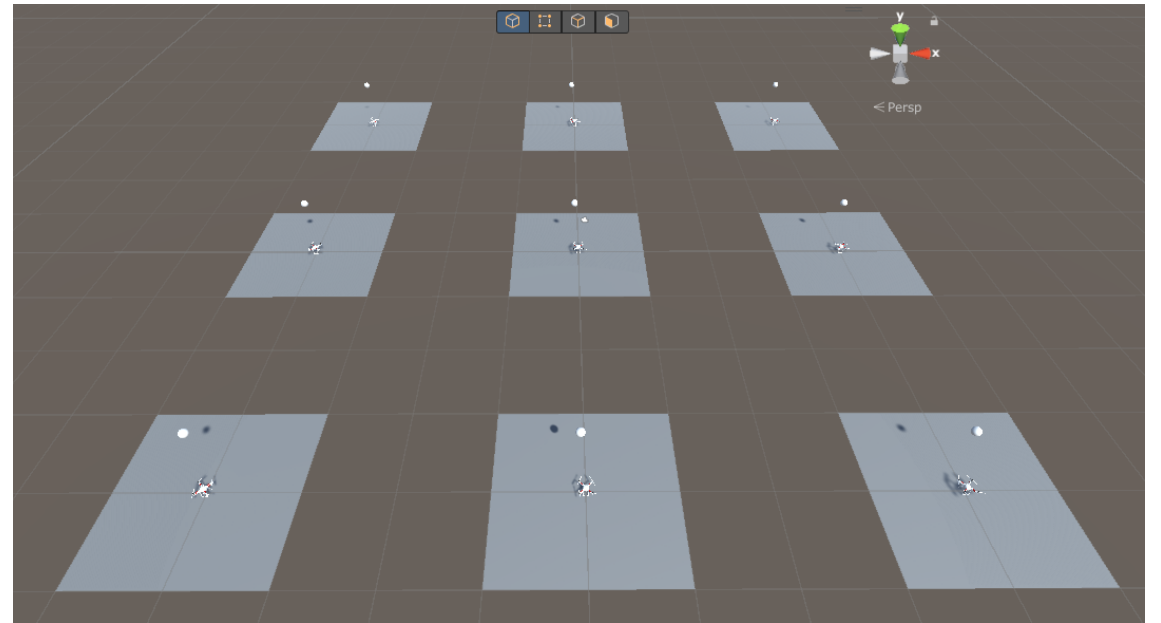
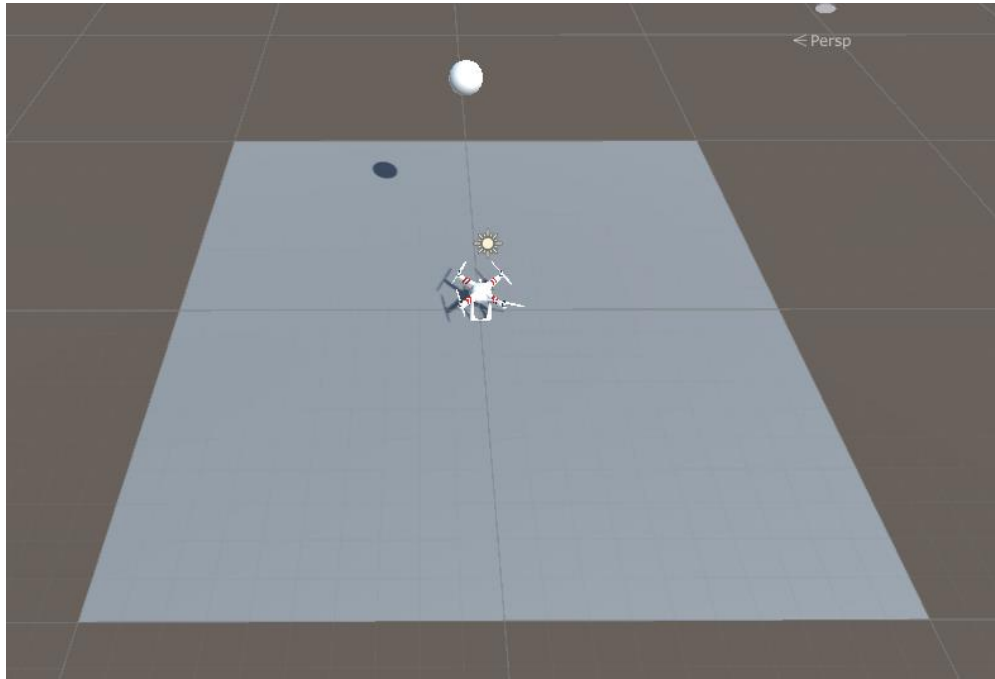


ml-agent를 활용한 드론 제어



2019305050 이제희

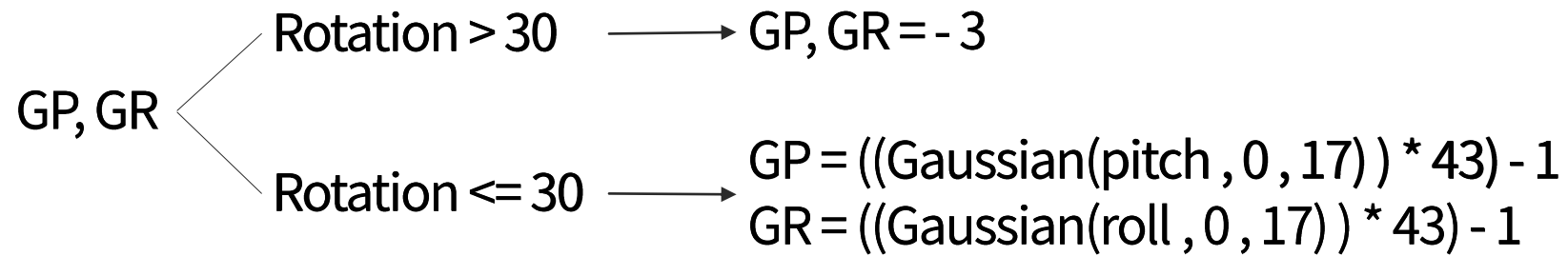
Unity 환경



모델 상태

state		value
Target location	x	-5 ~ 5
	y	9
	z	-5 ~ 5
Drone location	x	- ∞ ~ ∞
	y	
	z	
Drone rotation	roll	0 ~ 360
	pitch	
	yaw	
Drone Velocity (m/s)	fr	0 ~ 6
	fl	0 ~ 6
	br	0 ~ 6
	bl	0 ~ 6

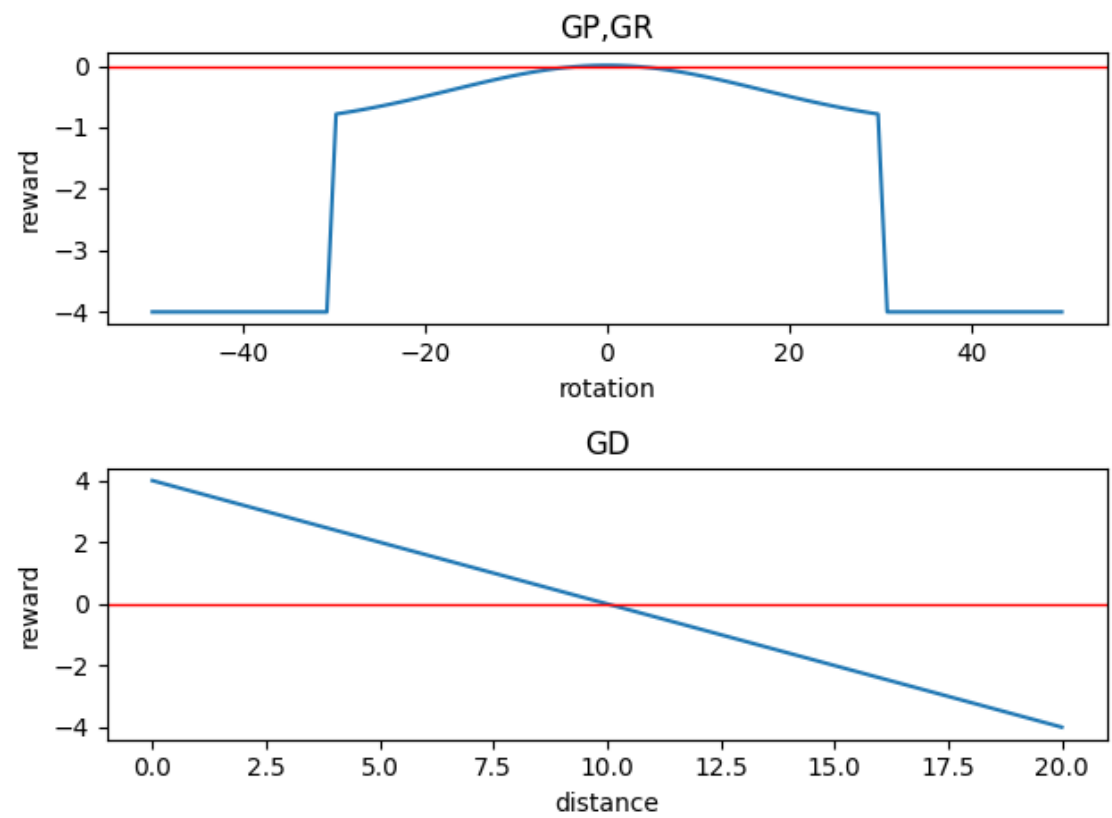
Reward



GD — Distance ≥ 0 → $GD = - (4 * distance / fix_distance) + 4$

* fix_distance:
episode 초기 distance를 구해둔 것

Reward



* Fix_distance=10인 경우

		GP	GR
reward	Rotation<=30	-0.787.. ~ 0	0.787.. ~ 0
	Rotation>30	-4	-4

		GD
reward	Distance>0	$-\infty \sim 4$



Reward 추가

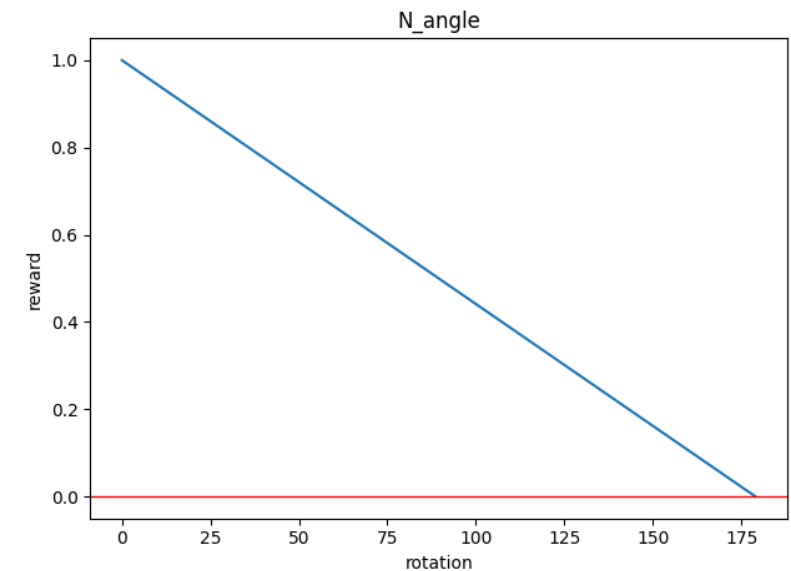
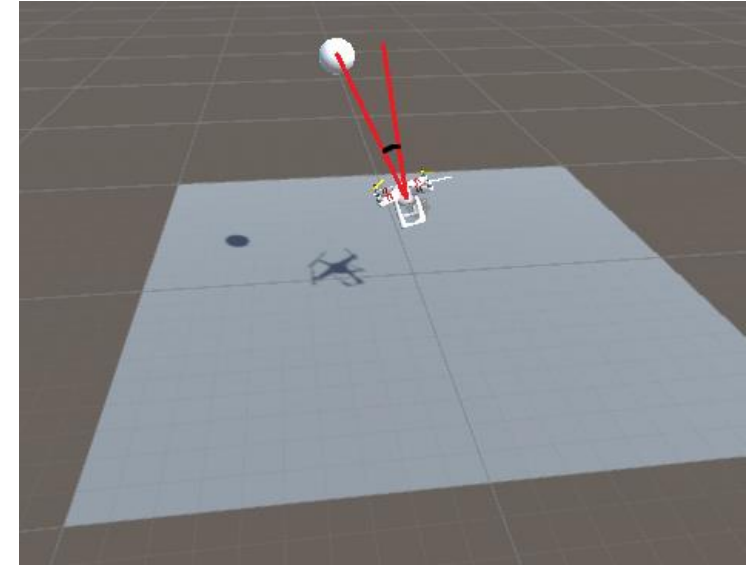
N_angle 추가

Angle= Drone2target vector – Drone direction vector

$N_angle = 1 - \text{angle} / 180$

N_angle 범위: 0 ~ 1

-> 드론의 방향까지 고려하기 위함



Reward

$$\text{Reward} = \text{GP} + \text{GR} + \text{GD} + \text{N_angle}$$

Step마다 -0.1

×

if

Pitch & roll < 90°

Reward = -1000

에피소드 종료

Drone location x, z > 10

Reward = -1000

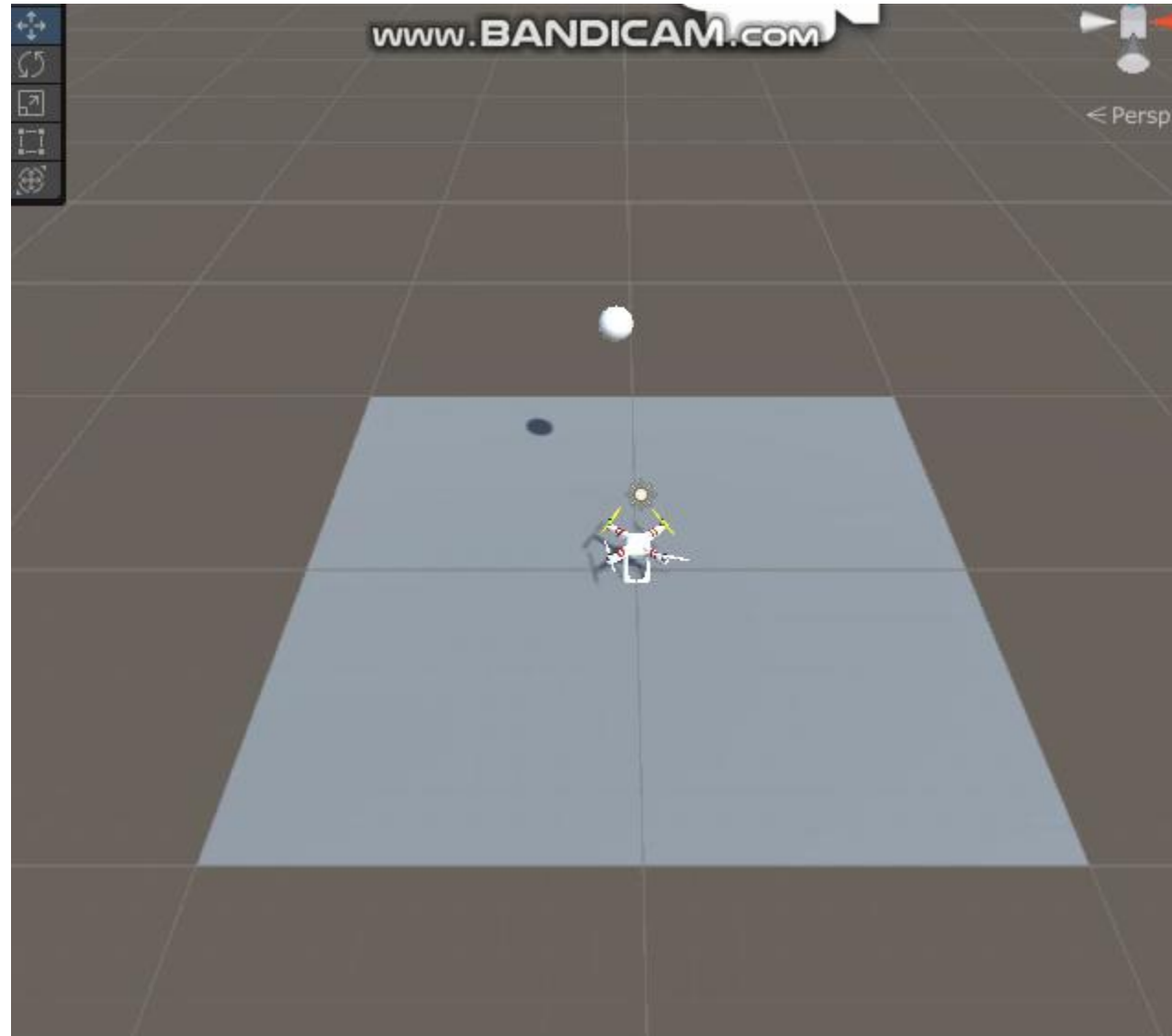
에피소드 종료

Step > 5000

Reward -= 1000

에피소드 종료

×



수행해본 것들

Drone 날개 회전 (실패)

×
Unity에서 회전으로 양력 생성 가능
그러나 너무 빠르면 물리엔진 한계로 인해 수행 불가능

드론의 몸통과 날개가 분리되는 현상 발생

이 부분은 더 이상 사용하지 않을 예정

수행해본 것들

Yaw 제어 (실패)

Vector의 x만 사용

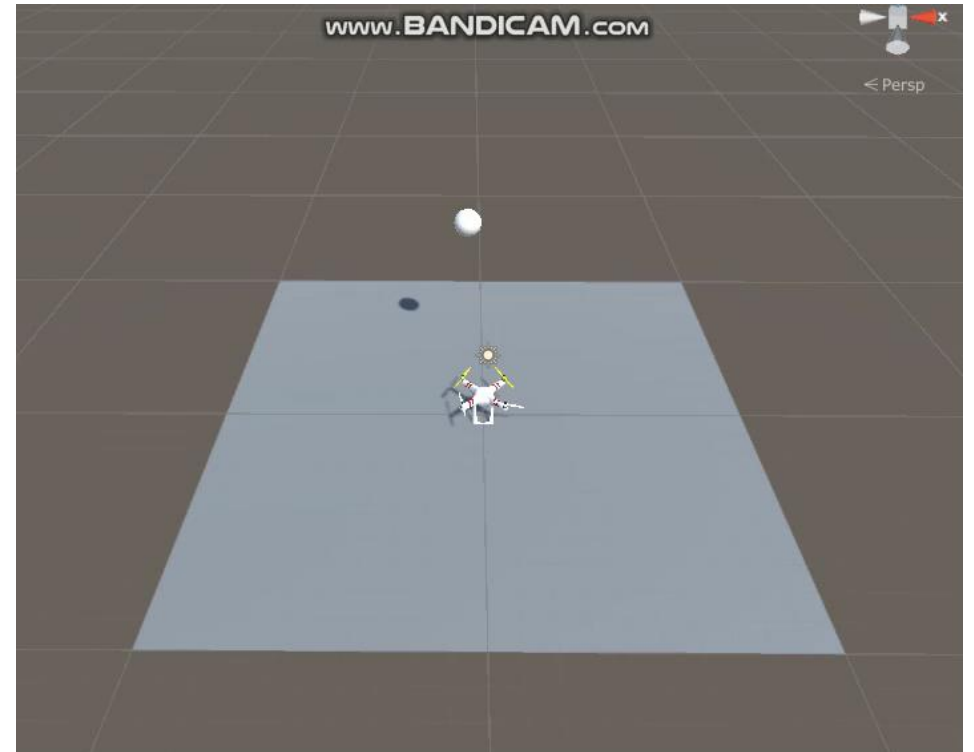
×

$GY = \text{Target direction Vector} - \text{Drone forward Vector}$

GY 표준화(-1 ~ 1)

$\text{Reward} = GP + GR + GD + GY$

계속 시도해볼 예정



수행해본 것들

Target 2개 생성

Target1, 2를 생성

×

Target1에 접촉하면 목표를 Target2로 변경 후 진행
Target2에 접촉시 episode종료

결과가 괜찮았던 $\text{reward} = GP + GR + GD + N_angle$ 로 진행

좀 더 보완 필요(보상 비율, 늘어난 step 수 등)

