Autonomous agents
Lecture 7, 20160208

Robotic behaviors I. Exploration and navigation (1)

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Today's learning goals

- After this lecture you should be able to
  - Describe different approaches to area coverage
  - Describe, in detail, a method for exploration of an initially unknown arena
  - Describe and implement best-first search.

Exploration

- Two reasons for carrying out exploration of an unknown arena:
  - Physical area coverage: Relevant in cleaning, lawn-moving, various agricultural applications, clearing mine fields etc.
  - Sensory area coverage: Relevant in map-building.

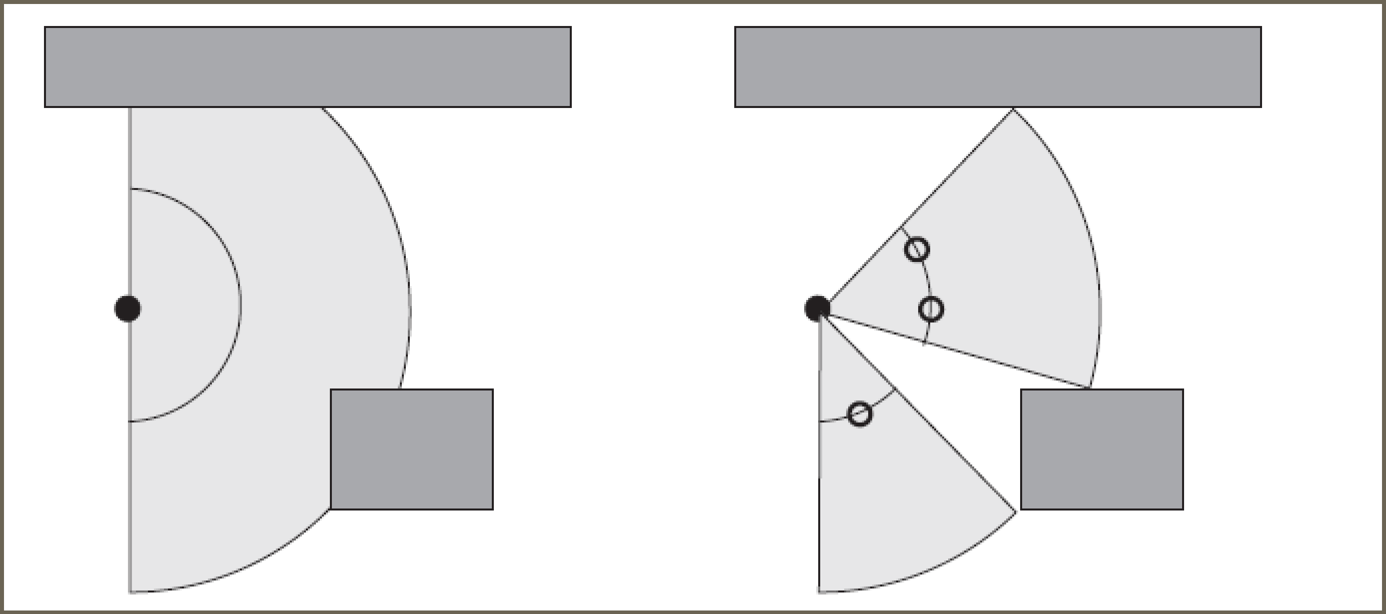
Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Today’s learning goals

- After this lecture you should be able to
  - Describe different approaches to area coverage
  - Describe, in detail, a method for exploration of an initially unknown arena
  - Describe and implement best-first search.

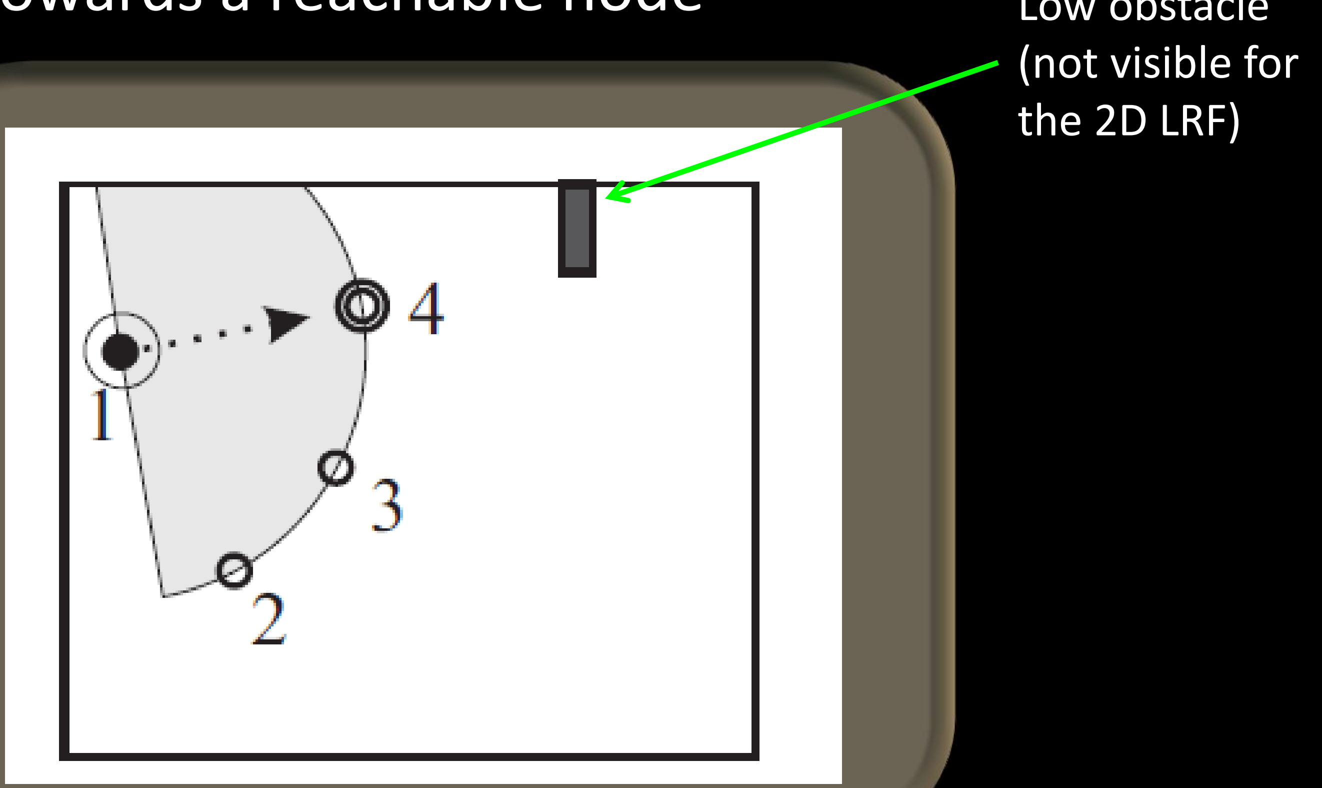
Exploration

- An exploration algorithm based on dynamic node placement (pp. 68-74)



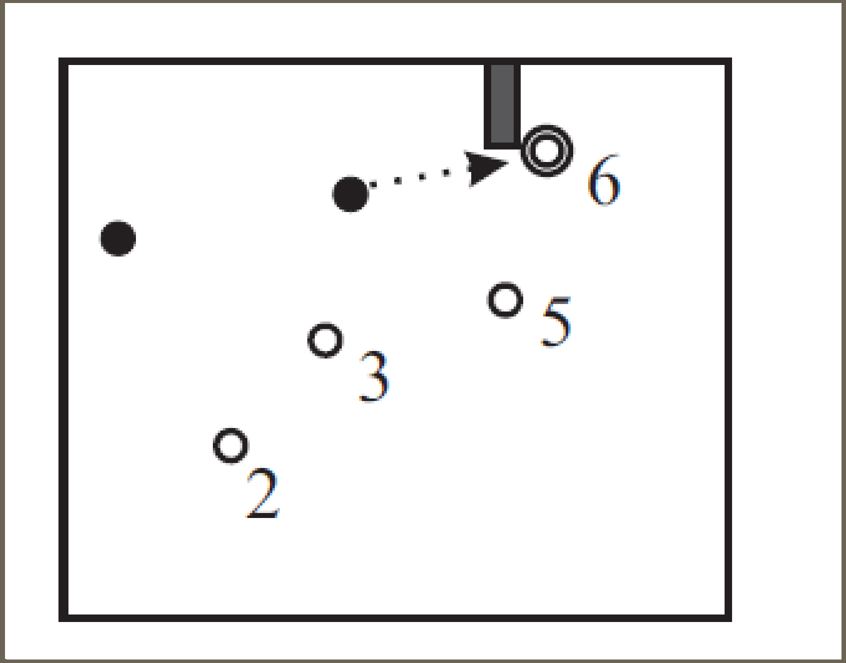
Exploration

- Movement towards a reachable node



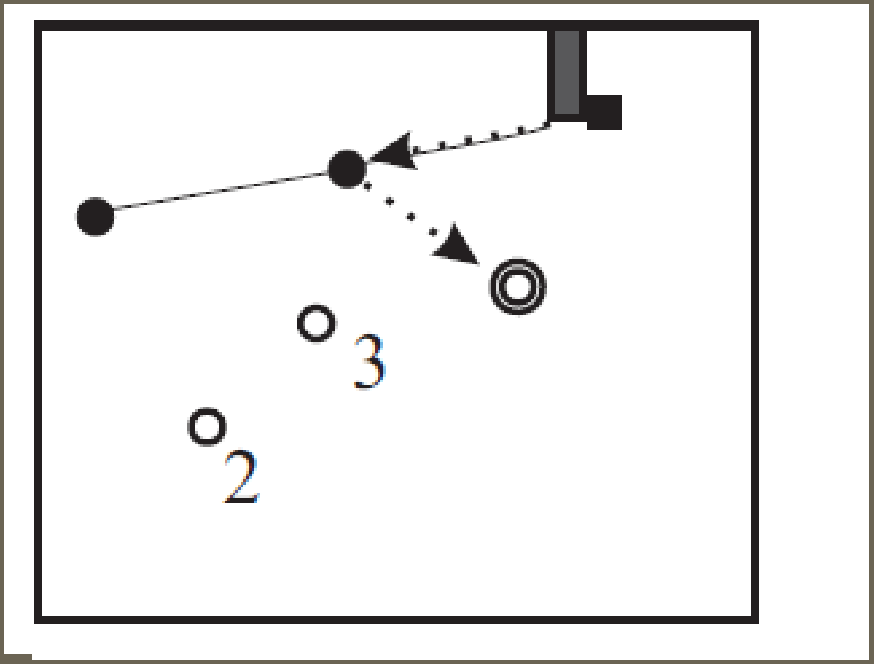
Exploration

- Movement towards an unreachable node (low obstacle).



Exploration

- Proceeding to another node instead:



Exploration

- The exploration algorithm is usually used in connection with map building. Thus, small distance between nodes, so that localization can be carried out accurately (see the lecture next Monday!)
- The minimum allowed distance between nodes makes the algorithm finite .
- Note that the minimum distance requirement is not applied to unreachable nodes!
- In order to make the algorithm robust, one must overcome various practical problems (see the next slide).

Exploration

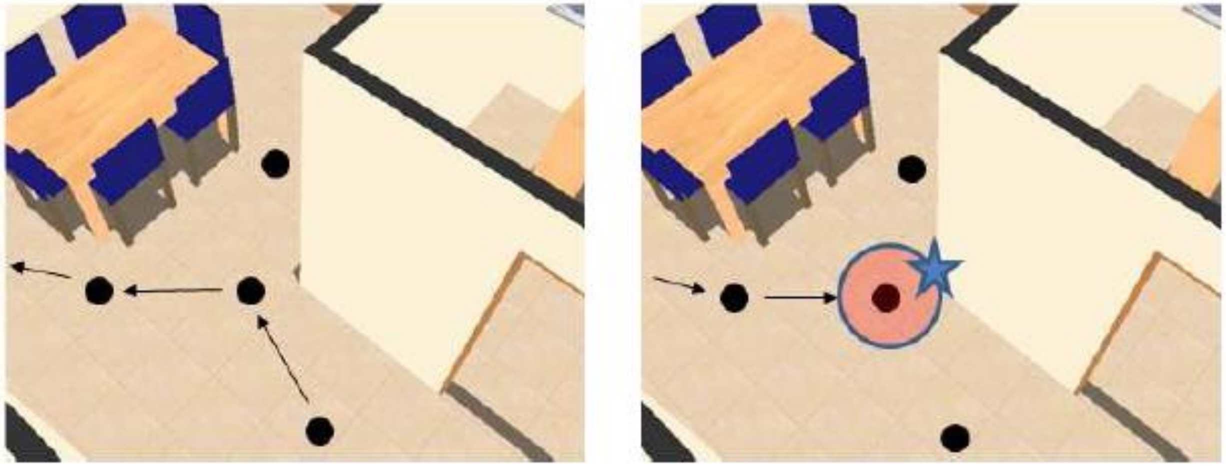


Figure 6.3: *An illustration of a problem that might occur during exploration. Moving in one particular direction (left panel) the robot is able to place and follow the nodes shown. However, upon returning (right panel), the robot may conclude that it will be unable to pass the node near the corner, due to the proximity detection triggered as the robot approaches the node, with the wall right in front of it.*

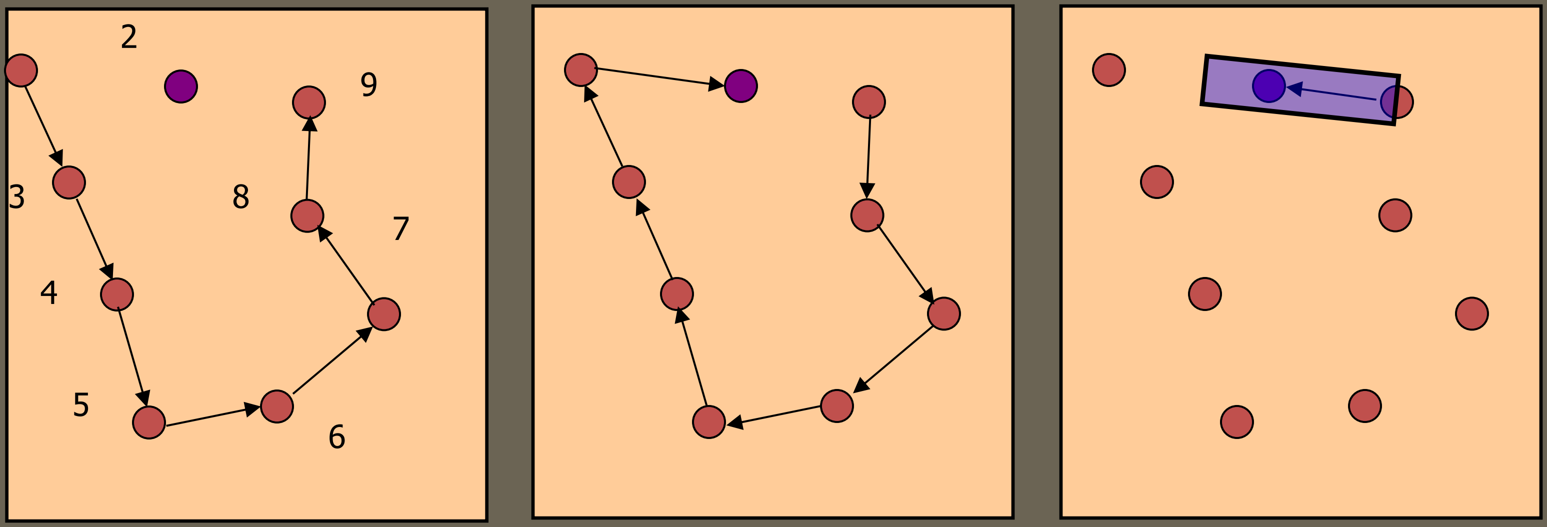
Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Exploration

- The path following nature of the algorithm (as the robot moves between known nodes) contributes to its robustness.
- However, on some (rather rare) occasions, the path distance can be much longer than a direct movement between two points (see the next slide).
- The algorithm contains a method for handling such cases.

Exploration

Direct movement between nodes. Attempted *only* if the distance to the current target node is < L < R (LRF sensor range). Note: low-lying obstacles may result in the target node not being reachable, in which case the path is followed instead.



Robot at node 9: One pending node remaining...

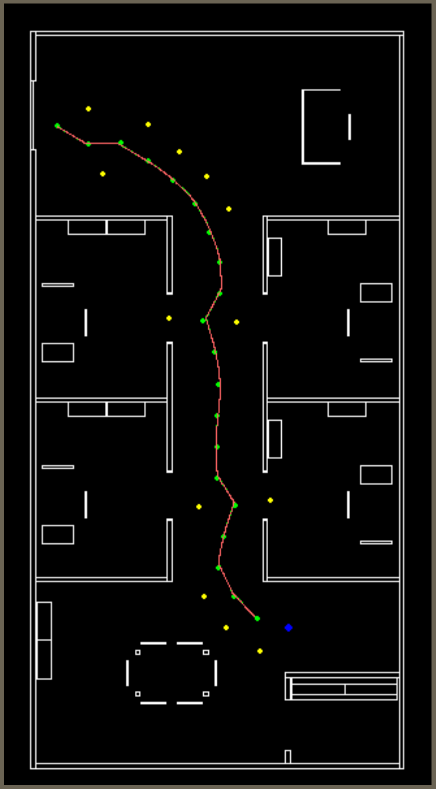
Following the path ((9,8,7,6,5,4,3,1,2)) Very long distance!

LRF indicates clear direct path: Attempt to move directly to target node.

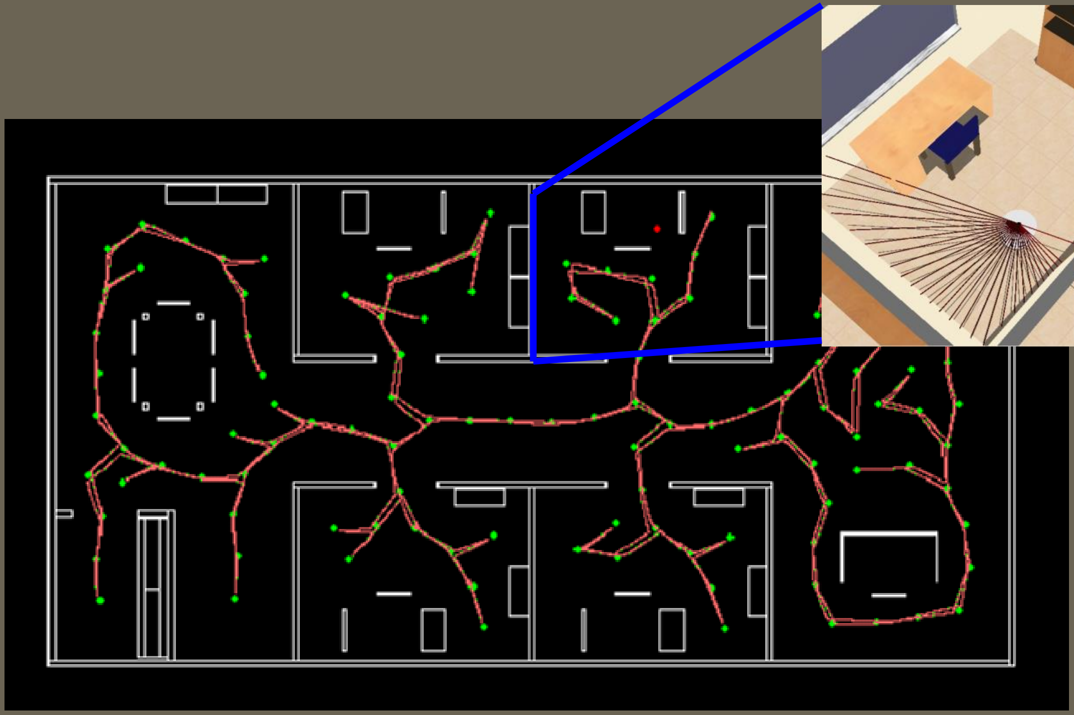
Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

CHALMERS

Exploration



Exploration



Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Today's learning goals

- After this lecture you should be able to
  - Describe different approaches to area coverage ✓
  - Describe, in detail, a method for exploration of an initially unknown arena ✓
  - Describe and implement best-first search.

Navigation: Grid-based methods

- In grid-based navigation methods, the arena is covered by a grid of (convex) cells.
- The grid can be represented either as nodes and vertices or as cells:

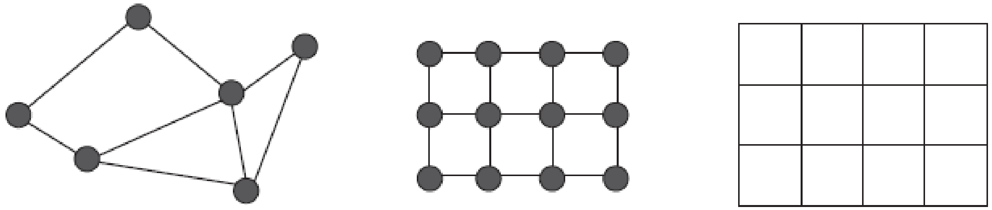


Figure 6.5: *Three examples of grids that can be used in connection with grid-based navigation methods. In the left panel, the nodes are not equidistant, unlike the middle panel which shows a regular lattice with equidistant nodes. The regular lattice can also be represented as grid cells, as shown in the right panel. Note that the right and middle panels show equivalent grids.*

Navigation: Grid-based methods

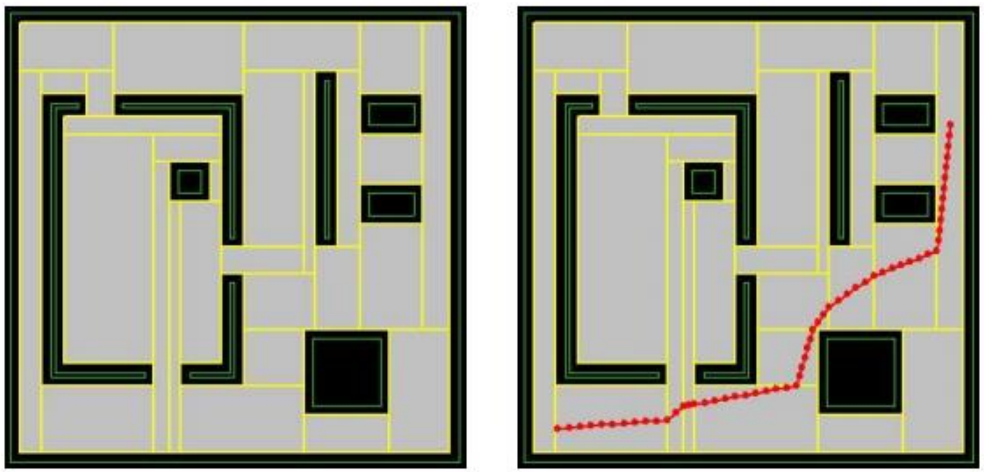


Figure 6.6: *Left panel: An example of automatic grid generation. The walls of the arena are shown as green thin lines. The black regions represent the forbidden parts of the arena, either unreachable locations or positions near walls and other obstacles. The grid cell boundaries are shown as thick yellow lines. Right panel: An example of a path between two points in the arena. The basic path (connecting grid cells) was generated using Dijkstra's algorithm (see below). The final path, shown in the figure, was adjusted to include changes of directions within grid cells, thus minimizing the length of the path. Note that all cells are convex, so that the path segments within a cell can safely be generated as straight lines between consecutive waypoints.*

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde

Best-first searh

1. Place the robot at the start node, which then becomes the current node. Assign the status *unvisited* to all nodes.
2. Go through each of the cells a_i that are (i) unvisited and (ii) directly reachable (via an edge) from the current node c. Such nodes are referred to as **neighbors** of the current node. Compute the cost of going from a_i to the target node t, using the heuristic $f(a_i)$.
3. Select the node $a_{\min}$ associated with the lowest cost, based on the cost values computed in Step 2.
4. Set the status of the current node c as *visited*, and move to $a_{\min}$ which then becomes the current node.
5. Return to Step 2.

Figure 6.6: *The best-first search algorithm.*

Best-first searh

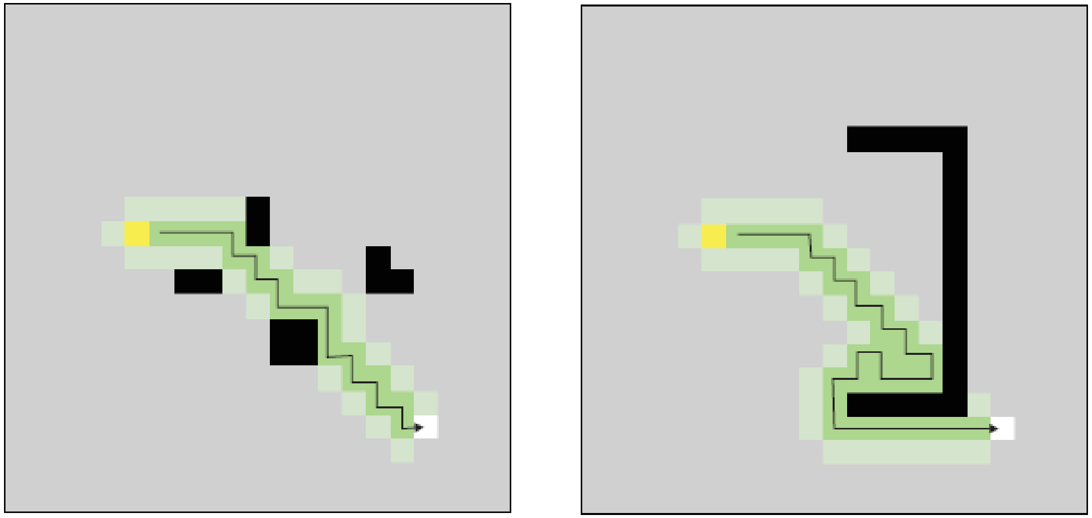
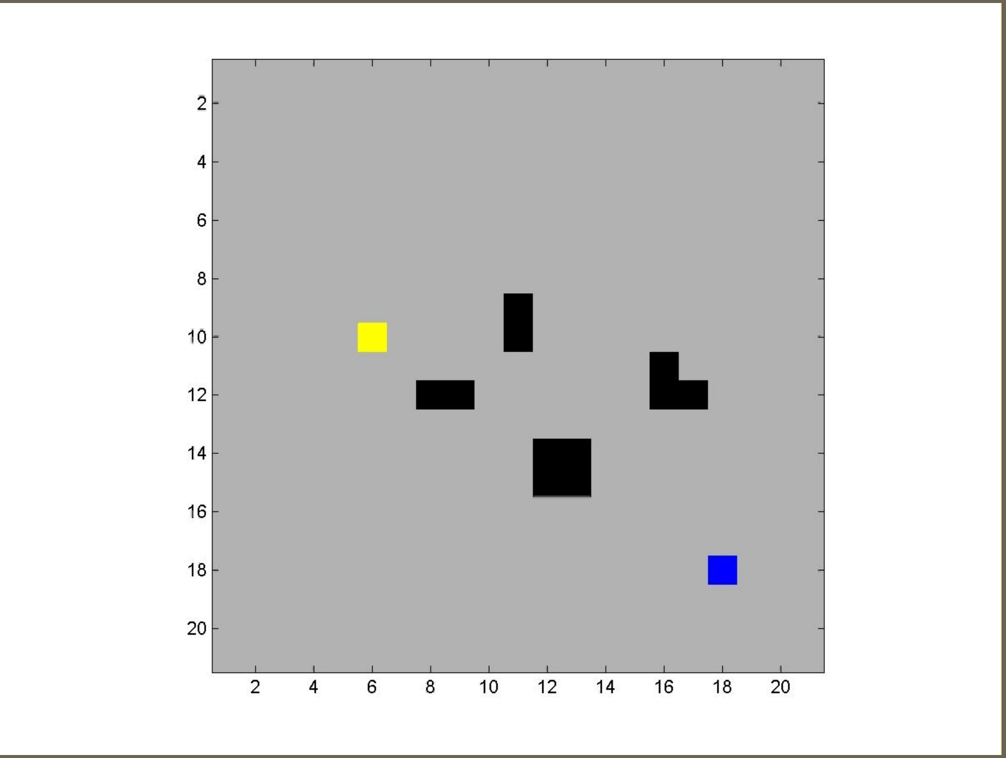


Figure 6.7: *Two examples of paths generated using the BFS algorithm. The cells (nodes) that were checked during path generation are shown in light green, whereas the actual path is shown in dark green and with a solid line. The yellow cell is the start node and the white cell is the target node.*

Best-first searh



Today’s learning goals

- After this lecture you should be able to
  - Describe different approaches to area coverage ✓
  - Describe, in detail, a method for exploration of an initially unknown arena ✓
  - Describe and implement best-first search. ✓

Mattias Wahde, PhD, Professor, Chalmers University of Technology
e-mail: mattias.wahde@chalmers.se, http://www.me.chalmers.se/~mwahde