

AI Assignment 2
Nrupesh Surya U – 2017B2A70767G

Heuristics used for DPLL Algorithm:

1. Early Termination : If any one of the symbols used is evaluated to true then the clause is set to true, if any one of the clause derives a box clause, Algorithm terminates.
2. Pure Symbols : Symbols whose polarity remains the same throughout are assigned favorable values and are removed.
3. Unit Clauses : Clauses which contain only one literal can be removed with satisfying assignments.

DPLL Algorithm followed :

step 1 : find all pure symbols and eliminate all clauses with the pure symbol

step 2 : find all unit clauses and terminate if there are conflicting unit clauses like $p, \sim p$
if not, remove the unit clauses, clauses containing the literal and negation of the literal from the clauses which contained it.

Step 2.5 : check for length of knowledge base : if zero return true, check length of all clauses , if any are zero return false

Step 3 : do the above process recursively until there are no pure symbol/unit clauses to remove

step 4: pick a literal and assign a value to the literal. Remove clauses which contain the literal and remove the negated forms from the respective clauses.

Step 5: repeat from step1

step 6: return true if satisfying assignment has been found

step 6: assign a different value from that of step 4 and do the same again

step 7 : return true if satisfying assignment has been found

step 7 : no satisfying assignment was found. Return false

Simulation Algorithm followed :

- i. start from 1,1 and maintain a visited and to_explore sets and a list to mark safe tiles
- ii. check for stench and breeze and add it to the knowledge base
- iii. check if there is no wumpus / pit in the adjacent rooms by using dpll algorithm ($kb \wedge \text{not alpha}$ unsatisfiable) and add corresponding sentence to kb if both are not present and update to_explore set and also update the list with safe tiles
- iv. if we can't say for sure that the adjacent room is safe from both wumpus and pit, check for the possibility that there is wumpus and pit
- v. explore new tiles from to_explore set
- vi. keep exploring the to_explore tiles using bfs algorithm to go from one safe tile to another (using the list which contains all safe tiles)
- vii. if the tile is 4,4 exit.
- viii. If the to_explore set is empty, backtrack to 1,1 and start exploring again (This is useful because some tiles which might have been previously unsafe might be marked as safe when explored again because the wumpus and pit would have been discovered accurately)

Values obtained by different combination of heuristics :

Current Wumpus world :

```
["", "", "", "], # Rooms [1,1] to [4,1]
["", "", 'P', ""], # Rooms [1,2] to [4,2]
['W', "", "", ""], # Rooms [1,3] to [4,3]
["", "", "", ""], # Rooms [1,4] to [4,4]
```

Using early termination alone : Number of calls to dpll \rightarrow 13,968

Using unit clause + early termination \rightarrow Number of calls to dpll \rightarrow 87

Using pure symbol + early termination \rightarrow Number of calls to dpll \rightarrow 6580

Using all three heuristics \rightarrow Number of calls to dpll \rightarrow 82

This shows how effective unit clause is in case of our problem which has a lot of unit clauses as we progress deep into the level. This skewed result might also be due to the fact that unit clause heuristic in this case also detects failure ($p, \sim p$) in this case.

Note : Isolating early termination was not possible since unit clause and pure symbol used a combination of early termination in the implementation as well