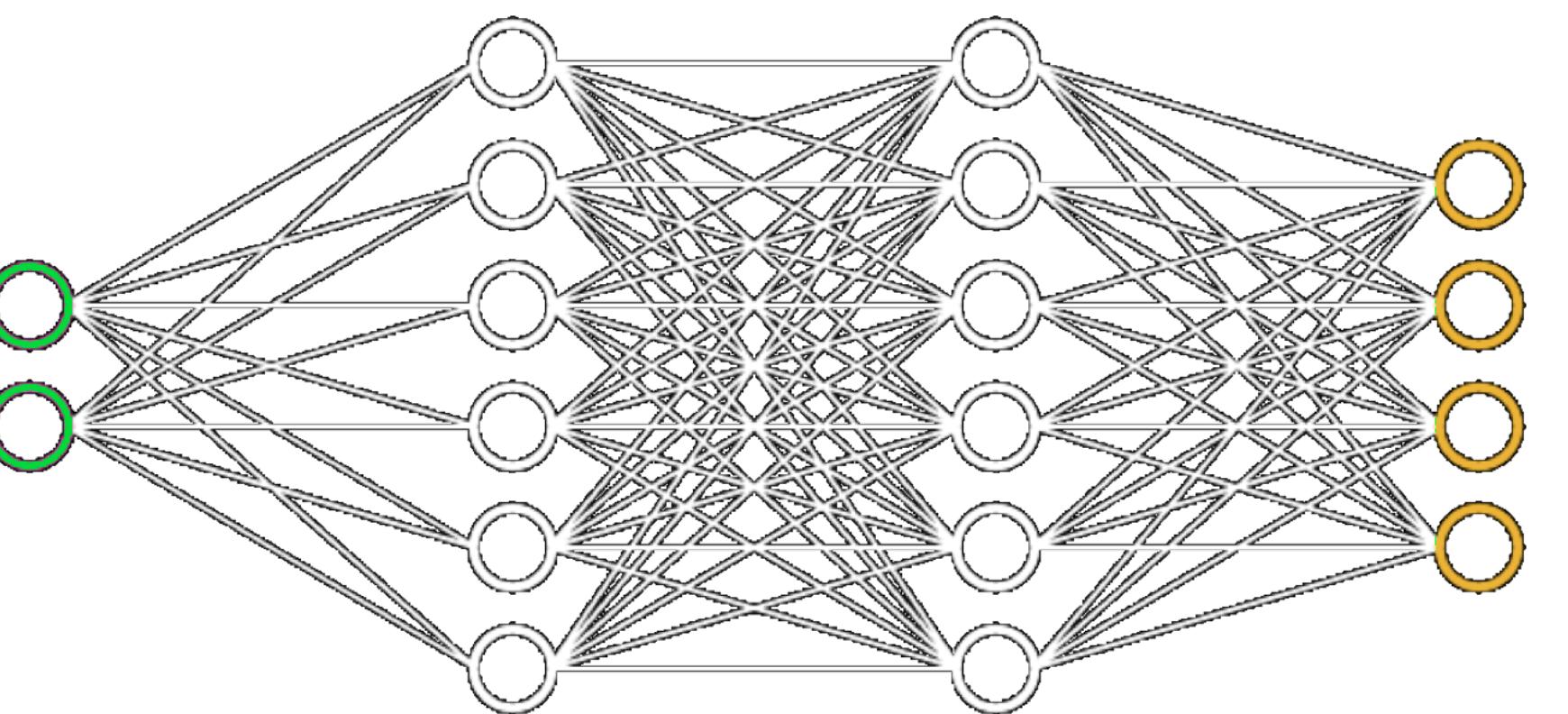
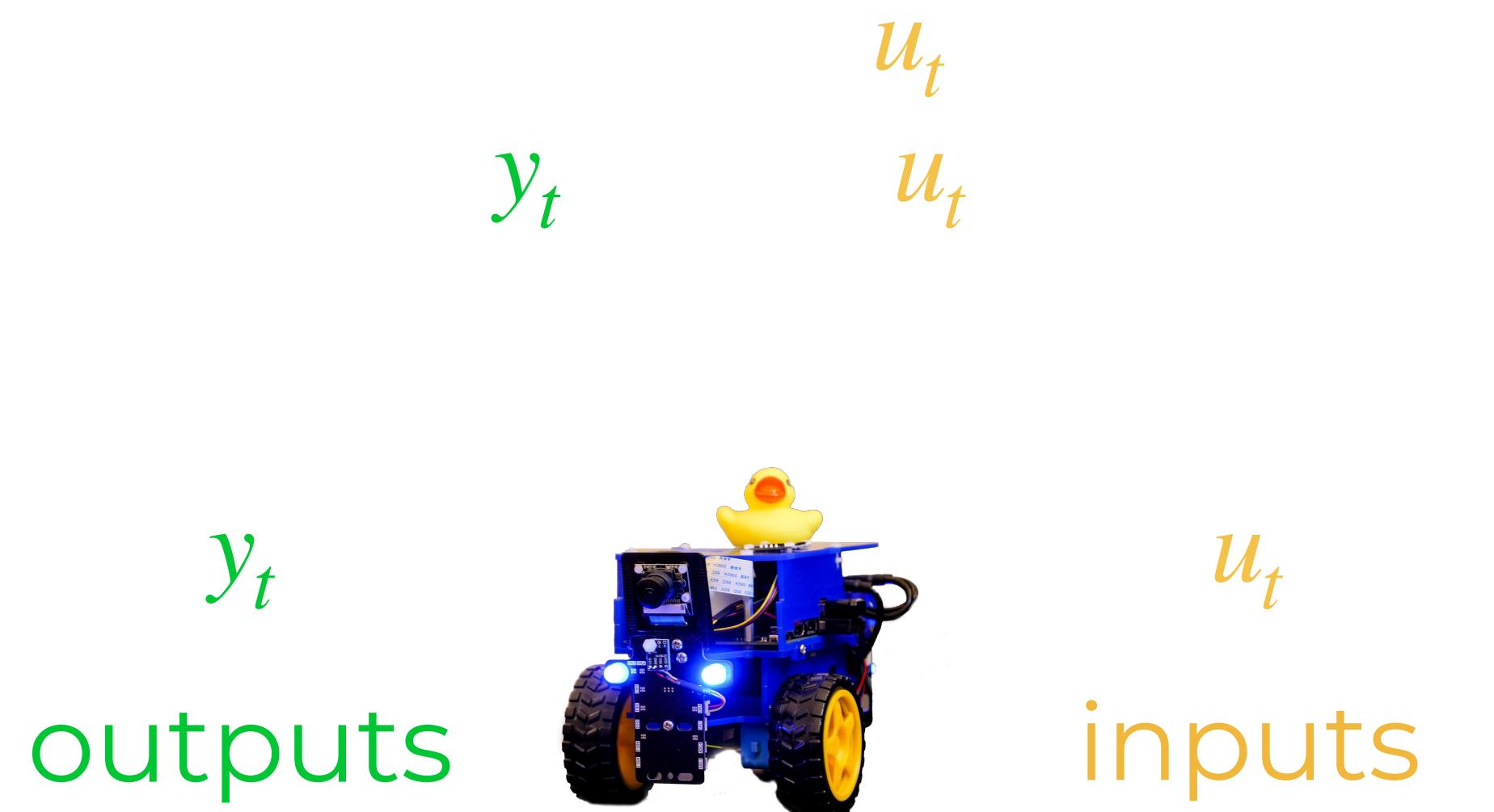
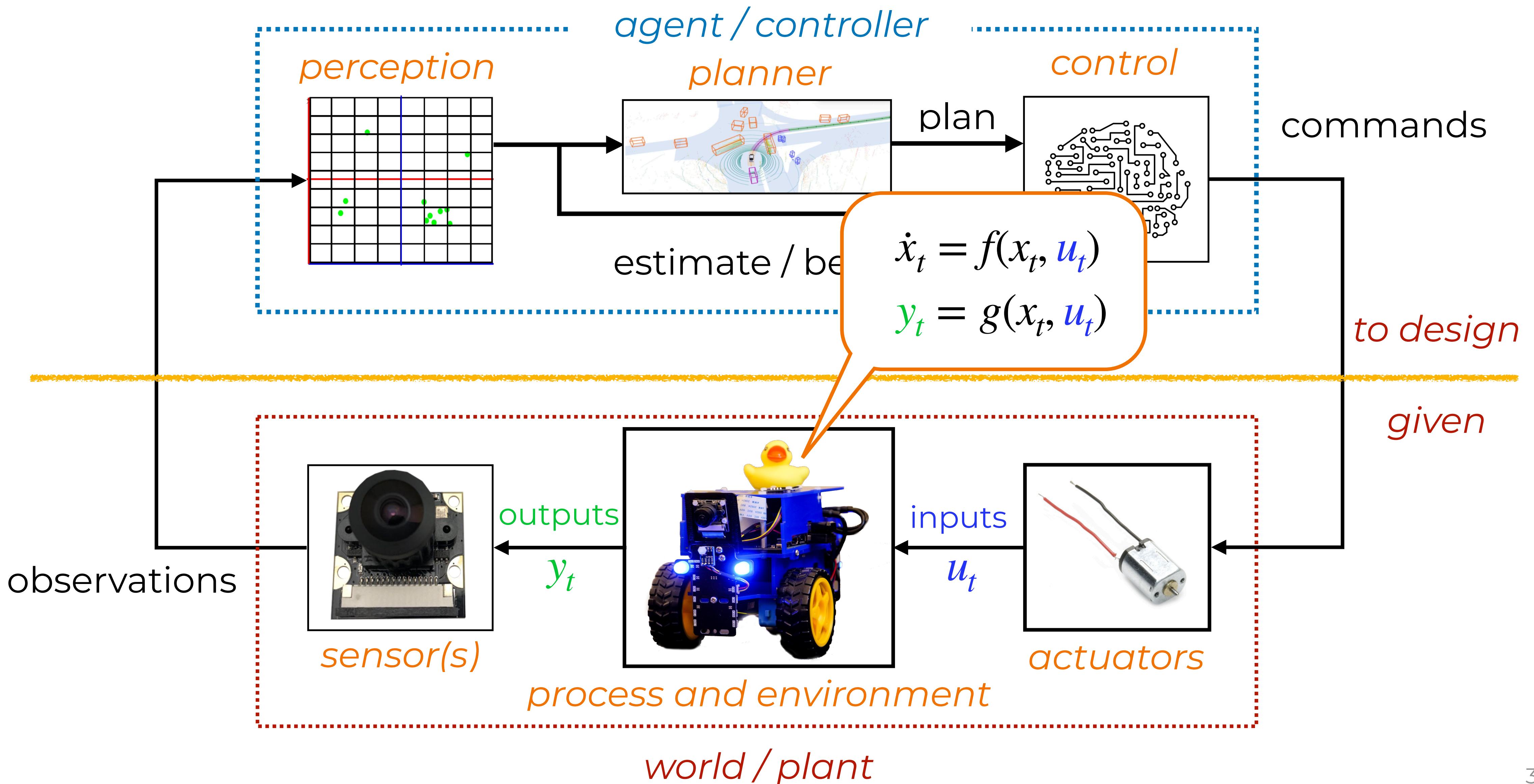




Modeling

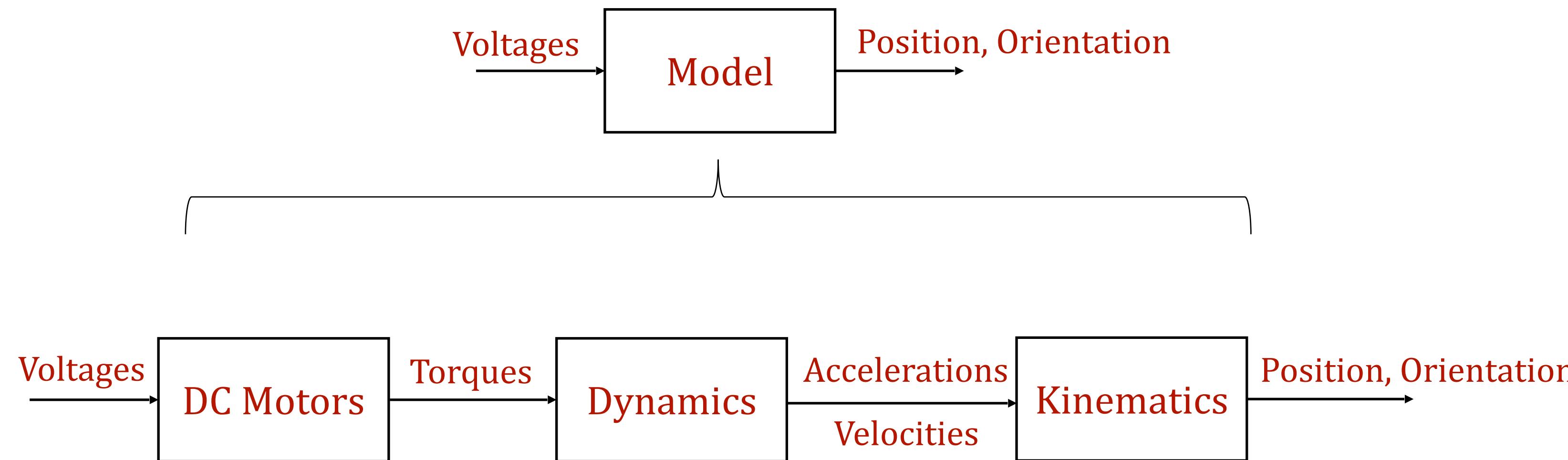


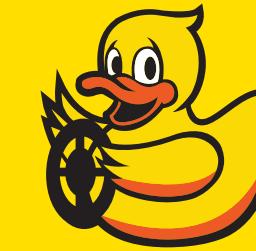
Models



Today's objective

Derive a usable model of a Duckiebot





Modeling

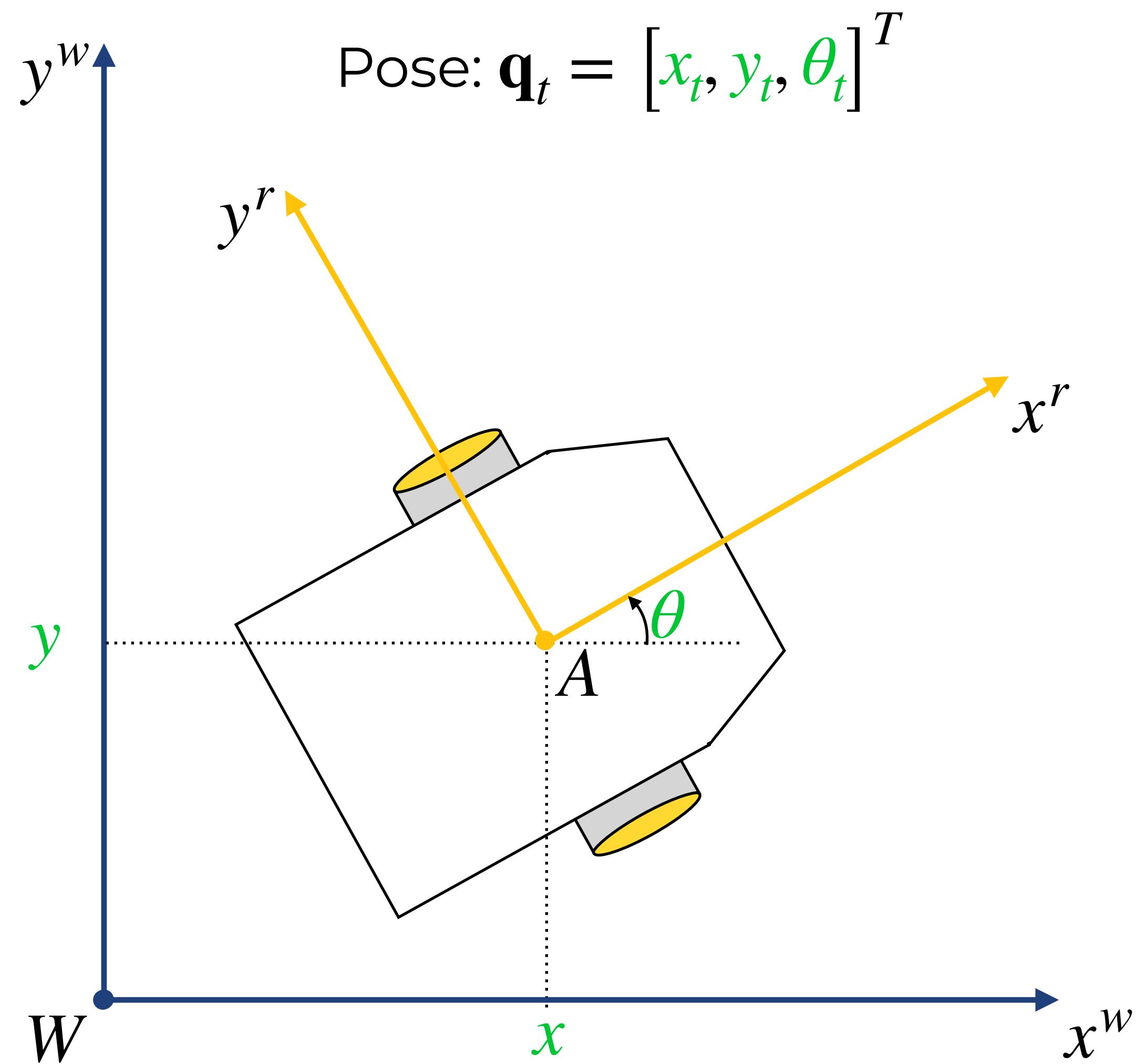
Kinematics	6
Dynamics	28
Motor Basics	35
Encoder Odometry	40
Summary	60

Kinematics

Notations

World Frame: $\{x^w, y^w\}$

Body (robot) frame: $\{x^r, y^r\}$

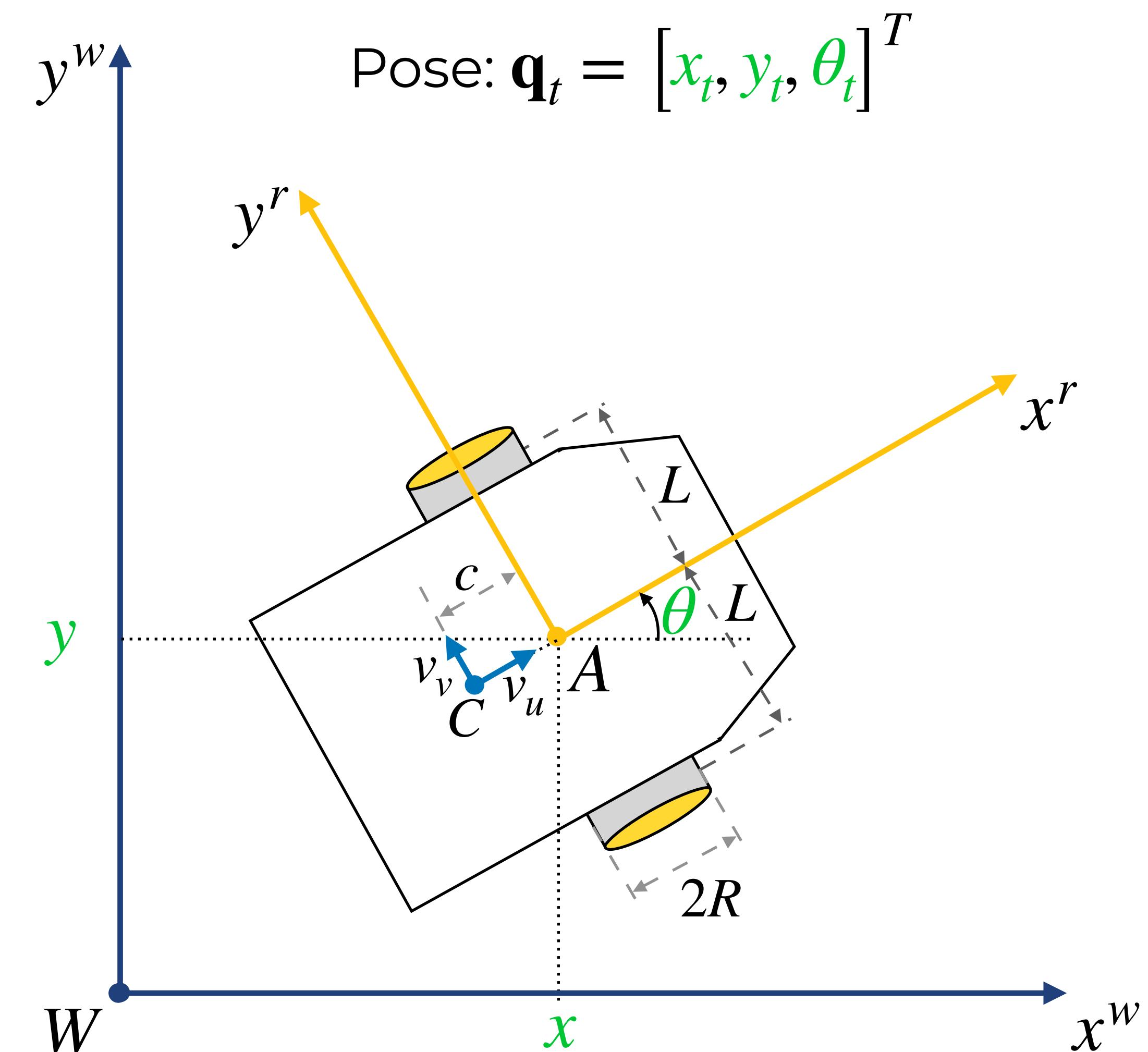


Notations

World Frame: $\{x^w, y^w\}$

Body (robot) frame: $\{x^r, y^r\}$

- **Assumption 1:** robot is **symmetric** along longitudinal axis (x^r)
 - Equidistant wheels (axle length = $2L$)
 - Identical wheels ($R_l = R_r = R$)
 - Center of mass on x^r at distance c from A

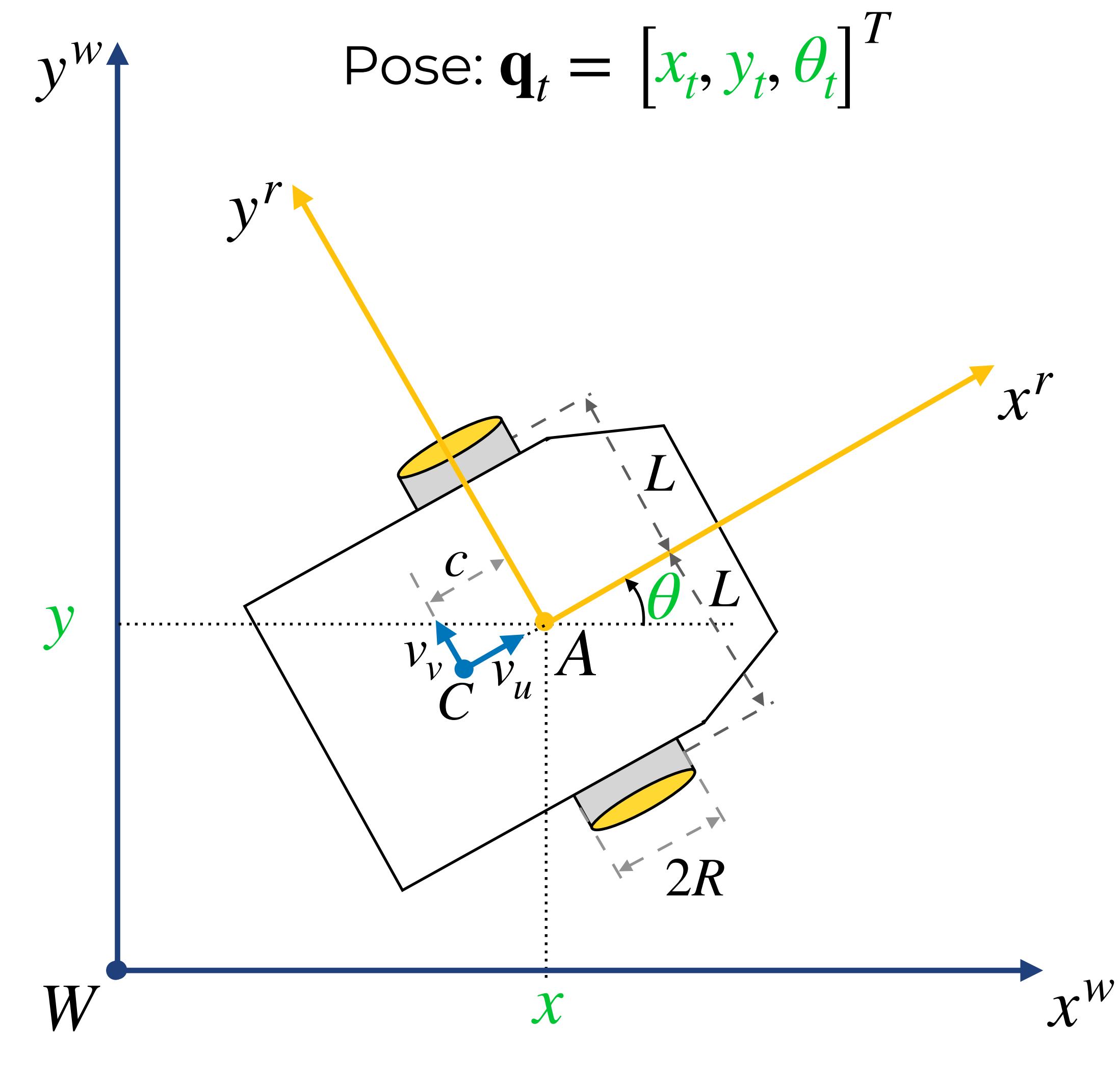


Notations

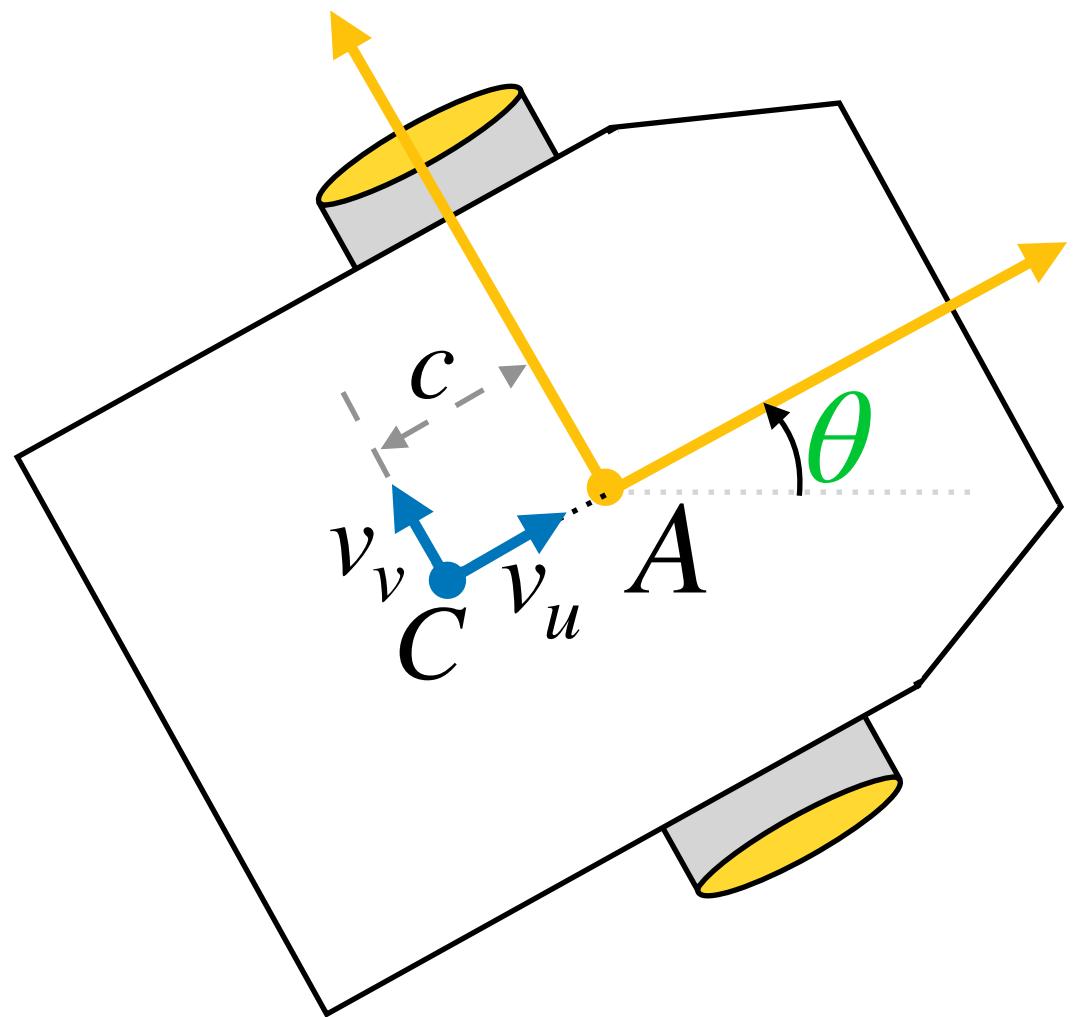
World Frame: $\{x^w, y^w\}$

Body (robot) frame: $\{x^r, y^r\}$

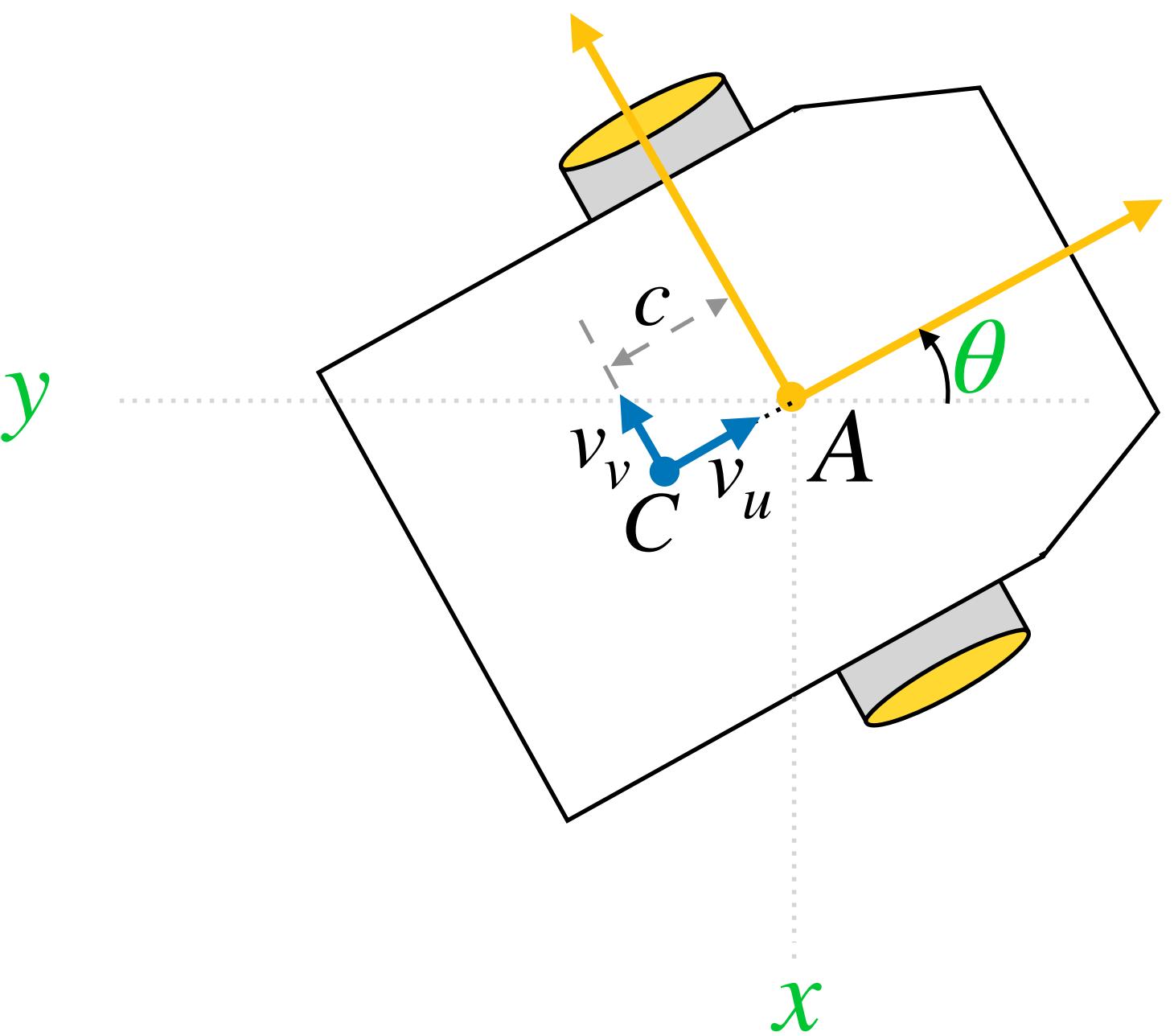
- **Assumption 1:** robot is **symmetric** along longitudinal axis (x^r)
 - Equidistant wheels (axle length = $2L$)
 - Identical wheels ($R_l = R_r = R$)
 - Center of mass on x^r at distance c from A
- **Assumption 2:** robot is a **rigid body**
 - distance between any two points of the robot does not change in time
 - in particular $\dot{c} = 0$, where $(\star) = \frac{d(\star)}{dt}$



$$\begin{aligned} v_u \\ \mathbf{v}_A^r \\ v_v - c\dot{\theta} \end{aligned}$$



$$\begin{aligned} v_u \\ \mathbf{v}_A^r \\ v_v - c\dot{\theta} \end{aligned}$$

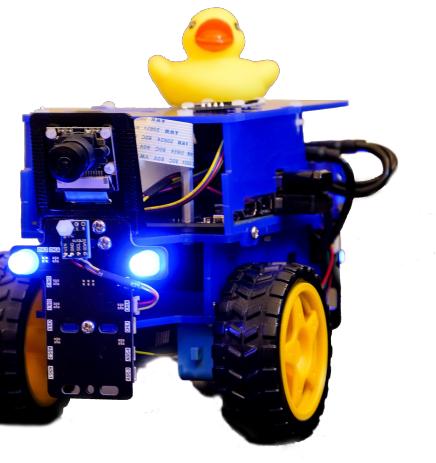


$$v_u$$

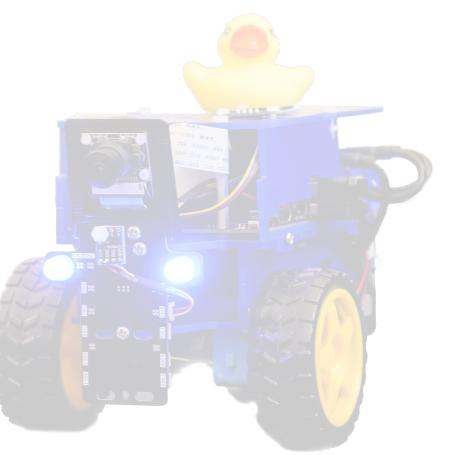
$$v_v - c\dot{\theta}$$

$$\mathbf{v}_A^r$$

13



Voltage



$V_{l,t}$



$V_{r,t}$



Pose



16

Voltage

$V_{l,t}$



\mathbf{q}_t

Kinematic constraint: pure rolling

- Assumption: Pure rolling
 - (1) No moving sideways (**skidding**): point A has null lateral component:

$$\mathbf{v}_A^r = \begin{bmatrix} v_u \\ v_v - c\dot{\theta} \end{bmatrix} = \begin{bmatrix} v_u \\ 0 \end{bmatrix}$$

- (2) No wheel **slipping**: every a full revolution ($\Delta\phi = 2\pi$), each wheel travels a distance equal to its circumference ($\Delta x = 2\pi R$)

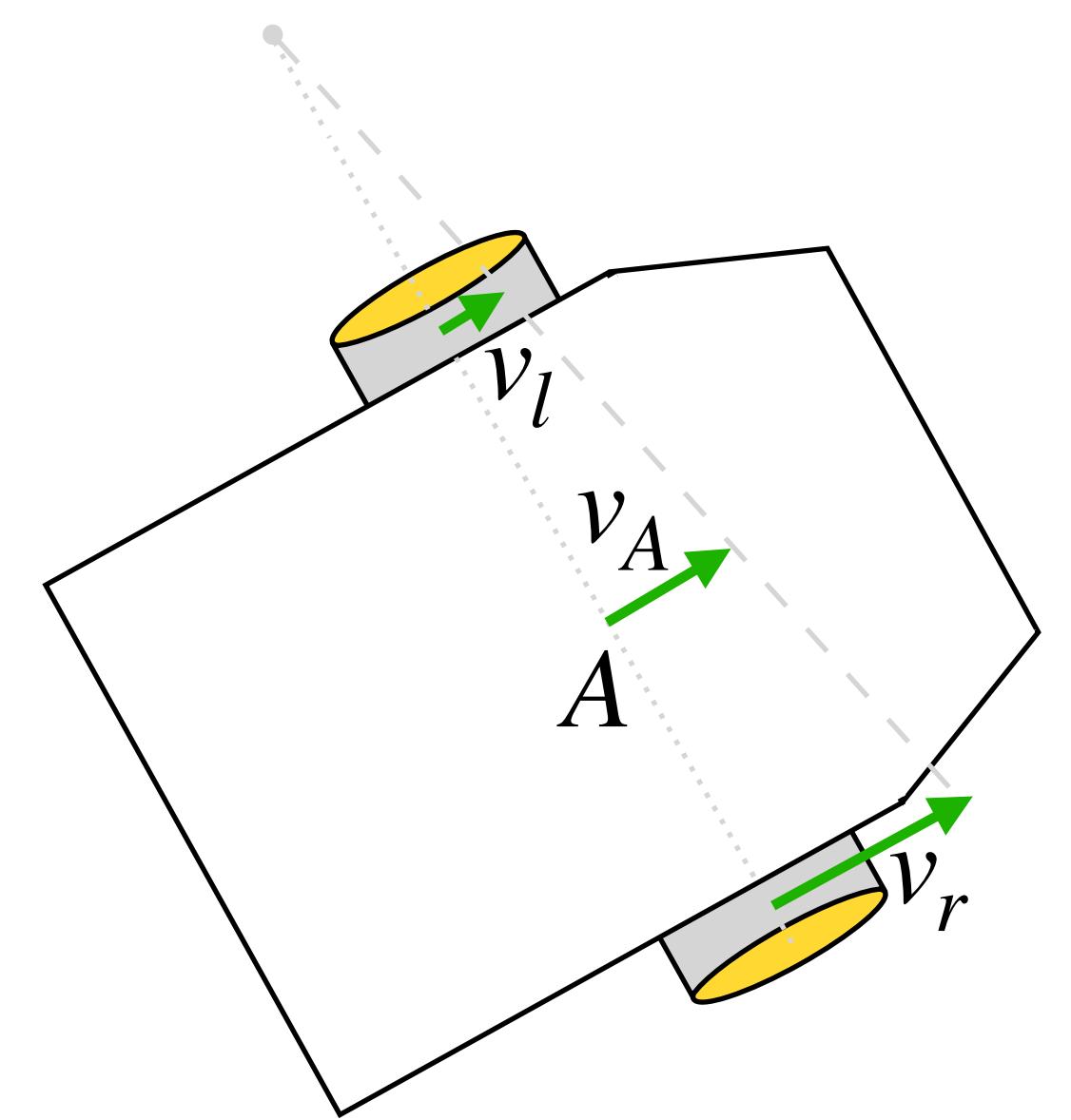


Velocities

$$\dot{\mathbf{q}}_t$$

$$\begin{matrix} \mathbf{v}_{A,t} \\ \boldsymbol{\omega}_t \end{matrix}$$

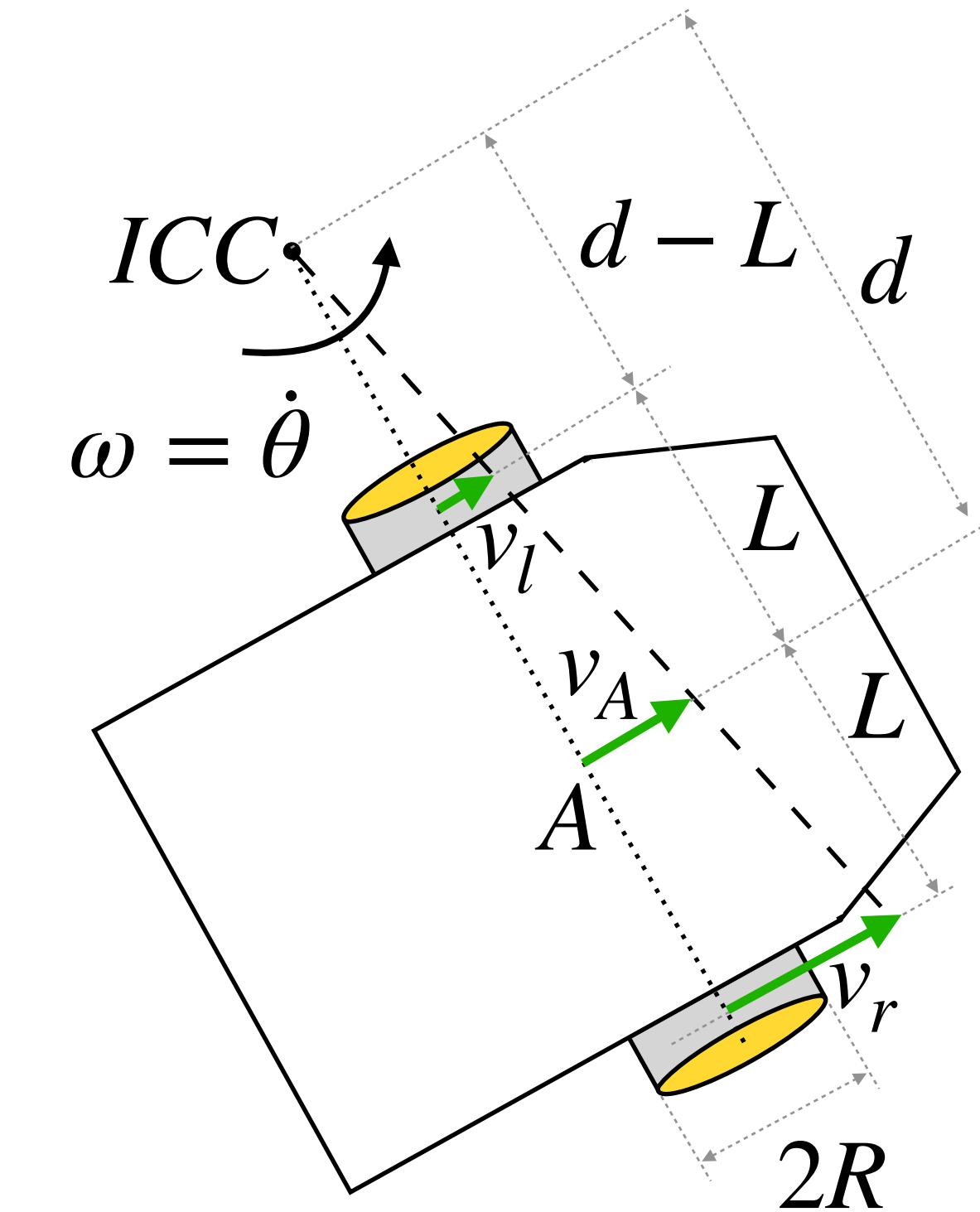
Pose
variation



Instantaneous center of curvature (ICC)

No slipping assumption \rightarrow all points in a pure rotation field (centered at *ICC*) have velocity orthogonal to vector that points to ICC

$$\begin{cases} v_l &= \omega(d - L) \\ v_A &= \omega d \\ v_r &= \omega(d + L) \end{cases} \Rightarrow d = L \frac{v_r + v_l}{v_r - v_l}$$

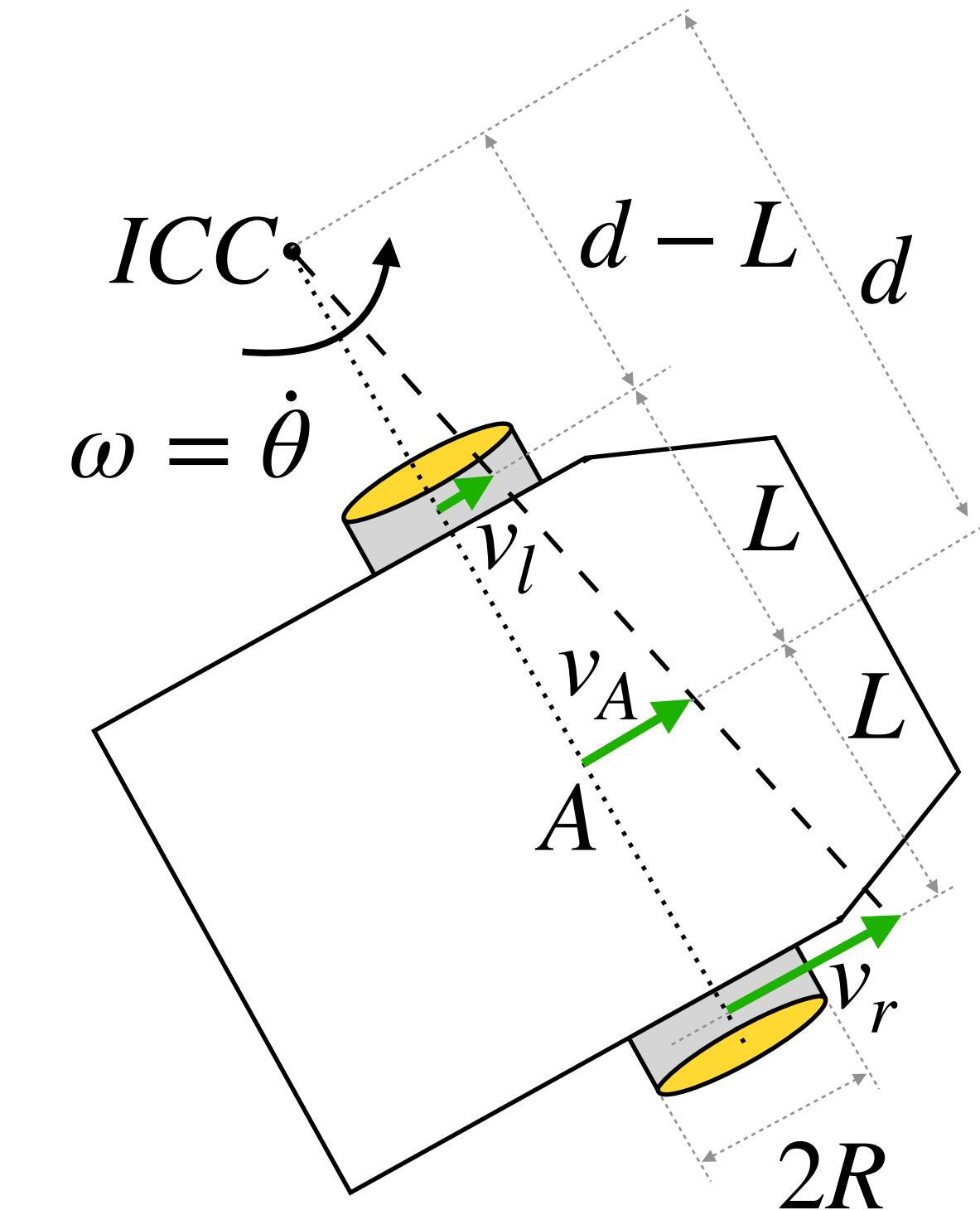


Instantaneous center of curvature (ICC)

No slipping assumption \rightarrow all points in a pure rotation field (centered at *ICC*) have velocity orthogonal to distance to ICC

$$\begin{cases} v_l &= \omega(d - L) \\ v_A &= \omega d \\ v_r &= \omega(d + L) \end{cases} \Rightarrow d = L \frac{v_r + v_l}{v_r - v_l}$$

1. If $v_r = v_l \Rightarrow$ no turn \Rightarrow ICC is undefined

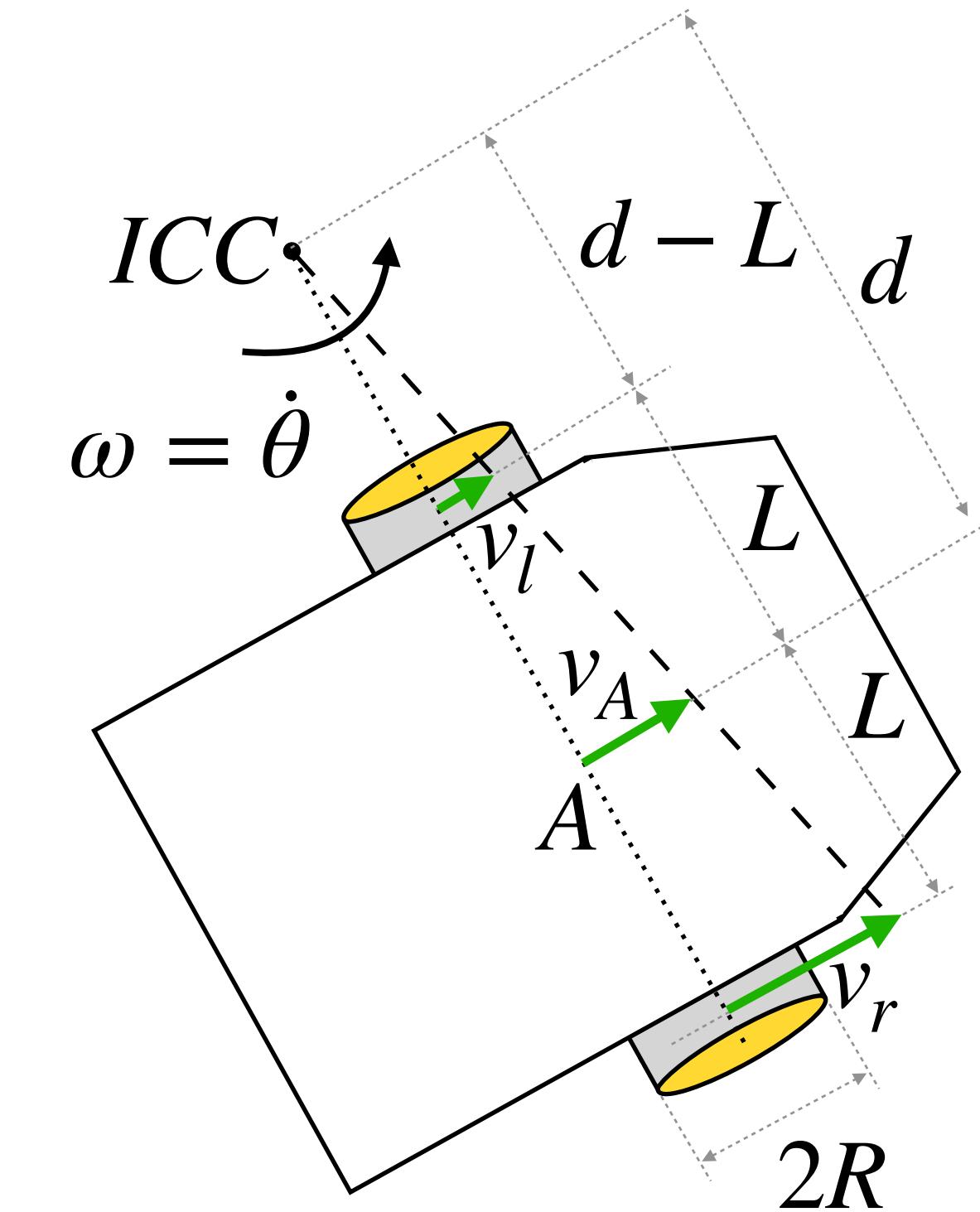


Instantaneous center of curvature (ICC)

No slipping assumption \rightarrow all points in a pure rotation field (centered at *ICC*) have velocity orthogonal to distance to ICC

$$\begin{cases} v_l &= \omega(d - L) \\ v_A &= \omega d \\ v_r &= \omega(d + L) \end{cases} \Rightarrow d = L \frac{v_r + v_l}{v_r - v_l}$$

1. If $v_r = v_l \Rightarrow$ no turn \Rightarrow *ICC* is undefined
2. If $v_r = -v_l \Rightarrow$ turn “on itself” \Rightarrow *ICC* $\equiv A$

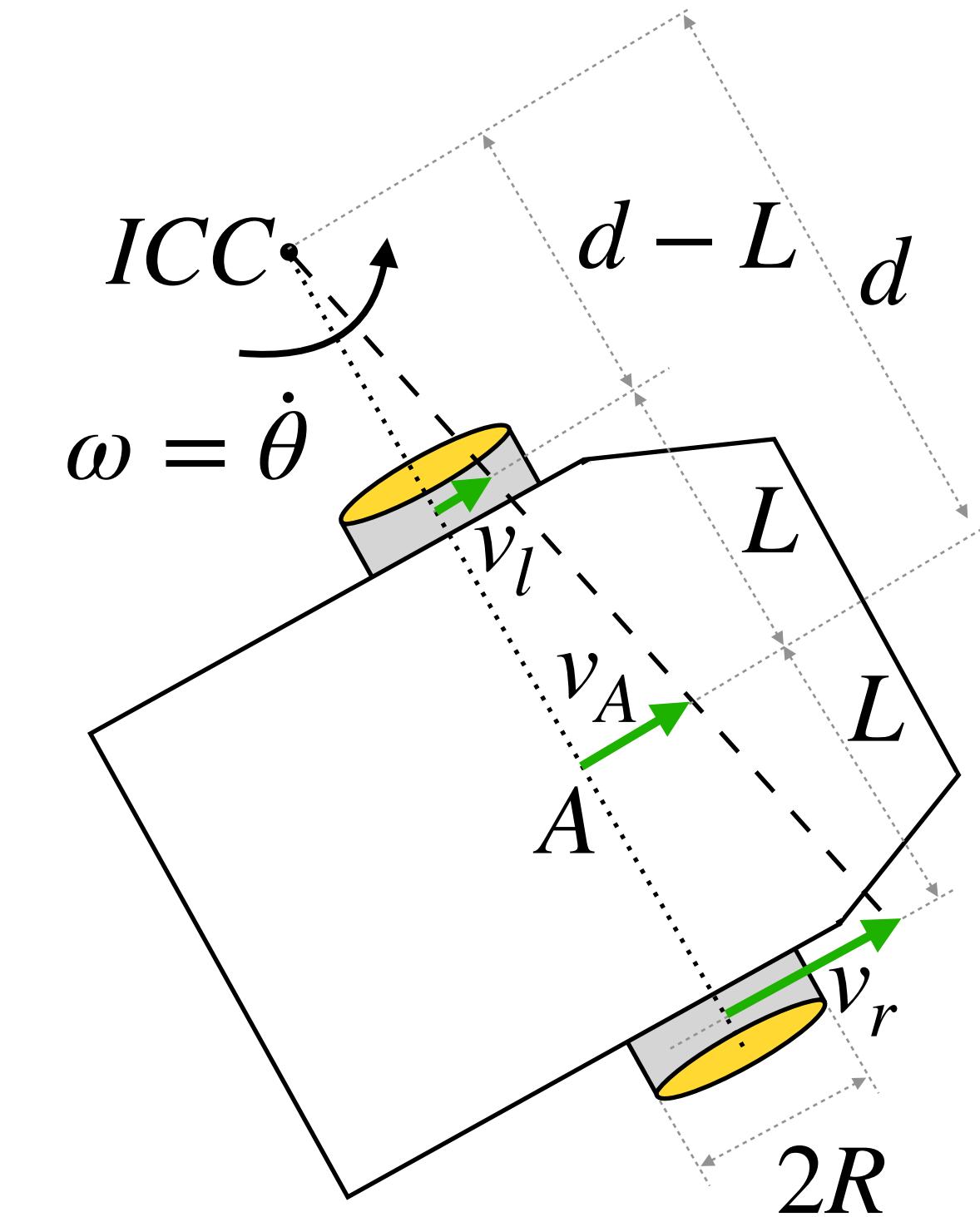


Instantaneous center of curvature (ICC)

No slipping assumption \rightarrow all points in a pure rotation field (centered at *ICC*) have velocity orthogonal to distance to ICC

$$\begin{cases} v_l = \omega(d - L) \\ v_A = \omega d \\ v_r = \omega(d + L) \end{cases} \Rightarrow d = L \frac{v_r + v_l}{v_r - v_l}$$

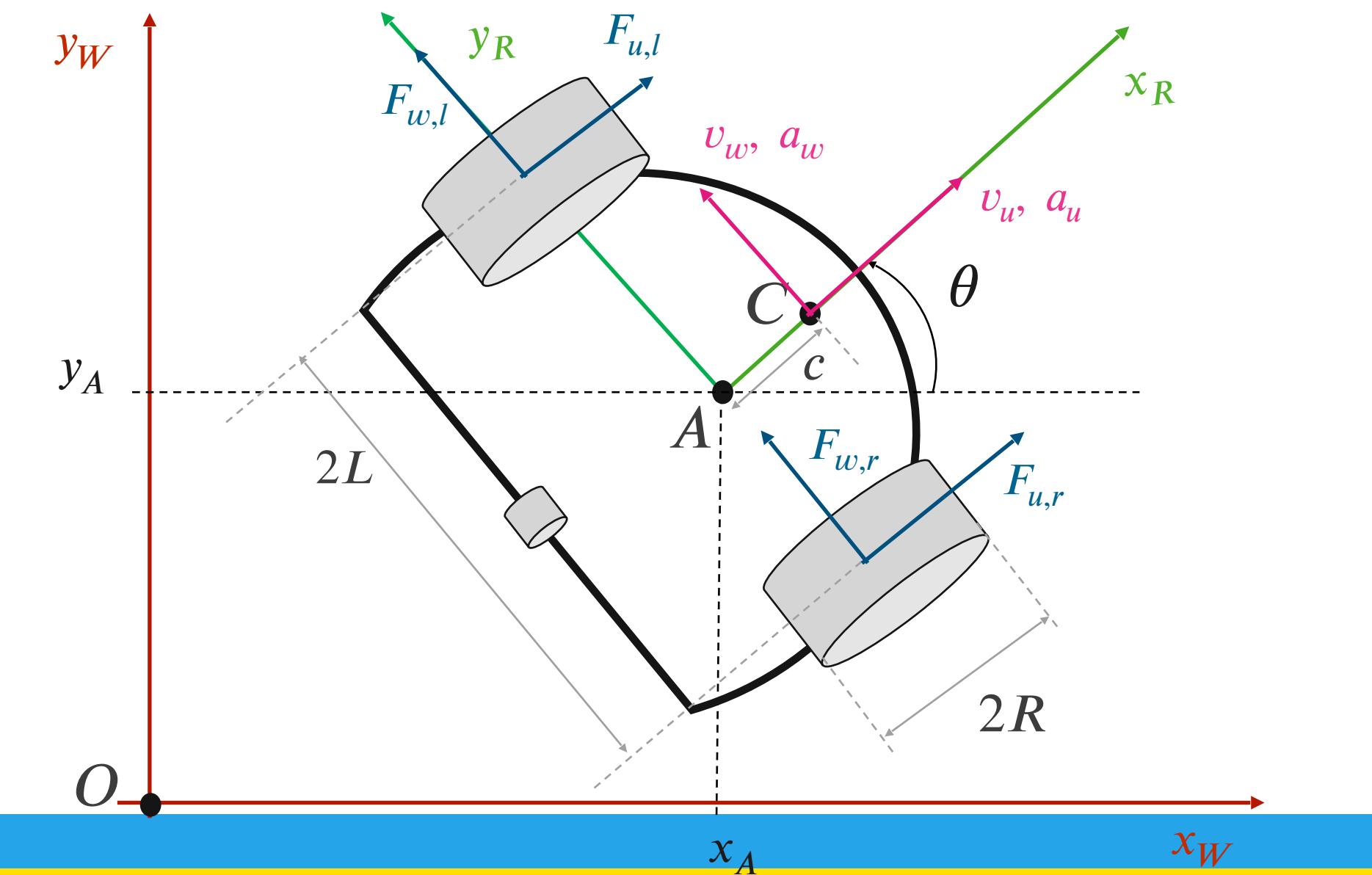
1. If $v_r = v_l \Rightarrow$ no turn \Rightarrow *ICC* is undefined
2. If $v_r = -v_l \Rightarrow$ turn “on itself” \Rightarrow *ICC* $\equiv A$
3. If $v_r = 0$ ($v_l = 0$) \Rightarrow turn “on wheel” $\Rightarrow d = -L$ ($d = L$)



Dynamics

Dynamics Notations

(v_u, v_w)	Longitudinal and lateral velocities of C , robot frame
(a_u, a_w)	Longitudinal and lateral accelerations of C , robot frame
(F_{uR}, F_{uL})	Longitudinal forces exerted on the vehicle by the right and left wheels
(F_{wR}, F_{wL})	Lateral forces exerted on the vehicle by the right and left wheels
(τ_R, τ_L)	Torques acting on right and left wheel
$\theta, \omega = \dot{\theta}$	Vehicle orientation and angular velocity
M	Vehicle mass
J	Vehicle yaw moment of inertia with respect to the center of mass C



Dynamic equilibria

The only modelled forces are those acting on the chassis from the wheels

- Forces: **radial** direction

$$Ma_u(t) = F_{u,L} + F_{u,R}$$

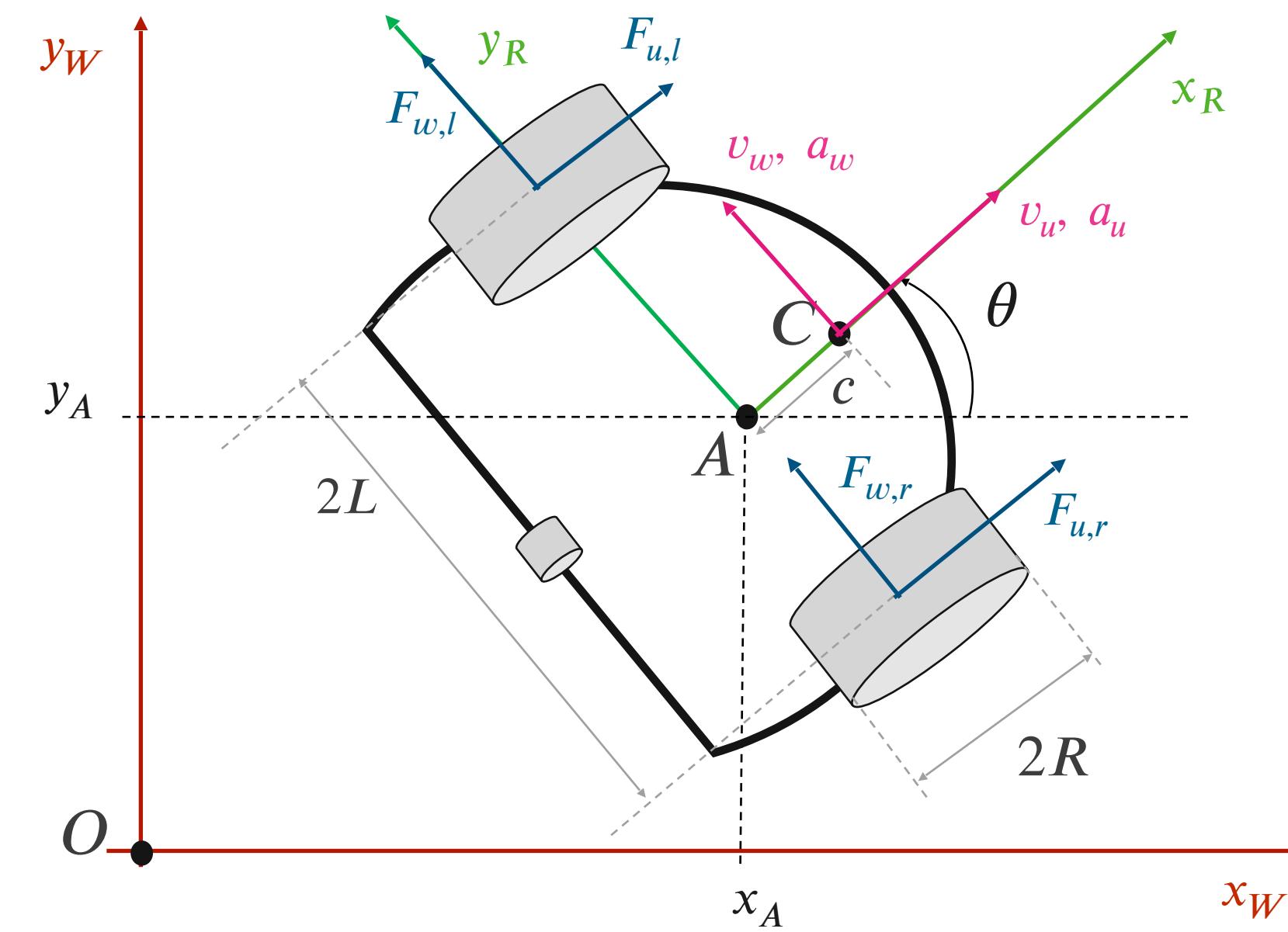
- **Dynamic hypothesis:**
 1. No caster wheel force

- Forces: **tangential** direction

$$Ma_w(t) = F_{w,L} + F_{w,R}$$

- Moments: yaw axis passing through

$$J\ddot{\theta}(t) = L(F_{u,R} - F_{u,L}) - c(F_{w,R} + F_{w,L})$$



General Dynamics Model

$$M\dot{a}_u(t) = F_{u,L} + F_{u,R}$$

$$M\dot{a}_w(t) = F_{w,L} - F_{w,R}$$

$$\ddot{\theta}(t) = \frac{L}{J}(F_{u,R} - F_{u,L}) + \frac{c}{J}(F_{w,R} - F_{w,L})$$

$$v_u(t) = \dot{r}(t)$$

$$v_w(t) = r(t)\dot{\theta}(t)$$

$$\dot{a}_u(t) = \ddot{r}(t) - r(t)\dot{\theta}^2(t)$$

$$\dot{a}_w(t) = 2\dot{r}(t)\dot{\theta}(t) - r(t)\ddot{\theta}(t)$$

$F_{u,L}(t), F_{u,R}(t)$

$$\begin{aligned}\dot{v}_u(t) &= \dot{v}_w(t)\theta(t) + \frac{F_{u,L} + F_{u,R}}{M} \\ \dot{v}_w(t) &= -\dot{v}_u(t)\theta(t) + \frac{F_{w,L} - F_{w,R}}{M} \\ \ddot{\theta}(t) &= \frac{L}{J}(F_{u,R} - F_{u,L}) + \frac{c}{J}(F_{w,R} - F_{w,L})\end{aligned}$$

$\begin{pmatrix} v_u(t) \\ v_w(t) \\ \omega(t) \end{pmatrix}$

General dynamic model in polar coordinates: coupled and nonlinear.
Does not account for kinematic constraints yet

Imposing Kinematic Constraints

Objective: Express velocity of in local frame and set to zero (no lateral slipping):

Steps:

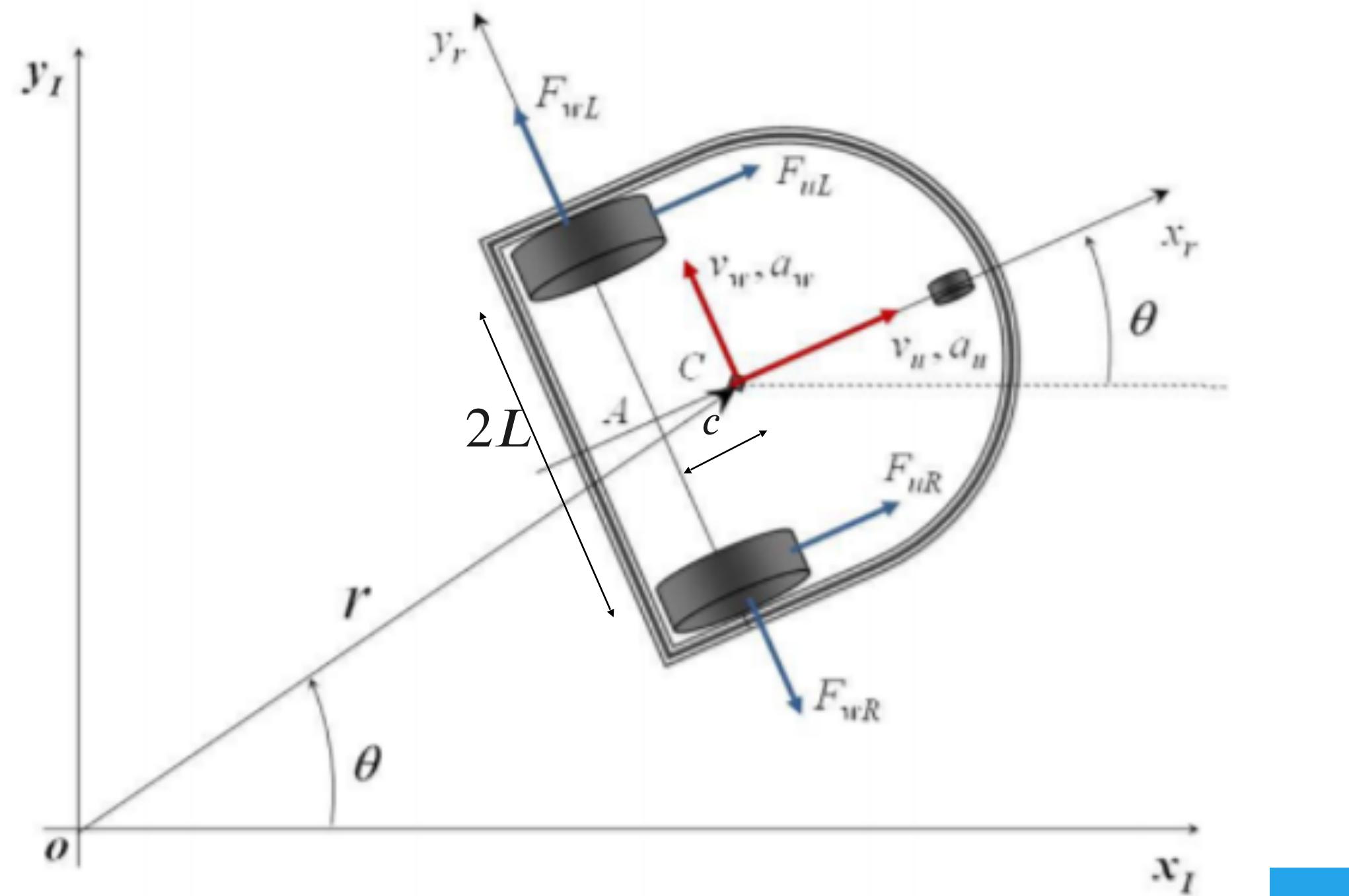
1. Express $q_{polar}(t)$ back in Cartesian components, obtain
2. Derive $v_A^I(t)$ as a function of $v_C^I(t)$ from:

$$\begin{cases} x_A^I(t) = x_C^I(t) - c \cos\theta \\ y_A^I(t) = y_C^I(t) - c \sin\theta \end{cases}$$

3. Obtain: $v_A^r(t) = R^T(\theta)v_A^I(t)$
4. Set: $\dot{y}_A^r(t) = 0$

Result:

$$v_w(t) = c \dot{\theta}$$



Simplified Dynamics Model

$$\dot{v}_u(t) = v_w(t)\dot{\theta}(t) + \frac{F_{u,L} + F_{u,R}}{M}$$

$$\dot{v}_w(t) = -v_u(t)\dot{\theta}(t) + \frac{F_{w,L} + F_{w,R}}{M}$$

$$\ddot{\theta}(t) = \frac{L}{J}(F_{u,R} - F_{u,L}) - \frac{c}{J}(F_{w,R} + F_{w,L})$$

$$\dot{v}_w(t) = c \ddot{\theta}(t)$$

$$F_{u,(\cdot)}R = \tau_{(\cdot)}$$

$\tau_R(t), \tau_L(t)$
Torques

$$\dot{v}_u(t) = c \dot{\theta}^2(t) + \frac{\tau_R(t) + \tau_L(t)}{RM}$$

$$v_w(t) = c \dot{\theta}(t)$$

$$\ddot{\theta}(t) = -\frac{Mc}{Mc^2 + J}\dot{\theta}(t)v_u(t) + \frac{L}{R(Mc^2 + J)}(\tau_R(t) - \tau_L(t))$$

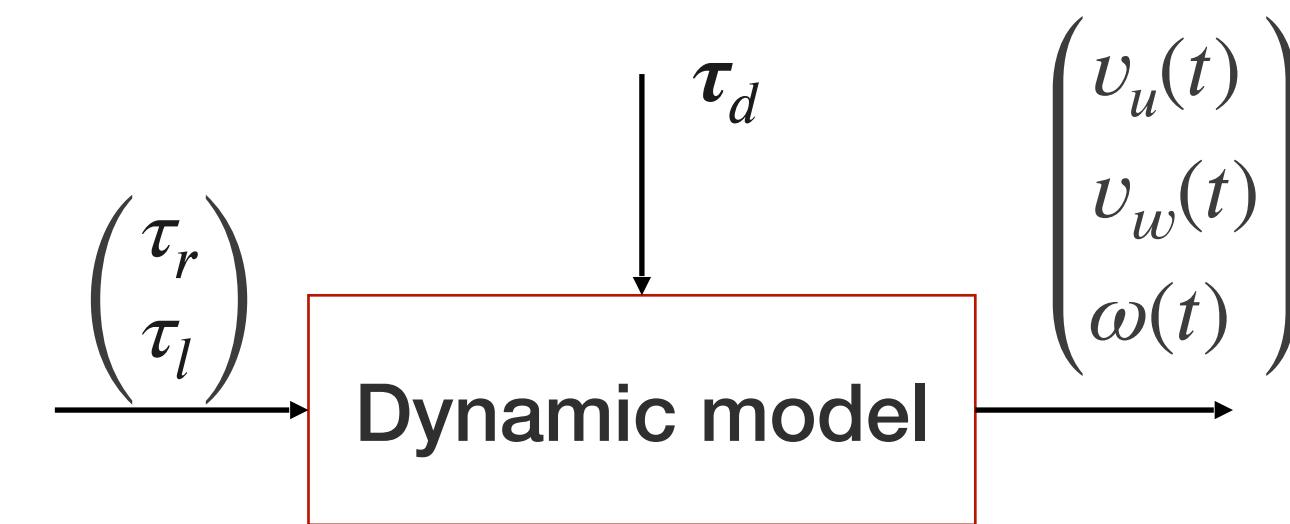
$$\begin{pmatrix} v_u(t) \\ v_w(t) \\ \omega(t) \end{pmatrix}$$

Accelerations
Velocities

Result

- The dynamic model:

$$\begin{bmatrix} M & 0 \\ 0 & Mc^2 + J \end{bmatrix} \begin{bmatrix} \dot{v}_u \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & -Mc\dot{\theta} \\ Mc\dot{\theta} & 0 \end{bmatrix} \begin{bmatrix} v_u \\ \dot{\theta} \end{bmatrix} = \frac{1}{R} \begin{bmatrix} 1 & 1 \\ L & -L \end{bmatrix} \begin{bmatrix} \tau_r \\ \tau_l \end{bmatrix} + \boldsymbol{\tau}_d$$
$$v_w(t) = c \dot{\theta}$$



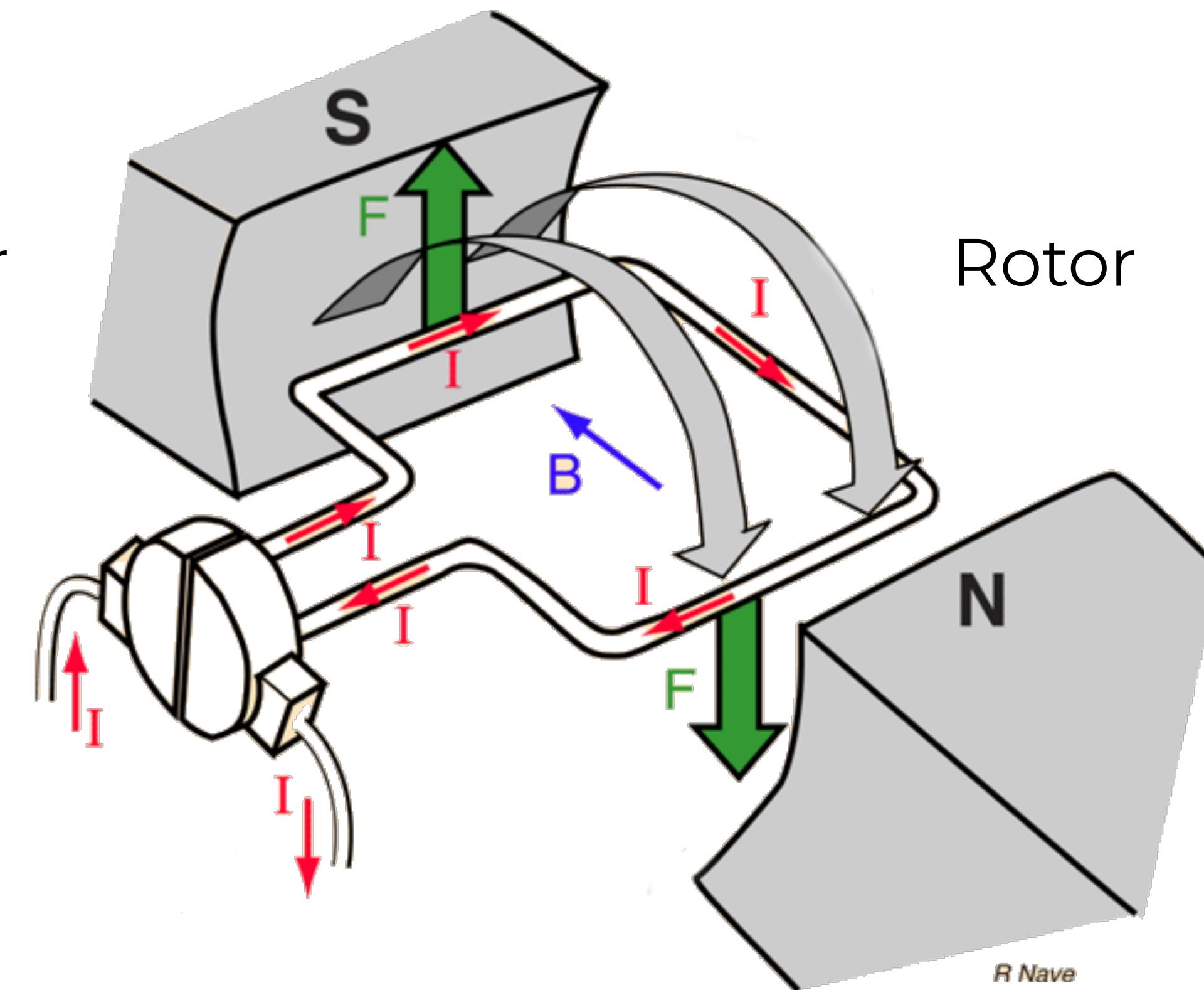
Motor Basics

DC motor intuition

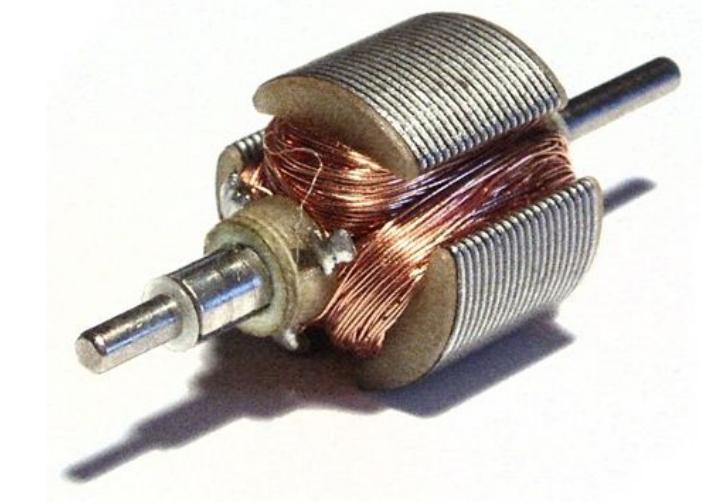
The dynamics of a DC motor is governed by the coupling of electrical and mechanical subsystems:

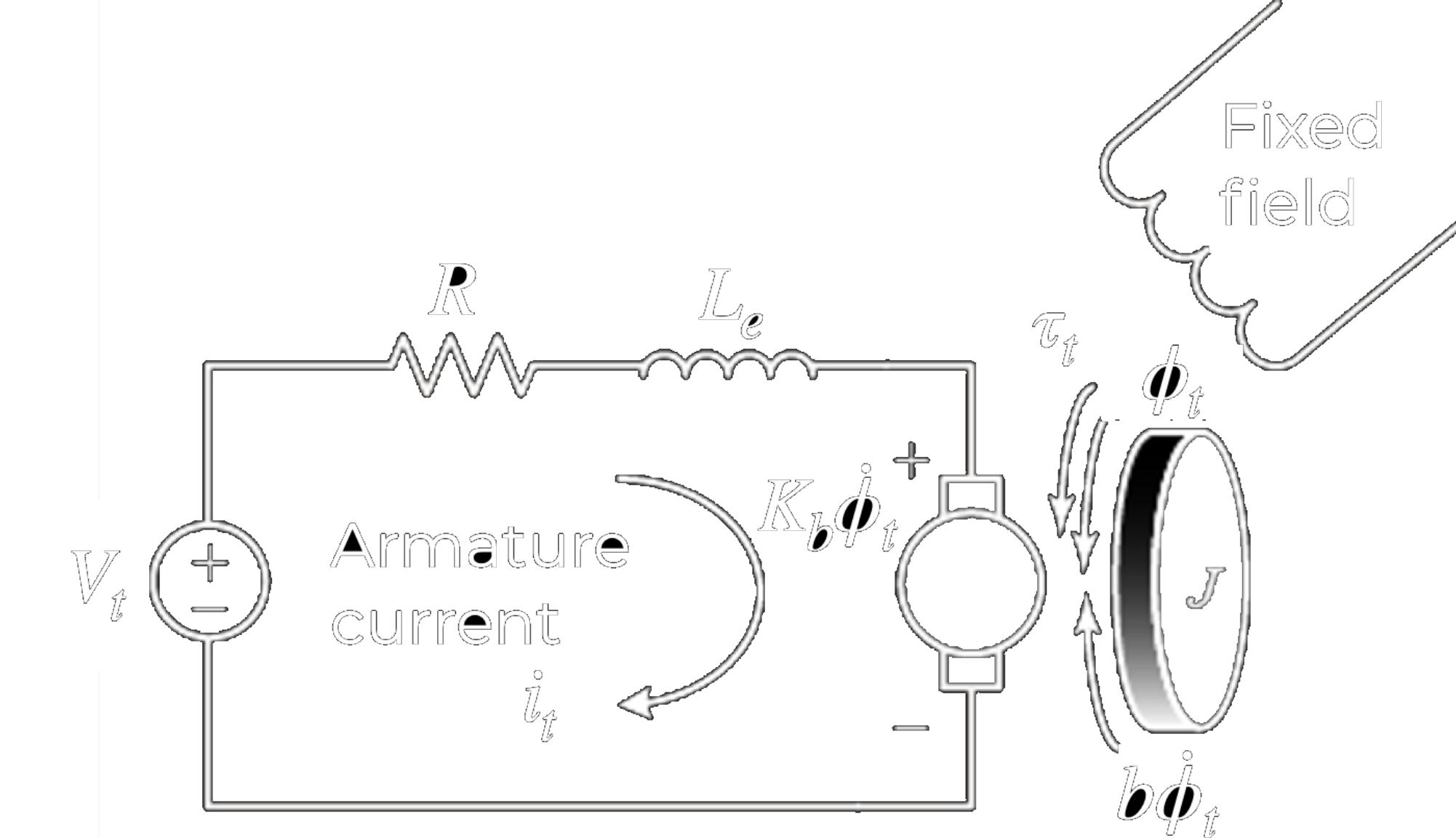


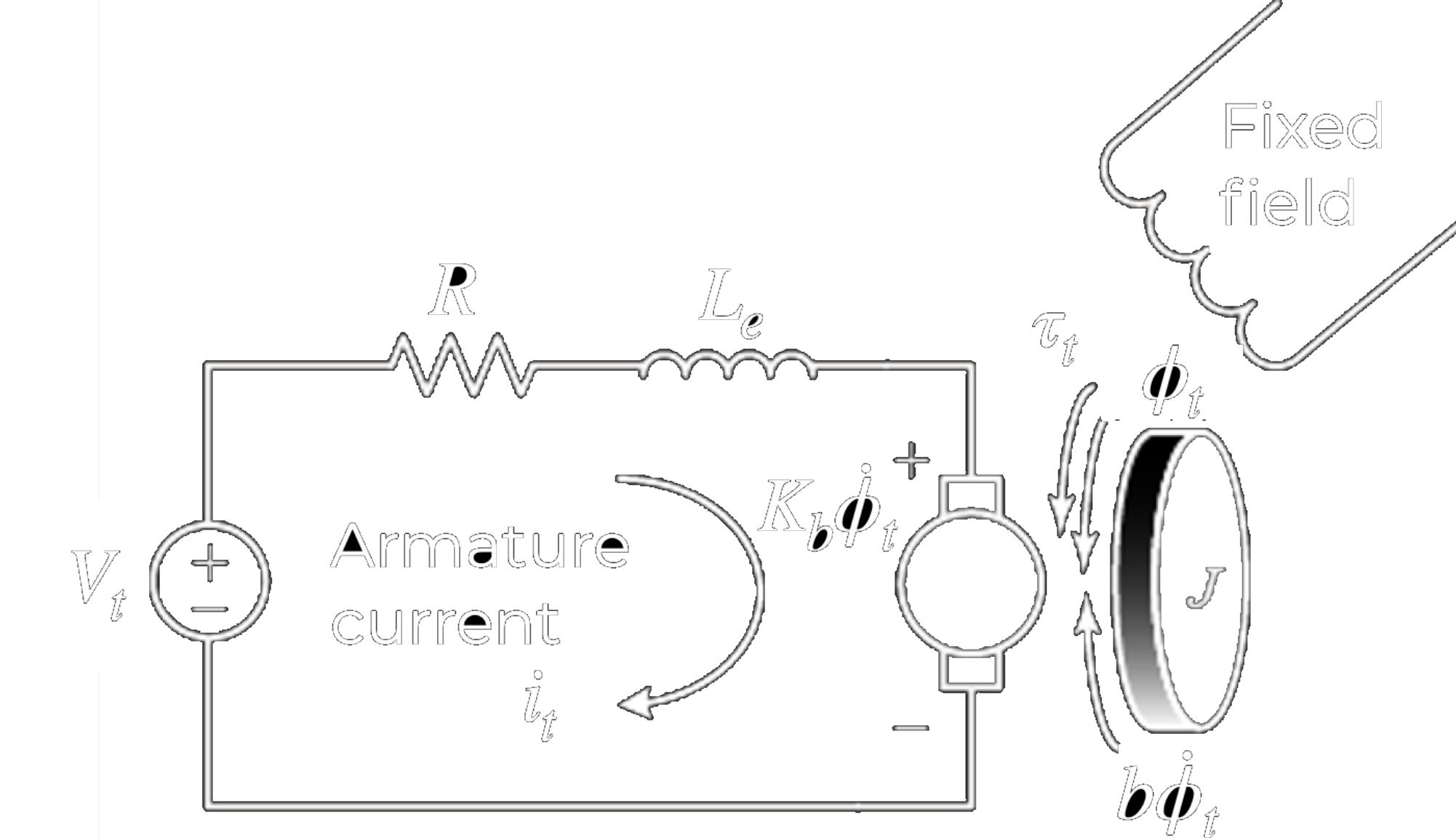
Stator



Rotor







Encoder Odometry

Odometry

όδός (route) +

μέτρον (measurement)

Given:

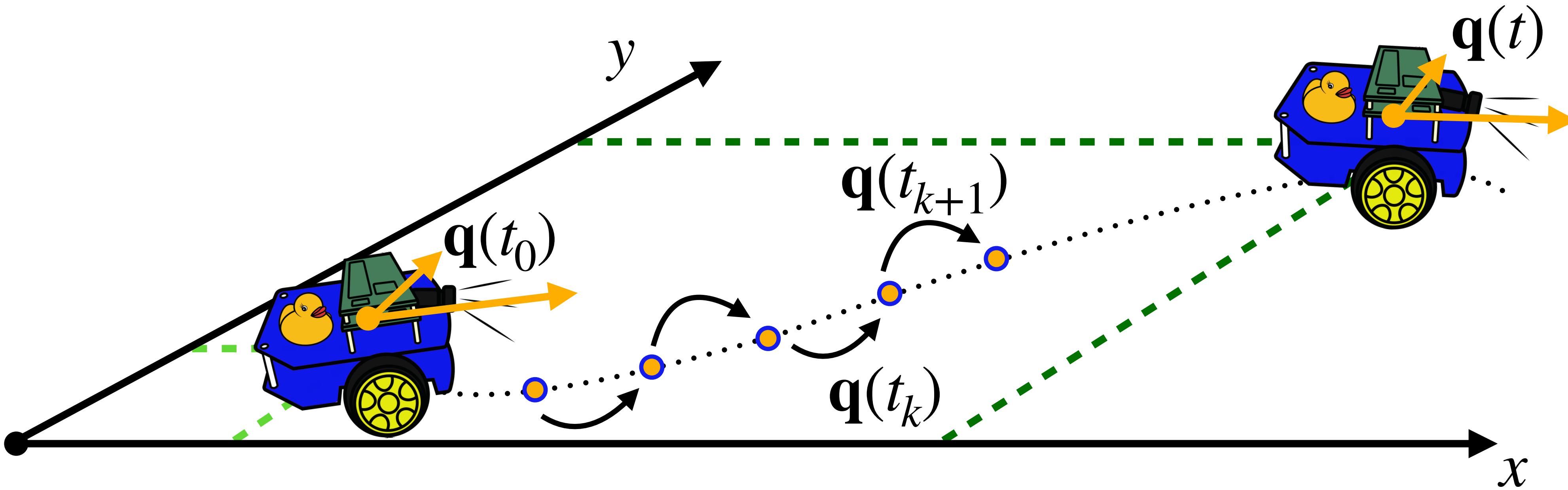
$$\mathbf{q}(t_0) = [x(t_0) \quad y(t_0) \quad \theta(t_0)]^T$$

Find:

$$\mathbf{q}(t_0 + \Delta t), \forall \Delta t > 0$$

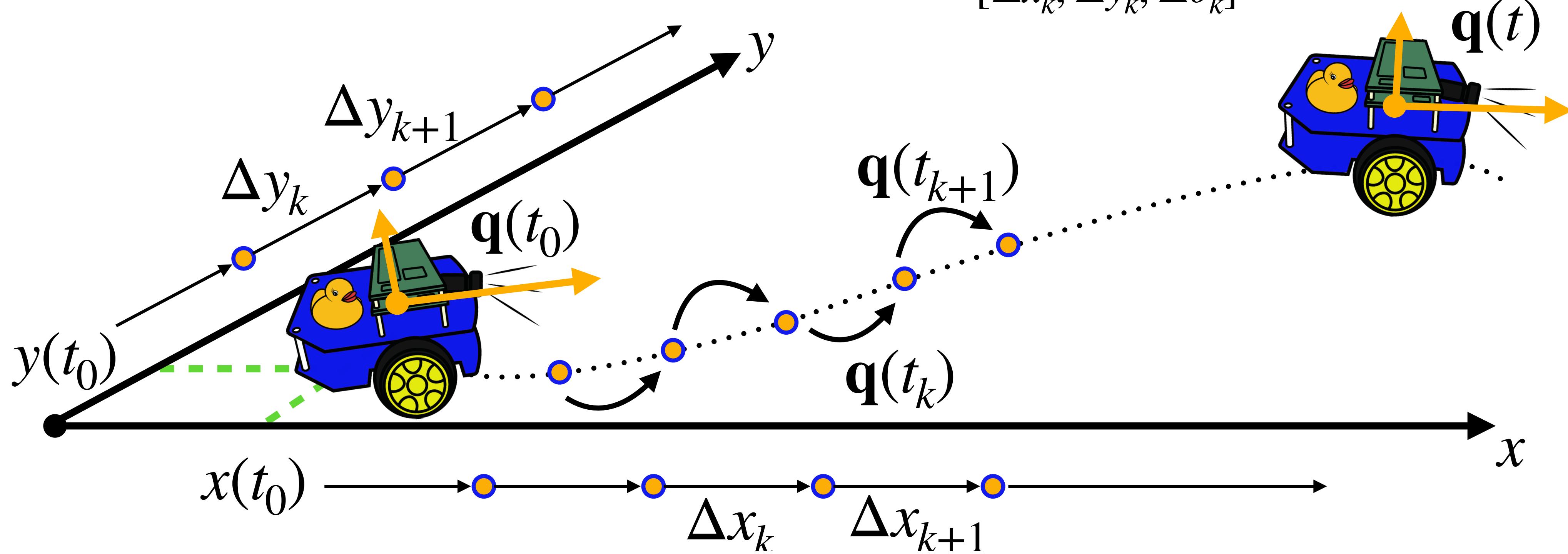
$$\mathbf{q}(t) = [x(t) \quad y(t) \quad \theta(t)]^T$$

$$\mathbf{q}(t_{k+1}) \simeq \mathbf{q}(t_k) + \dot{\mathbf{q}}(t_k)(t_{k+1} - t_k)$$



$$\mathbf{q}(t) = [x(t) \quad y(t) \quad \theta(t)]^T$$

$$\mathbf{q}(t_{k+1}) \simeq \mathbf{q}(t_k) + \underbrace{\dot{\mathbf{q}}(t_k)(t_{k+1} - t_k)}_{[\Delta x_k, \Delta y_k, \Delta \theta_k]^T}$$



$$\mathbf{q}(t_{k+1}) \simeq \mathbf{q}(t_k) + \dot{\mathbf{q}}(t_k)(t_{k+1} - t_k)$$

Kinematic model

$$\dot{\mathbf{q}}(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = \frac{R}{2} \begin{bmatrix} \cos\theta(t) & 0 \\ \sin\theta(t) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ \frac{1}{L} & -\frac{1}{L} \end{bmatrix} \begin{bmatrix} \dot{\phi}_r(t) \\ \dot{\phi}_l(t) \end{bmatrix}$$

$$\mathbf{q}(t_{k+1}) \simeq \mathbf{q}(t_k) + \dot{\mathbf{q}}(t_k)(t_{k+1} - t_k)$$

Kinematic model

$$\dot{\mathbf{q}}(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = \frac{R}{2} \begin{bmatrix} \cos\theta(t) & 0 \\ \sin\theta(t) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ \frac{1}{L} & -\frac{1}{L} \end{bmatrix} \begin{bmatrix} \dot{\phi}_r(t) \\ \dot{\phi}_l(t) \end{bmatrix}$$

Parameters

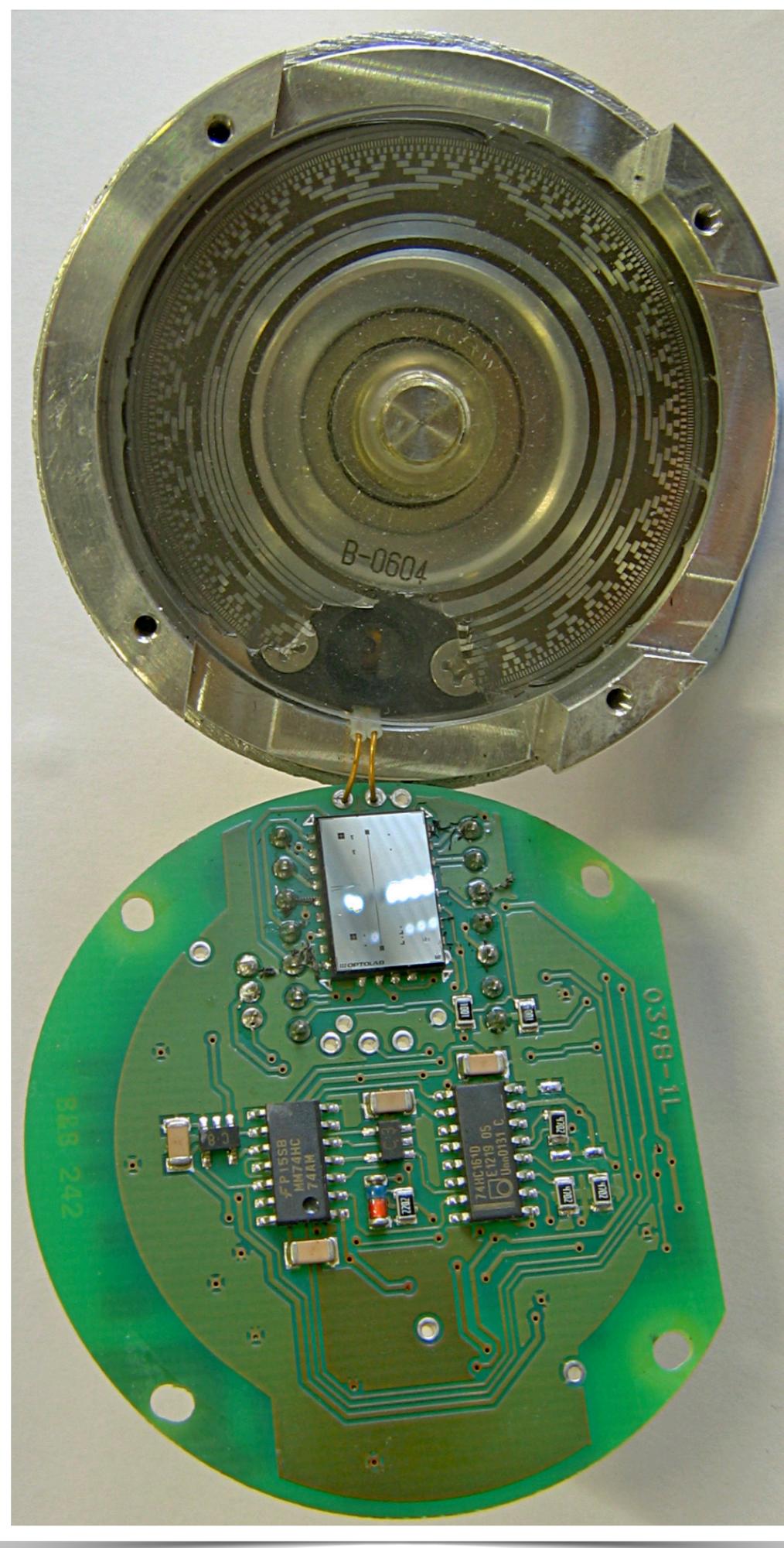
- R : wheel radius
- $2L$: baseline

Measurements

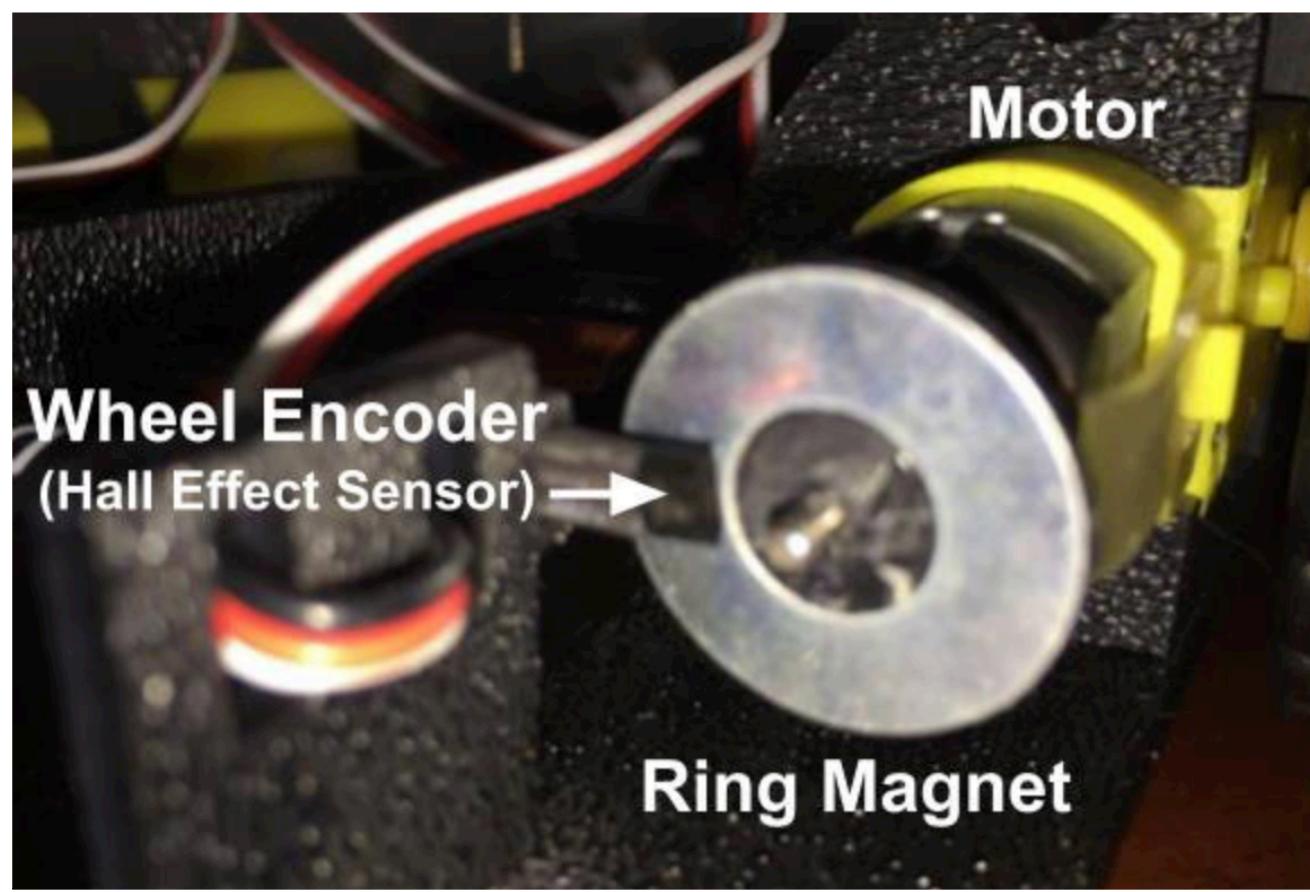
- $\dot{\phi}_{r/l}(t)$: wheel angular speeds

Wheel encoders

Absolute rotary encoder

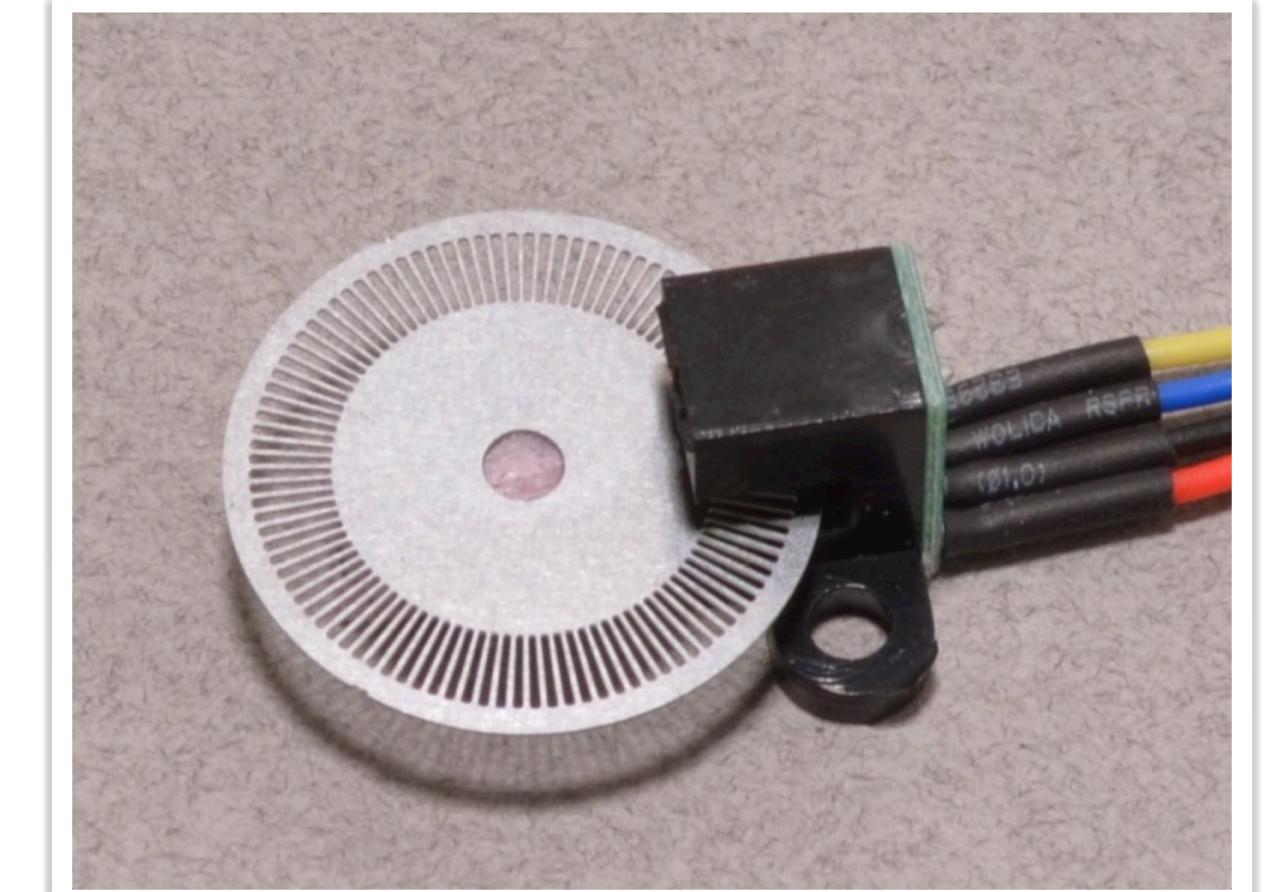


Hall-effect incremental encoder



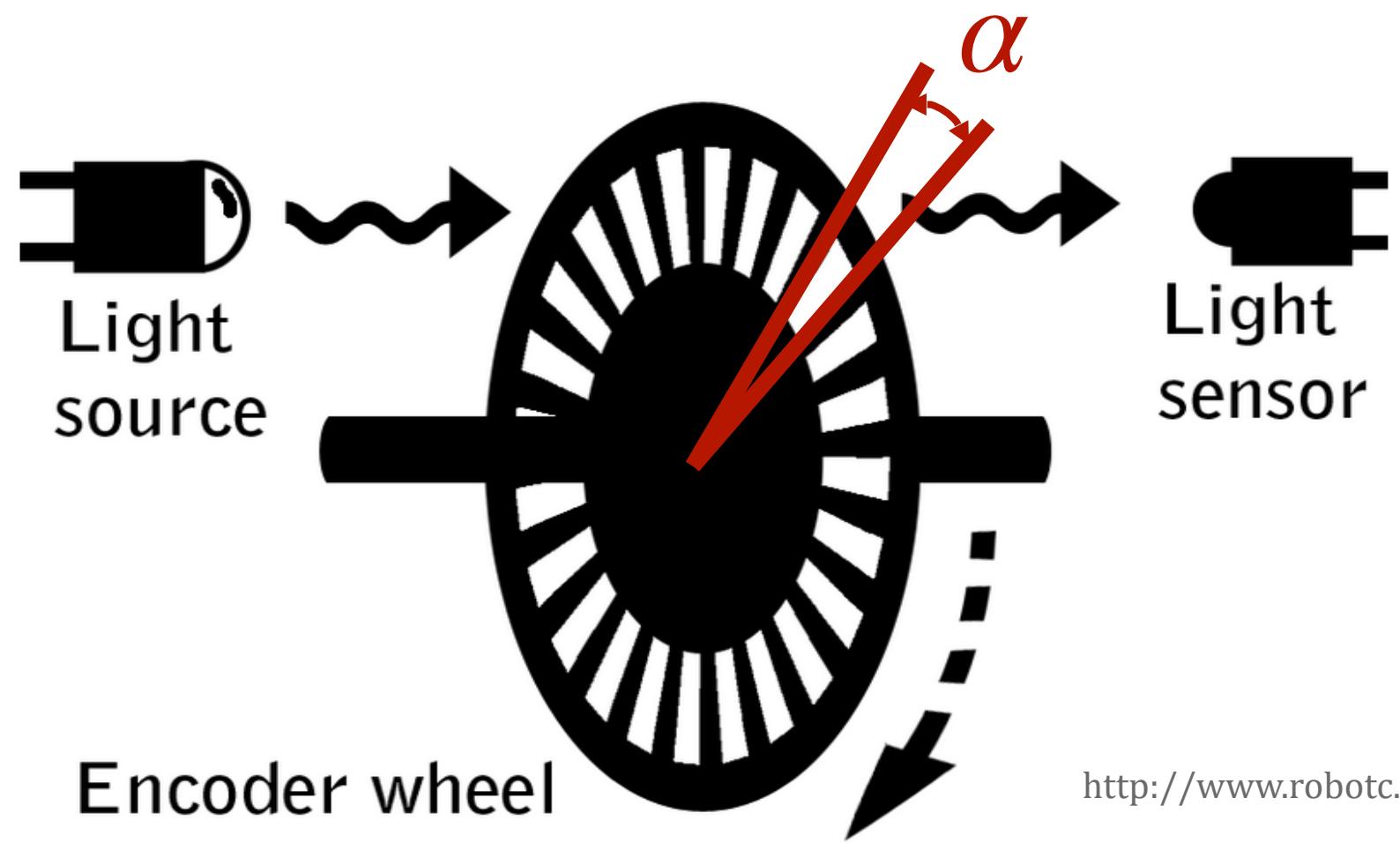
<https://docs.idew.org/code-robotics/references/physical-inputs/wheel-encoders>

Optical incremental encoder



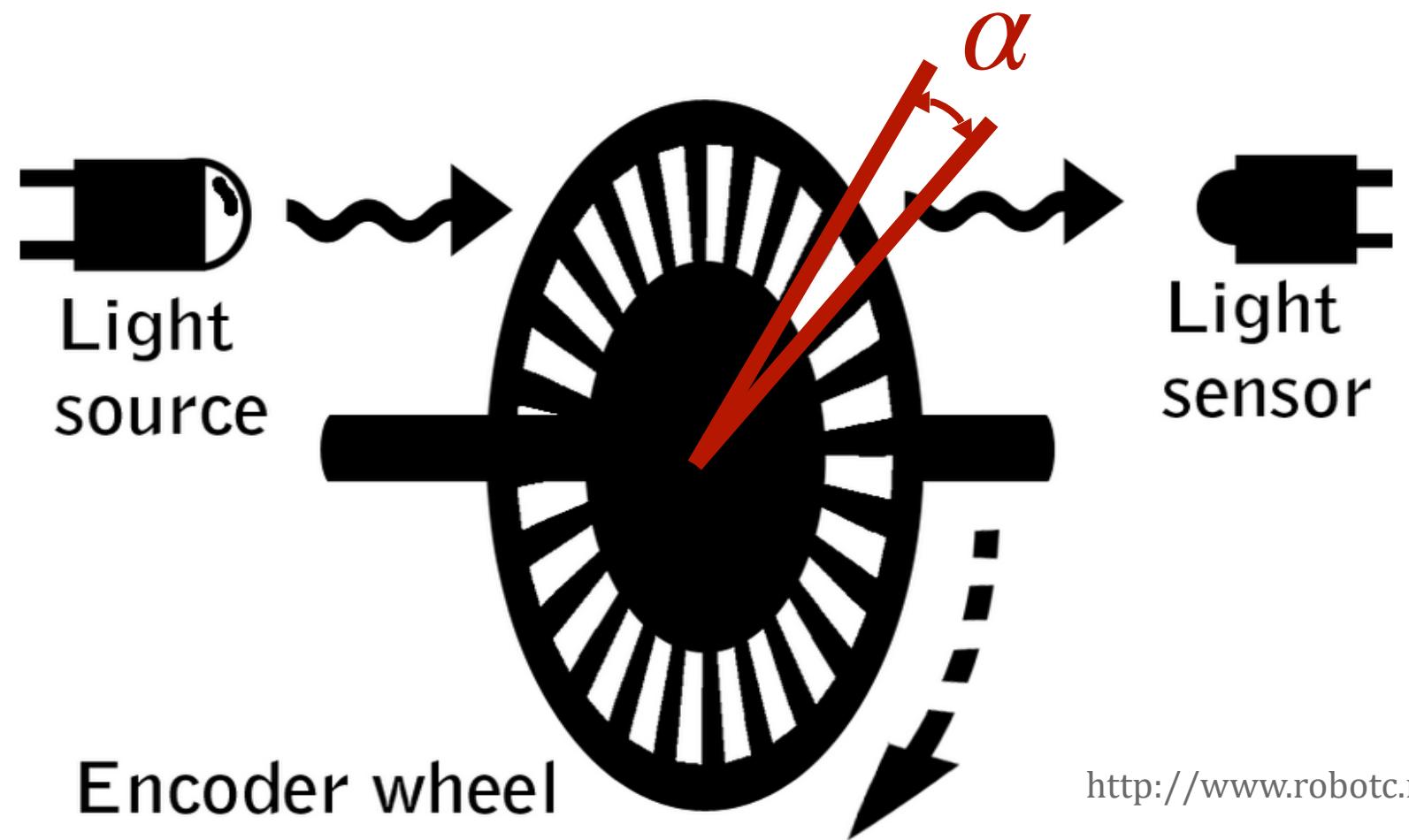
<https://bc-robotics.com/shop/wheel-encoder-sensor/>

Wheel Encoders



<http://www.robotc.net>

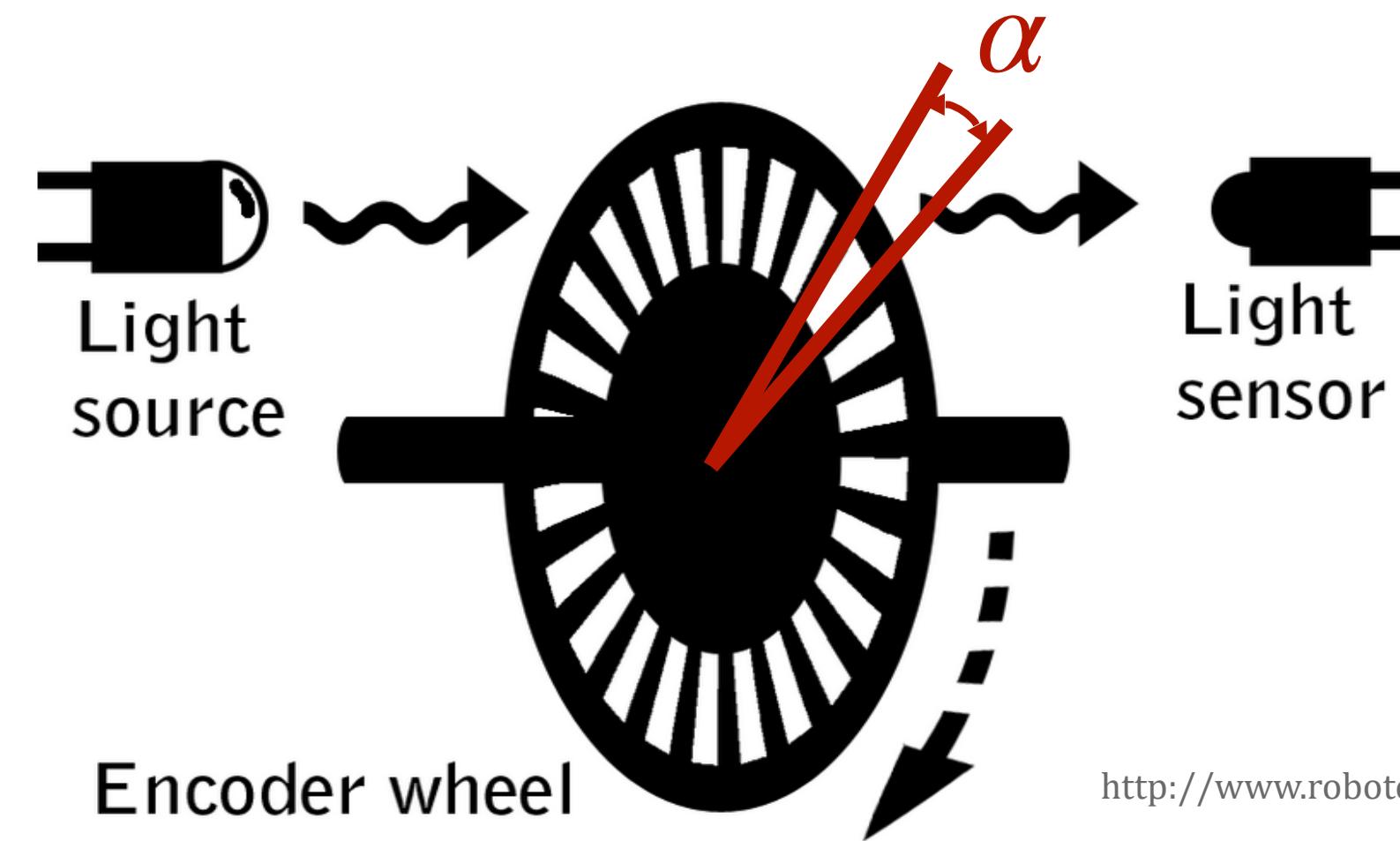
Wheel Encoders



<http://www.robotc.net>

One pulse every $\alpha = \frac{2\pi}{N_{tot}}$ radians

Wheel Encoders



<http://www.robotc.net>

One pulse every $\alpha = \frac{2\pi}{N_{tot}}$ radians

Wheel rotation (in Δt_k): $\Delta\phi_k = N_k \cdot \alpha$

Angular speed: $\dot{\phi}(t_k) \simeq \frac{\Delta\phi_k}{\Delta t_k}$

$$\dot{\mathbf{q}}(t_k) = \frac{R}{2} \begin{bmatrix} \cos\theta(t_k) & 0 \\ \sin\theta(t_k) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ \frac{1}{L} \\ -\frac{1}{L} \end{bmatrix} \begin{bmatrix} \dot{\phi}_r(t_k) \\ \dot{\phi}_l(t_k) \end{bmatrix}$$

\downarrow

$$\mathbf{q}(t_{k+1}) \simeq \mathbf{q}(t_k) + \dot{\mathbf{q}}(t_k) \overbrace{(t_{k+1} - t_k)}^{\Delta t_k}$$

\downarrow

$$x(t_{k+1}) \simeq x(t_k) + \frac{R}{2\cancel{\Delta t_k}} (\Delta\phi_r(t_k) + \Delta\phi_l(t_k)) \cos\theta(t_k) \cancel{\Delta t_k}$$

$$y(t_{k+1}) \simeq y(t_k) + \frac{R}{2} (\Delta\phi_r(t_k) + \Delta\phi_l(t_k)) \sin\theta(t_k)$$

$$\theta(t_{k+1}) \simeq \theta(t_k) + \frac{R}{2L} (\Delta\phi_r(t_k) - \Delta\phi_l(t_k))$$

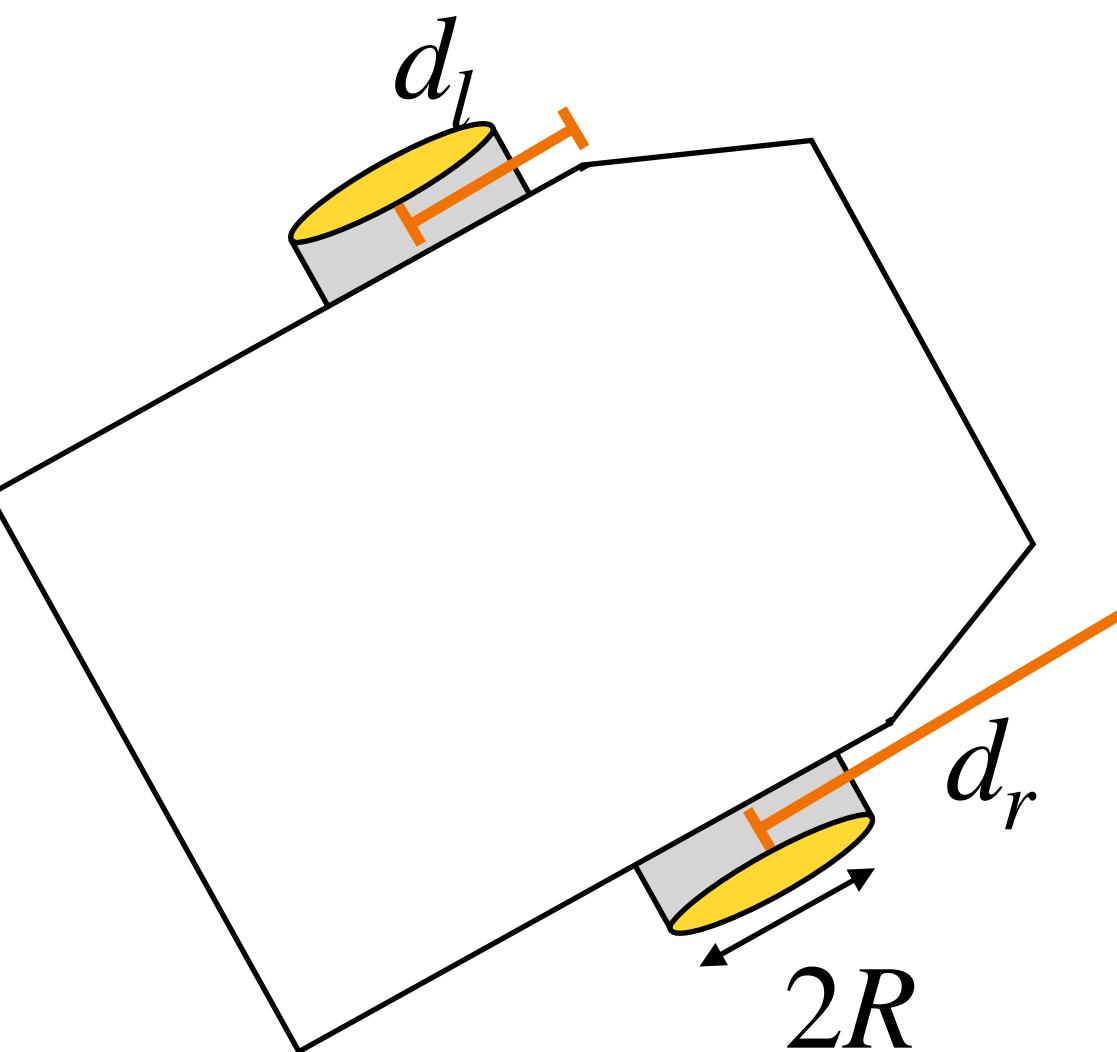
$$\dot{\phi}(t_k) \simeq \frac{\Delta\phi_k}{\Delta t_k}$$

$$x(t_{k+1}) \simeq x(t_k) + \frac{R}{2} (\Delta\phi_r(t_k) + \Delta\phi_l(t_k)) \cos\theta(t_k)$$

$$y(t_{k+1}) \simeq y(t_k) + \frac{R}{2} (\Delta\phi_r(t_k) + \Delta\phi_l(t_k)) \sin\theta(t_k)$$

$$\theta(t_{k+1}) \simeq \theta(t_k) + \frac{R}{2L} (\Delta\phi_r(t_k) - \Delta\phi_l(t_k))$$

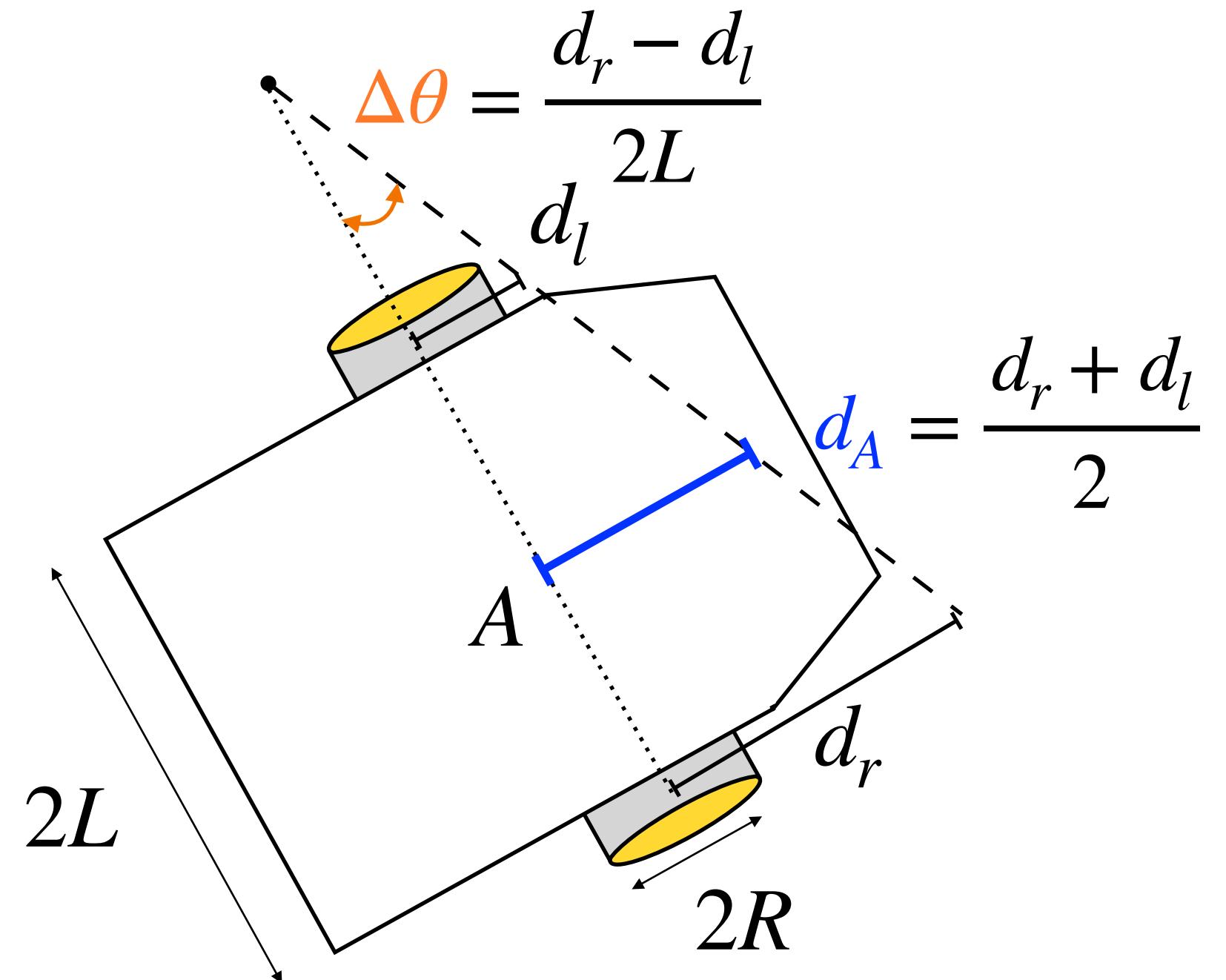
Wheel travelled distance: $d_{r/l} = R\Delta\phi_{r/l}$



$$x(t_{k+1}) \simeq x(t_k) + \frac{d_r(t_k) + d_l(t_k)}{2} \cos \theta(t_k)$$

$$y(t_{k+1}) \simeq y(t_k) + \frac{d_r(t_k) + d_l(t_k)}{2} \sin \theta(t_k)$$

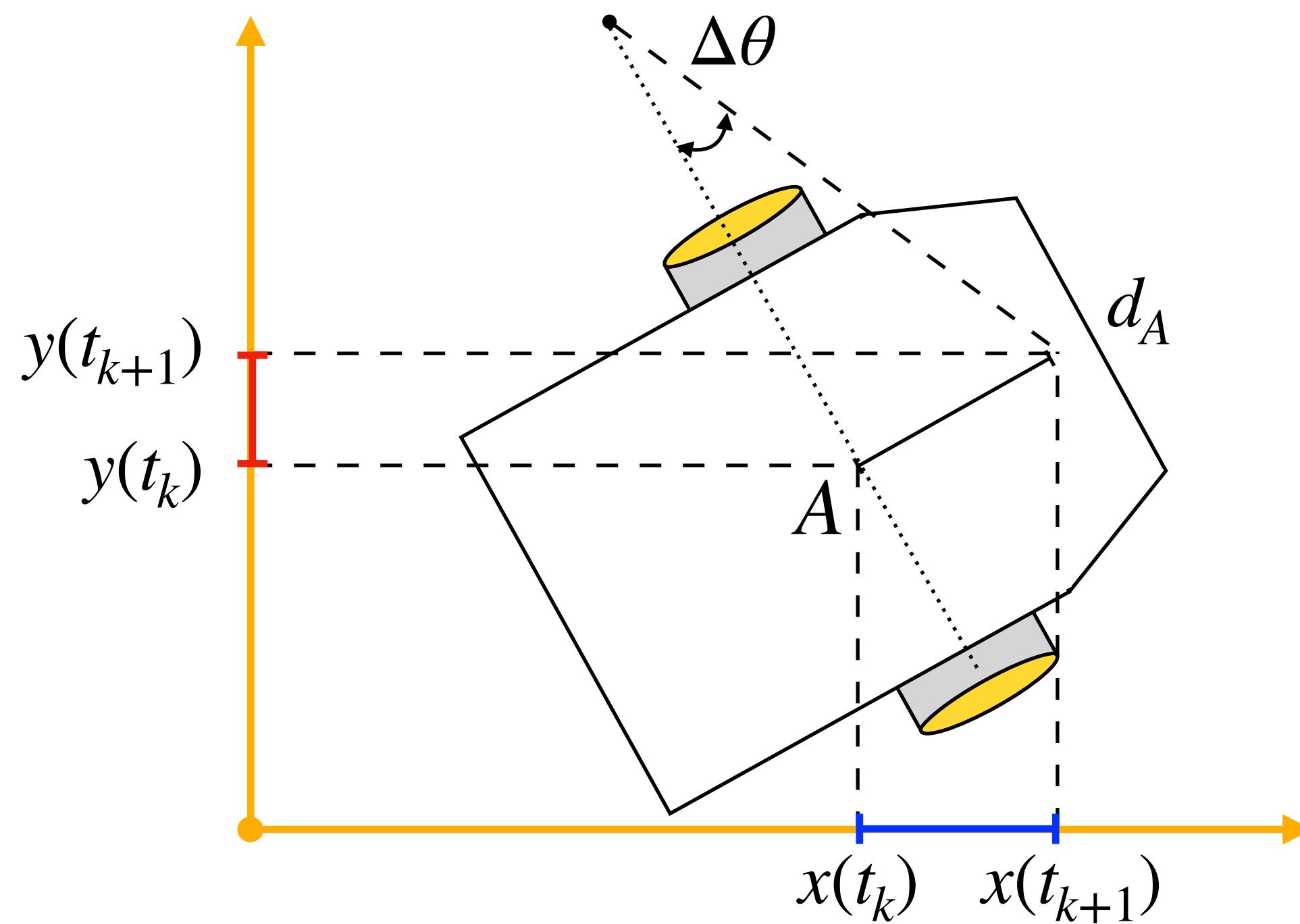
$$\theta(t_{k+1}) \simeq \theta(t_k) + \frac{d_r(t_k) - d_l(t_k)}{2L}$$



$$x(t_{k+1}) \simeq x(t_k) + d_A(t_k) \cos \theta(t_k)$$

$$y(t_{k+1}) \simeq y(t_k) + d_A(t_k) \sin \theta(t_k)$$

$$\theta(t_{k+1}) \simeq \theta(t_k) + \Delta \theta(t_k)$$



$$x(t_{k+1}) \simeq x(t_k) + d_A(t_K) \cos\theta(t_k)$$

$$y(t_{k+1}) \simeq y(t_k) + d_A(t_K) \sin\theta(t_k)$$

$$\theta(t_{k+1}) \simeq \theta(t_k) + \Delta\theta(t_k)$$

$$d_A(t_k) = \frac{d_l(t_k) + d_r(t_k)}{2}$$

$$\Delta\theta(t_k) = \frac{d_r(t_k) - d_l(t_k)}{2L}$$

$$d_{r/l}(t_k) = 2\pi R \frac{N(t_k)}{N_{tot}}$$

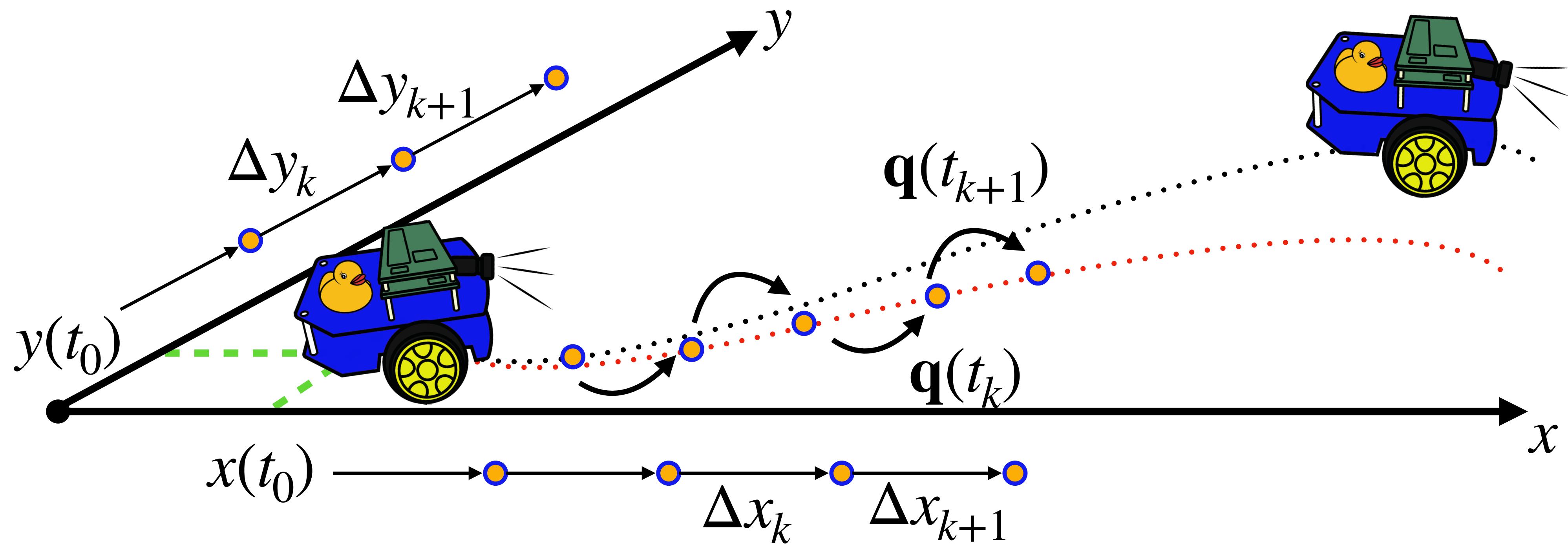
Challenges in odometry

"Dead reckoning"

$$x(t_{k+1}) \simeq x(t_k) + \Delta x(t_k)$$

$$y(t_{k+1}) \simeq y(t_k) + \Delta y(t_k)$$

$$\theta(t_{k+1}) \simeq \theta(t_k) + \Delta \theta(t_k)$$



Challenges in odometry

"Dead reckoning"

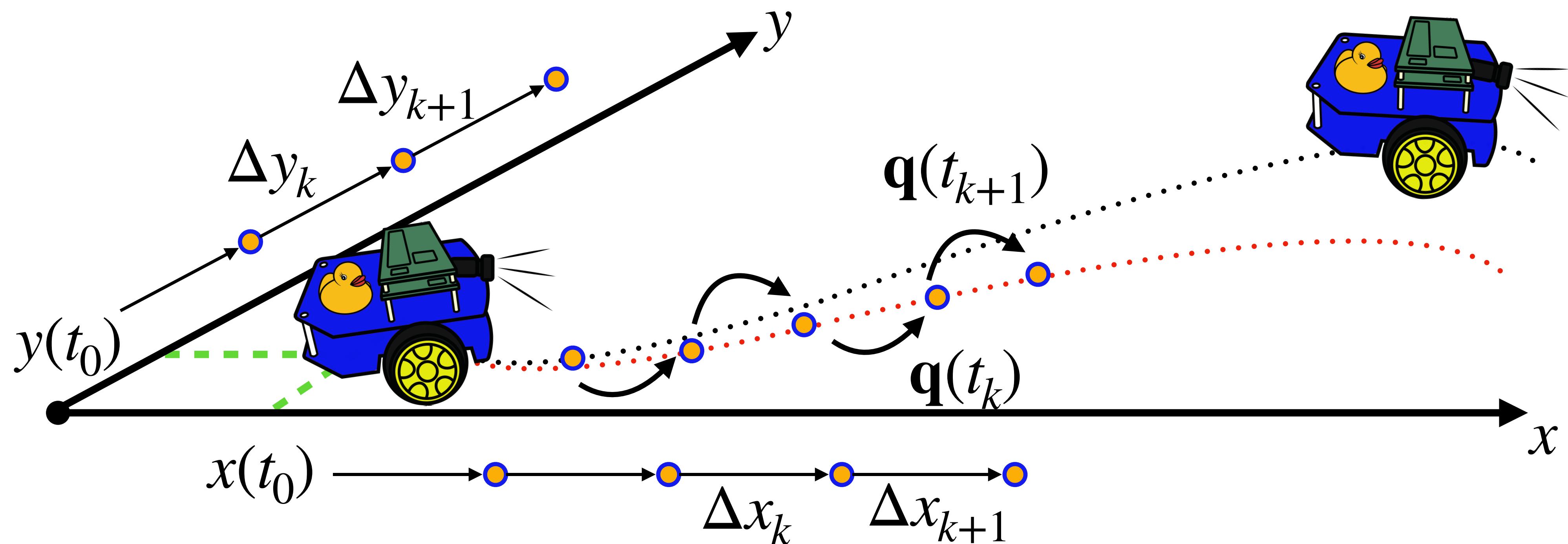
$$x(t_{k+1}) \simeq x(t_k) + \Delta x(t_k)$$

$$y(t_{k+1}) \simeq y(t_k) + \Delta y(t_k)$$

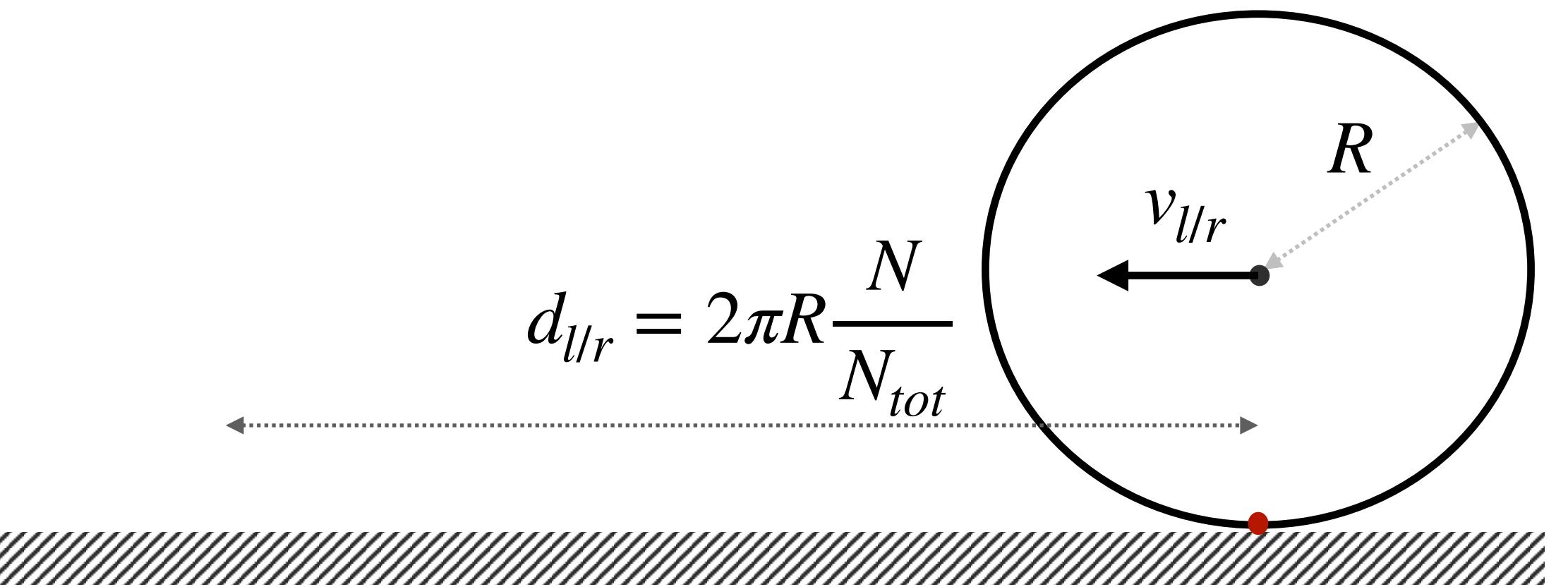
$$\theta(t_{k+1}) \simeq \theta(t_k) + \Delta\theta(t_k)$$

Kinematics model

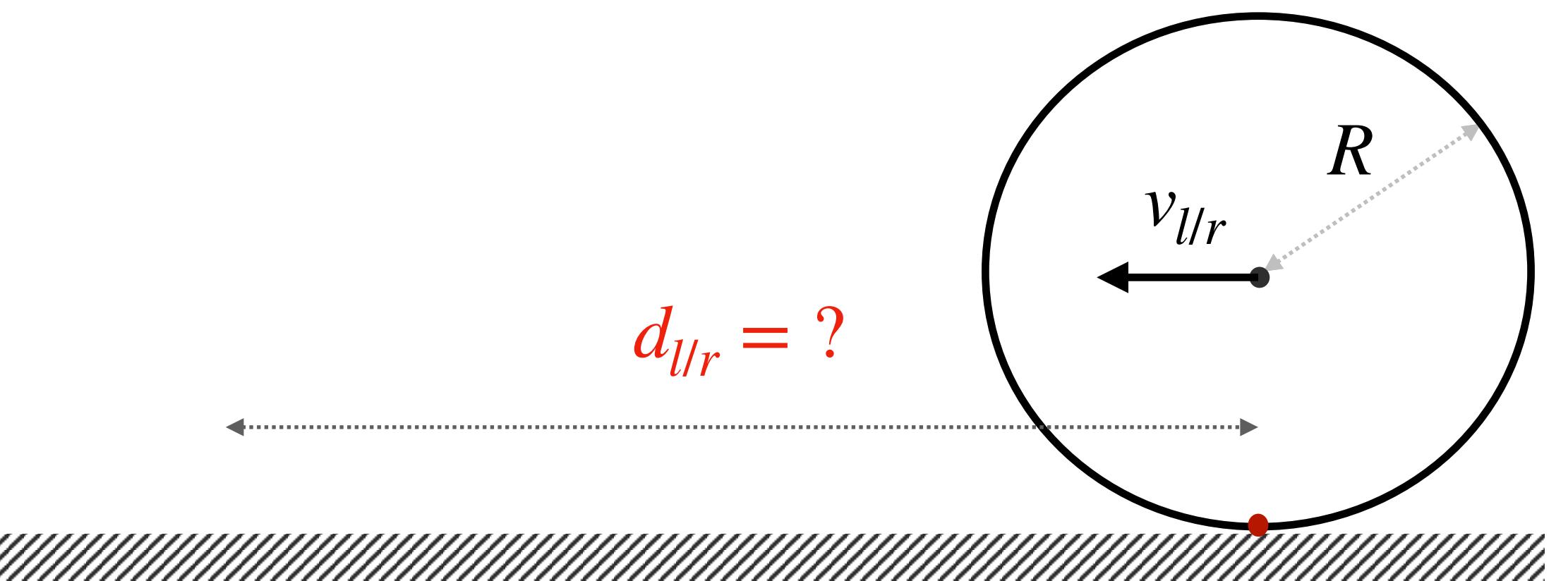
$$\dot{\mathbf{q}}(t_k) = \frac{R}{2} \begin{bmatrix} \cos\theta(t_k) & 0 \\ \sin\theta(t_k) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ \frac{1}{L} & -\frac{1}{L} \end{bmatrix} \begin{bmatrix} \dot{\phi}_r(t_k) \\ \dot{\phi}_l(t_k) \end{bmatrix}$$



$$d_{l/r} = 2\pi R \frac{N}{N_{tot}}$$



$$d_{l/r} = ?$$



Odometry calibration

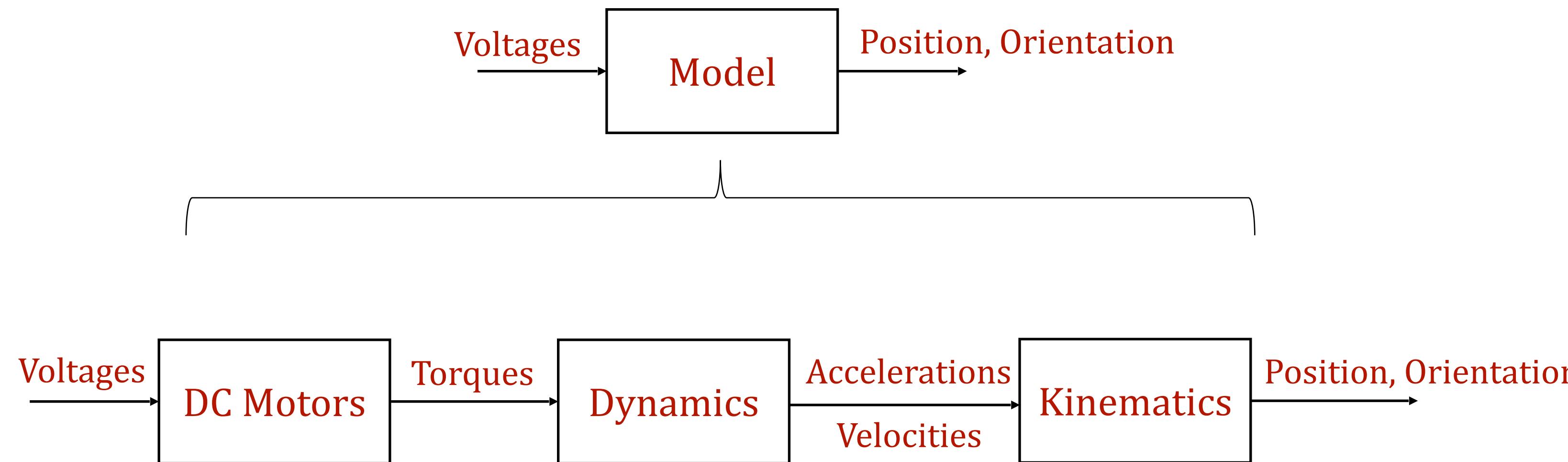
$$\Delta\theta(t_k) = \frac{d_r(t_k) - d_l(t_k)}{2\mathbf{L}}$$

$$d_{r/l}(t_k) = 2\pi\mathbf{R} \frac{N(t_k)}{N_{tot}}$$

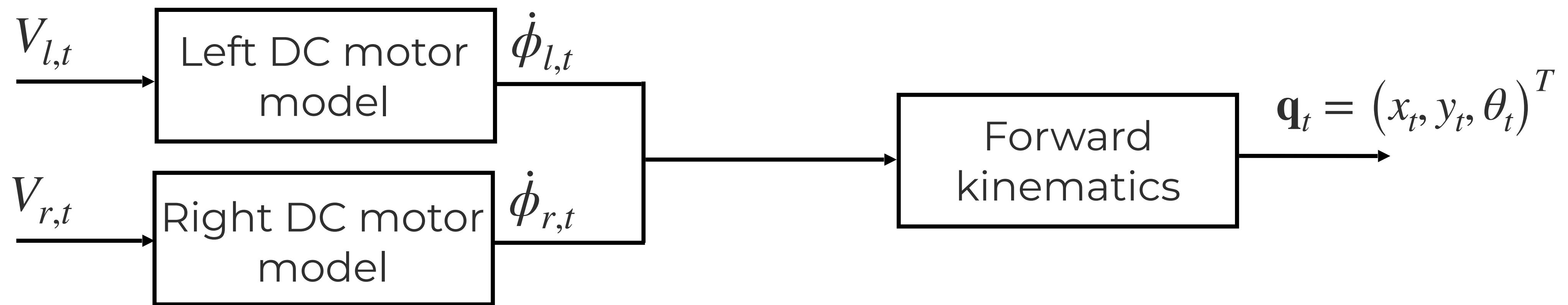
Summary

Recall Today's objective

Derive a usable model of a differential drive robot



Duckiebot model



$$\dot{\phi}_t = \left(R \frac{b}{K_i} + K_b \right)^{-1} V_t$$

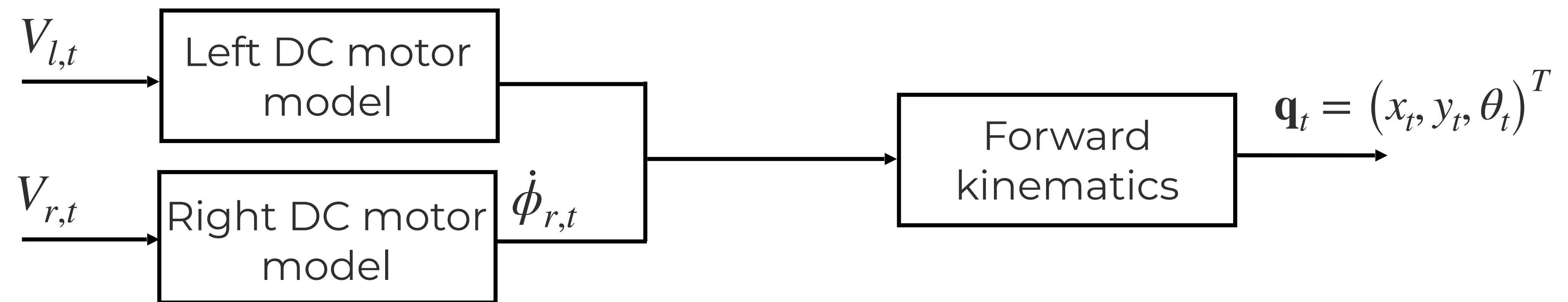
$$\dot{\mathbf{q}}_t = \frac{R}{2} \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ \frac{1}{L} & -\frac{1}{L} \end{bmatrix} \begin{bmatrix} \dot{\phi}_r \\ \dot{\phi}_l \end{bmatrix}$$

Duckiebot model

Assumption 1: robot is symmetric along longitudinal axis (x')

Assumption 2: robot is a rigid body

Assumption 3: robot wheels do not slip or skid



$$\dot{\phi}_t = \left(R \frac{b}{K_i} + K_b \right)^{-1} V_t$$

$$\dot{\mathbf{q}}_t = \frac{R}{2} \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ \frac{1}{L} & -\frac{1}{L} \end{bmatrix} \begin{bmatrix} \dot{\phi}_r \\ \dot{\phi}_l \end{bmatrix}$$

Limitations of the model:

- Geometric hypothesis:
 1. Identical wheels (of diameter)
 2. Symmetric Bot (axle length =)
 3. Symmetric
 4. Center of mass on symmetry axis
- Kinematic hypothesis
 1. Rigid body
- Kinematic constraints
 1. No lateral slipping
 2. Pure rolling
- Unmodeled dynamics
 1. Castor wheel
 2. Out of plane dynamics
 3. Friction

In Practice

- The motor driver allows us to directly set motor speeds
 - voltage-torque curve is internal
 - doesn't consider dynamics
- The angular and linear momentum of the duckiebot is sufficiently small that we can assume that velocities are instantaneously applied
 - don't really need to account for dynamics