Deduction Systems for First-order and Intuitionistic Logic

Lilia Georgieva
lilia@macs.hw.ac.uk

Deduction systems

There are several types of deduction systems. Some of the most important are:

- Sequent calculi: conclusions and premises are treated equally.
- Resolution: oriented towards mechanised reasoning.
- Natural deduction: follows natural style of reasoning.

Today and in the forthcoming lecture we will study resolution for propositional and first-order logic and natural deduction for intuitionistic logic.

First-order Resolution

- First-order logic expresses facts/statements about the world which involve quantification over variables.
- We looked at the building blocks of first-order logic: variables, constants and terms,
- Today: Clausal normal form transformation (as a preprocessing step) and first-order resolution.

Basic concepts

- A literal in propositional logic is a variable or a negation of a variable.
- Clauses are disjunctions of literals.
- The empty clause is denoted by $\bot$.
- A clause set Γ is satisfiable iff there is an I such that $I \models \Gamma$ for all clauses $C \in \Gamma$.

A set of clauses containing $\bot$ is unsatisfiable.

Transformation to clausal normal form

Transformations (in this order):

- Expansion of $\longrightarrow$ and $\leftrightarrow$.
- Convertion to negation normal form.

$$\neg\neg\phi = phi \quad \neg(\phi \wedge \psi) = \neg\phi \vee \neg\psi \quad \neg(\phi \vee \psi) = \neg\phi \wedge \neg\psi$$

- Conversion to conjunctive normal form (pulling $\wedge$ in front of $\vee$).

$$(\psi \wedge \phi_1) \vee \phi_2 = (\psi \vee \phi_2) \wedge (\phi_1 \vee \phi_2)$$

Formulae in conjunctive normal form are identified with a finite set of clauses.

Propositional resolution

- Resolution rule $\frac{\alpha \vee \beta \qquad\qquad \neg\beta \vee \gamma}{\alpha \vee \gamma}$
- Resolution refutation
  - Convert all sentences to CNF.
  - Negate the desired conjecture.
  - Apply resolution rule until either:
    * derive false
    * no more applications of the rule are possible.
  - Factoring: simplify $\phi \vee \psi \vee \psi$ to $\phi \vee \psi$

Terminology $\alpha \vee \beta$ and $\beta \vee \gamma$: resolvents; β: resolved upon.

Note: $\vee$ and $\wedge$ are associative and communitative: the order of litrals does not matter.

Resolution calculus: reflexive transitive closure

Reflexive transitive closure of the resolution inference system *Res*.

$$
\begin{aligned}
Res(\Gamma) &= C \text{where C is a resolvent or factor of clauses in } \Gamma \\
Res^0(\Gamma) &= \Gamma \\
Res^{n+1}(\Gamma) &= Res(Res^n(\Gamma)) \cup Res^n(\Gamma) \\
Res^*(\Gamma) &= \Cup_{n \geq 0} Res^n(\Gamma)
\end{aligned}
$$

Logical properties:

- soundness: If $I \models \Gamma$ then $I \models Res^*(\Gamma)$
- completeness: If $\bot \in Res^*(\Gamma)$ then Γ is unsatisfiable.

Resolution in First-order logic

- The resolution rule:

$$\frac{\boldsymbol{\alpha} \vee \boldsymbol{\phi}\ \text{(rename)} \qquad \neg\boldsymbol{\psi} \vee \boldsymbol{\beta}\ \text{(rename)}}{(\boldsymbol{\alpha} \vee \boldsymbol{\beta})\boldsymbol{\theta}}$$

  - $\boldsymbol{\alpha}$, $\boldsymbol{\phi}$, $\neg\boldsymbol{\psi}$, $\boldsymbol{\beta}$- first-order formulae.
  - $\boldsymbol{\theta}$-unifier (substitution).

Example

Lets try to resolve the following:

$$\frac{P(x) \vee Q(x,y) \qquad \neg P(A) \vee R(B,z)}{(Q(x,y) \vee R(B,z))\boldsymbol{\theta}}$$

- We can unify $P(x)$ and $\neg P(A)$.
- The unifier is $\boldsymbol{\theta} = \{x/A\}$
- We have to apply the substitution to the end result.
- The result is $Q(A,y) \vee R(B,z)$.

Renaming of variables

- All variables are implicitly universally quantified.

$$\forall xy P(x) \vee Q(x, y) \forall z P(A) \vee R(B, z)$$

- The scope of the quantifier is local to each clause.
- Renaming keeps the variables distinct.

$$\forall x_1 y P(x) \vee Q(x_1, y) \forall x_2 P(A) \vee R(B, x_2)$$

- $\boldsymbol{\theta} = \{x_1/A\}$

Resolution

- Inputs are sentences (clauses) in conjunctive normal form with no explicit quantifiers (implicit universal quantifiers).
- How do we eliminate existential quantifiers?
- First, we convert the sentences into clausal normal form.

Converting to clausal normal form

1. Eliminate $\longrightarrow$ and $\leftrightarrow$.

$$\boldsymbol{\alpha} \longrightarrow \boldsymbol{\beta} \quad \neg\boldsymbol{\alpha} \vee \boldsymbol{\beta}$$

2. Push the negation inwards.

$$\begin{array}{ll} \neg(\boldsymbol{\alpha} \vee \boldsymbol{\beta}) & \neg\boldsymbol{\alpha} \wedge \neg\boldsymbol{\beta} \\ \neg(\boldsymbol{\alpha} \wedge \boldsymbol{\beta}) & \neg\boldsymbol{\alpha} \vee \neg\boldsymbol{\beta} \\ \neg\neg\boldsymbol{\alpha} & \boldsymbol{\alpha} \\ \neg\exists x P(x) & \forall x \neg P(x) \\ \neg\forall x P(x) & \exists x \neg P(x) \end{array}$$

3. Rename the variables apart so that every quantifier binds a different variable.

$$\forall x \exists y (P(x) \longrightarrow \forall x Q(x, y))$$

$$\forall x_1 \exists y_2 (P(x_1) \longrightarrow \forall x_3 Q(x_3, y_2))$$

Converting to clausal normal form. Skolemization

Skolemize.

- Substitute brand new name for each existentially quantified variable.

$$\exists x.P(x) \implies P(a)$$

- Different variables are given different names.

$$\exists xyR(x, y) \implies R(a, b)$$

- The name of the same variable persists.

$$\exists xF(x) \wedge Mx) \implies F(a) \wedge M(a)$$

- One more case:

$$\forall x.\exists y.R(x, y)$$

$$\exists y.\forall x.R(x, y)$$

Example

- One more case:

$$\forall x.\exists y.R(x, y)$$

$$\exists y.\forall x.R(x, y)$$

- Substitute a new function of all universally quantified variables in enclosing scopes for each existentially quantified variable (in that case the substitution function for y in the second sentence is a function of x, e.g. $f(x)$.

$$\forall x.\exists y.R(x, y) \longrightarrow \forall x R(x, f(x))$$

$$\exists y.\forall x.R(x, y) \longrightarrow \forall x R(x, a)$$

- Next step: drop the universal quantifiers.
- Finally, convert to conjunctive normal form.
- Rename the variables in each clause.

Example: converting to clausal normal form

- $$\exists x P(x) \wedge R(a, x)$$
  - For this sentence we start with Skolemization because the preceding steps are unnecessary.
  $$P(b) \wedge R(a, b)$$
  - Result: two clauses $\{P(b), R(a, b)\}$
- $$\forall x(\exists y(P(y) \wedge R(x, y)) \implies Q(x)$$
  - Eliminate the arrow, push in the negation, no existential quantifiers=no Skolemization.

Example Resolution proof

1	a	$P(a)$
2	a	$R(b, a)$
3	b	$\neg P(y) \vee R(x, y) \vee Q(x)$
4	c	$\neg Q(x) \vee \neg A(y) \vee K(x, y)$
5	d	$\neg K(a, b) \vee K(e, b)$
6	e	$\neg C(b)$
7	f	$\neg C(x) \vee A(x)$
8	Neg	$\neg K(e, b)$
9	$(5, 8)$	$\neg K(a, b)$
10	$(6, 7)$	$A(b)$
11	$(4, 9)$	$\neg Q(a) \vee \neg A(b)$
12	$(10, 11)$	$\neg Q(a)$
13	$(3, 12)$	$\neg P(y) \vee R(a, y)$
14	$(13, 2)$	$\neg P(a)$
15	$\perp$	$(14, 1)$

Proving validity

How do we use resolution refutation to prove that something s valid?

- Normally we prove that a sentence is entailed by a set of axioms.
- Valid sentences are entailed by the empty set of sentences.
  - ϕ is valid.
  - $\{\} \models \phi$ empty set of sentences entails ϕ.
  - $\{\} \vdash \phi$ empty set of sentences proves ϕ.
- To prove validity by refutation negate the sentence and try to derive contradiction.

Proving validity: Example

Prove validity of

$$\exists x(P(x) \longrightarrow P(A) \wedge P(x) \longrightarrow P(B))$$

Steps:

- Negate the sentence.
- Convert the sentence in clausal normal form.
- You get
$$\forall x(P(x) \wedge \neg P(A)) \vee (P(x) \wedge \neg P(B))$$
- Distribute the logical connectives.
- You get the four clauses:
$$P(x) \vee P(x), P(x) \vee \neg P(B), \neg P(A) \vee \neg P(B), \neg P(A) \vee \neg P(B)$$
.
- Use resolution to derive contradiction.

Completeness and decidability

Complete: IF $KB \models \alpha$ then $KB \vdash \alpha$

- If it is entailed there is a proof.

Semi-decidable:

- If there is a proof we will halt with it.
- If not, maybe halt, maybe not.

Gödel's completeness theorem: There exists a complete proof system for FOL.
Robinson's Completeness Theorem: Resolution refutation is a complete proof system for FOL.
FOL is semi decidable: if the desired conclusion follows form the premises then eventually the resolution refutation will find a contradiction.