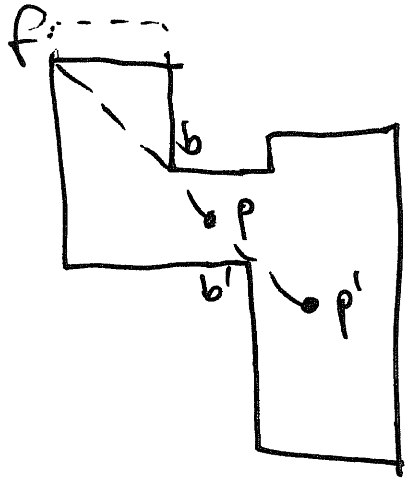
GREEDY-ONLINE - a competitive online strategy for finding an optimal watchman route in a simple ~~orthogonal~~ rectil. polygon
(„How to learn an unknown environment I: The rectilinear case",
Deng, Kameda, Papadimitriou 1997)

Given: * robot/watchman with unlimited, continuous vision
* simple rectilinear polygon P, start/end point x_0
We ask for: * shortest tour (L_1-norm), starting and ending in x_0, (of) ~~such~~
that allows to draw a map of P
(~~P must be~~ each point in P must be visible from one point along the route)

Visibility:



robot at p:

* sees all points below f, not f itself

robot at p':

* sees nothing of edge on which f lies

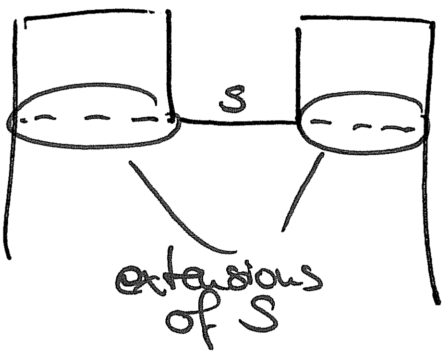
A simple observation:

<u>lemma 1:</u> ~~A~~ path sees all ~~parts of a sim~~ points of a simple polygon $\Longleftrightarrow$ it sees all edges of the polygon.

extensions

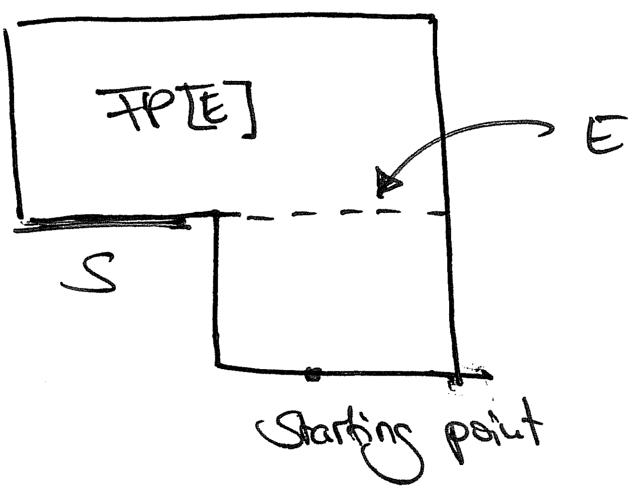
We extend each ~~side~~ ~~edge~~ side S of the polygons (possibly in both directions) until we touch P's boundary.

Each connected segment of this line segment without S is called an <u>extension</u> of S. (0, 1 or 2 per side).



Each extension E of a side S partitions the polygon into two subpolygons:

* the polygon that contains the starting point, the <u>home polygon</u>
* the other subpolygon defined by E: the <u>foreign polygon</u> $FP[E]$.

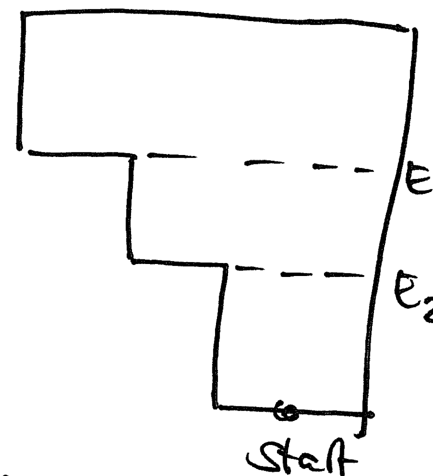


From the starting point a side $S \in FP[E]$ becomes visible only if E is visited (crossed or touched)

$\Rightarrow$ E is a necessary extension

dominance

Let E_1, E_2 be two necessary extensions
E_1 dominates E_2, if E_1 is fully contained in $FP[E_2]$.



A non-dominated necessary extension is called an essential extension
(// Depends on the starting point! //)

Claim 1: ~~In order to see all sides of the polygon,~~ (V)

To see all sides of a polygon, starting at the entry, it is necessary and sufficient to touch every ~~critical~~ essential extension.

Proof sketch: * sufficient to see FP[E] : touch E

* cut off FP[E](P') ⟶ new essential extension E' in P'

↳ proof by induction

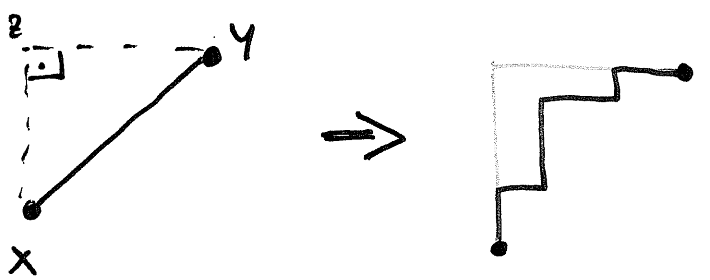
Problem: It is impossible to recognize essential extensions during the online exploration!

↳ So: Which to visit first?

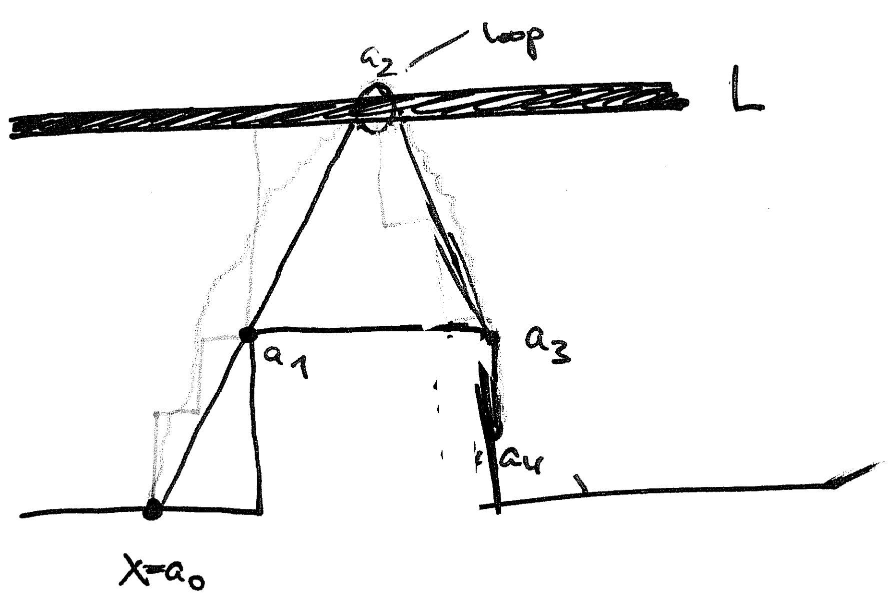
⟹ We consider a greedy algorithm

We need:

* rectilinearization:



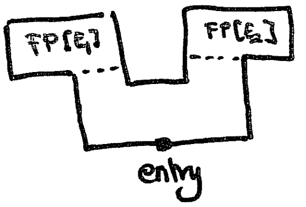
* taut-thread-principle:



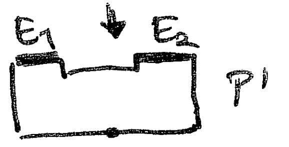
GREEDY approaches and necessary extensions

entry point $\rightarrow$ gives "natural" order of essential extensions in which they should be visited:
in clockwise order along the boundary of
$P \setminus \bigcup_{E_i} FP[E_i] =: P'$

E_i essential extension



$\Rightarrow E_1, E_2, \dots, E_m$



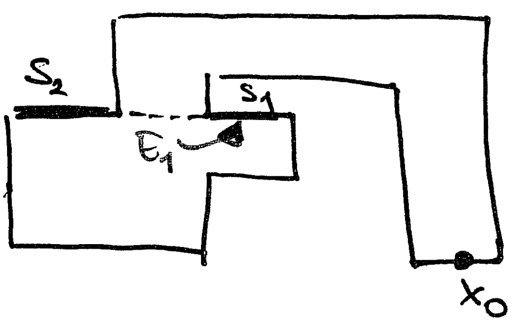
Greedy algorithm that follows that order: Greedy Critical GC_*

$\Uparrow$ that's the offline optimum !!

<u>Problem</u>: We cannot follow a shortest route connecting E_1 to E_2 ... E_m in an online fashion !!

Each essential extension „belongs" to a side of the polygon P

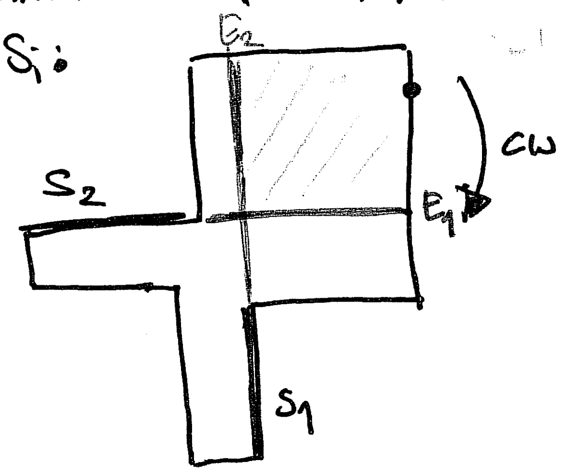
$\Rightarrow$ „natural" order for sides along P's boundary
(tie breaking: extension that belongs to two sides is associated with the one that comes first in cw order)



$\Rightarrow$ $\forall i = 1, \dots, m$: unique side S for E_i
$\hookrightarrow$ $S_1, S_2, \dots, S_m$

Attention! E_i is not necessarily an extension of S_i !!
Let σ be the permutation of $\{1, 2, \dots, m\}$ such that $E_{\sigma(i)}$ is an extension of S_i:

Example:



⇒ GREEDY : Greedy Sides (GS) (offline)

- start in $y_0 = x_0$
- Move to $E_{G(1)}$ along the shortest path → point y_1

⋮

↑ path GS_s: visit extensions of $S_1, S_2, \ldots, S_m$ in this order

Lemma: If entry and exit point are the same point on ~~the~~ P's boundary, GS is an optimal watchman route.
(w/o proof).

→ An Online Strategy:

* use the „taut-thread" - principle

* try to ~~follow G~~ find and follow GS online

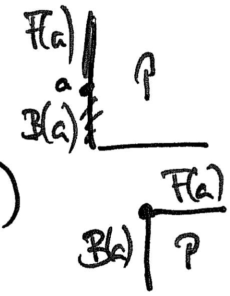
BUT: we cannot always identify essential extentions when we first visit them
– possible for necessary extensions!

basic idea: follow the rectilinearizations of the segments of the "taut-thread" path that starts at the entry, has loops around all essential extensions, and ends at the exit.
(If two extensions intersect the loop is placed around the intersection point.)

$\Rightarrow$ maintain a current map of P

a few definitions:

For a point a on the boundary:



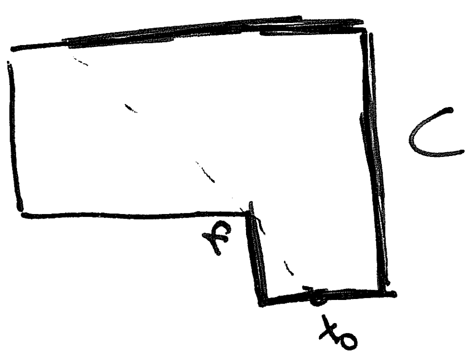
– $F(a)$: side cw ~~before a~~ after a (forward)

– $B(a)$: — " ———— before a (backward)

Let z be the current position:

We need to identify the next (cw) extensions.

Let C be the continuous part of the boundary, that from x_0 has been visible at some point in time so far. Let f be C's end in cw direction.



If C grows, f moves, but not in a continuous way!

GREEDY ONLINE

Input: simple rect. pol. P, exit/entry $z_0 = x_0$

Output: shortest route, starting/ending in x_0, to draw a map of P

- $z = z_0$
- $\mathcal{H}$ = part of boundary of P, visible from z_0
- f: initial frontier
- C: part between z and f
- b: blocking corner for f (if exists)

~~Repeat~~

Repeat (1)-(3), update $\mathcal{H}$, C, f

If $\mathcal{H}$ complete → finish   Michael: „fertisch"

Let Ext(F(f)) be the extension incident to f

Ext(B(b)) —— " —— " —— b

(1) • If f is a 270° corner: $t = f$

$E = Ext(F(f))$

• Otherwise (f located on an edge): let b be the blocking corner, where f^- was visible

$t = b$

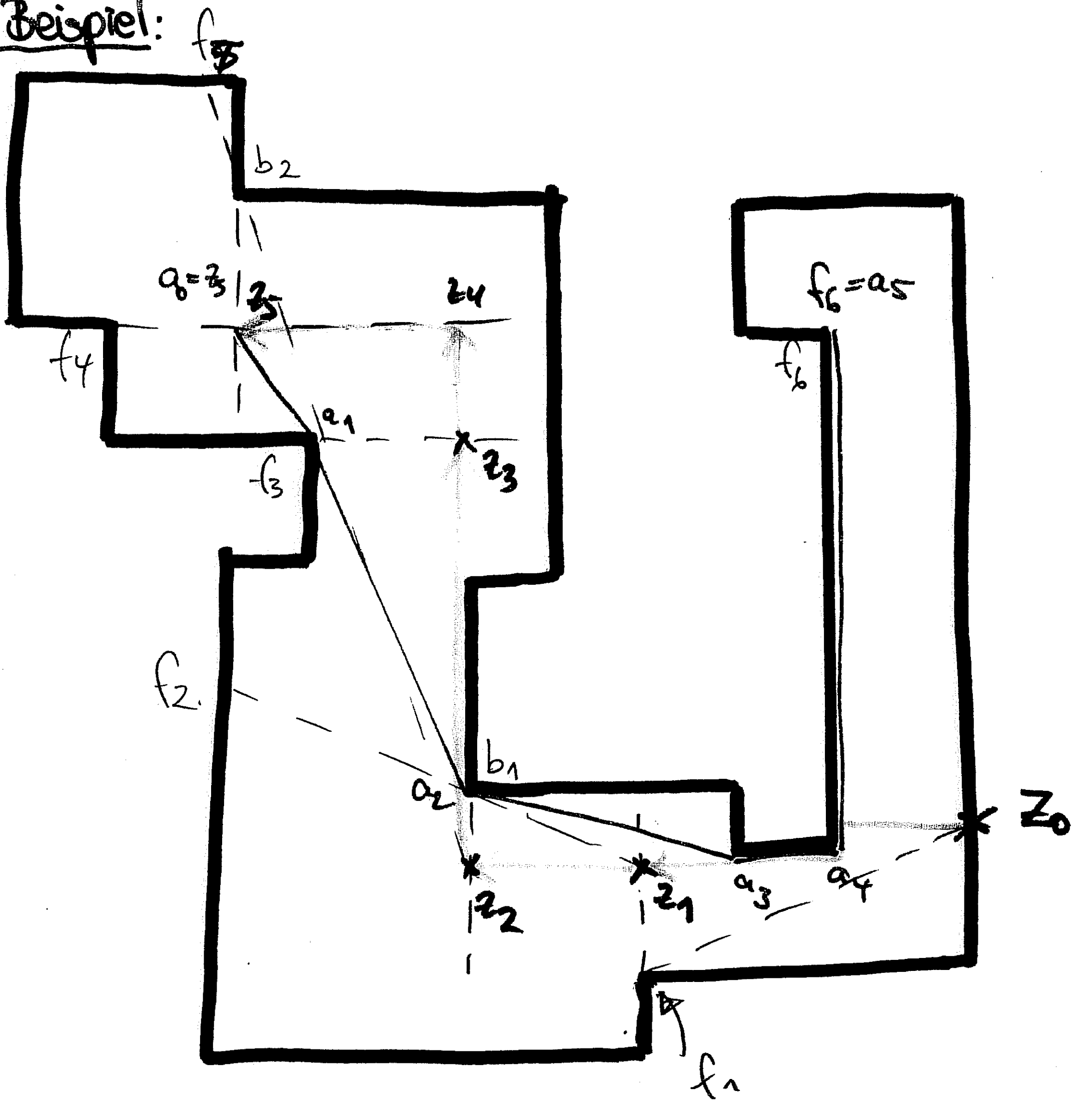
$E = Ext(B(b))$

(2) Consider a thread between z and E (touches vertices of P at $a_1, \dots, a_e$)

(3) If $e \geq 1$: use rectilinearization with $a_1, \dots, a_{e-1}$

: blabla

4. Ein Beispiel:



~~5. Optimalität von GREEDY-ONLINE~~

~~GO: Pfad von GREEDY-ONLINE~~

Go is 1-competitive for entry/exit on boundary (and L_1-norm!)
and 2-competitive — " — in the interior.