planning project

Prakrit Tyagi

December 2023

1 Introduction

Algorithm 1 Life Long Multi-agent Conflict Based Search

```
Paths = ∅
list<bool> agents_to_plan = {True for all}
bool run ← true
while t ≤ T_sim do
    if run then
        Paths ← CBS(Paths, goal_list, agents_to_plan)
        run ← false
        agents_to_plan= {False for all}
    end if
    action ← Paths[t]
    for each agent a do
        if action leads to goal then
            run ← true
            agents_to_plan[a] = true
        end if
    end for
    t++
end while
```

```
1: function CBS(Paths, goal_list, agents_to_plan)
2:     Root.solution = Paths[t,End]
3:     Update Root.solution using low_level(goal_list[t],agents_to_plan)
4:     Root.cost = Sum_of_cost(Root.solution)
5:     Root.constraints = ∅
6:     push Root to OPEN
7:     while OPEN is not empty do
8:         P ← best node from OPEN
9:         Check the paths in P for conflicts
10:        if P has no conflict then
11:            return P.Solution
12:        end if
13:        C ← first conflict(a_i, a_j, v, t) in P
14:        for each agent a in C do
15:            A ← new node
16:            A.constraints ← P.constraints + (a, v, t)
17:            A.solution ← P.solution
18:            Update A.solution
19:            A.cost = Sum_of_cost(A.solution)
20:            push A to OPEN
21:        end for
22:    end while
23: end function
```