Causal Belief Decomposition for Planning with Sensing: Completeness Results and Practical Approximation

Blai Bonet[1] and Hector Geffner[2]

[1]Universidad Simón Bolívar
[2]ICREA & Universitat Pompeu Fabra

IJCAI. Beijing, China. August 2013.

Motivation

Planning in the **non-deterministic** and **partially observable** setting

Setting is similar to qualitative POMDPs, where uncertainty is encoded by **sets of states** rather than probability distributions

Two **fundamental tasks** to be solved, both intractable for problems in **compact** form:

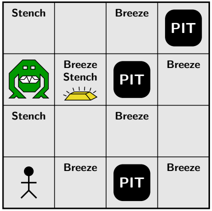
1. Tracking of belief states
2. Action selection for achieving goal

We focus on belief tracking

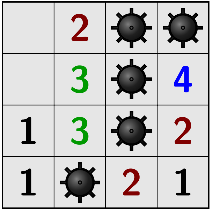
Main Contributions

- We build on a earlier **sound and complete** algorithm for belief tracking for **non-deterministic** partially observable planning that is time and space exponential in a **width** parameter (B&G, 2012)
- Many domains have bounded and small width, but others don't
- We present a more **practical** algorithm, **Beam Tracking**, that is time and space exponential in the much smaller **causal width**
- Beam tracking is powerful but not complete; however, completeness studied over class of **causally decomposable problems**

Example: Wumpus and Minesweeper



Wumpus



Minesweeper

Factored belief tracking (B&G, 2012): exponential in **width** which grows $O(n^2)$ for dimension n

Beam tracking: exponential in **causal width** which is

- Wumpus: constant 4 for any dimension n
- Minesweeper: constant 9 for any dimension n

Outline for the Rest of the Talk

- Model and Language for Planning with Sensing
- Belief Tracking in Planning
- Basic Algorithm: Flat Belief Tracking
- Key Idea in B&G (2012)
- New Idea: Explicit Decompositions
- Causal Belief Tracking and Beam Tracking
- Experiments
- Conclusions

Model for Non-Deterministic Contingent Planning

Contingent model $\mathcal{S} = \langle S, S_0, S_G, A, F, O \rangle$ given by

- finite **state space** S
- non-empty subset of **initial states** $S_0 \subseteq S$
- non-empty subset of **goal states** $S_G \subseteq S$
- **actions** A where $A(s) \subseteq A$ are the actions applicable at state s
- **non-deterministic** transitions $F(s, a) \subseteq S$ for $s \in S, a \in A(s)$
- **non-determinisitc** sensor model $O(s', a) \subseteq O$ for $s' \in S, a \in A$

Language

Model expressed in **compact form** as tuple $P = \langle V, A, I, G, V', W \rangle$:

- V is set of **multi-valued variables**, each X has finite domain D_X
- A is set of actions; each action $a \in A$ has precondition $Pre(a)$ and conditional **non-deterministic** effects $C \rightarrow E^1 | \cdots | E^n$
- Sets of V-literals I and G defining the initial and goal states
- V' is set of observable variables (not necessarily disjoint from V). Observations o are **valuations** over V'
- **Sensing model** is formula $W_a(\ell)$ for each $a \in A$ and observable literal ℓ that is true in states that follow a where ℓ may be observed

Note: a literal is an atom of the form '$X = x$' or '$X \neq x$'

Example: Wumpus

$$
\begin{aligned}
\textit{rotate-right}: \quad & \textit{heading} = N \rightarrow \textit{heading} := E \\
& \textit{heading} = E \rightarrow \textit{heading} := S \\
& \ldots \\
\textit{rotate-left}: \quad & \ldots \\
\textit{move-forward}: \quad & \textit{heading} = N \wedge \textit{pos} = (x, y) \rightarrow \textit{pos} := (x, y+1) \\
& \ldots \\
\textit{grab-gold}: \quad & \textit{gold-pos} = (x, y) \wedge \textit{pos} = (x, y) \rightarrow \textit{gold-pos} := \textsf{hand}
\end{aligned}
$$

$$
\begin{aligned}
W_a(\textit{stench}_{x,y} = \textit{true}) &= \textit{wump}_{x-1,y} \vee \textit{wump}_{x,y+1} \vee \textit{wump}_{x,y-1} \vee \textit{wump}_{x+1,y} \\
W_a(\textit{breeze}_{x,y} = \textit{true}) &= \textit{pit}_{x-1,y} \vee \textit{pit}_{x,y+1} \vee \textit{pit}_{x,y-1} \vee \textit{pit}_{x+1,y} \\
W_a(\textit{glitter}_{x,y} = \textit{true}) &= \big[\textit{gold-pos} = (x, y) \wedge \textit{pos} = (x, y)\big] \\
W_a(\textit{dead}_{x,y} = \textit{true}) &= \big[\textit{pos} = (x, y) \wedge (\textit{pit}_{x,y} \vee \textit{wump}_{x,y})\big]
\end{aligned}
$$

Belief Tracking in Planning (BTP)

Definition (BTP)

Given execution $\tau = \langle a_0, o_0, a_1, o_1, \ldots, a_n, o_n \rangle$ **determine** *whether*

- *execution* τ *is possible, and*
- *whether* b_τ*, the belief that results of executing* τ*, achieves the goal*

In planning **only need** beliefs about preconditions and goals

Theorem

BTP is NP-hard and coNP-hard.

Basic Algorithm: Flat Belief Tracking

Definition (Flat Tracking)

Given belief b at time t, and action a (applied) and observation o (obtained), the belief at time $t+1$ is the belief b_a^o given by

$$b_a = \{s' : s' \in F(s,a) \text{ and } s \in b\}$$
$$b_a^o = \{s' : s' \in b_a \text{ and } s' \models W_a(\ell) \text{ for each } \ell \text{ s.t. } o \models \ell\}$$

- Flat belief tracking is sound and complete for **every formula**
- Time complexity is **exponential in** $|V \cap V_U|$ where $V_U = V \setminus V_K$ and V_K are the variables that are **determined** (aka always known)
- However, in planning, we only need to be complete for literals '$X = x$' involving goal or precondition variables X

Key Idea in B&G (2012)

Beliefs b_X about precondition and goal variables X suffice

Beliefs b_X obtained by applying **flat belief tracking** to smaller subproblems P_X

Subproblem P_X only involves state variables that are **relevant** to X

Resulting algorithm, **Factored Belief Tracking**, is sound and complete for planning, and exponential in **width of P**:

> *maximum number of state variables that are all relevant to a given precondition or goal variable X*

New Idea: Explicit Decompositions

A decomposition of problem P is pair $D = \langle T, B \rangle$ where

- T is subset of **target** variables, and
- $B(X)$ for X in T is a subset of state variables

Decomposition $D = \langle T, B \rangle$ decomposes P into subproblems:

- one subproblem P_X for each variable X in T
- subproblem P_X involves only the state variables in $B(X)$

Belief tracking over a decomposition refers to belief tracking over the subproblems defined by the decomposition

Factored and Causal Decompositions

Definition (Factored Decomposition)

$F = \langle T_F, B_F \rangle$ where T_F are state variables appearing in preconditions or goals, and $B_F(X)$ are all variables that are **relevant** *to X*

Belief tracking over the factored decomposition is sound and complete, and exponential in the **width**

Definition (Causal Decomposition)

$C = \langle T_C, B_C \rangle$ where T_C are variables in preconditions or goals, or **observables**, *and $B_C(X)$ are all variables* **causally relevant** *to X*

Belief tracking over the causal decomposition is sound but not complete, and exponential in the **causal width**

Complete Tracking over Causal Decomposition

Belief tracking over causal decomposition is **incomplete** because

- two beliefs b_X and b_Y associated with target variables X and Y may interact and are not independent

Algorithm can be made complete by enforcing **consistency** of beliefs:

$$b_X := \Pi_{B_C(X)} \bowtie \{(b_Y)_a^o : Y \in T_C \text{ and relevant to } X\}$$

Resulting algorithm is:

- **complete** for **causally decomposable problems** (see paper)
- space exponential in **causal width**
- time exponential in **width**

Wumpus, Minesweeper and Battleship are causally decomposable

Effective Tracking over Causal Decomposition: Beam Tracking

Replaces the costly join (exponential in problem width) with **local consistency** (aka relational arc consistency) until **fix point**:

$$b_X := \Pi_{B_C(X)}(\, b_X^{i+1} \bowtie b_Y^{i+1})$$

Beam tracking is time and space exponential in **causal width**

Beam tracking is sound and powerful but not complete

Beam tracking is **practical algorithm**: general and effective

Incompleteness on causally decomposable problems is the result of replacing the global consistency by local consistency

Experiments

Beam tracking tested on Wumpus, Minesweeper and Battleship using simple heuristics for action selection

Belief tracking on these is intractable (Kaye, 2000; Scott et al., 2011)

Size of tested instances is well beyond scope of contingent planners

Compared with hand-tuned UCT solvers for two of the domains:

- Battleship (Silver and Veness, 2010)
- Minesweeper (Lin et al., 2012)

Obtained similar or superior quality in orders-of-magnitude less time

Experiments: Battleship

				avg. time per	
dim	policy	#ships	#torpedos	decision	game
10×10	greedy	4	40.0 ± 6.9	2.4E-4	9.6E-3
20×20	greedy	8	163.1 ± 32.1	6.6E-4	1.0E-1
30×30	greedy	12	389.4 ± 73.4	1.2E-3	4.9E-1
40×40	greedy	16	723.8 ± 129.2	2.1E-3	1.5

Data for 10,000 runs

On 10×10, achieved same quality as Silver and Veness (2010)
but their UCT takes 3 orders of magnitude more time per move

Experiments: Minesweeper

					avg. time per	
dim	#mines	density	%win	#guess	decision	game
8×8	10	15.6%	83.4	606	8.3E-3	0.21
16×16	40	15.6%	79.8	670	1.2E-2	1.42
16×30	99	20.6%	35.9	2,476	1.1E-2	2.86
32×64	320	15.6%	80.3	672	1.3E-2	2.89

Data for 1,000 runs

Success rates of Lin et al. (2012):

- 8×8: $80.2 \pm 0.4\%$ vs. 83.4%
- 16×16: $74.4 \pm 0.5\%$ vs. 79.8%
- 16×30: $38.7 \pm 1.8\%$ vs. 35.9

No times reported in Lin et al. (2012)

Conclusions

- Planning with sensing is belief tracking and action selection
- Developed a new **effective** and **practical** algorithm for belief tracking, called **beam tracking**
- Beam tracking is time and space exponential in the **causal width** which is often much smaller than the **width** of the problem
- Beam tracking is sound but not complete, yet over the large class of **causally decomposable problems** the incompleteness is the result of replacing the global consistency operation by local approximation
- Challenge: **probabilistic belief tracking**

Thanks. Questions?