CS 2710 Foundations of AI
Lecture 3

Uninformed search methods

Milos Hauskrecht
milos@cs.pitt.edu
5329 Sennott Square

Announcements

- **Homework 1**
  - Access through the course web page
    http://www.cs.pitt.edu/~milos/courses/cs2710/
  - Two things to download:
    - Problem statement
    - C/C++ programs you will need for the assignment
- **Due date**: September 14, 2004 before the lecture
- **Submission:**
  - **Reports**: on the paper at the lecture
  - **Programs**: electronic submissions
    Submission guidelines:
    http://www.cs.pitt.edu/~milos/courses/cs2710/program-submissions.html

Formulating a search problem

Many challenging problems in practice require search

- **Search (process)**
  - The process of exploration of the search space
- **Search space:**
  - alternatives (objects) among which we search for the solution
- **The efficiency of the search depends on:**
  - The **search space and its size**
  - Method used to explore (traverse) the search space
  - Condition to test the satisfaction of the search objective (what it takes to determine I found the desired goal object
- **Think twice before solving the problem by search:**
  - Choose the **search space** and the **exploration policy**

Uninformed search methods

- Many different ways to explore the state space (build a tree)

Uninformed search methods:
  - use only information available in the problem definition
- **Breadth first search**
- **Depth first search**
- **Iterative deepening**
- **Bi-directional search**

For the minimum cost path problem:

- **Uniform cost search**

Search methods

Properties of search methods :

- **Completeness.**
  - Does the method find the solution if it exists?

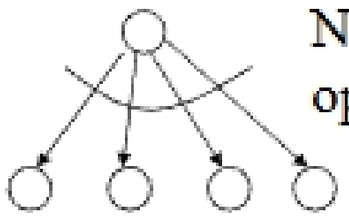
- **Optimality.**
  - Is the solution returned by the algorithm optimal? Does it give a minimum length path?

- **Space and time complexity.**
  - How much time it takes to find the solution?
  - How much memory is needed to do this?

Parameters to measure complexities.

- **Space and time complexity.**
  - **Complexities** are measured in terms of parameters:
    - b – maximum branching factor
    - d – depth of the optimal solution
    - m – maximum depth of the state space

Branching factor



Number of operators