Kinematics of Wheeled Robots
https://www.youtube.com/watch?v=giS41utjlbU
Wheeled Mobile Robots

- robot can have one or more wheels that can provide
  - steering (directional control)
  - power (exert a force against the ground)
- an ideal wheel is
  - perfectly round (perimeter $2\pi r$)
  - moves in the direction perpendicular to its axis
Wheel
Deviations from Ideal

This illustration gives a good sense of the steering and throttling you'll have to input to keep your car drifting. When snapping the car from its full drift angle in one direction to full drift in the opposite direction, be prepared for the rear end to come around with more force then when initiating a drift from straight-ahead running. Give yourself plenty of space as you master your technique so you don't slap a curb or something equally immobile!

1. Steer hard into the turn to initiate a slide, then counter-steer before the car loops out.

2. Continue countersteering to maintain the slide. It's a balancing act!

3. Now you're setting up for the next turn. The goal is to drift all the way through without straightening out for more than a moment as the rear end swings around.

4. You'll have burned off some speed by now, so be prepared to pin the throttle to keep the rear wheels slipping. Traction kills drift!

5. Here's where it gets tricky. Keeping the rear end sliding through turns is relatively easy, but drifting down a straight is tough. As you lose momentum, the car will straighten out, but you'll be surprised how long you can hang it out. Good luck!

Instantaneous Center of Curvature

- for smooth rolling motion, all wheels in ground contact must
  - follow a circular path about a common axis of revolution
    - each wheel must be pointing in its correct direction
  - revolve with an angular velocity consistent with the motion of the robot
    - each wheel must revolve at its correct speed
(a) 3 wheels with roll axes intersecting at a common point (the instantaneous center of curvature, ICC). (b) No ICC exists. A robot having wheels shown in (a) can exhibit smooth rolling motion, whereas a robot with wheel arrangement (b) cannot.
Castor Wheels

- provide support but not steering nor propulsion
Differential Drive

- two independently driven wheels mounted on a common axis
Differential Drive

- Angular velocity $\omega$ about the ICC defines the wheel ground velocities $v_r$ and $v_\ell$

$$v_r = \omega(R + \frac{\ell}{2})$$

$$v_\ell = \omega(R - \frac{\ell}{2})$$

Differential Drive

- given the wheel ground velocities it is easy to solve for the radius, $R$, and angular velocity $\omega$

\[
R = \frac{\ell}{2} \frac{(v_r + v_\ell)}{(v_r - v_\ell)}
\]

\[
\omega = \frac{(v_r - v_\ell)}{\ell}
\]

- interesting cases:
  - $v_\ell = v_r$
  - $v_\ell = -v_r$
Tracked Vehicles

- similar to differential drive but relies on ground slip or skid to change direction
- kinematics poorly determined by motion of treads

Steered Wheels: Bicycle

\[ d \]

\[ 90^\circ - \alpha \]

\[ v_f \]

\[ h \]

\[ \alpha \]

ICC

\[ r \]
Steered Wheels: Bicycle

- important to remember the assumptions in the kinematic model
  - smooth rolling motion in the plane
- does not capture all possible motions
  - [http://www.youtube.com/watch?v=Cj6ho1-G6tw&NR=1#t=0m25s](http://www.youtube.com/watch?v=Cj6ho1-G6tw&NR=1#t=0m25s)
Mecanum Wheel

- a normal wheel with rollers mounted on the circumference


- https://www.youtube.com/watch?v=O7FbDy-gE70
- https://www.youtube.com/watch?v=mUoftURFsxM
Mecanum Wheel

<table>
<thead>
<tr>
<th>Direction of Movement</th>
<th>Wheel Actuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forward</td>
<td>All wheels forward same speed</td>
</tr>
<tr>
<td>Reverse</td>
<td>All wheels backward same speed</td>
</tr>
<tr>
<td>Right Shift</td>
<td>Wheels 1, 4 forward; 2, 3 backward</td>
</tr>
<tr>
<td>Left Shift</td>
<td>Wheels 2, 3 forward; 1, 4 backward</td>
</tr>
<tr>
<td>CW Turn</td>
<td>Wheels 1, 3 forward; 2, 4 backward</td>
</tr>
<tr>
<td>CCW Turn</td>
<td>Wheels 2, 4 forward; 1, 3 backward</td>
</tr>
</tbody>
</table>

*To the right:* This is a top view looking down on the drive platform. Wheels in Positions 1, 4 should make X-pattern with Wheels 2, 3. If not set up like shown, wheels will not operate correctly.

AndyMark Mecanum wheel specification sheet
http://d1pytrrjwm20z9.cloudfront.net/MecanumWheelSpecSheet.pdf
Forward Kinematics

- serial manipulators
  - given the joint variables, find the pose of the end-effector

- mobile robot
  - given the control variables as a *function of time*, find the pose of the robot
    - for the differential drive the control variables are often taken to be the ground velocities of the left and right wheels
      - it is important to note that the wheel velocities are needed as functions of time; a differential drive that moves forward and then turns right ends up in a very different position than one that turns right then moves forward!
Forward Kinematics

- robot with pose \([x \ y \ \theta]^T\) moving with velocity \(V\) in a direction \(\theta\) measured relative the \(x\) axis of \(\{W\}\):
Forward Kinematics

- for a robot starting with pose $[x_0 \ y_0 \ \theta_0]^T$ moving with velocity $V(t)$ in a direction $\theta(t)$:

\[
\begin{align*}
  x(t) &= x_0 + \int_0^t V(t) \cos(\theta(t)) \, dt \\
  y(t) &= y_0 + \int_0^t V(t) \sin(\theta(t)) \, dt \\
  \theta(t) &= \theta_0 + \int_0^t \omega(t) \, dt
\end{align*}
\]
Forward Kinematics

- for differential drive:

\[ x(t) = x_0 + \frac{1}{2} \int_0^t (v_r(t) + v_\ell(t)) \cos(\theta(t)) \, dt \]

\[ y(t) = y_0 + \frac{1}{2} \int_0^t (v_r(t) + v_\ell(t)) \sin(\theta(t)) \, dt \]

\[ \theta(t) = \theta_0 + \frac{1}{\ell} \int_0^t (v_r(t) - v_\ell(t)) \, dt \]
Sensitivity to Wheel Velocity

\[ v_r(t) = 1 + \mathcal{N}(0, \sigma^2) \]
\[ v_\ell(t) = 1 + \mathcal{N}(0, \sigma^2) \]
\[ \theta(0) = 0 \]
\[ t = 0 \ldots 10 \]
\[ \ell = 0.2 \]
Sensitivity to Wheel Velocity

L = 0.2;
sigma = 0.05;
figure hold on
for i = 1:1000
    vR = 1 + normrnd(0, sigma);
    vL = 1 + normrnd(0, sigma);
    theta = 0;
    x = 0;
    y = 0;
    dt = 0.1;
    for t = 0.1:dt:10
        x = x + 0.5 * (vR + vL) * cos(theta) * dt;
        y = y + 0.5 * (vR + vL) * sin(theta) * dt;
        theta = theta + 1 / L * (vR - vL) * dt;
        vR = 1 + normrnd(0, sigma);
        vL = 1 + normrnd(0, sigma);
    end
    plot(x, y, 'b.');
end
Mobile Robot Forward Kinematics
what is the position of the ICC in $\{W\}$?
Forward Kinematics: Differential Drive

\[ ICC = \begin{bmatrix} x - R \sin \theta \\ y + R \cos \theta \end{bmatrix} \]
assuming smooth rolling motion at each point in time the differential drive is moving in a circular path centered on the ICC

thus, for a small interval of time $\delta t$ the change in pose can be computed as a rotation about the ICC
 Forward Kinematics : Differential Drive

- computing the rotation about the ICC
  1. translate so that the ICC moves to the origin of $\{W\}$
  2. rotate about the origin of $\{W\}$
  3. translate back to the original ICC
Forward Kinematics: Differential Drive

- computing the rotation about the ICC
  1. translate so that the ICC moves to the origin of $\{W\}$
  2. rotate about the origin of $\{W\}$
  3. translate back to the original ICC

$$ICC = \begin{bmatrix} x - R \sin \theta \\ y + R \cos \theta \end{bmatrix} = \begin{bmatrix} ICC_x \\ ICC_y \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} ICC_x \\ ICC_y \end{bmatrix} = \begin{bmatrix} x - ICC_x \\ y - ICC_y \end{bmatrix}$$
Forward Kinematics : Differential Drive

- computing the rotation about the ICC
  1. translate so that the ICC moves to the origin of \( \{W\} \)
  2. rotate about the origin of \( \{W\} \)
  3. translate back to the original ICC

- how much rotation over the time interval?
  - angular velocity * elapsed time = \( \omega \Delta t \)

\[
\begin{bmatrix}
\cos(\omega \Delta t) & -\sin(\omega \Delta t) \\
\sin(\omega \Delta t) & \cos(\omega \Delta t)
\end{bmatrix}
\begin{bmatrix}
x - ICC_x \\
y - ICC_y
\end{bmatrix}
\]
Forward Kinematics: Differential Drive

- computing the rotation about the ICC
  1. translate so that the ICC moves to the origin of \( \{W\} \)
  2. rotate about the origin of \( \{W\} \)
  3. translate back to the original ICC

\[
\begin{bmatrix}
x(t + \delta t) \\
y(t + \delta t)
\end{bmatrix} = \begin{bmatrix}
\cos(\omega \delta t) & -\sin(\omega \delta t) \\
\sin(\omega \delta t) & \cos(\omega \delta t)
\end{bmatrix} \begin{bmatrix}
x - ICC_x \\
y - ICC_y
\end{bmatrix} + \begin{bmatrix}
ICC_x \\
ICC_y
\end{bmatrix}
\]
Forward Kinematics : Differential Drive

- what about the orientation $\theta(t + \delta t)$?
- just add the rotation for the time interval

new pose

$$\begin{bmatrix} x(t + \delta t) \\ y(t + \delta t) \\ \theta(t + \delta t) \end{bmatrix} = \begin{bmatrix} \cos(\omega \delta t) & -\sin(\omega \delta t) \\ \sin(\omega \delta t) & \cos(\omega \delta t) \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x - ICC_x \\ y - ICC_y \\ \theta \end{bmatrix} + \begin{bmatrix} ICC_x \\ ICC_y \\ \omega \delta t \end{bmatrix}$$

$\theta(t + \delta t) = \theta + \omega \delta t$

- which can be written as

$$\begin{bmatrix} x(t + \delta t) \\ y(t + \delta t) \\ \theta(t + \delta t) \end{bmatrix} = \begin{bmatrix} \cos(\omega \delta t) & -\sin(\omega \delta t) & 0 \\ \sin(\omega \delta t) & \cos(\omega \delta t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x - ICC_x \\ y - ICC_y \\ \theta \end{bmatrix} + \begin{bmatrix} ICC_x \\ ICC_y \\ \omega \delta t \end{bmatrix}$$
Forward Kinematics: Differential Drive

- The previous equation is valid if $v_L \neq v_R$
  - i.e., if the differential drive is not travelling in a straight line
- If $v_L = v_R = v$ then

$$
\begin{bmatrix}
  x(t + \delta t) \\
  y(t + \delta t) \\
  \theta(t + \delta t)
\end{bmatrix} =
\begin{bmatrix}
  x + v \delta t \cos \theta \\
  y + v \delta t \sin \theta \\
  \theta
\end{bmatrix}
$$
Sensitivity to Wheel Velocity

\[ v_r(t) = 1 + \mathcal{N}(0, \sigma^2) \]
\[ v_\ell(t) = 1 + \mathcal{N}(0, \sigma^2) \]
\[ \theta(0) = 0 \]
\[ t = 0 \ldots 10 \]
\[ \ell = 0.2 \]
Sensitivity to Wheel Velocity

- given the forward kinematics of the differential drive it is easy to write a simulation of the motion
- we need a way to draw random numbers from a normal distribution
- in Matlab
  - `randn(n)` returns an n-by-n matrix containing pseudorandom values drawn from the standard normal distribution
  - see `mvnrnd` for random values from a multivariate normal distribution
Sensitivity to Wheel Velocity

POSE = []; % final pose of robot after each trial
sigma = 0.01; % noise standard deviation
L = 0.2; % distance between wheels
dt = 0.1; % time step
TRIALS = 1000; % number of trials

for trial = 1:TRIALS
    -run each trial-
    see next slide
end
Sensitivity to Wheel Velocity

\[ vr = 1; \quad \text{% initial right-wheel velocity} \]

\[ vl = 1; \quad \text{% initial left-wheel velocity} \]

\[ \text{pose} = [0; 0; 0]; \quad \text{% initial pose of robot} \]

\[ \text{for } t = 0:dt:10 \]

\[-\text{move the robot one time step -} \]

\[-\text{see next slide} \]

\[ \text{end} \]

\[ \text{POSE} = [\text{POSE pose}]; \quad \text{% record final pose after trial t} \]
Sensitivity to Wheel Velocity

\[
\theta = \text{pose}(3);
\]

\[
\text{if } v_r == v_l \\
\begin{align*}
\text{pose} &= \text{pose} + \begin{bmatrix}
v_r \cdot \cos(\theta) \cdot dt; \\
v_r \cdot \sin(\theta) \cdot dt; \\
0;
\end{bmatrix}; \\
\end{align*}
\]

\[
\text{else} \\
\begin{align*}
\omega &= (v_r - v_l) / L; \\
R &= (L / 2) \cdot (v_r + v_l) / (v_r - v_l); \\
\text{ICC} &= \text{pose} + \begin{bmatrix}
-R \cdot \sin(\theta); \\
R \cdot \cos(\theta); \\
0;
\end{bmatrix}; \\
\text{pose} &= \text{rz}(\omega \cdot dt) \cdot (\text{pose} - \text{ICC}) + \text{ICC} + \\
&\hspace{1cm} \begin{bmatrix}
0; 0; \omega \cdot dt;
\end{bmatrix}; \\
\end{align*}
\]

\[
\text{end}
\]

\[
v_r = 1 + \sigma \cdot \text{randn}(1); \\
v_l = 1 + \sigma \cdot \text{randn}(1);
\]