**INT-404 Artificial Intelligence ASSIGNMENT**

**Topic- Block World Problem using A\* Algorithm.**

**GROUP Members**

**1.AKANKSHA TRIPATHI**

**REG NO:11815685**

**ROLL NO:34**



**LOVELY PROFESSIONAL UNVERSITY**

**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

**Abstract**

The project includes study and solving of the Block World Problem.

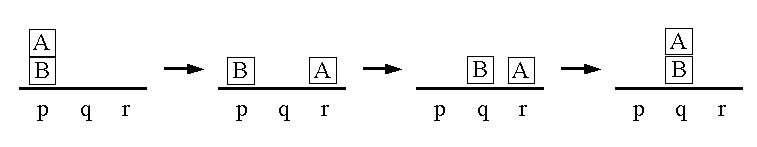
The block world problem is a simple and a popular problem used in AI to understand planning domain of the Artificial Intelligence.

The project includes solving Block World Problem using:

1.A\* algorithm

which is based on the BEST FIRST SEARCH strategy and uses heuristic function to give a cost to each state and compare one state to another to find the most optimal solution.

The implementation of these algorithm is in Python.



Introduction

The Block World environment has ,

* Square blocks of same size which are identified by letters.
* Blocks can be stacked one upon another to form a tower.
* Flat surface (table) on which blocks can be placed.
* And only one block can be moved at a time.
* There is arm also which can hold and move only one block at a time.

In block world problem, the state is described by a set of predicates representing the facts that were true in that state. At every new state the changes in the states is described. We have four types of probable operations which could be applied in block world environment .They are

UNSTACK

* Move a block from top of the other blocks, but at the time of moving no block should be on top of the block which is been moved.

STACK

* This step includes placing the block oon top of another block .

PICKUP

* This step includes picking up a block placed on the table or another block and the block which is been picked up should not have another block on top of it.

PUTDOWN

* Putting the block down on the table .

Under all the above operations the arm used can have only one block at any given time.

Along with the operations there are five main predicates :

ON(A,B) --One block is on another.

ONTABLE(A) -- Block is on the table.

CLEAR(A) -- Block has nothing on it.

HOLDING(A) -- The arm holds a block.

ARMEMPTY --The arm holds nothing.

All these operations and predicates combined make the Block world environment.

In this project we use two popular AI algorithms to solve the Blocks World problem.

**1.A\* search Algorithm**

It is a search algorithm primarily used to find best path to reach a Goal.

It is one of the best and popular technique used in path-finding and graph traversals.

The algorithm uses a heuristic function approach to solve a problem where it compares the heuristic of subsequent nodes with the previous heuristic value

and moves on to the path with the best value.

The heuristic function f(n) uses a cost function along with the usual heuristic value at every state to get a more precise value to compare among various subsequent states and then evaluate the most promising one and this added use of cost function make A\* search strategy a more promising and fast algorithm among it’s competition.

**Literature Review**

The Block world problem is among the interesting problems used in study of AI.

The following Literature has helped us to properly understand the complete Block World Environment and also gave us the knowledge about the Search strategy used to solve the Problem :

1. Blocks world -Wikipedia.

The page consisted of all the necessary information about what exactly is the Block World problem . It also introduced us with the constraints/rules needed to follow while solving a Block world problem.

2.Block World revisited -by john slaney

The research paper presents a detailed information on the block world problem which is a part of the Toy domains in AI.it main focus is on how a simple problem like it can actually provide information on planning strategies in AI.

3.Artificial Intelligence(AI):Planning

the above page consisted of material which provided a brief understanding of the problem and included the approaches that can used to solve the problem.it taught about the rules and predicates used in a Block World Environment.

4.A\* search Algorithm-GeeksforGeeks

The above page includes the details about the search strategy and also explains its theoritical as well as practical application of the Algorithm.

It explains about the background on which the algorithm is based i.e,using heuristic function approach to compare the states.

It also provided us with the Pseudo code for the implementation of the Algorithm.

**Proposed Methodology**

Both the above methods A\* search and OR graph are based on the BEST FIRST SERACH strategy and both use a Heuristic function to evaluate there successors for the most promising path for solving the problem.

1.The A\* Search Algorithm.

This Method requires a heuristic function.

f(n)=g(n)+h(n).

For the Block World Problem we have used a heuristic function f(n) that estimates the merits of each node we generate. This is sum of two components ,g(n) and h(n).

g(n)-> it is the cost to reach the state i.e, initial to the current state.

h(n)->it is the estimated cost to get from the current state to the goal state.

So we use the above f(n) which we calculate after the application of each rule and then find the most promising path and continue till we reach our goal state or there are no more states left to explore.

Algorithm

Step1:Start with OPEN containing only Initial node. Set that

node’s ‘g’ value to 0, its h value to whatever it is, and

its f value to h+0. Set CLOSED to empty list.

Step2: Until a goal node is found, repeat the following procedure:

1.If there are no nodes on OPEN, report failure.

2.Otherwise pick the node on OPEN with the lowest f value.

Call it BESTNODE. Remove it from OPEN.Place it in CLOSED.

a.See if the BESTNODE is a goal state. If so exit and

report a solution.

b.Otherwise, generate the successors of BESTNODE but

do not set the BESTNODE to point to them yet.

c.For each of the Successor, do the following:

i. Set SUCCESSOR to point back to BESTNODE. These

backwards links will make it possible to

recover the path once a solution is found.

ii. compute g(SUCCESSOR)=G(BESTNODE) + the cost of

getting from BESTNODE to SUCCESSOR.

iii. See if SUCCESSOR is the same as any node on

OPEN. If so call the node OLD.

iv. If SUCCESSOR was not on OPEN, see if it is on

CLOSED.If so, call the node and CLOSED OLD

and add OLD to the list of BESTNODE’s

successors.

v. If SUCCESSOR was not already on either OPEN or

CLOSED, then put it on OPEN and add it to the

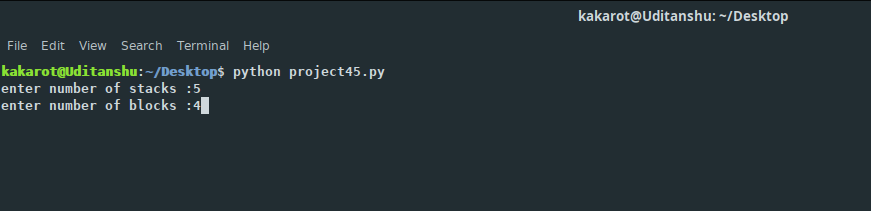
list of BESTNODE’s successors.Compute

f(SUCCESSOR)=g(SUCCESSOR)+h(SUCCESSOR).

1. END.

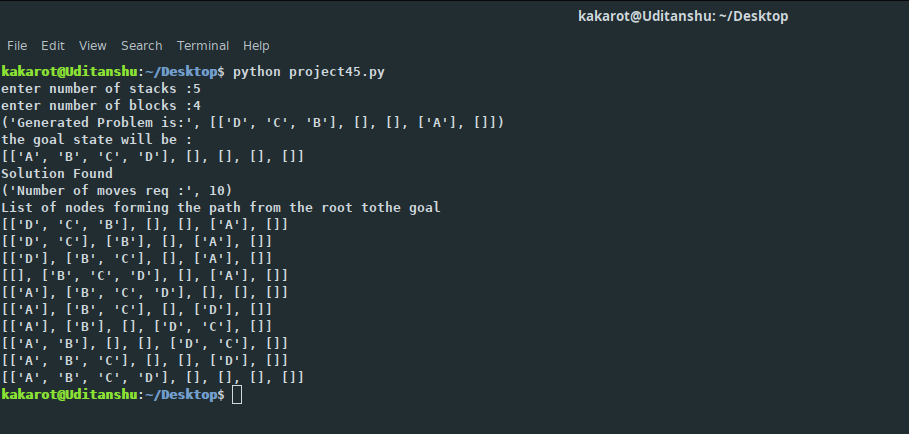
**RESULT AND DICUSSION**

1.The program starts with asks the user to input the number of blocks.



1. Then the program generates a random initial state and a random Goal state.

3.After the generation of the states it shows the best path i.e, showing the states used to reach the goal state.



**Conclusion:**

Artificial Intelligence can be researched with theory and with practical problems like the block world problem which is sufficiently simple and well behaved and is a easily understood problem.

Yet still provides a good sample environment to study.

Thus in the above report we have discussed about the implementation of various algorithms like A\* search strategy which can be used to solve the Block world problem and also increase one’s knowledge in AI.

The code for the above Problem can be found at github.

Github link: https://github.com/UjJWgithub/AI-Project/commit/7f0ebe685712cefde7529538c6231c64616bf54a

**References**

1.Blocks World - Wikipedia

https://en.wikipedia.org/wiki/Blocks\_world

2.A\* Search Algorithm - Wikipedia

[https://en.wikipedia.org/wiki/A\*\_search\_algorithm](https://en.wikipedia.org/wiki/A*_search_algorithm)

3.A\* Search Algorithm- GeeksforGeeks

https://www.geeksforgeeks.org/a-search-algorithm/

4.Artificial Intelligence(AI):Planning

https://aimaterials.blogspot.com/p/planning.html?m=1