Introduction to Mobile Robotics

Wheeled Locomotion

Wolfram Burgard, Michael Ruhnke, Bastian Steder
Locomotion of Wheeled Robots

Locomotion (Oxford Dict.): Power of motion from place to place

- Differential drive (AmigoBot, Pioneer 2-DX)
- Car drive (Ackerman steering)
- Synchronous drive (B21)
- XR4000
- Mecanum wheels

we also allow wheels to rotate around the z axis
Instantaneous Center of Curvature

- For rolling motion to occur, each wheel has to move along its y-axis
Differential Drive

$$ICC = [x - R \sin \theta, y + R \cos \theta]$$

$$(R + l / 2) = v_r$$

$$(R - l / 2) = v_l$$

$$R = \frac{l (v_l + v_r)}{2 (v_r - v_l)}$$

$$v = \frac{v_r + v_l}{2}$$
Differential Drive: Forward Kinematics

\[
\begin{bmatrix}
  x' \\
  y' \\
  \theta'
\end{bmatrix} =
\begin{bmatrix}
  \cos(\omega \delta t) & -\sin(\omega \delta t) & 0 \\
  \sin(\omega \delta t) & \cos(\omega \delta t) & 0 \\
  0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
  x - \text{ICC}_x \\
  y - \text{ICC}_y \\
  \theta
\end{bmatrix}
+ \begin{bmatrix}
  \text{ICC}_x \\
  \text{ICC}_y \\
  \omega \delta t
\end{bmatrix}
\]

\[
x(t) = \int_0^t v(t') \cos[\theta(t')] \, dt'
\]

\[
y(t) = \int_0^t v(t') \sin[\theta(t')] \, dt'
\]

\[
\theta(t) = \int_0^t \omega(t') \, dt'
\]
Differential Drive: Forward Kinematics

\[
\begin{bmatrix}
x' \\
y' \\
\theta'
\end{bmatrix} =
\begin{bmatrix}
\cos(\omega \delta t) & -\sin(\omega \delta t) & 0 \\
\sin(\omega \delta t) & \cos(\omega \delta t) & 0 \\
0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
x - ICC_x \\
y - ICC_y \\
\theta
\end{bmatrix} +
\begin{bmatrix}
ICC_x \\
ICC_y \\
\omega \delta t
\end{bmatrix}
\]

\[
x(t) = \frac{1}{2} \int_{0}^{t} [v_r(t') + v_l(t')] \cos[\theta(t')] dt'
\]

\[
y(t) = \frac{1}{2} \int_{0}^{t} [v_r(t') + v_l(t')] \sin[\theta(t')] dt'
\]

\[
\theta(t) = \frac{1}{l} \int_{0}^{t} [v_r(t') - v_l(t')] dt'
\]
**Ackermann Drive**

\[ \text{ICC} = [x - R \sin \theta, y + R \cos \theta] \]

\[ R = \frac{d}{\tan \phi} \]

\[ \omega (R + l/2) = v_r \]

\[ \omega (R - l/2) = v_l \]

\[ R = \frac{l (v_l + v_r)}{2 (v_r - v_l)} \]

\[ \omega = \frac{v_r - v_l}{l} \]
Synchronous Drive

\[
x(t) = \int_{0}^{t} v(t') \cos[\theta(t')] dt'
\]

\[
y(t) = \int_{0}^{t} v(t') \sin[\theta(t')] dt'
\]

\[
\theta(t) = \int_{0}^{t} \omega(t') dt'
\]
XR4000 Drive

\[
x(t) = \int_0^t v(t') \cos[\theta(t')] dt'
\]

\[
y(t) = \int_0^t v(t') \sin[\theta(t')] dt'
\]

\[
\theta(t) = \int_0^t \omega(t') dt'
\]
Mecanum Wheels

\[
\begin{align*}
v_y &= \left( v_0 + v_1 + v_2 + v_3 \right) / 4 \\
v_x &= \left( v_0 - v_1 + v_2 - v_3 \right) / 4 \\
v_\theta &= \left( v_0 + v_1 - v_2 - v_3 \right) / 4 \\
v_{\text{error}} &= \left( v_0 - v_1 - v_2 + v_3 \right) / 4
\end{align*}
\]
The Kuka OmniRob Platform
Example: KUKA youBot
Tracked Vehicles
Other Robots: OmniTread

[courtesy by Johann Borenstein]
Non-Holonomic Constraints

- Non-holonomic constraints limit the possible incremental movements within the configuration space of the robot.
- Robots with differential drive or synchro-drive move on a circular trajectory and cannot move sideways.
- XR-4000 or Mecanum-wheeled robots can move sideways (they have no non-holonomic constraints).
Holonomic vs. Non-Holonomic

- Non-holonomic constraints reduce the control space with respect to the current configuration
  - E.g., moving sideways is impossible.

- Holonomic constraints reduce the configuration space.
  - E.g., a train on tracks (not all positions and orientations are possible)
Drives with Non-Holonomic Constraints

- Synchro-drive
- Differential drive
- Ackermann drive
Drives without Non-Holonomic Constraints

- XR4000 drive
- Mecanum wheels
Dead Reckoning and Odometry

- Estimating the motion based on the issued controls/wheel encoder readings
- Integrated over time
Summary

- Introduced different types of drives for wheeled robots
- Math to describe the motion of the basic drives given the speed of the wheels
- Non-holonomic constraints
- Odometry and dead reckoning