LECTURE 7-1

Metric Path Planning

Introduction to AI Robotics (Sec. 10.4 – 10.6)

Quote of the Week

"Making realistic robots is going to polarize the market, if you will. You will have some people who love it and some people who will really be disturbed."

David Hanson, CNN.com, 11/23/06

ANNOUNCEMENTS

- Lab 7 due on *Tuesday, 4/27/10*
- Quiz 13 on Sec. 10.4 – 10.6, Lec. 7-1 on *Thursday, 4/29/10*
- Lab 7 memo and code is due on Angel by midnight on *Thursday, 4/29/10*

OBJECTIVES

Upon completion of this lecture the student should be able to:

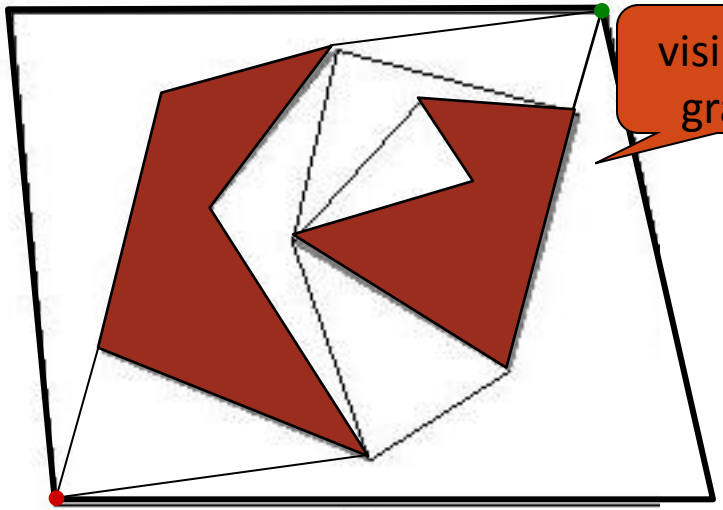
- Define Cspace, path relaxation, digitization bias, subgoal obsession, termination condition
- Explain the difference between graph and wavefront planners
- Represent an indoor environment with a generalized Voronoi graph, a regular grid, or a quadtree, and create a graph suitable for path planning
- Apply wavefront propagation to a regular grid
- Explain the differences between continuous and event-driven replanning

GLOBAL PATH PLANNING

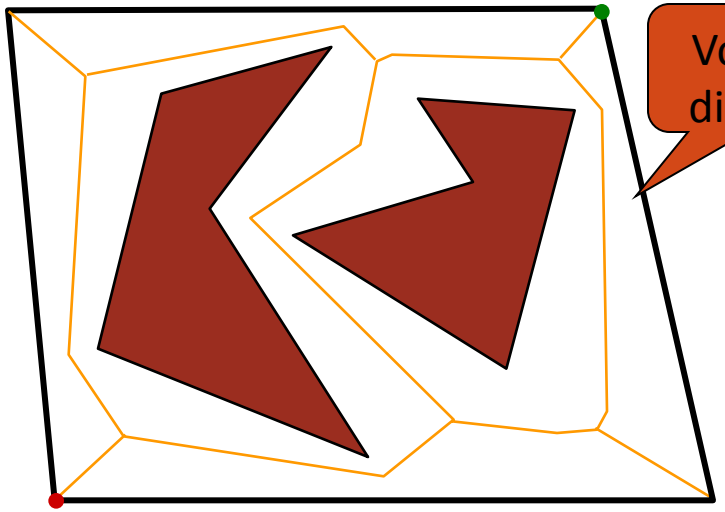
- The robot's environment representation can range from a continuous geometric description to a decomposition-based geometric map or a topological map
- Assumption: there exists a good enough map of the environment for navigation.
- Three general strategies for decomposition
  - ***road map*** - identify a set of routes within the free space
  - ***cell decomposition*** – discriminate between free and occupied cells
  - **potential field** – impose a mathematical function over the space

FULL-KNOWLEDGE PATH PLANNING

Roadmaps

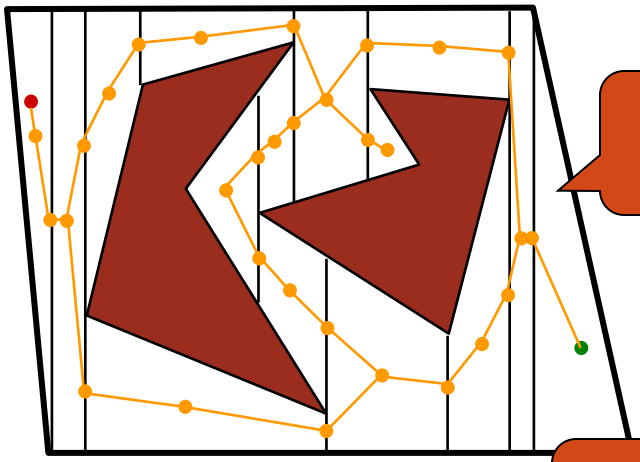


visibility graph



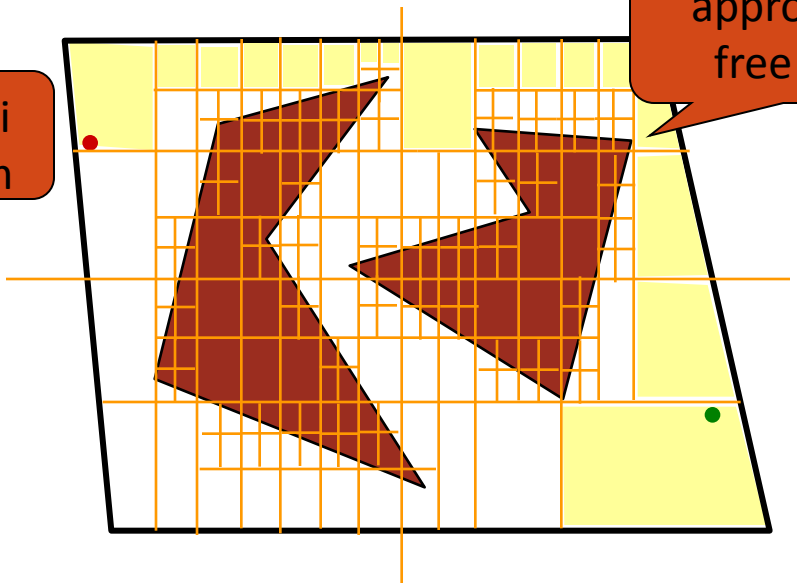
Voronoi diagram

Cell decompositions

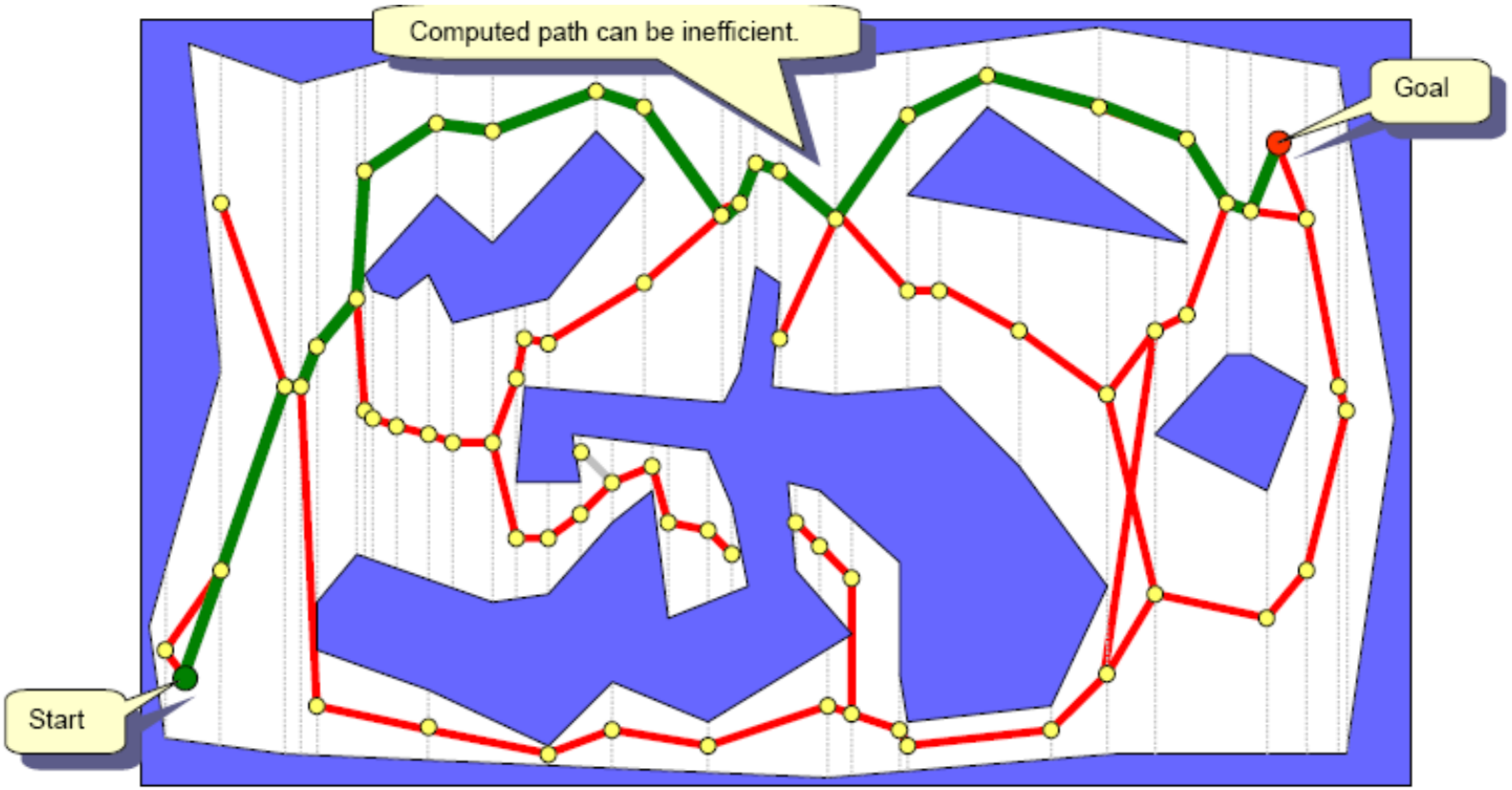


exact free space

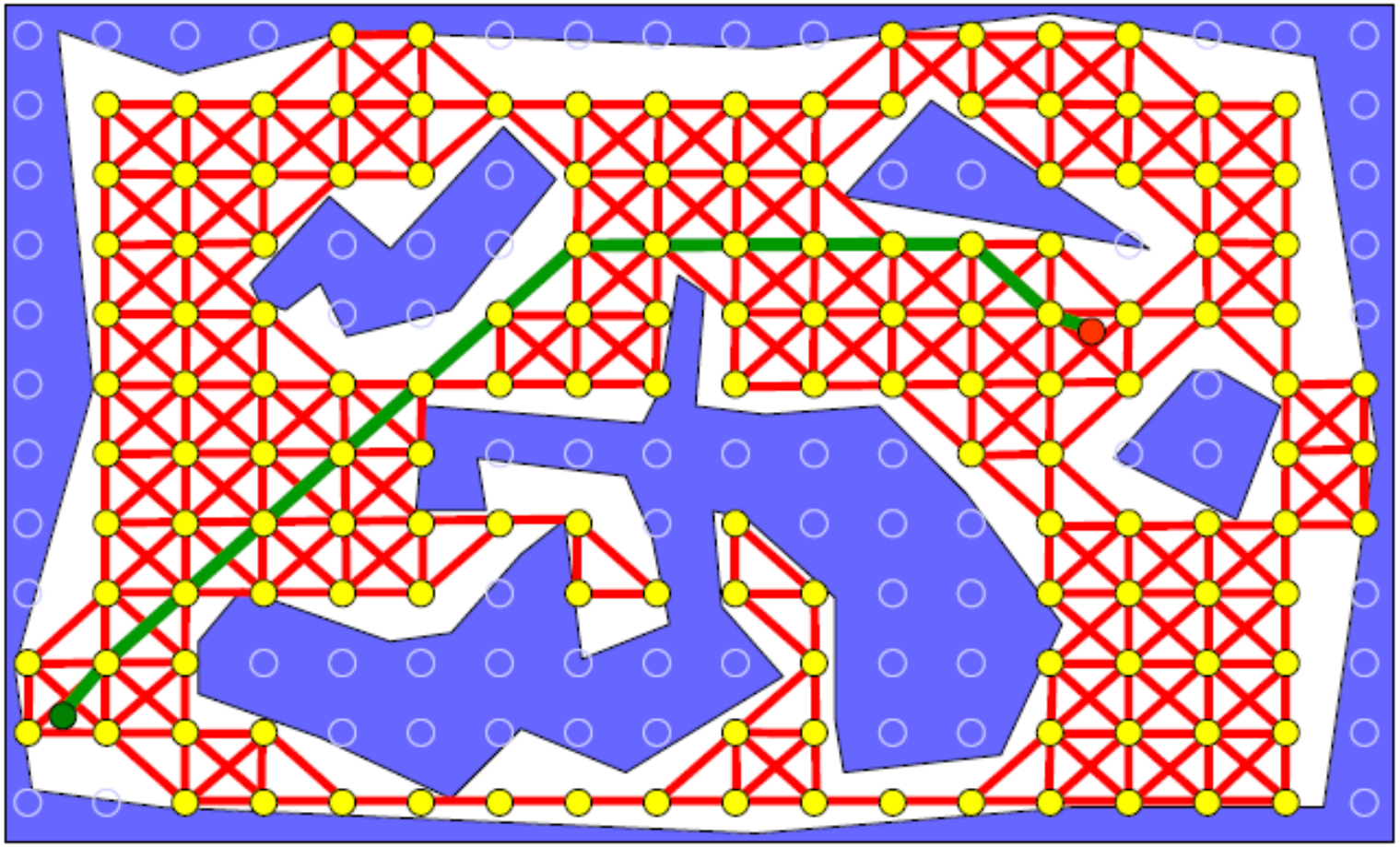
approximate free space



ROAD MAP-BASED PATH PLANNING



ROADMAP-BASED PATH PLANNING

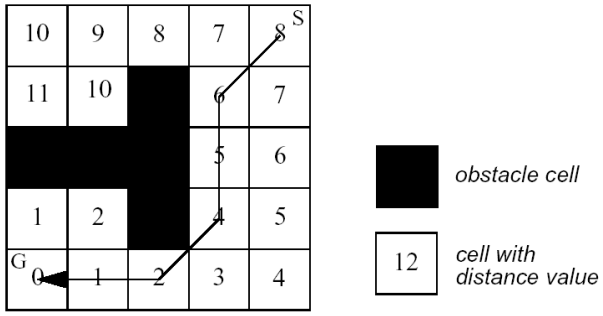


GRAPH BASED PLANNERS

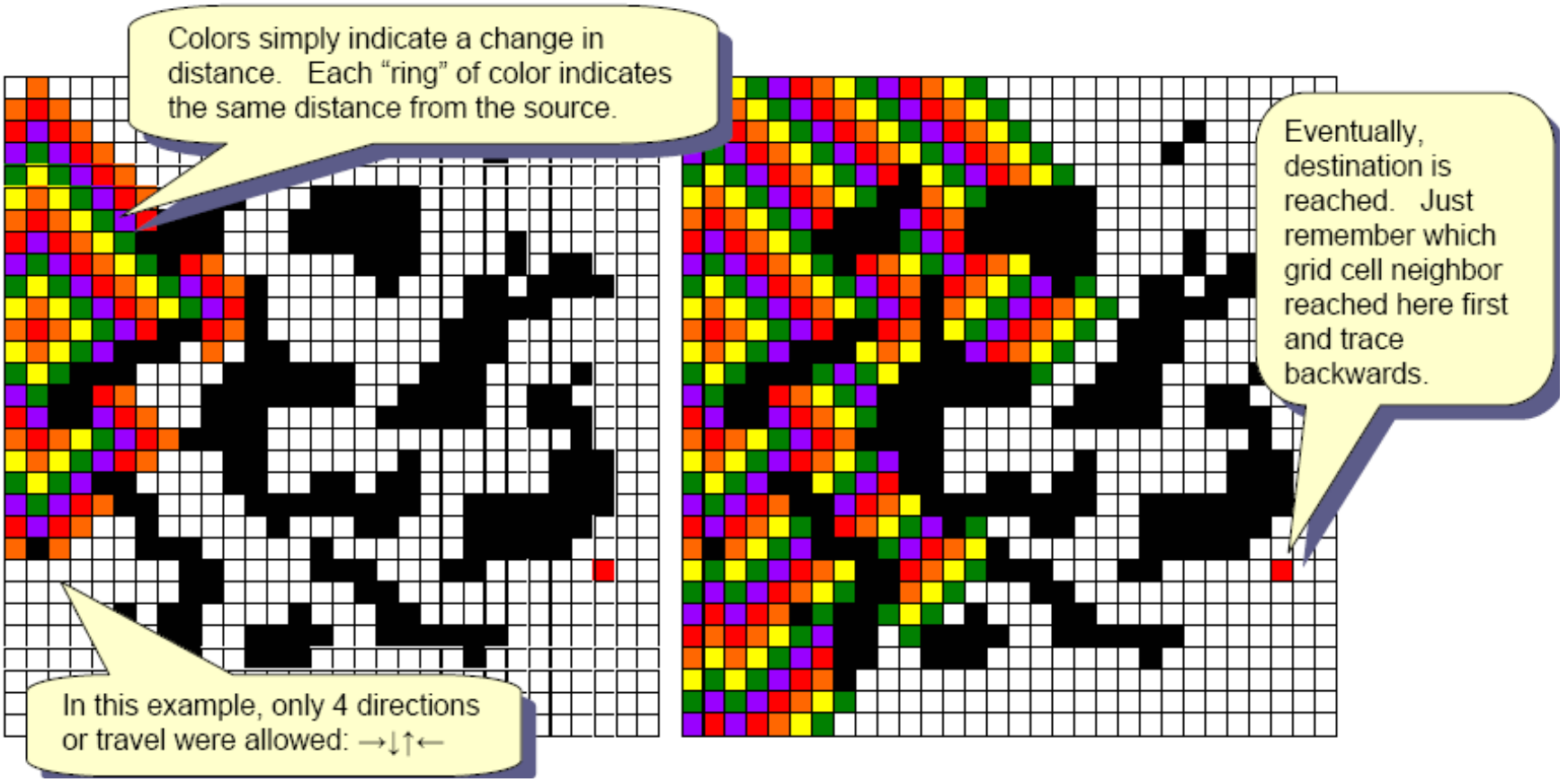
- Cspace representations can be converted to graphs so that the path between the initial node and the goal node can be computed using ***graph search algorithms***
- The search algorithm requires visiting every node which is computationally tractable for a Voronoi diagram but computationally expensive for a regular grid
- The A* search algorithm is the classic method for computing optimal paths for holonomic robots
- A*evaluates each node and the set in front at each point in time and then chooses the best one and throws away the other choices

APPROXIMATE CELL DECOMPOSITION

- **Wavefront expansion** or **grassfire** is an efficient and simple to implement technique for finding routes in fixed-size cell arrays
- employs wavefront expansion from the goal position outward, marking each cell's ***distance to the goal***
- this continues until the wave reaches the initial position
- the planner can then estimate the robot's distance to the goal as well as recover a specific solution trajectory by linking together adjacent cells that are always closer to the goal

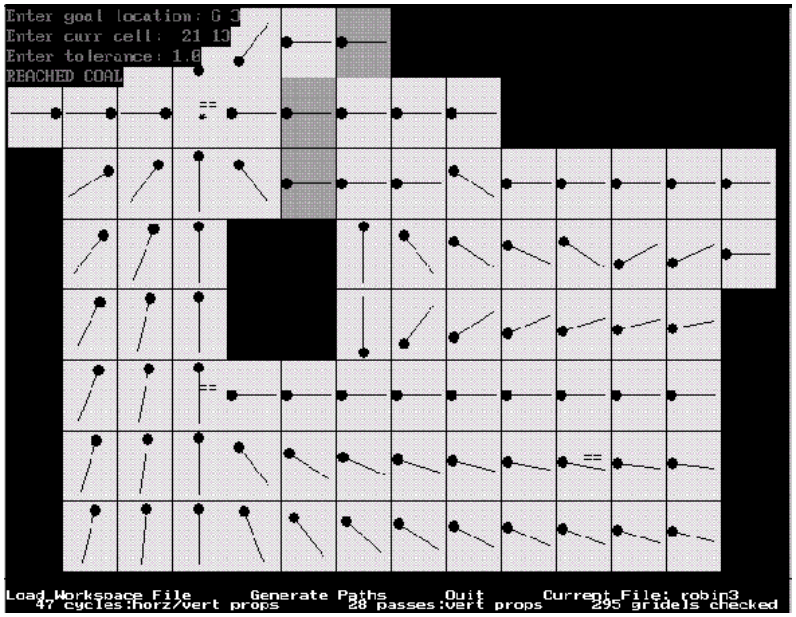


WAVEFRONT PROPAGATION



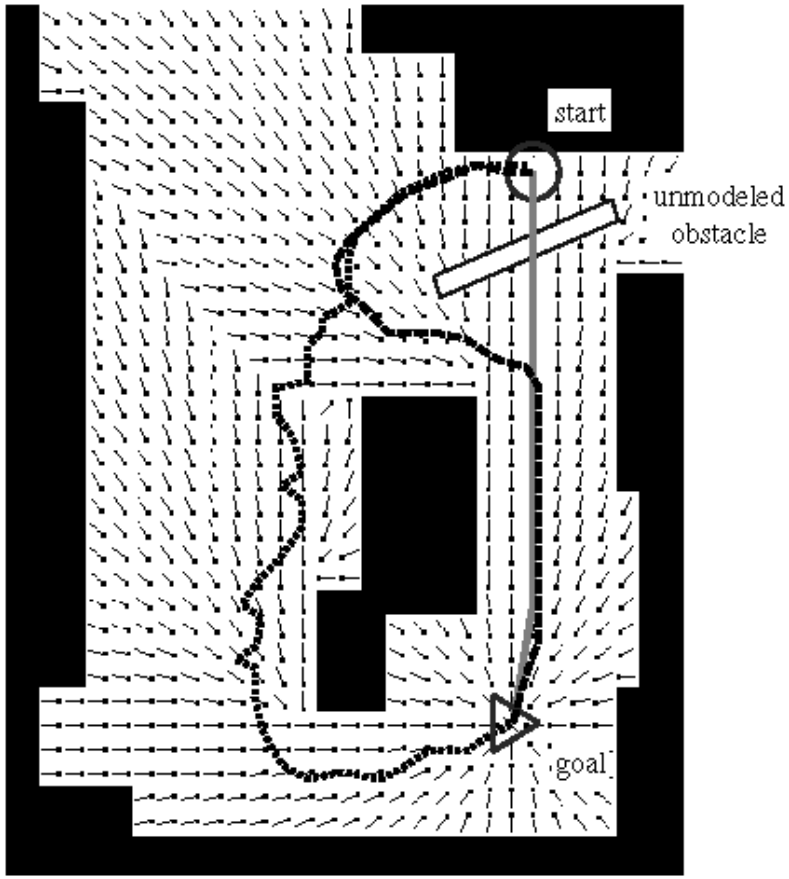
TRULLA

- The result of a wavefront propagation results in a map that looks like a potential field
- The path itself represents what the robot should do as a sensor observation
- Undesirable terrain is shown on the Trulla planner as gray, obstacles in black and open areas in white



INTERLEAVING PATH PLANNING AND REACTIVE EXECUTION

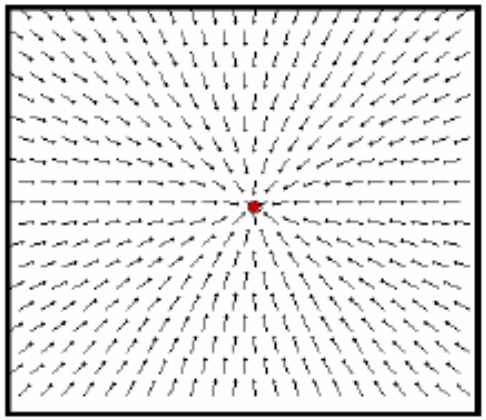
- Most path planning algorithms are employed in a plan once and then reactively execute which is hybrid control
- The problems with reactive execution are
  - Subgoal obsession
  - Lack of opportunistic replanning
- Use D* algorithm to correct for errors (*continuous replanning*)



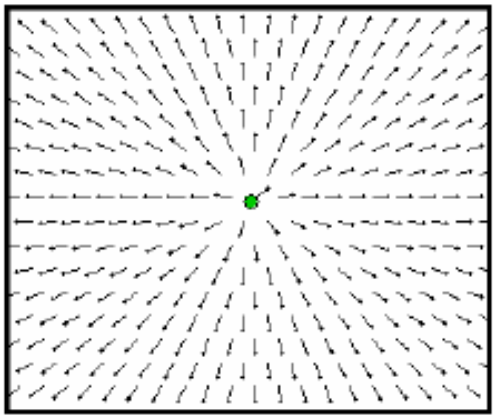
POTENTIAL FIELD PATH PLANNING

- **Potential field path planning** creates a field, or gradient, across the robot's map that directs the robot to the goal position from multiple prior positions
- Robot is treated as a **point under the influence** of an artificial potential field.
  - Generated robot movement is similar to a ball rolling down the hill
  - Goal generates attractive force
  - Obstacles are repulsive forces
  - the superposition of all forces is applied to the robot
  - artificial potential field smoothly guides the robot toward the goal while simultaneously avoiding obstacles

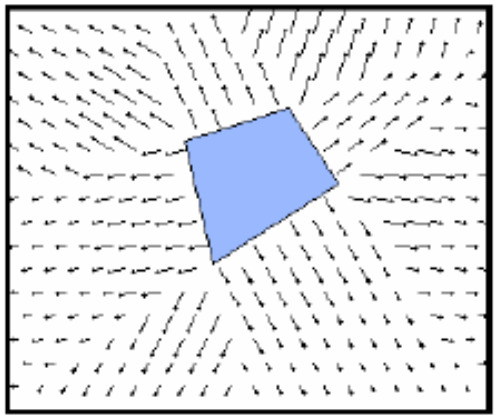
POTENTIAL FIELD PATH PLANNING



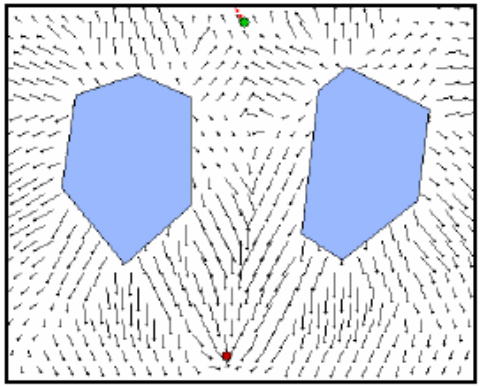
Attract to goal



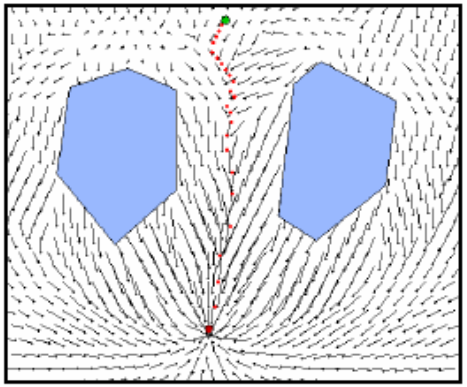
Repel from source



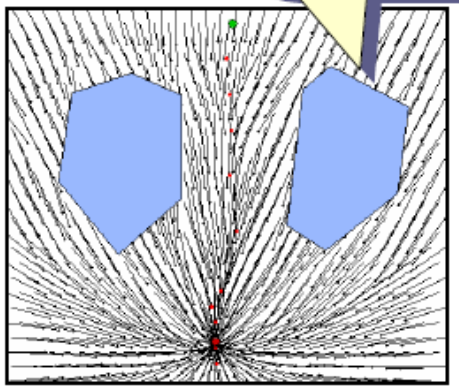
Repel from obstacle



weak goal attraction

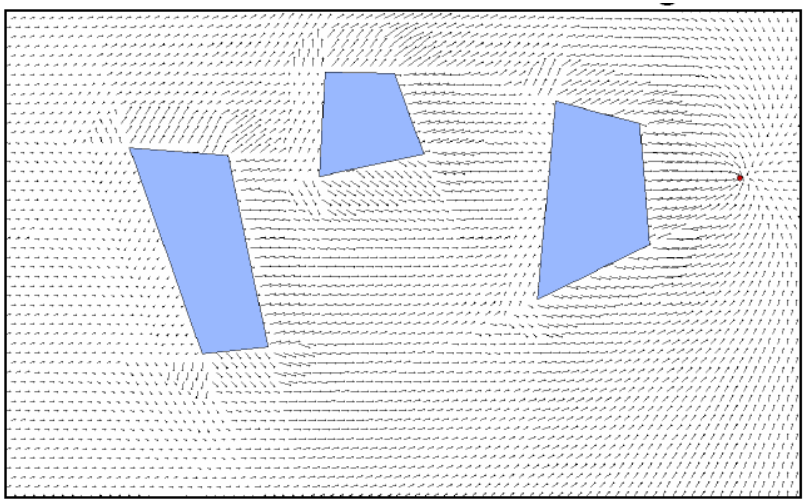


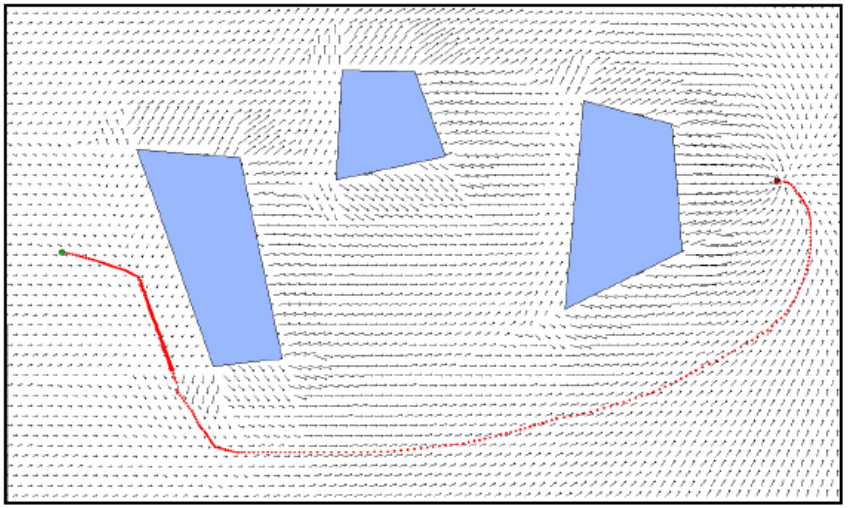
medium goal attraction



strong goal attraction

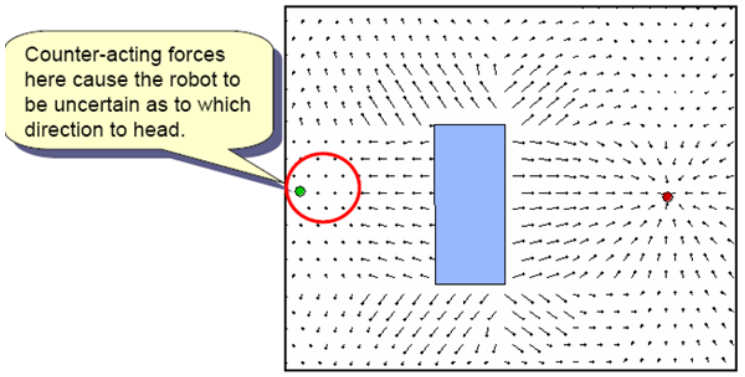
POTENTIAL FIELD PATH PLANNING

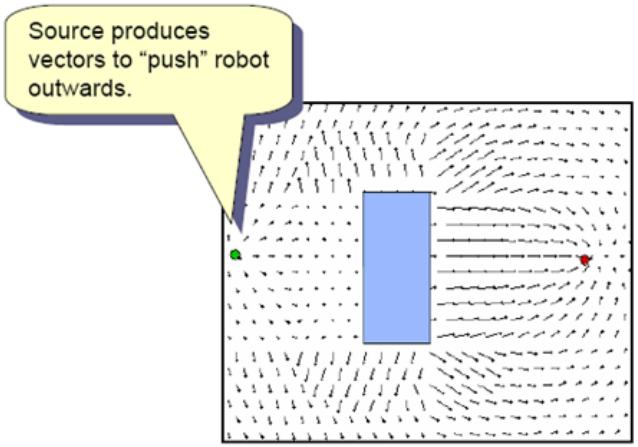




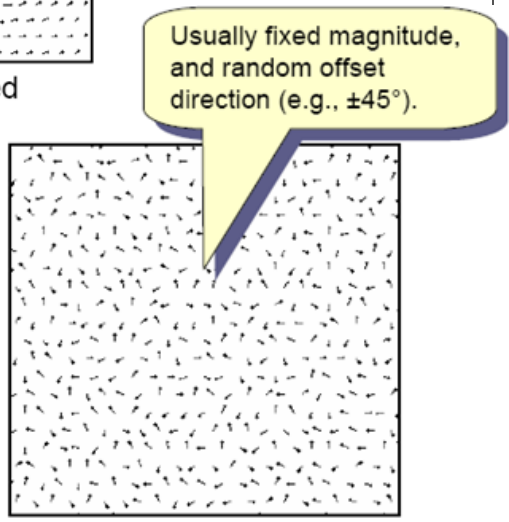
LOCAL MINIMA

- A local minima occurs when the robot gets stuck due to counteracting forces
- To overcome a local minima problem
  - Add a field from the source to push away from it
  - Introduce noise into the environment
  - The outward source may still lead to local minima but noise may overcome this



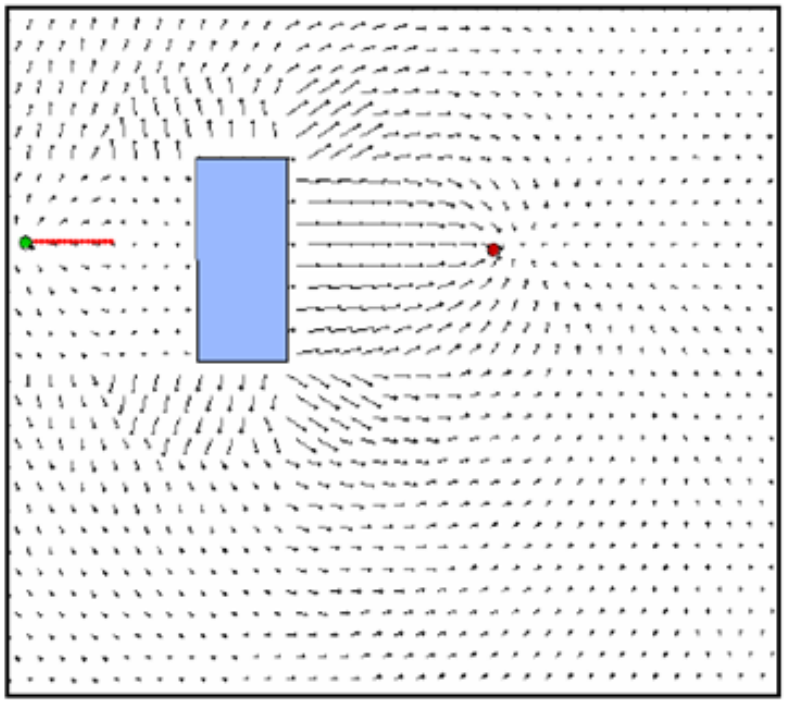


source field added

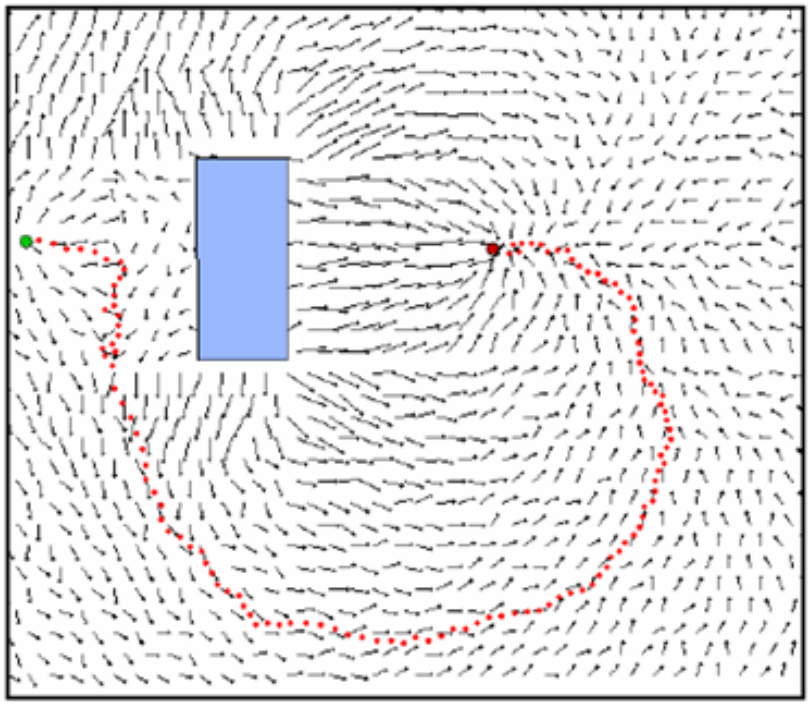


noise (i.e., random vectors)

LOCAL MINIMA



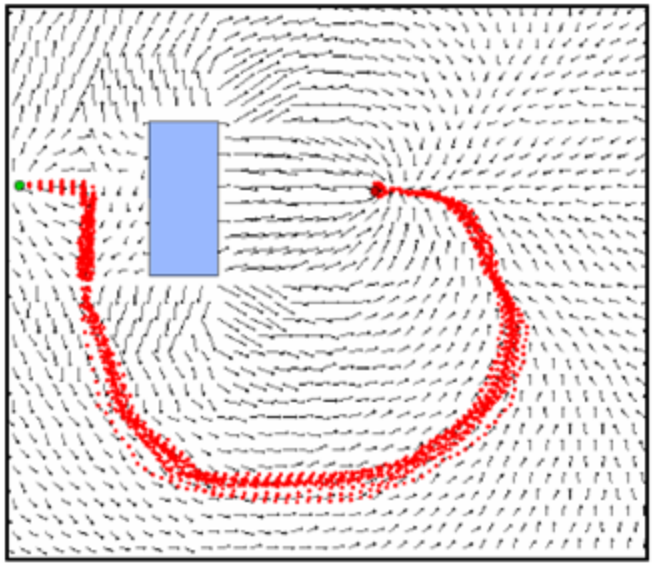
without noise, no path



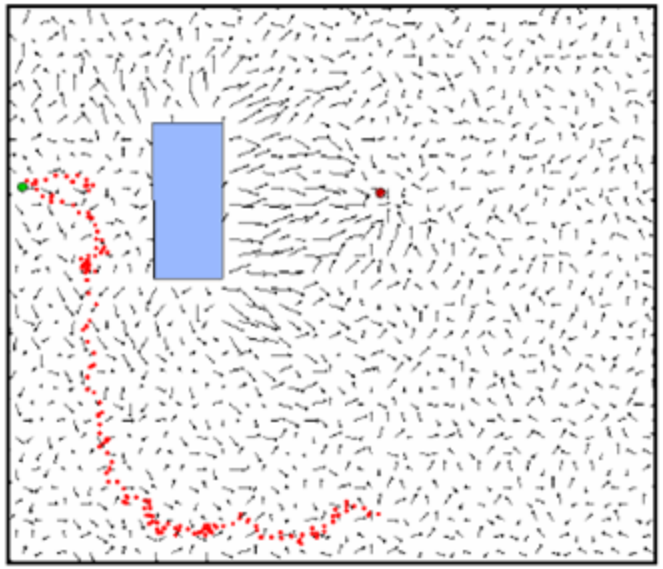
with noise, path found

LOCAL MINIMA – RANDOM NOISE

- The path will vary depending on the random values of the noise vector
- Too much noise will not work and no path will be found

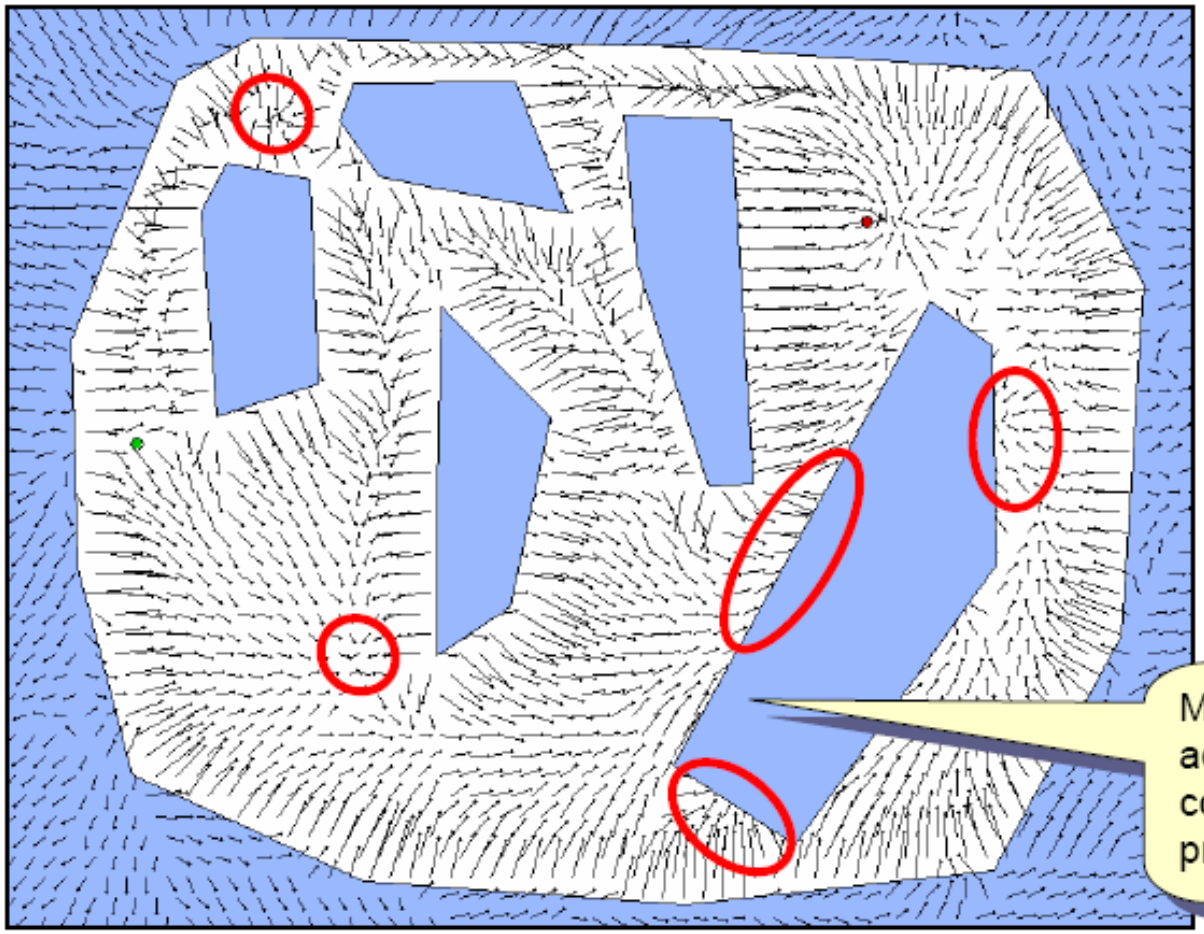


multiple iterations

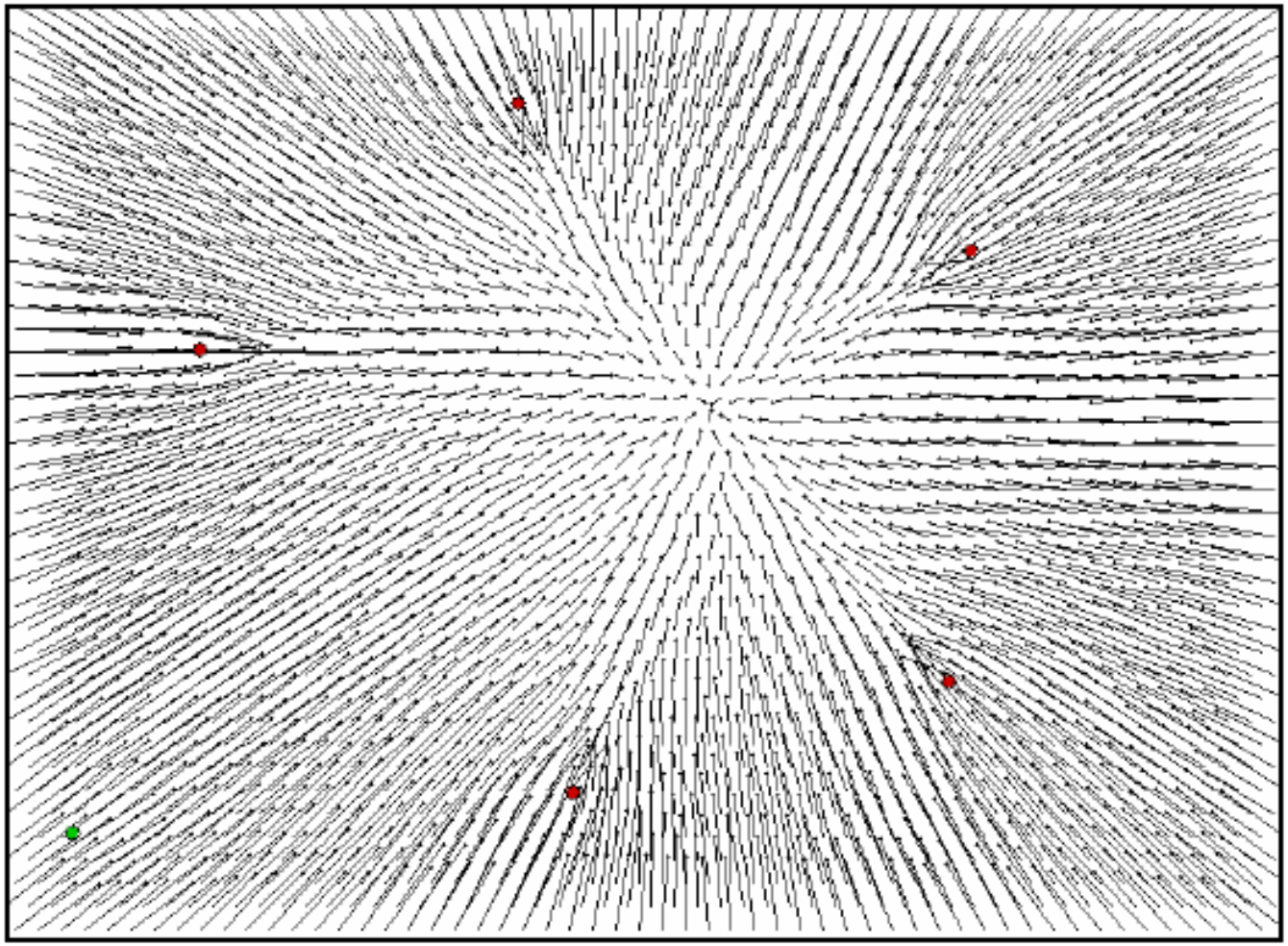


too much noise, no path found

COUNTERACTING FIELDS ENVIRONMENT BOUNDARY AND OBSTACLES

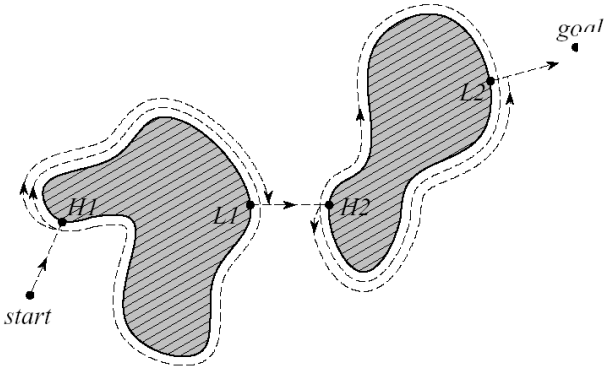


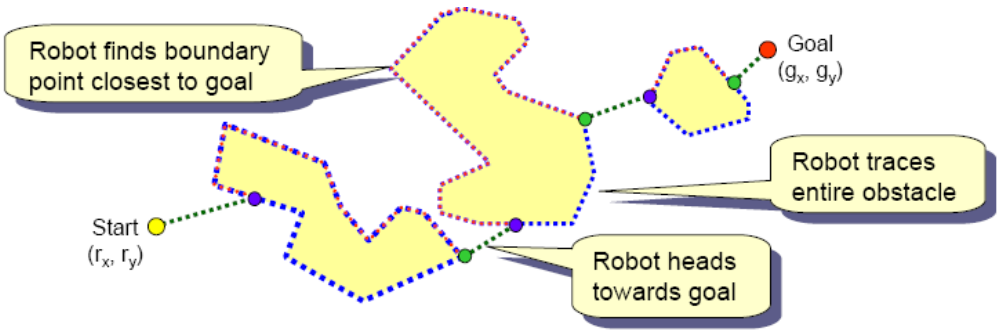
POTENTIAL FIELDS CANNOT HANDLE MULTIPLE GOALS



OBSTACLE AVOIDANCE
BUG ALGORITHM

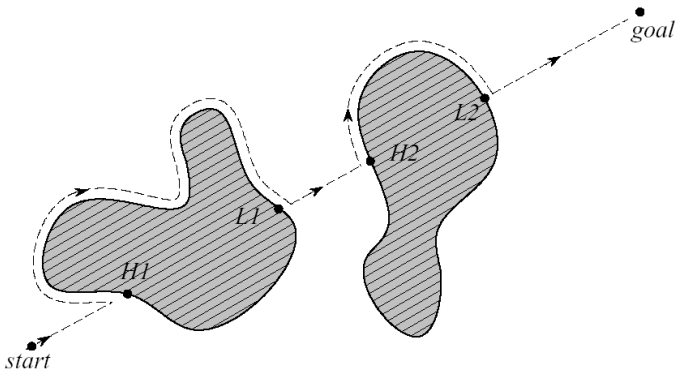
- This is the simplest algorithm
- Follow the contour of each obstacle until it is fully circled before it is left at the point closest to the goal
- Very inefficient but it guarantees that the robot will reach any reachable goal

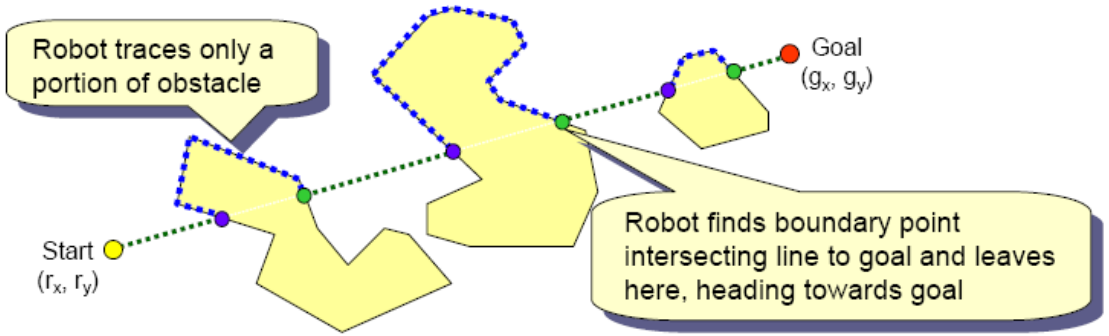




OBSTACLE AVOIDANCE
BUG2 ALGORITHM

- Follows the obstacle always on the left or right side
- Leaves the obstacle if the direct connection between start and goal is crossed
- Has significantly shorter total robot travel





OBSTACLE AVOIDANCE
TANGENT BUG

- The tangent bug adds range sensing and a local environmental representation termed the *local tangent graph (LTG)*
- The *LTG* approaches globally optimal paths