B. Sc. (H) Computer Science Semester VI
Core Paper XIII – Artificial Intelligence

Topic:

- **Best First Search**
- **Constraint Satisfaction Problem**

Best First Search

Best First Search (Heuristic Search) - a greedy search

We evaluate a heuristic function.

Movement should be such that we move in a forward direction.

(1)Begin with a set of rules which are consistent and try to arrive at a solution.

(2)User did not explicitly specify the sequence of use of rules. Sequence of rules is decided by the KB (i.e. by the system).

In some way, the solution space should be explored. AI systems attempt to get solution best in some sense (acceptable solution), they do not attempt to get optimality. Solutions obtained by AI systems may not be optimal.

The best does not mean optimal. **Best is in heuristic sense**.

This avoids searching completely. This is a Greedy algorithm.

The word **heuristic** comes from **Greek word heuriskein** meaning **to discover**.

Heuristic function is a function that gives a judgement from given stage to goal stage, e.g., nearest neighbor.

At each step of the best first search process, we select the most promising of the nodes we have generated so far. We then expand the chosen node by using the rules to generate its successors.

If one of them is a solution, we then quit. If not, all those new nodes are added to the nodes generated so far. Again the most promising node is selected and the process continues.

Usually, what happens is that a bit of depth-first occurs as the most promising branch is explored. But eventually, if a solution is not found, that branch will start to look less promising than one of the top level branches that had been ignored. At that point, the new more promising previously ignored branch will be explored. But the old branch is not forgotten. Its last node remains in the set of generated but unexpanded nodes. The search can return to it whenever all the others get bad enough that it is again the most promising path.

Algorithm : Best-First Search

1. Start with OPEN containing the initial state.

2. **Until** a goal is found or there are no nodes left in OPEN **do**

   (a) Pick the best node in OPEN.

   (b) Generate its successors.

   (c) For each successor do

      (i)if it has not been generated before, evaluate it, add it to OPEN and record its parents.

      (ii)if it has been generated before, change the parent if this new path is better than the previous one.

In that case, update the cost of getting to this node and to any successors that this node may already have.

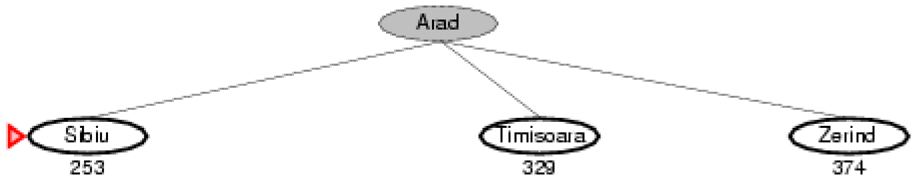
Greedy best-first search

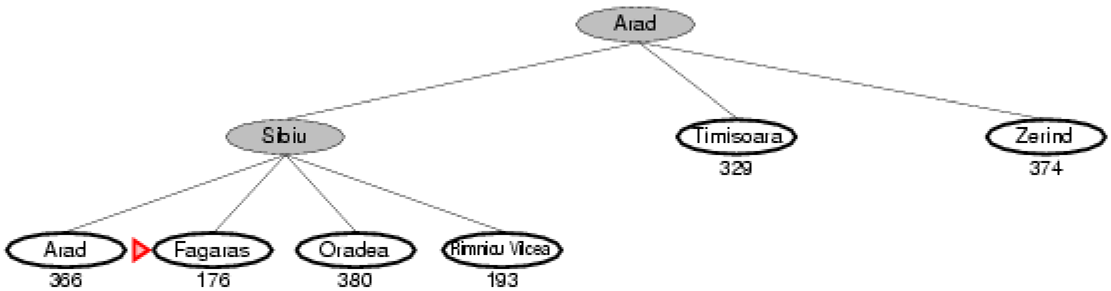
Evaluation function $f(n) = h(n)$ (heuristic)= estimate of cost from n to *goal*

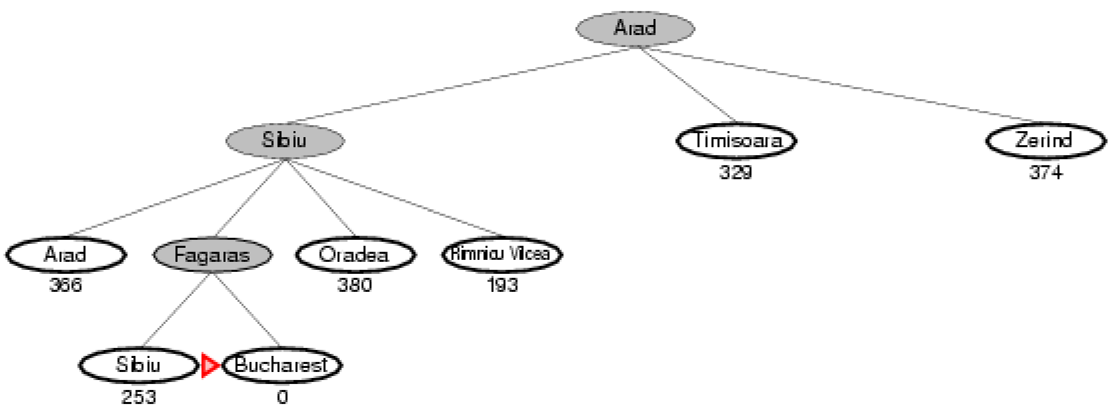
e.g., $h_{SLD}(n)$ = straight-line distance from n to Bucharest

Greedy best-first search expands the node that appears to be closest to goal

Greedy best-first search example







Properties of greedy best-first search

<u>Complete?</u> No – can get stuck in loops, e.g., Iasi → Neamt → Iasi → Neamt →
<u>Time?</u> $O(b^m)$, but a good heuristic can give dramatic improvement
<u>Space?</u> $O(b^m)$ -- keeps all nodes in memory
<u>Optimal?</u> No

Note: Please solve example on the topic discussed above from reference book [*]. (pg 74-75)

*Elaine Rich, Kevin Knight, & Shivashankar B Nair, Artificial Intelligence, McGraw Hill, 3rd ed.,2009.

Also refer http://www.vssut.ac.in/lecture_notes/lecture1423725949.pdf and solve the given examples.

Constraint Satisfaction Problem

A constraint satisfaction problem (CSP) is a problem that requires its solution within some limitations/conditions also known as constraints. It consists of the following:

- A finite set of **variables** which stores the solution. (V = {V1, V2, V3,....., Vn})
- A set of **discrete** values known as **domain** from which the solution is picked. (D = {D1, D2, D3,.....,Dn})
- A finite set of **constraints.** (C = {C1, C2, C3,......, Cn})

Please note that the elements in the domain can be both continuous and discrete but in AI, we generally only deal with discrete values.

Also, note that all these sets should be finite except for the domain set. Each variable in the variable set can have different domains. For example, consider the Sudoku problem again. Suppose that a row, column and block already have 3,5 and 7 filled in. Then the domain for all the variables in that row, column and block will be {1,2,4,6,8,9}.

Converting problems to CSPs

A problem to be converted to CSP requires the following steps:

Step 1: Create a variable set.

Step 2: Create a domain set.

Step 3: Create a constraint set with variables and domains (if possible) after considering the constraints.

Step 4: Find an optimal solution.

Note: Please solve example on the topic discussed above from reference book [*].(pg 92-93)

*Elaine Rich, Kevin Knight, & Shivashankar B Nair, Artificial Intelligence, McGraw Hill, 3rd ed.,2009.

Also refer http://www.vssut.ac.in/lecture_notes/lecture1423725949.pdf and solve the given examples.

Assignment

Q1.Write short notes onthe following:

i) Expert Systems

ii) Propositional Logic

Q2. Explain uses of Cut and Fail predicates in PROLOG.

Q3. Give the Percept, Actions, Goals and Environment of the following Agent types:

i) Part Picking Robot
ii) Interactive English Tutor

Q4. Differentiate between breadth first search and best first search.

Q5. Find the meaning of the statement

$$(\sim P \vee Q) \wedge R \rightarrow S \vee (\sim R \vee Q)$$

for the following interpretation: P is true, Q is true, R is false, S is true.

Q6. Here are some simple clauses.

```
likes(mary,food).
likes(mary,wine).
likes(john,wine).
likes(john,mary).
```

How do you add the following facts?

1. John likes anything that Mary likes.
2. John likes anyone who likes wine.
3. John likes anyone who likes themselves.

Q7. How Rational Agent approach is different from Turing Test approach in AI.

Q8. Analyze 8-puzzle problem and give a suitable state space representation for this problem.

Q9. How is steepest ascent hill climbing different from simple hill climbing? Explain.

Q10. Explain backtracking in PROLOG with a suitable example.

Q11. Trace the constraint satisfaction procedure solving the crypt-arithmetic problem:

```
  TWO
 +TWO
______
 FOUR
```

Q12. Trace the constraint satisfaction procedure solving the crypt-arithmetic problem:

```
  SEND
 +MORE
______
 MONEY
```

Q13. Find the most general unifier for the following sets.

(i) w = {P(a,x,f(g(y))), P(z,f(z),f(u))}
(ii) w = {P(A,B,B), P(x,y,z)}

Q14. Suppose the following facts are given:
teaches(dr_fred,history).
teaches(dr_fred,english).
teaches(dr_fred,drama).
teaches(dr_fiona,physics).
studies(alice,english).
studies(angus,english).
studies(amelia,drama).
studies(alex,physics).

Write the output for each of the following and explain the difference between them with respect to the usage of cut.

1. ?- teaches(dr_fred,Course),studies(Student,Course).
2. ?- teaches(dr_fred,Course),!,studies(Student,Course).
3. ?- teaches(dr_fred,Course),studies(Student,Course),!.

Q15. You are given two jugs of capacity 4-gallon and 3-gallon respectively. Neither has any measuring marker on it. There is a pump that can be used to fill the jugs with water. How can you get exactly 2 gallons of water into the 4-gallon jug? Write down solution by showing all intermediate states.