Schedule contraction before expansion: eager contraction

Implementation of contraction

- Forward contraction:
  contract new C wrt already existing clauses: C'
- Backward contraction:
  contract already existing clauses wrt C'
- Implement backward contraction by forward contraction: reducible clause as new clause

Implementation of eager contraction

- $Red(S_i) = \emptyset$ for all i: not if every step is single inference
- $Red(S_i) = \emptyset$ for some i (periodically):
  given-clause loop with $active \cup passive$ inter-reduced
- $Red(B_i) = \emptyset$ for some $B_i \subseteq S_i$ and some i:
  given-clause loop with *active* inter-reduced

Example I: conditional equations

Also conditions rewrite:

$\{a \simeq b \supset f(a) \simeq c,\ a \simeq b \supset f(b) \simeq c\}$

$f \succ a \succ b \succ c$

$a \simeq b \supset f(a) \simeq c$ reduces to $a \simeq b \supset c \simeq c$ which is deleted

Example II

- $a \succ b \succ c$
- $\{a \simeq b \supset b \simeq c, \ a \simeq b \supset a \simeq c\}$ is saturated
- $\{a \simeq b \supset b \simeq c\}$ is equivalent, complete and reduced
- $a \simeq b \supset a \simeq c$ self-reduces to $a \simeq b \supset b \simeq c$ which is subsumed
  or is reduced to $a \simeq c \supset a \simeq c$ which is deleted

Discussion

- Fairness should earn something weaker than saturation
- Proof orderings vs. formula orderings
- Non-trivially fair and eager contracting search plans

References

- Maria Paola Bonacina and Nachum Dershowitz. Canonical ground Horn theories. In Andrei Voronkov and Christoph Weidenbach (Eds.) *Programming Logics: Essays in Memory of Harald Ganzinger*. Springer, Lecture Notes in Artificial Intelligence 7797, 35–71, March 2013.
- Maria Paola Bonacina and Nachum Dershowitz. Abstract canonical inference. *ACM Transactions on Computational Logic*, 8(1):180-208, January 2007.
- Maria Paola Bonacina and Jieh Hsiang. Towards a foundation of completion procedures as semidecision procedures. *Theoretical Computer Science*, 146:199-242, July 1995.
- Maria Paola Bonacina. Distributed Automated Deduction. PhD Thesis, Dept. of CS, SUNY at Stony Brook, December 1992.