ARTUN SARIOGLU CS 404 HOMEWORK 3 ***23555***

Domain

```
(define (domain fox-goose-beans-domain)
(:requirements :strips)
(:predicates (onLeft ?x) )
(:action cross
   :parameters (?x)
   :precondition(and
      (or
         ( = ?x goose)
         (and
            (= ?x fox)
            (or
               (and
                  (onLeft goose)
                  (not(onLeft beans))
               )
               (and
                  (not(onLeft goose))
                  (onLeft beans)
               )
            )
         )
         (and
            (= ?x beans)
            (or
               (and
                  (onLeft fox)
                  (not(onLeft goose))
               )
               (and
                  (not(onLeft fox))
                  (onLeft goose)
               )
            )
         )
         (and
            (= ?x farmer)
            (or
```

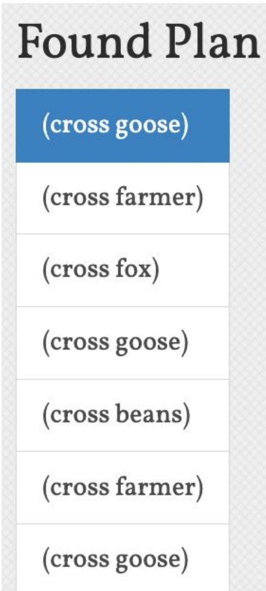
```
            (and
                (onLeft goose)
                (and
                    (not(onLeft fox))
                    (not(onLeft beans))
                )
            )

            (and
                (not(onLeft goose))
                (and
                    (onLeft fox)
                    (onLeft beans)
                )
            )
        )
    )
)

(or
    (and
        (onLeft ?x)
        (onLeft farmer)
    )
    (and
        (not(onLeft ?x))
        (not(onLeft farmer))
    )
)


)
:effect
(and
    (when (onLeft ?x)
    (and
        (not (onLeft ?x))
        (not (onLeft farmer)))
    )
    (when (not (onLeft ?x))
```

```
            (and
                (onLeft ?x)
                (onLeft farmer))
            )
        )
    )
)
```

Problem

```
(define (problem fox-goose-beans)
    (:domain fox-goose-beans-domain)
    (:objects
        fox goose beans farmer
    )
    (:init
        (onLeft fox)
        (onLeft goose)
        (onLeft beans)
        (onLeft farmer)
    )
    (:goal
        (and
            (not(onLeft fox))
            (not(onLeft goose))
            (not(onLeft beans))
            (not(onLeft farmer))
        )
    )
)
```

Observed Output

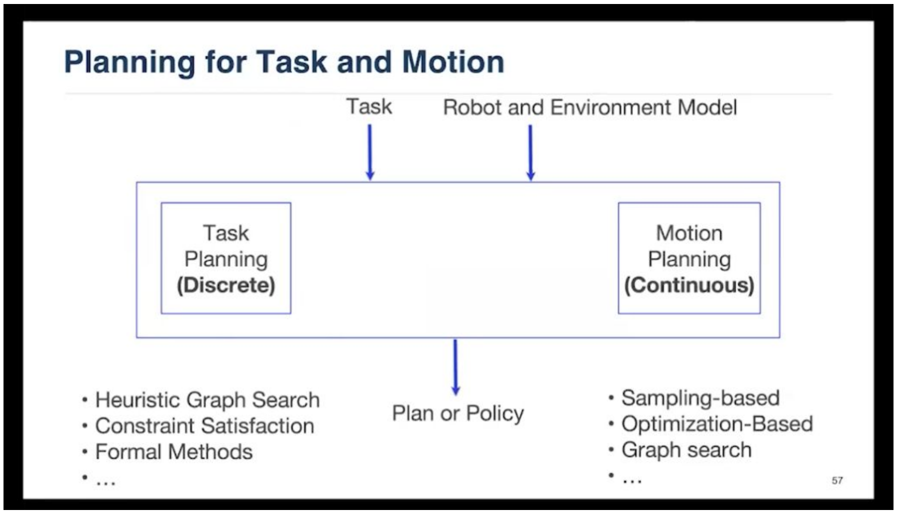


Part 2

1) What is Motion Planning ?

When we are given a movable object and a description of the environment, motion planning is basically reaching the destination state from goal state without any collisions and respecting all the constraints.

2) How is motion planning different from task planning ?



Task Planning	**Motion Planning**
-Requires discrete reasoning	-Requires continuous reasoning

3) Describe three sample real-world applications of motion planning mentioned in the talks.

i) Autonomous Underwater Vehicles (AUVs)

-It has non-linear dynamics.

-Several motion constraints.

-Unknow or predictable environment

- Limited computational resources.

Geometric planner : Computes path from start to to goal state.

Lead path extracted from the geometric planner and it's used to bias the algorithm of the second motion planner(dynamical motion planner) and

with the help of both motion planning algorithms to extract a 2-D

continuous path from a discrete representation of the environment.

(P.S: . I've written what I understood and

also used below paper to understand the high-level meaning of motion

planning on autonomous underwater vehicles.

https://www.researchgate.net/publication/3450401_Path_Planning_for_Autonomous_Underwater_Vehicles)

ii) Robonaut2 of NASA (different examples)

- R2 has manifold constraints which are designed with motion planning.
- In the example on the video, when the robot is turning a valve, it's doing a circular motion . When it tries to push the door, it's doing a whole body motion.
- Above examples show different motions for different cases. It's chosen with manifold constraints.