

*특정 동작을 하는
Agent 생성*

1

우리 팀 모두 상대팀 골대에 일렬로 서서 상대팀 득점 획득하는 Agent

- Player_rulebasedC를 기반으로 Agent 구현
- 우리 팀 5명 모두 상대 팀 골대에 일렬로 서서 상대 팀이 자책 골을 넣는 경우를 방어하고, 상대 팀이 점수를 많이 낼 수 있도록 하는 Agent
- 상대 팀 골대에 위치하거나 공을 잡게 되면 이를 막도록 코딩

1

우리 팀 모두 상대팀 골대에 일렬로 서서 상대팀 득점 획득하는 Agent

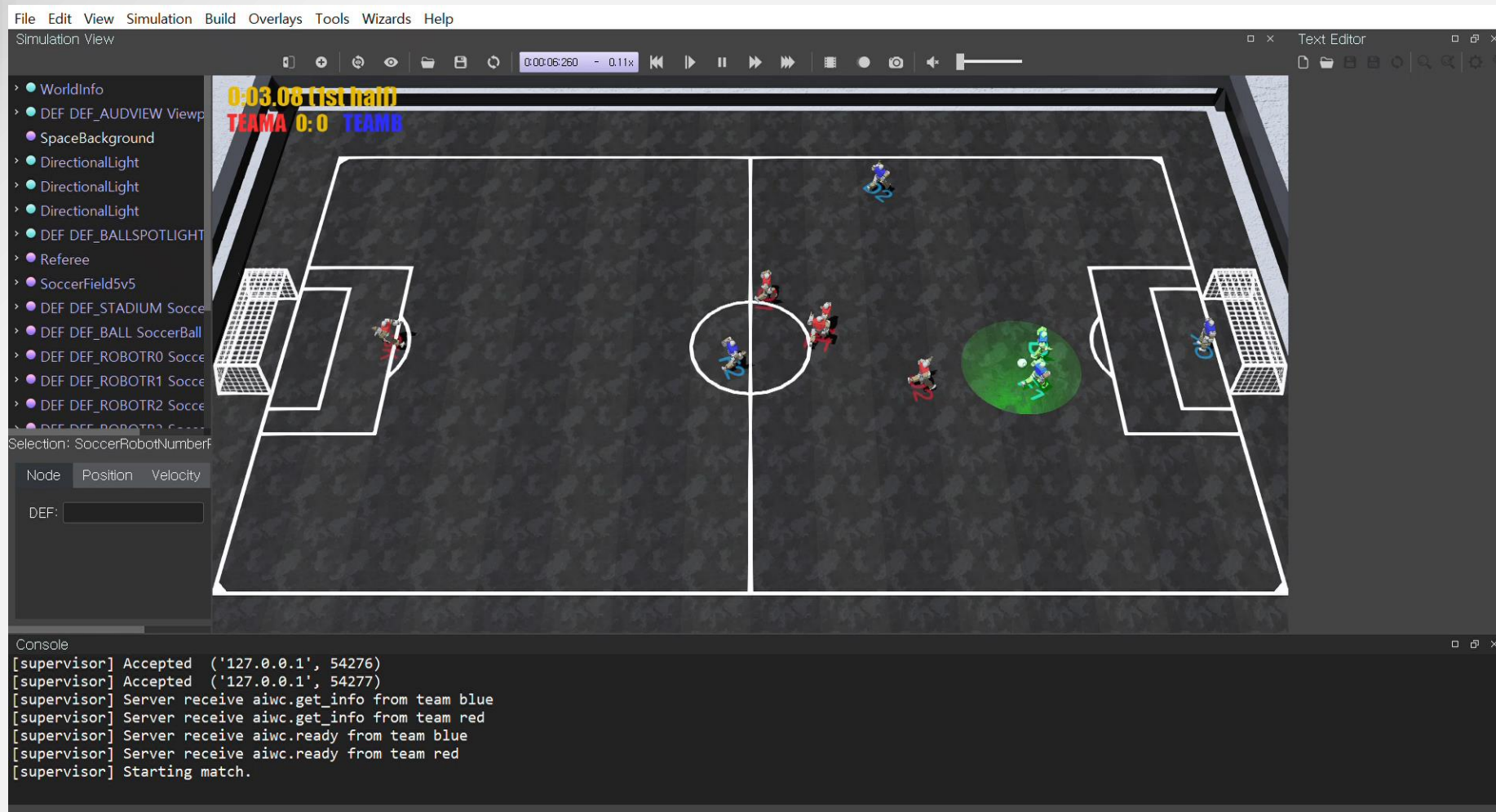
- 방법
- `players.py` 파일에서 포지션마다 (x, y) 좌표를 수정하여 골대 앞에 서 있도록 코딩
- 상대 : rulebasedC

- 문제점
- 우리 팀 5명 모두 상대팀 골대 앞에 위치하지 않음(RULE 13)
- 지정된 자리로 이동하는 도중에 골이 먹히거나, 공이 전혀 움직이지 않는 등의 문제로 인해 경기가 다시 시작되는 문제가 발생

- **RULE 13) Goalkeeper Relocation**
- If a goalkeeper leaves its own team's penalty area for more than 3 seconds, the goalkeeper is relocated back to the default position which is the same as the goalkeeper return position in a robot send out situation.
- 우리 팀이 상대 penalty area로 들어가면 다시 원래 위치로 돌아감

1

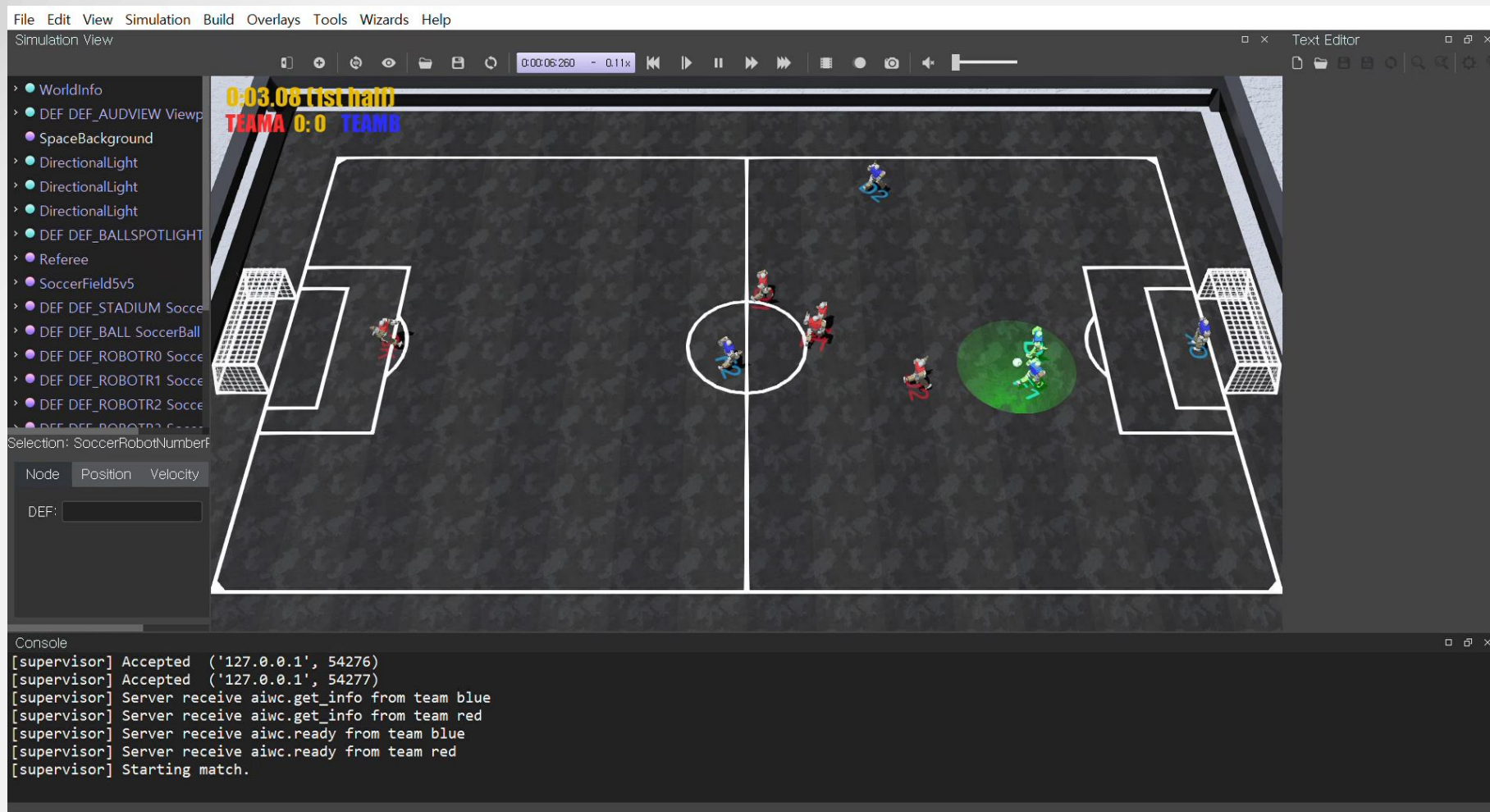
우리 팀 모두 상대팀 골대에 일렬로 서서 상대팀 득점 획득하는 Agent



우리 팀 5명 모두 상대팀 골대 앞에 위치하지 않는 문제

1

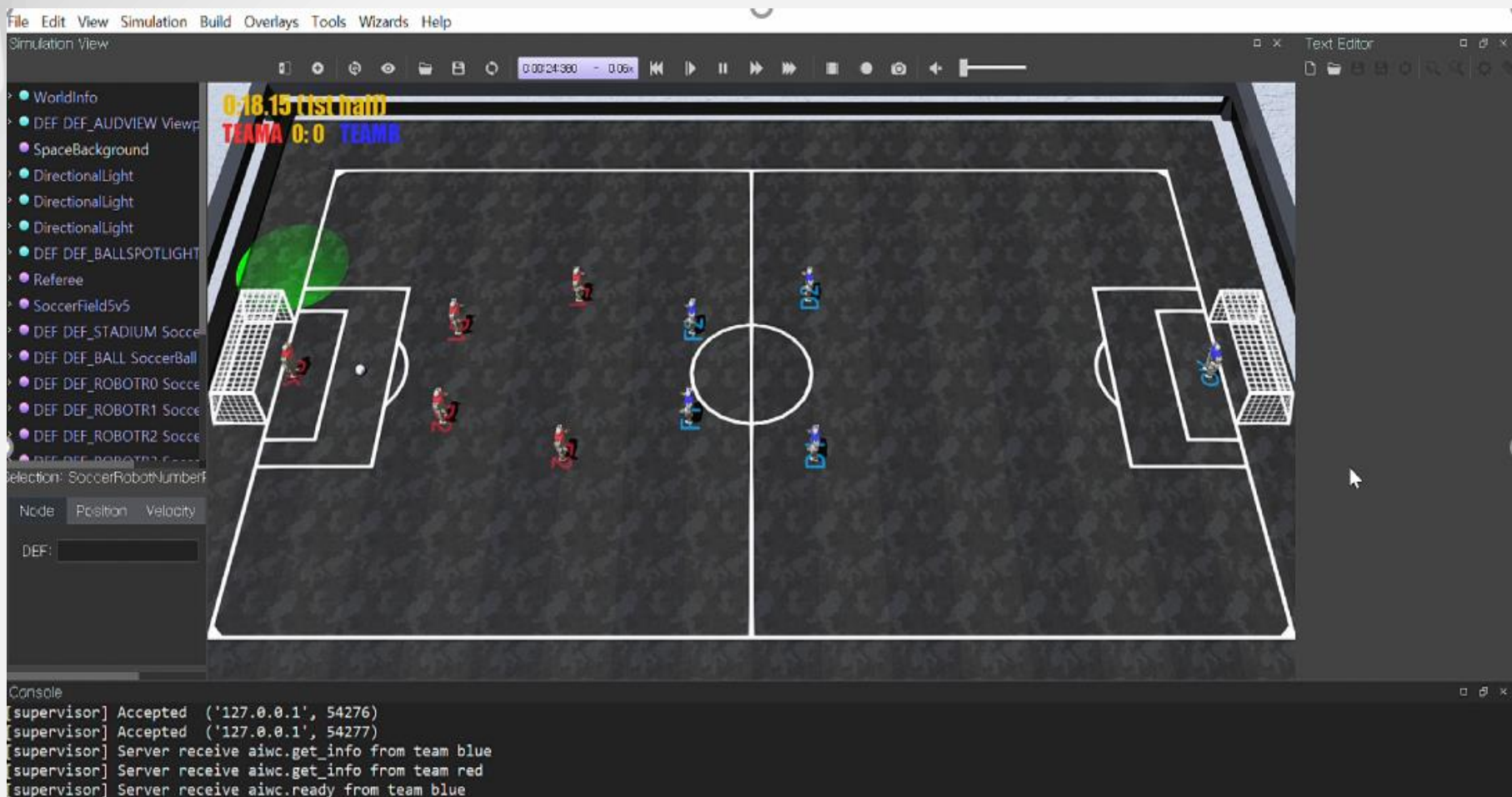
우리 팀 모두 상대팀 골대에 일렬로 서서 상대팀 득점 획득하는 Agent



공이 전혀 움직이지 않는 등의 문제

1

우리 팀 모두 상대팀 골대에 일렬로 서서 상대팀 득점 획득하는 Agent



경기가 Reset 되는 문제

2

Penalty Area 밖에서 수비하는 Agent

- Player_rulebasedC를 기반으로 Agent 구현
- 골키퍼를 제외한 우리 팀 4명 모두 각각 지정된 Penalty Area 밖에서 수비하는 Agent

Penalty Area 밖에서 수비하는 Agent

■ 방법

■ 각 포지션별로 위치하는 영역 변경

```
# GK_ZONE
if helper.ball_is_gk_zone(predicted_ball, self.field, self.goal_area):
    # GK
    gk_control = self.action.defend_ball(self.gk_index)
    if gk_control == None:
        if (cur_posture[self.gk_index][BALL_POSSESSION]):
            gk_control = self.action.shoot_to(self.gk_index, 0, 0, 10, 10)
        else:
            if -self.field[X]/2 - 0.05 < cur_posture[self.gk_index][X] < -self.field[X],
                gk_control = self.action.turn_to(self.gk_index, 0, 0)
            else:
                gk_control = self.action.go_to(self.gk_index, -self.field[X]/2, 0)

# D1
d1_control = self.action.go_to(self.d1_index, -5.0, self.penalty_area[Y]/2)
# D2
d2_control = self.action.go_to(self.d2_index, -5.0, -self.penalty_area[Y]/2)
# F1
f1_control = self.action.go_to(self.f1_index, -3.9, 1)
# F2
f2_control = self.action.go_to(self.f2_index, -3.9, -1)
self.flag = 1
```

2

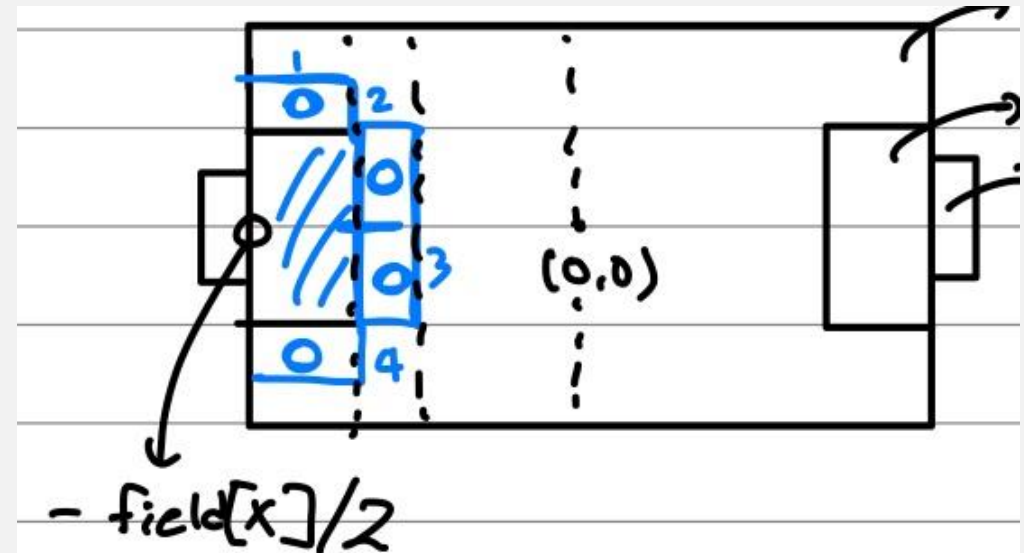
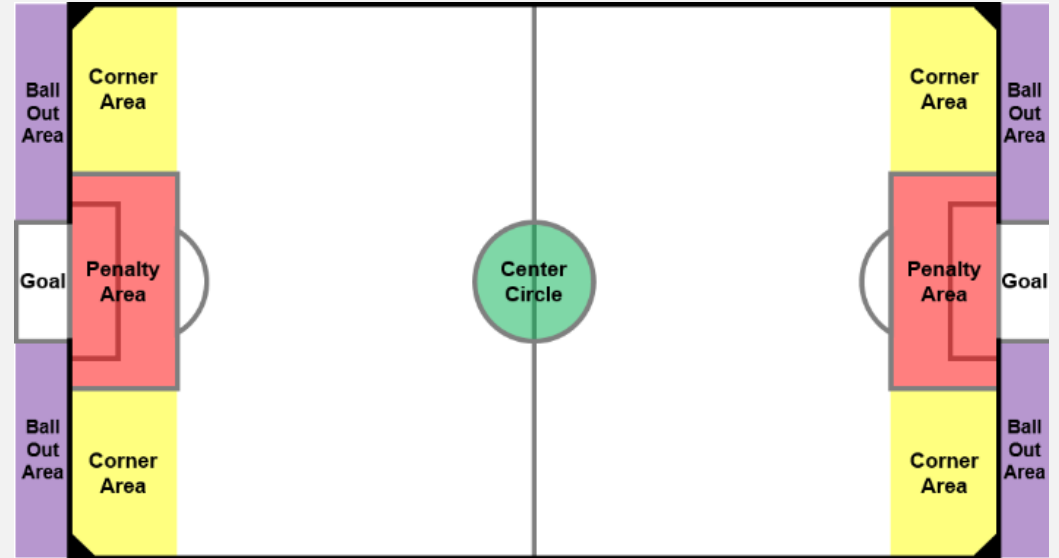
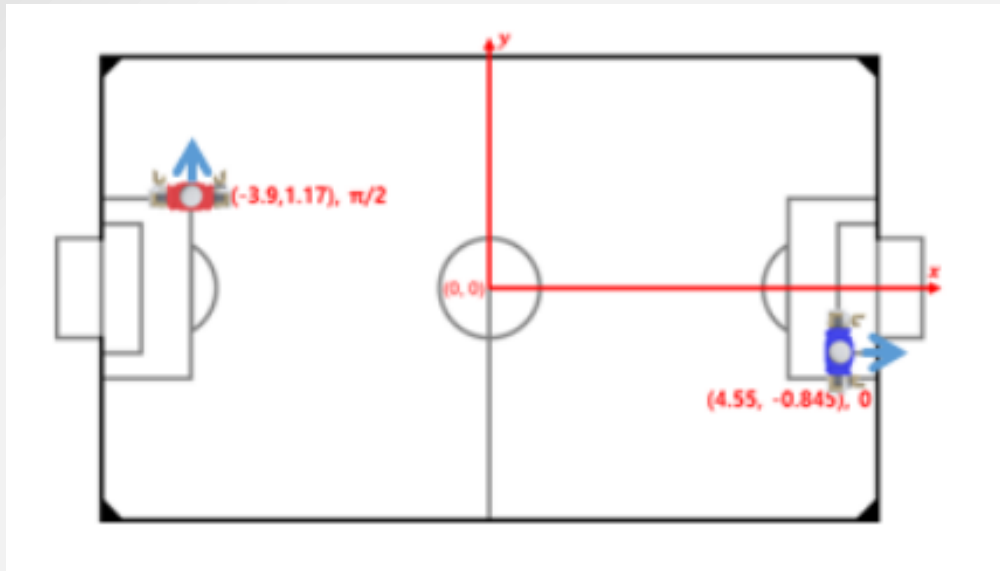
Penalty Area 밖에서 수비하는 Agent

- 방법
 - 골키퍼는 원래 위치에서 기존 역할 수행
 - 특수한 상황(PENALTYKICK, KICKOFF, CORNERKICK 등)을 제외하고, 각자 정해진 영역에서 수비 역할 수행
- 상대 : rulebasedC
- 포지션별로 위치하는데 성공

2

Penalty Area 밖에서 수비하는 Agent

■ 방법

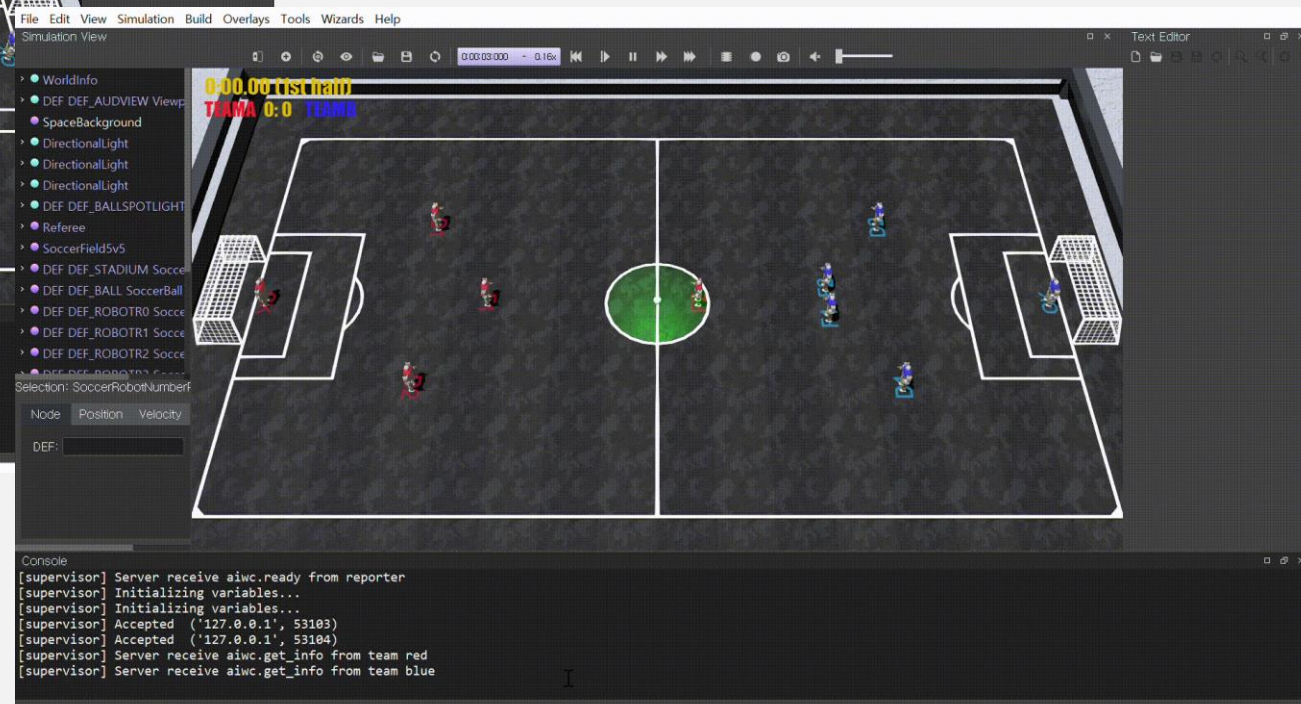
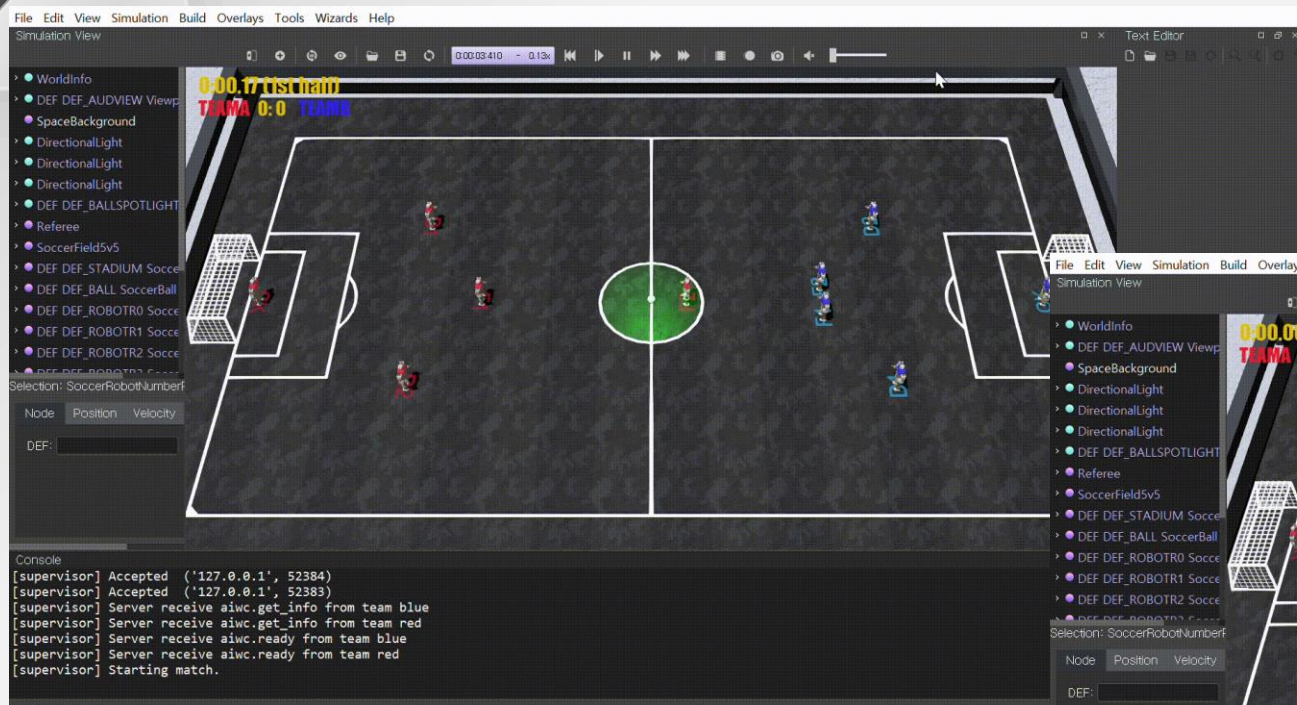


Penalty Area 밖에서 수비하는 Agent

- 문제점
- 골을 막다가 특수한 상황(PENALTYKICK, KICKOFF, CORNERKICK 등)이 발생하기도 함 – 수비에 방해
 - 예 : 수비수가 수비를 하다 Penalty_Area에 들어가는 경우 Penalty Kick 발생
- 공이 수비 영역 내 있지 않으면 Delay가 발생
- 해당 영역이 아닌 위치에서 수비를 진행하게 되어 수비 범위 내 이동이 자유롭지 못함

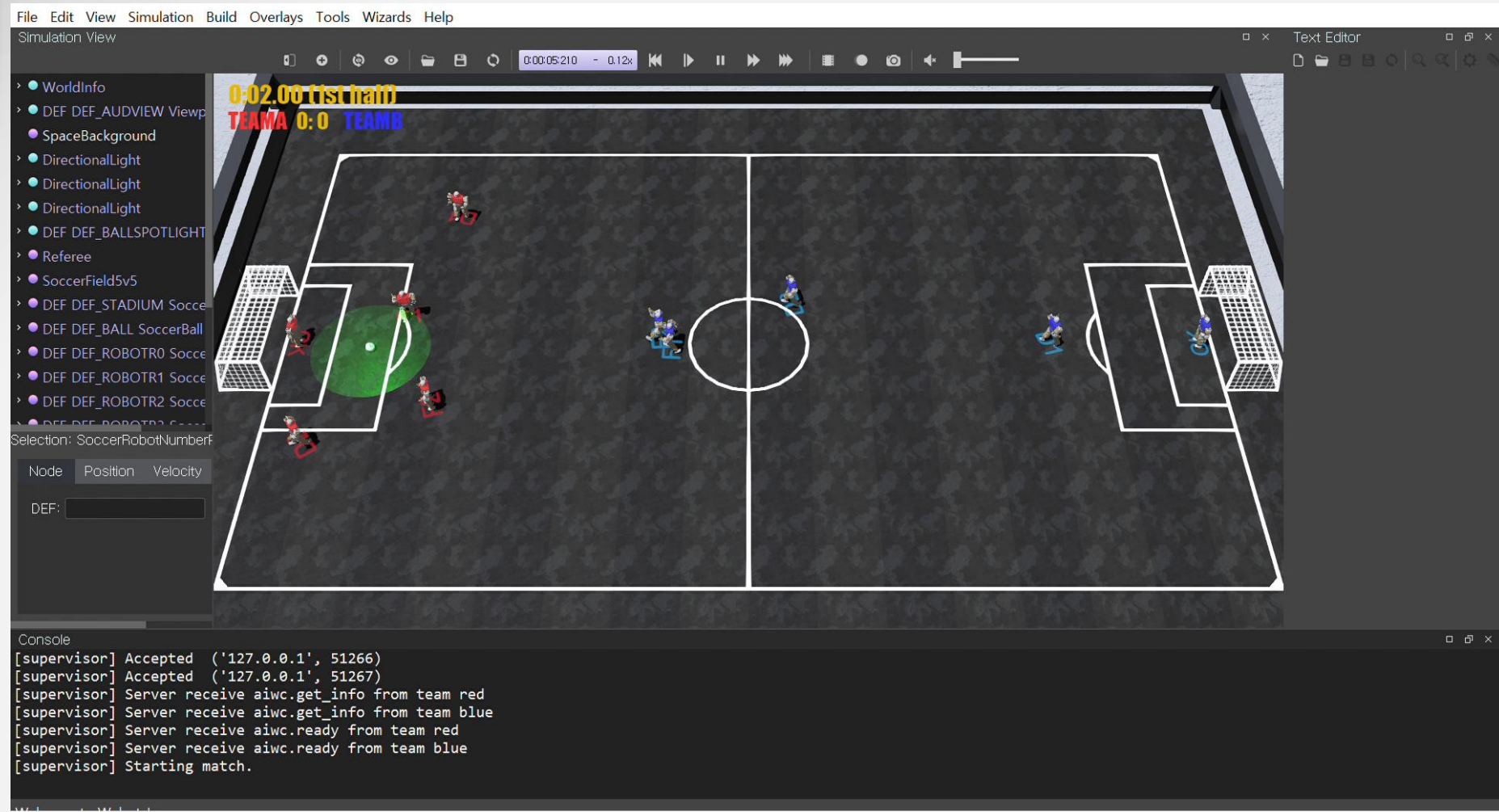
2

Penalty Area 밖에서 수비하는 Agent - 영상



- 골을 막다가 특수한 상황(PENALTYKICK, KICKOFF, CORNERKICK 등)이 발생하기도 함
 - Corner Kick 발생/ Penalty_Area에 들어가는 경우 Penalty Kick 발생

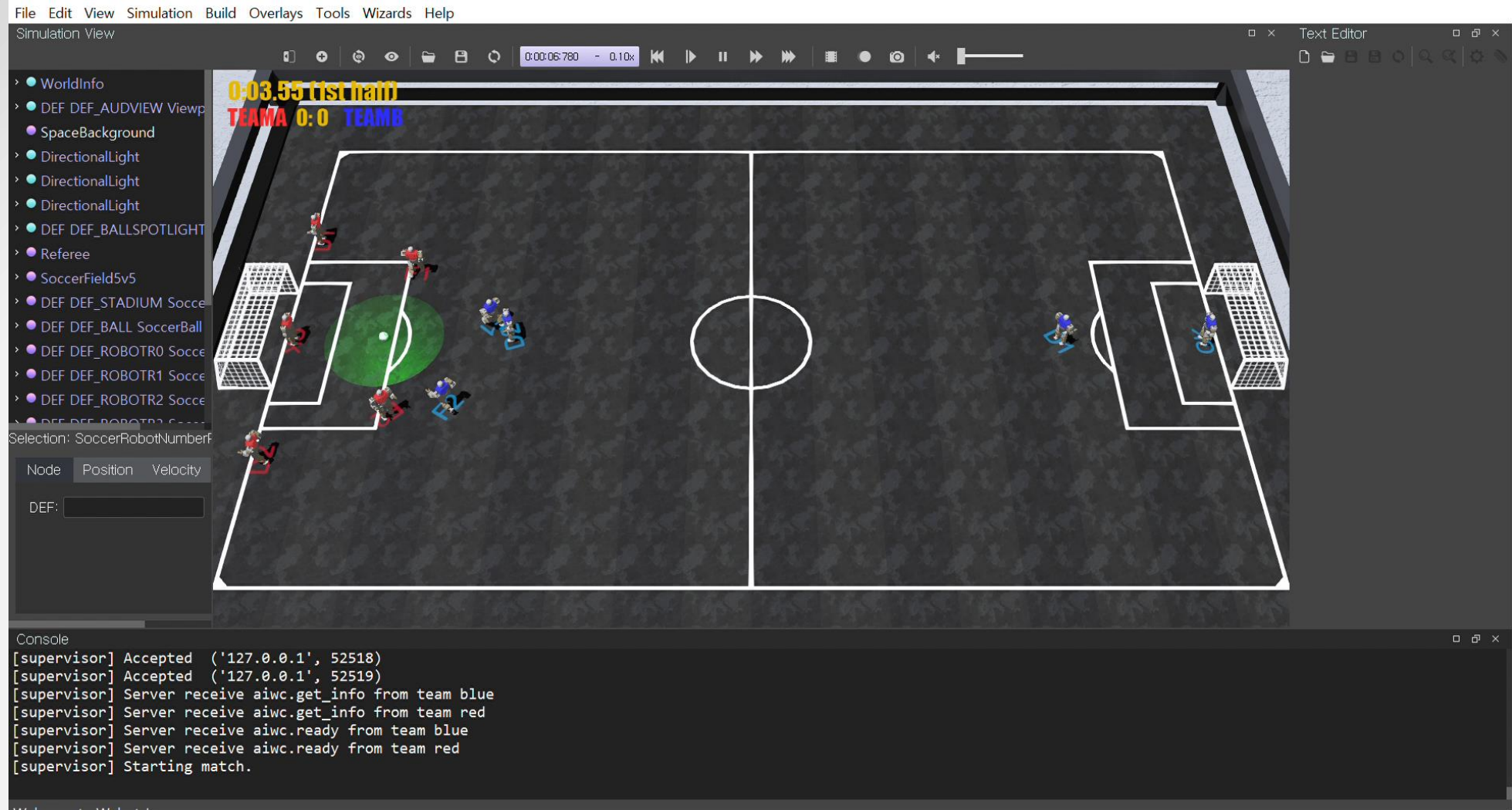
Penalty Area 밖에서 수비하는 Agent - 영상



공이 수비 영역 내 있지 않으면 Delay가 발생

2

Penalty Area 밖에서 수비하는 Agent - 영상



- 영역 내 수비가 제대로 이뤄지지 않은 경우