**MA collisions handling ideas**

CONS: every time we modify the paths (by adding NoOps or deviations) we need to recheck that there are no collisions or boxes on the way from the start

* Identify RepoCells, as positions in which there are no agent passing by, initial and to be updated after any path deviation update. Note, if there aren’t any, priority should be given to the most well connected cells (D number), outside tunnels possibly. DONE
* Method to find with A\* the path to the closest reachable RepoCell, given an Agent position (just another pathfinding agent-box; box-repocell) TODO
* Have each agent dynamically inherit the priority level depending of the box-goal he is moving towards TODO

**1a) COLLISION DETECTION between agents**

* this should check also for agents who have no goals (agents without goals must have only a list of NoOps, not sure if this is already implemented).
* **IDEA:** to compare the lists of positions of agents (agentOccupiedCells) one2one looking for collisions, each agent path is compared to all the others until they all can run without collisions. DONE

**1b) COLLISION CORRECTION**

* IF there are safe positions THEN NoOp action in safe positions DONE
* IF there aren’t any safe positions THEN MOVEAWAY actions TODO

**2a) BOXES ON THE WAY DETECTION** TODO

When all the collisions between agents are solved scan for boxes lying on the solving paths by overlapping path and initial positions of boxes.

If a box is not in its initial position, it is being moved by an agent, so any collision would have been already solved by 1b.

* if no overlap, no problem
* else : are these boxes still there at (around? or exactly? ) the timestep when the agent has to move forward?
  + if no, no problem
  + else, identify the timestep of the agent-box collision and proceed with correction by creating path deviations, see 2

**2b) BOXES ON THE WAY CORRECTION** TODO

Is the box of the same color of the Agent moving towards a goal?

* if yes (eg 0 A Aa)
  + MOVEAWAYBOXACTION = A\* path (agent = Agent, box = boxontheway, goal = closest free reachable REPOcell)
  + Agent moved back to the position before deviation
  + Add deviation path actions to the Agent action list
  + take away the REPOCell from the notOccupiedCells list
* if no (eg 0 B Aa)
  + search matchingAgent of same color, which will see his path devieted
  + MOVEAWAYBOXACTION = A\* path (agent = matchingAgent, box = boxontheway, goal = closest free reachable REPOcell)
  + Agent moved back to the position before deviation
  + Add deviation path actions to the Agent action list(4 timesteps before the Agent-box collision timestep)
  + Add equal number of NoOp actions for Agent waiting that the box is moved