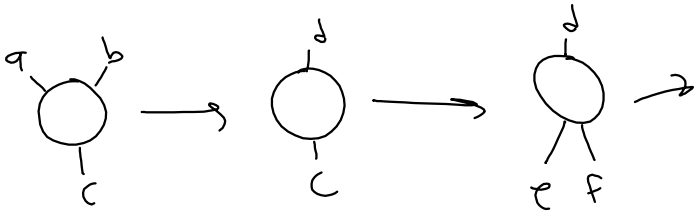
Recall 4 kinds of critical events:

1) appear
2) disappear
3) split
4) merge

Imagine a "~~movie~~ movie", $\tilde{Y}_t$ that shows gaps

A derived I-space that records only the sequence of cyclic orderings of gaps



$((a,b,c),(d,c),(d,e,f),\ldots)$

cyclic

Make an even smaller I-space

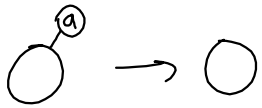
"compress this into a tree"

Consider the 4 critical events:

1) appear:

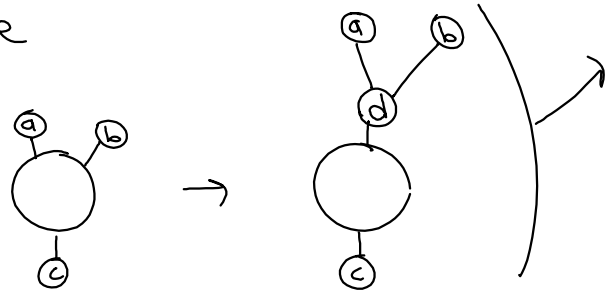
- New node, attached to root
- New label

2) disappear

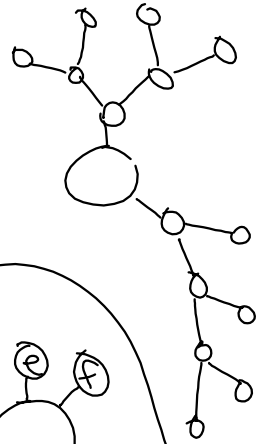


- Delete node

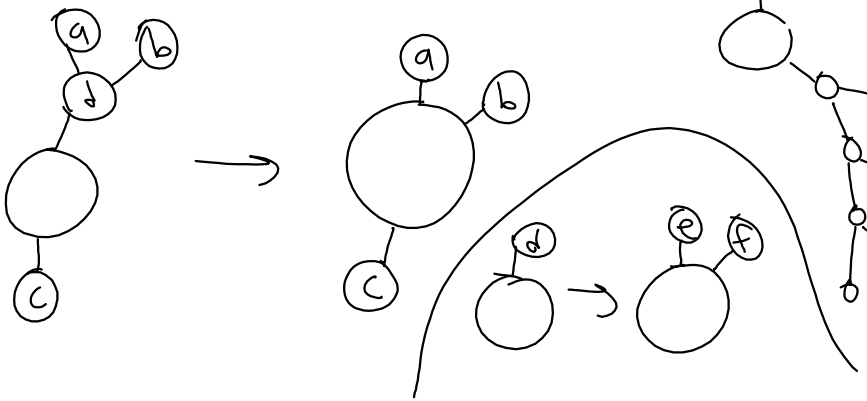
3) merge



Shows the "merge structure"

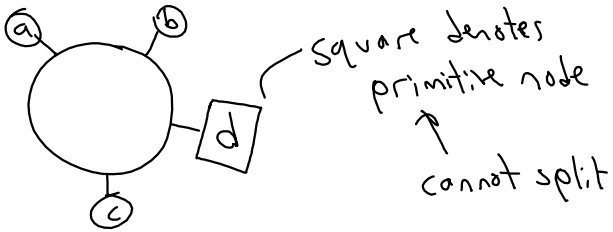


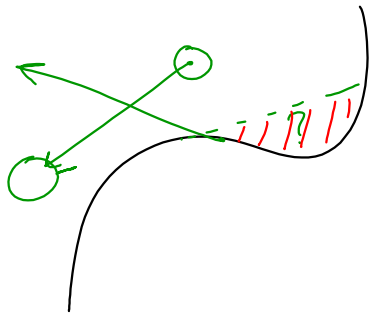
4) split



Consider growing a tree that "represents" the whole environment / configuration

Initially:



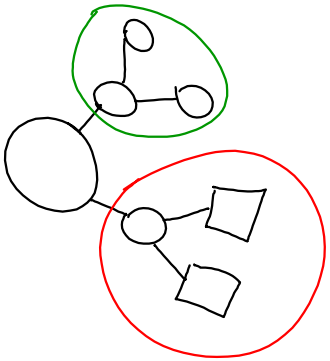


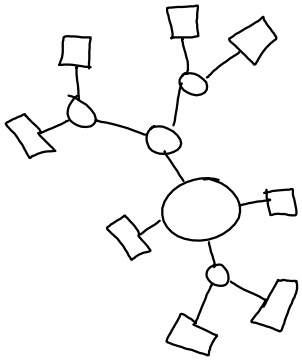
Modify the tree at the critical events

For an appear event, the new node is called <u>primitive</u>

"Learn" the environment/configuration

Keep chasing nonprimitive gaps/nodes
(any branch that has at least
one non-primitive leaf node)





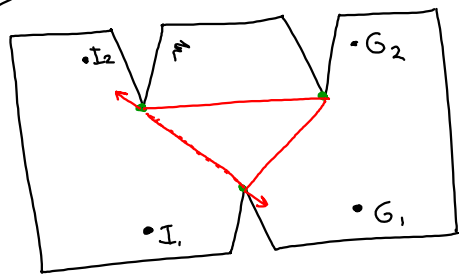
If a split occurs, chase a non-primitive child
If a disappear occurs, chase a different non-primitive node.

If all leaves are primitive, then we are done exploring.

What is it good for? The robot can navigate optimally, in terms of shortest possible Euclidean distance travelled.

$X(\eta) \subseteq X = \mathbb{R}^2 \times S^1 \times E$

tree

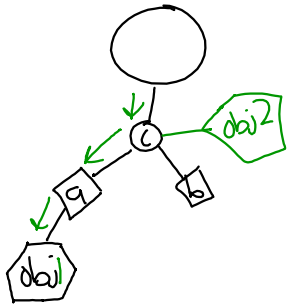


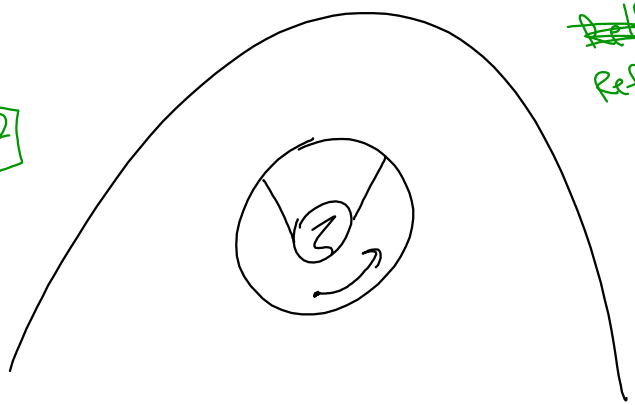
Sec. 6.2.4

Objects in the environment

Chase(object)

- No distances or angles
- Unknown whether e is polygonal or smooth
- Big or Small?





~~Refl~~ Reflex vertices

Tovar, Murrieta, LaValle: Distance Optimal Navigation..... IEEE Trans. 2007
July?, June?

Can we learn about the state with no sensors?

Actuation only

Project Teams?

(Olli, Timo)

Umeå – individual projects: Erik, Ola, Kalle

Markus

Tero

Rejo, Lassi

Nazia