Planning Graphs and Knowledge Compilation

Héctor Geffner
ICREA and Universitat Pompeu Fabra
Barcelona, SPAIN
6/2004

Planning as SAT (Kautz and Selman)

- **Encode:** Map Strips problem P with horizon n into a propositional theory T
- **Solve:** Using a SAT solver, determine if T is consistent, and if so, find a model
- **Decode:** Extract plan from model

Our goal

Use of propositional logic for **defining and computing lower bounds for planning** (admissible heuristics)

- understand the **planning graph** construction as a **precise form of inference**
- exploit account to uncover **relations** (e.g., to variable elimination) and introduce **generalizations** (e.g., incomplete information)

Strips Refresher

- A **problem** in Strips is a tuple $\langle A, O, I, G \rangle$ where
  - A stands for set of all **atoms** (boolean vars)
  - O stands for set of all **operators** (ground actions)
  - $I \subseteq A$ stands for **initial situation**
  - $G \subseteq A$ stands for **goal situation**

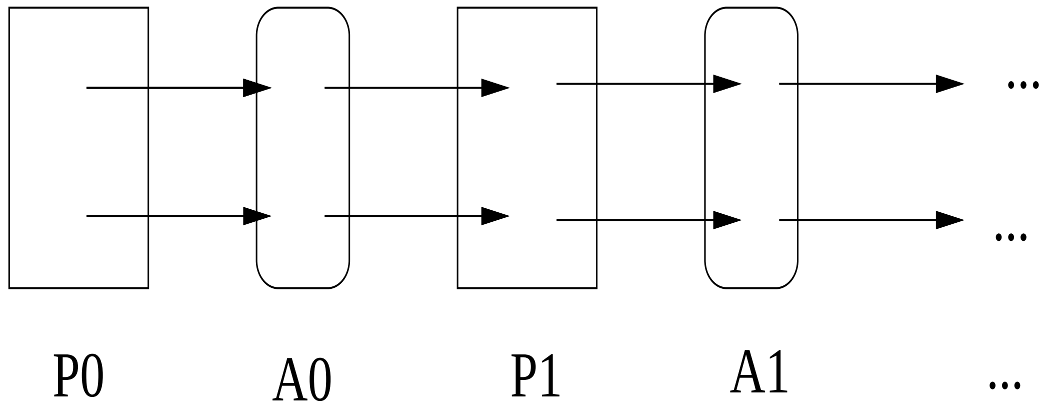
- The **operators** $o \in O$ **represented** by three lists

  -- the **Add** list $Add(o) \subseteq A$
  -- the **Delete** list $Del(o) \subseteq A$
  -- the **Precondition** list $Pre(o) \subseteq A$

- The **task** is to find a plan: a sequence of applicable actions that maps I into G . . .

Lower Bounds and Planning Graphs

- Build graph with layers P_0, A_0, P_1, A_1, . . . where



$$P_0 = \{p \in s\}$$
$$A_i = \{a \in O \mid Prec(a) \subseteq P_i\}$$
$$P_{i+1} = \{p \in Add(a) \mid a \in A_i\}$$

- Graph represents **lower bound** for achieving G from s:

$$h_{max}(s) = \text{min } i \text{ such that } G \subseteq P_i$$

Need No-op(p) action for each p: $Prec = Add = \{p\}$

More Informed h in Graphplan

- Planning graph in Graphplan also keeps track of **pairs** that cannot be reached **simultaneously** in i steps, $i = 0, 1, \ldots$
  - **action** pair **mutex** at i if incompatible or preconditions **mutex** at i
  - **atom** pair **mutex** at $i+1$ if supporting action pairs all **mutex** at i

- **Mutexes** computed along with planning graph and yield **more informed** admissible h

$$h_G(s) \stackrel{\text{def}}{=} \text{min } i \text{ s.t. } G \subseteq P_i \quad \text{and} \quad G \text{ not mutex at } i$$

Graphplan is an IDA* regression solver driven by this heuristic

Lower Bounds crucial in Planning and Problem Solving

- LBs explain performance gap between Graphplan and predecessors

- In **SAT/CSP** planning models, LBs represent implicit constraints that speed up the search:

  SAT/CSP approaches to planning indeed do not encode the planning problem directly but its **planning graph**

- Our main goal in this work: understand **derivation of these LBs or implicit constraints in the planning graph as a precise form of inference**

Deductive Inference and Lower Bounds for Planning

- Consider following heuristic h where T encodes Strips problem with horizon n **without the goal**

$$h(G) \stackrel{\text{def}}{=} \text{ min } i \leq n \quad \text{such that } T \not\models \neg G_i$$

i.e., $h(G)$ encodes first time i at which goal G consistent with T

- Such h is well defined
  - Good news: h very **informative**; indeed $h(G) = h^*(G)$ (optimal)
  - Bad news: h **intractable**

Deductive Inference and Lower Bounds (cont'd)

Consider now approximation h_Γ given by sets $\Gamma_0, \ldots, \Gamma_n$ of **deductive consequences** of T at the various time points $0, \ldots, n$:

$$h_\Gamma(G) \stackrel{\text{def}}{=} \text{min } i \leq n \quad \text{such that } \Gamma_i \not\models \neg G_i$$

- If sets $\Gamma_i = \emptyset$, then $h_\Gamma(G) = 0$ (non-informative)
- If sets $\Gamma_i = PI_i(T)$, then $h_\Gamma(G) = h(G)$ (intractable)
- Always $0 \leq h_\Gamma \leq h$

Question: how to define sets Γ_i so that resulting LBs are **informative and tractable**?

($PI_i(T) =$ *prime implicates of* T *at time* i)

Prime Implicates and Lower Bounds: First attempt

Stratify Strips theory T (without the goal) as

$$T = T_0 \cup T_1 \cup \cdots \cup T_m$$

Define sequence of sets Γ_i iteratively as

$$\Gamma_0 \stackrel{\text{def}}{=} PI_0(T_0)$$

$$\Gamma_{i+1} \stackrel{\text{def}}{=} PI_{i+1}(\Gamma_i \cup T_{i+1})$$

It follows that no info lost in iteration, and same sets and h result:

$$\Gamma_i = PI_i(T)$$

$$h_\Gamma = h = h^*$$

But then computation of h_Γ remains **intractable** . . .

Prime Implicates and Tractable Lower Bounds

Define sequence of sets Γ_i iteratively as

$$\Gamma_0 \stackrel{\text{def}}{=} PI_0^k(T_0)$$
$$\Gamma_{i+1} \stackrel{\text{def}}{=} PI_{i+1}^k(\Gamma_i \cup T_{i+1})$$

for a fixed $k = 1, 2, \ldots$, where $PI_i^k(T)$ stands for set of **prime implicates of T at time i with size no greater than k**

Key result: We show in paper that for **Strips theories T**

- sequence of Γ_i sets and h_Γ informative **and** tractable
- h_Γ **equal to Graphplan** h_G for $k = 2$, and
- $x \in Layer_i$ iff $\neg x_i \notin \Gamma_i$ AND $(x, y) \in Layer_i$ iff $\neg x_i \vee \neg y_i \in \Gamma_i$

where $x \in Layer_i$ and $(x, y) \in Layer_i$ stand for atom and mutex pair in layer i of planning graph

General Framework: Stratified Theories

Propositional theories T defined over indexed variables $x_i \in L_i$, $0 \leq i \leq m$, that can be expressed as union of subtheories T_0, . . . , T_m where

- T_0 made up of clauses $C_0 \in L_0$
- T_{i+1} made up of clauses $C_i \vee C_{i+1}$, where $C_{i+1} \in L_{i+1}$ and $C_i \in L_i$ (C_{i+1} non-empty)

Example: Stratified theory for Strips with horizon n

1. **Init** T_0**:** p_0 for $p \in I$, and $\neg q_0$ for $q \in A$ not in I
2. **Action Layers** T_{i+1}**:** for $i = 0, 2, \ldots, n-2$
   - $p_i \vee \neg a_{i+1}$ for each $a \in O$ and $p \in pre(a)$
   - $\neg a_{i+1} \vee \neg a'_{i+1}$ for interfering a, a' in O
3. **Propositional Layers** T_{i+1}**:** for $i = 1, 3, \ldots, n-1$
   - $\neg a_i \vee p_{i+1}$ for each $a \in O$ and $p \in add(a)$
   - $\neg a_i \vee \neg p_{i+1}$ for each $a \in O$ and $p \in del(a)$
   - $a_i^1 \vee a_i^2 \vee \cdots \vee a_i^{np} \vee \neg p_{i+1}$ for each $p \in A$

Tractable PI-k Inference over Stratified Theories

Three conditions guarantee that the **iterative computation of prime implicates of bounded size** remains **tractable** for stratified theories T:

$$\Gamma_0 \stackrel{\text{def}}{=} PI_0^k(T_0)$$

$$\Gamma_{i+1} \stackrel{\text{def}}{=} PI_{i+1}^k(\Gamma_i \cup T_{i+1})$$

1. T is **compiled:** resolvents over variables x_{i+1} in T_{i+1} subsumed in T
2. T has **bounded support width:** number of clauses $C_i \vee C_{i+1}$ in T_{i+1} with common literal $l_{i+1} \in C_{i+1}$ and body $|C_i| > 1$, bounded
3. T is **pure:** only x_{i+1} or $\neg x_{i+1}$ occur in T_{i+1}

- Stratified **Strips** theories are **compiled**, have **support width 1**, and can easily be made **pure** (3. not needed for $k \leq 2$)
- Paper contains **sound algorithm** for computing Γ_i sets that under conditions 1--3 is **complete** and **polynomial**

Graphplan vs. Variable Elimination and Variations

- **Variable Elimination** is a family of algorithms for solving SAT, CSPs, Bayesian Networks, etc (Dechter et al) that follows the pattern of **gaussian elimination** for solving linear equations

- Given a theory $T = T_0$ over variables $x_0, \ldots, x_n$
  - **Forward pass:** eliminate var x_i from T_i resulting in theory T_{i+1} over $x_{i+1}, \ldots, x_n$, $0 \leq i < n$
  - **Backward pass:** Solve theories T_n, T_{n-1}, . . . , T_0 in order, each for a single variable; result is a model (if T is satisfiable)

-- **Good:** backward pass (solution extraction) is **backtrack free**

-- **Bad:** forward pass (elimination pass) is **exponential** in time and space

Alternative 1: Bounded-k Variable Elimination

- Restricts size of constraints induced by elimination of vars to k
- Elimination sound but not complete; performs in **polynomial time** (removes some but not all backtracks)

Alternative 2: Bounded-k Block Elimination

- Eliminates **blocks** of vars in one-shot, inducing constraints of size $\leq k$ only
- Stronger than Bounded-k Var Elimination, but **exponential in size of blocks**

Graphplan and Bounded-k Elimination

As a corollary of earlier results we get that:

- For **Strips theories**, Bounded-k Block Elimination is **polynomial** in the size of the blocks (blocks are the sets of vars in same layer)
- Graphplan actually does a **Bounded-2 Block Elimination** pass foward **exactly**, followed by a backward Backtrack Search
- Thus Graphplan fits nicely in the variable elimination framework, where it exploits the **special structure of Strips theories**

Negative vs. Positive Deductive Lower Bound

LB scheme based on proving **negation of the goal**

$$h(G) \stackrel{\text{def}}{=} \text{min } i \leq n \quad \text{such that } T \not\models \neg G_i$$

$h(G)$ is a LB because

if $\exists$ Plan that achieves G in $m \leq n$ steps,
then $\exists$ M of $T \wedge G_m$,
then $T \not\models \neg G_m$

Question: Can we define LBs based on the **proving the goal itself**, possibly from transformed theory T^+?

$$h^+(G) \stackrel{\text{def}}{=} \text{min } i \leq n \quad \text{such that } T^+ \models G_i$$

A Positive Deductive Lower Bound

- Define T^+ as Strips encoding (without the goal) **but** with
  - deletes removed
  - all possible actions applied: $prec(a)_i \supset a_i$

- Then it turns out
  - T^+ consistent and tractable
  - $T+ \models G_i$ iff $\Gamma_i \not\models G_i$ for $k = 1$

- Thus
  - Positive and Negative LBs coincide for $k = 1$
  - Positive LBs weaker than negative ones for $k > 1$
  - Nonetheless former useful in non-Strips settings . . .

Positive Deductive LBs when Information is Incomplete

$$h^+(G) \stackrel{\text{def}}{=} \min\ i \leq n \quad \text{such that } T^+ \models G_i$$

- With incomplete info, test $T^+ \models G_i$ **intractable**
- Still heuristic h^{++} defined as

$$h^{++}(G) \stackrel{\text{def}}{=} \min\ i \leq n \quad \text{such that } T^{++} \models G_i$$

  for any theory T^{++} **stronger** than T^+ remains a **LB**
- Thus **tractable LB** can be obtained by mapping T^+ into **stronger and tractable** T^{++}
- So 'bounds' in Planning and Knowledge Compilation (Kautz and Selman) related after all . . .
- Indeed, h used in Brafman-Hoffmann ICAPS 04, can be understood in terms of a compilation of T^+ into a 2-CNF theory T^{++} (which is not necessarily the 2-CNF LUB of T^+)

Summary

- **Framework: Iterative computation of prime-implicates of bounded size over stratified theories**

- Conditions under which this computation is **tractable**; **Strips** theories as special case

- Correspondence with **planning graph** computation and weak forms of **variable elimination**

- **Positive** vs. **Negative** Deductive Lower bounds

- Uses beyond Strips: conditional effects; **incomplete information**