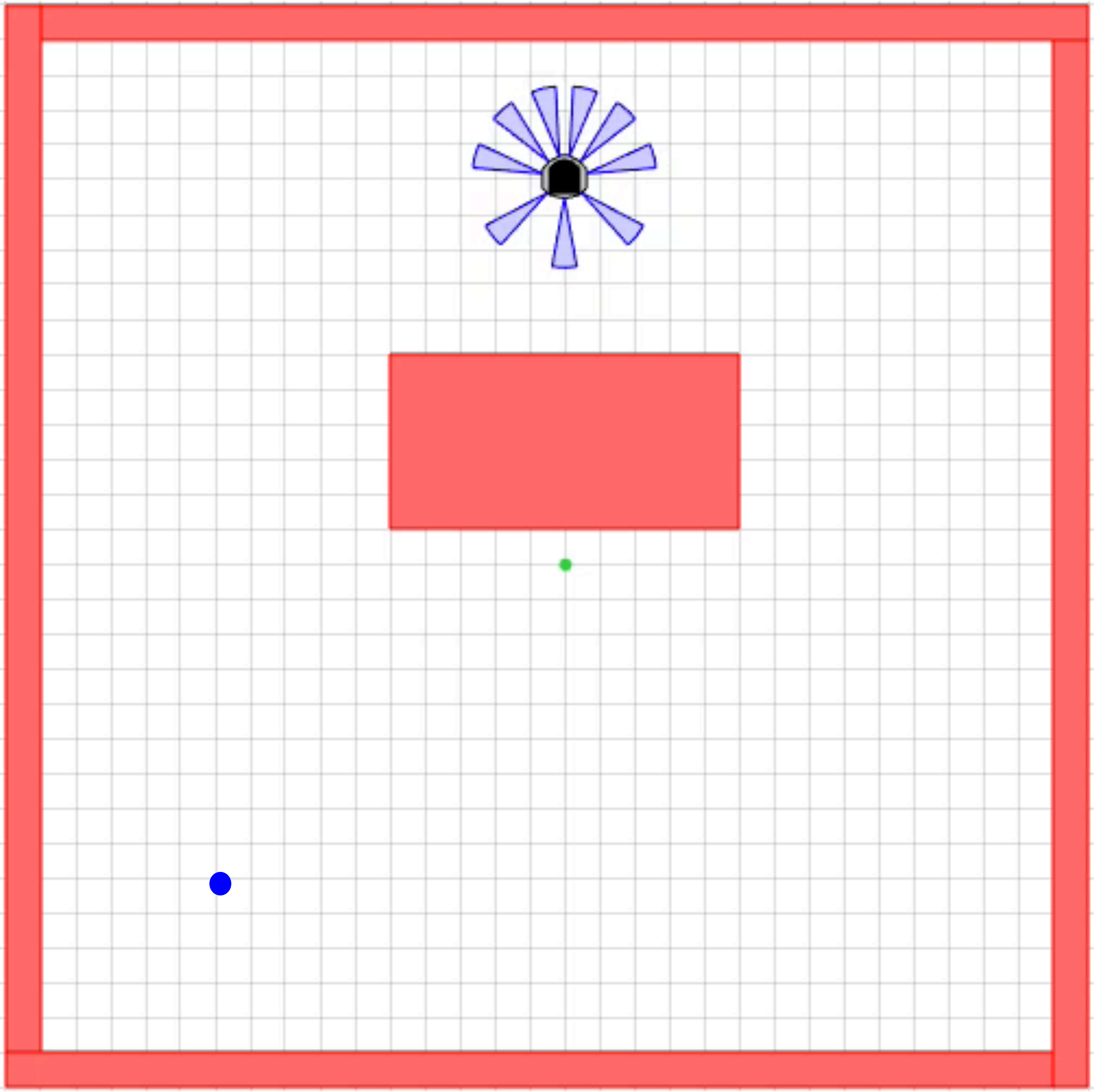
電気情報工学基礎演習B

Simulation of Controlling Mobile Robot
Lecture 5

Instructor:　蔡　凱
TA:　笠原　萌人



Two methods

- Method 1:

  Mixed GoToGoal & AvoidObstacles control

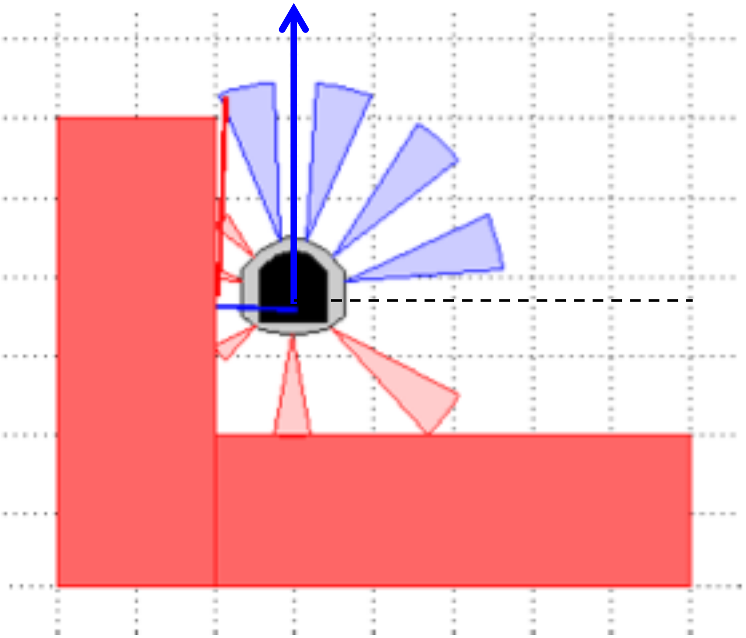
- Method 2:

  Switching between different controllers

Go To Goal & Avoid Obstacle

- Objective: steer the robot to reach a goal *and* avoid nearby obstacles

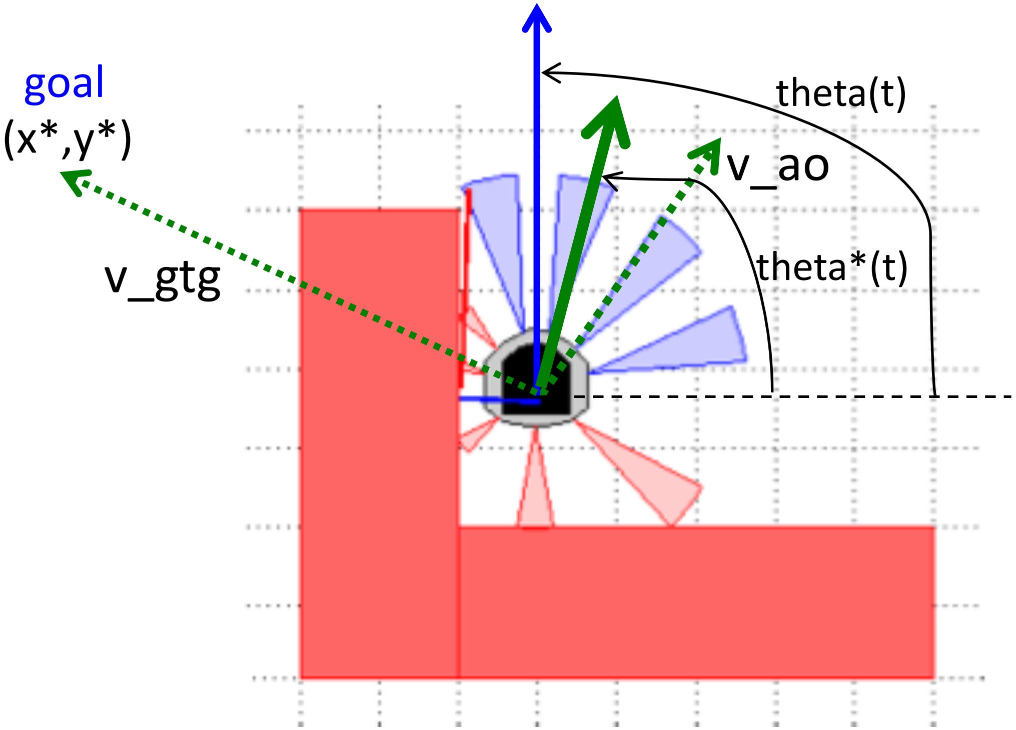
goal
(x*,y*)



Assume robot is moving at linear velocity v=constant.

We only control robot's angular velocity w: $\frac{d\theta}{dt} = u(t)$, control input

AOandGTG Controller



$$v^*(t) = \alpha \cdot \text{v\_ao}(t) + (1-\alpha) \cdot \text{v\_gtg}(t), \quad \alpha \in (0,1)$$

$$\theta^*(t) = \tan^{-1}\left(\frac{v_y^*(t)}{v_x^*(t)}\right)$$

Use P-controller $u(t) = K_p(\theta^*(t) - \theta(t))$ to achieve $\theta(t) \to \theta^*(t)$

Code

- +simiam/+controller/+khepera3/K3Supervisor.m
  - function obj=K3Supervisor()

```
% Input your code below %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Specified (constant) speed
obj.v = 0; (change this to see what happens)
%Goal location
obj.goal = [-1,-1]; (change this to see what happens)
%Stop condition
obj.d_stop = 0.1; (change this to see what happens)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Code

- +simiam/+controller/AOandGTG.m
  - function obj=AOandGTG()

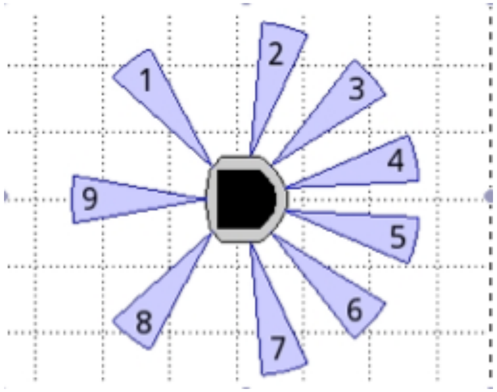
```
% Input your code below %
%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Propositional control gain %
obj.Kp = 0; (change this to see what happens)
%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Code

- +simiam/+controller/AOandGTG.m
  - function outputs = execute(...)

```
% Input your code below %
%%%%%%%%%%%%%%%%%%%%
% Weights for the nine sensors %
sensor_gains = [1 1 1 1 1 1 1 1 1];
(change this to see what happens)
%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%
% Weight for blending the gtg and ao vectors %
alpha = 0.5; (change this to see what happens)
u_ao_gtg = alpha*u_gtg+(1-alpha)*u_ao;
%%%%%%%%%%%%%%%%%%%%
```



Exercises

- Use package: simiam_lecture5.zip
- Change robot’s initial pose in settings.xml
- Set robot’s linear speed, goal location, and stop distance in K3Supervisor.m
- Adjust control gain parameter in AOandGTG.m
- Adjust sensor weights in AOandGTG.m
- Adjust controller blending weight in AOandGTG.m

Two methods

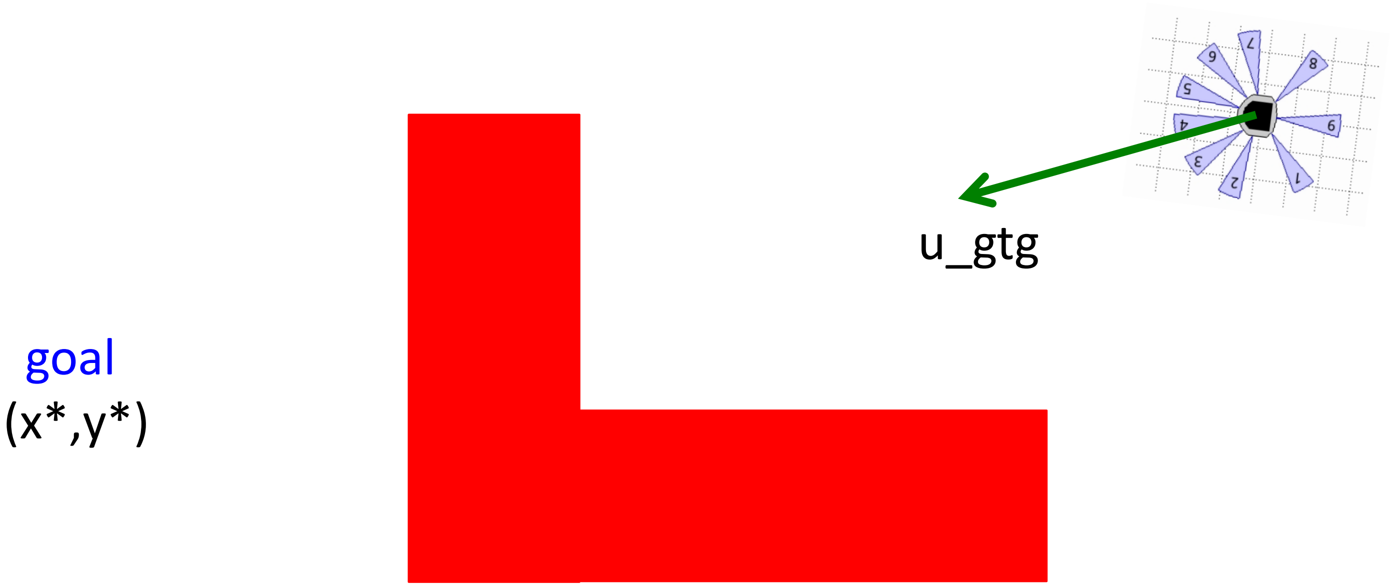
- Method 1:

  Mixed GoToGoal & AvoidObstacles control

- Method 2:

  Switching between different controllers

No Obstacle Around



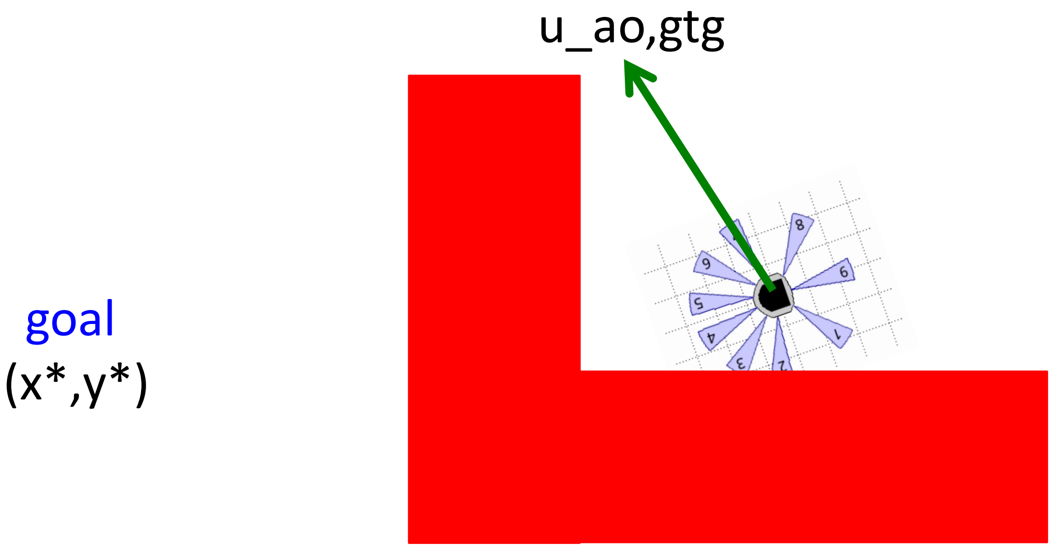
u_gtg

goal

(x*,y*)

Initially, use GoToGoal control.

Obstacle Detected



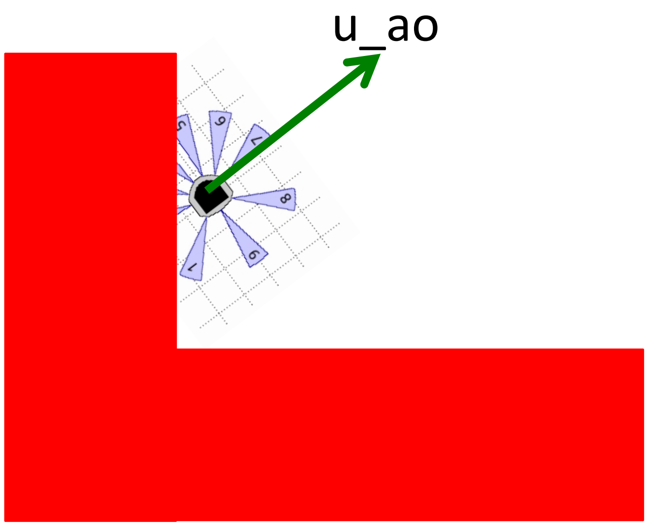
u_ao,gtg

goal
(x*,y*)

When close to an obstacle, use AOandGTG control.

Too Close To Obstacle

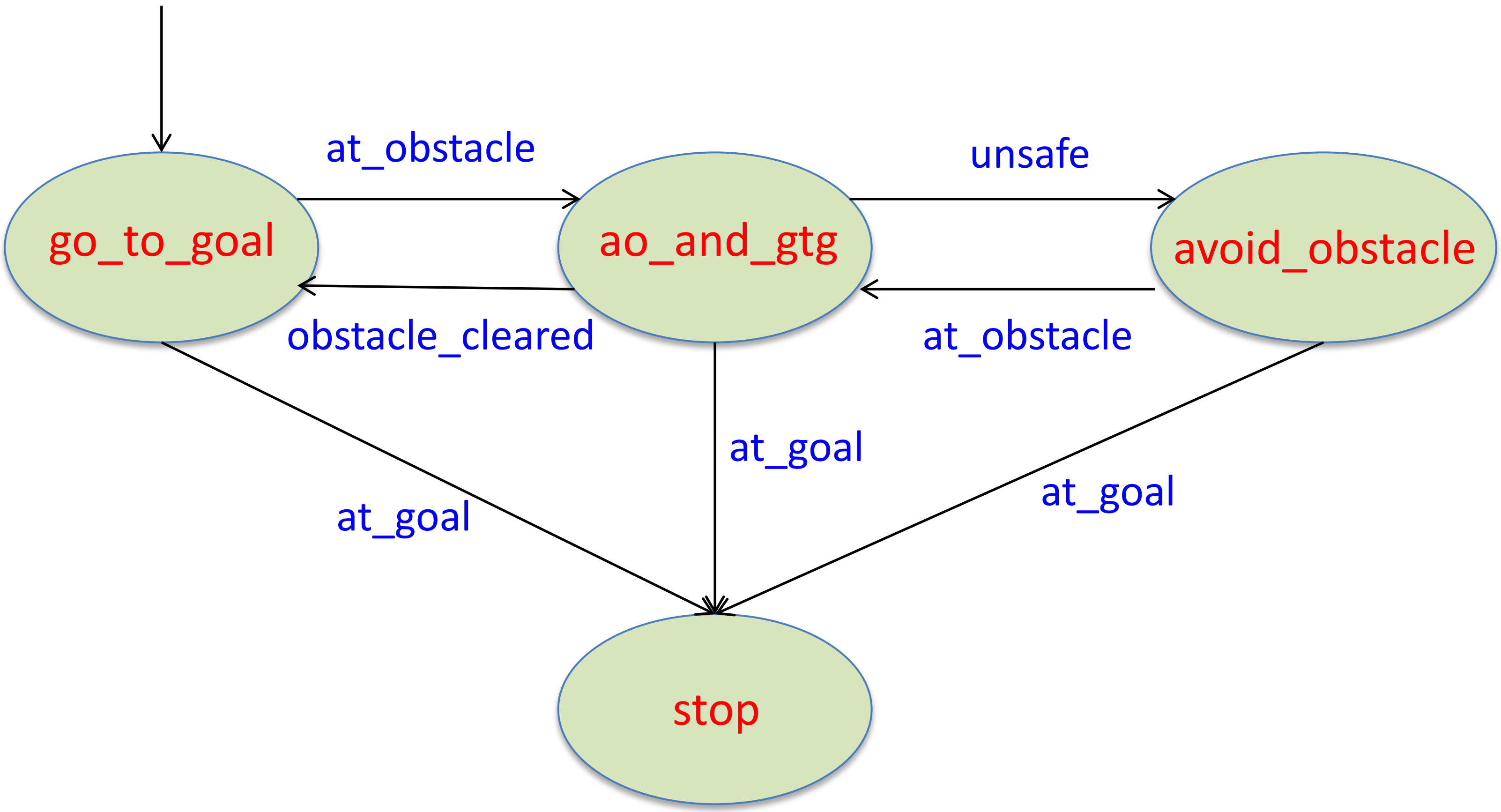
goal
(x*,y*)



u_ao

If too close to an obstacle (unsafe), use AvoidObstacle control.

Switching Controller



Code

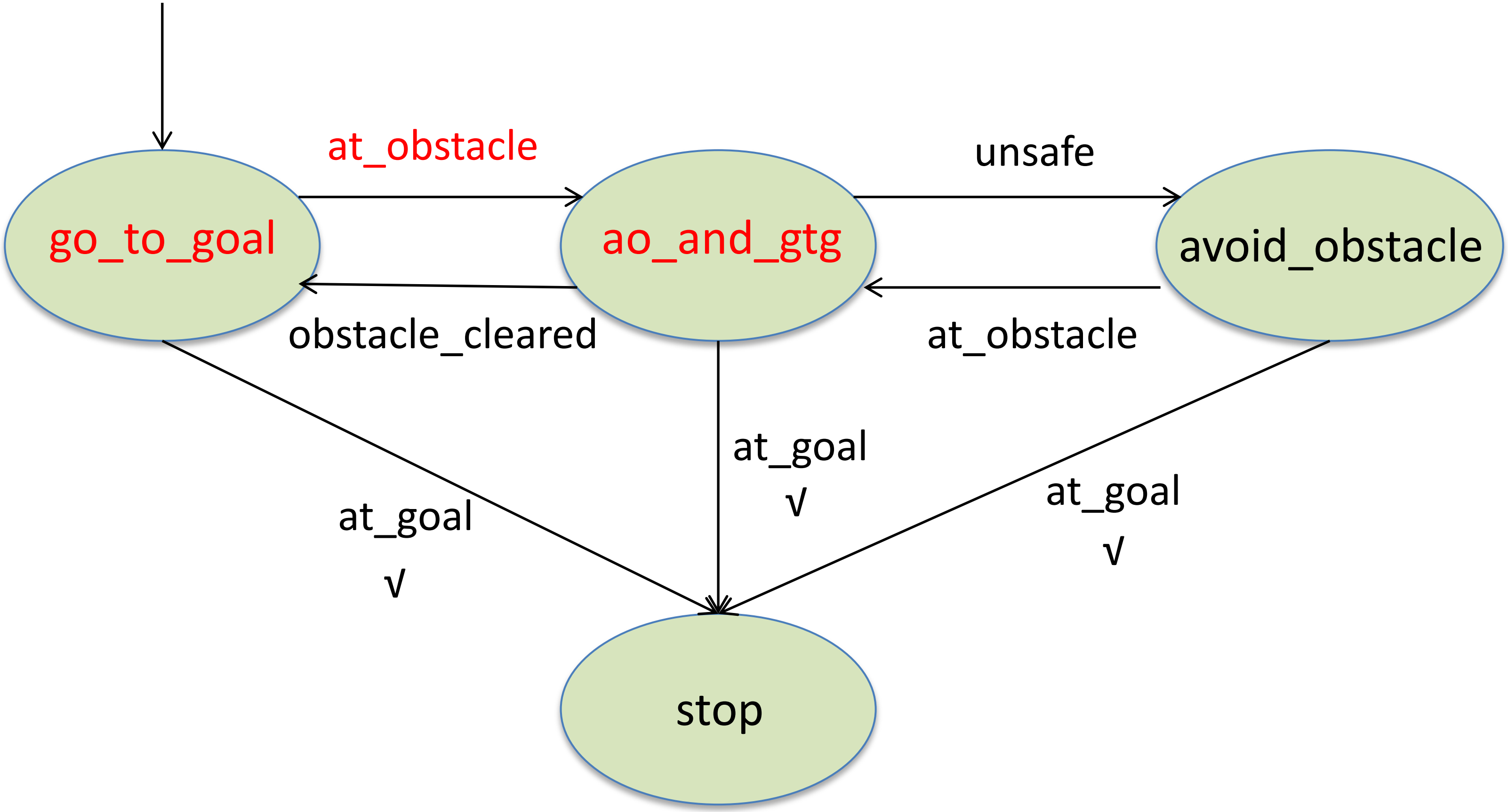
- +simiam/+controller/+khepera3/K3Supervisor.m
  - function obj=K3Supervisor()

```
% Input your code below %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Specified (constant) speed
obj.v = 0; (change this to see what happens)
%Goal location
obj.goal = [-1,-1]; (change this to see what happens)
%Stop condition
obj.d_stop = 0.1; (change this to see what happens)
%Distance close to obstacles
obj.d_at_obs = 0.1; (change this to see what happens)
%Distance too close to obstacles (unsafe)
obj.d_unsafe = 0.1; (change this to see what happens)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Code

- +simiam/+controller/+khepera3/K3Supervisor.m
  - function execute(obj, dt)

```
% Input your code below %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% At controller 'go_to_goal', if 'at_obstacle' is true,
% then switch to controller 'ao_and_gtg’
    if(obj.is_in_state('go_to_goal'))
        if(obj.check_event('at_obstacle'))
            obj.switch_to_state('ao_and_gtg');
        end
    end
% Follow this example write all other switching rules
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```



```
% At controller 'go_to_goal', if 'at_obstacle' is true,
% then switch to controller 'ao_and_gtg'
        if(obj.is_in_state('go_to_goal'))
                if(obj.check_event('at_obstacle'))
                        obj.switch_to_state('ao_and_gtg');
                end
        end
```

Code

- +simiam/+controller/GoToGoal.m
  - function obj=GoToGoal(): Lecture 3

- +simiam/+controller/AvoidObstacles.m
  - function obj = AvoidObstacles(): Lecture 4

- +simiam/+controller/AOandGTG.m
  - function obj=AOandGTG(): this Lecture

Exercises

- Use package: simiam_lecture5.zip
- Change robot's initial pose in settings.xml
- Set robot's linear speed, goal location, and stop distance, distance close to obstacles, distance too close to obstacles in K3Supervisor.m
- Design controller switching logic in K3Supervisor.m
- Adjust parameters in GoToGoal.m, AvoidObstacles.m, and AOandGTG.m

Task

- Set robot’s pose (-1,1,0) and the following two obstacles in settings.xml; set robot’s goal location [-1,0] in K3Supervisor.m

```
% Obstacle 1
    <pose x="-0.5" y="-0.8" theta="0.79" />
    <geometry>
      <point x="0" y="0" />
      <point x="1" y="0" />
      <point x="1" y="0.5" />
      <point x="0" y="0.5" />
    </geometry>
% Obstacle 2
    <pose x="-1" y="0.26" theta="0.0" />
    <geometry>
      <point x="0" y="0" />
      <point x="2" y="0" />
      <point x="2" y="0.5" />
      <point x="0" y="0.5" />
    </geometry>
```

- Find suitable mixed control and/or switching control that works “smoothly”