Irrelevant Actions and Fluents in Plan Generation

Vladimir Lifschitz and Wanwan Ren

Department of Computer Sciences
The University of Texas at Austin
Example: Open the door

A door is initially locked, how to open it?
Available actions:

- unlocking the door
- opening it
- posting a picture on it

- Solution 1: first unlock the door and post a picture on it simultaneously, and then open it.
- Solution 2: first unlock the door, and then open it.
The door domain in C+

Fluent constants: locked, opened, posted
Action constants: unlock, open, post

Causal laws:

**constraint** locked ⊃ ¬opened

unlock causes ¬locked
nonexecutable unlock if ¬locked

open causes opened
nonexecutable open if opened

post causes posted
nonexecutable post if posted

exogenous c for every action constant c
inertial c for every fluent constant c
Two plans for solving the door problem

• Initial conditions: locked, ¬posted
• Goal: opened

Plan A
1. {unlock, post}
2. {open}

Plan B
1. {unlock}
2. {open}
Action *post* is irrelevant

If $e_1, e_2, ..., e_m$ is a solution to the door problem, then so is $e_1 \setminus \{post\}, e_2 \setminus \{post\}, ..., e_m \setminus \{post\}$.

Why? —— *post* is an irrelevant action to the goal *opened*.

Two “isolated” parts of the set of all constants:

\{unlock, open, locked, opened\}

\{post, posted\}
Plan $B$ is a conformant solution to a modified problem

Incomplete initial conditions

Initial conditions: $locked$

Goal: $opened$

Is there a plan for all possible initial states?

$s_{0a}$: \{locked, $\neg$posted\}, $s_{0b}$: \{locked, posted\}

Plan $A$ ($\times$)

1. \{unlock, post\}
2. \{open\}

Plan $B$ ($\checkmark$)

1. \{unlock\}
2. \{open\}

nonexecutable post if posted

Why?——$posted$ is an irrelevant fluent to the goal $opened$. 
Other examples of irrelevant actions

- Exchanging hats by the missionaries and cannibals. (John McCarthy. Elaboration tolerance. 1999)
  The action of exchanging hats is irrelevant to the goal of crossing the river.

- MS Word commands.
  e.g. edit and paste are irrelevant to the goal of saving a file.
Isolated sets

- $D$: action description in C+
- $\sigma^{all}$: the set of all action and fluent constants in $D$

A set $\sigma \subseteq \sigma^{all}$ is isolated if, for every causal law in $D$ that contains a constant from $\sigma$, all the constants occurring in that law belong to $\sigma$ also.

For instance:

- **constraint** locked $\Rightarrow \neg$opened
- unlock **causes** $\neg$locked
- open **causes** opened

-- If an isolated set contains at least one of the constants unlock, open, locked, opened, it must contain all four.
Isolated sets in the door domain

The graph shows that which constants occur in the same causal law.

Isolated sets: 1. $\emptyset$.
   2. $\{unlock, open, locked, opened\}$.
   3. $\{post, posted\}$.
   4. $\{unlock, open, locked, opened, post, posted\}$.
Theorem

Assumption: For every state $s$, there exists a state $s_1$ such that $<s, \emptyset, s_1>$ is a transition.

Let $D$ be an action description, and $\sigma$ a set of constants isolated with respect to $D$. For any history of $D$

$$<s_0, e_0, s_1, \ldots, s_{m-1}, e_{m-1}, s_m>$$

there exist states $s'_1, \ldots, s'_m$ such that

$$<s_0, e_0 \cap \sigma, s'_1, \ldots, s'_{m-1}, e_{m-1} \cap \sigma, s'_m>$$

is a history also, and, for every $c \in \sigma^f \cap \sigma$, $s'_i(c) = s_i(c)$ ($i=1, \ldots, m$).

The theorem implies: If $e_1, e_2, \ldots, e_m$ is a solution to the door problem, then so is $e_1 \setminus \{\text{post}\}, e_2 \setminus \{\text{post}\}, \ldots, e_m \setminus \{\text{post}\}$. 
Future work

• Extend the theorem to actions with attributes.
• Justify the use of isolated sets in conformant planning.
• Teach CCALC to identify isolated sets so that it can generate better plans, and probably more efficiently.