Planning and Optimization
X1. Hands-On and Repetition

Florian Pommerening

Universität Basel

September 25, 2017

X1.1 PDDL

X1.2 Hands-On

Introduction

Hands-On: Outline for this week
Working with an existing planning system (Fast Downward).

- ▶ Domain modeling
- ▶ Recognizing the difference: blind vs. informed planning
- ▶ Implementation in Fast Downward

Hands-On: Overview

Chapter overview: hands-on

- ▶ 1. The Planning Domain Definition Language (PDDL)
- ▶ 2. Getting to Know a Planner
- ▶ 3. Heuristics
- ▶ 4. A* search algorithm

X1.1 PDDL

Representation of State Spaces

Representation of State Spaces
- ▶ explicit graphs
- ▶ black box
- ▶ declarative representations
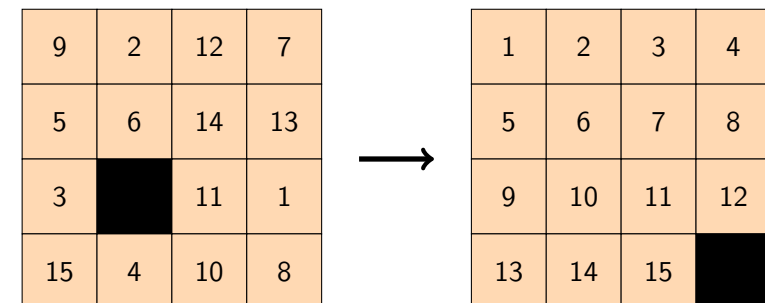
In this Course: Declarative Representations
- ▶ compact description of state space as input to algorithms
  ↝ state space exponentially larger than input
- ▶ algorithms operate directly on compact description
- ↝ allows automatic reasoning about problem (abstractions etc.)

Representation of State Spaces

PDDL: Planning Domain Definition Language
- ▶ PDDL is the standard language used in practice
  to describe planning tasks.
- ▶ descriptions in (restricted) predicate logic instead of
  propositional logic (↝ even more compact)
- ▶ There exist defined PDDL fragments for STRIPS and ADL;
  many planners only support the STRIPS fragment.
- ▶ In this week: restriction to STRIPS

Illustrating Example: 15-Puzzle

15-Puzzle in PDDL

Example: 15-Puzzle in PDDL

X1.2 Hands-On

Hands-On

clone the course repository
```
hg clone https://bitbucket.org/aibasel/planopt-hs17
```

update the course repository
```
cd planopt-hs17
hg pull -u
```

compile the planner
```
cd hands-on-1/fast-downward
./build.py
```

work on the hands-on exercises
- ▶ evaluate and modify the 15-puzzle
- ▶ model your own domain