اصول علم ربات – مبحث ۵–۴

Fundamentals of Robotics

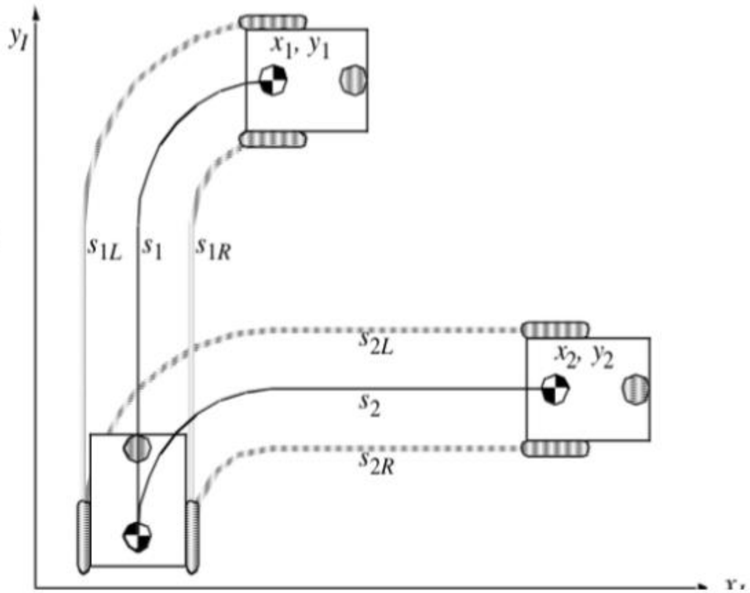
دکتر جوانمردی

زمستان ۹۹

سیستم‌های non-holonomic

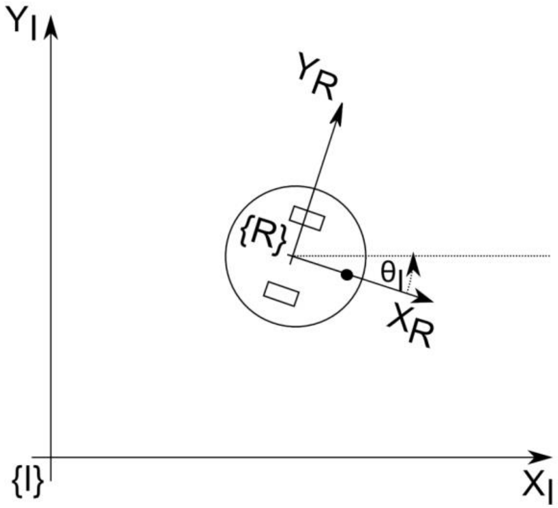
- لزوم کار با تبدیلات روی سرعت و نه موقعیت در سیستم‌های غیر–هولونومیک
- نمیتوان از روی معادلات حرکت/جابجایی به موقعیت نهایی دست یافت
- مثال: ربات دیفرانسیلی یا differential wheel robot
  مسافت پیموده شده توسط هر چرخ برای محاسبه موقعیت ربات کافی نیست!
  باید بدانیم این حرکت در مرور زمان چگونه اتفاق افتاده است.

$$s_1 = s_2, s_{1R} = s_{2R}, s_{1L} = s_{2L}$$
$$x_1 \neq x_2, y_1 \neq y_2$$



نیاز به کار با تبدیلات روی سرعت و نه موقعیت
در موبایل‌های متحرک

سینماتیک مستقیم ربات دیفرانسیلی differential wheel robot



$$\boxed{{}^I\xi = [{}^I x, {}^I y, {}^I\theta]^T \longrightarrow \text{هدف ما}}$$

$$\dot{x}_I = cos(\theta)\dot{x}_R - sin(\theta)\dot{y}_R.$$

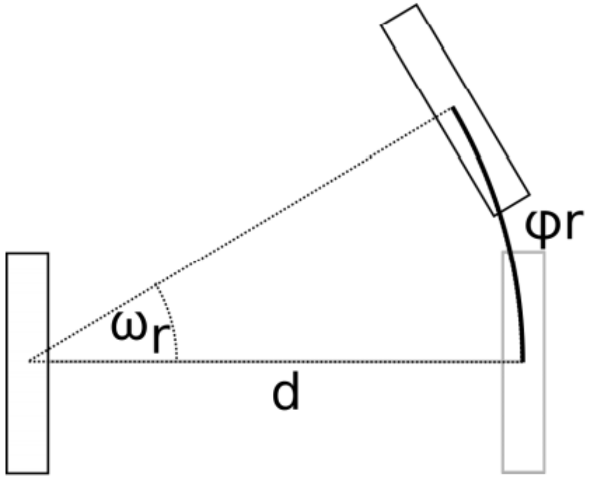
$$\dot{y}_I = sin(\theta)\dot{x}_R + cos(\theta)\dot{y}_R$$

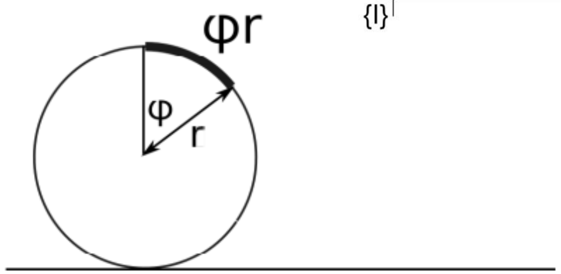
$$\dot{\theta}_I = \dot{\theta}_R$$

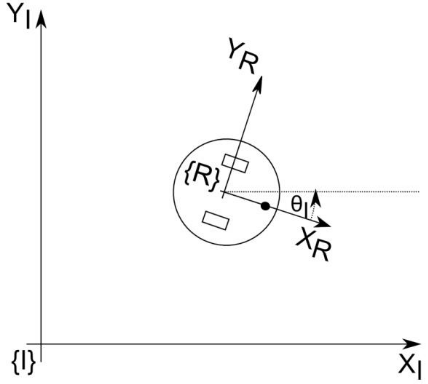
$${}^I_R T(\theta) = \begin{pmatrix} cos(\theta) & -sin(\theta) & 0 \\ sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\boxed{\dot{\xi}_I = {}^I_R T(\theta)\dot{\xi}_R}$$

سینماتیک مستقیم ربات دیفرانسیلی differential wheel robot







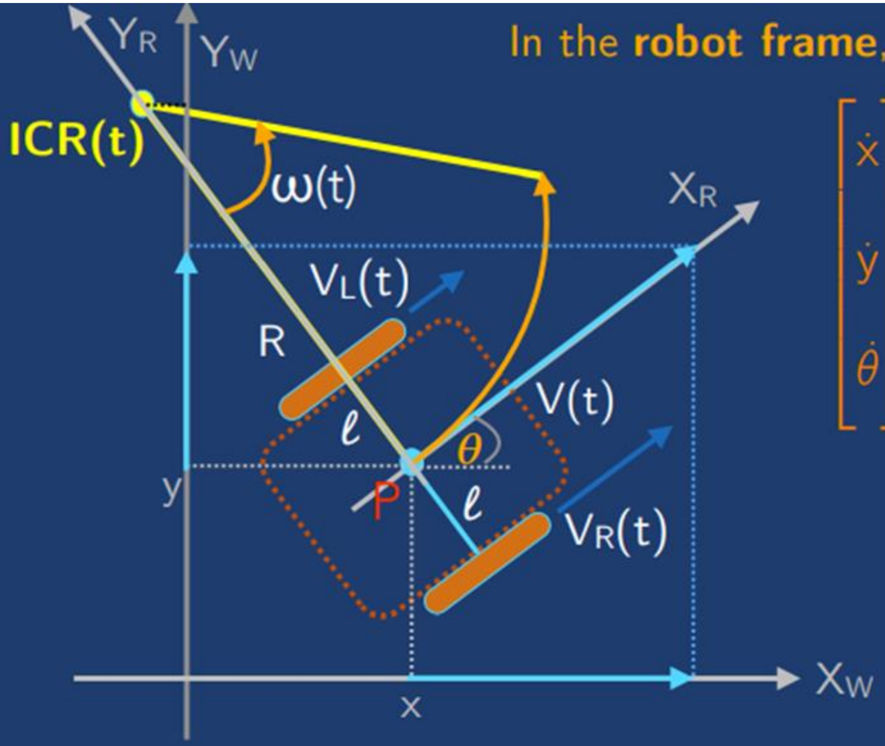
$$\dot{x_R} = \frac{r\dot{\phi_l}}{2} + \frac{r\dot{\phi_r}}{2}$$

$$\dot{\theta} = \frac{\dot{\phi_r} r}{d} - \frac{\dot{\phi_l} r}{d}$$

$$\omega_r d = \phi_r r$$

$$\boxed{\dot{\xi_I} = {}^I_R T(\theta)\dot{\xi_R}} \longrightarrow \boxed{\begin{pmatrix} \dot{x_I} \\ \dot{y_I} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} cos(\theta) & -sin(\theta) & 0 \\ sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{r\dot{\phi_l}}{2} + \frac{r\dot{\phi_r}}{2} \\ 0 \\ \frac{\dot{\phi_r} r}{d} - \frac{\dot{\phi_l} r}{d} \end{pmatrix}}$$

سینماتیک مستقیم ربات دیفرانسیلی differential wheel robot



In the **robot frame**, given the spinning velocity controls for wheels

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix}_R = \begin{bmatrix}\frac{r\dot{\varphi}_R(t)+r\dot{\varphi}_L(t)}{2}\\ 0\\ \frac{r\dot{\varphi}_R(t)-r\dot{\varphi}_L(t)}{2\ell}\end{bmatrix} = \begin{bmatrix}\frac{v_R(t)+v_L(t)}{2}\\ 0\\ \frac{v_R(t)-v_L(t)}{2\ell}\end{bmatrix}$$

Two control inputs, linear and angular velocity

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix}_R = \begin{bmatrix}v(t)\\ 0\\ \omega(t)\end{bmatrix}$$

In the **world reference frame**, given the local inputs v(t), ω(t)

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{\theta}\end{bmatrix}_W = \begin{bmatrix}\cos(\theta(t)) & -\sin(\theta(t)) & 0\\ \sin(\theta(t)) & \cos(\theta(t)) & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}v(t)\\ 0\\ \omega(t)\end{bmatrix}_R$$

$$\dot{x} = v(t)\cos(\theta(t))$$
$$\dot{y} = v(t)\sin(\theta(t))$$
$$\dot{\theta} = \omega(t)$$

Source: CMU

از سینماتیک مستقیم به اودومتری

$$\dot{\xi}_I = {}^I_R T(\theta)\dot{\xi}_R$$

- سینماتیک مستقیم ⟵ رابطه بین سرعت دوران چرخ‌ها و سرعت ربات در فریم $\{I\}$
- Odometry: محاسبه موقعیت ربات در فریم $\{I\}$ از روی سرعت آن
- انتگرال گرفتن از رابطه سرعت بین زمان 0 تا T
- کامپیوتر ربات با زمان گسسته کار میکند ⟵ تبدیل انتگرال به جمع یا یک حلقه for

$$\begin{pmatrix} x_I(T) \\ y_I(T) \\ \theta(T) \end{pmatrix} = \int_0^T \begin{pmatrix} \dot{x}_I(t) \\ \dot{y}_I(t) \\ \dot{\theta}(t) \end{pmatrix} dt \approx \sum_{k=0}^{k=T} \begin{pmatrix} \Delta x_I(k) \\ \Delta y_I(k) \\ \Delta\theta(k) \end{pmatrix} \Delta t$$

$$x_I(k+1) = x_I(k) + \Delta x(k)$$

$$\Delta x(k) \approx \dot{x}_I(t)$$

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

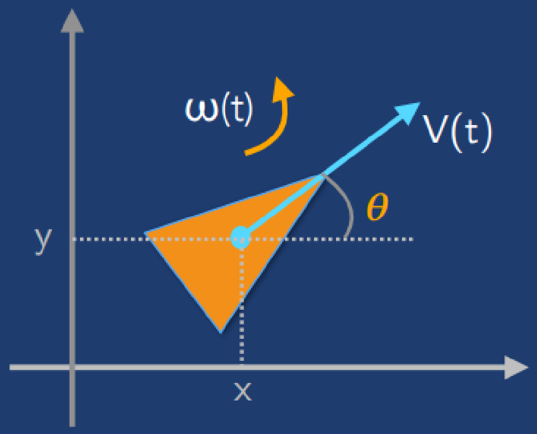
سینماتیک مستقیم کامل ربات دیفرانسیلی

For a generic robot, given v(t), ω(t) as local inputs, the velocity of pose change in the world reference frame:

$$\dot{x} = v(t)\cos(\theta(t))$$

$$\dot{y} = v(t)\sin(\theta(t))$$

$$\dot{\theta} = \omega(t)$$



If the time-profiles of the velocities are known, the equations can be integrated over time to predict the ***time trajectory:***

$$\boxed{\begin{aligned} x(t) &= \int_0^t v(t)\cos(\theta(t))\,dt \\ y(t) &= \int_0^t v(t)\sin(\theta(t))\,dt \\ \theta(t) &= \int_0^t \omega(t)\,dt \end{aligned}}$$

For a 2-wheeled differential robot

$$x(t) = \frac{1}{2}\int_0^t (v_R(t) + v_L(t))\cos(\theta(t))\,dt$$

$$y(t) = \frac{1}{2}\int_0^t (v_R(t) + v_L(t))\sin(\theta(t))\,dt$$

$$\theta(t) = \frac{1}{2\ell}\int_0^t (v_R(t) - v_L(t))\,dt$$

Source: CMU

محاسبه بردار سرعت ربات

A robot is positioned at an angle of 60 degrees with respect to the global reference frame and has wheels with a radius of 1 cm. The wheels are 2 cm from the center of the chassis. If the speeds of wheels 1 and 2, are 4 cm/s and 2 cm/s, respectively, what is the robot velocity with respect to the global reference frame?

$$\theta = \pi / 3$$

$$r = 1$$

$$l = 2$$

$$\dot{\varphi}_1 = 4$$

$$\dot{\varphi}_2 = 2$$

$$\dot{x}_{r1} = \frac{r\dot{\varphi}_1}{2}$$

$$\dot{x}_{r2} = \frac{r\dot{\varphi}_2}{2}$$

$$\omega_1 = \frac{r\dot{\varphi}_1}{2l}$$

$$\omega_2 = -\frac{r\dot{\varphi}_2}{2l}$$

$$\dot{\xi}_I = R(\theta)^{-1}\begin{bmatrix} \dot{x}_{r1} + \dot{x}_{r2} \\ 0 \\ \omega_1 + \omega_2 \end{bmatrix} = \begin{bmatrix} \cos \pi/3 & -\sin \pi/3 & 0 \\ \sin \pi/3 & \cos \pi/3 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 3.0 \\ 0 \\ 0.5 \end{bmatrix} = \begin{bmatrix} 1.5 \\ 2.5981 \\ 0.5 \end{bmatrix}$$

This robot will move instantaneously along the global reference frame x-axis with a speed of 1.5 cm/s and along the y-axis at 2.5981 cm/s while rotating with a speed of 0.5 radians/second.

سینماتیک مستقیم حالت‌های ساده و پرکاربرد

$$x(t) = \frac{1}{2}\int_0^t (v_R(t) + v_L(t)) \cos(\theta(t))\, dt$$

$$y(t) = \frac{1}{2}\int_0^t (v_R(t) + v_L(t)) \sin(\theta(t))\, dt$$

$$\theta(t) = \frac{1}{2\ell}\int_0^t (v_R(t) - v_L(t))\, dt$$

Equal (constant) forward speed for both wheels

$$v_L = v_R = v$$

Also for interval-wise changes in the common velocity

$$x(t) = x_0 + vt\cos(\theta)$$
$$y(t) = y_0 + vt\sin(\theta)$$
$$\theta(t) = \theta_0$$

The robot moves along a straight trajectory

Equal but opposite wheel speeds

$$v_L = -v_R$$

$$x(t) = x_0$$
$$y(t) = y_0$$
$$\theta(t) = \theta_0 + \frac{2vt}{2\ell}$$

The robot rotates in place

Constant (different) speeds for both wheels

$$v_L(t) = v_L, \quad v_R(t) = v_R, \quad v_L \neq v_R$$

$$x(t) = x_0 + \frac{\ell}{2}\frac{v_R + v_L}{v_R - v_L}\sin\left(\frac{t}{\ell}(v_R - v_L)\right)$$

$$y(t) = y_0 - \frac{\ell}{2}\frac{v_R + v_L}{v_R - v_L}\cos\left(\frac{t}{\ell}(v_R - v_L)\right)$$

$$\theta(t) = \theta_0 + \frac{t}{\ell}(v_R - v_L)$$

The robot moves along a circular trajectory of constant radius R

$$R = \ell\frac{v_R + v_L}{v_R - v_L}$$

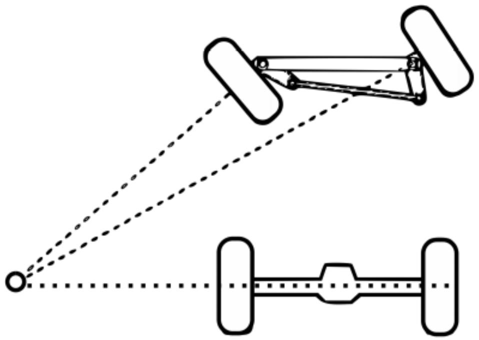
Source: CMU

سینماتیک مستقیم فرمان شبیه به خودرو

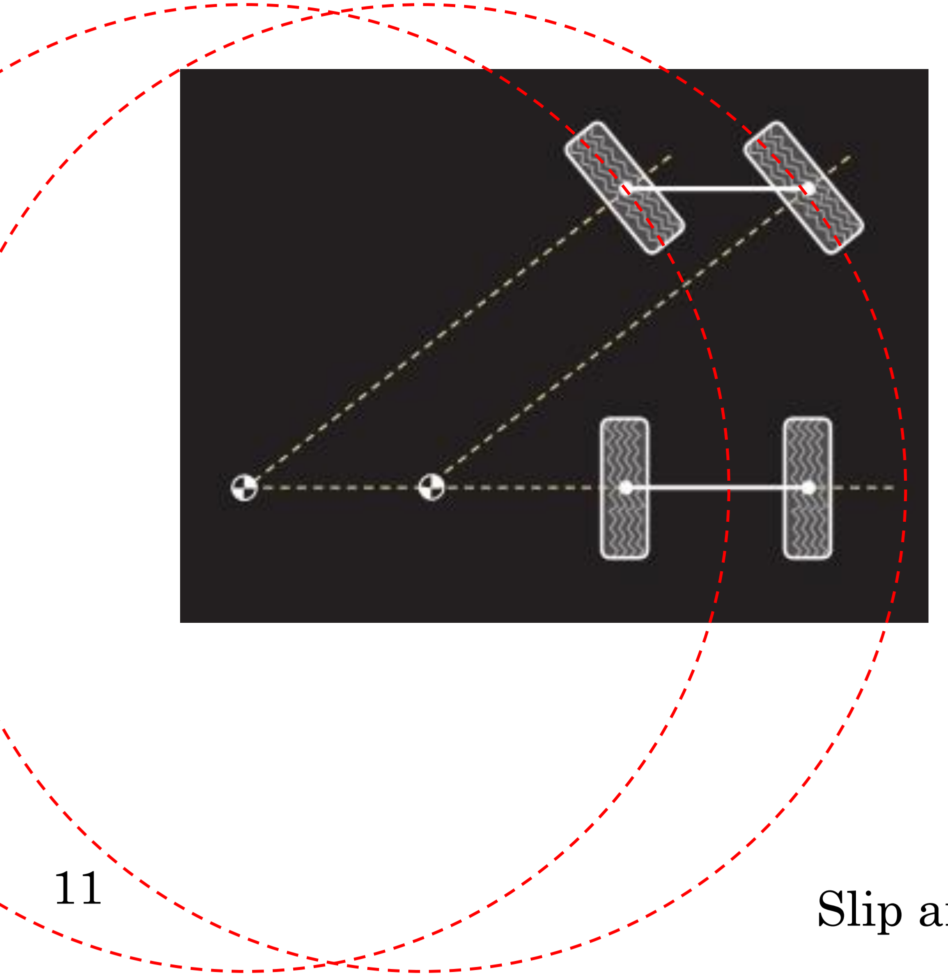
- رباتهای دیفرانسیلی از گزینههای محبوب در رباتیک
  1) ساخت و کنترل آسان
  2) non-holonomic است اما با چرخش heading در راستای هدف و جابجایی مستقیم میتواند holonomic را شبیهسازی کند

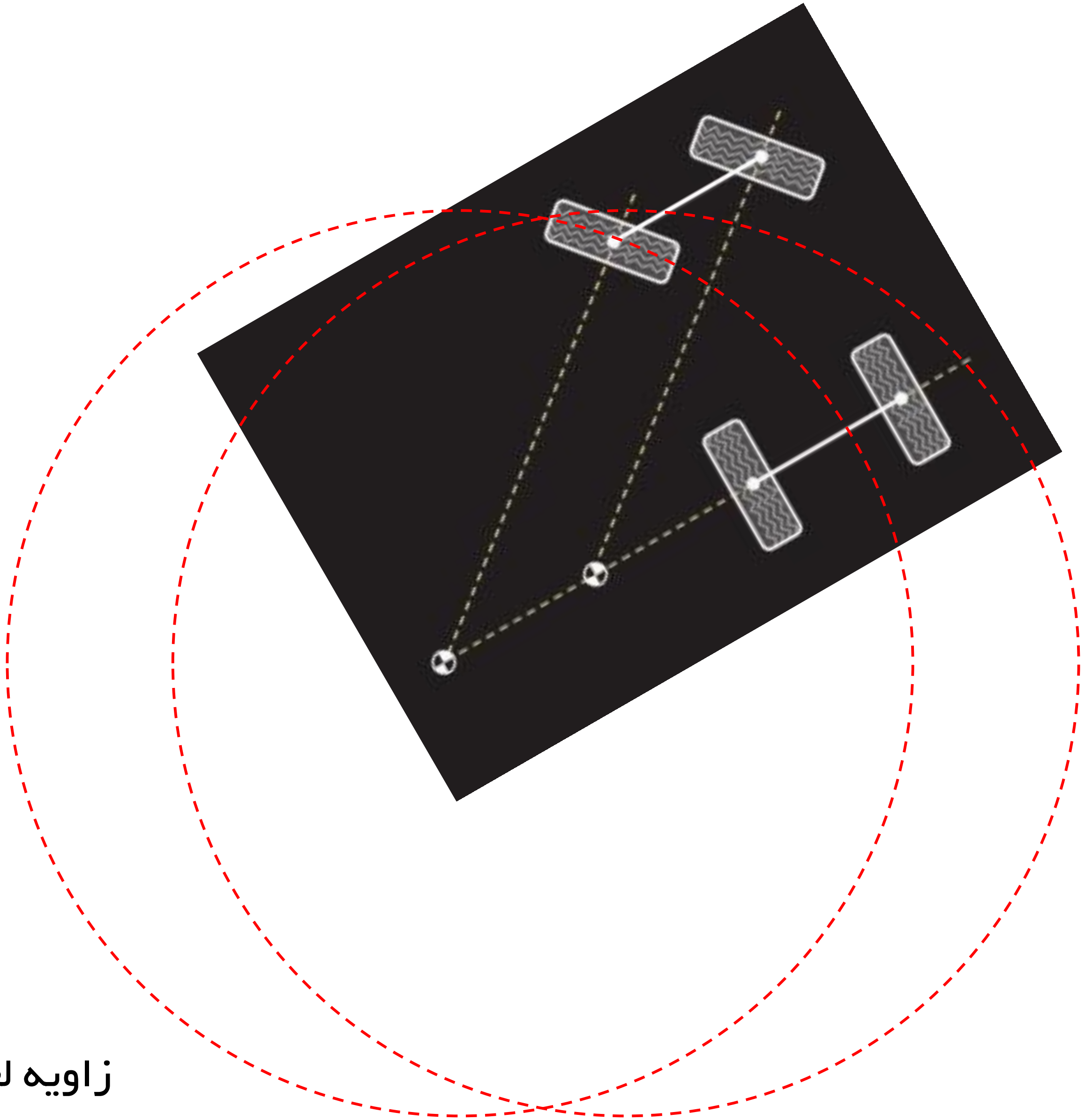
- چالشهای ربات دیفرانسیلی
  1) پیمودن مسیر مستقیم ← نیازمند هماهنگی بسیار بالا بین سرعت دوران دو چرخ
  2) حرکت با سرعت بالا

- راه حل
  - مکانیزم شبیه به خودرو ← صرفا یک موتور و فرمان دادن به چرخ جلو ← فرمان Ackermann



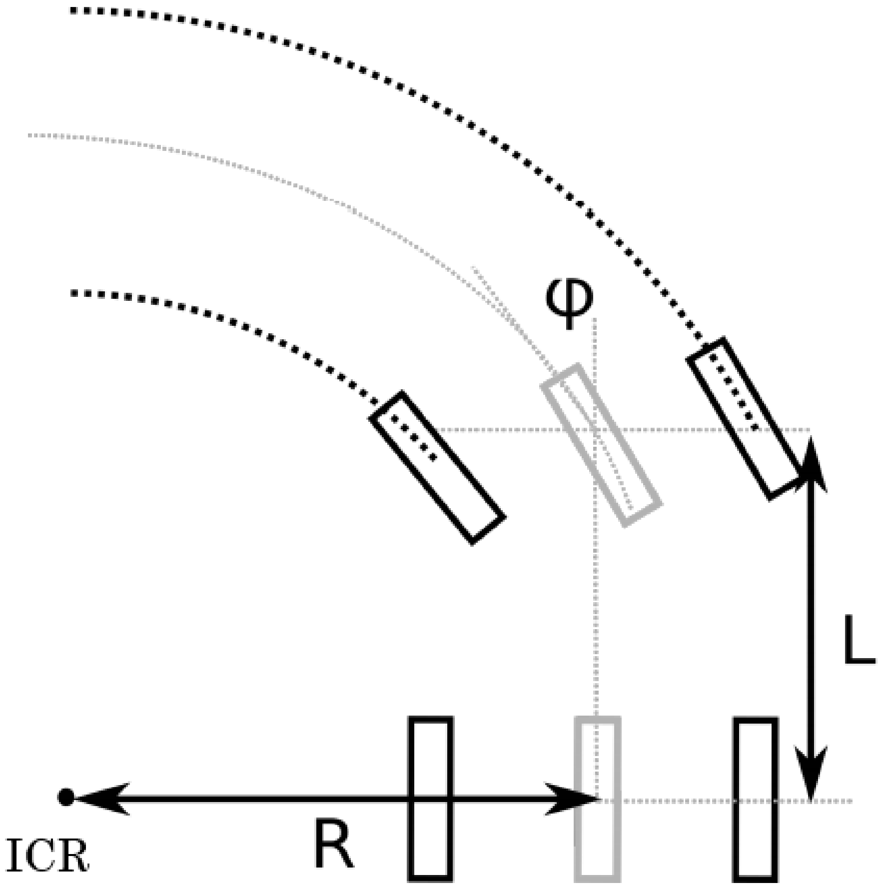
مشکل فرمان ساده – لغزش جانبی





Slip angle زاویه لغزش

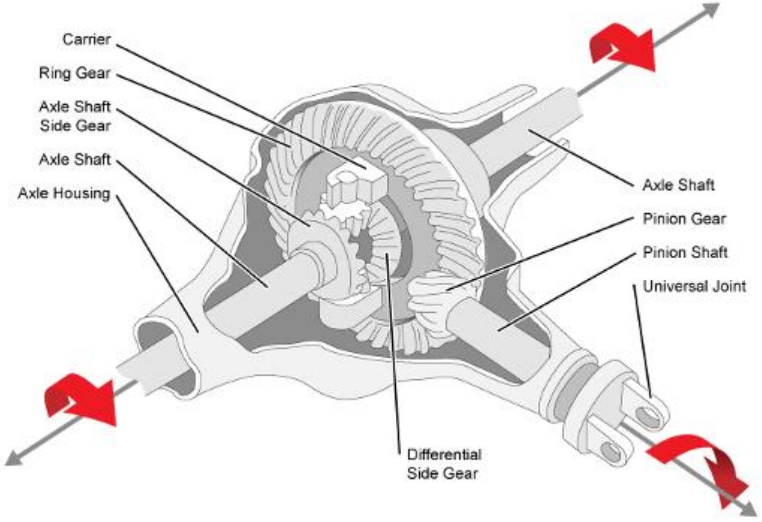
مدل فرمان Ackermann (1818)



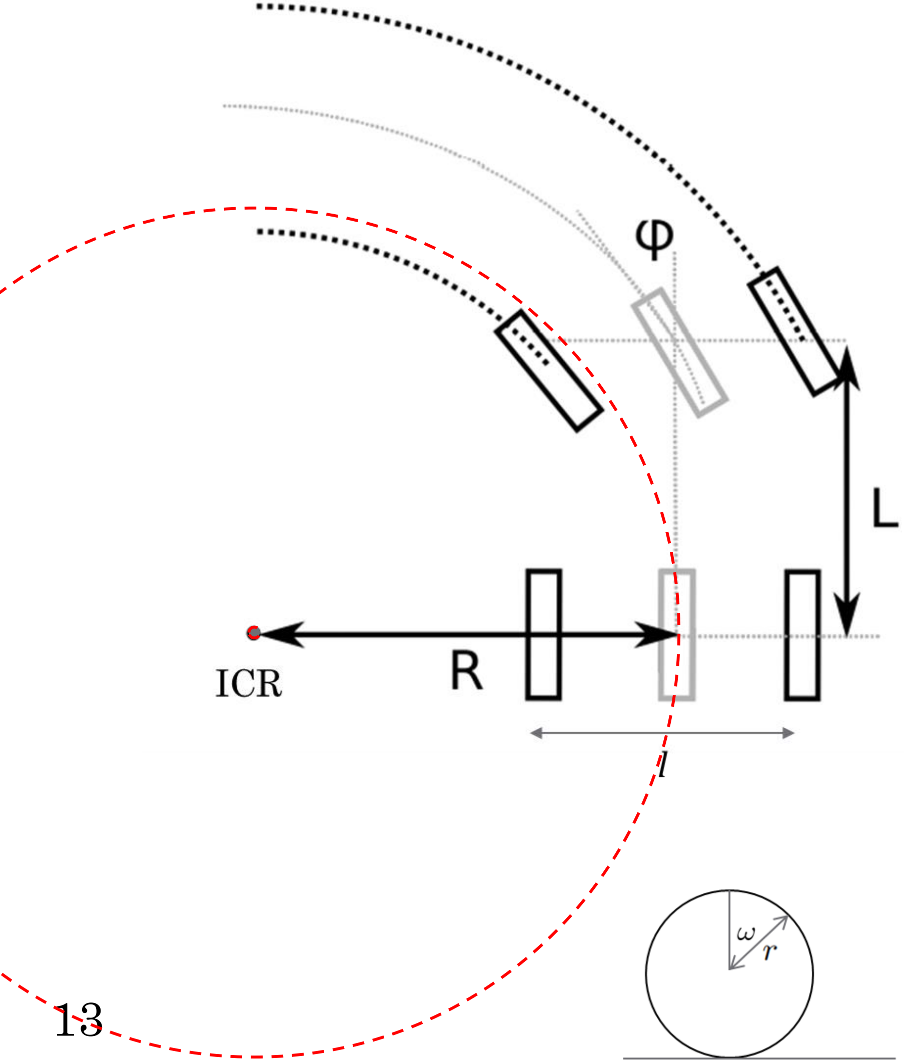
$R_2(t) > R_1(t)$

The front wheel must follow a longer path, and therefore must rotate faster than the rear wheel. With two front wheels a ***differential gear*** is necessary to implement this difference

گیربوکس دیفرانسیلی



معادلات مدل فرمان Ackermann

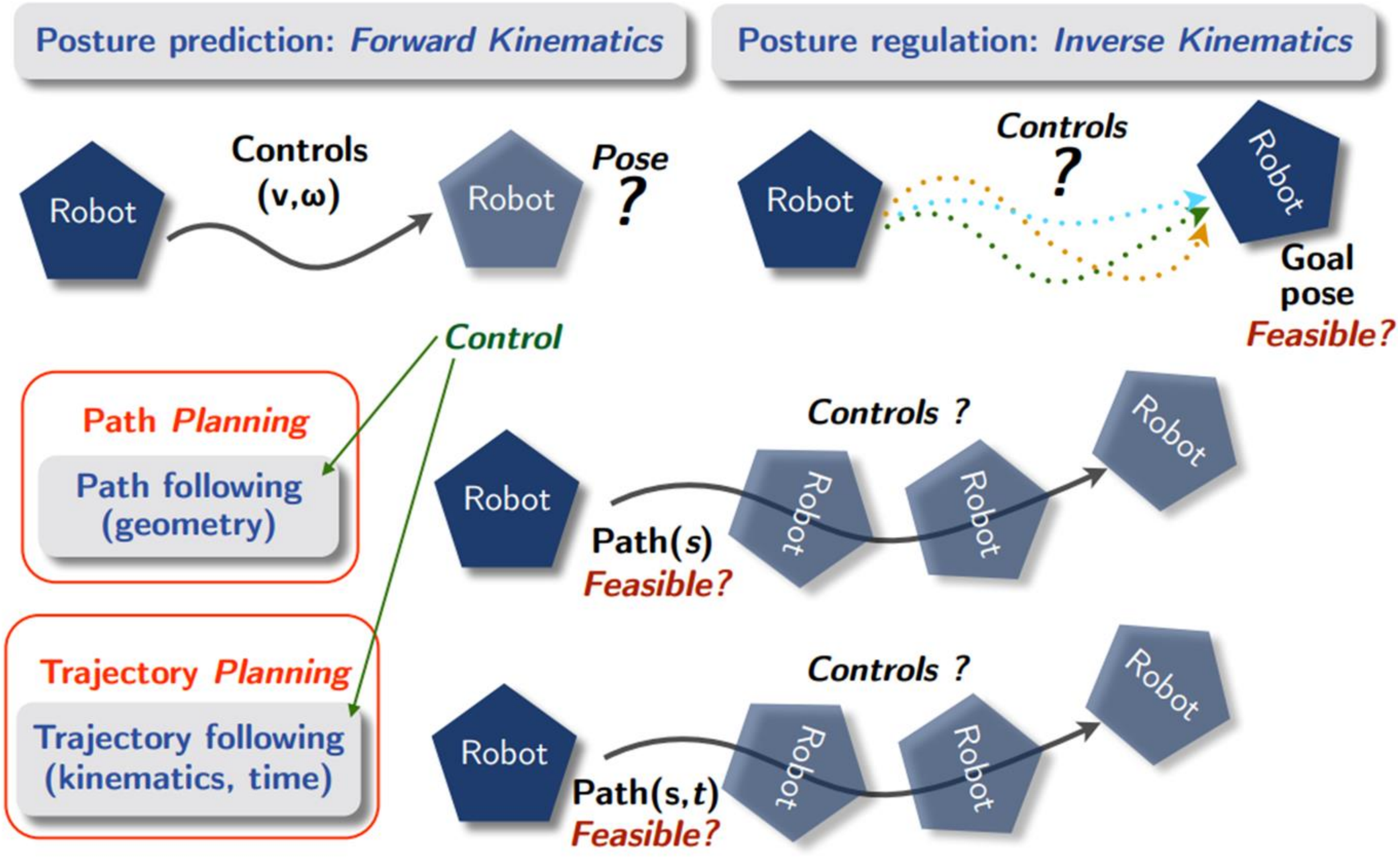


$$\tan\phi = \frac{L}{R} \longrightarrow R = \frac{L}{\tan\phi}$$

$$\begin{cases} \dfrac{L}{R - l/2} = \tan(\phi_l) \\ \dfrac{L}{R + l/2} = \tan(\phi_r) \end{cases}$$

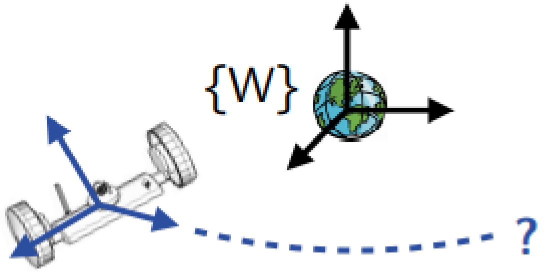
$$\dot{x}_r = \dot{\omega} r \qquad \dot{y}_r = 0 \qquad \dot{\theta}_r = \frac{\dot{\omega} r \tan\phi}{L}$$

سینماتیک مستقیم و معکوس



سینماتیک معکوس برای رباتهای متحرک

Given an initial and a goal pose, what are the velocity profiles to provide to the wheels to achieve the desired pose transition?



In the general case, a very hard problem, the presence of the non-holonomic sliding constraints makes computations difficult

$$v_L(t) = v_L,\ \ v_R(t) = v_R,\ \ v_L \neq v_R$$

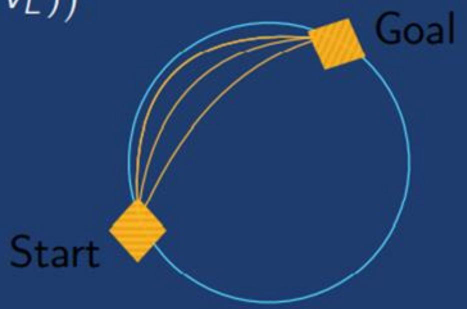
$$x(t) = x_0 + \frac{\ell}{2}\frac{v_R + v_L}{v_R - v_L}\sin\left(\tfrac{t}{\ell}(v_R - v_L)\right)$$

$$y(t) = y_0 - \frac{\ell}{2}\frac{v_R + v_L}{v_R - v_L}\cos\left(\tfrac{t}{\ell}(v_R - v_L)\right)$$

$$\theta(t) = \theta_0 + \frac{t}{\ell}(v_R - v_L)$$

Given a time t and a goal pose, the equations solve for v_L and v_R but do not provide an independent control for θ

The same final pose(t) can be achieved in many/infinite ways



Different radii, and/or multiple iterations over the same circular path to meet time requirements

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

سینماتیک معکوس ساده شده

بصورت کلی خود سینماتیک مستقیم به اندازه کافی پیچیدگی دارد،
روش حل مستقیم برای به دست آوردن سینماتیک معکوس مشکل خواهد بود

$v_L = v_R = v$

$x(t) = x_0 + vt\cos(\theta)$

$y(t) = y_0 + vt\sin(\theta)$

$\theta(t) = \theta_0$

حرکت مستقیم

مدل ساده سینماتیک مستقیم

دوران در محل

$v_L = -v_R$

$x(t) = x_0$

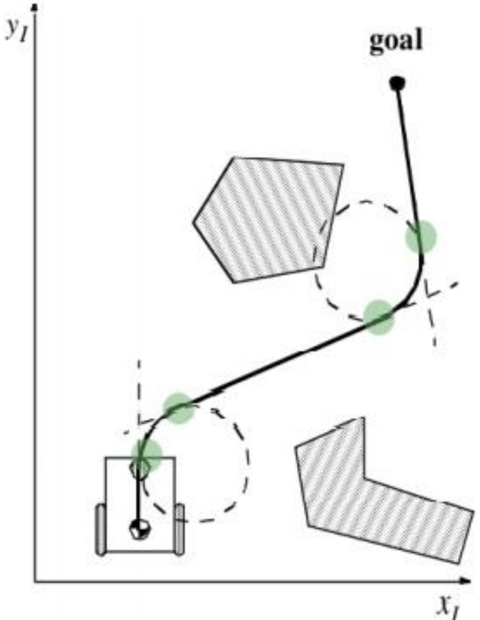
$y(t) = y_0$

$\theta(t) = \theta_0 + \frac{2vt}{2\ell}$

Solve the problem by ***decomposing the trajectory in primitive motion segments*** (very easy in open space):

- Straight lines
- Segments of a circle or rotation in place

Easier but not easy: a lot of issues to guarantee smoothness (and other quality constraints) and to deal with robot and environments' constraints



سینماتیک معکوس ساده شده

بصورت کلی خود سینماتیک مستقیم به اندازه کافی پیچیدگی دارد،
روش حل مستقیم برای به دست آوردن سینماتیک معکوس مشکل خواهد بود

$v_L = v_R = v$

$x(t) = x_0 + vt\cos(\theta)$

$y(t) = y_0 + vt\sin(\theta)$

$\theta(t) = \theta_0$

حرکت مستقیم

مدل ساده سینماتیک مستقیم

دوران در محل

$v_L = -v_R$

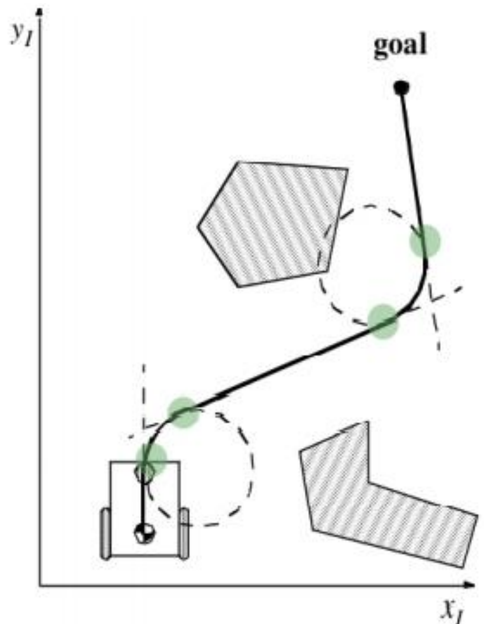
$x(t) = x_0$

$y(t) = y_0$

$\theta(t) = \theta_0 + \frac{2vt}{2\ell}$

Solve the problem by ***decomposing the trajectory in primitive motion segments*** (very easy in open space):

- Straight lines
- Segments of a circle or rotation in place

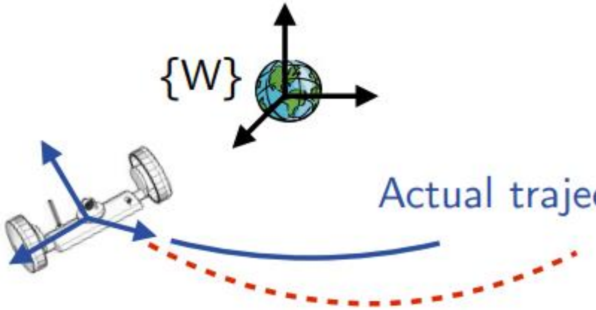


Easier but not easy: a lot of issues to guarantee smoothness (and other quality constraints) and to deal with robot and environments' constraints

محاسبه حالت / موقعیت

What is robot's *pose* in {W}
after moving at a velocity
(v,ω), for 1 minute?

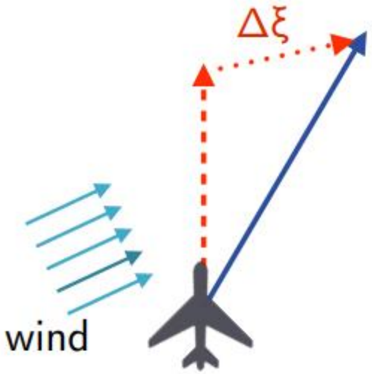
$\Delta\xi(t)$?

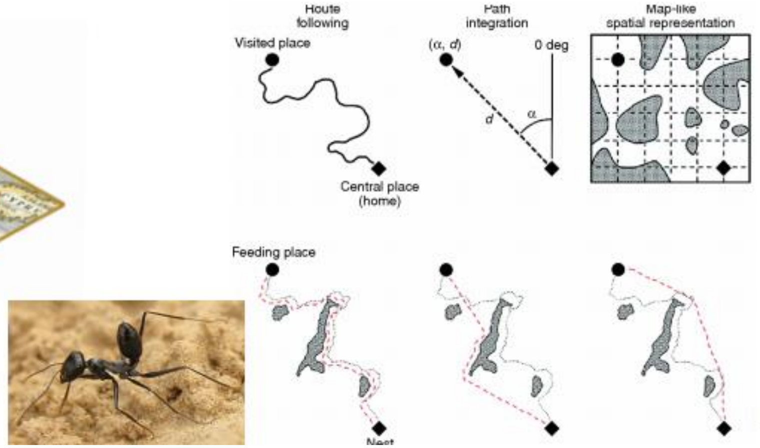


Actual trajectory $\xi(t) = [x(t)\ y(t)\ \theta(t)]$

Target trajectory $\xi_d(t) = [x_d(t)\ y_d(t)\ \theta_d(t)]$

Where am I? / What is my pose?
With respect to an initial reference point, a coordinate system, a map ....





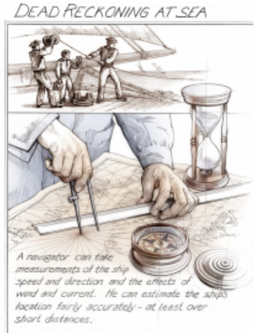
ناوبری کور یا ادومتری

In absence of an external infrastructure (e.g., GPS + Filters + Cameras) able to track the pose of the robot, ***numerical integration can be used***, based on the kinematic model of the robot and on the knowledge of the issued velocity commands, $[v(t)\ \omega(t)]^T$

➔ Incrementally build the state using on-board information

Deduced reckoning: The process of (incrementally, at discrete time steps) determining its own position based on the knowledge of some reference point (a *fix*) and the knowledge or the estimate of the velocities (speeds and headings) actuated over time. Data related to exogenous and endogenous disturbances can be included.

Time-integration of velocity vectors estimated through on-board data

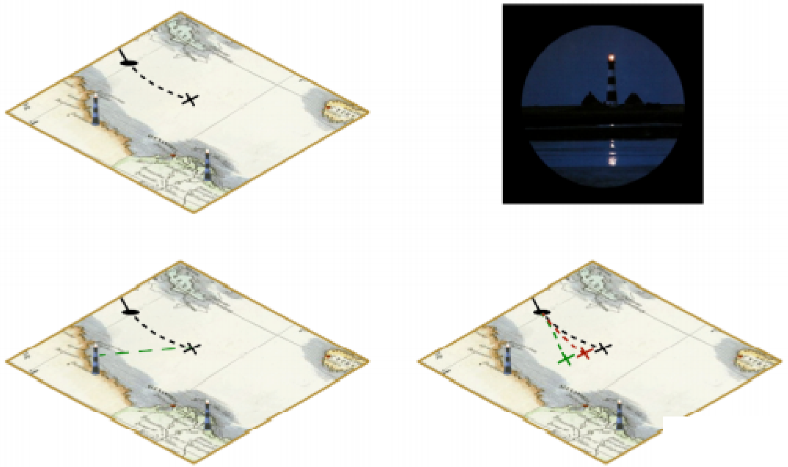


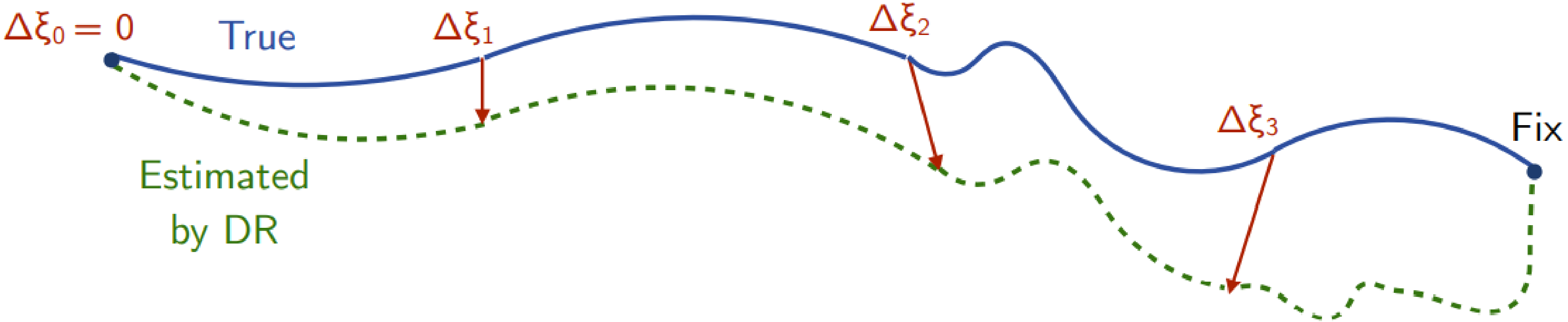
- ***Odometry*** is "basically" another way to say the same thing, that has different roots and etymology ...
- In the animal world, this is called ***path integration***

انتشار خطا در ناوبری کور

- The uncertainty of dead reckoning / odometry increases over time and maybe over distance ➔ A new fix is intermittently needed to determine a more reliable position from which a new dead reckoning process (i.e., integration) can be restarted

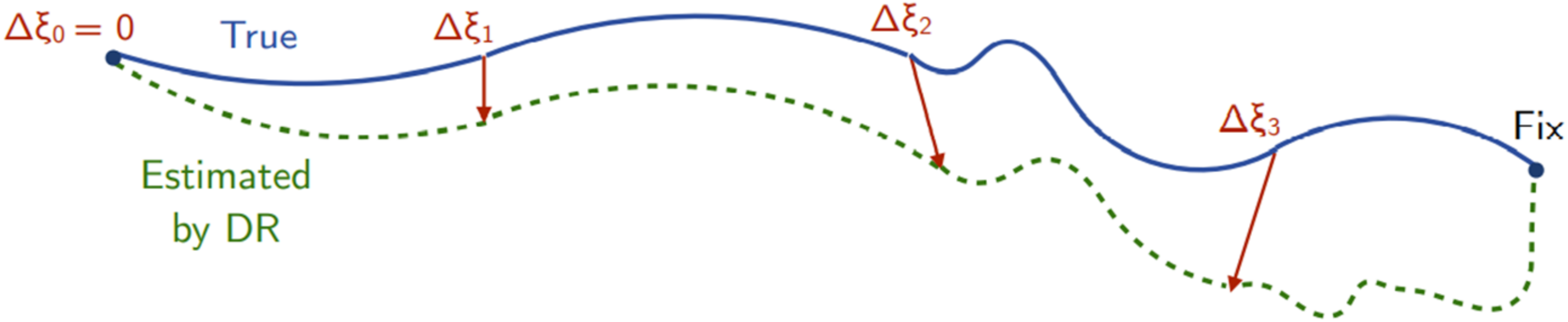
- In navigation, from where the terms comes from, lighthouses and/or celestial observations were used to get a new fix





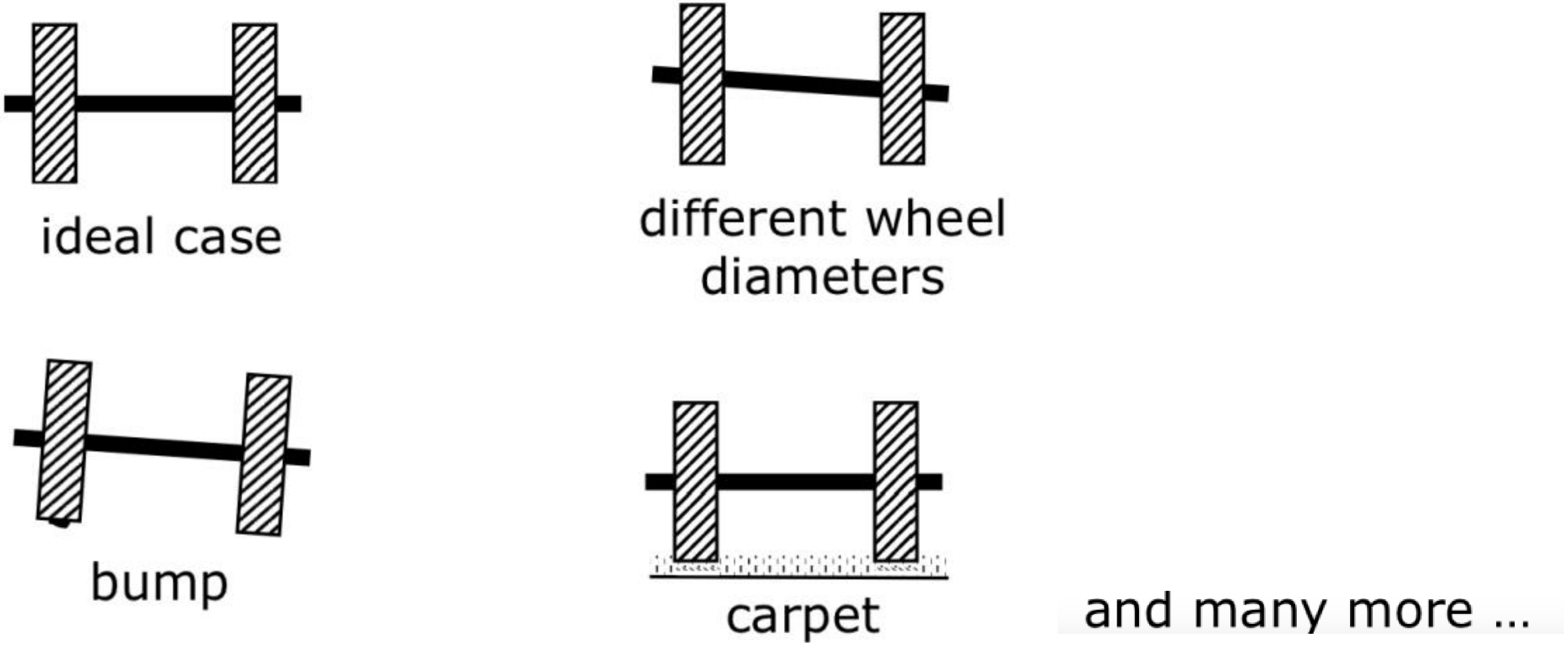
خطا در مکان‌یابی

1. Robot started in the initial configuration
2. Moved N steps in direction x, and then M steps in direction y ...
3. In general, will the new position be determined with high accuracy?



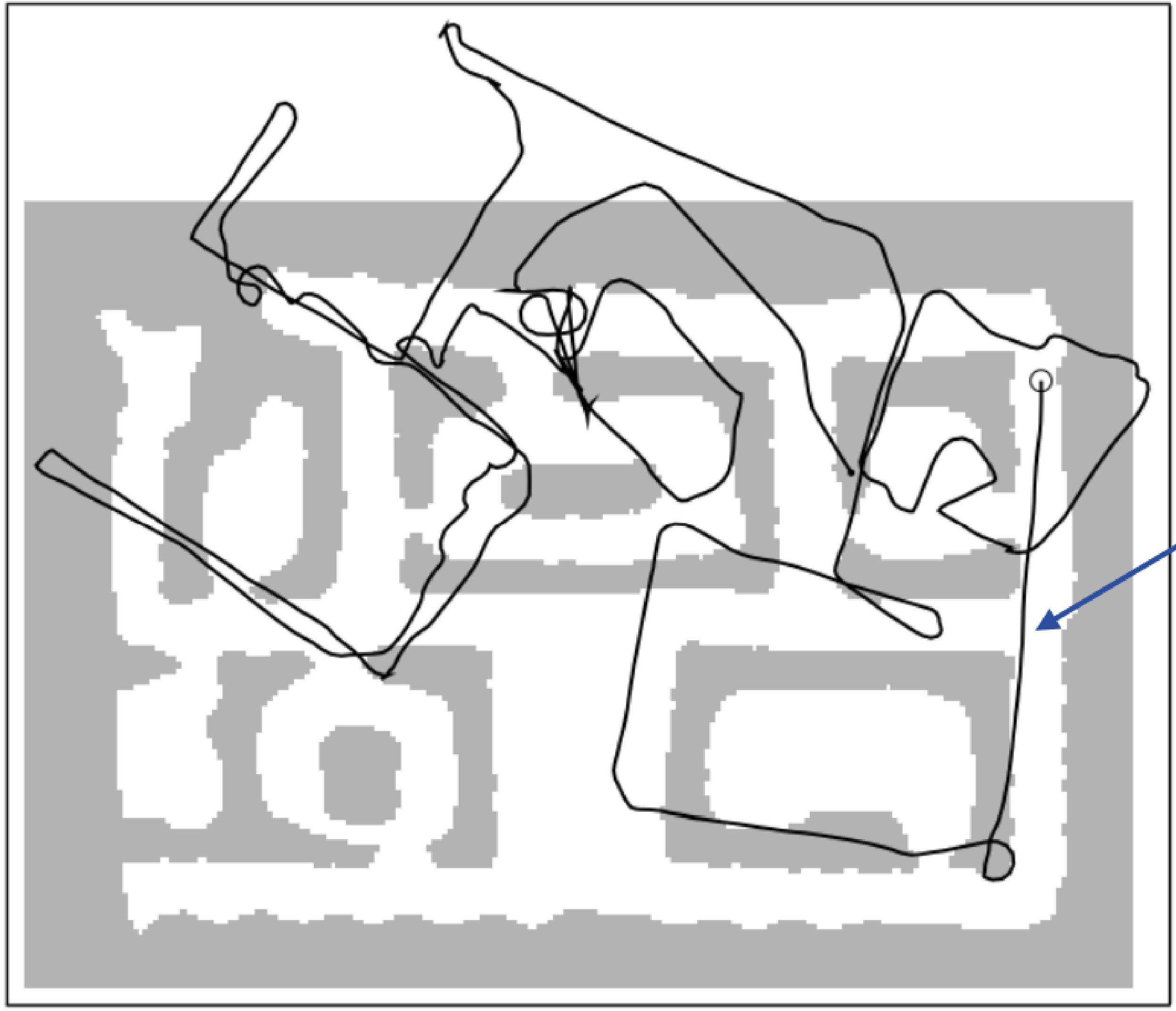
Small/Large discrepancies between issued commands and actual motion, due to friction, imprecision, approximations, ...
computations and real world intrinsically bring errors!

منشاء خطاهای مرسوم در ربات متحرک



Using wheels' measures to compute odometry is not perfect at all, it comes with a number of *uncertainties*!

نتیجه قابل انتظار در اودومتری



This the path the robot has estimated using odometry measures

جلسه آینده

انتشار خطا در ربات‌های متحرک و کنترل فیدبک دار