

Principles of AI Planning

18. Complexity of nondeterministic planning with full observability

Albert-Ludwigs-Universität Freiburg



Bernhard Nebel and Robert Mattmüller

February 11th, 2015

1 Motivation



Motivation

Review

Complexity
results

Summary

- 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?

Motivation

Review

Complexity
results

Summary

- 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.

2 Review



- Alternating Turing Machines
- Complexity classes

Motivation

Review

ATMs

Complexity classes

Complexity
results

Summary

Definition: Alternating Turing Machine

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

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

Motivation

Review

ATMs

Complexity classes

Complexity
results

Summary

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

Definition: Configuration

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

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

Let $M = \langle \Sigma, \square, Q, q_0, l, \delta \rangle$ 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 $a, a', b \in \Sigma_{\square}$, $w, x \in \Sigma_{\square}^*$, $q, q' \in Q$ and $((q, a), (q', a', \Delta)) \in \delta$:

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

Motivation

Review

ATMs

Complexity classes

Complexity
results

Summary

Let $M = \langle \Sigma, \square, Q, q_0, l, \delta \rangle$ 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 .

Motivation

Review

ATMs

Complexity classes

Complexity
results

Summary

Let $M = \langle \Sigma, \square, Q, q_0, l, \delta \rangle$ 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 $(\varepsilon, q_0, \square)$ in time (space) n .

Definition: Accepting languages

Let $f : \mathbb{N}_0 \rightarrow \mathbb{N}_0$.

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 $w \notin L$.

Motivation

Review

ATMs

Complexity classes

Complexity
results

Summary

Definition: ASPACE, APSPACE

Let $f : \mathbb{N}_0 \rightarrow \mathbb{N}_0$.

Complexity class **ASPACE**(f) contains all languages accepted in space f by some ATM.

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

$$\mathbf{APSPACE} := \bigcup_{p \in \mathcal{P}} \mathbf{ASPACE}(p)$$

Motivation

Review

ATMs

Complexity classes

Complexity results

Summary

Theorem

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

Motivation

Review

ATMs

Complexity classes

Complexity
results

Summary

Motivation

Review

ATMs

Complexity classes

Complexity
results

Summary

Theorem (Chandra et al. 1981)

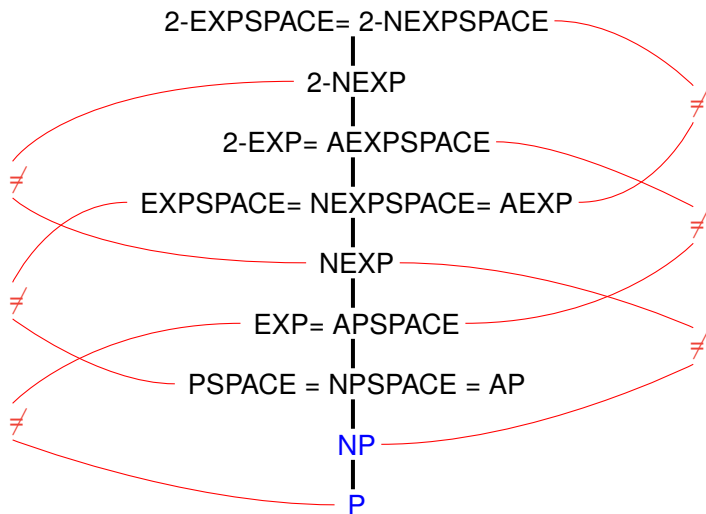
$$AP = PSPACE$$

$$APSPACE = EXP$$

$$AEXP = EXPSPACE$$

$$AEXPSPACE = 2-EXP$$

The hierarchy of complexity classes



Motivation

Review

ATMs

Complexity classes

Complexity results

Summary

3 Complexity results



- The strong planning problem
- APSPACE reduction
- EXP-completeness proof

Motivation

Review

Complexity results

The problem

The reduction

The proof

Summary

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).

- 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



- 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 $i \in X, a \in \Sigma \cup \square$

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 $_1$
- content $_{i,w_i}$ for all $i \in \{1, \dots, n\}$
- 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 $\langle \Sigma, \square, Q, q_0, l, \delta \rangle$ and input $w_1 \dots w_n$,
define **relevant tape positions** $X = \{1, \dots, p(n)\}$.

Goal

$\bigvee_{q \in Q_Y}$ 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

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** $X = \{1, \dots, p(n)\}$.

Operators

For $q, q' \in Q$, $a, a' \in \Sigma \cup \square$, $\Delta \in \{-1, +1\}$, $i \in X$, define

- $\text{pre}_{q,a,i} = \text{state}_q \wedge \text{head}_i \wedge \text{content}_{i,a}$
- $\text{eff}_{q,a,q',a',\Delta,i} = \neg \text{state}_q \wedge \neg \text{head}_i \wedge \neg \text{content}_{i,a}$
 $\quad \wedge \text{state}_{q'} \wedge \text{head}_{i+\Delta} \wedge \text{content}_{i,a'}$
 - 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 $\langle \Sigma, \square, Q, q_0, l, \delta \rangle$ and input $w_1 \dots w_n$,
define **relevant tape positions** $X = \{1, \dots, p(n)\}$.

Operators (ctd.)

For **existential** states $q \in Q_{\exists}$, $a \in \Sigma_{\square}$, $i \in X$:

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

Operators (ctd.)

For **universal** states $q \in Q_{\forall}$, $a \in \Sigma_{\square}$, $i \in X$:

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: $\text{eff}_{q,a,q'_1,a'_1,\Delta_1,i} \mid \dots \mid \text{eff}_{q,a,q'_k,a'_k,\Delta_k,i}$

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.

Beweis.

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. □

Motivation

Review

Complexity
results

The problem

The reduction

The proof

Summary

4 Summary



**UNI
FREIBURG**

Motivation

Review

Complexity
results

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.

Motivation

Review

Complexity
results

Summary