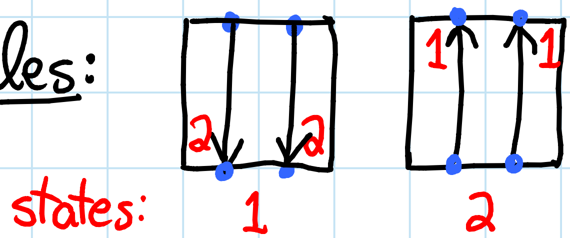
6.892	Class 5	Mar. 6, 2019

Gadget general model: (for robot motion planning)
- locations (entrances / exits)
- states
- transitions $(l_1, s_1) \to (l_2, s_2)$ } s_1: $(l_1) \xrightarrow{s_2} (l_2)$

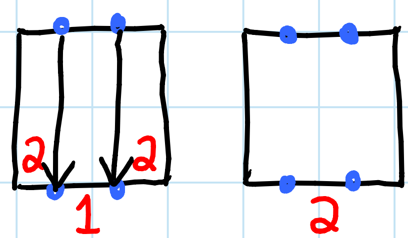
transition graph — visual notation

[Demaine, Grosof, Lynch, Rudoy – FUN 2018]
[Demaine, Hendrickson, Lynch – arXiv 2018]

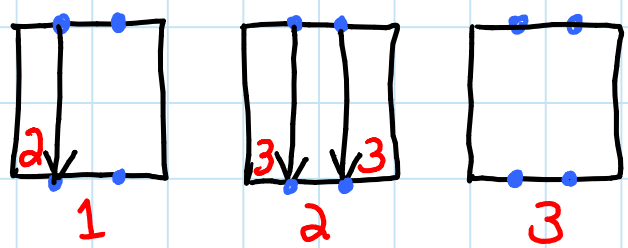
Examples:



2-toggle
[Demaine, Grosof, Lynch – CIAC 2017]



closing door
(directed, single use)



opening door
(directed, single use)

Gadget types:

- k-tunnel = all transitions are along edges of perfect matching on locations (states can control traversability & directions)
- DAG = state-transition graph is acyclic
  ↳ possible transitions on states (merging all locations)

Motion planning = traversal $s \to t$ in graph of
- gadgets (instances from some allowed set)
- connections between gadget locations

Characterization: motion planning with DAG k-tunnel gadgets is NP-complete iff:
- some gadget has distant opening: traversal that opens another tunnel

OR
- some gadget has forced distant closing: traversal that always closes another tunnel
  $s, l_1 \to l_2$ $\quad$ $(s_1, l_1) \to (*, l_2)$

Eventually static: allow loops in terminal states
e.g.: door closing is NP-hard
↳ ≈ Metatheorem 4b of [Viglietta - FUN 2012]
& door opening + diode is NP-hard
↳ ≈ Metatheorem 3 of [Forišek - FUN 2010]
(also need crossover for now)