

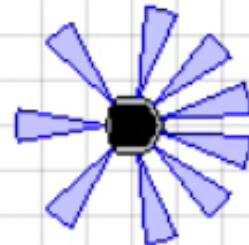
電気情報工学基礎演習B

Control a Mobile Robot: Lecture 2

Instructor: 蔡凱

TA: 川村聰志

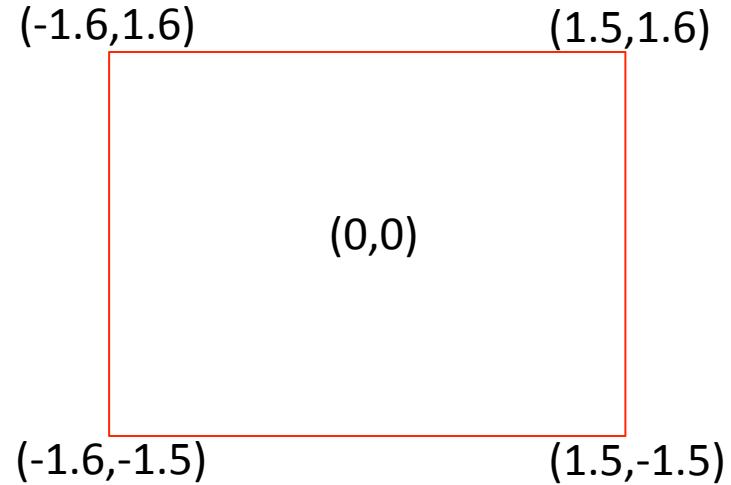
Control robot
to drive at
45 degrees



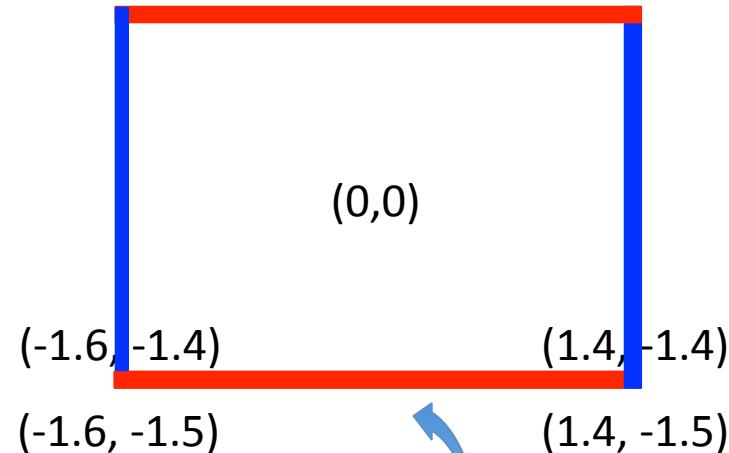
Lecture 2

- Simulator configuration
- Robot dynamics
- GoToAngle control

Simulator Configuration

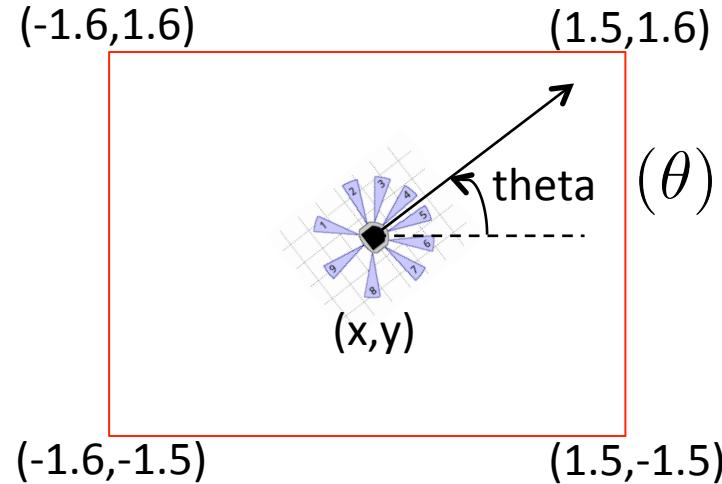


Simulator Configuration



- `settings.xml`:
 - `<obstacle>`
`<pose x="-1.6" y="-1.5" theta="0" />`
`<geometry>`
`<point x="0" y="0" />`
`<point x="3" y="0" />`
`<point x="3" y="0.1" />`
`<point x="0" y="0.1" />`
`</geometry>`
`</obstacle>`

Robot Location and Orientation

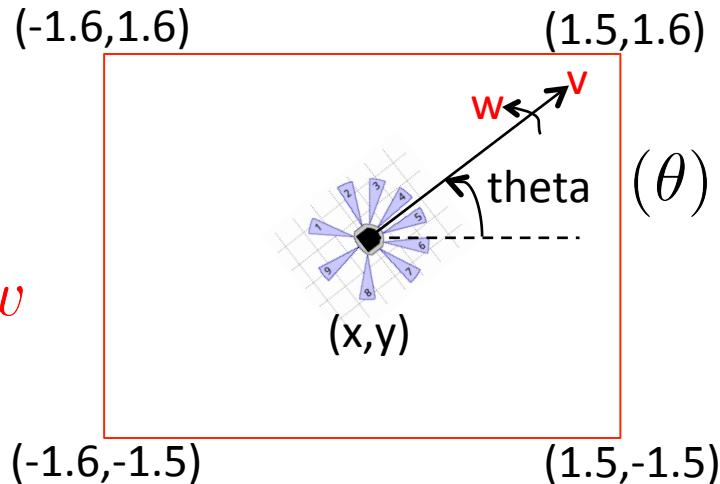


- [settings.xml](#):
 - ```
<robot type="Khepera3">
 <supervisor type="khepera3.K3Supervisor" />
 <pose x="0" y="0" theta="pi/4" />
</robot>
```

# Robot Dynamics

linear velocity:  $v$

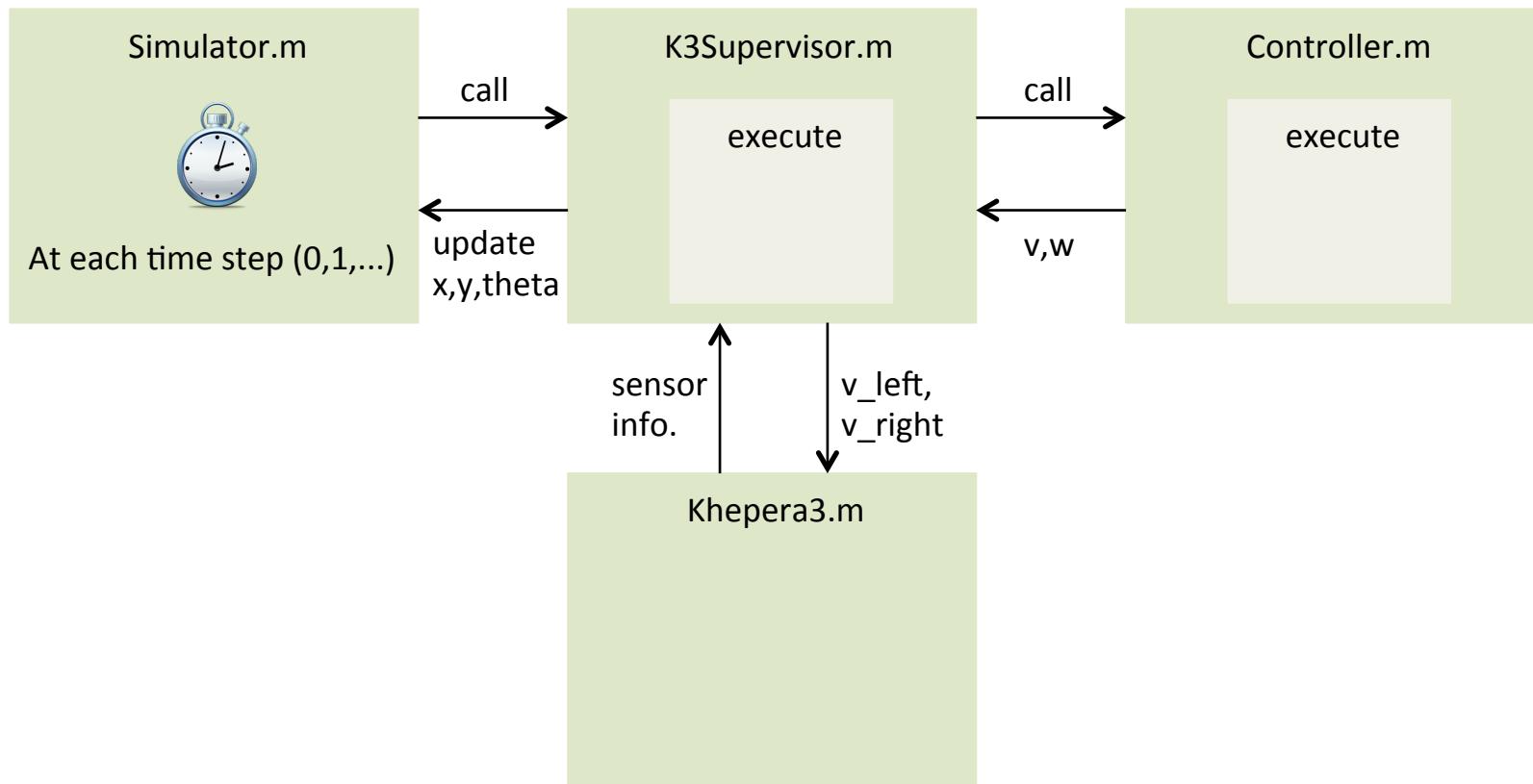
angular velocity:  $w$



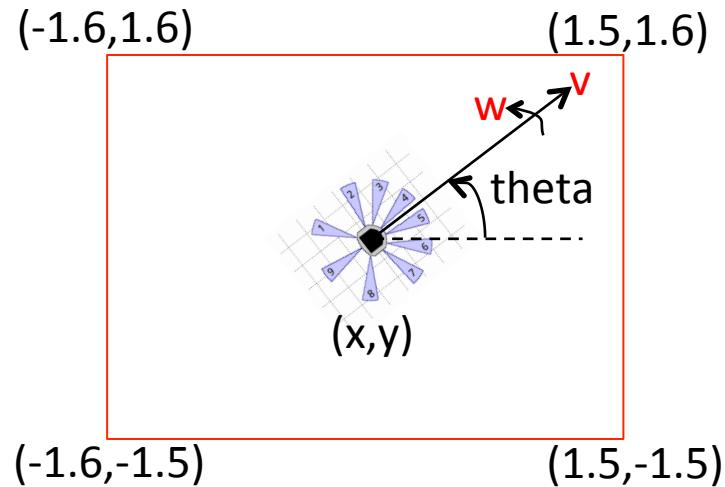
$$\begin{aligned}\frac{dx}{dt} &= v \cos(\theta) \\ \frac{dy}{dt} &= v \sin(\theta) \\ \frac{d\theta}{dt} &= w\end{aligned}$$

- In this course, we set  $v=\text{constant}$  and only control  $w$ .
- Our robot has two-wheel differential drive:  
 $[v_{\text{left}} \ v_{\text{right}}] = f(v, w)$

# How Simulation Works



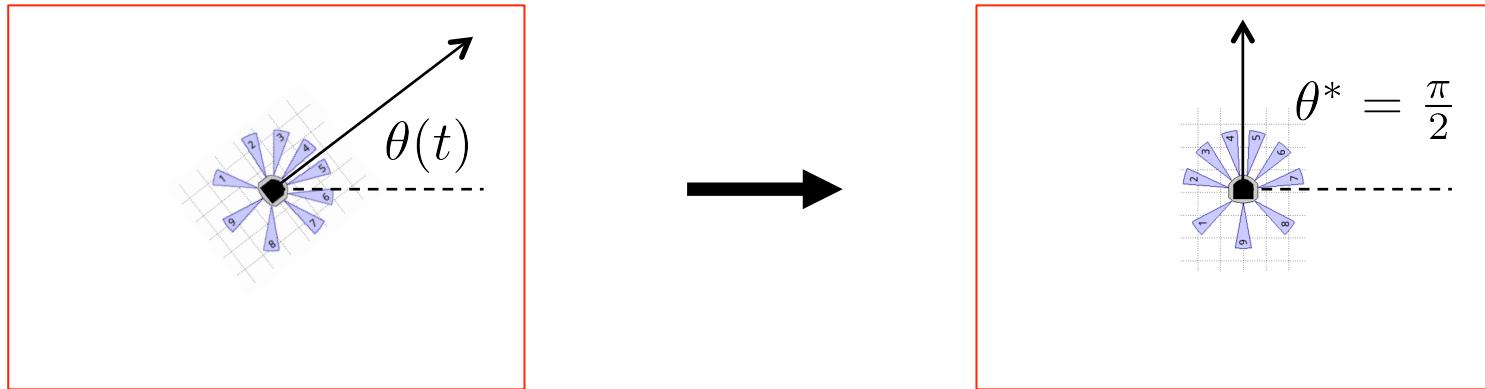
# Robot Dynamics



- Initial robot pose:  $x(0)$ ,  $y(0)$ ,  $\text{theta}(0)$  [in [settings.xml](#)]
- At each time  $t=0, 1, 2, \dots$ , robot pose:  $x(t)$ ,  $y(t)$ ,  $\text{theta}(t)$
- [+simiam/+controller/+khepera3/K3Supervisor.m](#)
  - function [execute\(obj, dt\)](#)
    - `fprintf('current_pose: (%0.3f, %0.3f, %0.3f)\n', x, y, theta);`

# Go To Angle

- Objective: steer the robot to a **specified angle**



Assume robot is moving at linear velocity  $v=\text{constant}$ .

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

Objective:  $\theta(t) \rightarrow \theta^*$  as  $t \rightarrow \infty$     ( $e(t) := \theta^* - \theta(t) \rightarrow 0$ )

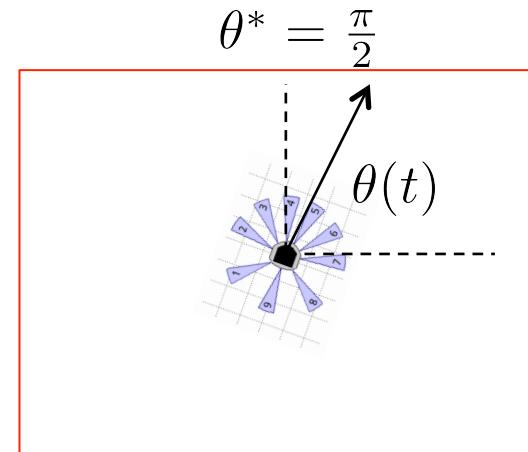
# GoToAngle Controller

- Objective: steer the robot to a **specified angle**

$$\frac{d\theta}{dt} = u(t), \text{ control input}$$

Objective:  $\theta(t) \rightarrow \theta^*$  as  $t \rightarrow \infty$     ( $e(t) := \theta^* - \theta(t) \rightarrow 0$ )

P(roportional)-controller:  $u(t) = K_p e(t)$   
– if  $e(t) > 0$ ,  $u(t) > 0$                           ( $K_p > 0$ )



# GoToAngle Controller

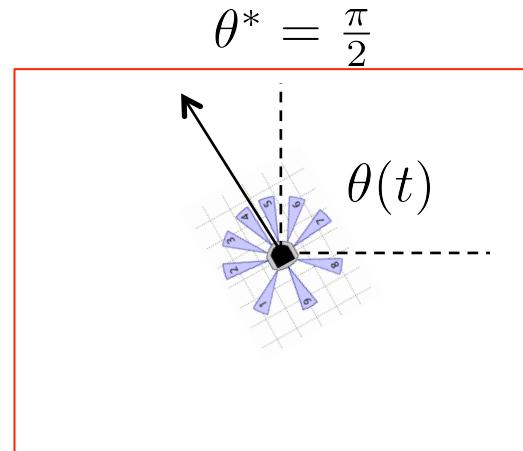
- Objective: steer the robot to a **specified angle**

$$\frac{d\theta}{dt} = u(t), \text{ control input}$$

Objective:  $\theta(t) \rightarrow \theta^*$  as  $t \rightarrow \infty$     ( $e(t) := \theta^* - \theta(t) \rightarrow 0$ )

P(roportional)-controller:  $u(t) = K_p e(t)$

- if  $e(t) > 0$ ,  $u(t) > 0$                          ( $K_p > 0$ )
- if  $e(t) < 0$ ,  $u(t) < 0$



# GoToAngle Controller

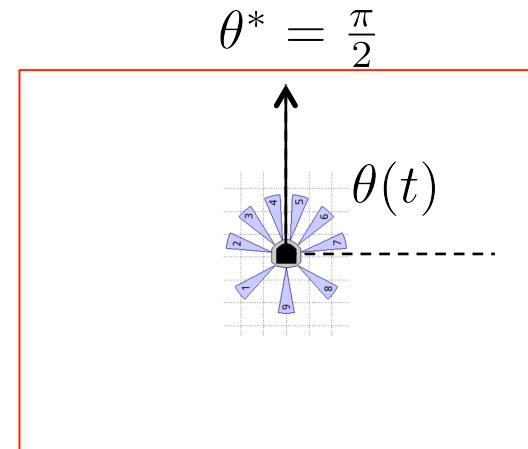
- Objective: steer the robot to a **specified angle**

$$\frac{d\theta}{dt} = u(t), \text{ control input}$$

Objective:  $\theta(t) \rightarrow \theta^*$  as  $t \rightarrow \infty$     ( $e(t) := \theta^* - \theta(t) \rightarrow 0$ )

P(roportional)-controller:  $u(t) = K_p e(t)$

- if  $e(t) > 0$ ,  $u(t) > 0$                               ( $K_p > 0$ )
- if  $e(t) < 0$ ,  $u(t) < 0$
- if  $e(t) = 0$ ,  $u(t) = 0$
- big/small  $e(t)$  yields big/small  $u(t)$



# Code

- +simiam/+controller/+khepera3/  
K3Supervisor.m
  - function `obj=K3Supervisor()`  
`% Input your code below %`  
`%%%%%%%%%%%%%`  
`%Specified angle`  
`obj.theta_d = 0; (change this to see what happens)`  
`%Specified (constant) speed`  
`obj.v = 0; (change this to see what happens)`  
`%%%%%%%%%%%%%`

# Code

- +simiam/+controller/GoToAngle.m

- function `obj=GoToAngle()`

% Input your code below %

%%%%%%%%%%%%%%%

% Propositional control gain %

`obj.Kp = 0;` (change this to see what happens)

%%%%%%%%%%%%%%%

# Tasks

- Change robot's pose in settings.xml
- Set robot's linear speed and desired angle in K3Supervisor.m
- Adjust control gain parameter in GoToAngle.m