NAO Challenge 2025

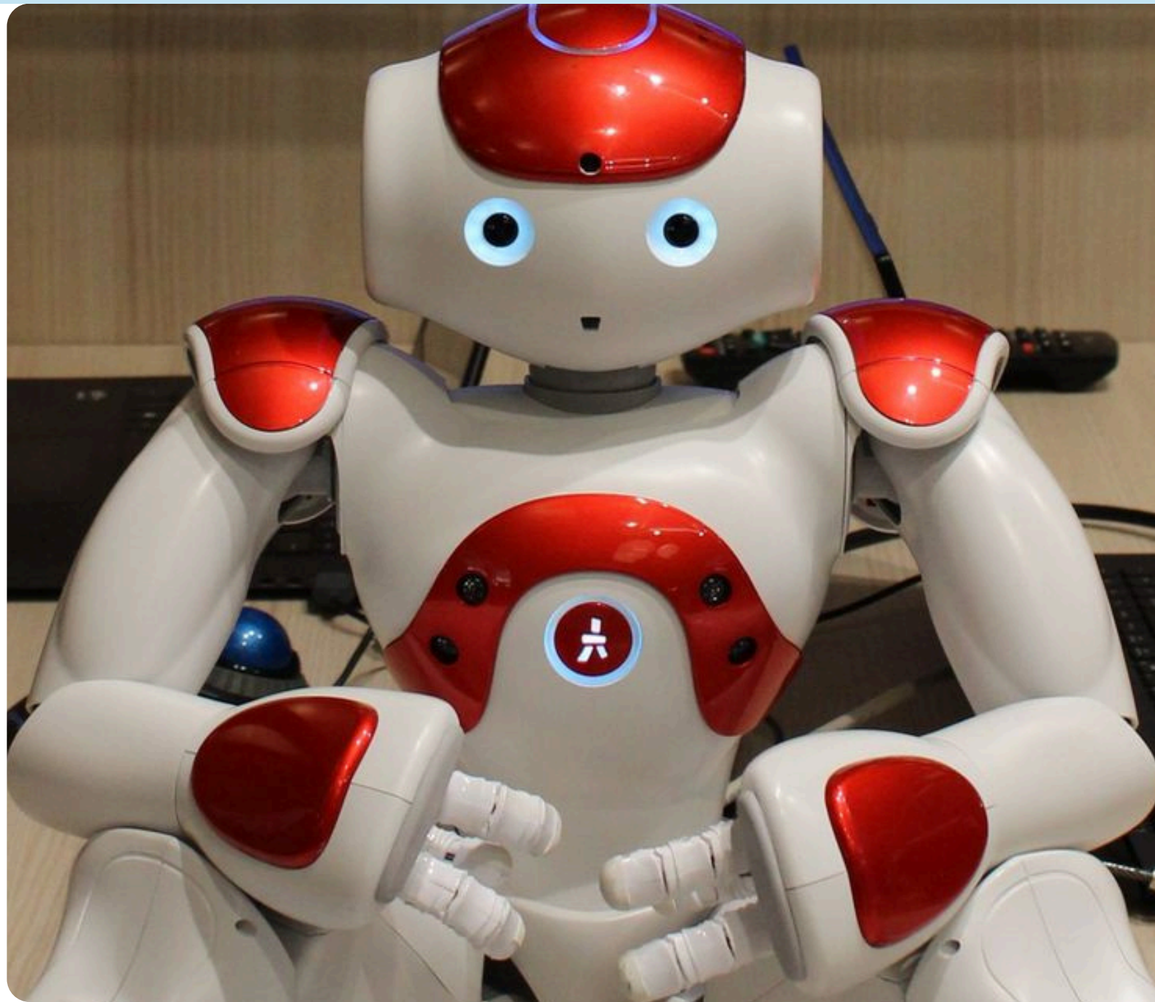
Saturday NAO Fever

Davide Tonelli

davide.tonelli8@studio.unibo.it

0001241443

Alma Mater Studiorum - M.S.c. Artificial Intelligence - A.Y. 2025-2026



Project Goal

Automatically generate a complete dance choreography for the NAO robot

- Generate a sequence of robot positions connecting all mandatory poses.
- Create a choreography within a maximum duration of 2 minutes.
- Respect robot constraints (standing / sitting conditions).
- Include at least 5 intermediate positions in the whole choreography.

System Architecture

main.py

- Defines mandatory poses
- Splits the available time into planning steps
- Creates A* subproblems between mandatory poses
- Executes moves

nao_problem.py

- Defines the search problem (state transitions, cost function, goal test, heuristic)

NaoMoves/

- Folder containing all movement scripts

Planning with A*

Path Cost Function

The cost of adding a move is:
cost = duration + λ · (repetitions)^2

Where:
duration = time of the move
repetitions = how many times this move already appears in the global choreography
λ = repetition penalty weight

The quadratic penalty discourages repeating the same move.

Heuristic Function

h(n) = max(0, R − (R_goal + tolerance))

Where:
R = remaining time in the current state
R_goal = desired remaining time (usually 0)
tolerance = allowed margin (e.g., 2.3s)