Knowledge Compilation and Theory Approximation

Henry Kautz and Bart Selman
Presented by Kelvin Ku
kelvin@cs.toronto.edu
Overview

- Problem
- Approach
- Horn Theory and Approximation
- Computing Horn Approximations
- Empirical Results
- Generalizations
Problem Formulation

• Goal: Inference in clausal propositional KBs
  – \( \Sigma \) is the KB/theory, \( \alpha \) is a CNF formula
  – inference is checking \( \Sigma \models \alpha \)

• NP-complete (reduction to SAT)

• Alternatives
  – Restricted language: limited expressiveness
  – Incomplete inference, either resource-bounded or non-traditional inference: inconclusive results
Approach

- Basis: Compile (rewrite) KB into tractable language
- Not always possible, e.g. with incomplete languages
- Compromise: approximate KB using tractable language
KB Approximation

• Approximate $\Sigma$ using $\Sigma_{lb}$ and $\Sigma_{ub}$ such that
  – inference in $\Sigma_{lb,ub}$ is fast
  – bound the original theory: $\Sigma_{lb} \models \Sigma \models \Sigma_{ub}$
  – in other words, $\mathcal{M}(\Sigma_{lb}) \subseteq \mathcal{M}(\Sigma) \subseteq \mathcal{M}(\Sigma_{ub})$
    • $\mathcal{M}(\Sigma)$: models of $\Sigma$

• $\Sigma_{glb}$ is greatest/weakest lower-bound
  – $\neg \exists \quad \Sigma' \cdot \mathcal{M}(\Sigma_{glb}) \subset \mathcal{M}(\Sigma') \subset \mathcal{M}(\Sigma)$

• $\Sigma_{lub}$ is least/strongest upper-bound
  – $\neg \exists \quad \Sigma' \cdot \mathcal{M}(\Sigma) \subseteq \mathcal{M}(\Sigma') \subset \mathcal{M}(\Sigma_{lub})$

• Transitivity of $\models$ leads to fast querying scheme

...
Fast Querying Scheme

\[ \Sigma \models \alpha ? \]

- If \( \Sigma_{ub} \models \alpha \):
  - Return "yes"
- If \( \Sigma_{lb} \not\models \alpha \):
  - "yes" → Return "no"
- Return "don’t know" or fall back to \( \Sigma \)
Horn Theory

- Form: Subset of CNF with at most one positive literal per clause
  - e.g. clause: \((\neg p \lor \neg q \lor r) \equiv (p \land q \Rightarrow r)\)
  - Natural form for some domains, e.g. Prolog rules

- Inference: Linear in size (number of literals) of KB and query

- Incomplete: e.g. cannot represent \(p \lor q\) as a Horn clause (since it has two models)
Horn Approximation Example

• Non-Horn theory $\Sigma$:
  
  \[ (\neg a \lor c) \land (\neg b \lor c) \land (a \lor b) \]

• e.g. Horn LB: $a \land b \land c$

• e.g. Horn GLBs: $a \land c$, $b \land c$

• No Horn theory $\Sigma' \neq a \land c$ such that
  
  \[ (a \land c) \models \Sigma' \models \Sigma \]

• e.g. Horn UB: $\neg a \lor c \land (\neg b \lor c)$

• Horn LUB: $c$

• No Horn theory $\Sigma' \neq c$ such that $\Sigma \models \Sigma' \models c$
Computing Horn Approximations

• Worst-case approximation time is $O(2^{|\Sigma|})$
  – either $|\text{approx}|$ is $O(2^{|\Sigma|})$ or
  – $|\text{approx}|$ is $O(|\Sigma|^n)$ and computation time is $O(2^{|\Sigma|})$

• Reasonable trade-off if we do many queries

• Compromise: incremental “anytime” approximation
Computing the Horn GLB

• Basis of method: Horn-strengthening
  – def: weakest Horn-clause subsuming a clause wrt. a positive literal
  • e.g. \{ \neg p, q, r \} has Horn-strengthenings \{ \neg p, q \} and \{ \neg p, r \}
  – so remove all but one positive literal
• Lemma 1: If a Horn theory entails clause \( \alpha \), then it entails some Horn-strengthening of \( \alpha \)
• Lemma 2: Every GLB of a theory is equivalent to some Horn-strengthening of the theory
• So searching through strengthenings for each clause will obtain the GLB; gives rise to algorithm ...

Generate_GLB

Input: a set of clauses $\Sigma = \{C_1, C_2, \ldots, C_n\}$
Output: a Horn GLB of $\Sigma$

begin
    L := first Horn-strengthening of $\Sigma$

    loop
        L' := next Horn-strengthening of $\Sigma$
        if none exists then exit loop
        if $L \vDash L'$ then $L := L'$ /* found weaker LB */
    end loop

    remove subsumed clauses from L

    return L
end
Computing the Horn LUB

• Basis of method: Prime implicate
  – A strongest clause implied by $\Sigma$
  – In other words: $\Sigma \models C$ and $\neg \exists C' \subset C \cdot \Sigma \models C'$

• Horn LUB $\equiv$ all Horn prime implicates of $\Sigma$

• Thus, naive Generate_LUB: Compute all resolutions in $\Sigma$, and collect Horn prime implicates
  – resolution is complete
Size of Approximations

• $|GLB| \leq |\Sigma|$ (recall Generate_GLB)

• Theorem: There exist theories $\Sigma$ such that $|LUB| \in O(2^{|\Sigma|})$, so LUB is EXP in the worst case

• Compromise: Theory Compaction
  – Transform original theory to obtain relatively smaller LUB
  – Theorem: compaction can’t always help either
Empirical Results

• Hypothesis: Fast querying with approximations can efficiently answer queries that are intractable in the original theory

• Theoretical Motivation
  – Finding model of theory with unique model is intractable
  – Probablistic analysis of inference in hard random theories
Empirical Results

• Experiment: Compare execution time and coverage against Davis-Putnam on hard random theories

• Results
  – All queries answered
  – Total time (bounds and querying) significantly better than DP
Generalizations

- Parameters: languages of original theory, approximation, and query
- Formally, elements of framework are
  - $\mathcal{L}, \models, \mathcal{L}_S, \mathcal{L}_T, \mathcal{L}_Q, f_{L,U} : \mathcal{L}_S \times \mathbb{N} \rightarrow \mathcal{L}_T$
- For all classes of propositional clauses $\theta$
  - $\checkmark$ Generate_GLB if (i) $\theta$ is closed under resolution and (ii) every clause is subsumed by a $\theta$ clause
  - $\checkmark$ Generate_LUB if (iii) $\theta$ is closed under subsumption
Alternative Clausal Languages

• Satisfying (i), (ii), and (iii)
  – Reverse-horn: \{p \lor q \lor \neg r\}
  – Binary: \{p \lor \neg q, r\}
  – Unit: \{p, \neg q, r\}

• Requiring modification to compilation algorithms:
  – k-Horn: Horn w/at most k literals per clause (violates (i), LUB has polynomial size)
Alternative Logics

• First-Order Logic
  – In general, GLB in ground clauses is not well-defined
    • e.g. for $\exists x. P(x)$ exists an infinite series of better LBs
  – Special case: first-order clauses (prenex, universal, clausal body) have Horn GLBs
    • e.g. $A(p) \lor B(q) \lor \neg C(r) \iff A(p) \lor \neg C(r)$
  – Exist finite first-order clausal theories with no finite Horn LUB (even without function symbols)

• Description Logic (concepts, subsumption)
  – Exists a tractable subset of $\mathcal{FL}$, $\mathcal{FL}^-$
  – Compute and store tractable subsumption bounds for each concept
Related Work

• Darwiche and Marquis paper considers compiling to complete languages with polytime entailment
  – Horn is incomplete but permits anytime approx
  – Only considering CE here
  – D&M concerned with more general queries and theory transformation