Example

- If $\neg c \models \bot$,
  - and hence would need to make $\bot$ false
  - Hence would make $q$ and $\neg q$ false
  - This would also make $a$ and $\neg a$ false
  - $\bot$ interpretation makes $c$ true
  - Can conclude that $c$ is false in all models of $\text{KB}$
  - If an interpretation makes $c$ true, it would also make $a$ and $b$ true, hence would need $a \land b$ true and hence would need $\bot$ true contradiction
  - So $\text{KB} \models \neg c$

Integrity Constraints (Chapter 7.3)

- Integrity Constraint
  - $\bot \leftarrow a_1 \land \ldots \land a_n$
  - Means that $a_1 \land \ldots \land a_n$ cannot be true
  - A model of $\text{KB}$ must make each clause true
  - If it makes the body true, then 'false' must be true
  - Which is a contradiction, so body cannot be true
  - $\bot \leftarrow a$ means that $a$ has to be false in all models

Horne Clause

- Is either a clause or an integrity constraint
- This is a clause in an integrity constraint
- $\bot \leftarrow a$ means that $a$ must be false in all models

Overview

- Disjunctive & Negative Knowledge
- Resolution Rule
- Bottom-Up
- Proof by Resolution
- Top-Down
- Inference Constraints
Syntax and Semantics of Or

- Add $\neg$ to syntax
- Semantics
  - If interpretation $I$ and variable assignment $U$ make $a \lor b$ true, then it makes $a \lor b$ true.

Syntax and Semantics of Not

- Add $\neg$ to syntax
- Semantics
  - If interpretation $I$ and variable assignment $U$ make $a$ true, then it makes $\neg a$ false, and vice versa.

Variations on Integrity Constraints

- Horne clause
  - Can be written as $\neg a \leftarrow a_1 \land \ldots \land a_n$
  - Can be written as $\neg a \leftarrow a_1 \lor \ldots \lor a_n$
  - Form: $a \leftarrow a_1 \land \ldots \land a_n$
  - Form: $a \leftarrow a_1 \lor \ldots \lor a_n$

- For any $d$
  - $d \leftarrow a$ if $d$ is true in all models of $KB$
  - $d \leftarrow \neg a$ if $d$ is false in all models of $KB$
Overview

• Integrity Constraints
  ⇒ Disjunctive & Negative Knowledge

• Resolution Rule

• Bottom-Up

• Proof by Refutation

• Top-Down

• Bottom-Up

• Resolution Rule

• Integrity Constraints

Horne Clauses

• Integrity constraints can be used for diagnostics

• Integrity constraints are used to prove negative facts

• Example: From electrical domain: a light can't be both on and off

• Integrity constraints can be used for diagnostics

Unsatisfiable

• Integrity constraints means there might not be a model of a KB

• KB is unsatisfiable

Example KB

false → a

false

Example KB

false → a

false

Example KB

false → a

false

Example KB

false → a

false
Example

$$a \land (b \lor c \lor \neg(d \leftarrow e))$$

$$a \land (b \lor c \lor \neg(d \lor \neg e))$$

$$a \land (b \lor c \lor \neg(d \land \neg e))$$

$$a \land (b \lor c \lor \neg d \land \neg e)$$

$$a \land (b \lor (c \lor \neg d \land e))$$

$$a \land (b \lor (c \lor \neg d \land e))$$

$$a \land (b \lor (c \lor \neg(d \land e)))$$

Conversion to Conjunctive Normal Form

• Any expression with $\land$, $\lor$, $\neg$, and $\leftarrow$ can be converted into a set of clauses in conjunctive normal form - no literals on right hand side of $\leftarrow$

• Step 1: Eliminate $\leftarrow$ - $(\phi \leftarrow \psi)$ replaced with $(\phi \lor \neg \psi)$

• Step 2: Distribute negation so only applies to atoms - $\neg \neg \phi$ replaced with $\phi$, $\neg(\phi \lor \psi)$ replaced with $\neg \phi \land \neg \psi$, $\neg(\phi \land \psi)$ replaced with $\neg \phi \lor \neg \psi$

• Step 3: Distribute $\land$ over $\lor$ - $(\phi \lor (\psi \land \chi))$ replaced by $((\phi \lor \psi) \land (\phi \lor \chi))$

Disjunctive Knowledge

In Conjunctive Normal Form

- Positive literals
- Negative literals

<table>
<thead>
<tr>
<th>Datalog</th>
<th>Exactly 1</th>
<th>any number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horne</td>
<td>At most 1</td>
<td>any number</td>
</tr>
<tr>
<td>Disjunctive &amp; Negative</td>
<td>any number</td>
<td>any number</td>
</tr>
</tbody>
</table>

- Remove restriction of Horne clauses
- In fact allow any combination with $\land$, $\lor$, $\neg$, and $\leftarrow$ and no restrictions on how many positive heads
- Can be converted into conjunctive normal form
- Allow atoms to be negated
- $a \leftarrow q \lor r$
- Allow disjunction in head
- In Conjunctive Normal Form

<table>
<thead>
<tr>
<th>Distinctive &amp; Negatives</th>
<th>At most 1</th>
<th>Exactly 1</th>
<th>Any number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Horne</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Negative Heads</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>In Conjunctive Normal Form</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Overview

• Integrity Constraints
• Disjunctive & Negative Knowledge
⇒ Resolution Rule
• Bottom-Up
• Proof by Refutation
• Top-Down

Variables

• Recap:

$$(a \lor (b \land c)) \land (a \lor b) \land (a \lor c)$$

Knowledge base:

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

Or in set notation for disjuncts:

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

• Now with variables:

$$\text{male}(X) \lor (\text{female}(X) \land \text{ownsdog}(X)) \land (\text{male}(X) \lor \text{ownsdog}(X))$$

- If we show the implicit quantifier $\forall X$:

$$\forall X \text{male}(X) \lor \text{female}(X) \land \text{male}(X) \lor \text{ownsdog}(X)$$

- Variables in two disjuncts seem bound together

+ Can we write it as two separate clauses, each with own $X$?

+ Yes, means the same thing:

$$\forall X \text{male}(X) \lor \text{female}(X) \land \forall X \text{male}(X) \lor \text{ownsdog}(X)$$

- So, just as in the non-variable case, we can write this as:

$$\{\{\text{male}(X)\}, \{\text{female}(X)\}\} \land \{\{\text{male}(X)\}, \{\text{ownsdog}(X)\}\}$$

Conjunctive Normal Form

• Convert all clauses into Conjunctive Normal Form

- Our KB is now a set of clauses:

$$\{\{\}, \{\neg p, q\}, \{q, r\}\}$$

Knowledged base: $\{q, r\}$

Recap: $\neg a \lor b$
Overview

• Integrity Constraints
• Disjunctive & Negative Knowledge
• Resolution Rule
• Top-Down
• Proof by Resolution

Examples

\{a, \neg b, \neg c\} \text{ with } \{d, \neg c, b\}

\{a(X), b(Y, Z), \neg c(Y)\} \text{ with } \{\neg b(Z, Z), c(a)\}

Resolution Rule

• Resolution Rule - Resolvent includes \neg a
• Resolvent includes \neg b and applied
• \neg a \text{ is in } \neg a \text{ and } \neg b
• a and b cannot be unified
• Resolution Rule

Resolution Rule
Example I

false ← a ∧ b

a ← c

b ← c

¬ c

Queries to Bottom-Up Proof Procedure

• Query can be disjunction of positive or negative literals

- For previous bottom-up procedure, was just a single positive literal

• Write query as a set of literals

- If query contains A and ¬ A obviously true + Since one of them is true in any model

- If query contains A ∨ ¬ A obviously true + If query contains a set of literals

- For previous bottom-up procedure, was just a single positive literal

Query can be disjunction of positive or negative literals

Bottom Up Proof Procedure

- Find set of minimal truths in KB

- Can have any arbitrary clause in conjunctive normal form

- Consistency set no longer just has atoms in it + If there are any A ∈ C such that A ⋀ ¬ A true, remove A since now implied by A + If there is any A ∈ C such that A ⋀ ¬ A true, remove A since already implied + If it contains A and ¬ A, drop ¬ A since already true + Apply resolution rule if you can derive A + Add two clauses from C + Repeat

Set of KB, i.e.

Bottom Up Proof Procedure (Section 7.5)
Horne Clauses and Resolution (not in textbook)

What is so special about Horne clauses?
- More powerful than Datalog
- And an efficient search solution for false

If $\text{KB} | \models \{\}$
- Where $\{\}$ represents the empty clause, which is the same as false
- Which means $\text{KB}$ has no model, which means it is inconsistent
- If this is the case, what would happen if we add $\neg b$?

And an efficient search solution for false
- More powerful than Datalog

What is so special about Horne clauses?

Overview

- Integrity Constraints
- Disjunctive & Negative Knowledge
- Resolution Rule
- Top-Down
  - Proof by Resolution
  - Bottom-Up
  - Resolution Rule

Example II

$\text{KB} = (a \lor \neg b) \leftarrow c$
$\neg e \leftarrow \neg c$
$b \lor d$
$(a \lor b) \leftarrow d$
$e \leftarrow \neg a$
$a \lor c$

$\neg e$
Overview

- Integrity Constraints
- Disjunctive & Negative Knowledge
- Resolution Rule
- Bottom-Up
- Proof by Refutation

⇒ Top-Down
- Proof by Resolution
- Unit Resolution

We say it is resolution complete if the number of resolvents the unit resolution can find
- First number of heads
- Second number of heads
- Results may have fewer than k literals
- Last clause cannot be the same as the literals
- Proof - Unit Resolution always halts

This is a restricted bottom-up proof procedure
Pick two resolvents where one of them is a unit clause

Inconsistencies

Given a KB, there is an efficient way to see if it is inconsistent if
\[ |\{b_{-}\} \cap |D_{\neg} \| = \text{false} \]

Proof by Refutation
Negative Ancestor Rule

- Can view proof as adding original answer clause KB and trying to prove yes by itself
- So, should be sound to resolve answer clause with a previous answer clause
- Didn’t need to do this for Datalog, as it did not need this to make proof procedure complete
- But we do need this ability here

Proof

\[ \text{Now we'll:} \]
\[ q \lor \neg r \]
\[ q \lor \neg s \]
\[ q \lor \neg t \]
\[ \neg q \lor \neg r \]
\[ \neg q \lor \neg s \]
\[ \neg q \lor \neg t \]

Proof

\[ q \lor \neg r \]
\[ \text{Turn into disjunctive normal form:} \]
\[ q \lor \neg r \]
\[ \neg q \lor \neg s \]
\[ \neg q \lor \neg t \]
\[ \neg q \lor \neg r \]
\[ \text{Consider KB:} \]

A problem

- Stop when just ‘yes’ in answer clause
- Mark first non-yes literal in answer clause
- Use resolution rule to derive new answer clauses
- Turn into disjunctive normal form
- For clauses top-down procedure: if is conjunction of positive literals, substitute A with true

Top Down Proof Procedure
Example

\[ \text{Proof} \]

<table>
<thead>
<tr>
<th>( (X) )</th>
<th>( \text{KB} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P(X) )</td>
<td>( q(X) )</td>
</tr>
<tr>
<td>( (X) )</td>
<td>( \neg q(X) )</td>
</tr>
</tbody>
</table>

\[ \text{Answer Clause} \]

\[ \text{KB} \]

\[ \text{New KB} \]

\[ \text{Proof} \]

Solution

\[ \text{KB} \]

\[ \text{New KB} \]