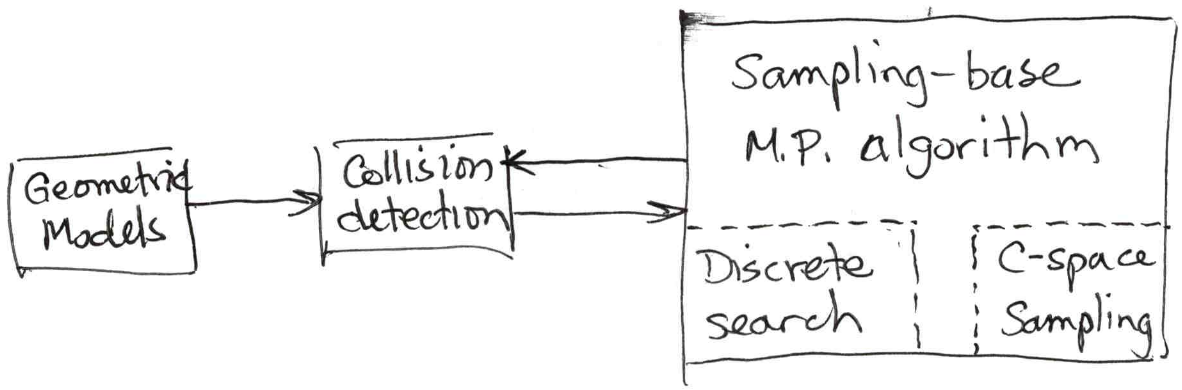
LaValle Ch5: Sample-Based Motion Planning 2/19/18

Two main philosophies:

- Sample-based (Ch5) - avoid constructing C_{obs}
- Exact (Ch6) - (a.k.a. Combinatorial) construct C_{obs}

Sampling-based M.P. characteristics

- Probe free space, C_{free}, w/ {random or grid} points
- Use collision detection as a black box
  - ∴ sampling-based M.P. algs are indep. of the types of geometric models used.
- Not complete! - more on this coming up.

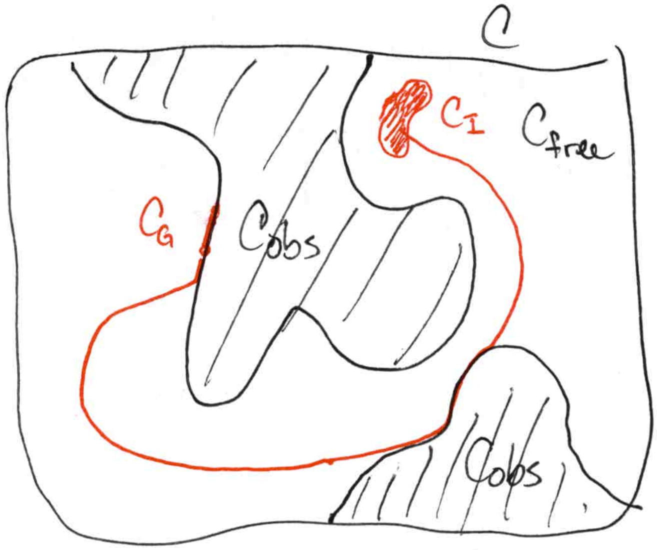


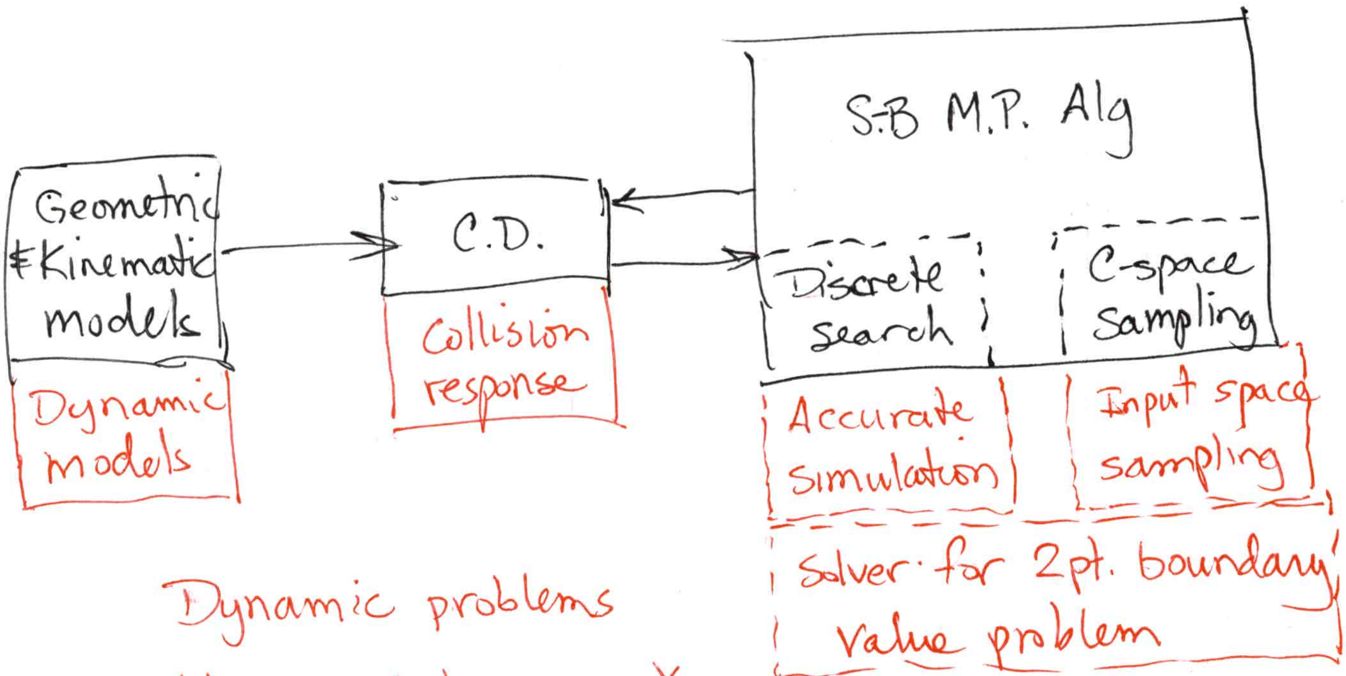
2/19/18

②

Geometric problems "live" in C-space

Some quasi-static problems also live in C-space



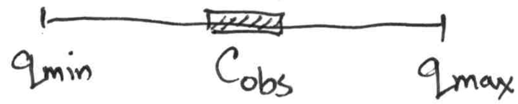


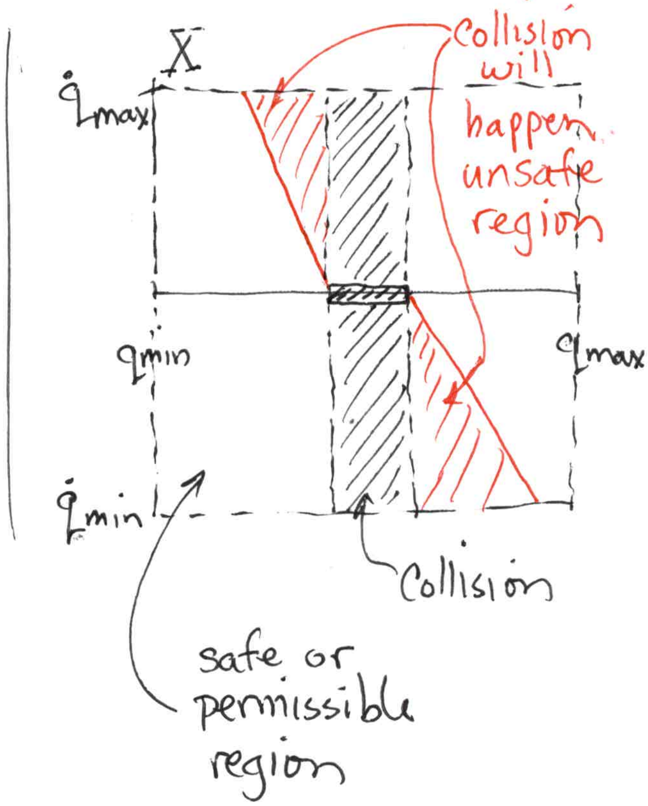
Dynamic problems live in state space X and require additional machinery, where $X = C \times V$ and V = space of velocities.

When planning for dynamic system, collision detection is not enough.

1D cspace, C

bead on a wire
car on a track





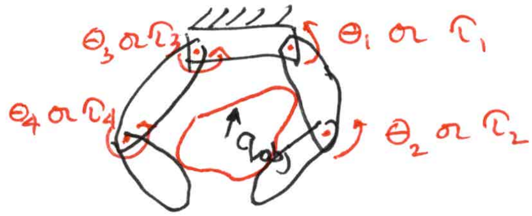
System Type	Space	Input Types	Validity Check
Geometric & Kinematic	C	Path segments in C. Assume controller can track any path	Collision detection
Dynamic (with kinematics and geometry) a.k.a. Kinodynamic	X	Path segments in X (if system is "small-time locally controllable") Path segments in actuator output space (forces & torques)	Simulation with collision detection. Check force/torque ~~control~~ limits too

2/19/18

4

Example of search in actuator space.

Control θ & $\dot{\theta}$ and use simulation to estimate q_{obj}, $\dot{q}_{obj}$, since they can't be controlled directly.



This is how the in-hand polygon manip. problem was solved!

Basic Algorithm for C-space

Let $G(E,V)$ be a topological graph with E and V being edge and vertex sets

1. Initialize $G(E,V)$
   $E = \{\}$, $V = q_I, q_{G_1}, q_{G_2}, \ldots$
2. Use VSM (vertex selection method)
   Choose a pair of vertices to try to connect — distance metric is desirable
3. Use LPM (local planning method)
   Solve 2-pt boundary problem ← edge could be directed
4. If LPM successful, insert edge into G
5. Search G for solution. If found return success
6. Iterate until soln found or termination condition met.

Two classes of application affect sampling: single-query and multiple-query

<u>Single-query</u> (solve one q_I, q_G pair)

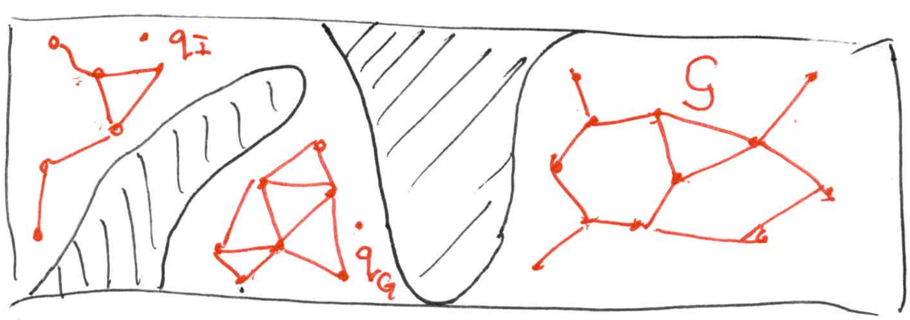
- choose points that are easy to connect
- choose pairs of pts to connect so that $\mathcal{G}$ grows quickly to absorb q_G.

depth-first mindset

<u>Multiple-query</u> (solve many q_I, q_G pairs)

- choose pts easy to connect
- choose pairs of pts, so that $\mathcal{G}$ covers $\mathcal{C}$ (or X) well
  - $\mathcal{G}$ and $\mathcal{C}$ (or X) have same # of connected components
  - Any point (e.g. from new query) can be connected to $\mathcal{G}$ quickly.

$\mathcal{G}$ has 3 connected components



$\mathcal{C}$ (or X)

Two connected components.

2/19/18

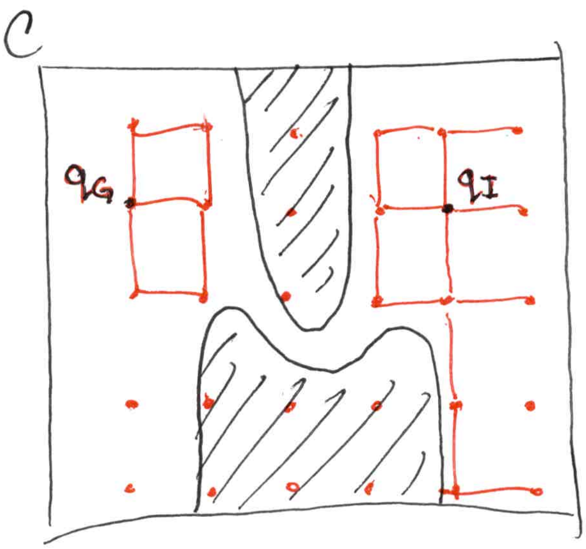
Algorithmic completeness

If soln exists, one will be found in finite time

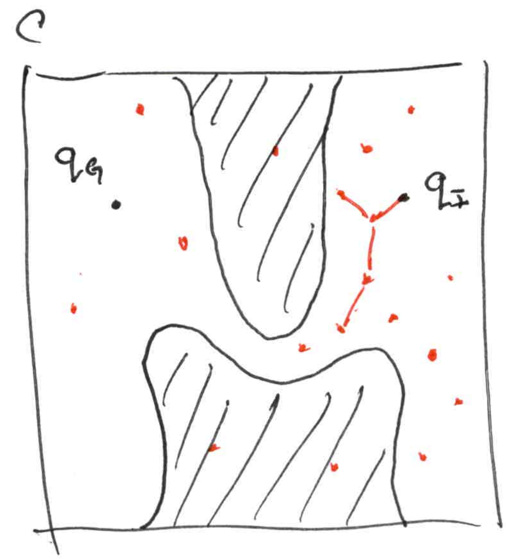
If soln ¬exists, that will be reported in finite time.

Sample-based M.P. algs are <u>not</u> complete!

With "good" sample selection sequences, algs. can be "<u>resolution complete</u>" or "<u>probabilistically complete</u>".



Need to increase resolution to find path.



Given enough points, eventually a path will be found.

In both cases, if a path exists one will be found as # of samples goes to ∞.

2/19/18
7

To guarantee resolution or probabilistic completeness, the sequence of sample points must be "dense" in C (or X).

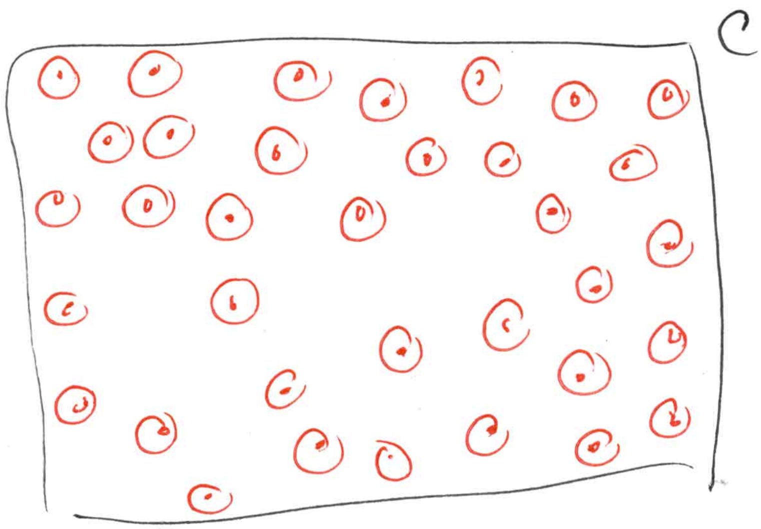
A set U of pts is dense in space A if $cl(U) = A$

Trivial example $(0,1)$ is dense in $[0,1]$

Example: An infinite sequence of points is dense in C if the sequence contains a point arbitrarily close to each point in C.

1.) Choose pts of sequence (deterministic or random)

2.) Center a circle on each point.



3.) The union of the circles must, in the limit, as # samples $\to \infty$ equal C even as radius goes to zero.

~~But any 1-D distrib covering possible values of q will work.~~

~~Random~~ Uniform random sampling of I^1 & S^1 is provably dense. $\therefore$ we can solve problems for planar robots $(X = (I^2 \times S^1)^n)$ ← because denseness in $I^1 \rightarrow$ denseness in $I^1 \times I^1$, $I^1 \times I^1 \times S^1$, ...

Fortunately $SO(3)$ can be sampled densely as follows:

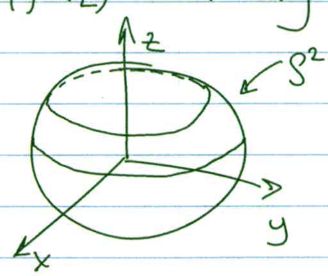
Choose (u_1, u_2, u_3) uniformly at random on $[0,1]$.

$$h = \left(\sqrt{1-u_1}\,\sin(|2\pi u_2|),\ \sqrt{1-u_1}\,\cos(2\pi u_2),\ \ldots\right.$$

$$\left.\ldots\ \sqrt{u_1}\,\sin(2\pi u_3),\ \sqrt{u_1}\,\cos(2\pi u_3)\right)$$

The random quaternions formed this way are "uniformly" distributed on $SO(3)$.

This method is similar to choosing (u_1, u_2) uniformly at random on $[0,\pi]$ & $[0, 2\pi]$.



This gives a dense sequence on S^2, but it is not uniform, since points are denser near the poles.

$$\boxed{\text{Uniform random sampling of 3-param reps of orientation do } \underline{\text{not}} \text{ give dense samples on } SO(3)!}$$

So now we can sample uniformly at random in all relevant spaces for problems discussed so far.

We still cannot sample on "varieties"

so far.

We still cannot sample on "varieties" defining valid configs of closed chains and contact spaces of arbitrary systems.

There is one important implied assumption in res. & prob. completeness of sample-based algs:

if 2 pts are close enough, if a solution to the 2pt. b.v.p. exists, it can be found in finite time!

Since 1995 there have been research papers written on how to best choose points to solve probs. with narrow passages.

Note: all assembly/dissembly problems have narrow passages.

Section 5.1 – Distance & Volume

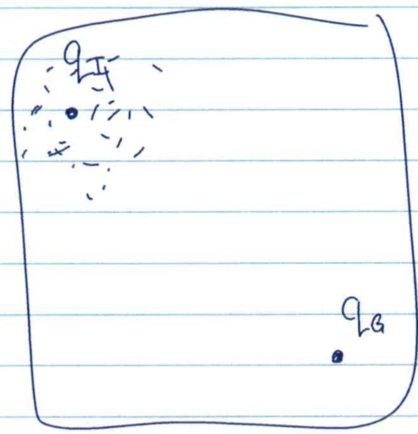
To make uniform sampling meaningful, we need a measure of distance (i.e., a metric) or volume (i.e., a measure).

Metric Space = a topological space w/ metric

Measure Space = a topological space w/ measure

without a measure or metric, you could do all your sampling in a "small" region

not defined



Review

Properties of metrics:

Let ρ denote a metric.

Let a, b, c, be points in the space, X

Nonnegativity: $\rho(a,b) \geq 0 \quad \forall\, a,b \in X$

Reflexivity: $\rho(a,b) = 0$ iff $a = b \in X$

Symmetry: $\rho(a,b) = \rho(b,a) \quad \forall\, a,b \in X$

Tri. Ineq.: $\rho(a,b) + \rho(b,c) \geq \rho(a,c) \quad \forall\, a,b,c \in X$

<u>Some metrics</u>

L_p metrics: $$\rho = \left(\sum_{i=1}^{n} |a_i - b_i|^p \right)^{1/p} \qquad p \geq 1, \quad a,b \in \mathbb{R}^n$$

L_2 = the "usually" Euclidean distance

L_1 = Manhattan distance

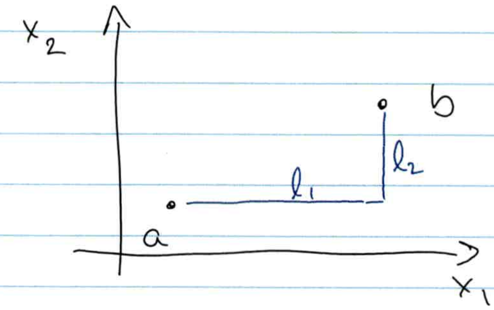
$L_\infty = \lim_{p \to \infty} \rho = \max_i \{|a_i - b_i|\}$

Example:

$L_1 = l_1 + l_2$

$L_2 = \sqrt{l_1^2 + l_2^2}$

$L_\infty = l_1$



Important! A product of metric spaces is a metric space.

Let (X_1, ρ_1) and (X_2, ρ_2) be metric spaces.

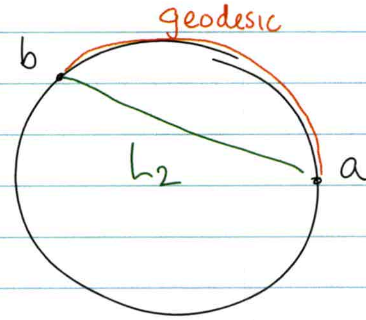
$(X_1 \times X_2, \ c_1\rho_1 + c_2\rho_2)$ with $c_1, c_2 > 0$

is a metric space.

$\Rightarrow$ If you have a metric on $\mathbb{R}^3$ and another on $SO(3)$, then you have one on $SE(3)$.

Metrics on $SO(2)$

$SO(2) = \{(a,b) \in \mathbb{R}^2 \mid a^2+b^2=1\}$



L_2 vs. geodesic

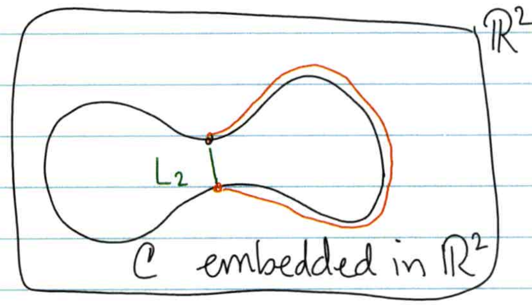
geodesic is shortest path in the space between two points

In arbitrarily curved spaces, geodesics can be hard to compute.

Also, in such spaces L_p can be very bad.

For example:

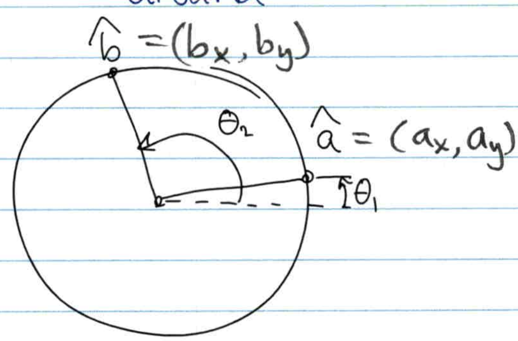
This kind of problem can arrise in closed kinematic chains



Good metric for S^1 is: (assume 1 parameter rep.)

$$\rho = \min\{|\theta_1-\theta_2|,\ 2\pi - |\theta_1-\theta_2|\}$$ ← handles wrap-around.

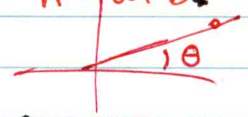
this metric assumes $\theta_1, \theta_2 \in [0, 2\pi)$



Suppose we want to use ~~4~~ 2 parameters, Use unit vectors.

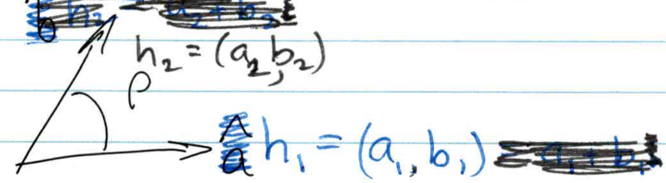
but instead of using cord length for the metric, use arc length?

S^1 represented by 2 numbers $h = a + bi$ w/ $a^2+b^2=1$



Here's how....

Alternative based on dot product in $\mathbb{R}^2$

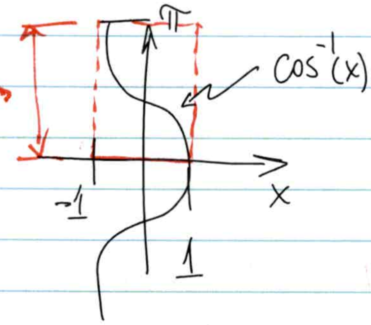


$$h_1 \cdot h_2 = \underbrace{\|h_1\|}_{1} \, \underbrace{\|h_2\|}_{1} \cos(\rho)$$

$$\rho(h_1, h_2) = \cos^{-1}(a_1 a_2 + b_1 b_2)$$

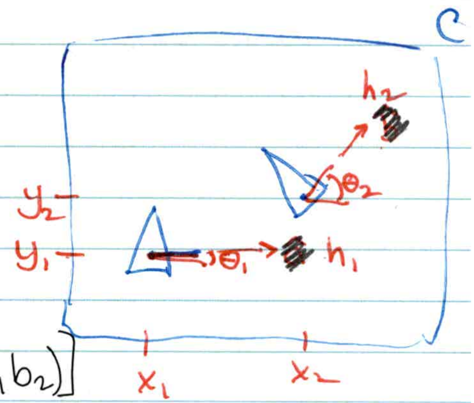
$$\cancel{\rho(a,b) = \cos^{-1}(a_x b_x + a_y b_y)}$$

To make arc cos a function must map to $[0, \pi)$



A metric on SE(2) embedded in $\mathbb{R}^4$

$$q = (x, y, a, b)$$



$$\rho = c_1 \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} + c_2 \left[\cos^{-1}(a_1 a_2 + b_1 b_2)\right]$$

or $$\rho = c_1 \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} + c_2 \min\{|\theta_1 - \theta_2|, 2\pi - |\theta_1 - \theta_2|\}$$

The usually Euclidean norm could be used too.

Note: c_1 & c_2 can be used to deal with units mismatch

radians, degrees, ...

feet, millimeters, ...

Sometimes useful to choose c_1 w/o units $c_1 = 1$

and choose c_2 with units of length.

Then c_2 plays role of "characteristic length"

length × radians = arc length

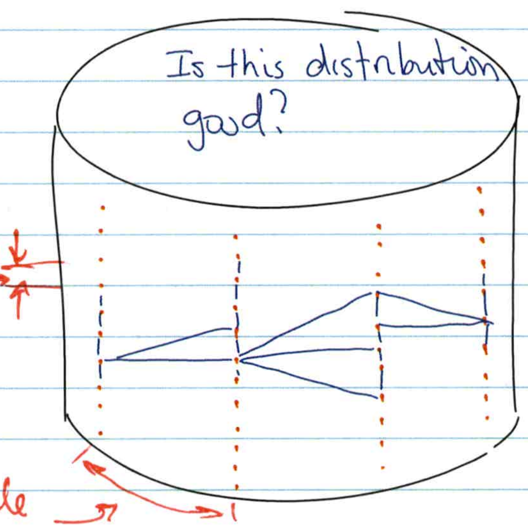
OR Suppose L is lenght of robot. Then use:

$$\frac{\sqrt{(x_1-x_2)^2+(y_1-y_2)^2}}{L}$$

in previous metric

E.g. $S^1 \times I^1$

Depending on metric, the vert. distance may be equal to the distance around the circle



Maybe it is good if G connects the columns

A Metric for SO(3): unit quaternion, $h = a + ib + jc + kd$

$h = (a, b, c, d)$

Use quaternions, h_1, h_2.

$$\rho(h_1, h_2) = \min(\rho_s(h_1, h_2), \rho_s(h_1, -h_2))$$

where

$$\rho_s(h_1, h_2) = \cos^{-1}(a_1a_2 + b_1b_2 + c_1c_2 + d_1d_2)$$

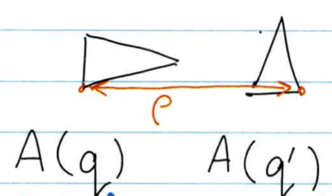
spherical linear interpolation.

This metric pushes the cord in $\mathbb{R}^4$ out onto the surface of the 3-sphere, S^3, on which the quaternions live. ρ_s is the length of the geodesic between 2 quaternions.

Preferably your metric represents a physical property important to your problem.

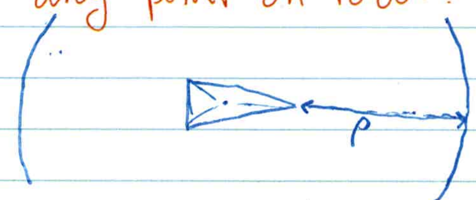
If collisions are an issue, maybe metric should be...

$$\rho(q,q') = \max_{a \in A} \{a(q) - a(q')\}$$

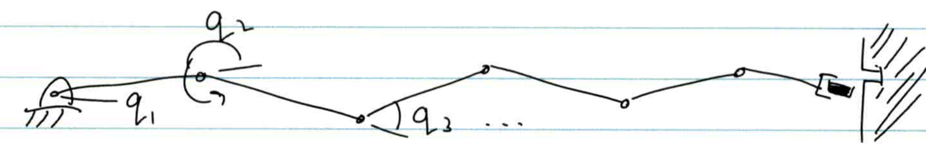


Max displacement of any point on robot.

? Put robot in circle for conservative, but fast collision checking.



Maybe doing assembly with a highly articulated robot.

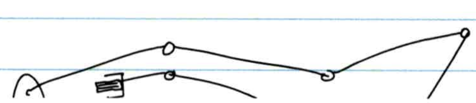


$$\rho = \cancel{c_1\sqrt{(q_1-q_1')^2} + c_2\sqrt{(q_2-q_2')^2} + \cdots} = c_1 \max\{|q_1-q_1'|, 2\pi-|q_1-q_1'|\} + c_2 \max\{|q_2-q_2'|, 2\pi-|q_2-q_2'|\} + c_3 \max\{ \quad \} \; \vdots$$

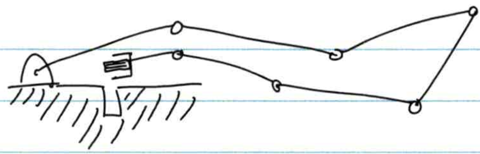
Maybe $0 < c_1 < c_2 < c_3 < \cdots$ ← Weights could be based on arm Jacobian.

$J\Delta q = \Delta x$

What if we have a different config?



Maybe c_is are



Maybe c_is are config dependent.

Metric based on Jacobian will provide good weighting.

$$\rho = \left(\mathrm{Det}\left(J(q)\, J^T(q)\right)\right)^{1/2}$$

if manipulator is not redundant, then this reduces

$$\rho = \left|\mathrm{Det}\left(J(q)\right)\right|$$

Configuration dependence would adjust sampling density across various parts of C-space.

These are like manipulability metrics

fast motion

Robotics I

manip metric $= \sqrt{\mathrm{Det}(J J^T)} =$ Vol of ellipsoid

This metric captures kinetic energy & inertial effort required to move.

manip with dyn $= \sqrt{(J M^{-1} M^{-T} J^{T})}$