CS 235: Artificial Intelligence

Week 2

Blind (Uninformed) Search

Dr. Moumita Roy
CSE Dept., IIITG

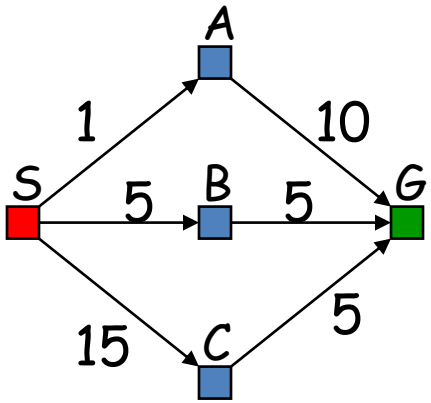
Reference: http://ai.stanford.edu/~latombe/cs121/2011/schedule.htm

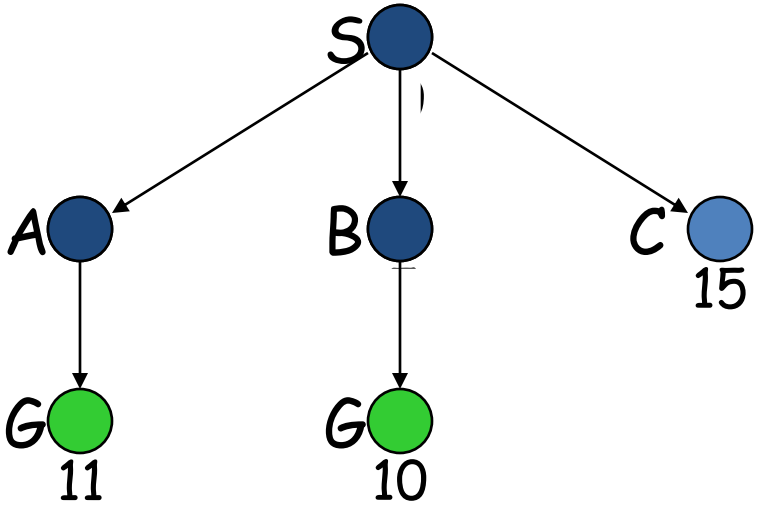
Uniform-Cost Search

- Each arc has some cost $c \geq \varepsilon > 0$
- The cost of the path to each node n is

  $g(n)$ = Σ costs of arcs

  w(n,m)= arc cost between node n and m
- The goal is to generate a solution path of minimal cost
- The nodes n in the queue FRINGE are sorted in increasing g(n)





Search algorithm (UCS)

1. **Initialize:** Set OPEN = {s},
   CLOSED = { } Set g(s) = 0
2. **Fail:** If OPEN = { }, Terminate & fail
3. **Select:**
   Select the minimum cost state, n,
   from OPEN and save n in CLOSED
4. **Terminate:**
   If n ∈ G, terminate with success

Search algorithm (UCS)

5. **Expand:**
   Generate the successors of n using successor function.
   For each successor, m:
   If m $\notin$[OPEN $\cup$ CLOSED]
   Set g(m) = g(n) + w(n,m)
   and insert m in OPEN

   If m $\in$ [OPEN $\cup$ CLOSED]
   Set g(m) = min {g(m), g(n) + w(n,m)}
   If g(m) has decreased and
   m $\in$ CLOSED, move it to OPEN

Evaluation

- **Completeness:** complete, such as if there is a solution, UCS will find it.

- What happens if negative/zero arc costs are allowed?

- **Optimal:** optimal as it only expand a node with the lowest path cost

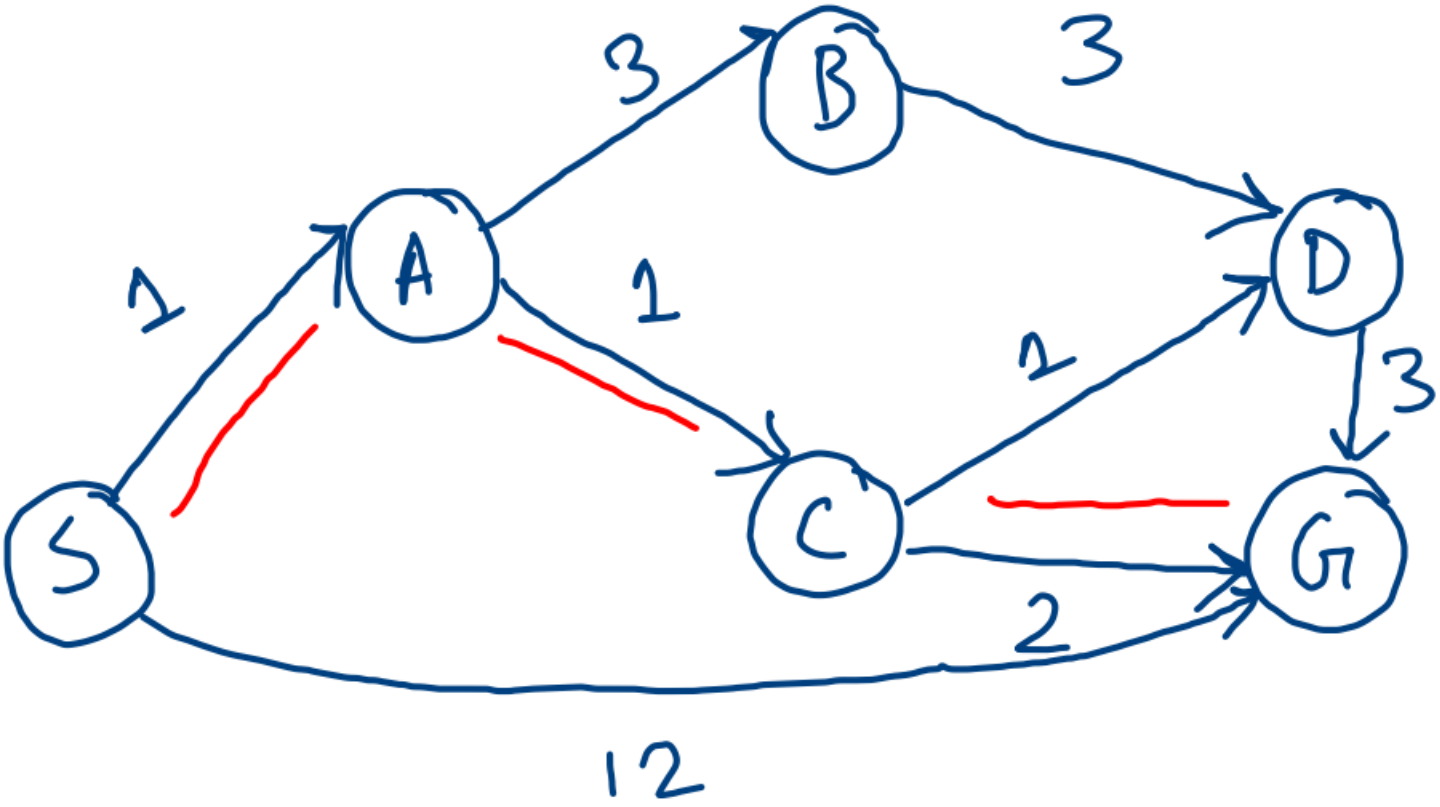
Evaluation

Time Complexity:

- UCS is guided by path cost, rather than depth (d or m)
- Let C* is Cost of the optimal solution, and assume that every action cost at least **ε**
- the number of steps is = (C*/ε)+1. (goal test at the time of expansion)
- Hence, the worst-case time complexity and space complexity: $O(b^{1 + floor(C^*/\varepsilon)})$
- Is it greater than $O(b^d)$?

  It can be greater than $O(b^d)$.It may explore large trees of small steps before exploring paths involving large steps.
- When all arc costs are same, UCS is similar to BFS.

Example



UCS (step-wise)

1. $OL = \{S^0\}$, $CL = \{\}$

2. $OL = \{A^1, Gr^{12}\}$
   $CL = \{S^0\}$

3. $OL = \{Gr^{12}, B^4, C^2\}$
   $CL = \{S^0, A^1\}$

4. $OL = \{Gr^{(12\not4)}, B^4, D^3\}$
   $CL = \{S^0, A^1, C^2\}$

5. $OL = \{Gr^4, B^4\}$
   $CL = \{S^0, A^1, C^2, D^3\}$

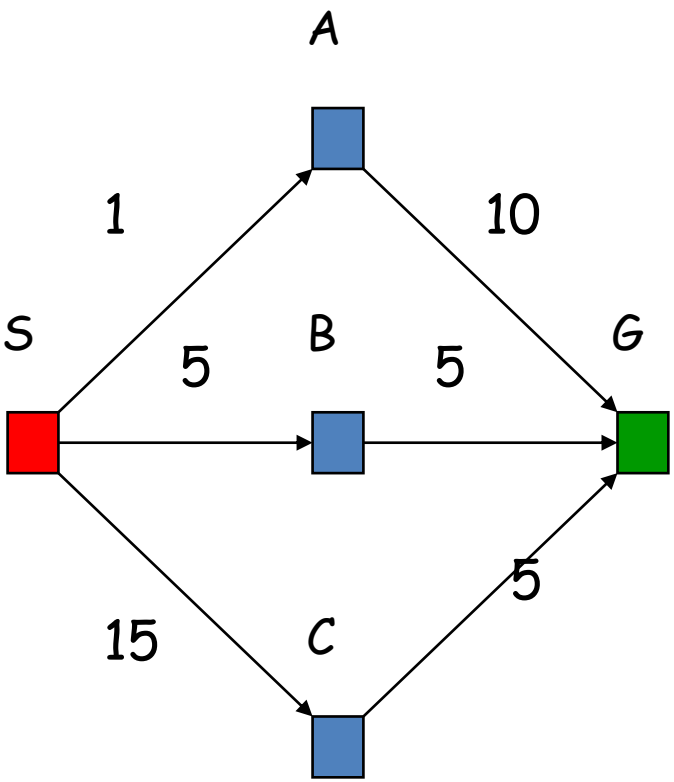
→ update parent node

6. $OL = \{B^4\}$
   $CL = \{S^0, A^1, C^2, D^3, Gr\}$

Solution path:
$S \to A \to C \to Gr$

Goal test during node expansion

Example 2



1. OL={S^0}, CL={}

2.OL={A^1, B^5, C^{15}}, CL={S^0}

3.OL={B^5, C^{15}, G^{11}}, CL={S^0, A^1 }

4.OL={C^{15}, G^{10}}, CL={S^0, A^1 , B^5}

5. OL={C^{15},}, CL={S^0, A^1 , B^5 , G^{10}}

Solution Path: S->B->G

The node A has been expanded.

Is there any way to stop such expansion?

Different Scenarios of UCS

- UCS with positive arc cost is optimal (first goal node selected for expansion must be the optimal solution)
- Here, path never be shorten when nodes added.
- UCS is complete when the arc cost exceeds some small positive constant
- It will stuck into infinite loop if there is infinite sequence of zero-cost action along the path
- In UCS with positive arc cost, all nodes are expanded once (no move from close to open). Verify?
- What happen in UCS for a state space with negative arc cost? Is it optimal?