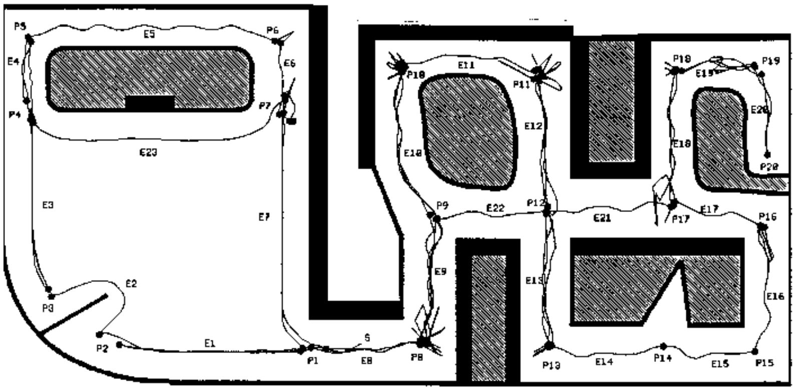
Lecture 18:
Voronoi Graphs and Distinctive States

CS 344R/393R: Robotics
Benjamin Kuipers

Problem with Metrical Maps

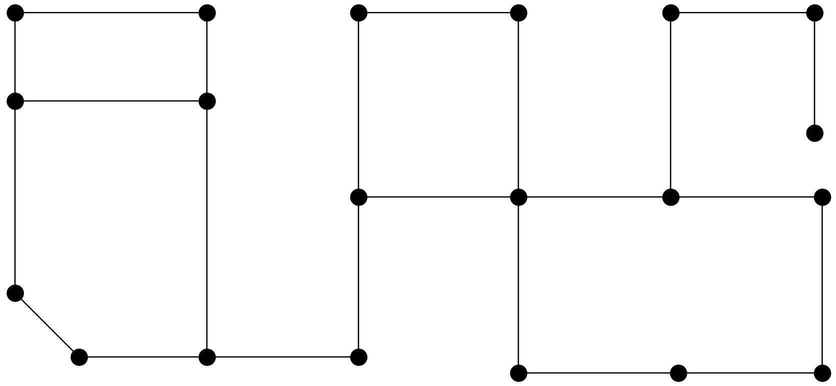
- Metrical maps are nice, but they don't scale.
  - Storage requirements go up with the square of environment diameter and map resolution.
  - Route-finding is hard, because of fine-grained representation.
- Solution: **Topological maps**
  - Abstract the continuous space to a graph of places and edges.
  - Storage is efficient.
  - Graph search is (relatively) inexpensive.

Exploration Defines Important Places and Paths



from Kuipers & Byun, 1991

Abstract the Exploration Pattern to the Topological Map



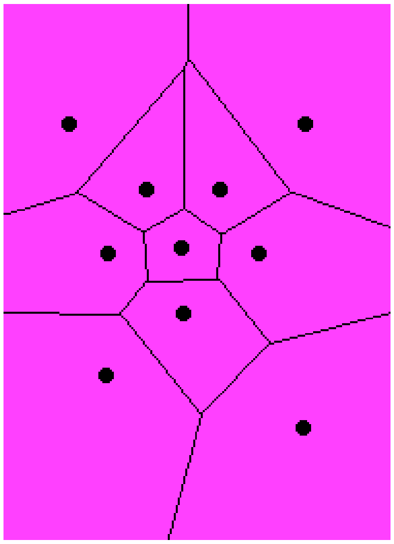
The Topological Map

- The *topological map* is the set of places and edges linking them.
- A *place* is a decision point among edges.
  - It has a *local topology*: cyclic order among edges.
  - It has a *local geometry*: directions of edges.
- An *edge* links two places.
  - A directed edge has a control law for travel.

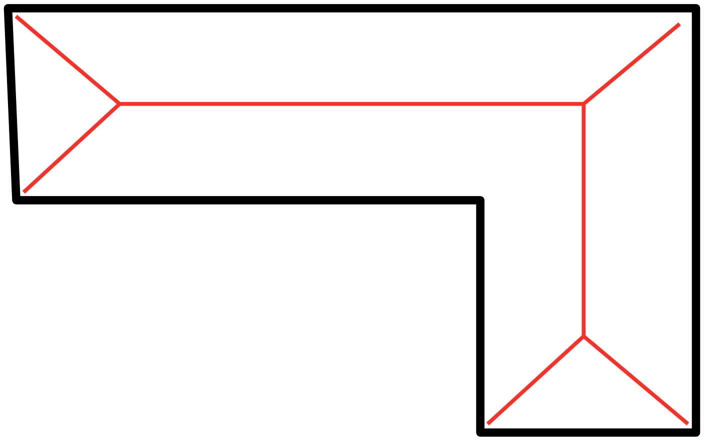
- The *decision-graph abstraction*.

Voronoi Diagram

- Given a discrete set of points in the plane, the Voronoi diagram partitions space into regions closest to each point.
- The Voronoi Graph consists of the region boundaries.



Voronoi Graph of a Robot Environment



Voronoi Graph (Medial-Axis Transform)

- Given a set P of points, find the set of points that have *more than one* closest point in P.
  - *Voronoi Edge*: points equidistant from exactly two boundary points.
  - *Voronoi Node*: points equidistant from three or more boundary points.
- The edges and nodes together make a graph.

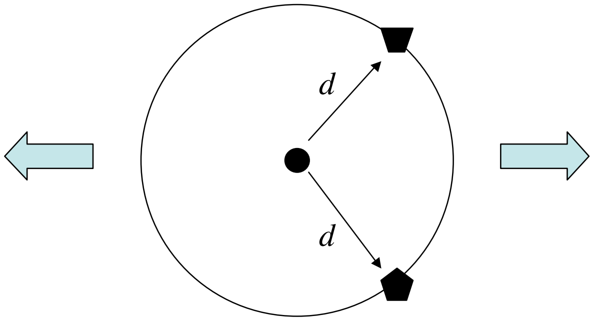
The “Voronoi Robot”

- Imagine a point robot that senses a range image surrounding it.
  - – Distance d to nearest object(s).
  - – Direction(s) to them: $\theta_1 \ldots \theta_k$
- Motion control law: *Follow-the-midline*
  - – When exactly two nearest objects.
  - – Move in direction $\phi = (\theta_1 + \theta_2)/2$ or $\phi + \pi$
- Define a *place* when there are three or more nearest objects.

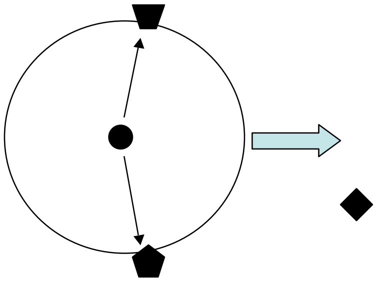
Range Sensing for Voronoi Robot

- Use local minima in the range image.
  - – We usually observe closest objects.
  - – Local minima are likely to be perpendicular reflections of a sonar wave.
- d_{max} = offset distance for wall-following.
  - – (We’ll discuss this extension later.)

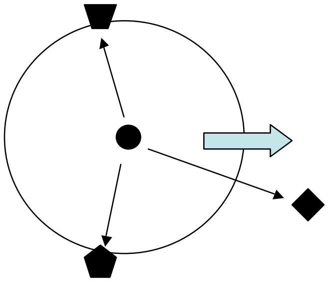
The Voronoi Robot in Motion Along an Edge (Medial Axis)



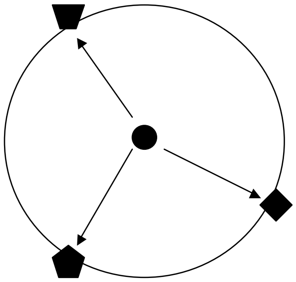
Moving Along a Voronoi Edge



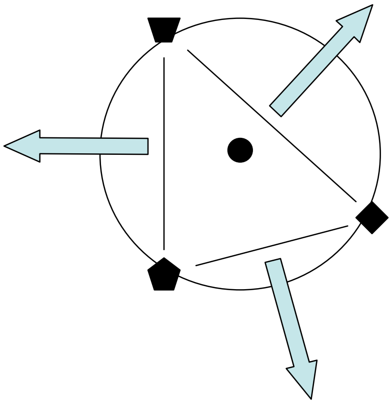
Detect a Third Object



Stop at the Voronoi Node
Define a Place



Describe the Local Geometry of the Place Neighborhood



Voronoi Robot Control Laws

- **Travel Action**
- **Hill-Climbing**
- **Turn Action**

Travel Actions

- Define a PD controller.

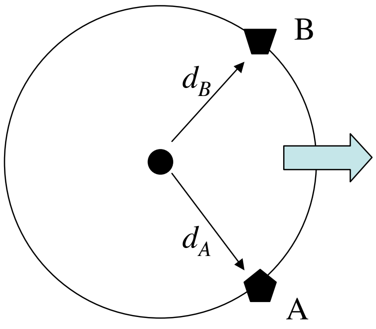
$$\omega = \dot{\theta} = -k_1 e - k_2 \dot{e}$$

- Error term:

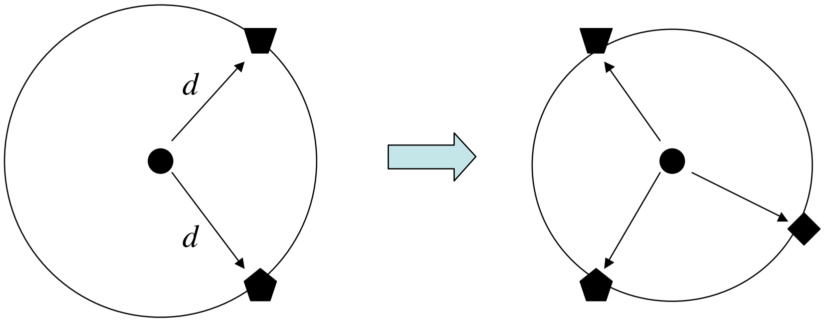
$$e(t) = d_A(t) - d_B(t)$$

$$e(t) = d_A(t) - d_{\max}$$

- *Applicability*:
  - – Nearby objects selected.
- *Termination*:
  - – Stopper object identified.



Hill-Climbing: Move to Equidistance from Three Objects

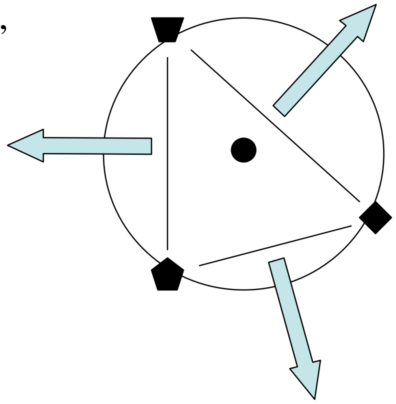


Hill-Climbing Algorithm

- Move, maintaining equal distance $d_A(t)=d_B(t)$ from objects A and B.
- Select object C with distance $d_C(t)$ such that eventually, $d_C(t) = d_A(t) = d_B(t)$.
  - Avoid pathological cases that are never equal, or only equal out of maximum sensor range.
- Same method works for *Follow-right-wall*:
  - maintain $d_A(t) = d_{max}$
  - until $d_B(t) = d_A(t) = d_{max}$.

Turn Actions

- Once at a place,
  - Select an outgoing edge,
  - Rotate to face that edge.
- *Applicability*
  - Located at a Voronoi node.
- *Termination*:
  - Facing along selected edge.

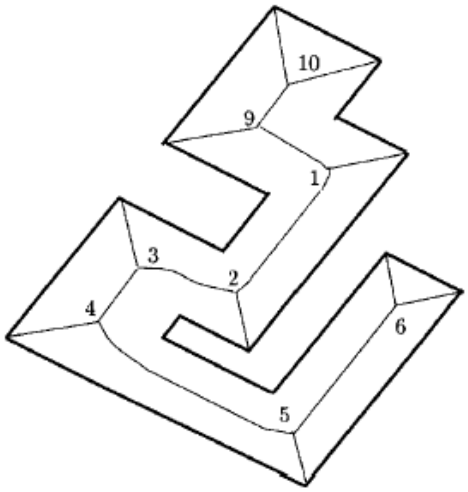


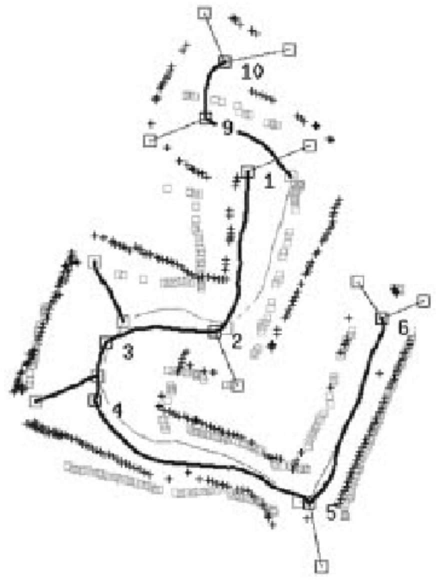
- Three distinctive poses at the same place (or six?)

Explore the Whole Environment

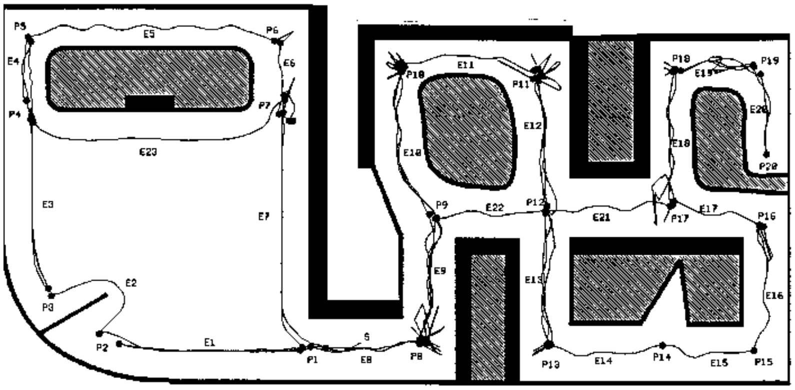
- To start:
  - – Find nearest object (wander, if necessary).
  - – Move away until a second object is found.
  - – *Follow-the-midline* to a third object.
  - – Define an initial place.
- While some place has an unexplored edge,
  - – Follow that edge to the place at the other end.
  - – Q: Closing loops? *Topological ambiguity*.
- Stop when all edges have been explored.

from Choset & Nagatani, 2001

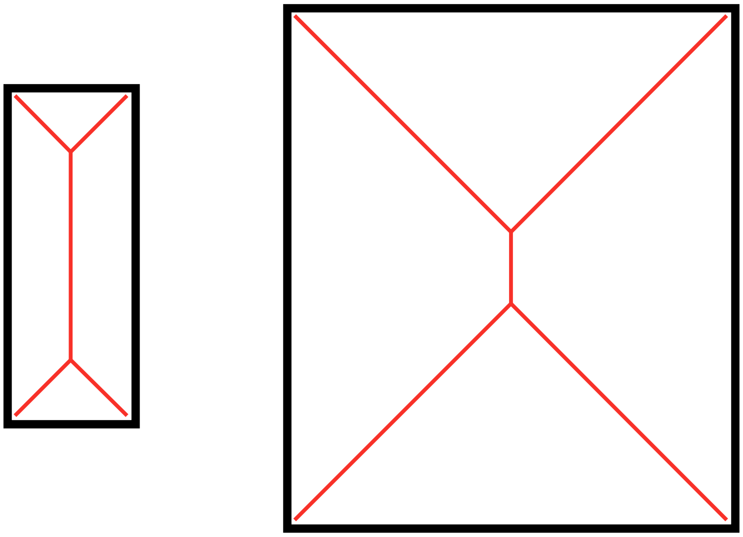




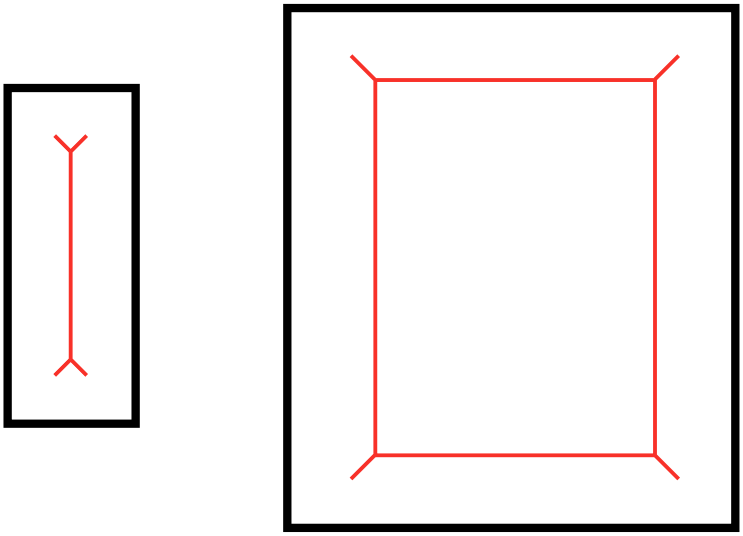
from Kuipers & Byun, 1991



Should Small and Large Spaces Have Similar Models?



Scale is a Relevant Distinction

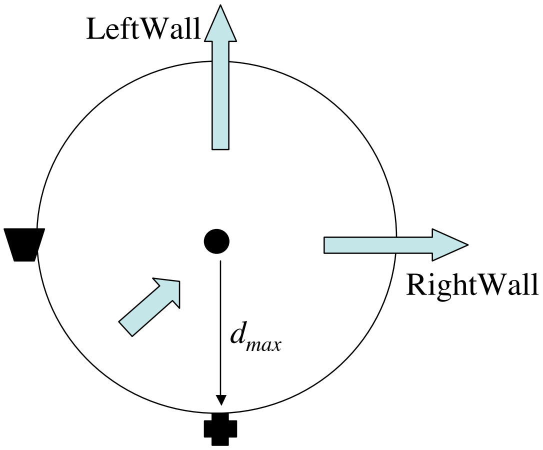


Generalize the Voronoi Robot

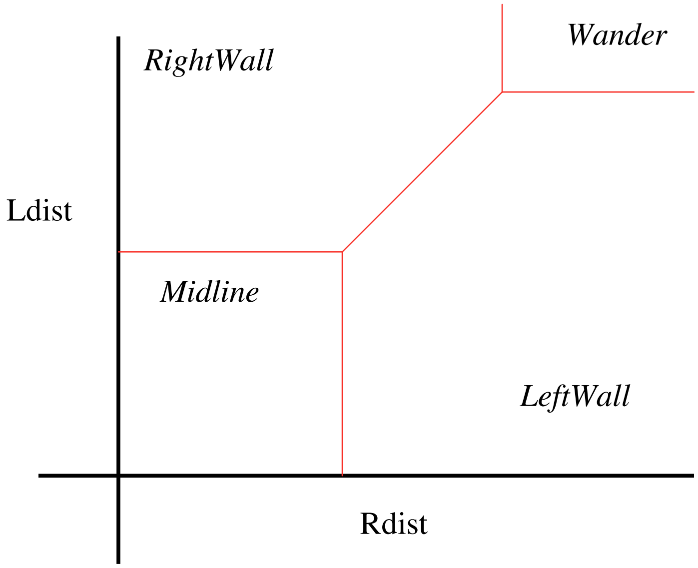
Make its sensors more like a real robot.

- Lower bound on *d*
  - – Don't go through tiny gaps in a wall.
  - – Don't dive too far into concave angles.
- Upper bound on *d*
  - – Range sensors have max effective range.
  - – Distinguish between large and small spaces.
  - – Add *Follow-left-wall* and *Follow-right-wall* control laws

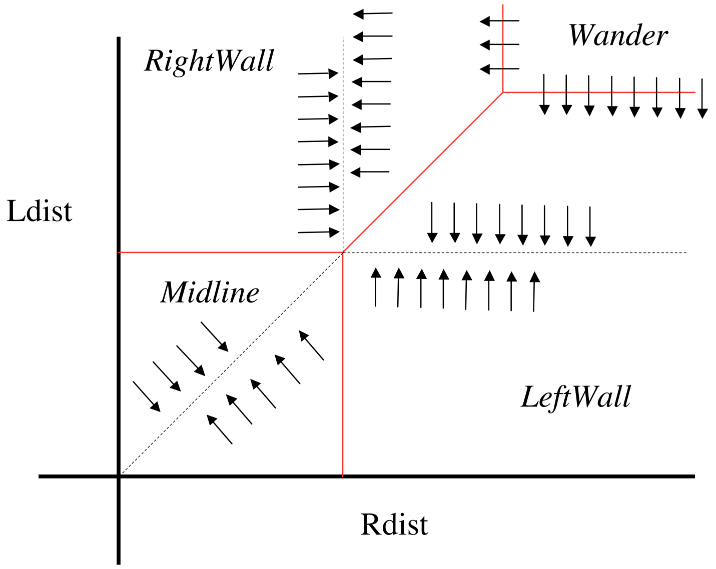
At Maximum Distance, Choose A Wall to Follow



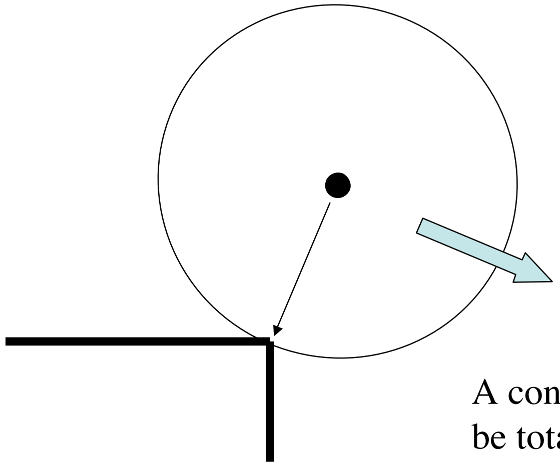
Selecting the Control Law



Selecting the Control Law

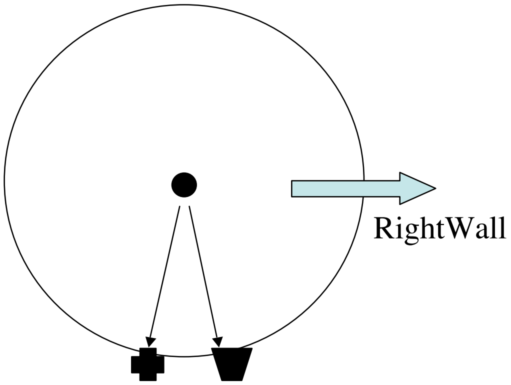


Local Metrical Maps Can Help Avoid Sensor Limitations

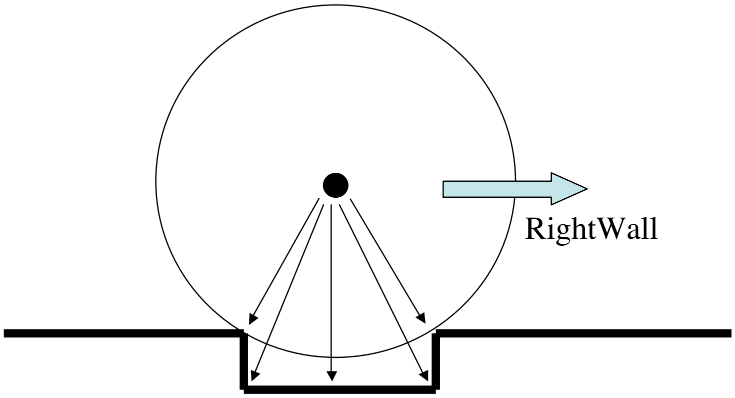


A convex corner may
be totally invisible due
to specular reflections.

Screen Out Small Openings

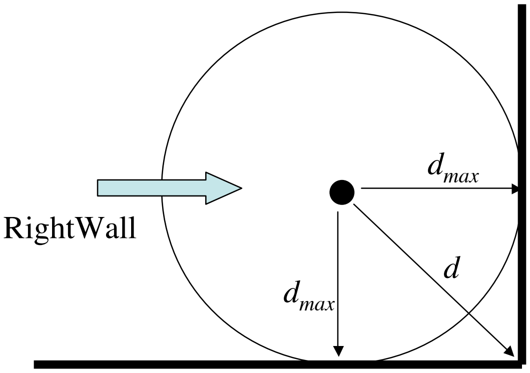


Screen Out Shallow Openings



Identify Right-Angle Spurs

- A predictable configuration. $d \approx \sqrt{2}\, d_{\max}$

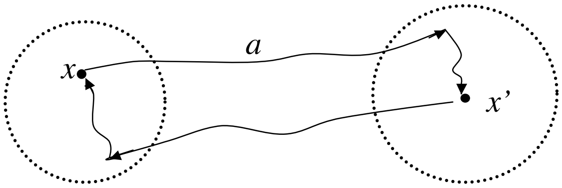


The Topological Map is defined by control laws.

- Places consist of *distinctive states*, which are defined by *hill-climbing* control laws.
  - A HC control law brings the robot to a distinctive state from anywhere in its neighborhood.
- Path segments are defined by *trajectory-following* control laws.
  - A TF control law brings the robot from one distinctive state to the neighborhood of the next

Distinctive States

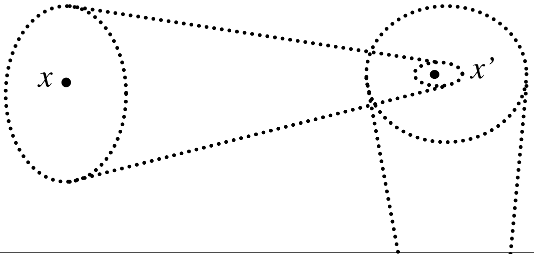
- A *distinctive state* (location plus orientation) is the isolated fixed-point of a hill-climbing control law.



- Hill-climbing to a distinctive state eliminates cumulative position error.
- It also reduces image variability due to pose variation, making place recognition easier.

Deterministic Actions

- Reliable motion abstracts to a causal schema $\langle x,a,x' \rangle$
  - *x* and *x'* are distinctive states (dstates),
  - Action *a* consists of trajectory-following then hill-climbing, leading *reliably* from *x* to *x'*.
- Between distinctive states, actions are *functionally deterministic*.



Two Types of Actions In the Topological Map

- **Travel**:
  - motion from a distinctive state at one place to a distinctive state at another place.
- **Turn**:
  - motion within a place neighborhood from one distinctive state to another.
- We have abstracted from continuous motion to discrete graph transitions.

What have we accomplished?

- We can define a topological map by finding distinctive places (and distinctive states).
  - The Voronoi graph is a simple way to do this.
- The topological map eliminates moderate amounts of cumulative position error.
  - Provides a deterministic model of motion, even with errors in continuous motion.
- Makes planning more efficient and reliable

Next

- Local metrical maps of place neighborhoods
  - Local geometry
- Building the global topological map
  - Solving the loop-closing problem
- Building global metrical maps
  - Using the topological map as a skeleton