

Generation of Strong Cyclic Plans with Incomplete Information and Sensing

joint work with: Luca Iocchi, Riccardo Rosati

⇒ **Incomplete information** about the world

⇒ **Partial observability** of the world

Incomplete information

The exact state of the world is not completely known by the agent

⇒ **incomplete initial state**

Moreover, the agent acts in an unpredictable environment, where other (unknown) agents may unpredictably change the state of the world

⇒ **presence of exogenous actions/effects** (i.e., the state of the world is non-inertial)

Partial observability

- The agent is able to execute **sensing actions** to acquire knowledge about the environment
- However, only some properties of the world can be sensed by the agent
- Moreover, the executability of such sensing actions is in general context-dependent

⇒ **partial observability**

Strong Cyclic Conditional planning

Conditional planning = *nondeterministic control problem over belief space*.

Strong Cyclic Conditional (or contingent) planning: plan with

- ordinary actions
- sensing actions
- conditional (if–then–else) statement (sensing)
- **loop (while) statement (sensing)**

Our proposal

1. definition of \mathcal{KL} , a language for representing planning domains with incomplete information and partial observability
2. semantics of \mathcal{KL} expressed through sets of **belief states**
3. definition of the notion of **strong cyclic conditional plan** in the \mathcal{KL} framework
4. algorithm for computing strong cyclic conditional plans in \mathcal{KL}
5. implementation and experimentation of the algorithm

The language \mathcal{KL} – Syntax

- initial state = formula $\mathbf{K}\phi_{init}$
- goal = formula $\mathbf{K}\phi_{goal}$
- precondition for action $a = pre_a : \mathbf{K}\alpha$
- effect of ordinary action $a = post_a : \bigwedge_i (\mathbf{K}\alpha_i \rightarrow \mathbf{K}\beta_i)$
- effect of sensing action $a = sense_a : P$
- forgetting effect of action $a = post_a : \neg\mathbf{K}P$
- static axiom = formula α

The language \mathcal{KL} –Semantics

- **belief state** = set of propositional interpretations
- a belief state b is denoted by an epistemic formula $\mathbf{K}\phi_b$ such that b is the set of interpretations satisfying ϕ_b
- initial belief state = $\mathbf{K}\phi_{init}$
- the **transition function** $f_a(b) = b_a^o$ is defined through the combination of: deterministic effects, sensing effects, forgetting effects and static axioms

\mathcal{KL} specification: Cleaning example

Cleaning Robot searching and cleaning objects out of a set of rooms.

Action a	pre_a	$post_a$	$sense_a$
enter_Rj	$\mathbf{K}in(C)$	$\mathbf{K}in(R_j)$	
exit_Rj	$\mathbf{K}in(R_j)$	$\mathbf{K}in(C)$	
takeout_Rj	$\mathbf{K}in(R_j) \wedge \mathbf{K}obj_in(R_j)$	$\mathbf{K}in(C) \wedge \neg \mathbf{K}obj_in(R_j)$	
scan_Rj	$\mathbf{K}in(R_j)$		$obj_in(R_j)$

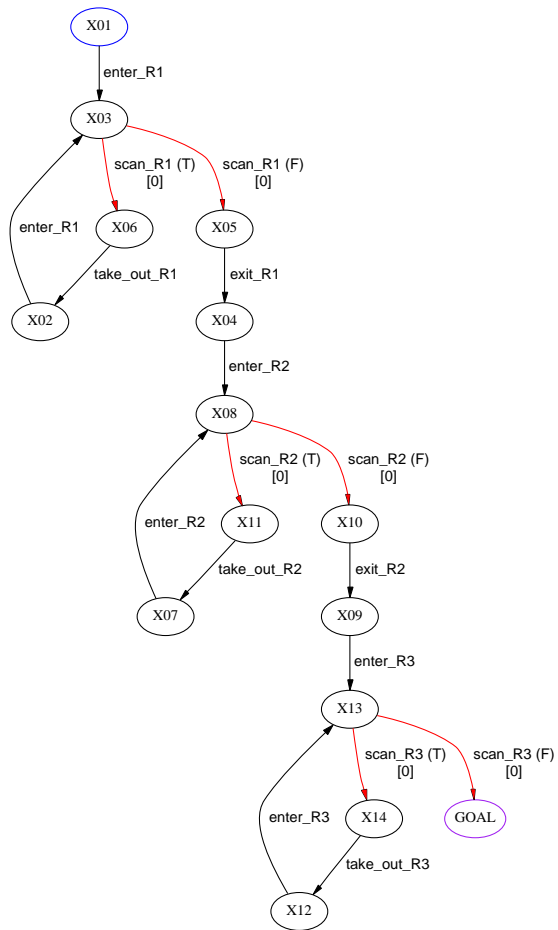
Without knowing how many objects are in every room
a conditional plan (without loops) solution does not exist.

Strong cyclic planning in \mathcal{KL}

- searching in the belief states space
- using heuristics to find linear paths (i.e. partial plans considering only one effect of sensing)
- merging partial plans to (possibly cyclic) conditional plans.

Epistemic reasoning framework that provides the semantics for \mathcal{KL} [Iocchi, Nardi, Rosati (KR-2000)] allows for **verifying equivalent belief states** and guarantees **soundness** and **completeness**.

Cyclic conditional plan for the Cleaning domain



Experimental results for conditional planning

Experimental results show that explicit representation of the agent knowledge (\mathcal{K} -Planner and PKS) significantly improve performance of reasoning, with respect to representations based on possible worlds.

Rooms	First Linear Path	Complete Linear Path
10	60	3
20	240	8
30	1040	35
40	3230	60
50	7900	85

Cleaning domain

Illnesses	\mathcal{K} -Planner	PKS [PeBa02]	MBP [BeCi01]
4	1	-	10
6	2	-	300
8	4	-	11500
10	8	-	?
20	65	80	?
50	620	1610	?
100	4620	20390	?

Medical domain

Conclusions

- definition of a language \mathcal{KL} for representing planning domains with incomplete information and partial observability
- algorithm for computing **strong cyclic conditional plans** in \mathcal{KL}
- cyclic plan generation in domains where a conditional plan solution does not exist
- efficient implementation using explicit knowledge representation and heuristic search