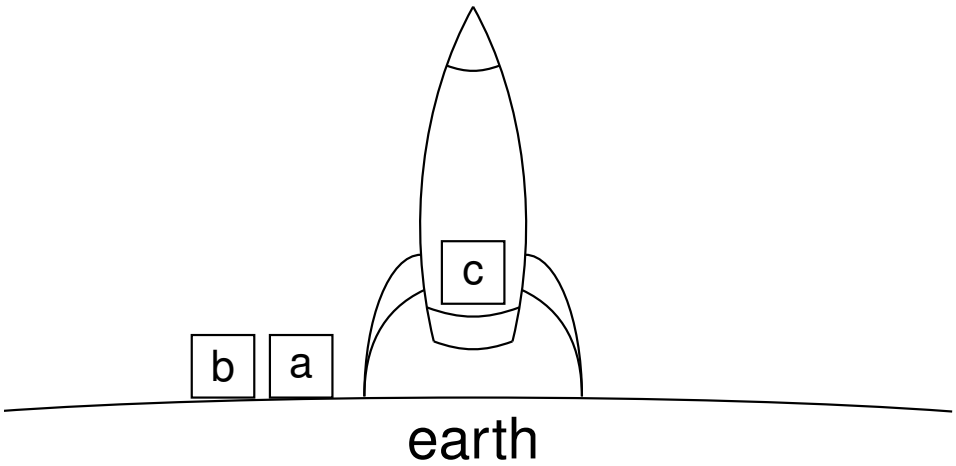
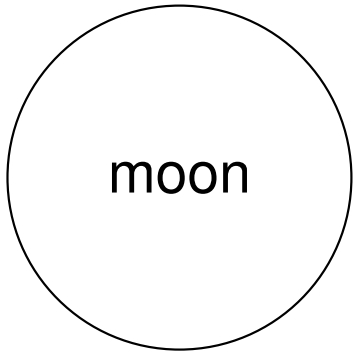
When Not to Act?

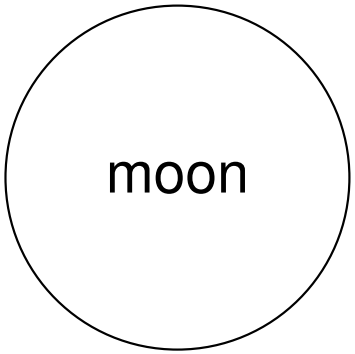
Michael Siebers

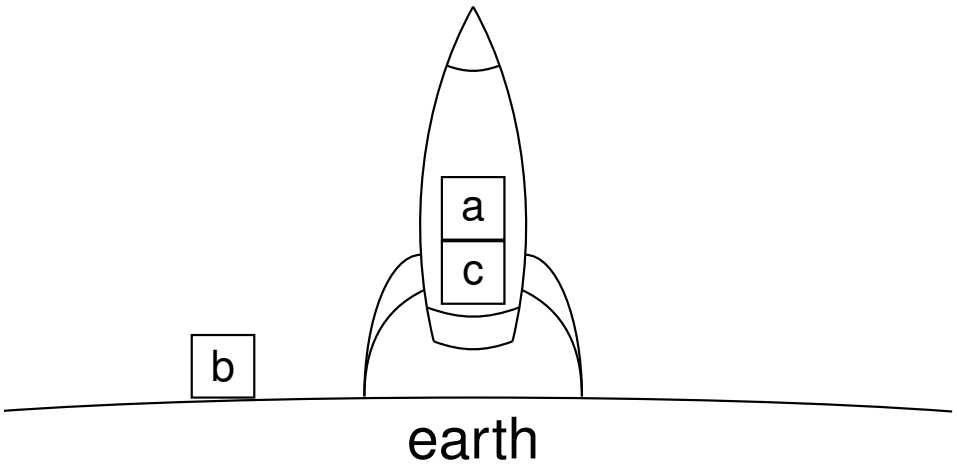
Cognitive Systems Group
Faculty Information Systems and
Applied Computer Science
Otto-Friedrich-Universität Bamberg

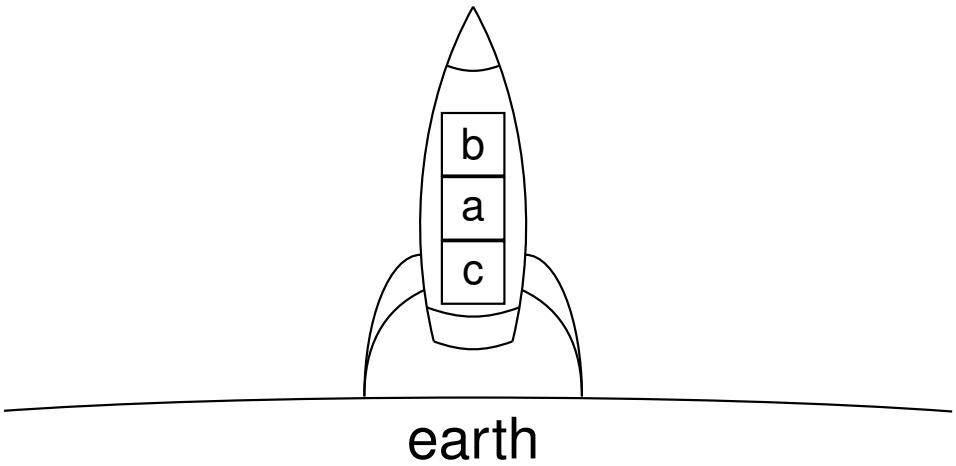
26th International Conference on Inductive Logic Programming
September 6, 2016

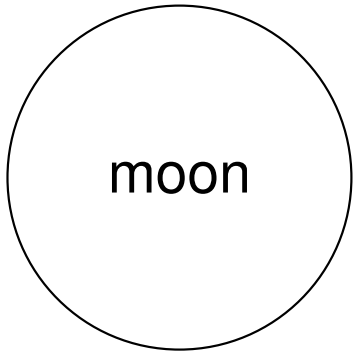

c
b
a
earth

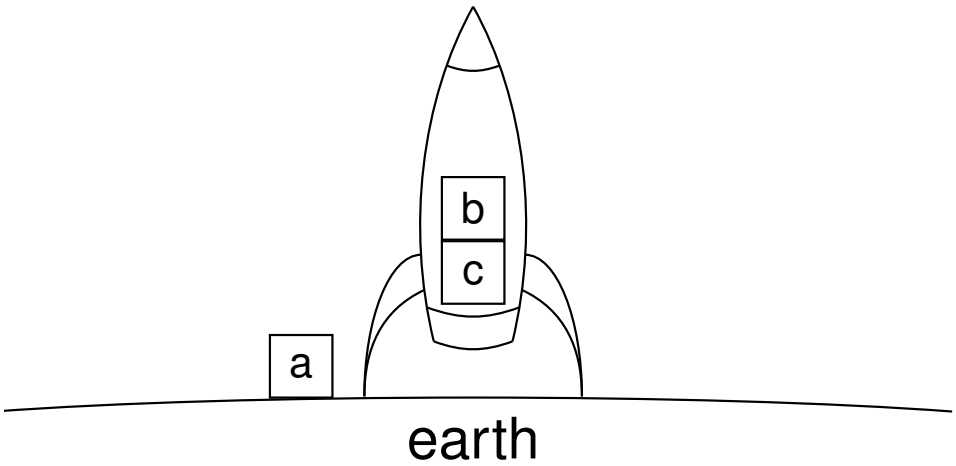

moon

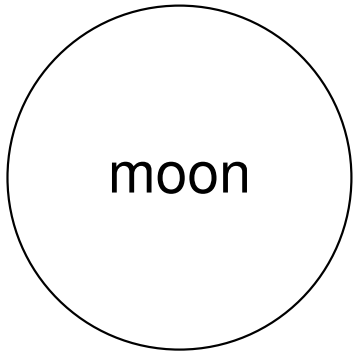

moon

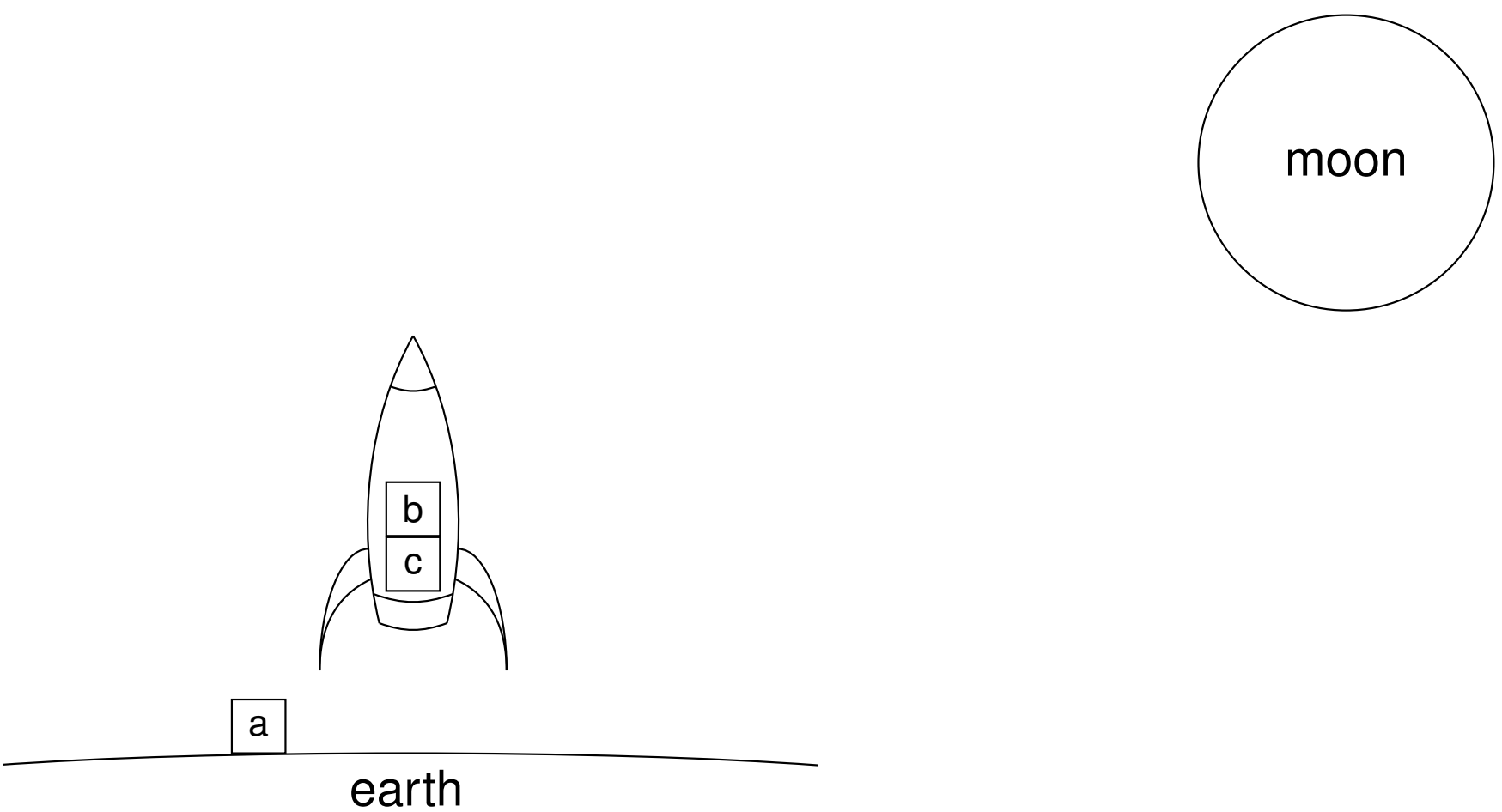

a
c
b
earth

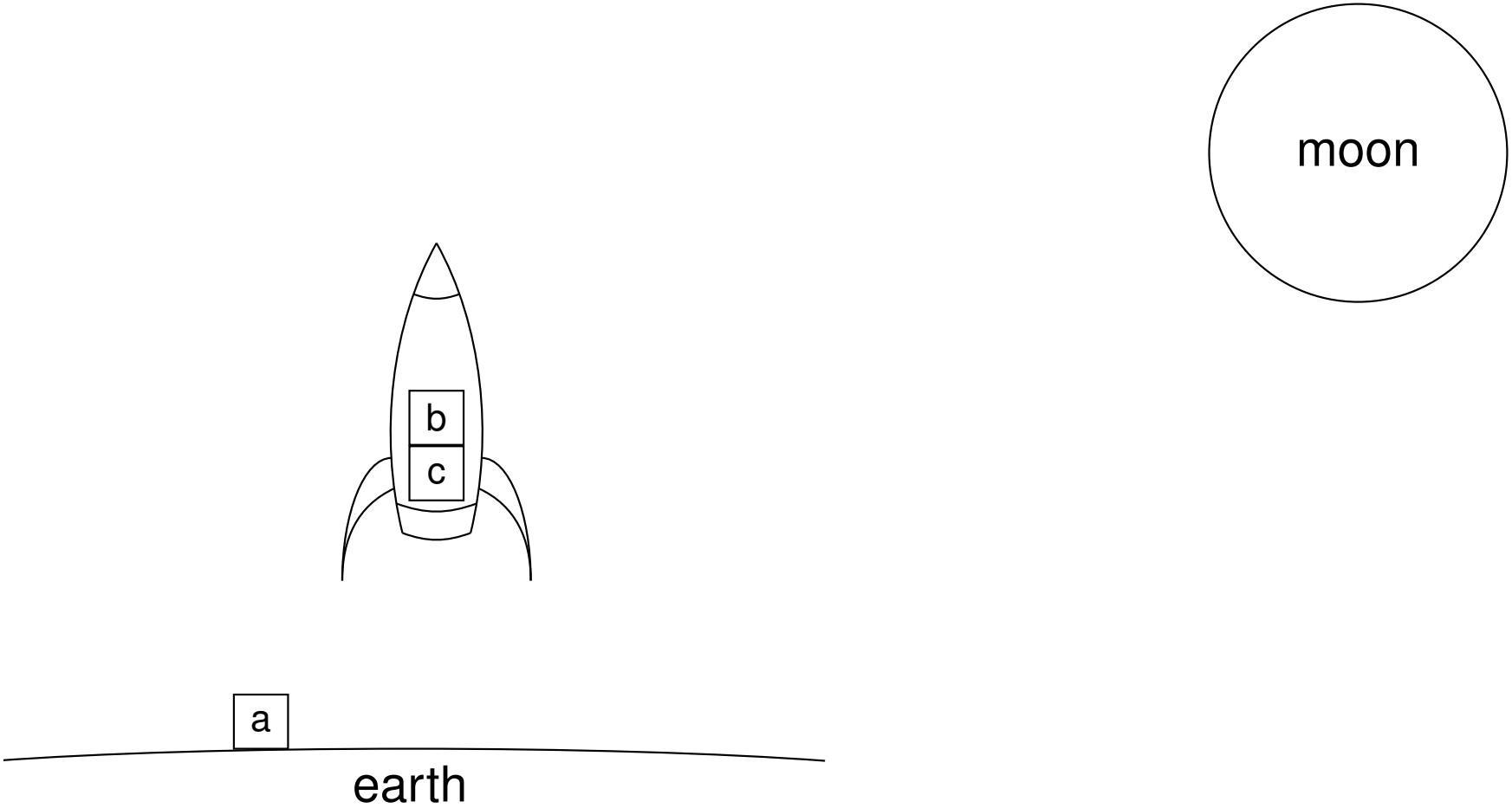

b
a
c
earth

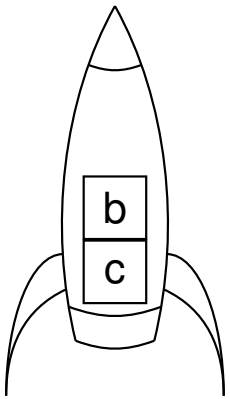

moon

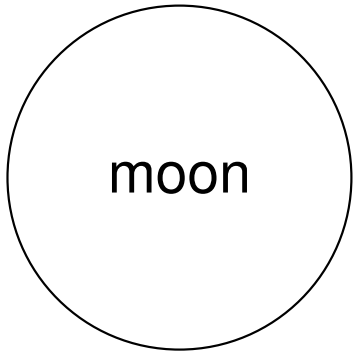

b
c
a
earth


moon

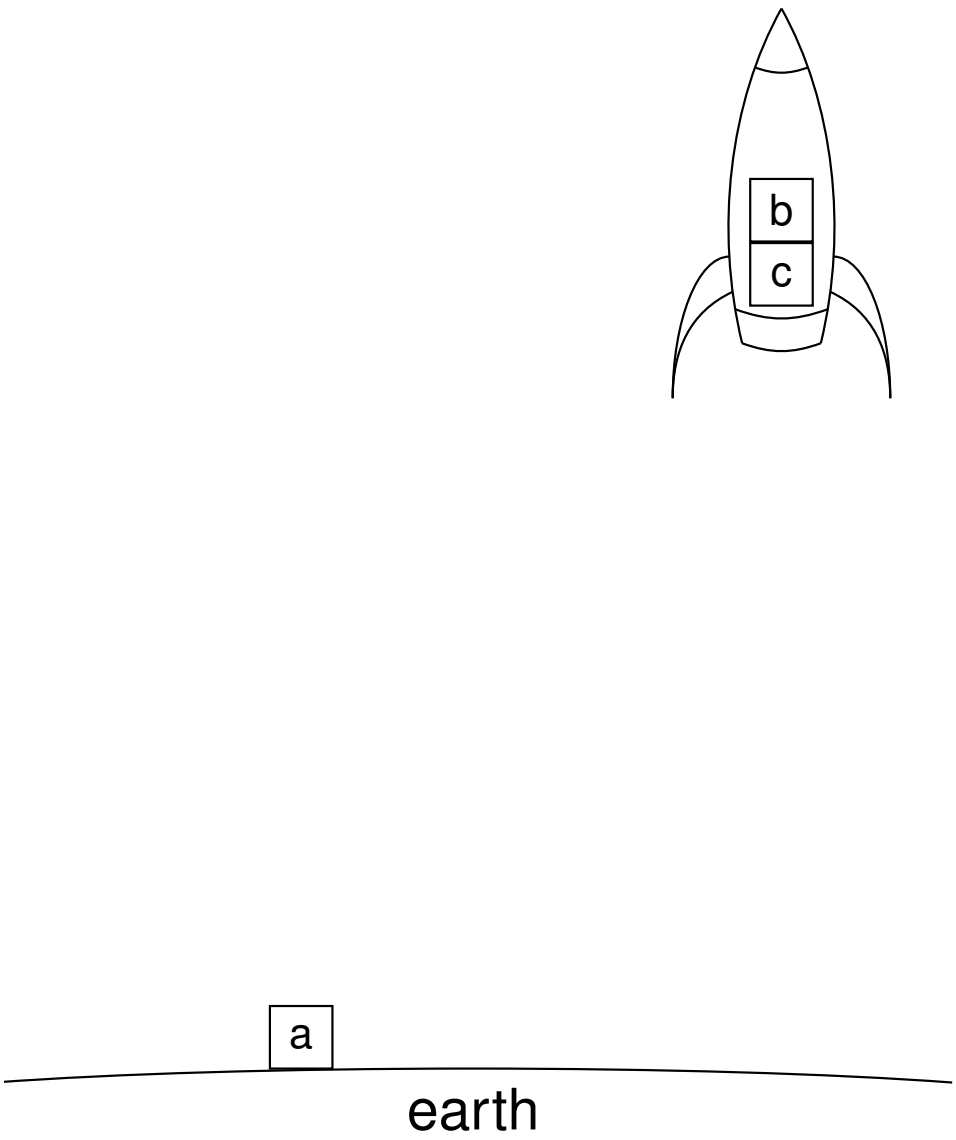

moon
b
c
a
earth

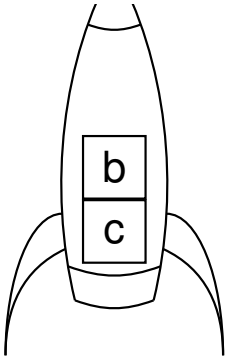

moon
b
c
a
earth

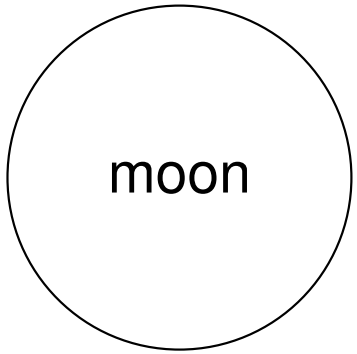

b
c


moon

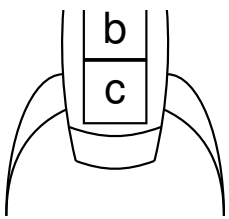

a
earth

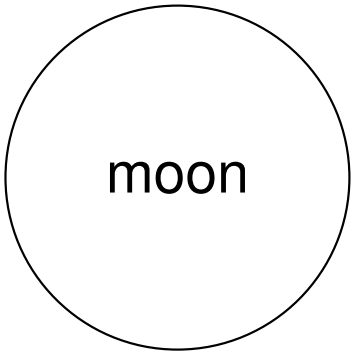

b
c
moon
a
earth


b
c

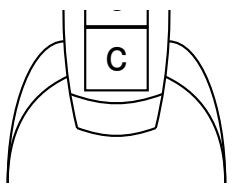

moon

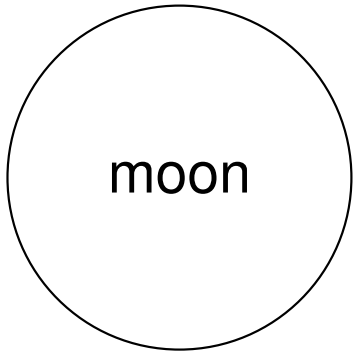

a
earth


b
c

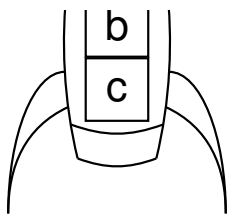

moon

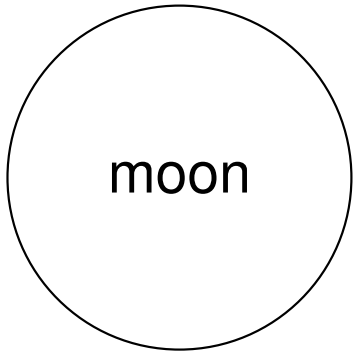

a
earth


c

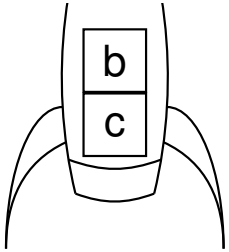

moon

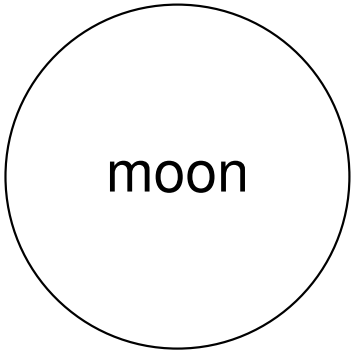

a
earth


b
c

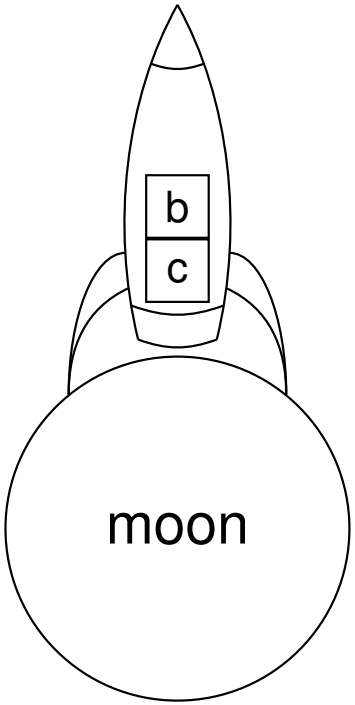

moon


a
earth

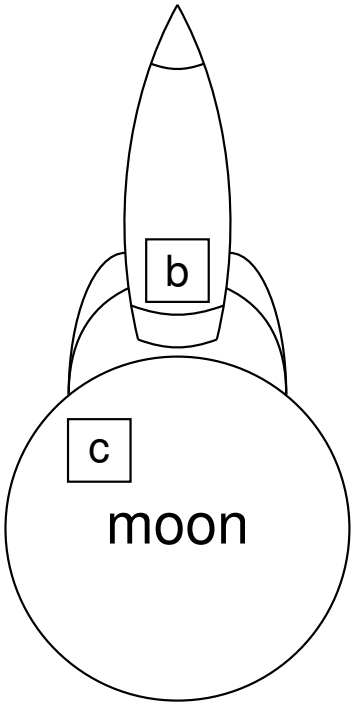

b
c


moon

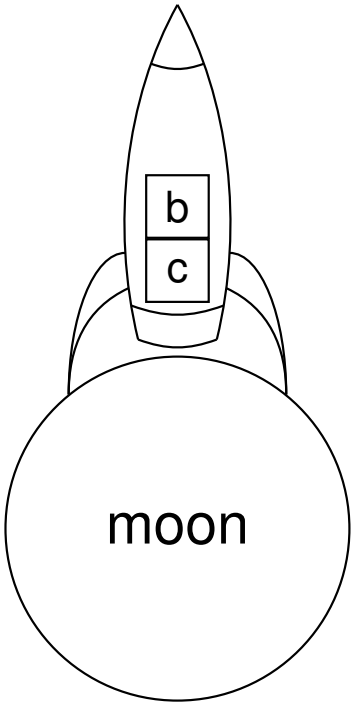

a
earth


b
c
moon


a
earth


b
c
moon


a
earth

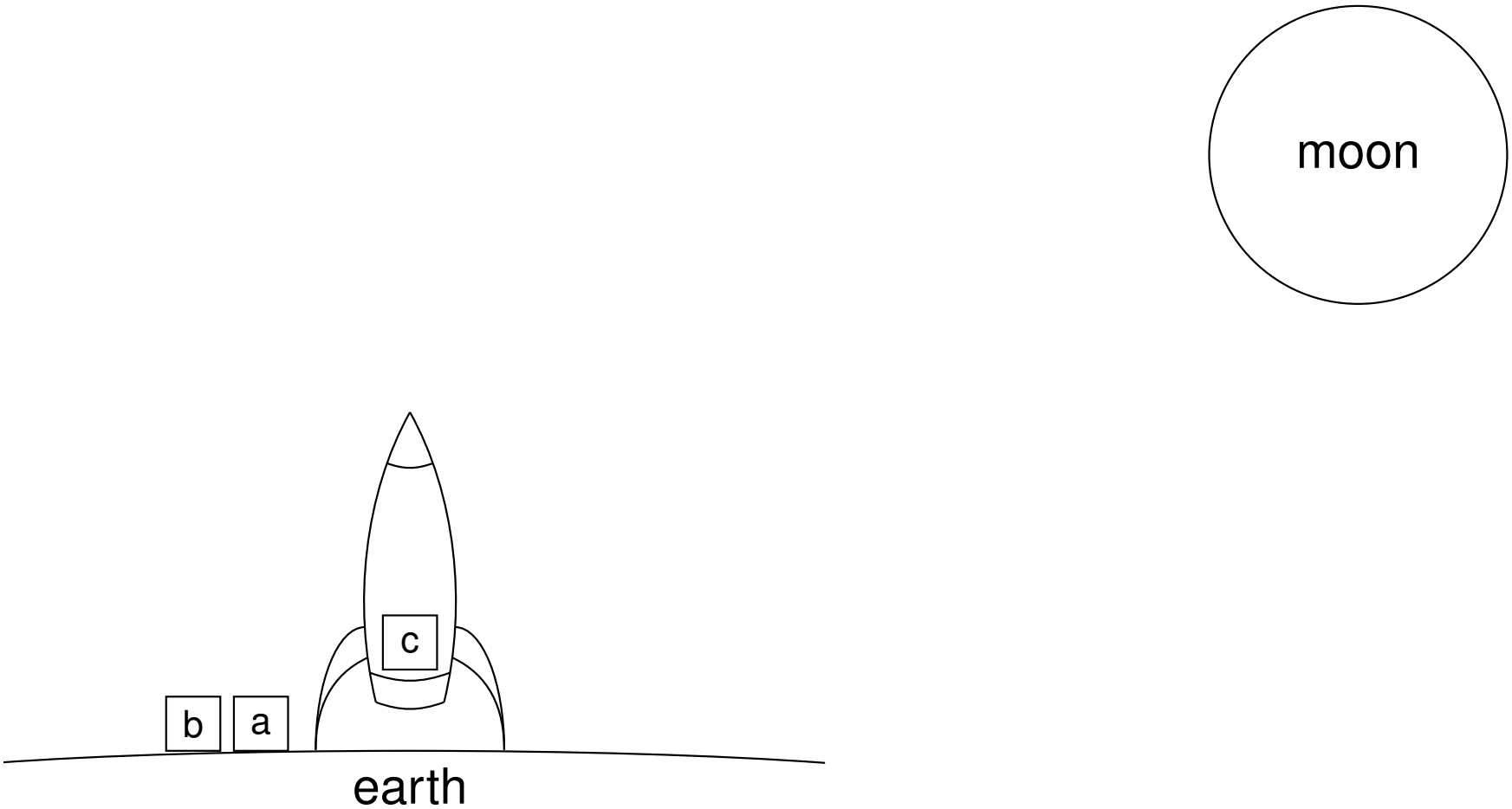

b
c
moon


a
earth

Outline

- Automated Planning
- Avoidance Expressions
- Inducing Avoidance Expressions
- Proof of Concept

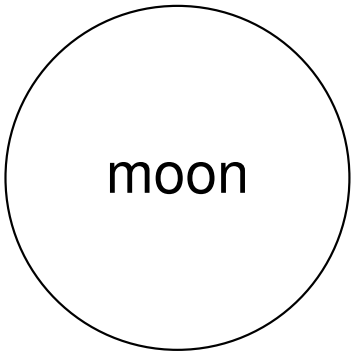
Automated Planning

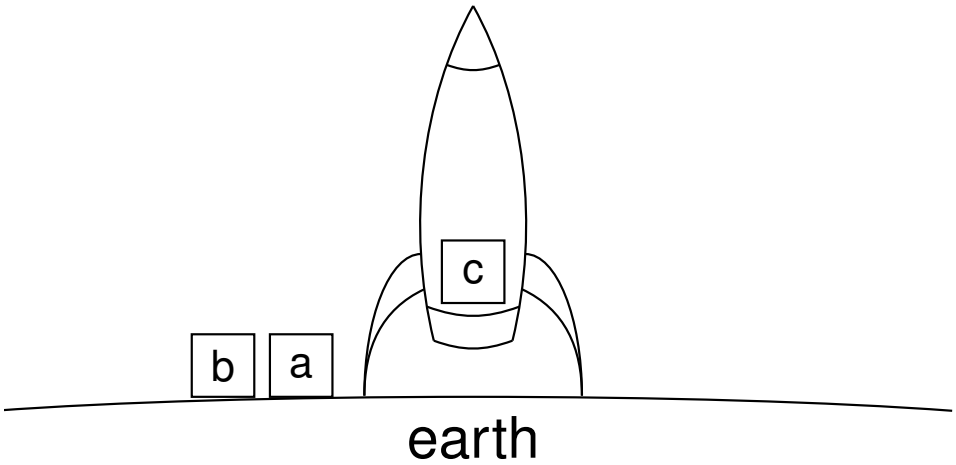


Automated Planning

States

$s = \{\texttt{on-earth}(\text{a}), \texttt{on-earth}(\text{b}), \texttt{loaded}(\text{c}), \texttt{on-earth}(\text{rocket})\}$





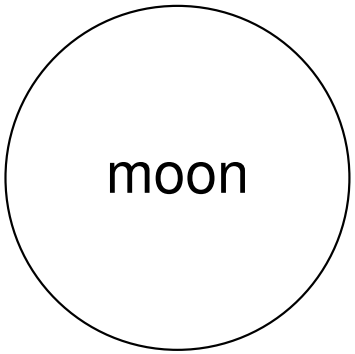
Automated Planning

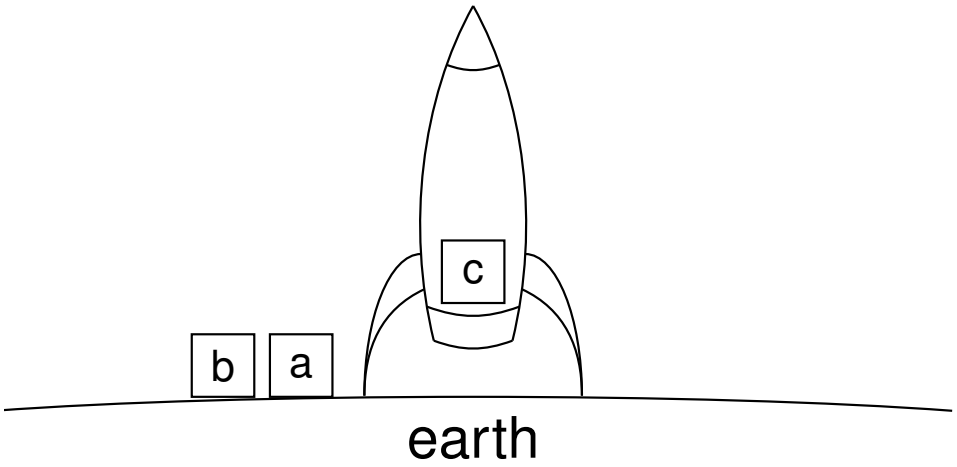
States

$s = \{\texttt{on-earth}(\text{a}), \texttt{on-earth}(\text{b}), \texttt{loaded}(\text{c}), \texttt{on-earth}(\text{rocket})\}$

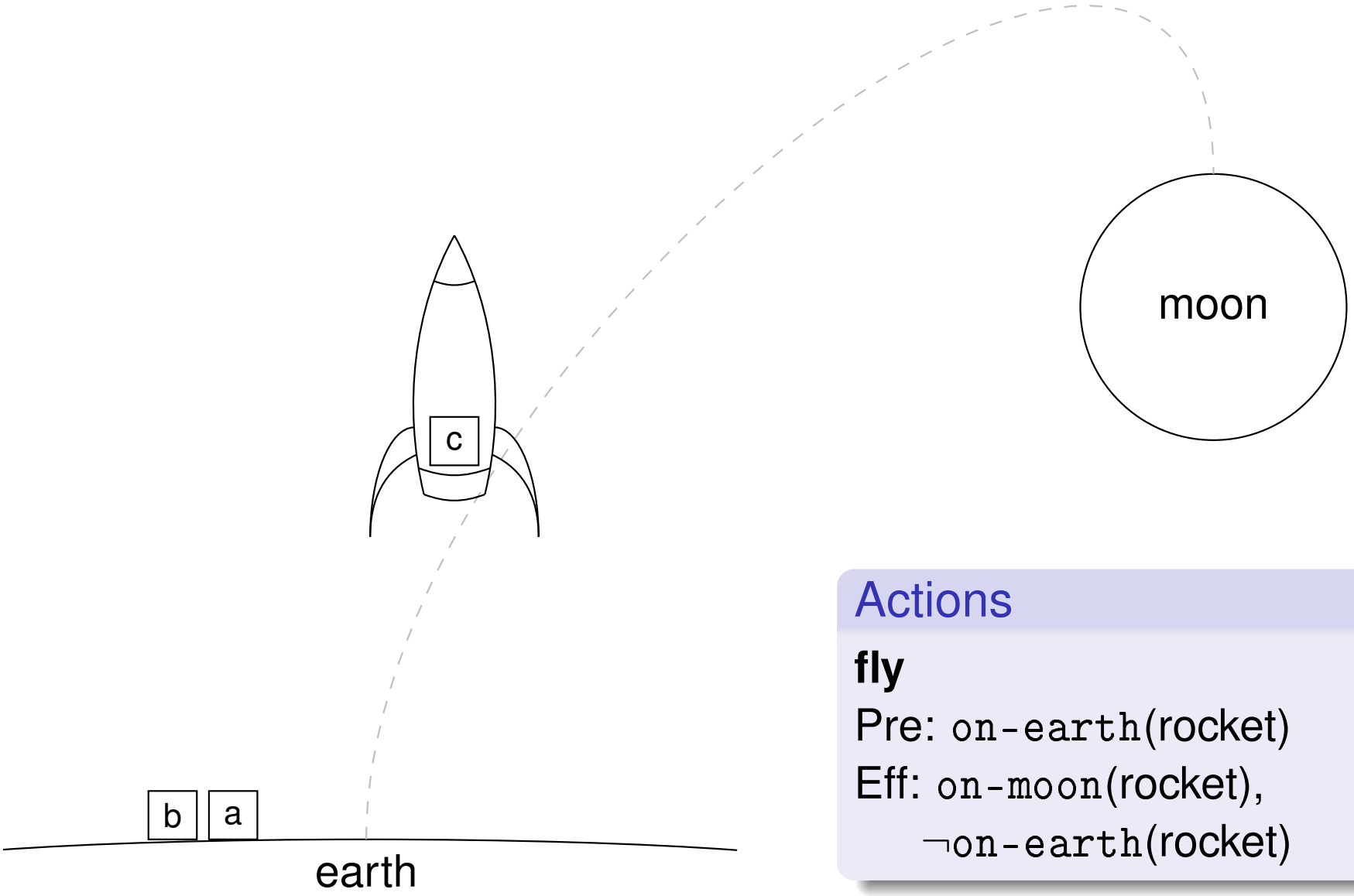
Goal

$g = \{\texttt{on-moon}(\text{a}), \texttt{on-moon}(\text{b}), \texttt{on-moon}(\text{c})\}$





Automated Planning



Actions

fly

Pre: `on-earth`(rocket)

Eff: `on-moon`(rocket),
¬`on-earth`(rocket)

Avoidance Expressions

Basics

- formula
- relates to
  - state
  - action
  - goal

Avoidance Expressions

Basics

- formula
- relates to
  - state
  - action
  - goal

Syntax

- state marker $:\mathfrak{s}(I)$
- goal marker $:\mathfrak{g}(I)$
- $\neg$, $\wedge$, $\vee$, $\forall$, $\exists$

Avoidance Expressions

Basics

- formula
- relates to
  - state
  - action
  - goal

Syntax

- state marker $:\mathtt{s}(l)$
- goal marker $:\mathtt{g}(l)$
- $\neg$, $\wedge$, $\vee$, $\forall$, $\exists$

Avoid unloading *a*

$:\mathtt{g}(\mathtt{on\text{-}moon}(a)) \wedge :\mathtt{s}(\mathtt{on\text{-}earth}(\mathrm{rocket}))$

Inducing Avoidance Expressions

Examples

- state-action-goal tuples
- exhaustive exploration

Inducing Avoidance Expressions

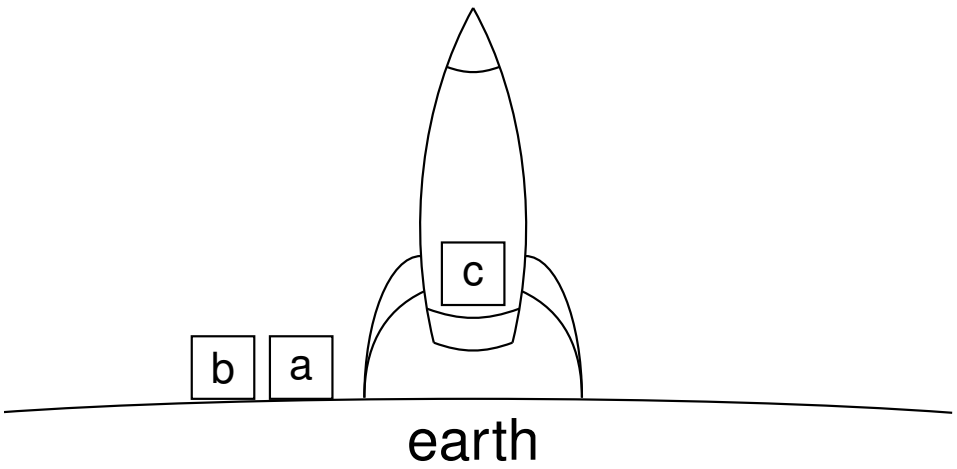
Examples

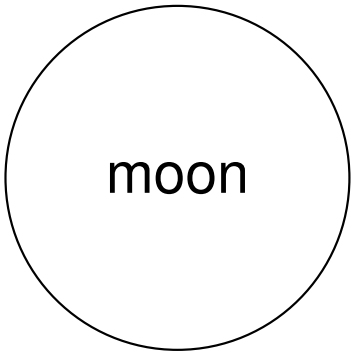
- state-action-goal tuples
- exhaustive exploration

ILP

- state id
- extend and replicate predicates
- target predicate

Induction Example





Induction Example

State s_5

```
s-on-earth(a,s5).
s-on-earth(b,s5).
s-on-earth(rocket,s5).
s-loaded(c,s5).
g-on-moon(a,s5).
g-on-moon(b,s5).
g-on-moon(c,s5).
```

Target Relation

```
avoid-fly(s5).
```

Proof-of-concept

Design

- rocket domain
- different problems
- 122 examples
- FOIL

Proof-of-concept

Design

- rocket domain
- different problems
- 122 examples
- FOIL

Avoid **fly**

$\exists x(\texttt{:s}(\texttt{on-earth}(x)) \wedge \texttt{:g}(\texttt{on-moon}(x))) \vee$
$\exists x(\texttt{:s}(\texttt{loaded}(x)) \wedge \texttt{:g}(\texttt{on-earth}(x)))$

Take-away Message

If your problem can be encoded as an automated-planning problem then you can learn additional domain knowledge from small problem instances.