Huy Viet Nguyen

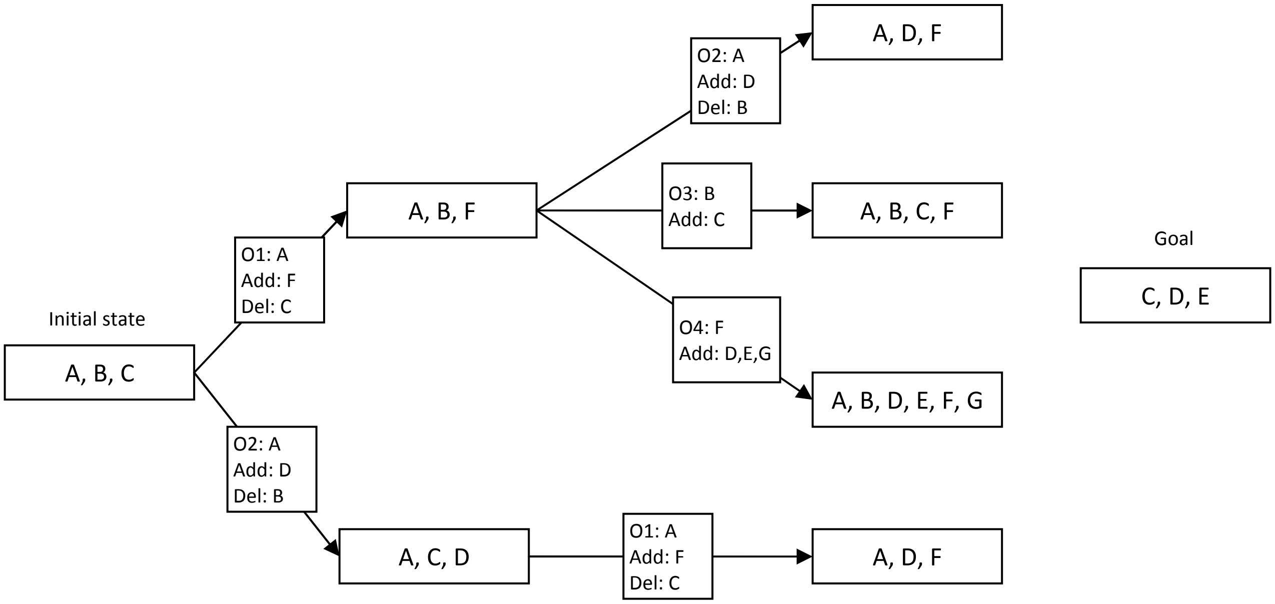
CS 2710: Foundations of Artificial Intelligence

Homework #6

Problem 1: Classical Planning

(a) State-Space Search (no repeated states)

The search tree with a depth limit = 2



As in the above search tree there's no state contains the Goal: {C, D, E}, so the search with depth limit = 2 cannot find any plan to achieve the goal. However the search with depth limit = 3 can obtain the goal, e.g. apply O4 to the state {A, B, C, F} or apply O3 to the state {A, B, D, E, F, G} to get {A, B, C, D, E, F, G}.

(d) Satisfiability

Planning domain:

Predicates: {A, B, C, D, E, F, G}

Initial state: {A, B, C}

Goal state: {C, D, E}

Operator: O1

Preconditions: A

Effects: $F \wedge \sim C$

Encode initial state[*]: $A_0 \wedge B_0 \wedge C_0 \wedge \sim D_0 \wedge \sim E_0 \wedge \sim F_0 \wedge \sim G_0$

*: Subscript indicates step

Encode goal state at step 1: $C_1 \wedge D_1 \wedge E_1$

Encode operator O1:

$O1_0 \Rightarrow A_0$

$O1_0 \Rightarrow F_1$

$O1_0 \Rightarrow \sim C_1$

Encode frame axioms:

$\sim F_0 \wedge F_1 \Rightarrow O1_0$

$C_0 \wedge \sim C_1 \Rightarrow O1_0$

Complete SAT instance

$A_0 \wedge B_0 \wedge C_0 \wedge \sim D_0 \wedge \sim E_0 \wedge \sim F_0 \wedge \sim G_0 \wedge$

$C_1 \wedge D_1 \wedge E_1 \wedge$

$O1_0 \Rightarrow A_0 \wedge$

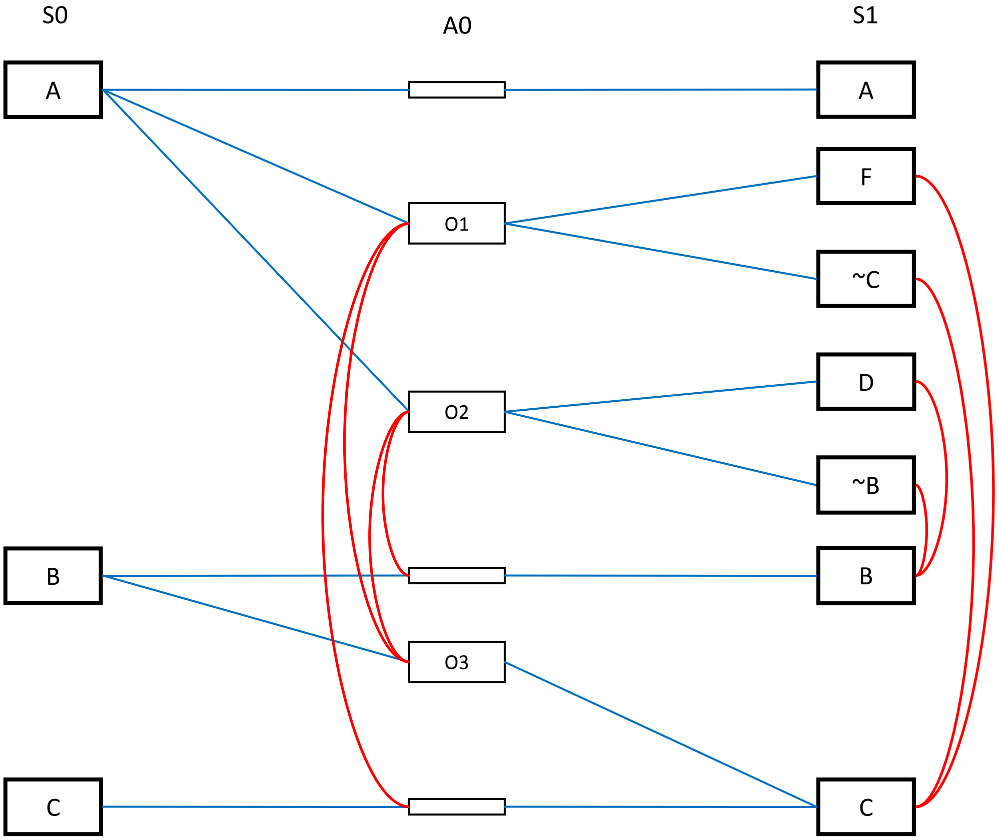
$O1_0 \Rightarrow F_1 \wedge$

$O1_0 \Rightarrow \sim C_1 \wedge$

$\sim F_0 \wedge F_1 \Rightarrow O1_0 \wedge$

$C_0 \wedge \sim C_1 \Rightarrow O1_0$

(b) Planning Graph



Red curve: mutex link

Blue line: connection between action and precondition/effect

In S1 level, C and D are present and not mutually exclusive, but E is absent, so EXTRACT_SOLUTION for this planning graph returns failure.

(c) Partial Ordered Planning

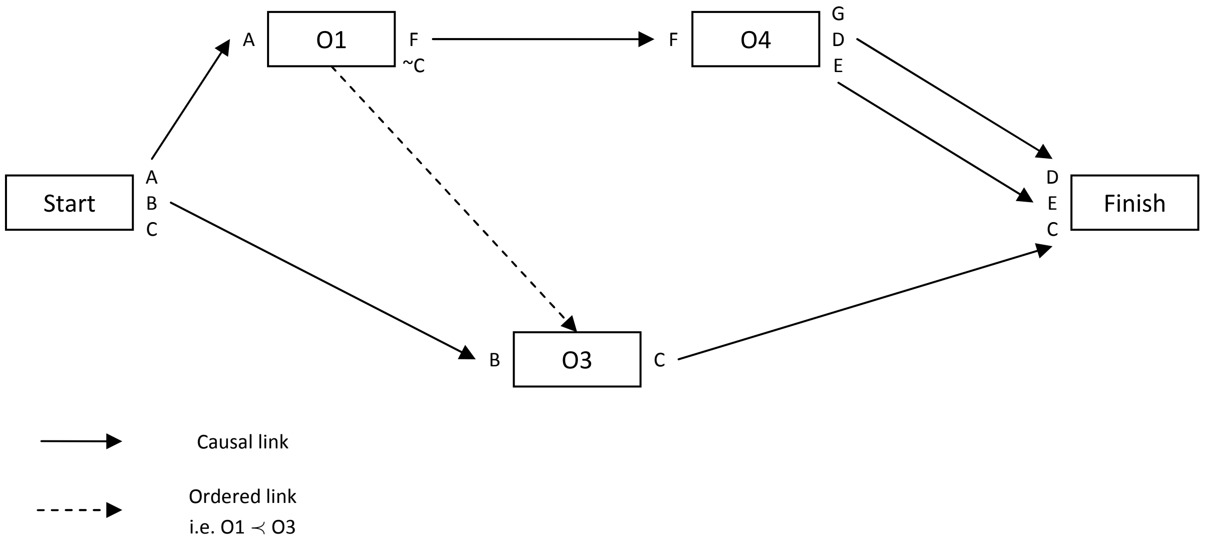
Boxes represent actions, with preconditions on the left and effects on the right.

The *Start* action which presents the initial state has no precondition, while the *finish* action which presents the goal state has no effect.

Example of a linearization for this partial-order plan:

Start → O1 → O3 → O4 → Finish or

Start → O1 → O4 → O3 → Finish



Problem 2:

(a) *PDDL sentences for Shakey's six actions*

```
(:predicates
(at ?x ?y) (on ?x ?y) (in ?x ?y) (box ?x) (turnedOn ?x))
(:action Go
:parameters (?x ?y ?r)
:precondition (and (on Shakey Floor) (at Shakey ?x) (in ?x ?r) (in ?y ?r))
:effect (and (at Shakey ?y) (not (at Shakey ?x))))
(:action Push
:parameters (?b ?x ?y ?r)
:precondition (and (on Shakey Floor) (at Shakey ?x) (box ?b) (at ?b ?x) (in ?x
?r) (in ?y ?r))
:effect (and (at Shakey ?y) (at ?b ?y) (not (at Shakey ?x)) (not (at ?b ?x))))
(:action ClimbUp
:parameters (?x ?b)
:precondition (and (on Shakey Floor) (at Shakey ?x) (at ?b ?x) (box ?b))
:effect (and (on Shakey ?b) (not (on Shakey Floor))))
(:action ClimbDown
:parameters (?b)
:precondition (and (on Shakey ?b) (box ?b))
:effect (and (on Shakey Floor) (not (on Shakey ?b))))
(:action TurnOn
:parameters (?s ?b)
:precondition (and (on Shakey ?b) (box ?b) (at ?b ?s))
:effect (turnedOn ?s))
(:action TurnOff
:parameters (?s ?b)
:precondition (and (on Shakey ?b) (box ?b) (at ?b ?s))
:effect (not (turnedOn ?s)))
```

** PDDL sentences for Shakey's initial state*

```
(:objects
Shakey Floor Corridor START
Room1 Room2 Room3 Room4
Door1 Door2 Door3 Door4
Switch1 Switch2 Switch3 Switch4
Box1 Box2 Box3 Box4
BX1 BX2 BX3 BX4)
(init:
(on Shakey Floor) (at Shakey START) (in START Room3)
(box Box1) (at Box1 BX1) (in BX1 Room1)
(box Box2) (at Box2 BX2) (in BX2 Room1)
(box Box3) (at Box3 BX3) (in BX3 Room1)
(box Box4) (at Box4 BX4) (in BX4 Room1)
(in Door1 Room1) (in Door1 Corridor) (in Switch1 Room1) (turnedOn Switch1)
(in Door2 Room2) (in Door2 Corridor) (in Switch2 Room2) (not (turnedOn Switch2))
(in Door3 Room3) (in Door3 Corridor) (in Switch3 Room3) (not (turnedOn Switch3))
(in Door4 Room4) (in Door4 Corridor) (in Switch4 Room4) (turnedOn Switch4))
```

(b) See attached files for domain, initial state, goal state and trace of Blackbox's run.

The output plan

```
----------------------------------------------------
Begin plan
1 (go start door3 room3)
2 (go door3 door1 corridor)
3 (go door1 bx2 room1)
4 (push box2 bx2 door1 room1)
5 (push box2 door1 door2 corridor)
6 (push box2 door2 switch2 room2)
End plan
----------------------------------------------------
```

(c) I choose number of boxes in Room1 and number of boxes has to be moved as two parameters to scale up. Looking back to the original problem, there are 4 boxes in room1 and Shakey plans to move Box2 into Room2. With two chosen parameters, in addition to the original problem, I create 7 extra problems for test; run test problems using Blackbox planner then record number of actions in found plan and run-time in ms.

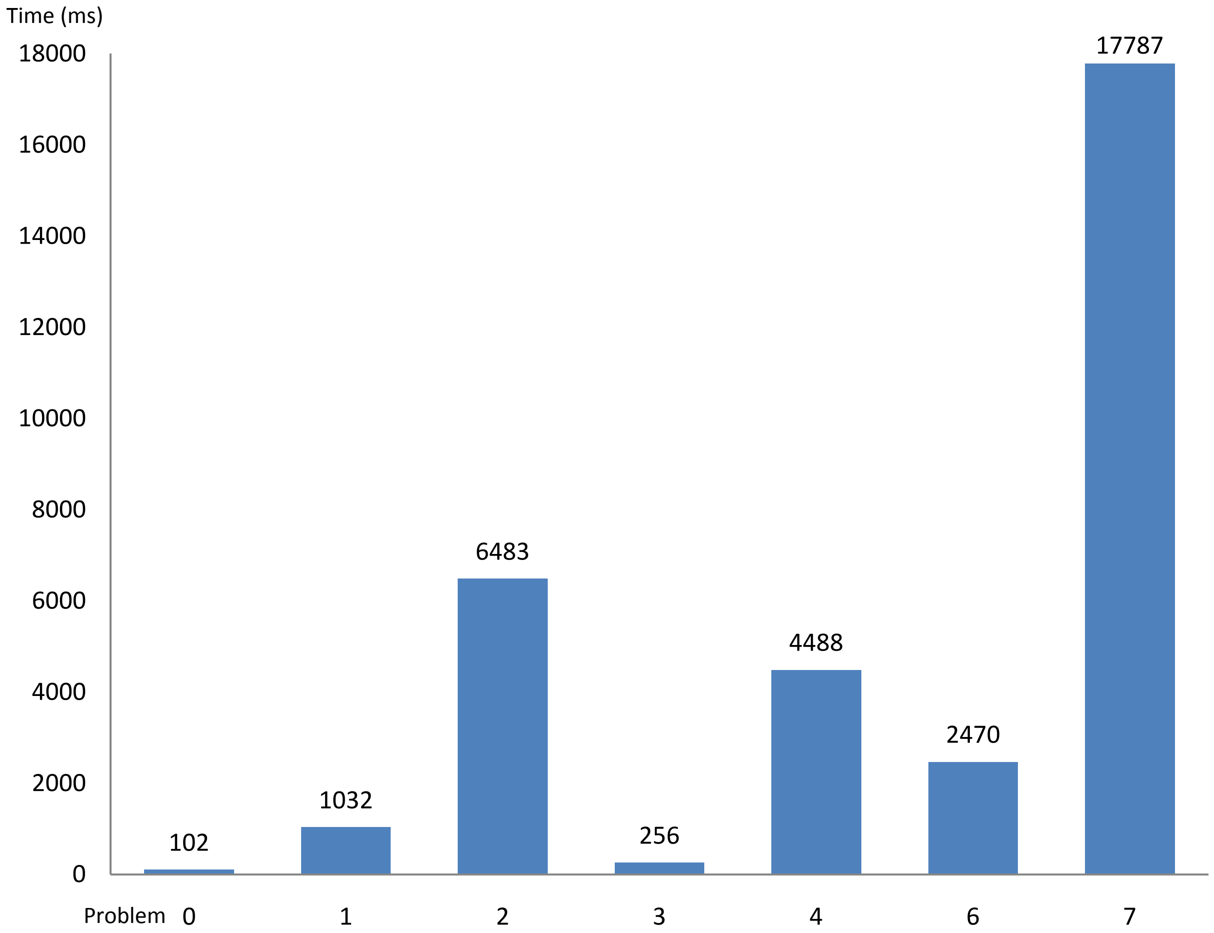
Problem#	# boxes in Room1	Boxes have to be moved	# actions	run-time (ms)
0*	4	Box2 into Room2	6	102
1	8	Box2 into Room2	6	1032
2	12	Box2 into Room2	6	6483
3	4	Box2 into Room2 Box1 into Room1	8	256
4	4	Box2 into Room2 Box1 into Room1 Box3 into Room3	14	4488
5	4	Box2 into Room2 Box1 into Room1 Box3 into Room3 Box4 into Room4	N/A	N/A
6	8	Box2 into Room2 Box1 into Room1	8	2470
7	12	Box2 into Room2 Box1 into Room1	8	17787

[*]: original problem.

All test problems can be divided into 3 groups:

- problem1, problem2: scaling only number of boxes in Room1, w.r.t problem0.
- problem3, problem4, problem5: scaling only number of boxes have to be moved, w.r.t problem0.
- problem6, problem7: scaling both two chosen parameters, w.r.t problem0.

The recorded run-time of test problems using Blackbox are shown in the below graph. We can see that the difficulty of problem increase when scaling up the parameters. In particular, Blackbox fails to solve the problem5 in limited time of 10 minutes.



```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; World of Shakey the robot
;;; CS 2710 HW6 Huy Viet Nguyen
;;; DOMAIN FILE
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(define (domain shakey-world)
    (:requirements :strips)

    (:predicates
        (at ?x ?y) (on ?x ?y) (in ?x ?y) (box ?x) (turnedOn ?x))

    (:action Go
    :parameters (?x ?y ?r)
    :precondition (and (on Shakey Floor) (at Shakey ?x) (in ?x ?r) (in ?y ?r))
    :effect (and (at Shakey ?y) (not (at Shakey ?x))))

    (:action Push
    :parameters (?b ?x ?y ?r)
    :precondition (and (on Shakey Floor) (at Shakey ?x) (box ?b) (at ?b ?x)
    (in ?x ?r) (in ?y ?r))
    :effect (and (at Shakey ?y) (at ?b ?y) (not (at Shakey ?x)) (not (at ?b ?x))))

    (:action ClimbUp
    :parameters (?x ?b)
    :precondition (and (on Shakey Floor) (at Shakey ?x) (at ?b ?x) (box ?b))
    :effect (and (on Shakey ?b) (not (on Shakey Floor))))

    (:action ClimbDown
    :parameters (?b)
    :precondition (and (on Shakey ?b) (box ?b))
    :effect (and (on Shakey Floor) (not (on Shakey ?b))))

    (:action TurnOn
    :parameters (?s ?b)
    :precondition (and (on Shakey ?b) (box ?b) (at ?b ?s))
    :effect (turnedOn ?s))

    (:action TurnOff
    :parameters (?s ?b)
    :precondition (and (on Shakey ?b) (box ?b) (at ?b ?s))
    :effect (not (turnedOn ?s)))
)
```

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; World of Shakey the robot
;;; CS 2710 HW6 Huy Viet Nguyen
;;; PROBLEM FILE
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(define (problem SHAKEY)
    (:domain shakey-world)

    (:objects
        Shakey Floor Corridor START
        Room1 Room2 Room3 Room4
        Door1 Door2 Door3 Door4
        Switch1 Switch2 Switch3 Switch4
        Box1 Box2 Box3 Box4
        BX1 BX2 BX3 BX4)

    (:init
        (on Shakey Floor) (at Shakey START) (in START Room3)

        (box Box1) (at Box1 BX1) (in BX1 Room1)
        (box Box2) (at Box2 BX2) (in BX2 Room1)
        (box Box3) (at Box3 BX3) (in BX3 Room1)
        (box Box4) (at Box4 BX4) (in BX4 Room1)

        (in Door1 Room1) (in Door1 Corridor) (in Switch1 Room1) (turnedOn Switch1)
        (in Door2 Room2) (in Door2 Corridor) (in Switch2 Room2) (not (turnedOn Switch2))
        (in Door3 Room3) (in Door3 Corridor) (in Switch3 Room3) (not (turnedOn Switch3))
        (in Door4 Room4) (in Door4 Corridor) (in Switch4 Room4) (turnedOn Switch4))

    (:goal (at Box2 Switch2))
)
```

```
blackbox version 42
command line:  blackbox -o shk_domain.pddl -f shk_problem.pddl

Begin solver specification
    -maxint        0   -maxsec 10.000000  graphplan
    -maxint        0   -maxsec 0.000000  chaff
End solver specification
Loading domain file: shk_domain.pddl
Loading fact file: shk_problem.pddl
Problem name: shakey
Facts loaded.
time: 1, 33 facts and 3 exclusive pairs.
time: 2, 36 facts and 15 exclusive pairs.
time: 3, 43 facts and 78 exclusive pairs.
time: 4, 67 facts and 666 exclusive pairs.
time: 5, 83 facts and 1156 exclusive pairs.
time: 6, 100 facts and 1764 exclusive pairs.
Goals first reachable in 6 steps.
1406 nodes created.

####################################################
goals at time 7:
  at_box2_switch2



----------------------------------------------------
Invoking solver graphplan
Result is Sat
Iteration was 50
Performing plan justification:
   0 actions were pruned in 0.00 seconds

----------------------------------------------------
Begin plan
1 (go start door3 room3)
2 (go door3 door1 corridor)
3 (go door1 bx2 room1)
4 (push box2 bx2 door1 room1)
5 (push box2 door1 door2 corridor)
6 (push box2 door2 switch2 room2)
End plan
----------------------------------------------------

6 total actions in plan
0 entries in hash table,
5 total set-creation steps (entries + hits + plan length - 1)
6 actions tried

####################################################
Total elapsed time:   0.09 seconds
Time in milliseconds: 91

####################################################
```