NAO PLANNING CHALLENGE

Agostino Aiezzo
0000982514

Emmanuele Bollino
0000985420

Artificial Intelligence
A.A. 2020/2021

MOVEMENTS

Positions

a. Predefined = ['StandInit', 'Stand', 'Sit', 'Crouch', ..]

b. Timeline = ['Guitar', 'Happy Birthday', 'Rotation Right Arm', ..]

Execution

Based on the type of the the position, we had to define two ways of executing an action:

a. **applyPosture:** Specific function, for Timeline Positions. In this case, we have exported the timeline from Coreographe and adapted into a python function

b. NAO QI library for nao interaction

Incompatibilities detection

o Semi-automatic algorithm

PLANNING

Algorithm

- Graphplan
- Python
- AIMA library

Modeling

a. **States** - position that nao can perform
b. **Actions** - movement from a position P1 to another position P2
   a. Preconditions: Nao is in P1 and P1 is compatible with P2
   b. Postconditions: Nao is in P2

Constraints

- Mandatory intermediate positions
  - Iterate over mandatory positions and generate intermediate independent plans and then unify all the plans into a single choreography
- Minimum number of positions
  - If an intermediate plan does not meet the minimum number of positions constraint, the planner generates another plan from the second-last position found so far and then generates the plan from that position to the intermediate goal after removing the compatibility from the second-last position to the goal
- Time
  - Modify the code of the positions in order to make them faster
- Artistic correction, avoiding repetitions
  - Random shuffle of the initial state
  - Deleting duplicates from the plan and generate a new plan removing compatibilities if possible

THANKS