Online Motion Planning MA-INF 1314
Bug-Algorithm

Elmar Langetepe
University of Bonn
Repetition: Pledge Algorithm with sensor errors

Pledge-like curve!

**Def.** \( K \) class of curves in \( C_{\text{frei}} \cup C_{\text{halb}} \), with the following conditions:

1. Parameterized curve with turn-angles and position:
   \[
   C(t) = (P(t), \varphi(t)) \text{ mit } P(t) = (X(t), Y(t))
   \]
2. Curve surrounds obstacle by Left-Hand-Rule
3. Leaves point is a vertex of an obstacle
4. \( C_{\text{free}} \)-condition holds:
   \[
   \forall t_1, t_2 \in C : P(t_1), P(t_2) \in C_{\text{free}} \Rightarrow |\varphi(t_1) - \varphi(t_2)| < \pi
   \]
5. \( C_{\text{halb}} \)-condition holds:
   \[
   \forall h_i, t \in C : P(t) = P(h_i) \Rightarrow \varphi(t) - \varphi(h_i) < \pi
   \]
Rep.: Proof correctness

- **Lemma** Curves from $\mathcal{K}$ do not self-intersect.
- **Lemma** Curves from $\mathcal{K}$ hit any edge once.
- **Lemma** For any curve from $\mathcal{K}$: Obstacle will no longer be left, then the curve is enclosed by the obstacle.
- **Theorem** Curves from $\mathcal{K}$ escape, if this is possible.
**Rep.: Applications of the model**

**Corollary** Compass with deviation maximal $\frac{\pi}{2}$ is sufficient for escaping from a labyrinth.

**Corollary** Axis-parallel scene, hold the direction in the range $\left(-\frac{\pi}{2}, +\frac{\pi}{2}\right)$ and distinguish between horizontal and vertical. Escape!
Rep: Pseudo orthogonal

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

![Diagram]

(i) \hspace{1cm} (ii)
Rep: $\delta$-pseudo orthogonal

**Corollary** $\delta$-pseudo-orthogonal scene $P$. Measure angles with precision $\rho$ 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 $\pi$
3. Distinguish: Horizontal/Vertical
Rep.: $\delta$-pseudo orthogonal scene

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

**Diagram:**
- Convex vertex
- Reflex vertex
Szene $\delta$-pseudo orthogonal

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

\[ \varphi = 0 \]
\[ \gamma = -\frac{\pi}{2} \]

\( (i) \) \hspace{2cm} \( (ii) \) \hspace{2cm} \( (iii) \)
Szene $\delta$-pseudo-orthogonal

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

- Searching for a given goal: Navigation
- Polygonal environment: Finite number of polygons
- Touch sensor: Hand-Rules
- Start \( s \), target \( t \), coordinates are given
- Finite storage: I.e. Own coordinates
- BUG Algorithms: Sojourner
Notations

- $|pq|$ distance between $p$ and $q$
- $D := |st|$ distance from start to goal
- $\Pi_S$ path of strategy $S$ from start to goal
- $|\Pi_S|$ length of the path $\Pi_S$
- $\text{UP}_i$ perimeter of obstacle $P_i$
- Actions:
  1. Move into direction of the target
  2. Follow the wall

- Leave-Points $l_i$, Hit-Points $h_i$
BUG1 strategy: Lumelsky/Stepanov
BUG1 strategy: Lumelsky/Stepanov

0. $l_0 := s$, $i := 1$

1. From $l_{i-1}$ move into target direction, until
   (a) Goal is reached: Stop!
   (b) An obstacle is met at $h_i$.

2. Surround the obstacle $O$ in cw order — continuously calculate and store the point $l_i$ on $O$ closest to $t$ —, until
   (a) Goal is reached: Stop!
   (b) $h_i$ is visited again!

3. Move along the shortest path along $O$ to $l_i$.

4. Increment $i$, GOTO 1.
Correctness BUG1 strategy

**Theorem** The strategy BUG1 finds a path from \( s \) to \( t \), if such a path exists.

**Proof:**

- Sequence of Hit- and Leave-Points \( h_i, l_i \)
- \(|st| \geq |h_1t| \geq |l_1t| \ldots \geq |h_kt| \geq |l_kt|\)
Theorem Correctness BUG1 strategy

- Point with smallest distance to $t$: Leave-Point $l_i$
- No free movement to $t \implies$ enclosed
- $l_i \neq l_j$, new obstacle!
- Finitely many obstacles $\implies$ correctness
Path length BUG1 strategy

**Theorem** Let $\Pi_{\text{Bug1}}$ be the path from $s$ to $t$, calculated by the BUG1-strategy. We have: $|\Pi_{\text{Bug1}}| \leq D + \frac{3}{2} \sum_i \text{UP}_i$.

**Proof:**

- Subdivision: Free space path, surrounding
- Surrounding, then shortest path to $l_i$
- $\frac{3}{2} \sum \text{UP}_i$
- Finally: Path $D'$ between the obstacles
**Theorem**  \(|\Pi_{\text{Bug1}}| \leq D + \frac{3}{2} \sum_i \text{UP}_i\).

**Proof:** \(D'\) between the obstacles

\[
D' = |sh_1| + |l_1h_2| + \ldots + |l_{k-1}h_k| + |l_k t|
\]

\[
\leq |sh_1| + |l_1h_2| + \ldots + |l_{k-1}h_k| + |h_k t|
\]

\[
= |sh_1| + |l_1h_2| + \ldots + |l_{k-1} t|
\]

\[
\ldots
\]

\[
\leq |sh_1| + |l_1 t| \leq |sh_1| + |h_1 t| = |st| = D
\]
Lower bound?

- Show: Bug1 is $\frac{3}{2}$-competitive
- Surround the obstacles along the path
- **Corollary** Bug1 is $\frac{3}{2}$-competitive
- Adversary strategy for the model
- Actions:
  1. Move into direction
  2. Follow the wall
- Leave-Points $l_i$, Hit-Points $h_i$
Lower bound

**Theorem** For any strategy $S$ (due to the action-model), and for any $K > 0$, there exist a strategy with arbitrary $D > 0$, such that for any $\delta > 0$: $|\Pi_S| \geq K \geq D + \sum \text{UP}_i - \delta$.

Arbitrarily large path!
\[ |\Pi_S| \geq K \geq D + \sum \text{UP}_i - \delta. \]

- Virtual horse-shoe, Width $2W$, Thickness $\epsilon \ll \delta$, Length $L$, Distance $D$

- Virtual gets precise: Touch the wall!

- For any strategy $S$

![Diagram of virtual horse-shoe and its components](attachment:image.png)
\[ |\Pi_S| \geq K \geq D + \sum \upPi_i - \delta. \]

- Idea: \[ D + W - \sqrt{D^2 + W^2} \leq \delta/2 \] and
\[ L + W - \sqrt{L^2 + W^2} \leq \delta/2, \] \( L, W \) large enough!

- \[ |\Pi_S| \geq \sqrt{L^2 + W^2} + L + \sqrt{D^2 + W^2} \geq D + W + L + W - \delta = D + 2(L + W) - \delta \]
\[ |\Pi_S| \geq K \geq D + \sum \text{UP}_i - \delta. \]

- Problem: Left and right part! Peri. \(4(L + W)\)
- Inside horse-shoe: \( |\Pi_{I_1}| \geq \sum \frac{1}{2} \text{UP}_i \) non-overlapping
- \( |\Pi_{I_2}| \geq \sum \text{UP}_j \) overlapping, \( r_j \) path back
- Outside horse-shoe: \( |\Pi_A| \geq L + C \) with \( C = \sqrt{D^2 + W^2} \)
- \( L_{A_1} \geq \sum \frac{1}{2} \text{UP}_i \) for non-overlapping
- Altogether: \( |\Pi_S| \geq \sqrt{D^2 + W^2} + \sum \text{UP}_i - 2n\epsilon \)
- \( n \leq \frac{2L}{\delta}, \epsilon \leq \delta^2/(4L) \) gives \( 2n\epsilon \leq \delta/2 \)
- \( |\Pi_S| \leq D + W + \sum \text{UP}_i - \delta \)
BUG2 strategy

Line $G$ passing $st$, target direction, surround obstacle, shortest curr. distance on $G$, move to target
BUG2 strategy

0. \( l_0 := s, \ j := 1 \)

1. From \( l_{j-1} \) move toward target, until

(a) Goal is reached: Stop!
(b) Obstacle is met at \( h_j \).

2. Surround obstacle cw order, until

(a) Goal is reached: Stop!
(b) Line \( G \) passing \( st \) is visited at \( q, \ |qt| < |h_jt| \) and \( qt \) locally free for a move

\( l_j := q, \ j := j + 1 \) and GOTO 1.

(c) \( h_j \) is reached again, no point \( q \) of case b) was found.

Reaching the goal is impossible.
BUG2 strategy: Analysis

- Structural properties
- Correctness and performance
- **Lemma** Bug2 visits finitely many obstacles

Proof, by precondition for the scene!
**BUG2 strategy: Property**

**Lemma** Let \( n_i \) denote the number of intersections between \( G \) (line passing \( st \)) and the obstacle \( P_i \). Any boundary point of \( P_i \) is visited at most \( \frac{n_i}{2} \) time.

- Bug2 defines pairs \((h_j, l_j)\) of hit- and leave points
- Jumping cond.: \( |h_j t| > |l_j t| > |h_{j+1} t| \)
- Any intersection with \( P_i \) is only once a leave or a hit point
- Meet current hit point \( \Rightarrow \) Stop

![Diagram showing Bug2 strategy with points labeled s, h_1, p_1, p_2, l_1, l_2, h_2, l_3, p_3, t, and lines connecting these points.](image-url)
Bug2 visits boundary points max $\frac{n_i}{2}$ times

- Pairs $(h_j, l_j)$ of hit-leave points
- $\frac{n_i}{2}$ pairs $(h_j, l_j)$
- Only then a surrounding is started
- Point on the boundary only $\frac{n_i}{2}$ times
**Corollary** Bug2 visits the goal, if this is possible.

- Finitely many visits, finitely many surroundings!
- Either goal is found or current hit point is visited again
- Current hit point $\Rightarrow$ no free path from a better point on the boundary. Goal is enclosed!
BUG2 strategy: Performance

Theorem Let $\Pi_{\text{Bug2}}$ denote the path from $s$ to $t$ designed by BUG2. We have $|\Pi_{\text{Bug2}}| \leq D + \sum_i \frac{n_i \text{UP}_i}{2}$. Proof:

- Subdivision: Surroundings, Free path
- $\sum_i \frac{n_i \text{UP}_i}{2}$ follows from the Lemma
- Length $D'$ between obstacles
BUG2 strategy: Performance

- Length $D'$ between obstacles
- Analogously **BUG1 Theorem** $D' \leq D$
- Altogether:

\[ |\Pi_{\text{Bug2}}| \leq D + \sum_{i} \frac{n_i \text{UP}_i}{2}. \]
Compare BUG2 and BUG1

- BUG2 not always better, sometimes worse (Exercise)
- Convex polygons: Optimal
- Many further variants!
- Visibility/Local improvements!
Change I

- Bug1 fully surrounds
- Bug2 avoids, but visits many times
- Change make use of old Leave/Hit Points, **One** order change!
Pseudocode: Change $I$

0. $\ell_0 := s, i := 1$

1. Move from $\ell_{i-1}$ along line passing $st$ toward goal, until

   (a) Goal is reached: Stop!

   (b) Obstacle is met at $h_i$.

2. Surround obstacle cw order, until

   (a) Goal is reached: Stop!

   (b) Line $G$ passing $st$ is visited at $q$, $|qt| < |h_jt|$ and $qt$ locally free for a move, $l_j := q, j := j + 1$ and GOTO 1.
(c) A hit- or leave point $h_j$ or $\ell_j$ with $j < i$ is met. Move back to $h_i$, use ccw order until (a), (b) oder (d) happens.

(d) $h_j$ is reached again, no point $q$ of case b) was found. Reaching the goal is impossible.