# Kinematic of Vehicles

Rafael Murrieta Cid

Centro de Investigación en Matemáticas (CIMAT)

murrieta@cimat.mx

June 2020

# Outline

# A Car's Model

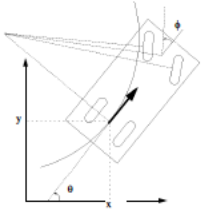



**Fig. 1.** The car-like model

**Figure:** A car-like robot

$$\begin{aligned}\dot{x} &= v\cos\theta\cos\phi \\ \dot{y} &= v\sin\theta\cos\phi \\ \dot{\theta} &= v\sin\phi\end{aligned} \tag{1}$$

$$\begin{aligned}u_1 &= v\cos\phi \\ u_2 &= v\sin\phi\end{aligned} \tag{2}$$

$$|u_1| \leq 1;\ |u_2| \leq 1;\ \phi < \frac{\pi}{4} \tag{3}$$

# A Car's Model

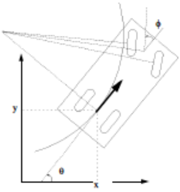



**Fig. 1.** The car-like model

**Figure:** A car-like robot

$$\dot{\mathbf{X}} = f(\mathbf{X}, \mathbf{U}) \tag{5}$$

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos\theta \\ \sin\theta \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u_2 \tag{6}$$

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \tag{7}$$

# A Second Car's Model

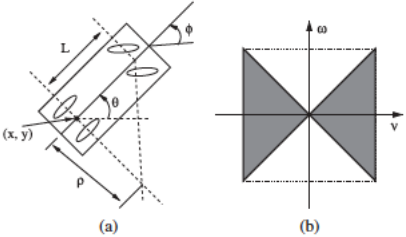
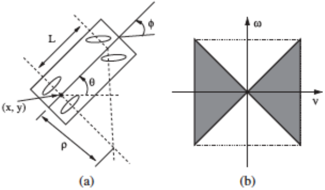


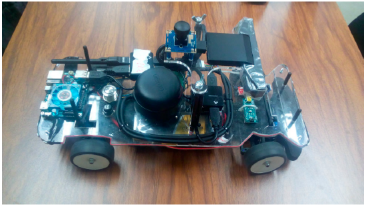

**Figure:** A second model.

$$
\begin{aligned}
\tan\phi &= \frac{L}{\rho} \\
\frac{dp}{dt} &= \rho\frac{d\theta}{dt} \\
\frac{d\theta}{dt} &= \frac{\tan\phi}{L}\frac{dp}{dt} \\
\dot{\theta} &= v\frac{\tan\phi}{L}
\end{aligned}
\tag{8}
$$

# A Second Car's Model




(a) Parameters and control space



(b) A car-like robot

**Figure:** A car-like robot

$$\begin{aligned} \dot{x} &= v\cos\theta\cos\phi \\ \dot{y} &= v\sin\theta\cos\phi \\ \dot{\theta} &= v\frac{\tan\phi}{L} \end{aligned} \tag{9}$$

$$u_1 = v\cos\phi; \;\; u_2 = v\frac{\tan\phi}{L} \tag{10}$$

# Differential Equation Modeling the Motion

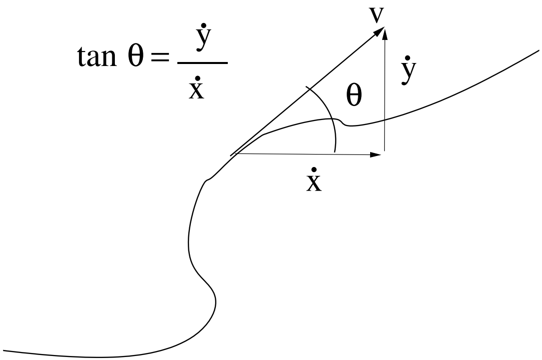
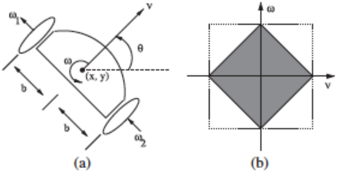



**Figure:** A car-like robot

$$\tan\theta = \frac{\dot{y}}{\dot{x}}$$
$$\sin\theta\dot{x} - \cos\theta\dot{y} = 0 \quad (11)$$

# Differential Drive Robot




(a) Differential Drive Robot

(b) Differential Drive Robot

**Figure:** A DDR

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} v \\ \omega \end{pmatrix} = \begin{pmatrix} \frac{\omega_l+\omega_r}{2} \\ \frac{\omega_l-\omega_r}{2b} \end{pmatrix}. \quad (12)$$

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \quad (13)$$

$$|\omega^{\max}| \leq \frac{1}{b}(V^{\max} - |v|). \quad (14)$$

# DDR: Second Order Dynamics

$$\dot{\mathbf{X}} = f(\mathbf{X}, \mathbf{U}) \tag{15}$$

$$\mathbf{X} = (x, y, \theta, v, \omega) \tag{16}$$

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{w}_r \\ \dot{w}_l \end{pmatrix} = \begin{pmatrix} \frac{w_r + w_l}{2} \cos\theta \\ \frac{w_r + w_l}{2} \sin\theta \\ \frac{w_r - w_l}{2b} \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} a_r + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} a_l \tag{17}$$

## Wheels Velocities $w_r$ and $w_l$ and Linear and Angular Velocities $v$ and $\omega$

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2b} & -\frac{1}{2b} \end{pmatrix} \begin{pmatrix} w_r \\ w_l \end{pmatrix} \tag{18}$$

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \tag{19}$$

$$A^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix} \tag{20}$$

$$A^{-1} = -\frac{1}{2b} \begin{pmatrix} -\frac{1}{2b} & -\frac{1}{2} \\ -\frac{1}{2b} & \frac{1}{2} \end{pmatrix} \tag{21}$$

$$\begin{pmatrix} w_r \\ w_l \end{pmatrix} = \frac{1}{2b} \begin{pmatrix} \frac{1}{2b} & \frac{1}{2} \\ \frac{1}{2b} & -\frac{1}{2} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \tag{22}$$

# Tools for Kinematic Planning with RRT

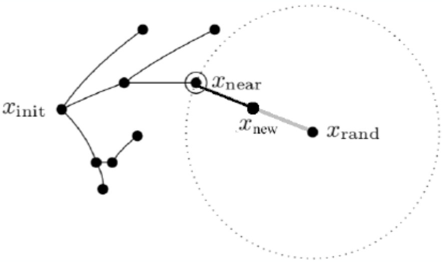



**Figure:** RRT

- How to generate the trajectory joining $x_{near}$ and $x_{new}$?
- A Two-Point Boundary Value Problem (BVP)

# Tools for Kinematic Planning with RRT

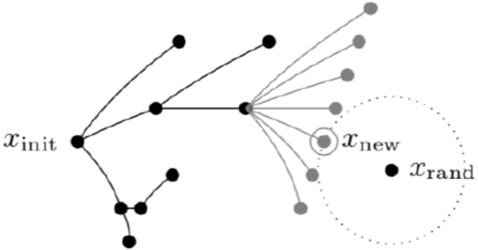



**Figure:** Generating Controls

- Answer: Simulate the controls
- Two options
  1. Choose $u_{new}$ such that $x_{new}$ is the closest to $x_{rand}$
  2. Sampling $u_{new}$ from a discret o continuous domain.

# Tools for Kinematic Planning with RRT

- $U = \{w_r, w_l\}$, where $w_i \in \{-1, 0, 1\}$
- $\Delta t$ fixed
- $(x_0, y_0, \theta_0)$ initial conditions
- Use a numerical method to integrate the state transition equation $f(\mathbf{X}, \mathbf{U})$

$$
\begin{aligned}
x_{k+1} &= x_k + \frac{w_r + w_l}{2}\cos(\theta_k)\Delta t \\
y_{k+1} &= y_k + \frac{w_r + w_l}{2}\sin(\theta_k)\Delta t \\
\theta_{k+1} &= \theta_k + \frac{w_r - w_l}{2b}\Delta t
\end{aligned} \tag{23}
$$

Metric to measure distances between state in $\mathbb{R}^2 \times S^1$.

$$d(X, X') = \sqrt{(x - x')^2 + (y - y')^2 + \alpha^2} \tag{24}$$

$$\alpha = \min\{|\theta - \theta'|, 2\pi - |\theta - \theta'|\} \tag{25}$$

# Tools for Kinematic Planning with RRT

**Algorithm 1:** BuildRRT($x_{\text{init}}, \mathcal{X}_{\text{goal}}$)

$V \leftarrow \{x_{\text{init}}\}$;
$E \leftarrow \emptyset$;
**while** $V \cap \mathcal{X}_{\text{goal}} = \emptyset$ **do**
    $x_{\text{rand}} \leftarrow \text{SampleState}()$;
    $x_{\text{near}} \leftarrow \text{NearestNeighbor}(V, x_{\text{rand}})$;
    $(x_{\text{new}}, u_{\text{new}}, \Delta t) \leftarrow \text{NewState}(x_{\text{near}}, x_{\text{rand}})$;
    **if** CollisionFree($x_{near}, x_{new}, u_{new}, \Delta t$) **then**
        $V \leftarrow V \cup \{x_{\text{new}}\}$;
        $E \leftarrow E \cup \{(x_{\text{near}}, x_{\text{new}}, u_{\text{new}}, \Delta t)\}$;
**return** $(V, E)$;

**Algorithm 2:** NewState($x_{\text{near}}, x_{\text{rand}}$)
(using fixed time step and best-input extension)

$\boldsymbol{u}_{\text{new}} \leftarrow \arg\min_{\boldsymbol{u} \in \mathcal{U}} \{\rho(\text{Simulate}(\boldsymbol{x}_{\text{near}}, \boldsymbol{u}, \Delta t), \boldsymbol{x}_{\text{rand}})\}$;
$\boldsymbol{x}_{\text{new}} \leftarrow \text{Simulate}(\boldsymbol{x}_{\text{near}}, \boldsymbol{u}_{\text{new}}, \Delta t)$ ;
**return** $(x_{\text{new}}, u_{\text{new}}, \Delta t)$;

**Figure:** Algorithms

- S. M. LaValle *Planning Algorithms*, Chapter 13, Cambridge University Press, 2006.
- T. Kunz and M. Stilman, Kinodynamic RRTs with fixed time step and best-input extension are not probabilistically complete, In Algorithmic foundations of robotics, pp. 233-244, Springer, 2015.