Planning to Avoid Side Effects

Toryn Q. Klassen[1,2,3] Sheila A. McIlraith[1,2,3]
Christian Muise[4] Jarvis Xu[4]

[1]Department of Computer Science, University of Toronto
[2]Vector Institute for Artificial Intelligence
[3]Schwartz Reisman Institute for Technology and Society
[4]School of Computing, Queen's University

AAAI 2022

Motivation

Underspecified objectives may lead to an AI system causing negative **side effects** (Amodei et al., 2016).

Examples:

- A robot directed to go to a location may **break a vase** on the shortest path (Amodei et al., 2016).
- A robot told to **fetch coffee** might think it was ok to **kill everyone in line** at the coffee shop (example from Stuart Russell[1]).

[1] www.cbc.ca/radio/quirks/apr-25-deepwater-horizon-10-years-later-covid-19-and-understanding-immunity-and-more-1.5541299/the-threat-from-ai-is-not-that-it-will-revolt-it-s-that-it-ll-do-exactly-as-it-s-told-1.5541304

Motivation (continued)

There are various works on avoiding or learning to avoid side effects in MDPs:

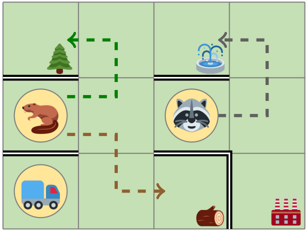
- e.g., Zhang et al. (2018); Krakovna et al. (2019); Turner et al. (2020); Krakovna et al. (2020); Saisubramanian et al. (2020)

Objective underspecification hasn't been much considered in **classical planning**.

- Symbolic planning problems were often **designed by hand** and didn't offer much opportunity for negative side effects.
- **Problem-specific** symbols may not even be able to represent side effects.
- More **realistically complicated** or **learned** models may present risks that can be avoided.

Contributions

- **formalize** the notion of side effect in classical planning
- define classes of **negative side effects** relating to **impact on other agents' ability** to subsequently realize their goals and plans
- propose and implement mechanisms for **computing** side-effect-minimizing plans for STRIPS problems



STRIPS planning problem $\xrightarrow{\textbf{compilation}}$ **Planning problem with costs**

Outline

Background on symbolic planning

Side effects

- Fluent side effects
- Goal side effects
- Policy side effects

Computing side-effect-minimizing plans

Experiments

Conclusion and future work

Background: symbolic planning

- A **state-transition system** is a tuple $\langle S, A, \delta\rangle$ where
  - S is a finite set of states,
  - A is a finite set of actions,
  - and $\delta : S \times A \to S$ is a partial function.
- A **planning problem** consists of
  - a state transition system $\langle S, A, \delta\rangle$,
  - an initial state $s_0 \in S$,
  - and a set of goal states $S_G \subseteq S$.
- A **plan** is an action sequence $\pi = a_1, a_2, \ldots, a_k$ that reaches a goal state.
- For a **multi-agent** setting, we write the set of actions as $A = \bigcup_{i=1}^{n} A_i$, giving each agent i its own action set A_i.

Background: STRIPS

In **STRIPS** planning problems:

- a set of **fluents** are used to represent properties that can change
  - e.g., `at_robot_A` could represent whether a robot is at location A
- a **state** is represented by a set of fluents (the fluents true in that state)
- the **goal** is a set of fluents which have to be made true (while the other fluents can take any value)
  - e.g., {`at_robot_B`}

Outline

An abstract definition of minimizing side effects

Definition (change-minimizing plan)

Given a planning problem and a **distance function** $d : S \times S \to [0, \infty)$, a plan π is change minimizing if it minimizes the distance between the initial and final states.

One simple distance function would count the **number of properties changed**, if states are described in terms of properties (as in STRIPS).

Fluent side effects

Definition (Fluent side effect (FSE))

A fluent f is a side effect of a plan π if f is true after executing π, even though f was neither initially true nor part of the goal. Similarly, $\neg f$ is a side effect if f was initially true.

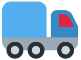
For example, if a fluent `vase_broken` is made true by a plan, then it would be a side effect unless it were part of the goal.

Definition (fluent-preserving)

A plan π for a STRIPS planning problem is **fluent-preserving** if no other plan has strictly fewer fluent side effects.

Negative side effects

- Fluent side effects might be **negative or positive**.
- To consider negative effects, we'll bring **other agents** into the picture.
- We'll define classes of side effects based on impact on **other agents' agency**.

Example: the Canadian wildlife domain

- The robot truck (🚛) wants to get to the factory (🏭), but each cell it touches is contaminated with oil (💧), after which it cannot be visited by animals.
- The beaver (🦫) might want to go to the tree (🌲) or wood (🪵), and the raccoon (🦝) might want to wash its hands in the fountain (⛲).

Example: the Canadian wildlife domain

- The robot truck (🚚) wants to get to the factory (🏭), but each cell it touches is contaminated with oil (🌢), after which it cannot be visited by animals.
- The beaver (🦫) might want to go to the tree (🌲) or wood (🪵), and the raccoon (🦝) might want to wash its hands in the fountain (⛲).

Example: the Canadian wildlife domain

- The robot truck (🚛) wants to get to the factory (🏭), but each cell it touches is contaminated with oil (💧), after which it cannot be visited by animals.
- The beaver (🦫) might want to go to the tree (🌲) or wood (🪵), and the raccoon (🦝) might want to wash its hands in the fountain (⛲).

Example: the Canadian wildlife domain

- The robot truck (🚚) wants to get to the factory (🏭), but each cell it touches is contaminated with oil (💧), after which it cannot be visited by animals.
- The beaver (🦫) might want to go to the tree (🌲) or wood (🪵), and the raccoon (🦝) might want to wash its hands in the fountain (⛲).

Example: the Canadian wildlife domain

- The robot truck (🚛) wants to get to the factory (🏭), but each cell it touches is contaminated with oil (💧), after which it cannot be visited by animals.
- The beaver (🦫) might want to go to the tree (🌲) or wood (🪵), and the raccoon (🦝) might want to wash its hands in the fountain (⛲).

Example: the Canadian wildlife domain

- The robot truck (🚚) wants to get to the factory (🏭), but each cell it touches is contaminated with oil (🌢), after which it cannot be visited by animals.
- The beaver (🦫) might want to go to the tree (🌲) or wood (🪵), and the raccoon (🦝) might want to wash its hands in the fountain (⛲).

Example: the Canadian wildlife domain

- The robot truck (🚛) wants to get to the factory (🏭), but each cell it touches is contaminated with oil (🌢), after which it cannot be visited by animals.
- The beaver (🦫) might want to go to the tree (🌲) or wood (🪵), and the raccoon (🦝) might want to wash its hands in the fountain (⛲).

Example: the Canadian wildlife domain

- The robot truck (🚚) wants to get to the factory (🏭), but each cell it touches is contaminated with oil (💧), after which it cannot be visited by animals.
- The beaver (🦫) might want to go to the tree (🌲) or wood (🪵), and the raccoon (🦝) might want to wash its hands in the fountain (⛲).

Goal side effects are a class of negative side effects

Definition (Goal side effect (GSE))

Given a multi-agent planning environment, suppose that **agent *i*** can achieve a **goal** S'_G from the initial state.
A plan π has a **goal side effect on agent *i*** w.r.t. goal S'_G if ***i* can no longer achieve** S'_G after π is executed.

Goal-preserving plans

Definition (goal-preserving)

Given a planning problem, a set H of goal-agent pairs (such that the given agent initially can achieve the goal), and a weight function $w : H \to \mathbb{R}$, a plan π is **goal-preserving** if it **minimizes the weighted sum of goals** from H that are **made unachievable for their corresponding agents**.

- One of the agents whose goals are being preserved could be the **same agent** who's executing the goal-preserving plan.
- This only considers the very **next goal** to be attempted, and doesn't try to deal with effects other agents might have on **each other**.

Roles of the weights in goal-preserving plans

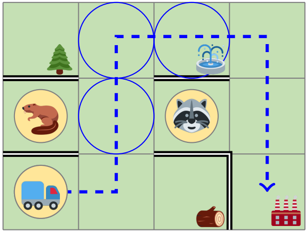
- There may be **uncertainty** about what future goal will be desired.
  - The weight of a goal-agent pair could reflect the **probability** that that agent would pursue that goal next.
- Even if some goals are not expected to be actually chosen, it may be important to preserve the **freedom** of other agents to choose.
  - The weights could reflect the **importance** of keeping particular options available.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver (🦫) reaches the tree (🌲),
- the beaver (🦫) reaches the wood (🪵), or
- the raccoon (🦝) reaches the fountain (⛲).

The robot can clean oil spills in up to three cells.



The robot plans to clean the circled cells.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver (🦫) reaches the tree (🌲),
- the beaver (🦫) reaches the wood (🪵), or
- the raccoon (🦝) reaches the fountain (⛲).

The robot can clean oil spills in up to three cells.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver (🦫) reaches the tree (🌲),
- the beaver (🦫) reaches the wood (🪵), or
- the raccoon (🦝) reaches the fountain (⛲).

The robot can clean oil spills in up to three cells.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver (🦫) reaches the tree (🌲),
- the beaver (🦫) reaches the wood (🪵), or
- the raccoon (🦝) reaches the fountain (⛲).

The robot can clean oil spills in up to three cells.

The robot cleans the cell.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver (🦫) reaches the tree (🌲),
- the beaver (🦫) reaches the wood (🪵), or
- the raccoon (🦝) reaches the fountain (⛲).

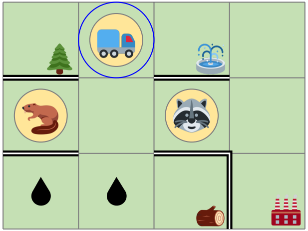
The robot can clean oil spills in up to three cells.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver (🦫) reaches the tree (🌲),
- the beaver (🦫) reaches the wood (🪵), or
- the raccoon (🦝) reaches the fountain (⛲).

The robot can clean oil spills in up to three cells.



The robot cleans the cell.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver (🦫) reaches the tree (🌲),
- the beaver (🦫) reaches the wood (🪵), or
- the raccoon (🦝) reaches the fountain (⛲).

The robot can clean oil spills in up to three cells.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver ( ) reaches the tree ( ),
- the beaver ( ) reaches the wood ( ), or
- the raccoon ( ) reaches the fountain ( ).

The robot can clean oil spills in up to three cells.

The robot cleans the cell.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver (🦫) reaches the tree (🌲),
- the beaver (🦫) reaches the wood (🪵), or
- the raccoon (🦝) reaches the fountain (⛲).

The robot can clean oil spills in up to three cells.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver (🦫) reaches the tree (🌲),
- the beaver (🦫) reaches the wood (🪵), or
- the raccoon (🦝) reaches the fountain (⛲).

The robot can clean oil spills in up to three cells.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver (🦫) reaches the tree (🌲),
- the beaver (🦫) reaches the wood (🪵), or
- the raccoon (🦝) reaches the fountain (⛲).

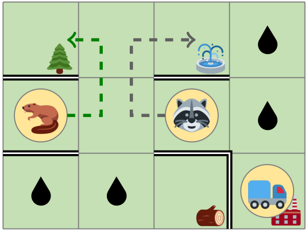
The robot can clean oil spills in up to three cells.

Planning to avoid (goal) side effects

Set of **possible goals**:

- the beaver (🦫) reaches the tree (🌲),
- the beaver (🦫) reaches the wood (🪵), or
- the raccoon (🦝) reaches the fountain (⛲).

The robot can clean oil spills in up to three cells.



The beaver can still reach the tree, and the raccoon the fountain.

Policy side effects are another class of negative side effects

Definition (Policy)

A (partial) policy is a (partial) function from states to actions.

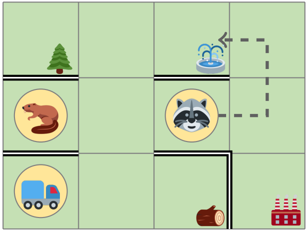
Definition (Policy side effect (PSE))

Given a multi-agent planning environment, suppose that **agent *i*** can achieve a goal S'_G from the initial state **using policy ρ**.
A plan π has a **policy side effect on agent *i*** w.r.t. goal S'_G and policy ρ if ***i* can no longer achieve S'_G using ρ** after π is executed.

Policy side effects are another class of negative side effects

Definition (Policy side effect (PSE))

Given a multi-agent planning environment, suppose that **agent *i*** can achieve a goal S'_G from the initial state **using policy ρ**.
A plan π has a **policy side effect on agent *i*** w.r.t. goal S'_G and policy ρ if ***i*** **can no longer achieve S'_G using ρ** after π is executed.

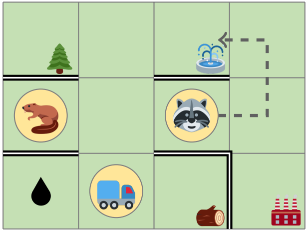


Policy side effects are another class of negative side effects

Definition (Policy side effect (PSE))

Given a multi-agent planning environment, suppose that **agent i** can achieve a goal S'_G from the initial state **using policy ρ**.

A plan π has a **policy side effect on agent i** w.r.t. goal S'_G and policy ρ if **i can no longer achieve S'_G using ρ** after π is executed.

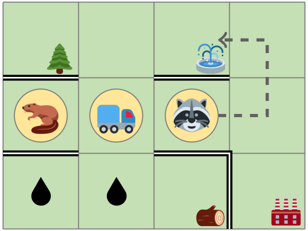


Policy side effects are another class of negative side effects

Definition (Policy side effect (PSE))

Given a multi-agent planning environment, suppose that **agent i** can achieve a goal S'_G from the initial state **using policy ρ**.

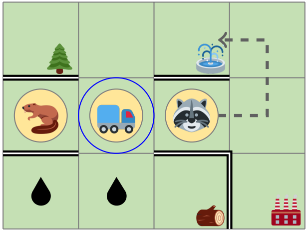
A plan π has a **policy side effect on agent i** w.r.t. goal S'_G and policy ρ if **i can no longer achieve S'_G using ρ** after π is executed.



Policy side effects are another class of negative side effects

Definition (Policy side effect (PSE))

Given a multi-agent planning environment, suppose that **agent i** can achieve a goal S'_G from the initial state **using policy ρ**.
A plan π has a **policy side effect on agent i** w.r.t. goal S'_G and policy ρ if **i can no longer achieve S'_G using ρ** after π is executed.

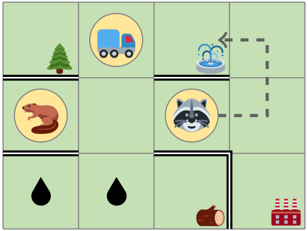


The robot cleans the cell.

Policy side effects are another class of negative side effects

Definition (Policy side effect (PSE))

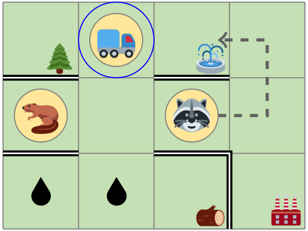
Given a multi-agent planning environment, suppose that **agent i** can achieve a goal S'_G from the initial state **using policy ρ**.
A plan π has a **policy side effect on agent i** w.r.t. goal S'_G and policy ρ if **i can no longer achieve S'_G using ρ** after π is executed.



Policy side effects are another class of negative side effects

Definition (Policy side effect (PSE))

Given a multi-agent planning environment, suppose that **agent *i*** can achieve a goal S'_G from the initial state **using policy ρ**.
A plan π has a **policy side effect on agent *i*** w.r.t. goal S'_G and policy ρ if ***i* can no longer achieve S'_G using ρ** after π is executed.

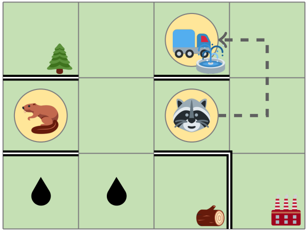


The robot cleans the cell.

Policy side effects are another class of negative side effects

Definition (Policy side effect (PSE))

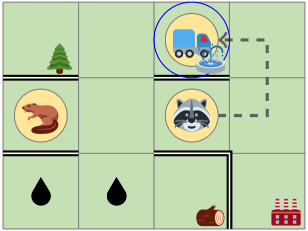
Given a multi-agent planning environment, suppose that **agent i** can achieve a goal S'_G from the initial state **using policy ρ**.
A plan π has a **policy side effect on agent i** w.r.t. goal S'_G and policy ρ if **i can no longer achieve S'_G using ρ** after π is executed.



Policy side effects are another class of negative side effects

Definition (Policy side effect (PSE))

Given a multi-agent planning environment, suppose that **agent *i*** can achieve a goal S'_G from the initial state **using policy ρ**.
A plan π has a **policy side effect on agent *i*** w.r.t. goal S'_G and policy ρ if ***i* can no longer achieve S'_G using ρ** after π is executed.

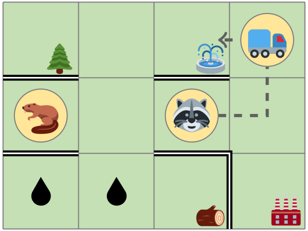


The robot cleans the cell.

Policy side effects are another class of negative side effects

Definition (Policy side effect (PSE))

Given a multi-agent planning environment, suppose that **agent *i*** can achieve a goal S'_G from the initial state **using policy ρ**.
A plan π has a **policy side effect on agent *i*** w.r.t. goal S'_G and policy ρ if ***i* can no longer achieve S'_G using ρ** after π is executed.



Policy side effects are another class of negative side effects

Definition (Policy side effect (PSE))

Given a multi-agent planning environment, suppose that **agent i** can achieve a goal S'_G from the initial state **using policy ρ**.
A plan π has a **policy side effect on agent i** w.r.t. goal S'_G and policy ρ if **i can no longer achieve S'_G using ρ** after π is executed.

Policy side effects are another class of negative side effects

Definition (Policy side effect (PSE))

Given a multi-agent planning environment, suppose that **agent i** can achieve a goal S'_G from the initial state **using policy ρ**.
A plan π has a **policy side effect on agent i** w.r.t. goal S'_G and policy ρ if **i can no longer achieve S'_G using ρ** after π is executed.

Policy-preserving plans

Definition (policy-preserving)

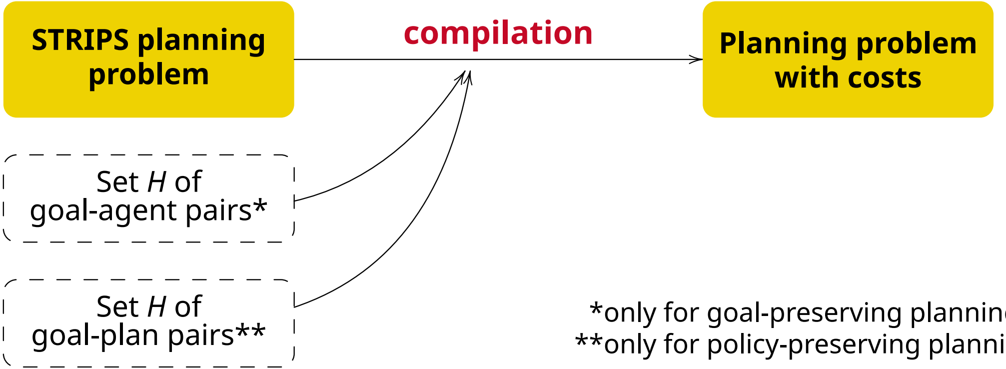
Given a planning problem, a set H of goal-**policy** pairs (such that the given policy initially can achieve the goal), and a weight function $w : H \to \mathbb{R}$, a plan π is **policy-preserving** if it **minimizes the weighted sum of goals** from H that are **made unachievable by their corresponding policies**.

Side-effect-minimizing objectives

- minimizing how many properties are changed (**fluent-preserving** plans)
- minimizing how many possible goals are made unachievable (**goal-preserving** plans)
- minimizing how many policies are made unable to achieve their goals (**policy-preserving** plans)

Outline

Computation of side-effect-minimizing plans



*only for goal-preserving planning
**only for policy-preserving planning

- based on the **soft goals** compilation by Keyder and Geffner (2009)
- see paper for details

Outline

Experimental results

|*H*|: number of goal-policy / goal-agent pairs
PT: planning time (seconds)
CT: compilation time (seconds)
FSE: fluent side effects
PSE: policy side effects
GSE: goal side effects

Domain & Problem	$\lvert H \rvert$	Standard planning				Fluent-preserving					Policy-preserving			Goal-preserving		
		FSE	PSE	GSE	PT	FSE	PSE	GSE	CT	PT	PSE	CT	PT	GSE	CT	PT
wildlife	3, 3	17	3	3	0.5	13	3	3	0.8	20.2	1	0.6	6.5	1	0.6	38.0
zeno-a	5, 2	7	4	0	0.5	5	4	0	17.6	10.6	3	17.6	9.5	0	17.3	23.3
zeno-b	4, 2	5	2	0	0.4	5	2	0	17.6	7.2	0	17.4	10.4	0	17.0	24.6
zeno-c	7, 4	5	3	0	0.4	3	3	0	18.2	12.3	3	17.9	7.9	0	17.2	26.3
floortile-a	4, 2	6	4	0	0.5	2	3	1	2.8	16.9	0	2.5	9.2	0	2.5	56.4
floortile-b	4, 2	5	4	0	0.4	1	3	0	2.8	11.6	0	2.4	7.3	0	2.5	54.6
floortile-c	8, 4	5	8	1	0.5	1	5	0	2.8	18.5	1	2.5	4.9	0	2.5	97.2
storage-a	6, 2	5	5	0	0.4	5	5	0	0.9	7.4	0	0.9	10.4	0	0.9	14.1
storage-b	4, 2	8	4	0	0.4	5	2	0	0.9	6.2	0	0.9	5.2	0	0.9	15.5
storage-c	7, 4	14	3	2	0.4	10	3	0	0.9	7.0	3	0.9	5.7	0	0.9	16.2
storage-c2	7, 4	14	3	2	0.4	10	3	0	10.2	44.0	3	10.0	48.8	0	10.1	21.0
storage-c3	7, 4	14	3	2	0.4	10	3	0	49.8	163.5	3	50.3	159.3	0	48.5	53.7

Experimental results

|*H*|: number of goal-policy / goal-agent pairs
PT: planning time (seconds)
CT: compilation time (seconds)
FSE: fluent side effects
PSE: policy side effects
GSE: goal side effects

Domain & Problem	$\lvert H \rvert$	Standard planning				Fluent-preserving					Policy-preserving			Goal-preserving		
		FSE	PSE	GSE	PT	FSE	PSE	GSE	CT	PT	PSE	CT	PT	GSE	CT	PT
wildlife	3, 3	17	3	3	0.5	13	3	3	0.8	20.2	1	0.6	6.5	1	0.6	38.0
zeno-a	5, 2	7	4	0	0.5	5	4	0	17.6	10.6	3	17.6	9.5	0	17.3	23.3
zeno-b	4, 2	5	2	0	0.4	5	2	0	17.6	7.2	0	17.4	10.4	0	17.0	24.6
zeno-c	7, 4	5	3	0	0.4	3	3	0	18.2	12.3	3	17.9	7.9	0	17.2	26.3
floortile-a	4, 2	6	4	0	0.5	2	3	1	2.8	16.9	0	2.5	9.2	0	2.5	56.4
floortile-b	4, 2	5	4	0	0.4	1	3	0	2.8	11.6	0	2.4	7.3	0	2.5	54.6
floortile-c	8, 4	5	8	1	0.5	1	5	0	2.8	18.5	1	2.5	4.9	0	2.5	97.2
storage-a	6, 2	5	5	0	0.4	5	5	0	0.9	7.4	0	0.9	10.4	0	0.9	14.1
storage-b	4, 2	8	4	0	0.4	5	2	0	0.9	6.2	0	0.9	5.2	0	0.9	15.5
storage-c	7, 4	14	3	2	0.4	10	3	0	0.9	7.0	3	0.9	5.7	0	0.9	16.2
storage-c2	7, 4	14	3	2	0.4	10	3	0	10.2	44.0	3	10.0	48.8	0	10.1	21.0
storage-c3	7, 4	14	3	2	0.4	10	3	0	49.8	163.5	3	50.3	159.3	0	48.5	53.7

Experimental results

$|H|$: number of goal-policy / goal-agent pairs
PT: planning time (seconds)
CT: compilation time (seconds)
FSE: fluent side effects
PSE: policy side effects
GSE: goal side effects

Domain & Problem	$\|H\|$	Standard planning				Fluent-preserving					Policy-preserving			Goal-preserving		
		FSE	PSE	GSE	PT	FSE	PSE	GSE	CT	PT	PSE	CT	PT	GSE	CT	PT
wildlife	3, 3	17	3	3	0.5	13	3	3	0.8	20.2	1	0.6	6.5	1	0.6	38.0
zeno-a	5, 2	7	4	0	0.5	5	4	0	17.6	10.6	3	17.6	9.5	0	17.3	23.3
zeno-b	4, 2	5	2	0	0.4	5	2	0	17.6	7.2	0	17.4	10.4	0	17.0	24.6
zeno-c	7, 4	5	3	0	0.4	3	3	0	18.2	12.3	3	17.9	7.9	0	17.2	26.3
floortile-a	4, 2	6	4	0	0.5	2	3	1	2.8	16.9	0	2.5	9.2	0	2.5	56.4
floortile-b	4, 2	5	4	0	0.4	1	3	0	2.8	11.6	0	2.4	7.3	0	2.5	54.6
floortile-c	8, 4	5	8	1	0.5	1	5	0	2.8	18.5	1	2.5	4.9	0	2.5	97.2
storage-a	6, 2	5	5	0	0.4	5	5	0	0.9	7.4	0	0.9	10.4	0	0.9	14.1
storage-b	4, 2	8	4	0	0.4	5	2	0	0.9	6.2	0	0.9	5.2	0	0.9	15.5
storage-c	7, 4	14	3	2	0.4	10	3	0	0.9	7.0	3	0.9	5.7	0	0.9	16.2
storage-c2	7, 4	14	3	2	0.4	10	3	0	10.2	44.0	3	10.0	48.8	0	10.1	21.0
storage-c3	7, 4	14	3	2	0.4	10	3	0	49.8	163.5	3	50.3	159.3	0	48.5	53.7

Experimental results

$|H|$: number of goal-policy / goal-agent pairs
PT: planning time (seconds)
CT: compilation time (seconds)
FSE: fluent side effects
PSE: policy side effects
GSE: goal side effects

Domain & Problem	$\lvert H \rvert$	Standard planning				Fluent-preserving					Policy-preserving			Goal-preserving		
		FSE	PSE	GSE	PT	FSE	PSE	GSE	CT	PT	PSE	CT	PT	GSE	CT	PT
wildlife	3, 3	17	3	3	0.5	13	3	3	0.8	20.2	1	0.6	6.5	1	0.6	38.0
zeno-a	5, 2	7	4	0	0.5	5	4	0	17.6	10.6	3	17.6	9.5	0	17.3	23.3
zeno-b	4, 2	5	2	0	0.4	5	2	0	17.6	7.2	0	17.4	10.4	0	17.0	24.6
zeno-c	7, 4	5	3	0	0.4	3	3	0	18.2	12.3	3	17.9	7.9	0	17.2	26.3
floortile-a	4, 2	6	4	0	0.5	2	3	1	2.8	16.9	0	2.5	9.2	0	2.5	56.4
floortile-b	4, 2	5	4	0	0.4	1	3	0	2.8	11.6	0	2.4	7.3	0	2.5	54.6
floortile-c	8, 4	5	8	1	0.5	1	5	0	2.8	18.5	1	2.5	4.9	0	2.5	97.2
storage-a	6, 2	5	5	0	0.4	5	5	0	0.9	7.4	0	0.9	10.4	0	0.9	14.1
storage-b	4, 2	8	4	0	0.4	5	2	0	0.9	6.2	0	0.9	5.2	0	0.9	15.5
storage-c	7, 4	14	3	2	0.4	10	3	0	0.9	7.0	3	0.9	5.7	0	0.9	16.2
storage-c2	7, 4	14	3	2	0.4	10	3	0	10.2	44.0	3	10.0	48.8	0	10.1	21.0
storage-c3	7, 4	14	3	2	0.4	10	3	0	49.8	163.5	3	50.3	159.3	0	48.5	53.7

Experimental results

|*H*|: number of goal-policy / goal-agent pairs
PT: planning time (seconds)
CT: compilation time (seconds)
FSE: fluent side effects
PSE: policy side effects
GSE: goal side effects

Domain & Problem	|*H*|	Standard planning				Fluent-preserving					Policy-preserving			Goal-preserving		
		FSE	PSE	GSE	PT	FSE	PSE	GSE	CT	PT	PSE	CT	PT	GSE	CT	PT
wildlife	3, 3	17	3	3	0.5	13	3	3	0.8	20.2	1	0.6	6.5	1	0.6	38.0
zeno-a	5, 2	7	4	0	0.5	5	4	0	17.6	10.6	3	17.6	9.5	0	17.3	23.3
zeno-b	4, 2	5	2	0	0.4	5	2	0	17.6	7.2	0	17.4	10.4	0	17.0	24.6
zeno-c	7, 4	5	3	0	0.4	3	3	0	18.2	12.3	3	17.9	7.9	0	17.2	26.3
floortile-a	4, 2	6	4	0	0.5	2	3	1	2.8	16.9	0	2.5	9.2	0	2.5	56.4
floortile-b	4, 2	5	4	0	0.4	1	3	0	2.8	11.6	0	2.4	7.3	0	2.5	54.6
floortile-c	8, 4	5	8	1	0.5	1	5	0	2.8	18.5	1	2.5	4.9	0	2.5	97.2
storage-a	6, 2	5	5	0	0.4	5	5	0	0.9	7.4	0	0.9	10.4	0	0.9	14.1
storage-b	4, 2	8	4	0	0.4	5	2	0	0.9	6.2	0	0.9	5.2	0	0.9	15.5
storage-c	7, 4	14	3	2	0.4	10	3	0	0.9	7.0	3	0.9	5.7	0	0.9	16.2
storage-c2	7, 4	14	3	2	0.4	10	3	0	10.2	44.0	3	10.0	48.8	0	10.1	21.0
storage-c3	7, 4	14	3	2	0.4	10	3	0	49.8	163.5	3	50.3	159.3	0	48.5	53.7

Experimental results

|*H*|: number of goal-policy / goal-agent pairs
PT: planning time (seconds)
CT: compilation time (seconds)
FSE: fluent side effects
PSE: policy side effects
GSE: goal side effects

Domain & Problem	|*H*|	Standard planning				Fluent-preserving					Policy-preserving			Goal-preserving		
		FSE	PSE	GSE	PT	FSE	PSE	GSE	CT	PT	PSE	CT	PT	GSE	CT	PT
wildlife	3, 3	17	3	3	0.5	13	3	3	0.8	20.2	1	0.6	6.5	1	0.6	38.0
zeno-a	5, 2	7	4	0	0.5	5	4	0	17.6	10.6	3	17.6	9.5	0	17.3	23.3
zeno-b	4, 2	5	2	0	0.4	5	2	0	17.6	7.2	0	17.4	10.4	0	17.0	24.6
zeno-c	7, 4	5	3	0	0.4	3	3	0	18.2	12.3	3	17.9	7.9	0	17.2	26.3
floortile-a	4, 2	6	4	0	0.5	2	3	1	2.8	16.9	0	2.5	9.2	0	2.5	56.4
floortile-b	4, 2	5	4	0	0.4	1	3	0	2.8	11.6	0	2.4	7.3	0	2.5	54.6
floortile-c	8, 4	5	8	1	0.5	1	5	0	2.8	18.5	1	2.5	4.9	0	2.5	97.2
storage-a	6, 2	5	5	0	0.4	5	5	0	0.9	7.4	0	0.9	10.4	0	0.9	14.1
storage-b	4, 2	8	4	0	0.4	5	2	0	0.9	6.2	0	0.9	5.2	0	0.9	15.5
storage-c	7, 4	14	3	2	0.4	10	3	0	0.9	7.0	3	0.9	5.7	0	0.9	16.2
storage-c2	7, 4	14	3	2	0.4	10	3	0	10.2	44.0	3	10.0	48.8	0	10.1	21.0
storage-c3	7, 4	14	3	2	0.4	10	3	0	49.8	163.5	3	50.3	159.3	0	48.5	53.7

Outline

Conclusion and future work

- Planning with complicated or learned models could lead to **side effects**.
- We've considered side effects on the **goals and plans of other agents.**

Future work:

- other types of negative side effects:
  - increasing the **cost** other agents incur in reaching their goals
  - side effects which occur **before** the end of the plan
- trade-off between cost of plan and side effects caused
- more efficient ways of minimizing side effects

References

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul F. Christiano, John Schulman, and Dan Mané. Concrete problems in AI safety. *arXiv preprint arXiv:1606.06565*, 2016. URL `http://arxiv.org/abs/1606.06565`.

Emil Keyder and Hector Geffner. Soft goals can be compiled away. *Journal of Artificial Intelligence Research*, 36:547–556, 2009. doi: 10.1613/jair.2857.

Victoria Krakovna, Laurent Orseau, Miljan Martic, and Shane Legg. Penalizing side effects using stepwise relative reachability. In *Proceedings of the Workshop on Artificial Intelligence Safety 2019 co-located with the 28th International Joint Conference on Artificial Intelligence, AISafety@IJCAI 2019*, volume 2419 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2019. URL `http://ceur-ws.org/Vol-2419/paper_1.pdf`.

Victoria Krakovna, Laurent Orseau, Richard Ngo, Miljan Martic, and Shane Legg. Avoiding side effects by considering future tasks. In *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, 2020. URL `https://papers.nips.cc/paper/2020/file/dc1913d422398c25c5f0b81cab94cc87-Paper.pdf`.

Sandhya Saisubramanian, Ece Kamar, and Shlomo Zilberstein. A multi-objective approach to mitigate negative side effects. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 354–361, 2020. doi: 10.24963/ijcai.2020/50.

Alexander Matt Turner, Dylan Hadfield-Menell, and Prasad Tadepalli. Conservative agency via attainable utility preservation. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, AIES '20, pages 385–391, 2020. doi: 10.1145/3375627.3375851.

Shun Zhang, Edmund H. Durfee, and Satinder P. Singh. Minimax-regret querying on side effects for safe optimality in factored Markov decision processes. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018*, pages 4867–4873, 2018. doi: 10.24963/ijcai.2018/676.