

Principles of AI Planning

4. Normal forms

Bernhard Nebel and Robert Mattmüller

Albert-Ludwigs-Universität Freiburg

November 4th, 2011

Principles of AI Planning

November 4th, 2011 — 4. Normal forms

4.1 Effect normal form

4.2 Positive normal form

4.3 STRIPS operators

Motivation

Motivation

Similarly to normal forms in propositional logic (DNF, CNF, NNF, ...) we can define **normal forms for effects, operators and planning tasks**.

This is useful because algorithms (and proofs) then only need to deal with effects (resp. operators or tasks) in normal form.

Effect normal form

4.2 Effect normal form

- Equivalence of operators and effects
- Definition
- Example

Equivalence of operators and effects

Definition (equivalent effects)

Two effects e and e' over state variables A are **equivalent**, written $e \equiv e'$, if for all states s over A , $[e]_s = [e']_s$.

Definition (equivalent operators)

Two operators o and o' over state variables A are **equivalent**, written $o \equiv o'$, if they are applicable in the same states, and for all states s where they are applicable, $app_o(s) = app_{o'}(s)$.

Theorem

Let $o = \langle \chi, e \rangle$ and $o' = \langle \chi', e' \rangle$ be operators with $\chi \equiv \chi'$ and $e \equiv e'$.
Then $o \equiv o'$.

Note: The converse is not true. (Why not?)

Equivalence transformations for effects

$$e_1 \wedge e_2 \equiv e_2 \wedge e_1 \quad (1)$$

$$(e_1 \wedge e_2) \wedge e_3 \equiv e_1 \wedge (e_2 \wedge e_3) \quad (2)$$

$$\top \wedge e \equiv e \quad (3)$$

$$\chi \triangleright e \equiv \chi' \triangleright e \quad \text{if } \chi \equiv \chi' \quad (4)$$

$$\top \triangleright e \equiv e \quad (5)$$

$$\perp \triangleright e \equiv \top \quad (6)$$

$$\chi_1 \triangleright (\chi_2 \triangleright e) \equiv (\chi_1 \wedge \chi_2) \triangleright e \quad (7)$$

$$\chi \triangleright (e_1 \wedge \dots \wedge e_n) \equiv (\chi \triangleright e_1) \wedge \dots \wedge (\chi \triangleright e_n) \quad (8)$$

$$(\chi_1 \triangleright e) \wedge (\chi_2 \triangleright e) \equiv (\chi_1 \vee \chi_2) \triangleright e \quad (9)$$

Normal form for effects

We can define a **normal form for effects**:

- ▶ Nesting of conditionals, as in $a \triangleright (b \triangleright c)$, can be eliminated.
- ▶ Effects e within a conditional effect $\varphi \triangleright e$ can be restricted to atomic effects (a or $\neg a$).

Transformation to this effect normal form only gives a small polynomial size increase.

Compare: transformation to CNF or DNF may increase formula size exponentially.

Normal form for operators and effects

Definition

An operator $\langle \chi, e \rangle$ is in **effect normal form (ENF)** if for all occurrences of $\chi' \triangleright e'$ in e the effect e' is either a or $\neg a$ for some $a \in A$, and there is at most one occurrence of any atomic effect in e .

Theorem

For every operator there is an equivalent one in effect normal form.

Proof is constructive: we can transform any operator into effect normal form using the equivalence transformations for effects.

Effect normal form example

Example

$$(a \triangleright (b \wedge (c \triangleright (\neg d \wedge e)))) \wedge (\neg b \triangleright e)$$

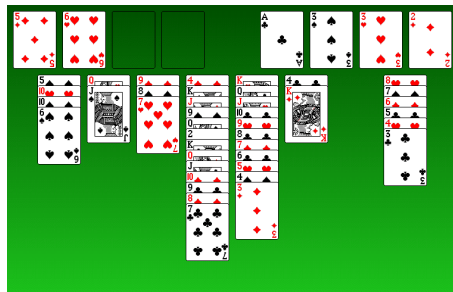
transformed to effect normal form is

$$(a \triangleright b) \wedge ((a \wedge c) \triangleright \neg d) \wedge ((\neg b \vee (a \wedge c)) \triangleright e)$$

4.3 Positive normal form

- Motivation
- Definition & algorithm
- Example
- Advantage

Example: Freecell



Example (good and bad effects)

If we move a card c to a free tableau position, the **good effect** is that the card formerly below c is now available.

The **bad effect** is that we lose one free tableau position.

What is a good or bad effect?

Question: Which operator effects are good, and which are bad?

Difficult to answer in general, because it depends on context:

- ▶ Locking the entrance door is **good** if we want to keep burglars out.
- ▶ Locking the entrance door is **bad** if we want to enter.

We will now consider a reformulation of planning tasks that makes the distinction between good and bad effects obvious.

Positive normal form

Definition (operators in positive normal form)

An operator $o = \langle \chi, e \rangle$ is in **positive normal form** if it is in effect normal form, no negation symbols appear in χ , and no negation symbols appear in any effect condition in e .

Definition (planning tasks in positive normal form)

A planning task $\langle A, I, O, \gamma \rangle$ is in **positive normal form** if all operators in O are in positive normal form and no negation symbols occur in the goal γ .

Positive normal form: existence

Theorem (positive normal form)

Every planning task Π has an equivalent planning task Π' in positive normal form.

Moreover, Π' can be computed from Π in polynomial time.

Note: Equivalence here means that the represented transition systems of Π and Π' , limited to the states that can be reached from the initial state, are isomorphic.

We prove the theorem by describing a suitable algorithm.
(However, we do not prove its correctness or complexity.)

Positive normal form: algorithm

Transformation of $\langle A, I, O, \gamma \rangle$ to positive normal form

Convert all operators $o \in O$ to effect normal form.

Convert all conditions to negation normal form (NNF).

while any condition contains a negative literal $\neg a$:

Let a be a variable which occurs negatively in a condition.

$A := A \cup \{\hat{a}\}$ for some new state variable \hat{a}

$I(\hat{a}) := 1 - I(a)$

Replace the effect a by $(a \wedge \neg \hat{a})$ in all operators $o \in O$.

Replace the effect $\neg a$ by $(\neg a \wedge \hat{a})$ in all operators $o \in O$.

Replace $\neg a$ by \hat{a} in all conditions.

Convert all operators $o \in O$ to effect normal form (again).

Here, *all conditions* refers to all operator preconditions, operator effect conditions and the goal.

Positive normal form: example

Example (transformation to positive normal form)

$$A = \{home, uni, lecture, bike, bike-locked\}$$

$$I = \{home \mapsto 1, bike \mapsto 1, bike-locked \mapsto 1, \\ uni \mapsto 0, lecture \mapsto 0\}$$

$$O = \{\langle home \wedge bike \wedge \neg bike-locked, \neg home \wedge uni \rangle, \\ \langle bike \wedge bike-locked, \neg bike-locked \rangle, \\ \langle bike \wedge \neg bike-locked, bike-locked \rangle, \\ \langle uni, lecture \wedge ((bike \wedge \neg bike-locked) \triangleright \neg bike) \rangle\}$$

$$\gamma = lecture \wedge bike$$

Positive normal form: example

Example (transformation to positive normal form)

$$\begin{aligned}
 A &= \{home, uni, lecture, bike, bike-locked\} \\
 I &= \{home \mapsto 1, bike \mapsto 1, bike-locked \mapsto 1, \\
 &\quad uni \mapsto 0, lecture \mapsto 0\} \\
 O &= \{\langle home \wedge bike \wedge \neg bike-locked, \neg home \wedge uni \rangle, \\
 &\quad \langle bike \wedge bike-locked, \neg bike-locked \rangle, \\
 &\quad \langle bike \wedge \neg bike-locked, bike-locked \rangle, \\
 &\quad \langle uni, lecture \wedge ((bike \wedge \neg bike-locked) \triangleright \neg bike) \rangle\} \\
 \gamma &= lecture \wedge bike
 \end{aligned}$$

Identify state variable a occurring negatively in conditions.

Positive normal form: example

Example (transformation to positive normal form)

$$\begin{aligned}
 A &= \{home, uni, lecture, bike, bike-locked, bike-unlocked\} \\
 I &= \{home \mapsto 1, bike \mapsto 1, bike-locked \mapsto 1, \\
 &\quad uni \mapsto 0, lecture \mapsto 0, bike-unlocked \mapsto 0\} \\
 O &= \{\langle home \wedge bike \wedge \neg bike-locked, \neg home \wedge uni \rangle, \\
 &\quad \langle bike \wedge bike-locked, \neg bike-locked \rangle, \\
 &\quad \langle bike \wedge \neg bike-locked, bike-locked \rangle, \\
 &\quad \langle uni, lecture \wedge ((bike \wedge \neg bike-locked) \triangleright \neg bike) \rangle\} \\
 \gamma &= lecture \wedge bike
 \end{aligned}$$

Introduce new variable \hat{a} with complementary initial value.

Positive normal form: example

Example (transformation to positive normal form)

$$\begin{aligned}
 A &= \{home, uni, lecture, bike, bike-locked, bike-unlocked\} \\
 I &= \{home \mapsto 1, bike \mapsto 1, bike-locked \mapsto 1, \\
 &\quad uni \mapsto 0, lecture \mapsto 0, bike-unlocked \mapsto 0\} \\
 O &= \{\langle home \wedge bike \wedge \neg bike-locked, \neg home \wedge uni \rangle, \\
 &\quad \langle bike \wedge bike-locked, \neg bike-locked \rangle, \\
 &\quad \langle bike \wedge \neg bike-locked, bike-locked \rangle, \\
 &\quad \langle uni, lecture \wedge ((bike \wedge \neg bike-locked) \triangleright \neg bike) \rangle\} \\
 \gamma &= lecture \wedge bike
 \end{aligned}$$

Identify effects on variable a .

Positive normal form: example

Example (transformation to positive normal form)

$$\begin{aligned}
 A &= \{home, uni, lecture, bike, bike-locked, bike-unlocked\} \\
 I &= \{home \mapsto 1, bike \mapsto 1, bike-locked \mapsto 1, \\
 &\quad uni \mapsto 0, lecture \mapsto 0, bike-unlocked \mapsto 0\} \\
 O &= \{\langle home \wedge bike \wedge \neg bike-locked, \neg home \wedge uni \rangle, \\
 &\quad \langle bike \wedge bike-locked, \neg bike-locked \wedge bike-unlocked \rangle, \\
 &\quad \langle bike \wedge \neg bike-locked, bike-locked \wedge \neg bike-unlocked \rangle, \\
 &\quad \langle uni, lecture \wedge ((bike \wedge \neg bike-locked) \triangleright \neg bike) \rangle\} \\
 \gamma &= lecture \wedge bike
 \end{aligned}$$

Introduce complementary effects for \hat{a} .

Positive normal form: example

Example (transformation to positive normal form)

$$A = \{home, uni, lecture, bike, bike-locked, bike-unlocked\}$$

$$I = \{home \mapsto 1, bike \mapsto 1, bike-locked \mapsto 1, \\ uni \mapsto 0, lecture \mapsto 0, bike-unlocked \mapsto 0\}$$

$$O = \{\langle home \wedge bike \wedge \neg bike-locked, \neg home \wedge uni \rangle, \\ \langle bike \wedge bike-locked, \neg bike-locked \wedge bike-unlocked \rangle, \\ \langle bike \wedge \neg bike-locked, bike-locked \wedge \neg bike-unlocked \rangle, \\ \langle uni, lecture \wedge ((bike \wedge \neg bike-locked) \triangleright \neg bike) \rangle\}$$

$$\gamma = lecture \wedge bike$$
Identify negative conditions for a .

Positive normal form: example

Example (transformation to positive normal form)

$$A = \{home, uni, lecture, bike, bike-locked, bike-unlocked\}$$

$$I = \{home \mapsto 1, bike \mapsto 1, bike-locked \mapsto 1, \\ uni \mapsto 0, lecture \mapsto 0, bike-unlocked \mapsto 0\}$$

$$O = \{\langle home \wedge bike \wedge \hat{bike-unlocked}, \neg home \wedge uni \rangle, \\ \langle bike \wedge bike-locked, \neg bike-locked \wedge bike-unlocked \rangle, \\ \langle bike \wedge \hat{bike-unlocked}, bike-locked \wedge \neg bike-unlocked \rangle, \\ \langle uni, lecture \wedge ((bike \wedge \hat{bike-unlocked}) \triangleright \neg bike) \rangle\}$$

$$\gamma = lecture \wedge bike$$
Replace by positive condition \hat{a} .

Positive normal form: example

Example (transformation to positive normal form)

$$A = \{home, uni, lecture, bike, bike-locked, bike-unlocked\}$$

$$I = \{home \mapsto 1, bike \mapsto 1, bike-locked \mapsto 1, \\ uni \mapsto 0, lecture \mapsto 0, bike-unlocked \mapsto 0\}$$

$$O = \{\langle home \wedge bike \wedge bike-unlocked, \neg home \wedge uni \rangle, \\ \langle bike \wedge bike-locked, \neg bike-locked \wedge bike-unlocked \rangle, \\ \langle bike \wedge bike-unlocked, bike-locked \wedge \neg bike-unlocked \rangle, \\ \langle uni, lecture \wedge ((bike \wedge bike-unlocked) \triangleright \neg bike) \rangle\}$$

$$\gamma = lecture \wedge bike$$

Why positive normal form is interesting

In positive normal form, good and bad effects are easy to distinguish:

- ▶ Effects that make state variables true are good (add effects).
- ▶ Effects that make state variables false are bad (delete effects).

This is of high relevance for some planning techniques that we will see later in this course.

4.4 STRIPS operators

- Definition
- Properties

STRIPS operators

Definition

An operator $\langle \chi, e \rangle$ is a **STRIPS operator** if

- ▶ χ is a conjunction of atoms, and
- ▶ e is a conjunction of atomic effects.

Hence every STRIPS operator is of the form

$$\langle a_1 \wedge \dots \wedge a_n, l_1 \wedge \dots \wedge l_m \rangle$$

where a_i are atoms and l_j are atomic effects.

Note: Sometimes we allow conjunctions of **literals** as preconditions. We denote this as **STRIPS with negative preconditions**.

Why STRIPS is interesting

- ▶ STRIPS operators are **particularly simple**, yet expressive enough to capture general planning problems.
- ▶ In particular, STRIPS planning is **no easier** than general planning problems.
- ▶ Most algorithms in the planning literature are **only presented for STRIPS operators** (generalization is often, but not always, obvious).

STRIPS

Stanford Research Institute Planning System
(Fikes & Nilsson, 1971)

Transformation to STRIPS

- ▶ Not every operator is equivalent to a STRIPS operator.
- ▶ However, each operator can be transformed into a **set** of STRIPS operators whose “combination” is equivalent to the original operator. (How?)
- ▶ However, this transformation may exponentially increase the number of required operators. There are planning tasks for which such a blow-up is unavoidable.
- ▶ There are polynomial transformations of planning tasks to STRIPS, but these do not preserve the structure of the transition system (e. g., length of shortest plans may change).

Summary

- ▶ **Effect normal form** simplifies the structure of the operator effects: conditional effects contain only atomic effects, and there is at most one occurrence of any atomic effect.
- ▶ **Positive normal form** allows to distinguish good and bad effects.
- ▶ The form of **STRIPS operators** is even more restrictive than effect normal form, forbidding complex preconditions and conditional effects.
- ▶ All three forms are expressive enough to capture general planning problems.
- ▶ Transformation to effect normal form and positive normal form can be done with a small polynomial size increase.
- ▶ Structure preserving transformations of planning tasks to STRIPS can increase the number of operators exponentially.