LaVallle Ch. 1 : Planning to Plan

1/19/18

①

Robot planning : Convert high-level description into a sequence of actions.

Can be executed :

Open-loop – just try, no feedback for adjustments

Closed-loop – react to changes in env received from sensors.
(a.k.a. reactive plan)

We need :

(1) A way to represent state of world

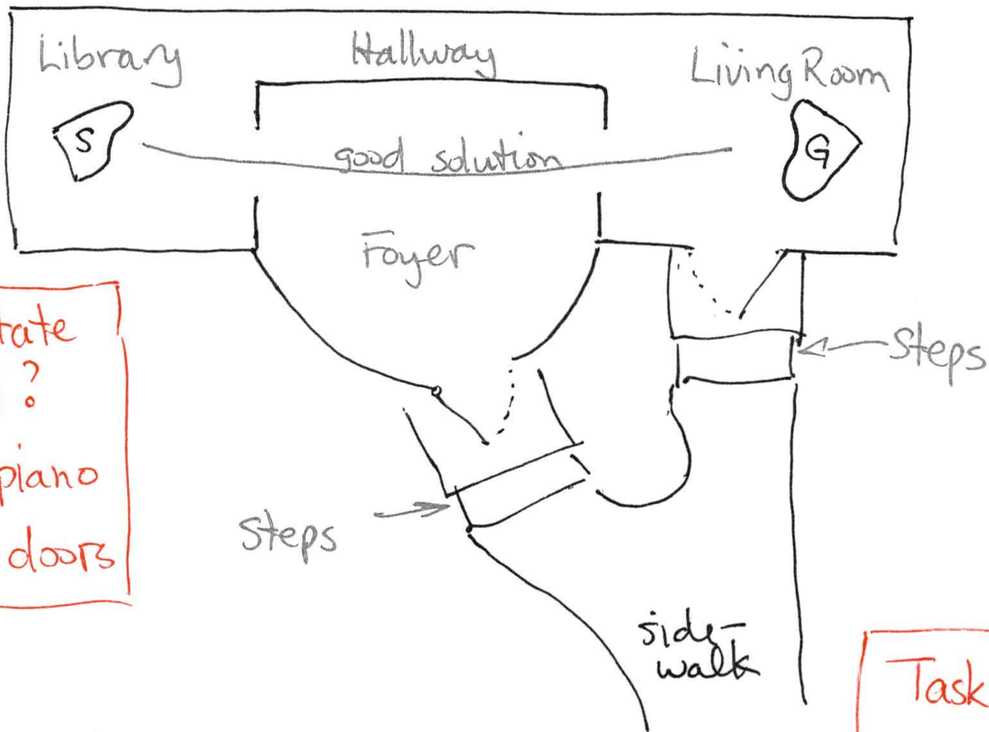
(2) A way to represent tasks

(3) A way to specify robot actions

(4) A way to predict (and possibly sense) how robot actions change state of world

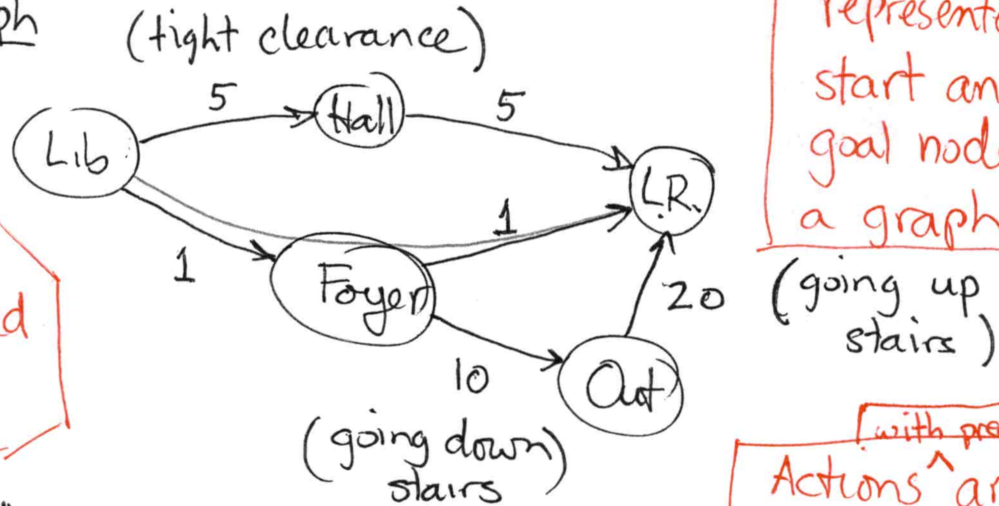
(5) A way to compute the "cost" of a plan. (if we seek optimal plan)

The Piano Movers Problem



What is state of world?
x, y, θ of piano
angles of doors

Graph



Possible states discretized and represented in a graph.

Task is represented as start and goal nodes in a graph

Actions (with predictions!) are represented as arcs in graph

The "best" plan is the one that is cheapest according to your metric.

Incremental costs are represented as arc costs/weights

A little history of planning ...

Early days (1980's) – piano movers problem only considered geometry of world & piano. No movers.

<u>Goal</u>: find feasible solution only (collision-free)

Used "exact" algorithms, i.e. geometry models and exhaustive search

Algorithm design focused on completeness.

<u>Complete</u>: An alg is complete if when a soln exists, it finds one in finite time, and when a soln does not exist, it reports this fact in finite time.

Most efficient algorithm is $\mathcal{O}(2^{D})$ – very bad – where D is # of degrees of freedom of state space.

Never implemented! Not practical $D > 3$

Around ~~200~~ 1990 research shifted to focus on sampling-based algorithms.

e.g. PRM, RRT

Sought feasibility more than optimality

Algorithms aim for probabilistic completeness. i.e., if alg runs long enough and a soln exists, one will be found.

No ~~ga~~ mechanism to guarantee soln non existence.

Around 2005 research shifted to find ~~algs~~ sample-based algs exhibiting probabilistic completeness, asymptotic optimality, and fast convergence

PRM*, RRT*, ~~*~~- - - - , RABIT* (≃ 2015)

Sampling-based methods are easy to implement!

But fundamentally they are still $O(2^D)$

Show videos of planning problems

α-puzzle

Karaman's FPGA-enhanced planner

My manipulation in hand

Others

Mention parallel parking with trailer.

planning w/ differential constraints.
