LaVallle 4.3

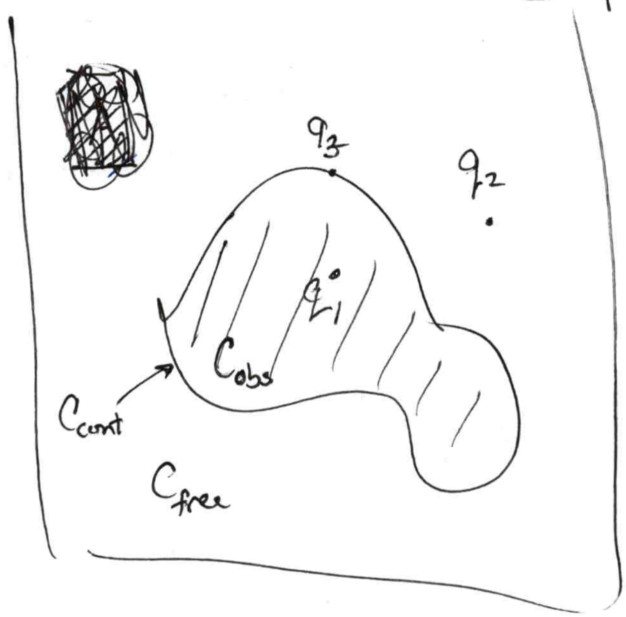
2/11/18

1

4.3.1: Def of Basic MP. Problem

Configuration space

$\mathcal{C}$-space



World

$W = \mathbb{R}^2$ or $\mathbb{R}^3$

Obstacle region

$\mathcal{O} \subset W$

a.k.a. "impermissible" region

Robot

$A \subset W$

Assume A & $\mathcal{O}$ are expressed as semi-algebraic models

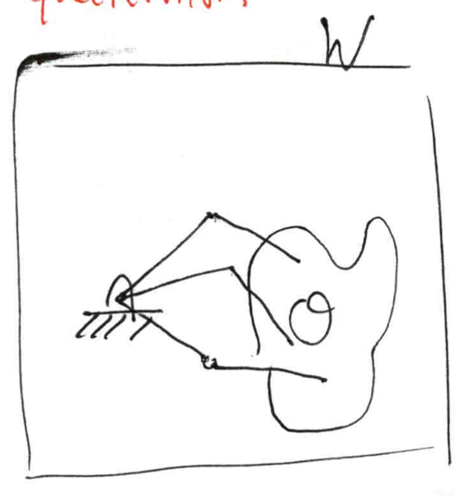
Let $q \in \mathcal{C}$ be a configuration of the robot A.

In $W = \mathbb{R}^2$, $q = (x_t, y_t, \theta)$

In $W = \mathbb{R}^3$, $q = (x_t, y_t, z_t, h)$ ← unit quaternion

The obstacle region mapped into C-space

$$C_{obs} = \{ q \in \mathcal{C} \mid A(q) \cap \mathcal{O} \neq \emptyset \}$$



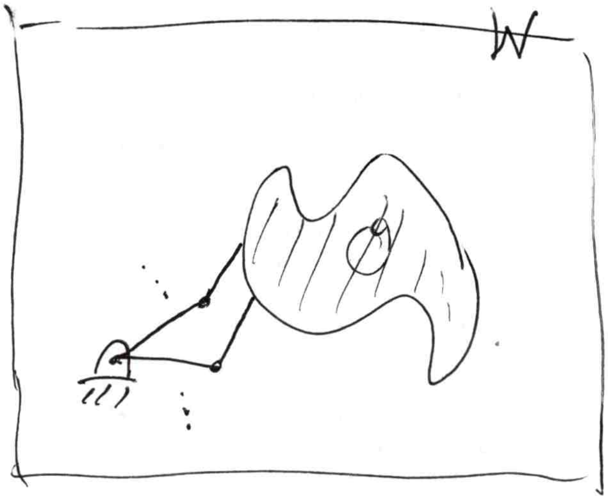
$$C_{free} = \mathcal{C} \setminus C_{obs}$$

$$= \{ q \in \mathcal{C} \mid A(q) \cap \mathcal{O} = \emptyset \}$$

$cl(\mathcal{O}) \cap cl(A(q))$

$$C_{cont} = \{ q \in \mathcal{C} \mid \underline{int}(\mathcal{O}) \cap \underline{int}(A(q)) = \emptyset \text{ AND } \mathcal{O} \cap A(q) \neq \emptyset \}$$

interior of a set



Motion Planning Problem general Formulation 4.1

① Given $W = \mathbb{R}^n$ ($n = 2$ or 3), $\mathcal{O}$, A.

② $\mathcal{C}$, C_{obs}, & C_{free} are implicitly derived

③ Initial & goal configs are given $\underbrace{q_I, q_G}$ $\in C_{free}$

a.k.a. a planning query.

④ A planning alg computes a continuous path $\tau : [0,1] \to C_{free}$ such that $\tau(0) = q_I$ and $\tau(1) = q_G$, OR correctly report that a path does not exist.

1980's or 1990's Reif showed that this problem is PSPACE-hard which implies NP-hard!

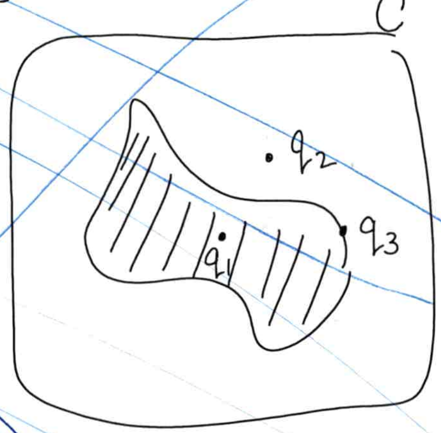
LaValle_Ch4-3

Monday, April 09, 2012
11:55 AM

4.3 The C-space Obstacle, C_{obs}

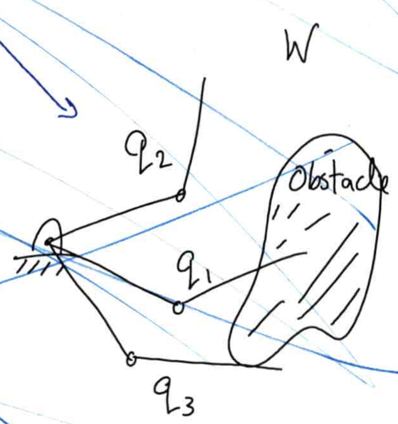
Let $q \in C$ denote a config. of the robot

Let $A(q)$ be the robot in config q.



$$C_{obs} = \{ q \in C \mid A(q) \cap \mathrm{Int}(O) \neq \emptyset \}$$

$$C_{cont} = \{ q \in C \mid A(q) \cap O \neq \emptyset, \; A(q) \cap \mathrm{Int}(O) = \emptyset \}$$



equivalently,

$$C_{cont} = \partial C_{obs}$$

$$C_{free} = C \setminus (C_{obs} \cup C_{cont}) = C \setminus cl(C_{obs})$$

closure of a set

4.3.2 Explicit representations of $C_{cont} = \partial C_{obs}$

(LaValle treats C_{obs} and $C_{obs} \cup C_{cont}$) $= cl(C_{obs})$

Definition: Minkowski difference of two sets in $\mathbb{R}^n$

$$X \ominus Y = \{ x - y \in \mathbb{R}^n \mid x \in X, y \in Y \}$$

Each set has a reference frame ....

Each set has a reference frame embedded in it.

$$X \ominus Y = \{ x - y \in \mathbb{R}^n \mid x \in X,\ y \in Y \}$$

vector subtraction

Definition: Minkowski sum

$$X \oplus Y = \{ x + y \in \mathbb{R}^n \mid x \in X,\ y \in Y \}$$

vector addition

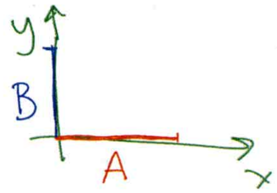
Note: $X \ominus Y = X \oplus (-Y)$ where $-Y$ is the set of negated elements of Y.

Example in $\mathbb{R}^2$

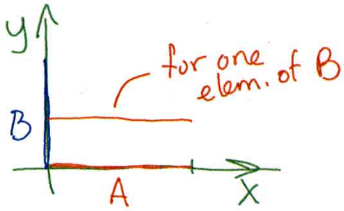
Let $A = I = ([0,1], 0) =$ interval on x-axis

$B = I = (0, [0,1]) =$ interval on y-axis

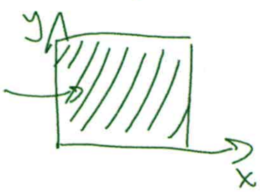
What is $A \oplus B$?



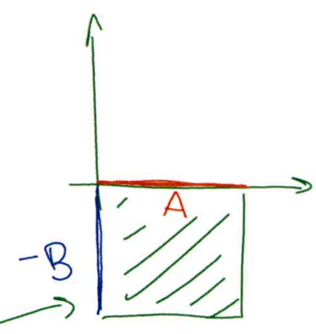
Add every element of A to every element of B

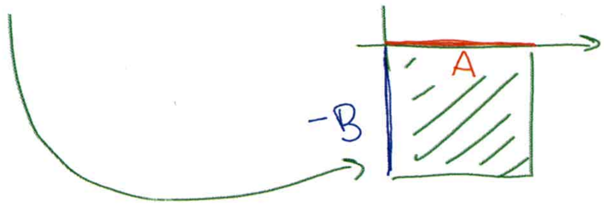


$A \oplus B = I^2$ →



$A \ominus B = ?$ →



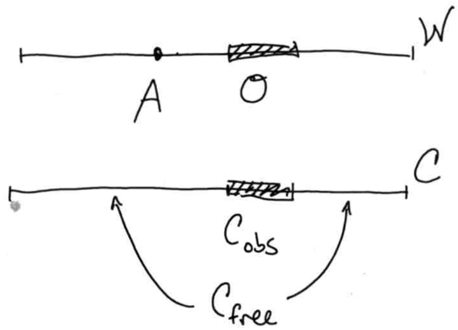


Using $\mathcal{O}$ to determine C_{free} & C_{obs}

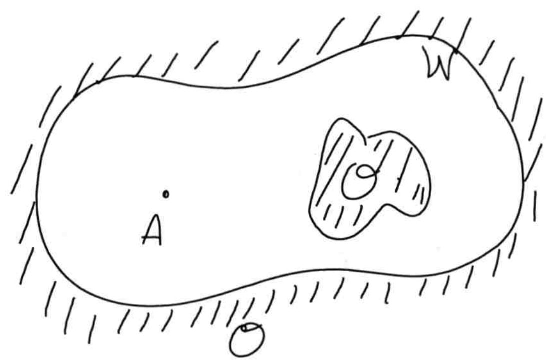
Simplest case: A is a particle.

It can be reasonable to treat robot as a point, if features in World are much larger than robot.

1D world



2D world



Again $C_{obs} = \mathrm{Int}(\mathcal{O})$

$C_{cont} = \partial \mathcal{O}$

$C_{free} = W \setminus \mathrm{cl}(\mathcal{O})$

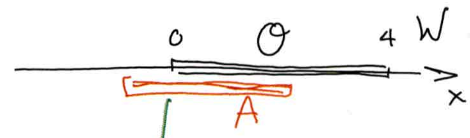
Same for 3D worlds

Robots of finite extent:

1D:

Let $A = [-1, 2]$

$\mathcal{O} = [0, 4]$



To create C_{obs}, C_{cont}, C_{free},
we need a ref. pt. on A.
(Choose the zero point)

C represents possible positions of A by possible positions of the ref. point.

Create the C_{obs} by negating A, then placing the ref. point at every point in $\mathcal{O}$. Perform union of all placed copies of $-A$.

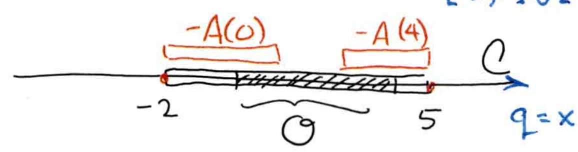
$cl(C_{obs}) = \mathcal{O} \oplus -A = [0,4] \oplus [-2,1] = \underset{\downarrow}{0} \quad \underset{\downarrow}{1} \quad \cdots \quad \underset{\downarrow}{4}$

$[-2,1] \cup [-1,2] \cdot \cup \cdots [2,5]$

$= [-2, 5]$

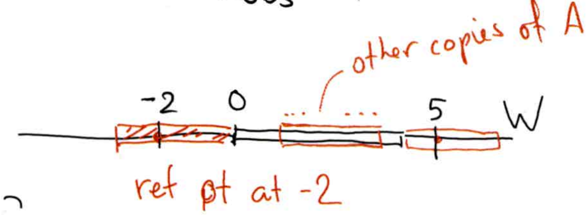
$cl(C_{obs}) = [-2, 5]$

$C_{obs} = \{x \mid -2 \leq x \leq 5\}$



Alternative view: place $A(q)$ at every q where A & $\mathcal{O}$ overlap or touch. The union of the ref pt locations is the C_{obs}.

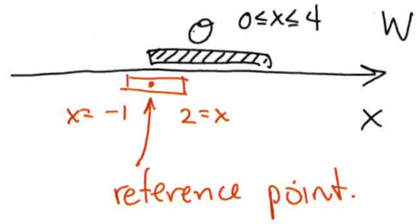
$C_{free} = \{x \mid x < -2 \vee x > 5\}$



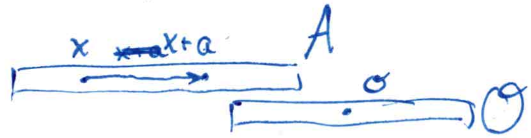
$C_{cont} = \{x \mid x = -2 \lor x = 5\}$ just touching

Why do we define C_{obs} by $\mathcal{O} \oplus (-A)$ instead of $\mathcal{O} \oplus A$?

Consider 1D case:



Write the non-collision constraint for the left side of $\mathcal{O}$:



position of ref pt in W

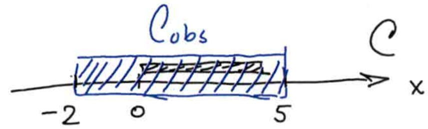
position of pt on robot relative to ref pt.

$$x + a < o, \quad \forall a \in A(x), \quad \forall o \in \mathcal{O}$$

point on obstacle in W

$$\Rightarrow x < \underbrace{o - a, \quad \forall a \in A(x), \quad \forall o \in \mathcal{O}}_{\text{This is Definition of Minkowski difference}}$$

Worst case: $x < -2$

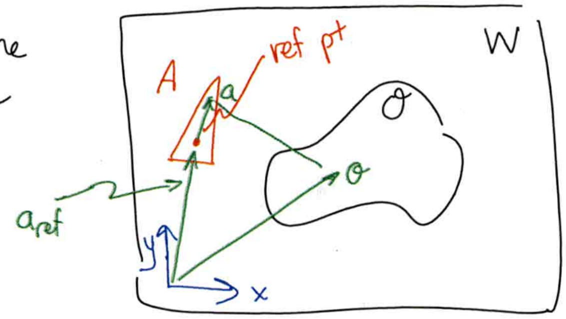


You can apply on right also

$$x > o - a, \quad \forall a \in A, \quad \forall o \in \mathcal{O}$$

Worst case: $x > 5$

2D example works the same way, but constraints are more complicated



pt on robot, v

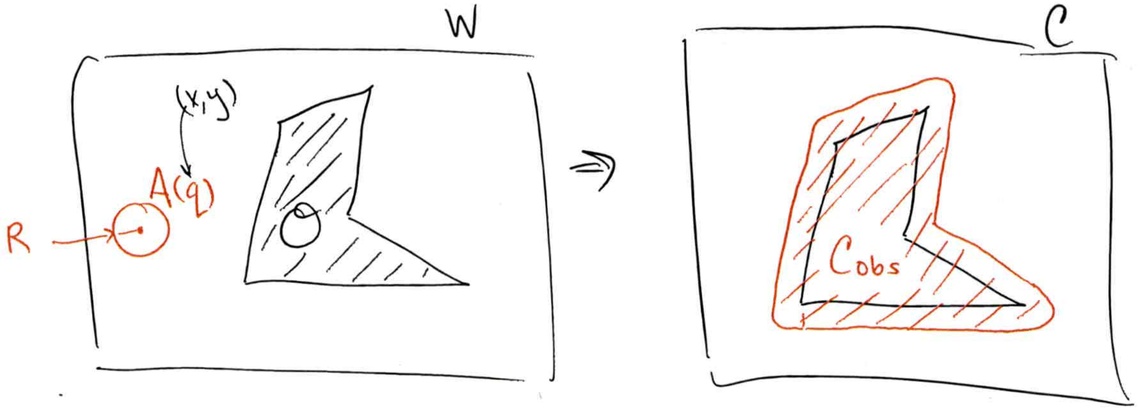
$v = \underset{(x,y)}{a_{ref}} + a$, $a \in A(q)$

$q \in C_{obs}$ if $a_{ref} + a = o$ for some $o \in \mathcal{O}$
for some $a \in A$

$\Downarrow$

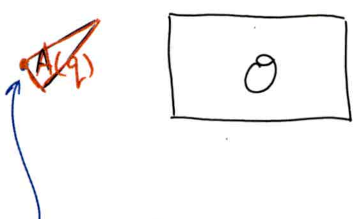
$q \notin C_{obs}$ if $a_{ref} \neq \underbrace{o - a \,, \forall a \in A(q), \forall o \in \mathcal{O}}_{\text{Minkowski difference}}$

Disc robot ← Minkowski sum = Minkowski difference



Planar Translation Only Robot.

C_{free} is set of positions of ref. point $\ni A \cap \mathcal{O} = \emptyset$

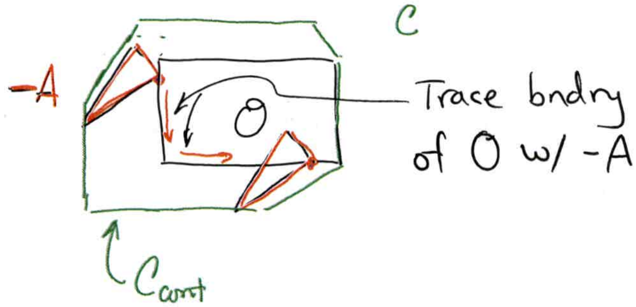


Choose ref. pt.

Choose ref. pt.

$cl(C_{obs}) = \mathcal{O} \oplus -A$

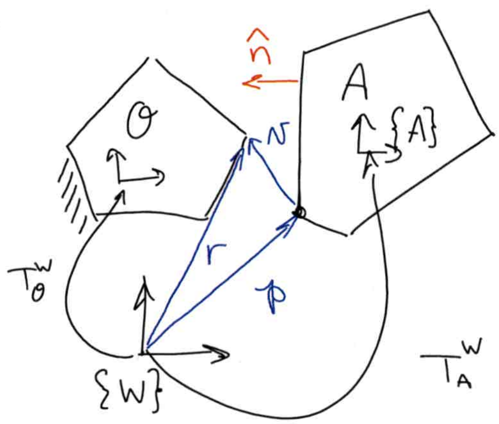
Add $-A$ to every point in $\mathcal{O}$



Skip to next page - star alg

David Hsu's slides have a few nice pictures & animations.

Formulas for the boundary of C_{obs} for planar world with $\mathcal{O}$ & A defined as polygons



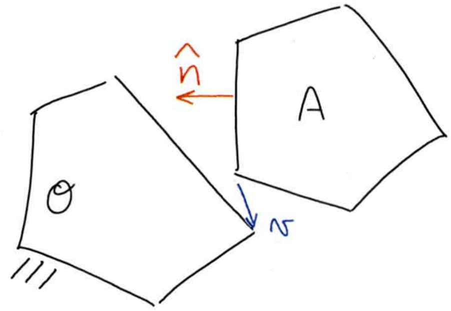
Show Brost Figure

EV Contact - edge of robot in contact with vertex of $\mathcal{O}$.

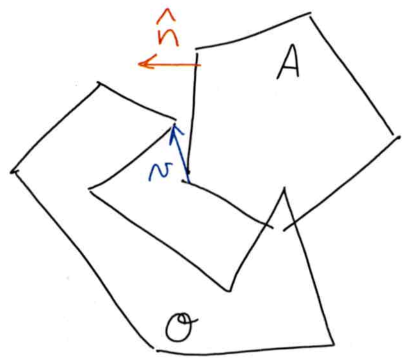
$\hat{n} \cdot v = 0$	potential contact
$\hat{n} \cdot v < 0$	potential penetration
$\hat{n} \cdot v > 0$	local separation

$v = r - p$

Why the qualifiers?



$\hat{n} \cdot v \leq 0$,
but no contact or penetration

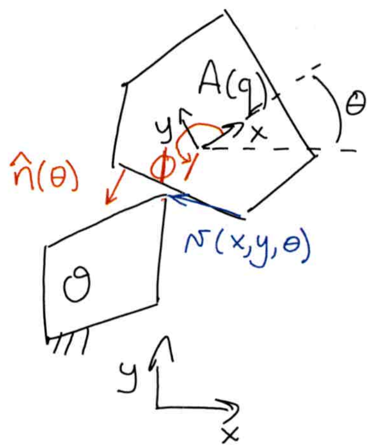


$\hat{n} \cdot v \geq 0$, but penetration occurs (non-locally).

To obtain manifold of a given facet, write contact condition as a function of $q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$

Necessary condition for contact:

$$\hat{n}(\theta) \cdot v(x,y,\theta) = 0$$



Let body-fixed frame be $\{A\}$

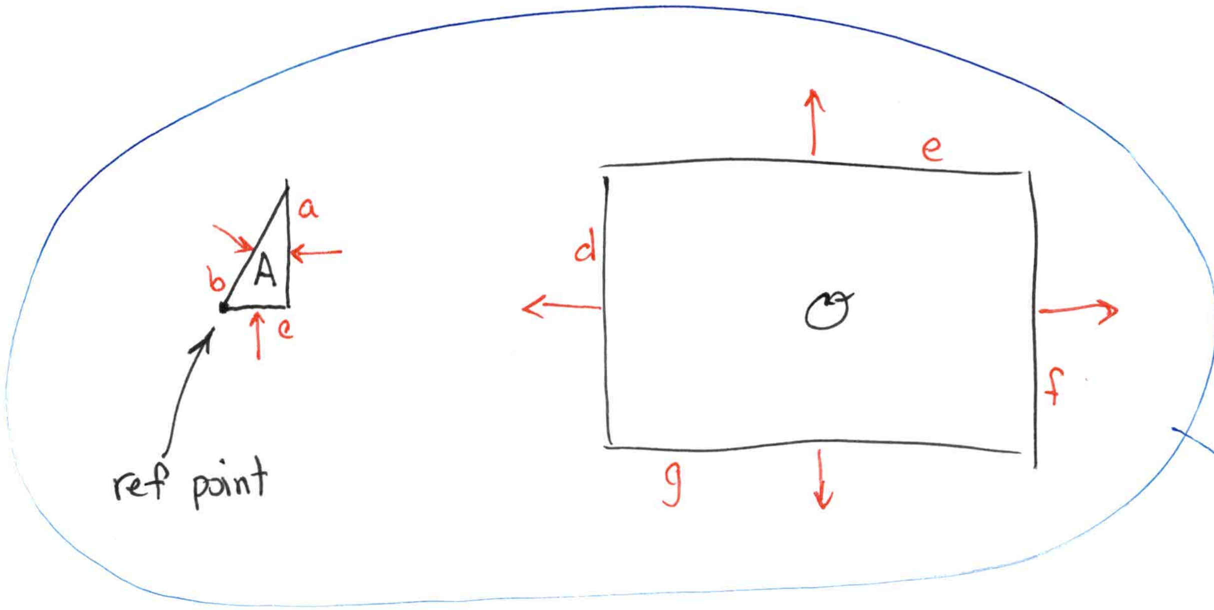
Let ϕ denote the angle from the body-fixed x-axis to the normal direction

$$\hat{n}(\theta) = \begin{bmatrix} \cos(\phi+\theta) \\ \sin(\phi+\theta) \end{bmatrix}$$

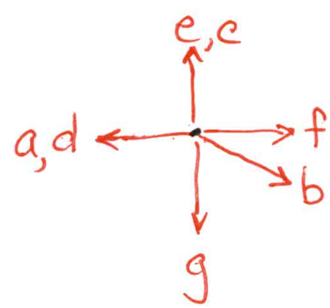
position of p in

2/11/18
③

Star Alg for making C-space obstacles in C-spaces with polygonal bodies w/o rotation.



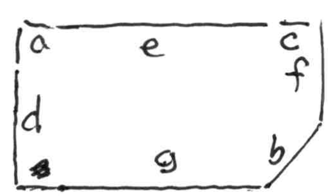
The C_{obs} can be constructed by laying down the edges of A & O in order of face normals



Easiest w/ ref pt on bndry

① Translate A so ref pt touches edge of O w/o overlap. Keep that edge in C_{obs}.

② Walk list of sorted normals placing that edge onto current end of C_{obs}



$$v^W = r^W - p^W = r^W - T_A^W \, p^A$$

← position of p in frame A

$${}^W T_A = \begin{bmatrix} c_\theta & -s_\theta & x \\ s_\theta & c_\theta & y \\ 0 & 0 & 1 \end{bmatrix} \in SE(2)$$

$$v(x,y,\theta) = \begin{bmatrix} \underbrace{r_x}_{\text{position of obstacle vertex in W}} - c_\theta p_x^A + s_\theta p_y^A - \underbrace{x}_{\text{position of origin of robot frame in W}} \\ r_y - s_\theta p_x^A - c_\theta p_y^A - y \end{bmatrix}$$

Substitute into $\hat{n}(\theta) \cdot v(x,y,\theta) = 0$

$$0 = \cos(\theta+\phi)[r_x - c_\theta p_x^A + s_\theta p_y^A - x] + \sin(\theta+\phi)[r_y - s_\theta p_x^A - c_\theta p_y^A - y]$$

(This eq. defines a 2D "variety" in 3D C-space (x,y,θ)).

Nonlinear trigonometric polynomial in θ
(e.g., $\cos(\theta+\phi) \cdot \cos(\theta)$)

Linear in (x,y) ..... This is why C_{obs} has linear surfaces for fixed orientations of A.

$$\boxed{ax + by + c = 0}$$

where $a = -\cos(\theta+\phi)$, $b = -\sin(\theta+\phi)$,

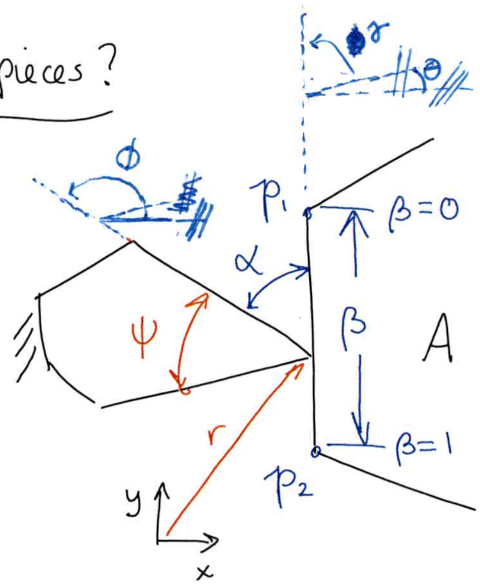
and $c = \cos(\theta+\phi)[r_x - c_\theta p_x^A + s_\theta p_y^A]$
$\quad + \sin(\theta+\phi)[r_y - s_\theta p_x^A - c_\theta p_y^A]$

Setting this contact eq. = 0 yields a 2D manifold in a 3D space $(\mathbb{R}^2 \times S^1) = SE(2)$

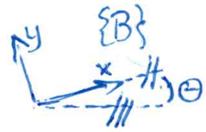
Pieces of this manifold appear in Brost's Cobs

How do we get the pieces?

Another useful formulation ...



$\theta+\gamma+\alpha=\phi$



$0 \leq \alpha \leq \pi-\psi$

$0 \leq \beta \leq 1$

r must be on the line segment $\overline{p_1 p_2}$

$$\begin{bmatrix} p_{ix}^W \\ p_{iy}^W \\ 1 \end{bmatrix} = \begin{bmatrix} c_\theta & -s_\theta & x \\ s_\theta & c_\theta & y \\ 0 & 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} p_{ix}^A \\ p_{iy}^A \\ 1 \end{bmatrix}}_{\text{given in geom. model}}$$

Substitute p_1^W, p_2^W into:

$i = 1, 2$

$$\boxed{\begin{array}{l} (1-\beta)p_1^W(x,y,\theta) + \beta p_2^W(x,y,\theta) = r \\ 0 \leq \beta \leq 1 \\ 0 \leq \alpha \leq \pi - \psi \end{array}}$$

$\leftarrow \alpha = \phi - \gamma - \theta$

Choose α, β and solve for x, y, θ to get a point on C_{obs}

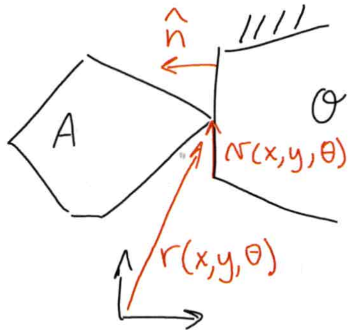
If A & $\mathcal{O}$ are convex, then this is a patch of Cobs

If A or $\mathcal{O}$ are nonconvex, then some of this patch may be cut away by nonlocal interpenetrations!

Other contact types.

VE contact :

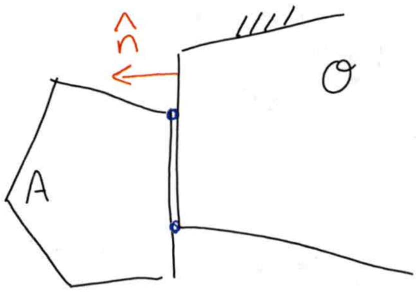
Same procedure, but
$\hat{n}$ is constant
v is $v(x,y,\theta)$
r is $r(x,y,\theta)$



VE $\Rightarrow$ 2D facet of C_{obs}

EE contact :

Treat as two contacts of type EV or VE

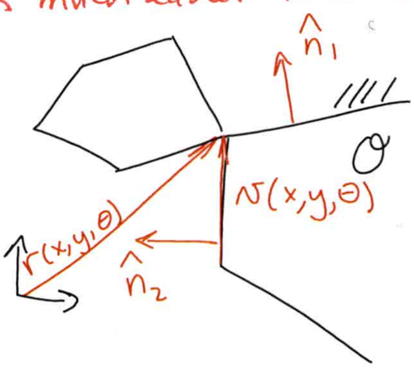


EE $\Rightarrow$ 1D facet of C_{obs}

However forming equations w/o rotation variable θ makes the equation much easier to solve.

VV contact :

Treat as two contacts of type EV or VE



VV contact eq gives the points on a curved facet on C_{obs}; where EV & VE facets meet

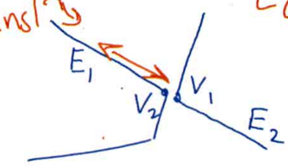
VV $\Rightarrow$ 1D facet of C_{obs}

You can also have (EV, EV) or (VE, VE) or (EV, VE)

You can choose (EV, EV) or (VE, VE) or ~~(EV, VE)~~

doesn't always work

1 DoF transl allowed



Simple example of a C_{obs} in SE(2).

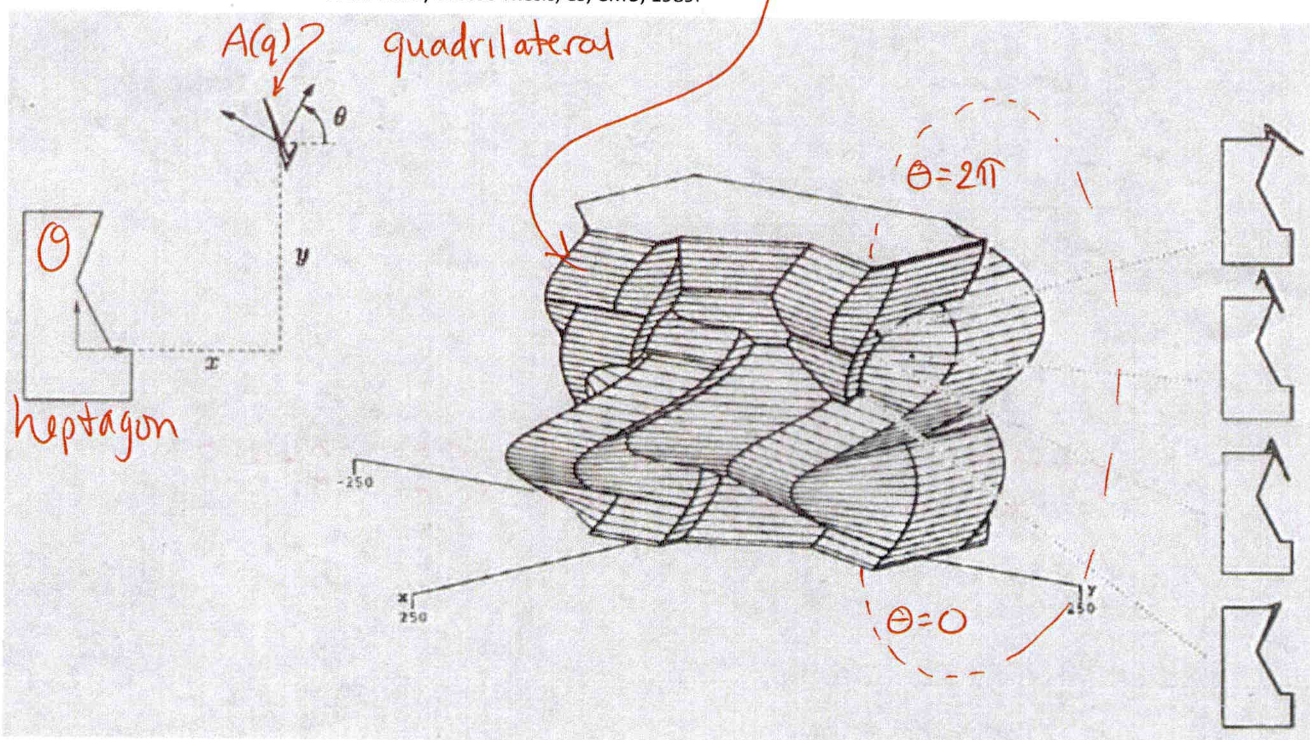
How many facets are there?

At most:

$|E_1| = 4 = |V_1| \Rightarrow 28\,(E,V)$ pairs

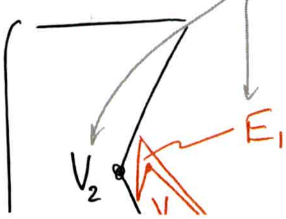
$|E_2| = 7 = |V_2| \Rightarrow 28\,(V,E)$ pairs

From Randy Brost's Thesis, CS, CMU, 1989.

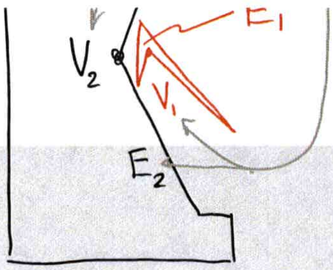


One infeasible (E,V) pair. There are 8 total.

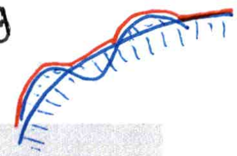
There are 7 (E,V) infeasible (E,V) pairs.

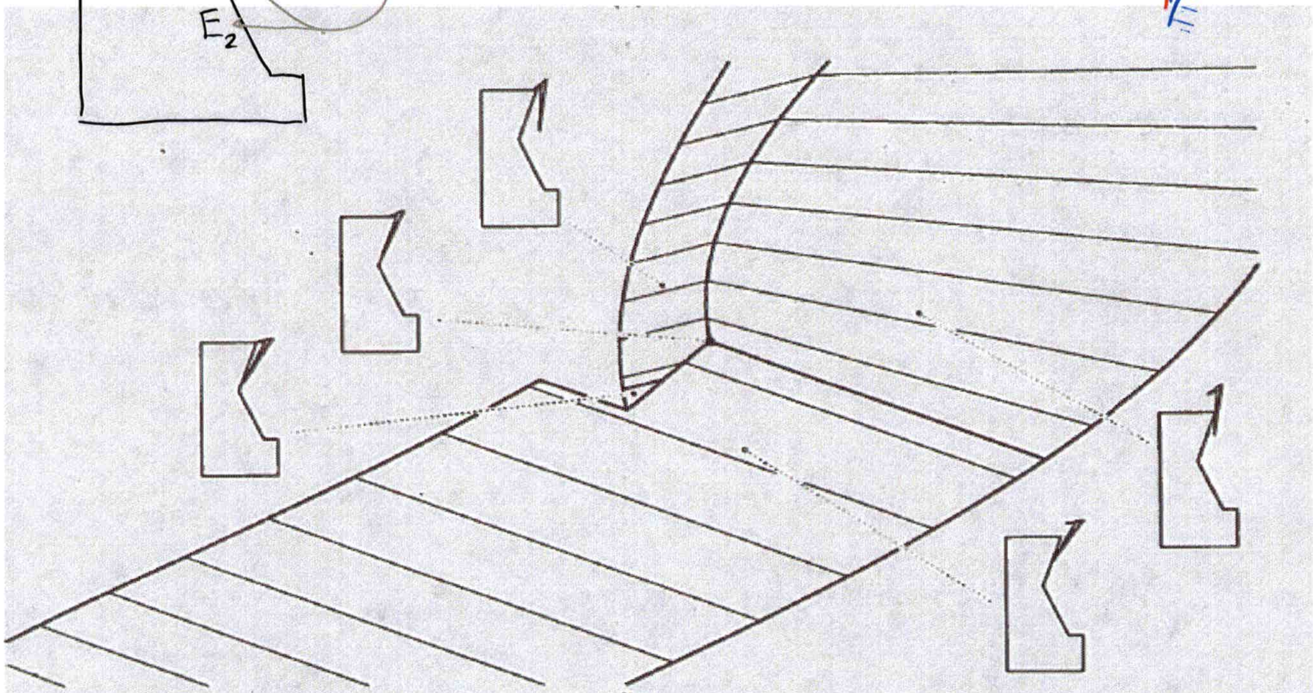


$\therefore C_{obs}$ has at most $56 - \cancel{15}\,13 = 43$



$\therefore$ C_{obs} has at most $56 - 13 = 43$ 2D facets?





(b)

Figure 6: (a) The (x, y, θ) configuration-space obstacle of the two polygonal objects. (b) An enlarged view of the obstacle features required for the mechanics analysis. The vertex corresponding to the goal configuration is labelled in both (a) and (b).

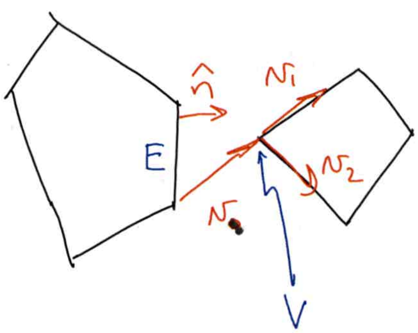
It's possible that some facets could appear as multiple surface patches, since portions could go under other facets.

Collision Checking for Polygons

Assume polygons are convex
(can partition non-convex ones)

Theorem:
A pair of convex polygons are separated iff $\exists$ a vertex/edge pair $(E, V) \ni$

$\hat{n} \cdot N > 0$



$\hat{n} \cdot v_1 > 0$

$\hat{n} \cdot v_2 > 0$

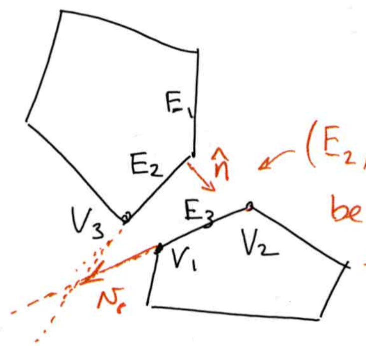
Example: ⟶

(E_1, V_1) fails test

(E_2, V_2) fails

(E_3, V_3) succeeds

(E_2, V_1) succeeds too



(E_2, V_2) fails because v points toward line supporting E_2.

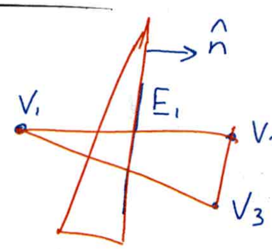
Collision Detection for Convex Polygons

```
q_in_Cfree = false
while q_in_Cfree = false
    if (n̂ · v1 > 0) AND (n̂ · v2 > 0) AND (n̂ · v) > 0
        q_in_Cfree = true
        break
    end if
end while
```

loop over all EV & VE pairs

Does this work for this case?

$V_j E_i$, $\left.\begin{matrix} i = 1,2,3 \\ j = 1,2,3 \end{matrix}\right\}$ satisfied



$V_j E_i$, $\left.\begin{matrix} i=4,5,6 \\ j=4,5,6 \end{matrix}\right\}$ satisfied Yes! It works!

Complexity:

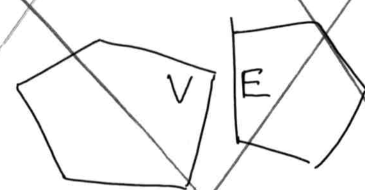
Let $|V_A|$ & $|V_O|$ be the numbers of vertices of the robot, A, and the obstacle, O.

For each vertex-edge pair there are 3 inequalities to test $\Longrightarrow O(1)$

There are $O(|V_A| \cdot |V_O|)$ EV and VE pairs.

$\therefore$ Collision detection is $O(|V_A| \cdot |V_O|)$

Lavalle's discussion in section 4.3.3 focuses on the negation of the collision-free conditions:

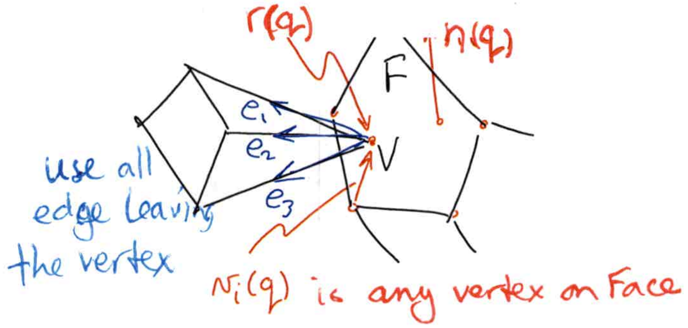
Consider 1 EV pair $\longrightarrow$ 

Let $H_A = H_1 \cup H_2 \cup H_3$

where $H_1 = \{q \in C \mid \hat{n} \cdot v_1 \leq 0\}$ } negations of previously

What about 3D?

Extension is analogous but a bit more complicated.



$q \in C_{free}$ if

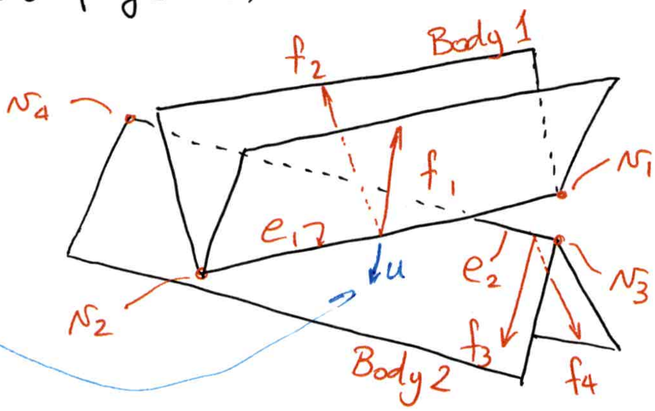
$\exists$ a V-F or F-V pair $\ni$

$$\hat{n} \cdot v > 0$$
$$\hat{n} \cdot e_i > 0 \quad i = 1, \dots, \#\text{ edges}$$

A facet of C_{obs} would be defined by $\hat{n} \cdot v = 0$ and $r(q)$ within the boundaries of face F.

In 3D we have another case, E-E.

(this extends beyond Lavalle page 165)



Let $u = e_1 \times e_2$

$\hat{u} = u / \|u\|$

(outward wrt Body 1)

$(v_3 - v_1) \cdot \hat{u} > 0 \Rightarrow e_1$ & e_2 are separated

f_1, f_2 are vectors in faces of Body 1 & $\perp$ to e_1

$f_1 \cdot \hat{u} \leq 0 \Rightarrow$ no overlap

$f_2 \cdot \hat{u} \leq 0 \Rightarrow$ no overlap

f_3, f_4 are vectors in faces of B2 & $\perp e_2$

$f_3 \cdot \hat{u} > 0 \Rightarrow$ no overlap

$f_4 \cdot \hat{u} > 0 \Rightarrow$ no overlap

3D Collision Checking

q_in_Cfree = false

loop over all FV VF EE pairs

```
while q_in_Cfree = false
    if (separation condition satisfied)
        q_in_Cfree = true
        break
    end if
end while
```

Complexity of CD in 3D.

[Latombe '91] It is known that the # of VF, FV, and EE pairs is $O(|V_A| \cdot |V_o|)$.

- Fast collision checking - intuition

If relative config does not change much, and polygons were not in collision at previous check, it should be "easy" to find the definitive witness pair.

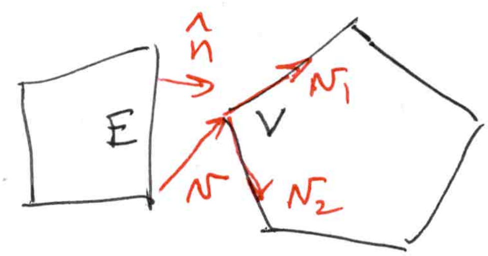
Section 4.3.3: Explicitly Modeling C_{obs}

2/12/18

Summary: Use primitives based on negation of the collision-free condition mentioned above.

Consider 2 convex polygons

Let $H_A = H_1 \cup H_2 \cup H_3$



where

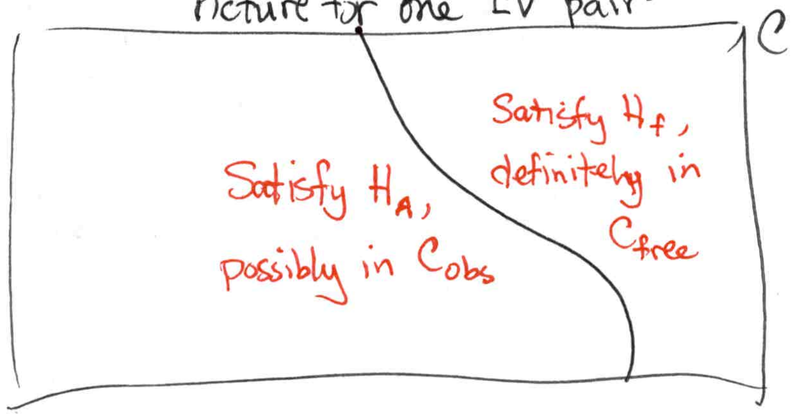
$$\left.\begin{aligned} H_1 &= \{q \in C \mid \hat{n} \cdot v_1 \leq 0\} \\ H_2 &= \{q \in C \mid \hat{n} \cdot v_2 \leq 0\} \\ H_3 &= \{q \in C \mid \hat{n} \cdot v < 0\} \end{aligned}\right\}$$

negations of previously defined collision-free conditions

$$C_{free} = \{q \in C \mid \text{for some EV or VE pair, } \overbrace{(\hat{n} \cdot v_1 \geq 0 \land \hat{n} \cdot v_2 \geq 0 \land \hat{n} \cdot v > 0)}^{H_f}\}$$

$$C_{obs} = \{q \in C \mid \text{for all pairs EV \& VE, } \overbrace{(\hat{n} \cdot v_1 < 0 \land \hat{n} \cdot v_2 < 0 \land \hat{n} \cdot v \leq 0)}^{H_A}\}$$

Picture for one EV pair



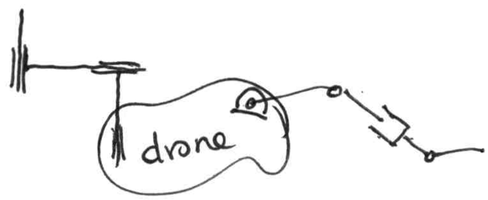
More EV, VE pairs cause more cuts to approximate C_{obs} until all are done.

LaValle Ch4.4 : Closed Kinematic Chains

2/11/18

The C-space of open chains is just the product of C-spaces of each joint

e.g. $\underbrace{\mathbb{R}^4}_{\text{four prismatic joints}} \times \underbrace{S^1 \times S^1}_{\text{2 revolute joints}} \times \underbrace{SE(3)}_{\text{drone}}$



The constraints associated with closing ~~the~~ a loop with kinematic chains, fundamentally changes the structure of C-space.

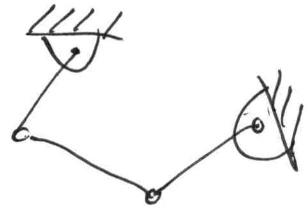
3R manip

$C = S \times S \times S$

If joint limits exist,

$C = I \times I \times I = I^3$

Now close $\xrightarrow{\hspace{2em}}$ the loop



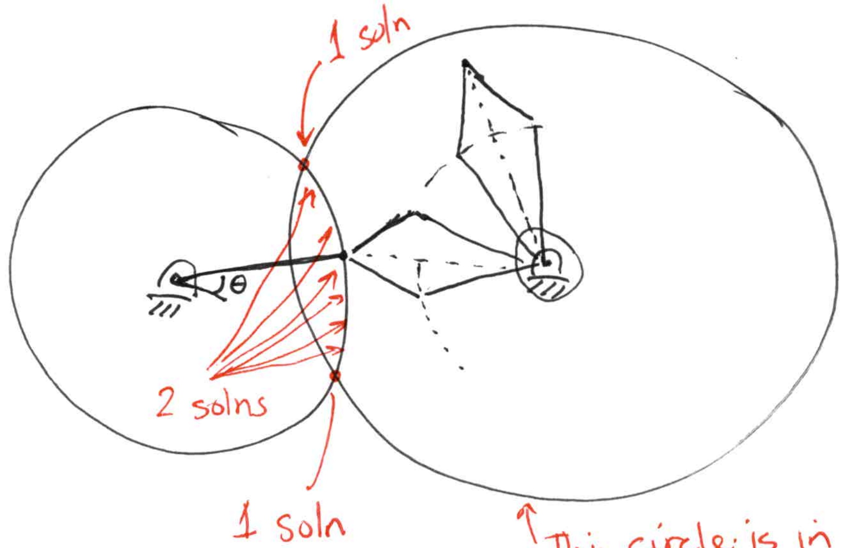
What's a good parameterization?

Now C-space is one-dimensional, not 3D.

2/11/18

⑤

Global structure of C-space for a planar four-bar.



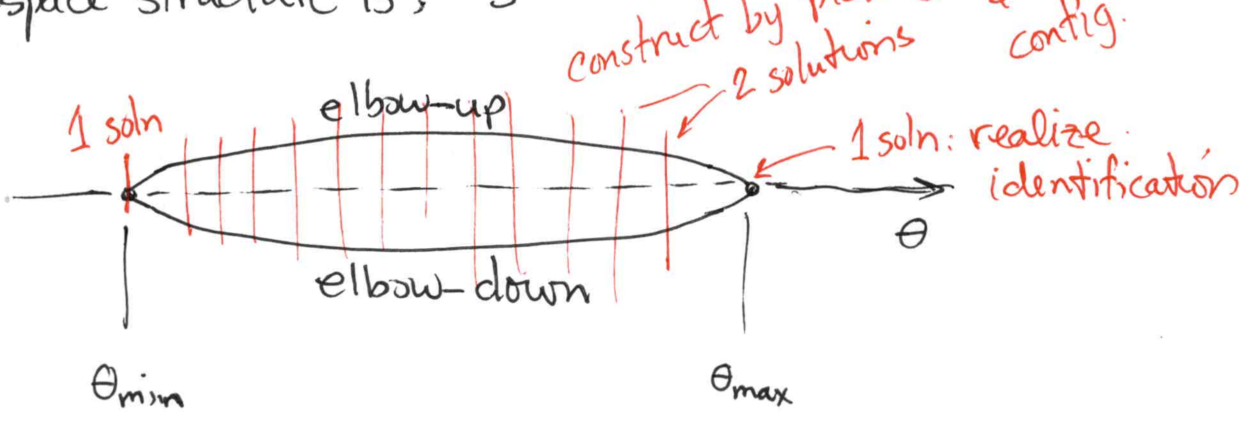
This circle is in some sense critical to C-space structure.

As θ changes the 2R on the right always has 2 solns; elbow-up and elbow-down.

Until it gets to an extreme position of its range of motion. Then elbow-up soln ≡ elbow-down solution.

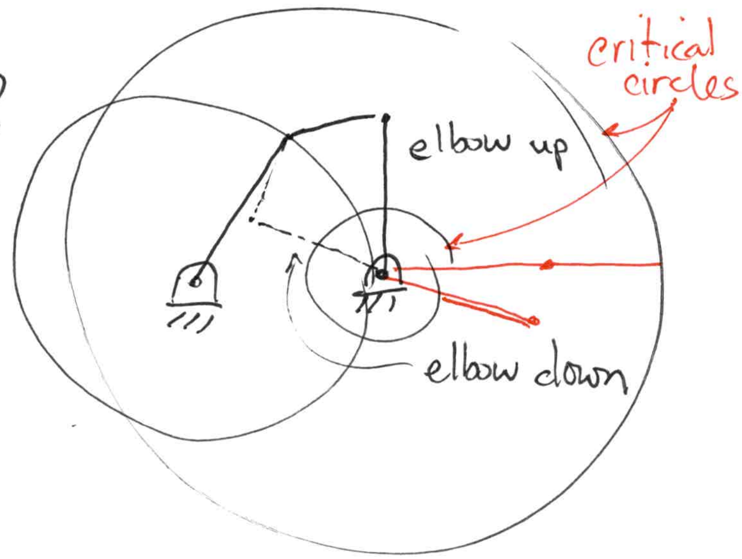
So C-space structure is: S^1

construct by picking θ values & computing config.



This knowledge allows your algorithms to generate motions that switch from elbow-down to elbow-up and vice versa as needed.

What's the C-space?

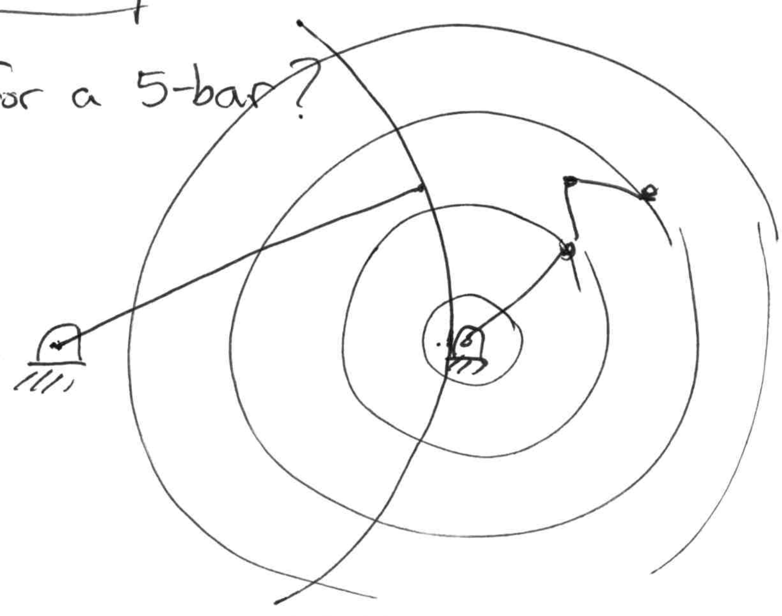


Two disconnected circles

$C = S^1 \sqcup S^1$

$\boxed{\text{Show slide 18 from ClosedChains.pptx}}$

What is C-space for a 5-bar?



In this case, you get a triple torus; 2D surface with 3 holes.