Online Motion Planning MA-INF 1314
Pledge with sensor errors

Elmar Langetepe
University of Bonn

Graph-exploration

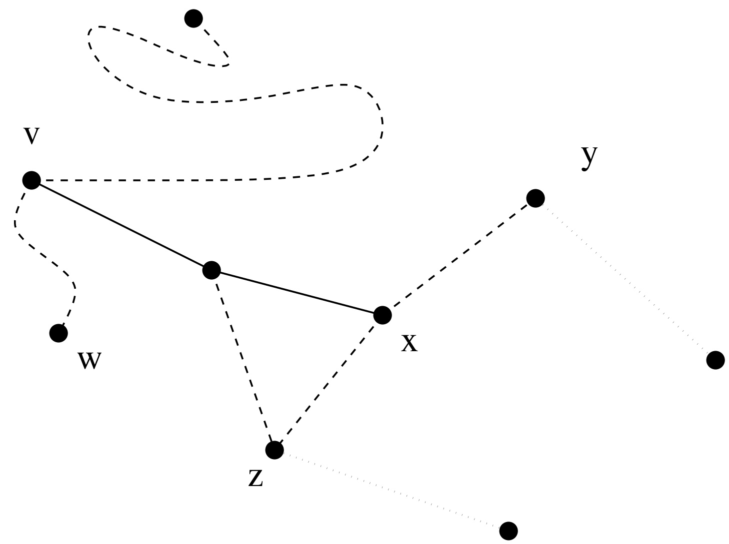
- Labyrinths, grid-graphs, gridpolygons, general graphs
- Graph-exploration: DFS and LB of 2
- Gridpolygons: Simple/general
- SmartDFS $\frac{4}{3}$, LB $\frac{7}{6}$
- STC Alg. $|C| + |B|$
- Tether/Accumulator/Depth variants: $\Theta(|E| + |V|/\alpha)$
- Marker Algorithm
- Online TSP for planar graphs!

Rep.: Shortcut algorithm

- Online DFS, boundary edge (x, y) *blocks* boundary edge (v, w)

  1. $|xy| < |vw|$ and
  2. $d_G(v, x) + |xy| < (1 + \delta)|vw|$

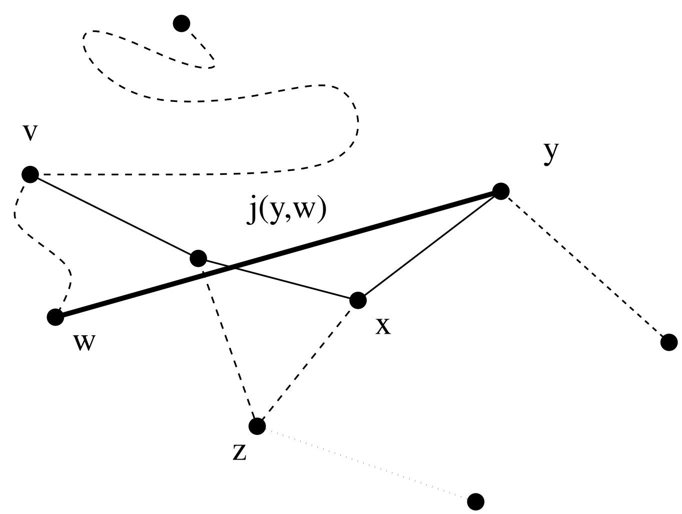
- Later: You eventually move from y over x to v!



Rep.: Shortcut algorithm

- *Shortcut* non-blocked boundary edge (x, y), charge edge (x, y)
- Blocked edge (v, w) becomes unblocked by visit of y
- Build jump edge $j(y, w)$ of length $d_G(v, w) < (2 + \delta)|vw|$, if visited, charge (v, w)

1. $|xy| < |vw|$ and
2. $d_G(v, x) + |xy| < (1 + \delta)|vw|$



Rep: Pseudocode: Shortcut(x, y, G)

Traveling from x, we have visited y for the first time

for all Boundary edges (v, w) **do**

  if Visiting y caused $Block((v, w))$ to become empty **then**

    Build jump edge $j(y, w)$, append jump edge

      on $Incident(y)$ and $Incident(w)$

  end if

end for

for all Edges $(y, z) \in Incident(y)$ **do**

  if z is boundary vertex and (y, z) shortcut **then**

    Traverse the egde (y, z); Shortcut(y, z, G)

  else if z is boundary vertex and (y, z) is jump edge **then**

    Move to z along the shortest known path; Shortcut(y, z, G)

  end if

end for

Return to x along the shortest known path

Rep: Shortcut algorithm/Analysis

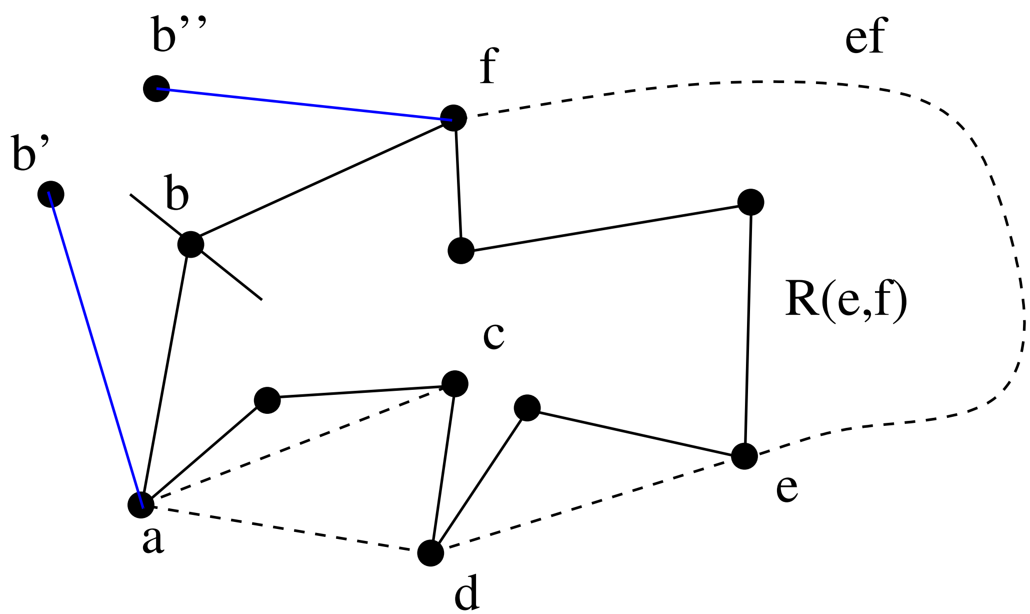
Theorem: For planar graphs the above Shortcut algorithm is 16-competitive.

Proof:

- Charged edges! Shortcuts, jump edges, set P
- $2(2+\delta)|P|$ in total! Factor 2 DFS forth and back!
- MST of G, $|OPT| \geq |MST|$
- Show $|P| \leq (1+2/\delta)|MST|$
- $2(2+\delta)|P| \leq 2(2+\delta)(1+2/\delta)|MST| \leq 2(2+\delta)(1+2/\delta)|OPT|$
- Minimize $2(2+\delta)(1+2/\delta)$, $\delta = 2$, 16!

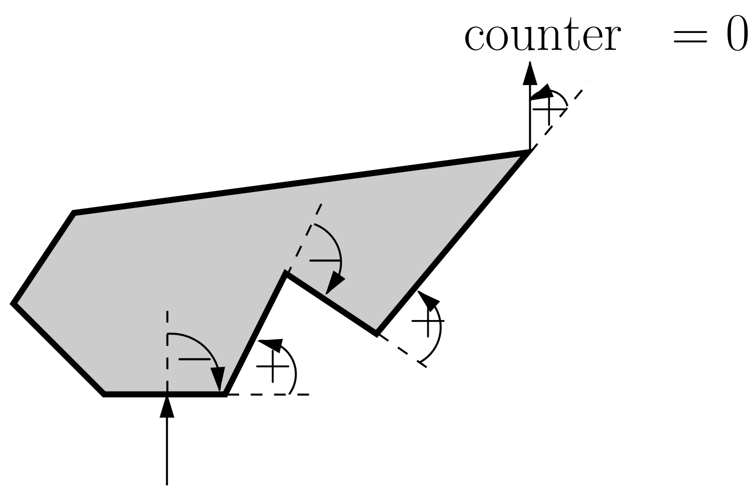
Rep.: Shortcut algorithm/Analysis

- Proof: $|P| \leq (1 + 2/\delta)|MST|$
- Show: $|C| \leq |R|/\delta$, $C = MST \setminus P$
- Open the ring accordingly
- Process chords from *inside* to outside, budget $|R|$



Rep.: Pledge Algorithmus

- Point-shaped agent/touch sensor
- Modi: Follow wall, follow a direction (exact)
- Single angluar counter



Rep.: Pledge Algorithmus

1. Move into starting direction φ, until the agent hits an obstacle.
2. Rotate (right-turn) and follow the wall by Left-Hand-Rule.
3. Sum up the rotational angles until total angular counter gets zero, then GOTO (1).

Possible errors: Counting angular rotations, hold the direction

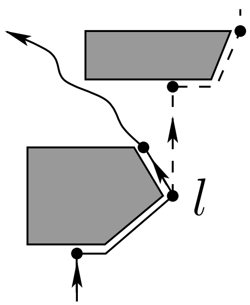
Rep: Correctness error-free pledge

- **Lemma** Angular counter is never positive.
- **Lemma** In case of failure: Finite path $\Pi_\circ$ is repeated again and again.
- **Lemma** In case of failure: $\Pi_\circ$ has no self-intersections.
- **Theorem** Pledge finds an exit, if there is an exit.

$\Pi_\circ$ cw-order, Left-Hand-Rule, enclosed!

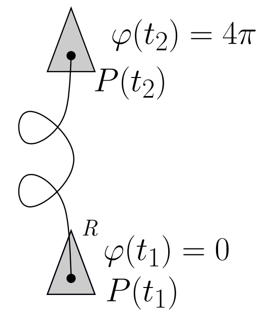
Pledge algorithm with sensor errors

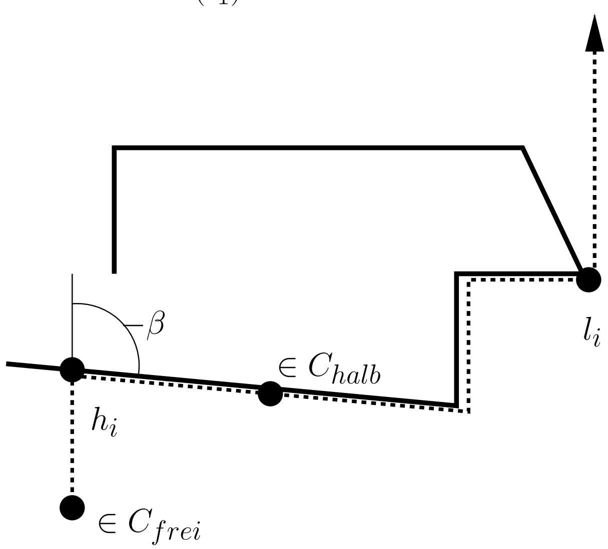
- Possible errors?
- Left-Hand-Rule, stable!
- Counting rotational angles!
- Hold the direction in the free space!
- For example: Compass!
- Full turns ok, but not precisely!
- Leave the obstacle slightly too early or too late!
- The main direction can be hold!
- Still correct?



Notation/Model

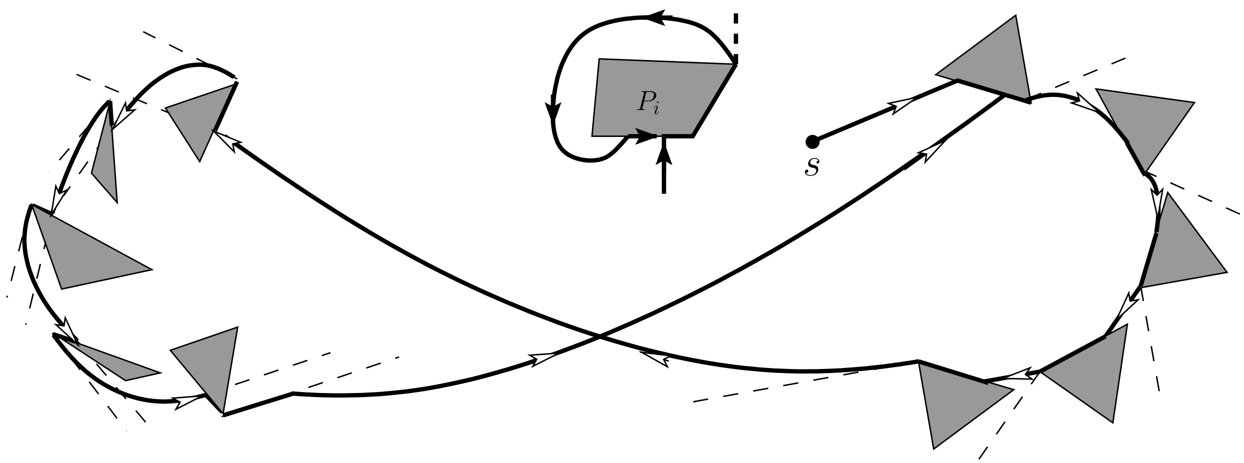
- Curves of the work-space
- Turning angle, position (ref. point): $C(t) = (P(t), \varphi(t))$ with $P(t) = (X(t), Y(t))$
- For simplicity: point-shaped agent
- Hit-Point obstacle: h_i
- Leave-Point obstacle: l_i
- Boundary: $\mathcal{C}_{\text{half}}$, Free-Space: $\mathcal{C}_{\text{free}}$





Typical errors!

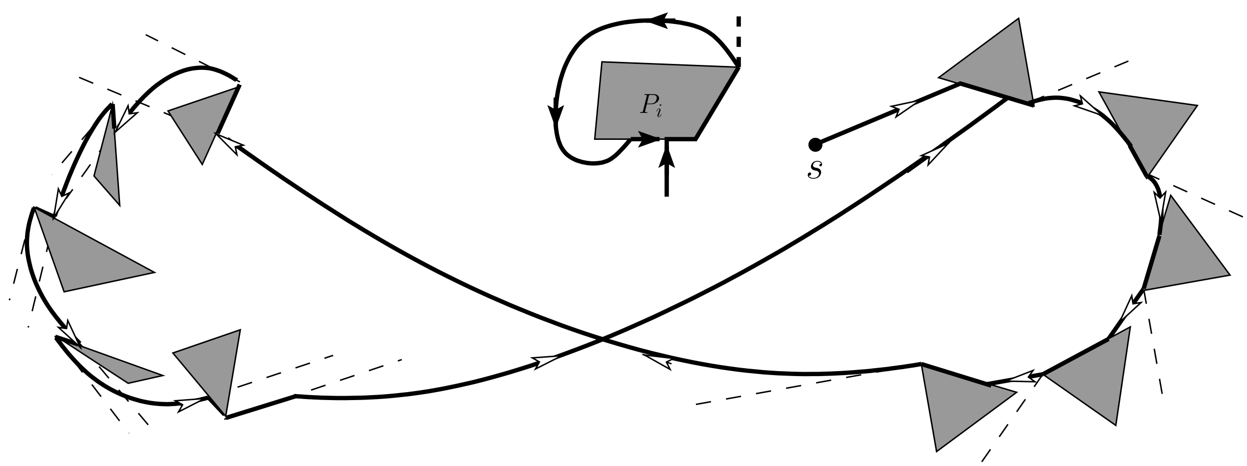
- Avoid infinite loop
- Leave into free space: Extreme direction error
- Or small errors sum up to large error
- Infinite loops!
- Condition: Leave direction has to be globally stable!



Typical errors!

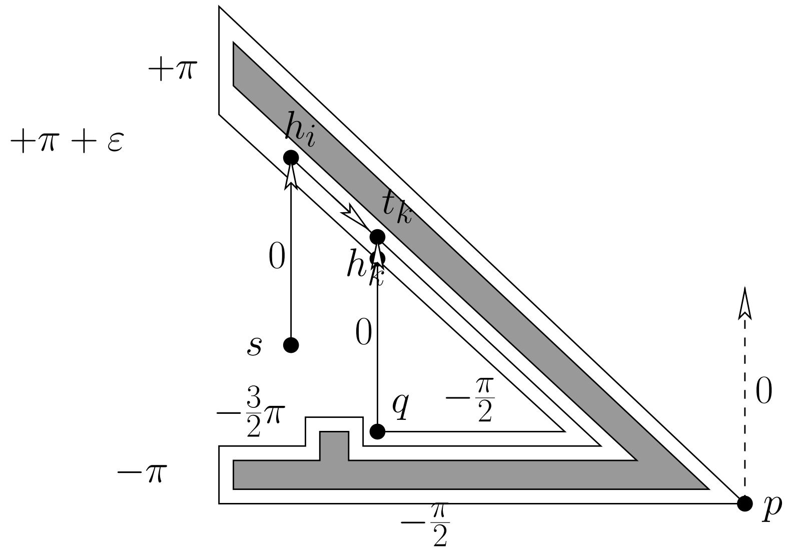
- Condition: Leave into direction X has to be globally stable!
- $\mathcal{C}_{\text{free}}$-condition for the curve!

$$\forall t_1, t_2 \in C : P(t_1), P(t_2) \in \mathcal{C}_{\text{free}} \Rightarrow |\varphi(t_1) - \varphi(t_2)| < \pi$$

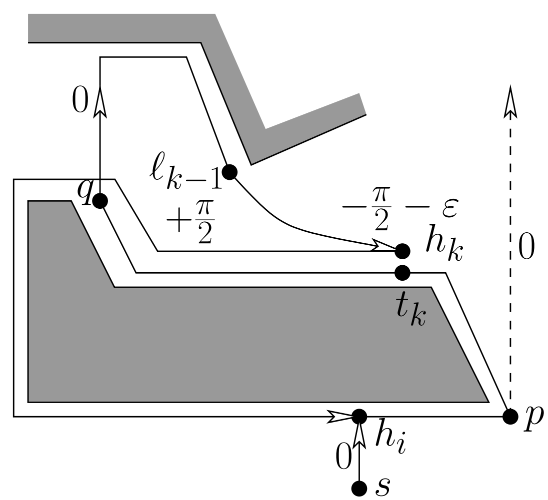


Typical errors!

- $\mathcal{C}_{\text{free}}$-condition is not sufficient
- Overturn the angular counter locally at the obstacle!
- Infinite loops



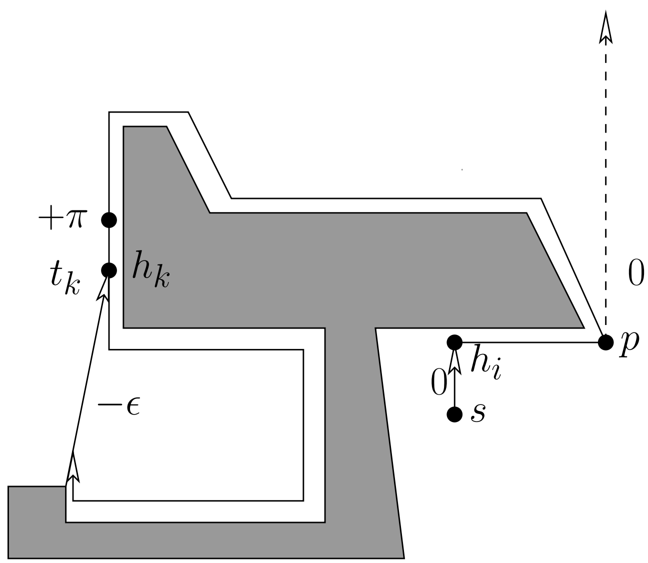
(i)



(ii)

Typical errors!

- Do not overturn the counter locally
- $\mathcal{C}_{\text{half}}$-condition: $\forall h_i, t \in C : P(t) = P(h_i) \Rightarrow \varphi(t) - \varphi(h_i) < \pi$



Pledge algorithm with sensor errors

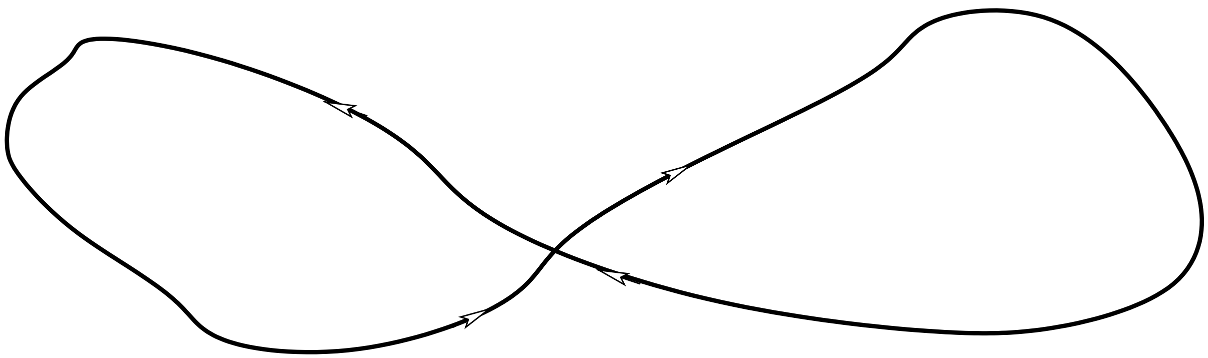
Pledge-like curve!

Def. $\mathcal{K}$ class of curves in $\mathcal{C}_{\text{free}} \cup \mathcal{C}_{\text{half}}$, with the following conditions:

1. Parameterized curve with turn-angles and position:
   $C(t) = (P(t), \varphi(t))$ mit $P(t) = (X(t), Y(t))$
2. Curve surrounds obstacel by Left-Hand-Rule
3. Leavs point is a vertex of an obstacle
4. $\mathcal{C}_{\text{free}}$-condition holds:
   $\forall t_1, t_2 \in C : P(t_1), P(t_2) \in \mathcal{C}_{\text{free}} \Rightarrow |\varphi(t_1) - \varphi(t_2)| < \pi$
5. $\mathcal{C}_{\text{half}}$-condition holds:
   $\forall h_i, t \in C : P(t) = P(h_i) \Rightarrow \varphi(t) - \varphi(h_i) < \pi$

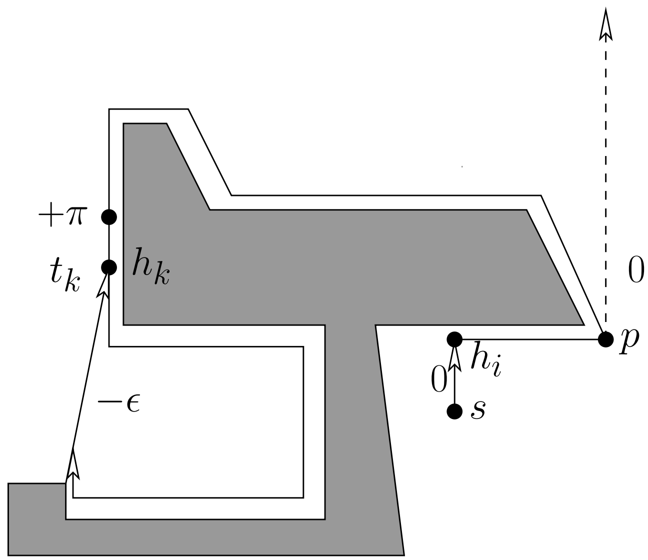
Reminder: Error situation!

- Escape direction is globally stable!
- $\mathcal{C}_{\text{free}}$-condition:
  $\forall t_1, t_2 \in C : P(t_1), P(t_2) \in \mathcal{C}_{\text{free}} \Rightarrow |\varphi(t_1) - \varphi(t_2)| < \pi$



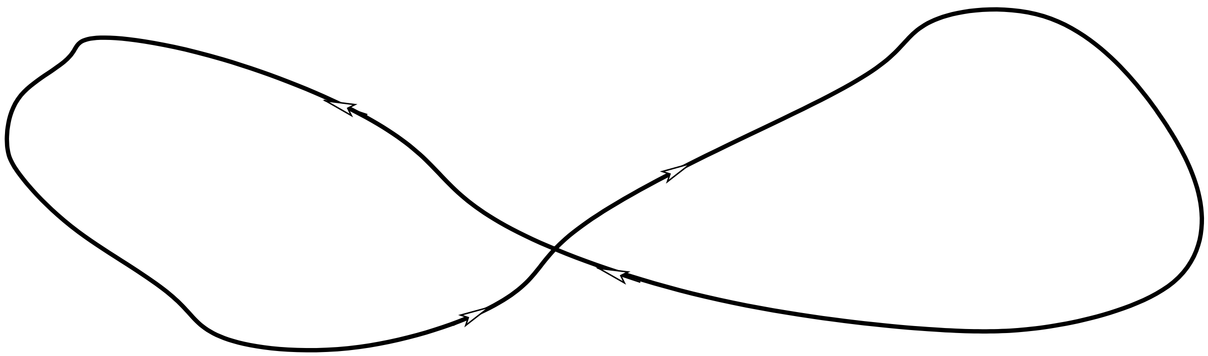
Reminder: Error situation!

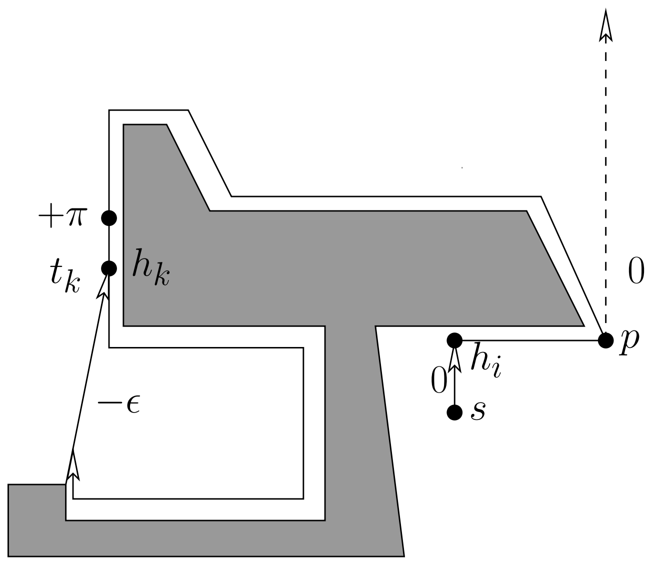
- Angular counter, no local overturn!
- $\mathcal{C}_{\text{half}}$-condition: $\forall h_i, t \in C : P(t) = P(h_i) \Rightarrow \varphi(t) - \varphi(h_i) < \pi$



Fulfill Curve-Definition: Hardware!

Compass with small deviation: Avoid situations!



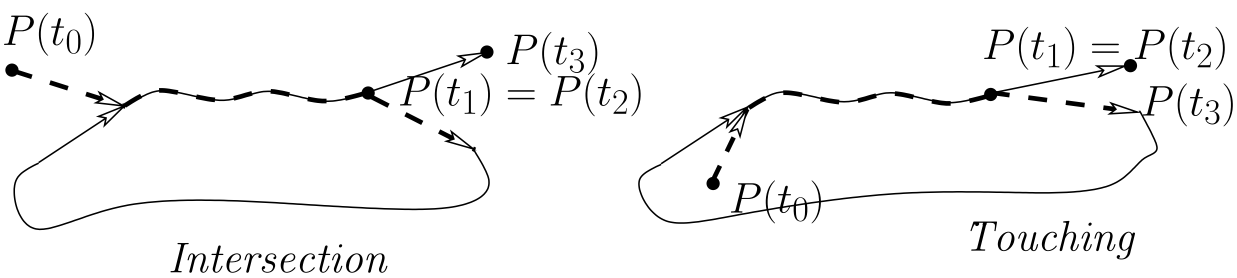


Correctness proof!

Lemma A curve from $\mathcal{K}$ has no self-intersection.

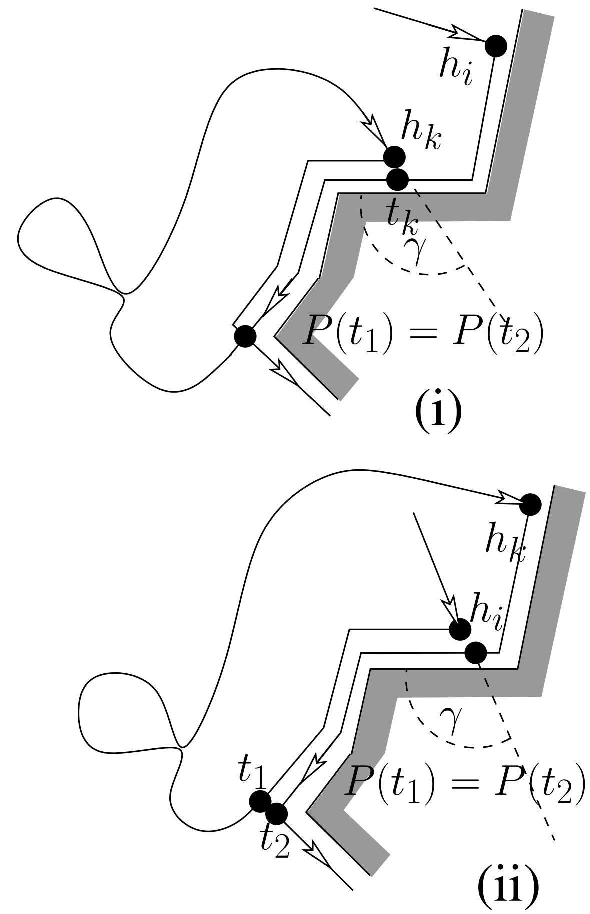
Proof:

- Assume: First crossing of C by t_1 and t_2
- Single loop from t_1 to t_2: cw or ccw
- Case 1: Crossing in $\mathcal{C}_{\text{free}}$: Contradicts $\mathcal{C}_{\text{free}}$-condition!
- Case 2: Crossing in $\mathcal{C}_{\text{half}}$



Curves of $\mathcal{K}$, no self-intersection

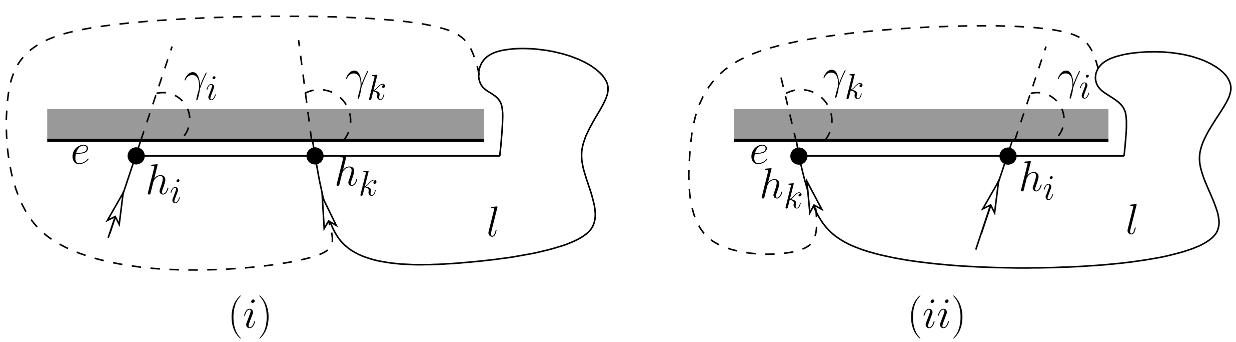
- First loop: Enter at h_i, enter at h_k again
- Intersection time t_2
- $P(h_k)$ also at t_k with $h_i < t_k < t_1$, otherwise (ii) only touching
- $\varphi(h_k^+) = \varphi(h_k) + \gamma$ with $-\pi < \gamma < 0$
- From t_k to h_k^+ full turn
- $\varphi(h_k^+) = \varphi(t_k) - 2\pi$
- $\varphi(t_k) - \varphi(h_k) < \pi$
- $\Leftrightarrow \varphi(h_k^+) + 2\pi - \varphi(h_k) = \varphi(h_k) + \gamma + 2\pi - \varphi(h_k) < \pi$
- $\Leftrightarrow \gamma < -\pi$, contradiction



Correctness proof, sensor errors

Lemma A curve from $\mathcal{K}$ *hits* any edge only once.

- By contradiction! Assume C hits e twice
- Hit at h_i, then cw (or ccw) and again at h_k
- In $P(h_i)$,$P(h_k)$ with $-\pi < \gamma_i, \gamma_k < 0$ to $\varphi(h_i^+)$,$\varphi(h_k^+)$
- h_i^+ and h_k^+ follow edge e: $\varphi(h_k^+) = \varphi(h_i^+) + 2j\pi, j \in Z$
- Loop without intersection: Two cases $\varphi(h_k^+) = \varphi(h_i^+) \pm 2\pi$
- $|\varphi(h_k^-) - \varphi(h_i^-)| = | \pm 2\pi - \gamma_k + \gamma_i| > \pi$
- $\mathcal{C}_{\text{free}}$-condition does not hold!



Correctness proof, sensor errors

Lemma For any curve from $\mathcal{K}$ we conclude: If the curve does not leave an obstacle any more, the obstacle encloses the curve.

Proof:

- Starting point free-space
- After the last hit, the curve fully surrounds the obstacle. Any round gives $\pm 2\pi$ to angular counter
- Positive? Compare to last hitpoint: $\mathcal{C}_{\text{half}}$-condition
- $\mathcal{C}_{\text{half}}$-cond.: $\forall h_i, t \in C : P(t) = P(h_i) \Rightarrow \varphi(t) - \varphi(h_i) < \pi$
- Therefore: -2π, Left-Hand-Rule, enclosed!

Correctness proof, sensor errors

Theorem Any curve from $\mathcal{K}$ leaves a labyrinth, if this is possible.

- Starting-point free-space
- Assume: There is a successful path!
- **Lemma**: Has to leave any obstacle after a while!
- **Lemma**: Hit any edges only once!
- Finally the labyrinth will be left!

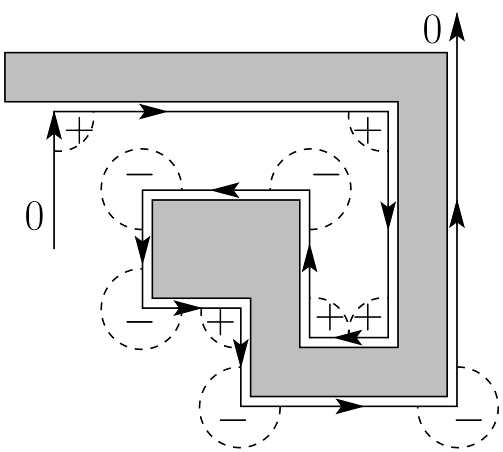
Make use of a compass

Corollary By a compass with deviation less than $\frac{\pi}{2}$, any labyrinth will be left by a pledge like algorithm.

- Free-space angular range $(-\frac{\pi}{2}, +\frac{\pi}{2})$
- Direction deviates at most π!
- $\mathcal{C}_{\text{free}}$-condition holds!
- Along the boundary: Maximal overturn $+\frac{\pi}{2}$
- Free-space minimal $-\frac{\pi}{2}$
- Together: $\forall h_i, t \in C : P(t) = P(h_i) \Rightarrow \varphi(t) - \varphi(h_i) < \pi$ holds!
- $\mathcal{C}_{\text{half}}$-condition holds

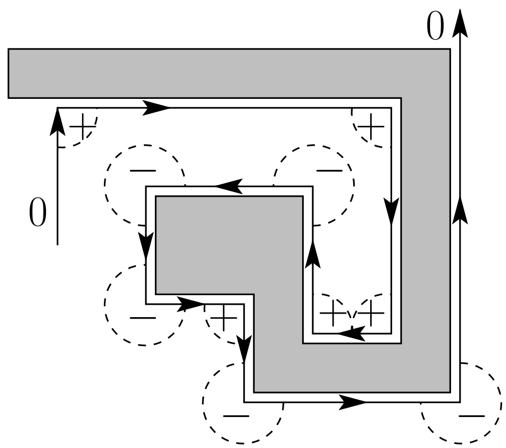
Application! Rectangular scene!

- Scene allows roughly correct counting!
- Axis-parallel edges!
- Right-Turn, Left-Turn, count +1, -1, exact leave!
- Turning detected at the polygons!
- Free-Space: Deviation in $(-\frac{\pi}{2}, +\frac{\pi}{2})$
- Horizontal edge
- Vertical egde can be ignored: Slip along the edge!



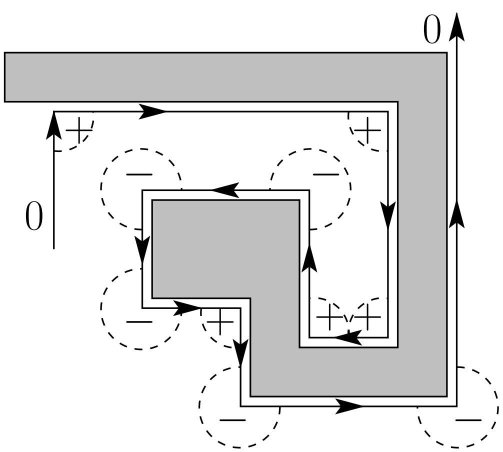
Applications!

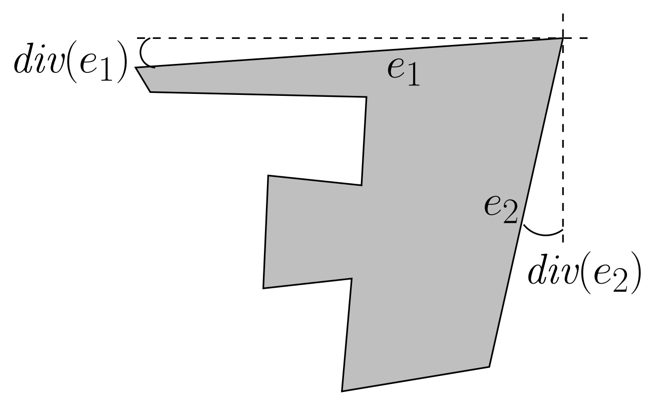
Corollary Axis-parallel environment, deviation in the free space within range $(-\frac{\pi}{2}, +\frac{\pi}{2})$, distinguish horizontal and vertical edges: Escape from a labyrinth!



Deviations from axis-parallel: Pseudo orthogonal

- Small devaitions at the vertices! From global coordinates!
- 1. Condition: Numbers convex vert. $=$ reflex vert. $+ 4$
- Small deviations!
- div(e) : $e = (v, w)$ largest deviation from horizontal/vertical line passing durch v und w
- $\mathrm{div}(P) := \max_{e \in P} \mathrm{div}(e) \leq \delta$, **Def.:** δ-pseudo orthogonal Scene





Szene δ-pseudo orthogonal

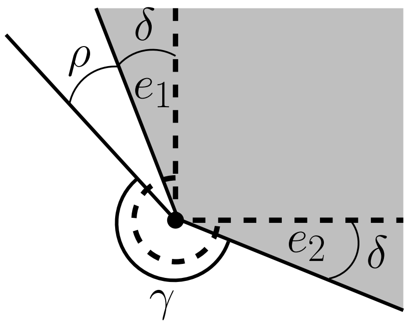
Corollary δ-pseudo-orthogonal scene P. Measure angles with precision ρ s.th. $\delta + \rho < \frac{\pi}{4}$. Deviation in the free space always smaller than $\frac{\pi}{4} - 2\delta - \rho$ from global starting direction. Escape from a labyrinth is guaranteed

1. Distinguish reflex/convex corners: Counting the turns!
2. Max. global deviation of starting direction: Intervall π
3. Distinguish: Horizontal/Vertical

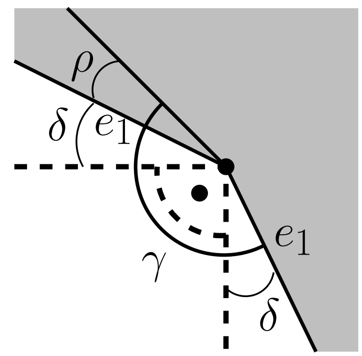
Proof: Blackboard!

δ-pseudo orthogonal scene

- Precision ρ with $\delta + \rho < \frac{\pi}{4}$
- Free-space max. deviation $\frac{\pi}{4} - 2\delta - \rho$
- 1. Distinguish reflex/convex corners: Worst-case



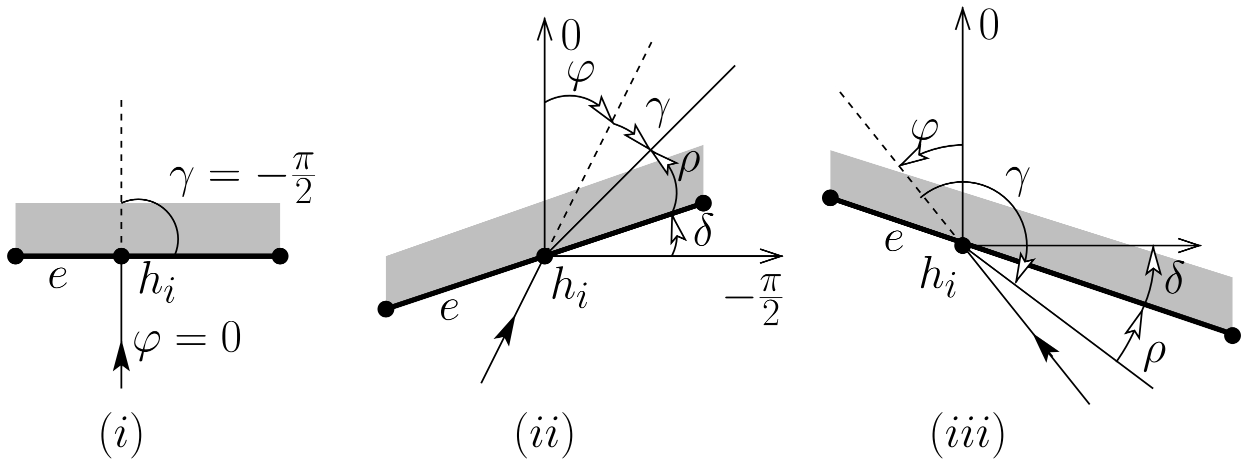
convex vertex



reflex vertex

Szene δ-pseudo orthogonal

- Precision ρ with $\delta + \rho < \frac{\pi}{4}$
- Free-space max. deviation $\frac{\pi}{4} - 2\delta - \rho$
- 3. Horizontal/vertical: Worst-case



Szene δ-pseudo-orthogonal

- Precision ρ with $\delta + \rho < \frac{\pi}{4}$
- Free-Space deviation $\frac{\pi}{4} - 2\delta - \rho$
- 2. Max. global deviation of starting direction: Intervall π
- Leave in $[-\delta, \delta]$
- Deviation for the next hit: $\frac{\pi}{4} - 2\delta - \rho$