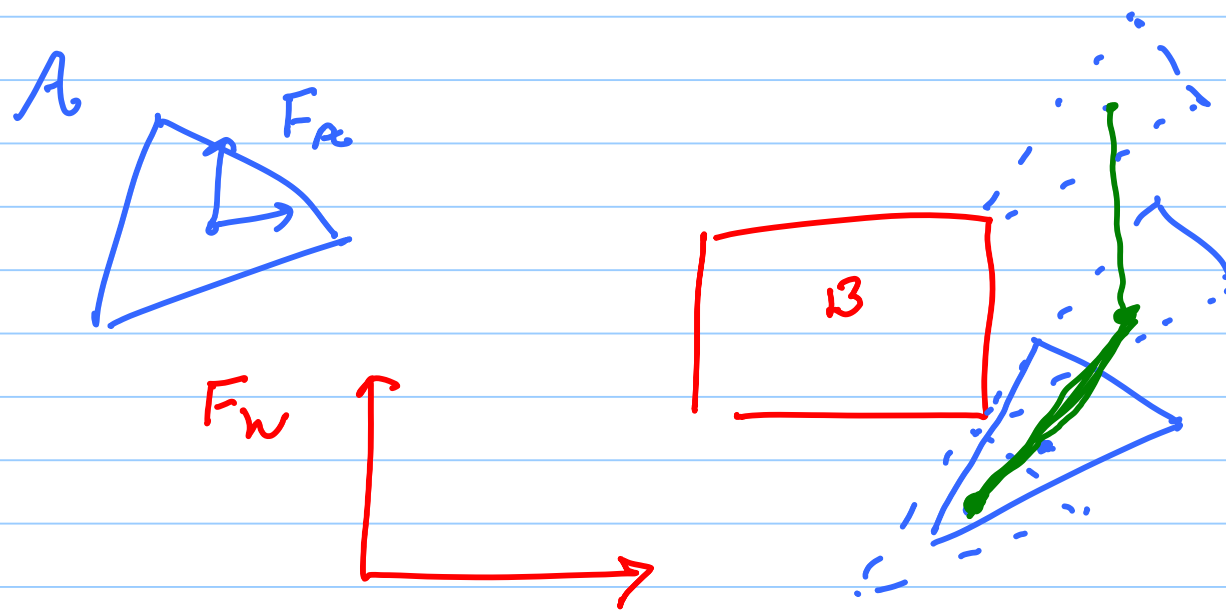
Note Title

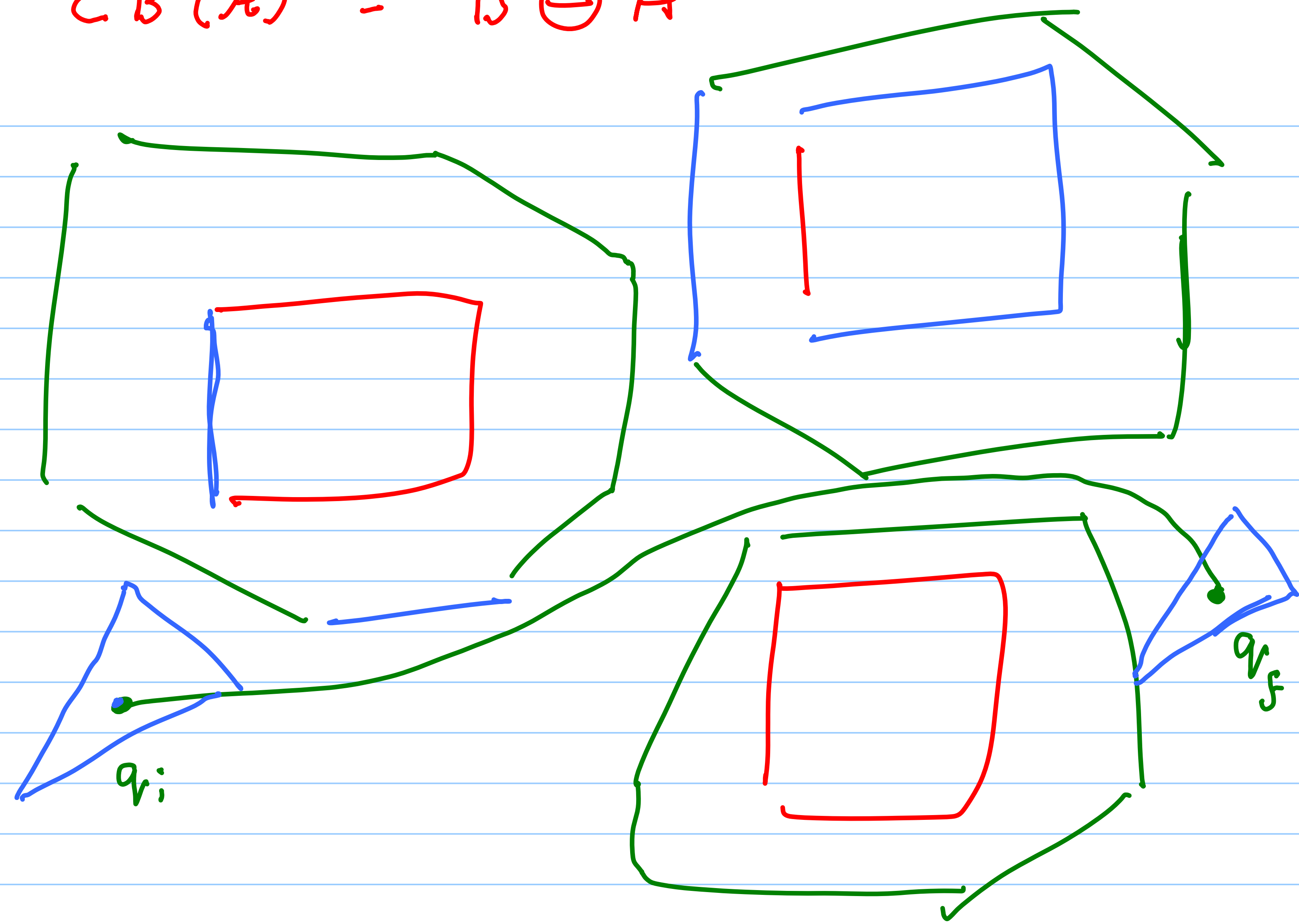
Lecture 8

1/31/2012

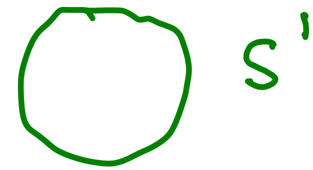
C-obstacle computation for polygonal, translation only case



$$C_B(A) = B \ominus A$$

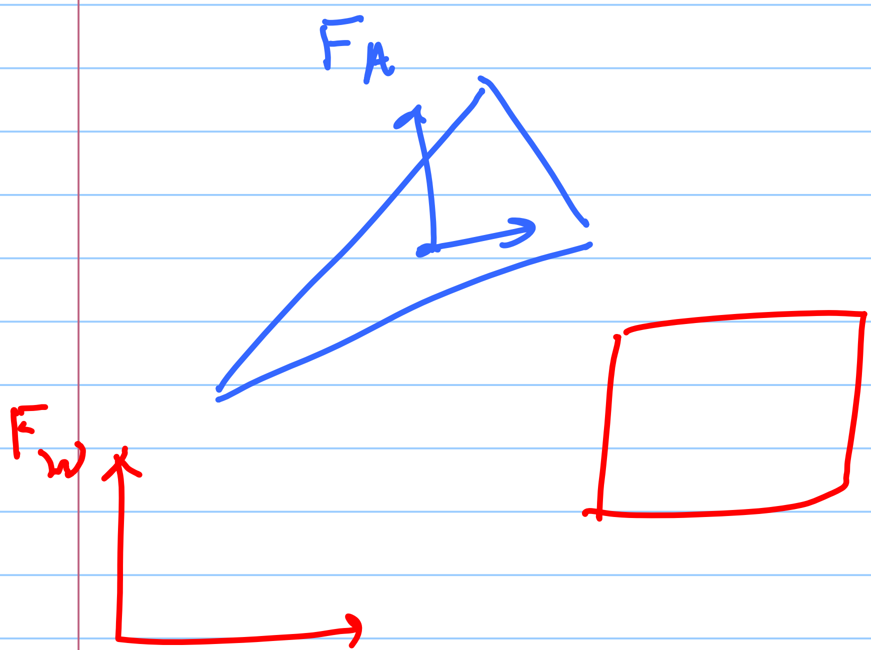


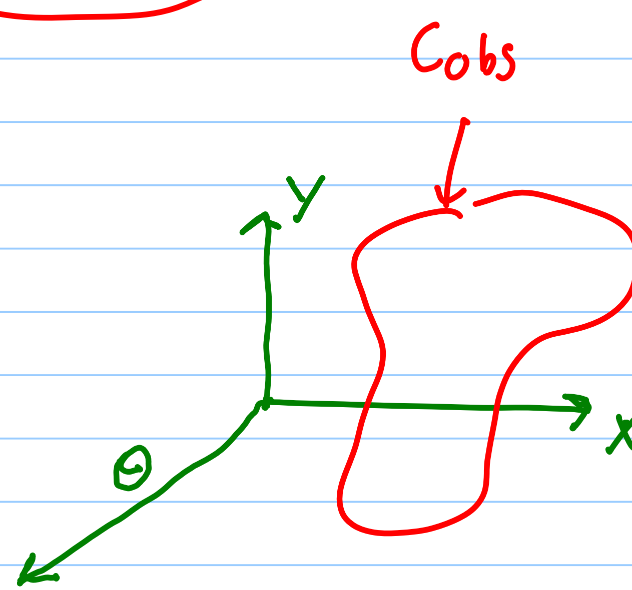
More general case: rotations are allowed.



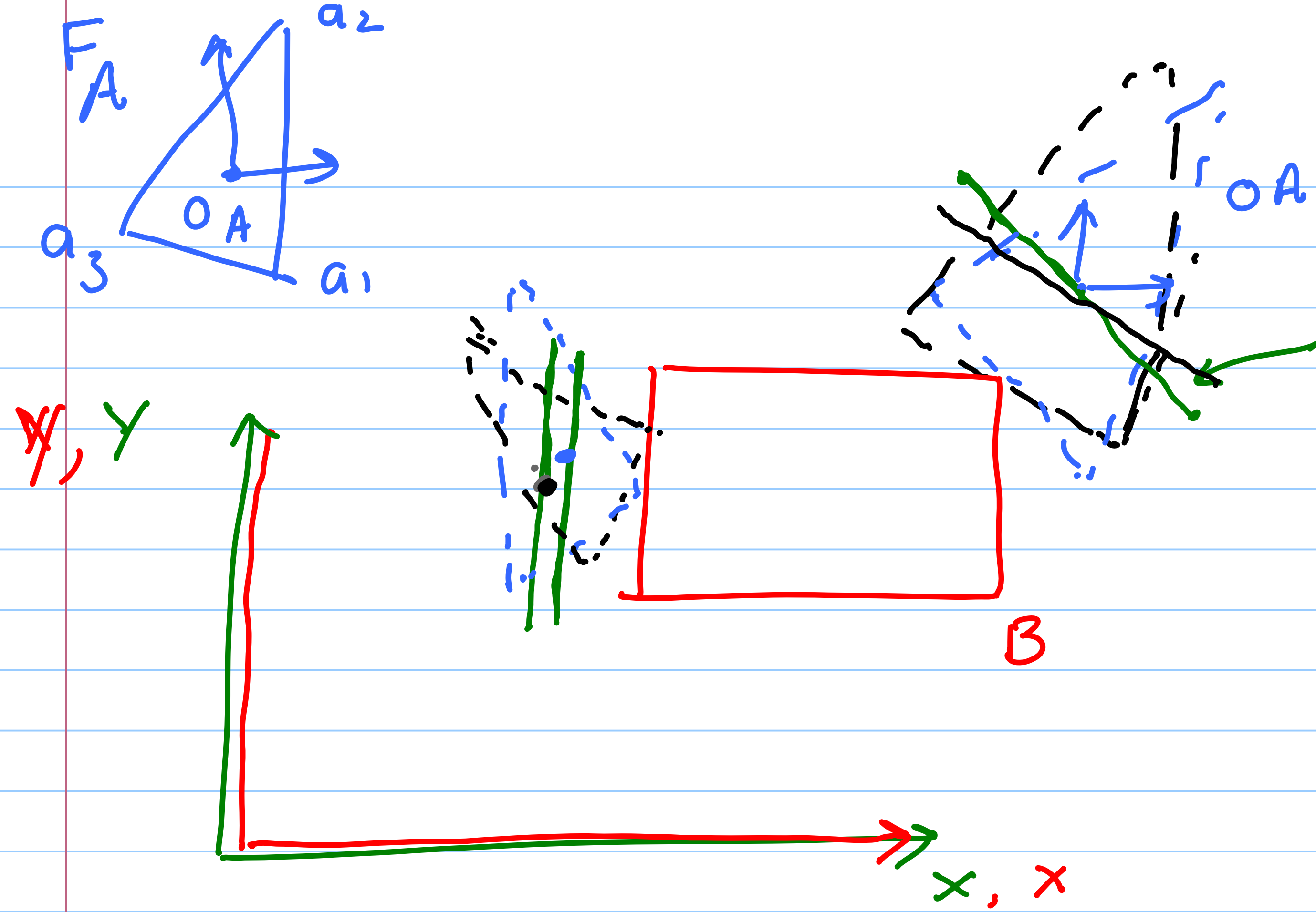
$\mathbb{R}^2 \times SO(2)$: $\mathbb{R}^2 \times \Theta$

$\Theta \equiv \mathbb{R}^1$ locally





e:



type A for given θ_0

See Fig in Ch3 in Latombe text.

Summary: 1) we know for fixed θ:

CB_{A_θ} is a convex polygon.

two types of interaction:

$\forall$ edge of CB_{A_θ} :

→ type A edge

" edge of A, vert. of B

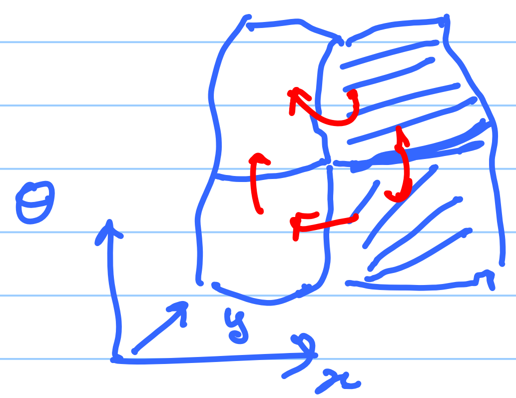
→ type B edge

edge of B, vertex of A

as θ changes :

1) type A edge "rotates"

2) type B edge "translate"



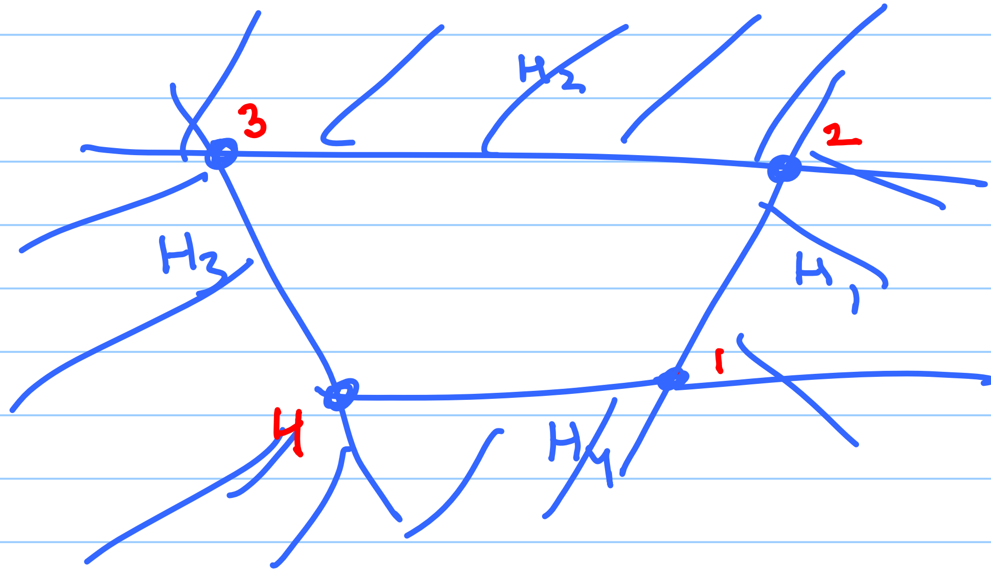
How do we represent such an entity

$H_i : a_i x \, b_i y + c_i \leq 0$

Representation of geometric entities

~~3D~~ Convex

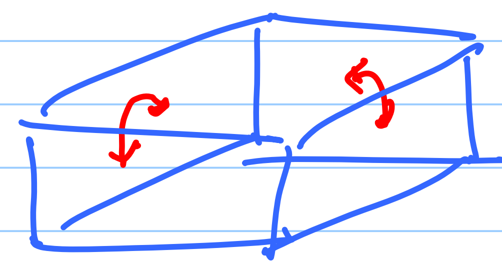
2D polygon: $H_1 \wedge H_2 \wedge H_3 \wedge H_4$



list of vertices in a pre-defined order

3D) polyhedra

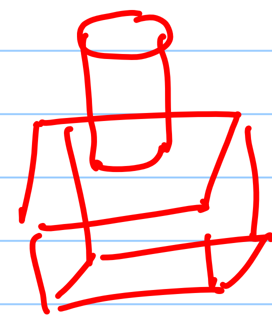
1)



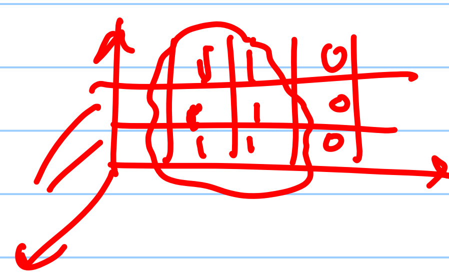
Boundary rep.

2) Constructive Solid Geom.

CSG



3) spatial occupancy enumeration



for $CB_A : \underbrace{(x, y, \theta)}_{q}$

$CB_A(q)$: for a given q, if

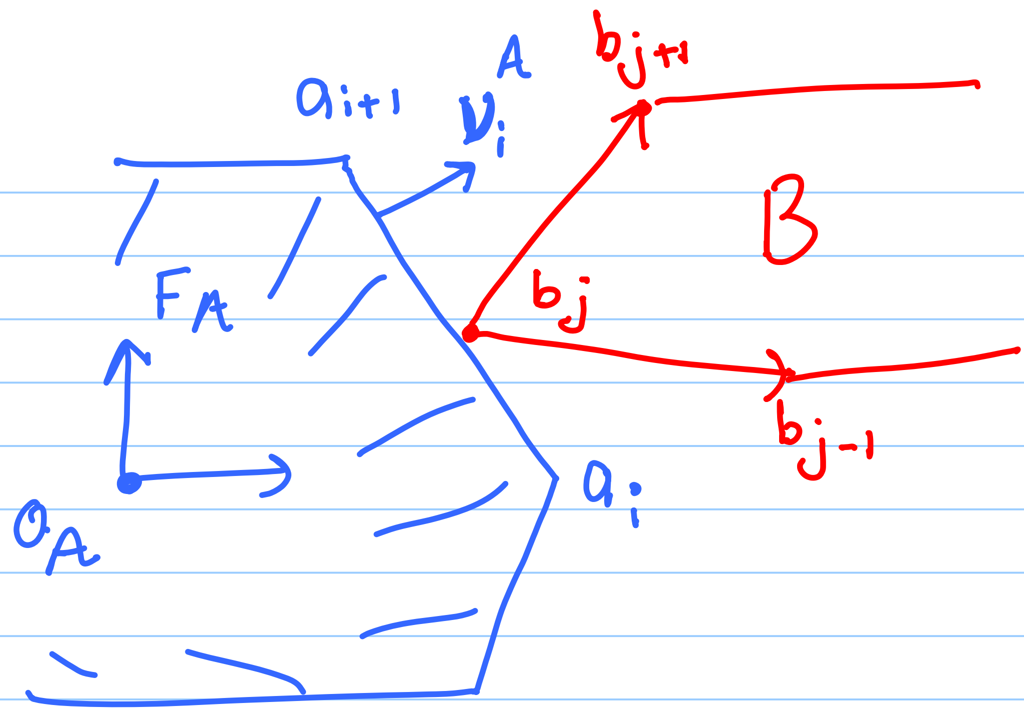
$q \in CB_A$

predicate type rep.

eqn. describing type A ruled surface :

1) valid over a θ interval

2) x, y "range" : finite edges



1) orient. Constraint:

$\mathrm{Appl.}_{ij}(q)$:

$$\nu_i^A \cdot (b_{j-1} - b_j) \geq 0$$

$$\wedge$$

$$\nu_i^A \cdot (b_{j+1} - b_j) \geq 0$$

2) patch surface eqn:

b_j lies on edge $\overline{a_i a_{i+1}}$

$$f^A_{ij} = \nu^A_i(q) \cdot [b_j - a_i(q)] = 0$$

≤ 0 for inside c-obs.

$$\mathrm{CONST}^A_{ij}(q) : \mathrm{Appl}^A_{ij}(q) \Rightarrow f^A_{ij}(q) \leq 0$$

for parametric forms (x, y, θ) : see

Latombe chapter 3.

Similarly : type B

$\mathrm{CONST}^{B}_{ij}(q)$: . . . -

$$CB(q) = \left(\bigwedge_{i,j} \mathrm{CONST}^{A}_{ij}(q) \right) \bigwedge_{i,j} \mathrm{CONST}^{B}_{ij}(q))$$

BOTTOM LINE:

C-obs surfaces are highly non-linear, complex entities, difficult to compute "explicit" boundary description.

|| Explicit description can be computed Avnaim & Boissonat

Q: How do we design path planning algorithms to search C_{free}.

"Roadmap" Approaches.