Principles of AI Planning

16. Complexity of nondeterministic planning with full observability

Albert-Ludwigs-Universität Freiburg

Bernhard Nebel and Robert Mattmüller

January 16, 2019

Motivation

Overview

- Similar to the earlier analysis of deterministic planning, we will now study the computational complexity of nondeterministic planning with full observability.
- We consider the case of strong planning.
- The results for strong cyclic planning are identical.

As usual, the main motivation for such a study is to determine the limit of what is possible algorithmically: Should we try to develop a polynomial algorithm?

Comparison to deterministic planning

- The basic proof idea is very similar to the PSPACE-completeness proof for deterministic planning.
- The main difference is that we consider alternating Turing Machines (ATMs) instead of deterministic Turing Machines (DTMs) in the reduction.
- Due to the similarity to the earlier proof, we first review some of the concepts introduced in the earlier lecture.

Review

Alternating Turing Machines

Motivation
Review
ATMs
Complexity classes
Complexity results
Summary

Definition: Alternating Turing Machine

Alternating Turing Machine (ATM) $\langle \Sigma, \Box, Q, q_0, l, \delta \rangle$:

1. input alphabet Σ and blank symbol $\Box \notin \Sigma$
   - alphabets always non-empty and finite
   - tape alphabet $\Sigma_\Box = \Sigma \cup \{\Box\}$
2. finite set Q of internal states with initial state $q_0 \in Q$
3. state labeling $l : Q \to \{\mathrm{Y}, \mathrm{N}, \exists, \forall\}$
   - accepting, rejecting, existential, universal states Q_Y, Q_N, $Q_\exists$, $Q_\forall$
   - terminal states $Q_\star = Q_\mathrm{Y} \cup Q_\mathrm{N}$
   - nonterminal states $Q' = Q_\exists \cup Q_\forall$
4. transition relation $\delta \subseteq (Q' \times \Sigma_\Box) \times (Q \times \Sigma_\Box \times \{-1, +1\})$

Turing Machine configurations

Motivation
Review
ATMs
Complexity classes
Complexity results
Summary

Let $M = \langle \Sigma, \Box, Q, q_0, l, \delta \rangle$ be an ATM.

Definition: Configuration

A configuration of M is a triple $(w, q, x) \in \Sigma_\Box^* \times Q \times \Sigma_\Box^+$.

- w: tape contents before tape head
- q: current state
- x: tape contents after and including tape head

Turing Machine transitions

Motivation
Review
ATMs
Complexity classes
Complexity results
Summary

Let $M = \langle \Sigma, \Box, Q, q_0, l, \delta \rangle$ be an ATM.

Definition: Yields relation

A configuration c of M yields a configuration c' of M, in symbols $c \vdash c'$, as defined by the following rules, where $a, a', b \in \Sigma_\Box$, $w, x \in \Sigma_\Box^*$, $q, q' \in Q$ and $((q, a), (q', a', \Delta)) \in \delta$:

$$
\begin{aligned}
(w, q, ax) &\vdash (wa', q', x) && \text{if } \Delta = +1, |x| \geq 1 \\
(w, q, a) &\vdash (wa', q', \Box) && \text{if } \Delta = +1 \\
(wb, q, ax) &\vdash (w, q', ba'x) && \text{if } \Delta = -1 \\
(\varepsilon, q, ax) &\vdash (\varepsilon, q', \Box a'x) && \text{if } \Delta = -1
\end{aligned}
$$

Acceptance (space)

Let $M = \langle \Sigma, \Box, Q, q_0, l, \delta \rangle$ be an ATM.

Definition: Acceptance (space)

Let $c = (w, q, x)$ be a configuration of M.

- M accepts $c = (w, q, x)$ with $q \in Q_Y$ in space n iff $|w| + |x| \leq n$.
- M accepts $c = (w, q, x)$ with $q \in Q_\exists$ in space n iff M accepts some c' with $c \vdash c'$ in space n.
- M accepts $c = (w, q, x)$ with $q \in Q_\forall$ in space n iff M accepts all c' with $c \vdash c'$ in space n.

Accepting words and languages

Let $M = \langle \Sigma, \Box, Q, q_0, l, \delta \rangle$ be an ATM.

Definition: Accepting words

M accepts the word $w \in \Sigma^*$ in space $n \in \mathbb{N}_0$ iff M accepts (ε, q_0, w) in space n.

- Special case: M accepts ε in time (space) $n \in \mathbb{N}_0$ iff M accepts $(\varepsilon, q_0, \Box)$ in time (space) n.

Definition: Accepting languages

Let $f : \mathbb{N}_0 \to \mathbb{N}_0$.
M accepts the language $L \subseteq \Sigma^*$ in space f iff M accepts each word $w \in L$ in space $f(|w|)$, and M does not accept any word $w \notin L$.

Alternating space complexity

Definition: ASPACE, APSPACE

Let $f : \mathbb{N}_0 \to \mathbb{N}_0$.
Complexity class ASPACE(f) contains all languages accepted in space f by some ATM.

Let $\mathscr{P}$ be the set of polynomials $p : \mathbb{N}_0 \to \mathbb{N}_0$.

$$\text{APSPACE} := \bigcup_{p \in \mathscr{P}} \text{ASPACE}(p)$$

Standard complexity classes relationships

Theorem

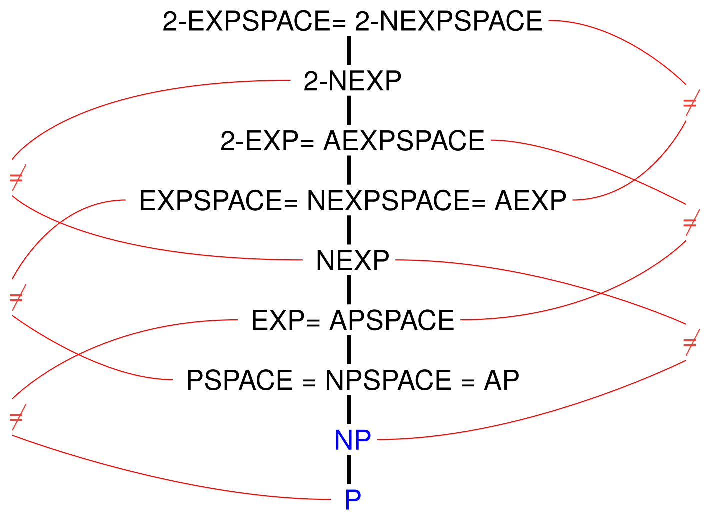
$$\begin{array}{rcl} \text{P} \subseteq & \text{NP} & \subseteq \text{AP} \\ \text{PSPACE} \subseteq & \text{NPSPACE} & \subseteq \text{APSPACE} \\ \text{EXP} \subseteq & \text{NEXP} & \subseteq \text{AEXP} \\ \text{EXPSPACE} \subseteq & \text{NEXPSPACE} & \subseteq \text{AEXPSPACE} \\ \text{2-EXP} \subseteq & \dots & \end{array}$$

The power of alternation

Theorem (Chandra et al. 1981)

$$\begin{aligned}
\text{AP} &= \text{PSPACE} \\
\text{APSPACE} &= \text{EXP} \\
\text{AEXP} &= \text{EXPSPACE} \\
\text{AEXPSPACE} &= \text{2-EXP}
\end{aligned}$$

The hierarchy of complexity classes



Complexity results

The strong planning problem

StrongPlanEx (strong plan existence)

GIVEN: nondeterministic planning task $\langle A, I, O, G, V\rangle$ with full observability ($A = V$)

QUESTION: Is there a strong plan for the task?

- We do not consider a nondeterministic analog of the bounded plan existence problem (PlanLen).

Proof idea

- We will prove that StrongPlanEx is EXP-complete.
- We already know that the problem belongs to EXP, because we have presented a dynamic programming algorithm that generates strong plans in exponential time.
- We prove hardness for EXP by providing a generic reduction for alternating Turing Machines with polynomial space and use Chandra et al.'s theorem showing APSPACE = EXP.

Motivation
Review
Complexity results
The problem
The reduction
The proof
Summary

Reduction
Overview

- For a fixed polynomial p, given ATM M and input w, generate planning task which is solvable by a strong plan iff M accepts w in space $p(|w|)$.
- For simplicity, restrict to ATMs which never move to the left of the initial head position (no loss of generality).
- Existential states of the ATM are modeled by states of the planning task where there are several applicable operators to choose from.
- Universal states of the ATM are modeled by states of the planning task where there is a single applicable operator with a nondeterministic effect.

Motivation
Review
Complexity results
The problem
The reduction
The proof
Summary

Reduction: state variables

Let p be the space-bound polynomial.

Given ATM $\langle \Sigma, \square, Q, q_0, l, \delta \rangle$ and input $w_1 \dots w_n$, define relevant tape positions $X = \{1, \dots, p(n)\}$.

State variables

- state_q for all $q \in Q$
- head_i for all $i \in X \cup \{0, p(n)+1\}$
- $\text{content}_{i,a}$ for all $i \in X$, $a \in \Sigma_\square$

Motivation
Review
Complexity results
The problem
The reduction
The proof
Summary

Reduction: initial state

Let p be the space bound polynomial.

Given ATM $\langle \Sigma, \square, Q, q_0, l, \delta \rangle$ and input $w_1 \dots w_n$, define relevant tape positions $X = \{1, \dots, p(n)\}$.

Initial state formula

Specify a unique initial state.

Initially true:

- state_{q_0}
- head_1
- $\text{content}_{i,w_i}$ for all $i \in \{1, \dots, n\}$
- $\text{content}_{i,\square}$ for all $i \in X \setminus \{1, \dots, n\}$

Initially false:

- all others

Motivation
Review
Complexity results
The problem
The reduction
The proof
Summary

Reduction: goal

Let p be the space bound polynomial.

Given ATM $\langle \Sigma, \Box, Q, q_0, l, \delta \rangle$ and input $w_1 \dots w_n$, define relevant tape positions $X = \{1, \dots, p(n)\}$.

Goal

$\bigvee_{q \in Q_Y} \text{state}_q$

- Without loss of generality, we can assume that Q_Y is a singleton set so that we do not need a disjunctive goal.
- This way, the hardness result also holds for a restricted class of planning tasks ("nondeterministic STRIPS").

Motivation
Review
Complexity results
The problem
The reduction
The proof
Summary

Reduction: operators

Let p be the space bound polynomial.

Given ATM $\langle \Sigma, \Box, Q, q_0, l, \delta \rangle$ and input $w_1 \dots w_n$, define relevant tape positions $X = \{1, \dots, p(n)\}$.

Operators

For $q, q' \in Q$, $a, a' \in \Sigma_\Box$, $\Delta \in \{-1, +1\}$, $i \in X$, define

- $\text{pre}_{q,a,i} = \text{state}_q \wedge \text{head}_i \wedge \text{content}_{i,a}$
- $\text{eff}_{q,a,q',a',\Delta,i} = \neg\text{state}_q \wedge \neg\text{head}_i \wedge \neg\text{content}_{i,a} \wedge \text{state}_{q'} \wedge \text{head}_{i+\Delta} \wedge \text{content}_{i,a'}$
  - If $q = q'$, omit the effects $\neg\text{state}_q$ and $\text{state}_{q'}$.
  - If $a = a'$, omit the effects $\neg\text{content}_{i,a}$ and $\text{content}_{i,a'}$.

Motivation
Review
Complexity results
The problem
The reduction
The proof
Summary

Reduction: operators (continued)

Let p be the space bound polynomial.

Given ATM $\langle \Sigma, \Box, Q, q_0, l, \delta \rangle$ and input $w_1 \dots w_n$, define relevant tape positions $X = \{1, \dots, p(n)\}$.

Operators (ctd.)

For existential states $q \in Q_\exists$, $a \in \Sigma_\Box$, $i \in X$:
Let $(q'_j, a'_j, \Delta_j)_{j \in \{1,\dots,k\}}$ be those triples with $((q,a),(q'_j,a'_j,\Delta_j)) \in \delta$.

For each $j \in \{1, \dots, k\}$, introduce one operator:

- precondition: $\text{pre}_{q,a,i}$
- effect: $\text{eff}_{q,a,q'_j,a'_j,\Delta_j,i}$

Motivation
Review
Complexity results
The problem
The reduction
The proof
Summary

Reduction: operators (continued)

Let p be the space bound polynomial.

Given ATM $\langle \Sigma, \Box, Q, q_0, l, \delta \rangle$ and input $w_1 \dots w_n$, define relevant tape positions $X = \{1, \dots, p(n)\}$.

Operators (ctd.)

For universal states $q \in Q_\forall$, $a \in \Sigma_\Box$, $i \in X$:
Let $(q'_j, a'_j, \Delta_j)_{j \in \{1,\dots,k\}}$ be those triples with $((q,a),(q'_j,a'_j,\Delta_j)) \in \delta$.

Introduce only one operator:

- precondition: $\text{pre}_{q,a,i}$
- effect: $\text{eff}_{q,a,q'_1,a'_1,\Delta_1,i} | \dots | \text{eff}_{q,a,q'_k,a'_k,\Delta_k,i}$

Motivation
Review
Complexity results
The problem
The reduction
The proof
Summary

EXP-completeness of strong planning with full observability

Theorem (Rintanen)

STRONGPLANEX *is* EXP-*complete.*

This is true even if we only allow operators in unary nondeterminism normal form where all deterministic sub-effects and the goal satisfy the STRIPS restriction and if we require a deterministic initial state.

Proof.

Membership in EXP has been shown by providing exponential-time algorithms that generate strong plans (and decide if one exists as a side effect).

Hardness follows from the previous generic reduction for ATMs with polynomial space bound and Chandra et al.'s theorem. □

Summary

Summary

- Nondeterministic planning is harder than deterministic planning.
- In particular, it is EXP-complete in the fully observable case, compared to the PSPACE-completeness of deterministic planning.
- The hardness result already holds if the operators and goals satisfy some fairly strong syntactic restrictions and there is a unique initial state.
- The introduction of nondeterministic effects corresponds to the introduction of alternation in Turing Machines.
- Later, we will see that restricted observability has an even more dramatic effect on the complexity of the planning problem.