CSCE274 Robotic Applications and Design
Fall 2021
Wheeled Locomotion

Ioannis REKLEITIS, Ibrahim SALMAN

Computer Science and Engineering

University of South Carolina

yiannisr@cse.sc.edu

Locomotion

- **Locomotion** refers to the way a robot moves through their environments

- It can be classified according to the environment:
  – Terrestrial
  – Airborne
  – Aquatic

Locomotion objective

Convert desire to move A->B into an actual motion

Locomotion objective

- Different aspects to consider:
  - How to *mechanically design* the robots, i.e., arrange actuators and effectors
  - Model the relationship
    actuator output ←→ Incremental motion

    known as **kinematics**

Terrestrial locomotion

As the focus of the class is going to be on ground mobile robots, let's see some mechanical design examples

Single wheel

- Pros:
  - Space compact
  - Energy efficient
- Cons:
  - Inherently unstable

Seiko-chan

Source: murata.com

Two wheels

- Pros:
  - Greater stability
  - Greater maneuverability
- Cons:
  - Still unstable

Source: smu.edu/~dpa-www

Except!

Modeling, simulation and control of an electric diwheel
Cazzolato, et al. ACRA 2011.

Two wheels + caster wheel

- Pros:
  - Statically stable
  - Simple structure
- Cons:
  - Difficulty of moving on irregular surfaces

Pioneer 2DX

Source: cse.unsw.edu.au

Four wheels

- Pros:
  - Stable during high-speed motion
- Cons:
  - More complicated steering mechanism

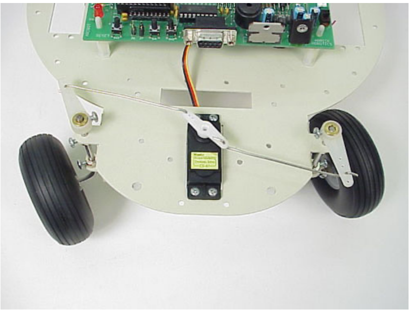
Google self-driving car

Source: google.com

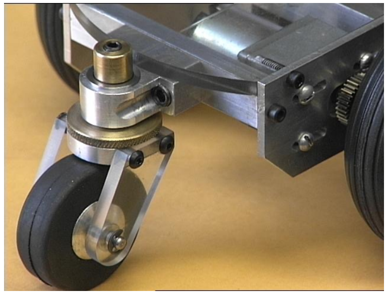
Types of wheels

- Wheels can be classified according to their function:
  - Drive wheels
  - Steered wheels
  - Passive wheels

Source: parallax.com



Source: arrickrobotics.com



Source: smu.edu/dpa-www

Types of wheels

- Wheels can be classified according to their design:
  - Standard wheel
  - Orientable wheel
  - Ball wheel
  - Omni wheel

Standard Wheel

Center Orientable Wheel   Off- Centered Orientable Wheels

Caster Ball Wheel

Omni Wheels

Mecanum Wheel

Source: robotplatform.com

Types of control mechanism

- Differential drive
- Synchronous drive
- Ackerman steering
- Skid steering
- …

Design tradeoffs

- Maneuverability
- Controllability
- Traction
- Climbing ability
- Stability
- Efficiency
- Maintenance
- Cost

Kinematics

- Once the mechanical design is fixed, it is necessary to model the correspondence between actuator motion and the resulting effector motion
  - **Forward kinematics**: given some controls, where the robot is going to be?
  - **Inverse kinematics**: given a robot pose, what control should be given to the robot?

Model

- The following should be considered
  - How should the robot's state be represented?
  - What are the inputs for the system?
  - How does the state change in response to the input?

Robot’s state

- It is fundamental to encode a complete description of the robot’s *situation* in the *environment*

- The information contained should at least include everything that the robot needs to know about the past to predict the future

Robot's state

- **State**, usually denoted as x, is fundamental to encode a complete description of the robot's *situation* in the *environment*

- The information contained should at least include everything that the robot needs to know about the past to predict the future

- The set of all possible states is called the robot's **state space**, usually denoted as X

Action

- The current state can be changed by executing **actions** (inputs or controls), denoted as u

- The set of all possible actions is called the **action space**, denoted as U

Transition function

- The function that determines a new state given the current state and an action is called **transition function** (forward kinematics)

$$x_{t+1} = f(x_t, u_t)$$

$$f : X \times U \rightarrow X$$

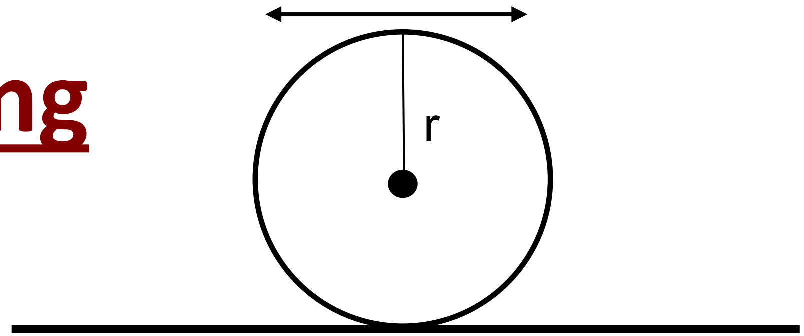
Inverse kinematics

- Given a starting state and a desired state find the action that moves the robot from x to x’

$$u_t = f(x_t, x_{t+1})$$

$$f : X \times X \rightarrow U$$

Example of single rolling wheel



- Given a single rolling wheel with radius r
  - State: position of the robot $x \in \mathbb{R}$
  - Action: rotation angle $u \in \mathbb{R}$
  - Transition Function $f(x_t, u_t) = x_t + u_t * r = x_{t+1}$
  - Inverse kinematics $g(x_t, x_{t+1}) = \frac{x_t - x_{t+1}}{r} = u_t$

Models

- It is important to define a mathematical model to describe the forward and the inverse kinematics

- Any model usually *approximates* how the system works

Limitations of models

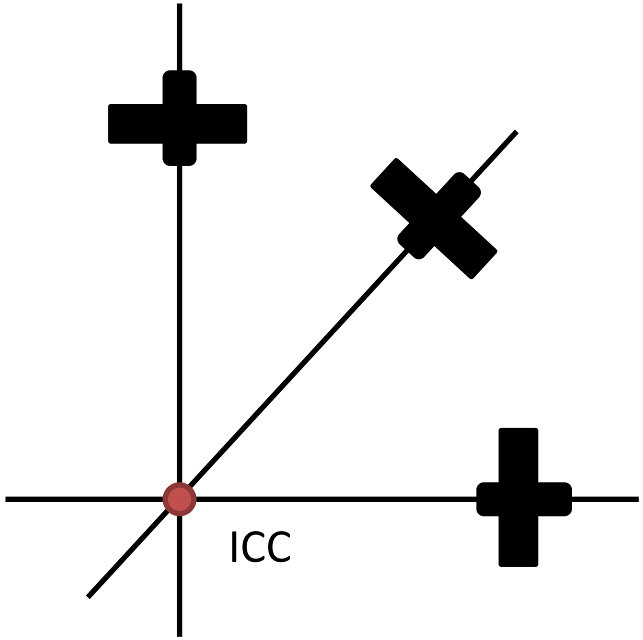
- E.g., with wheels
  - Slipping – rotating without moving along the ground
  - Sliding – moving along the ground without rotating
  - Lateral motion
  - …

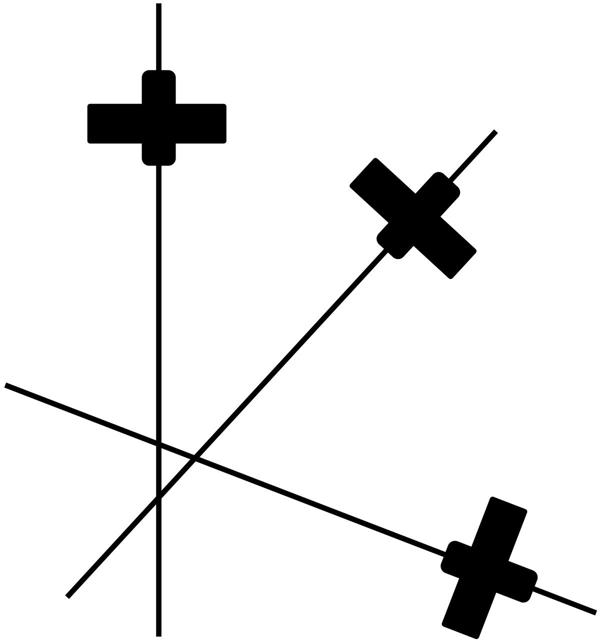
Instantaneous center of rotation

- With multiple wheels the following concept should be considered for modeling the locomotion

- The **instantaneous center of rotation** or **instantaneous center of curvature** (ICR or ICC) is the point around which each wheel moves in a circular motion

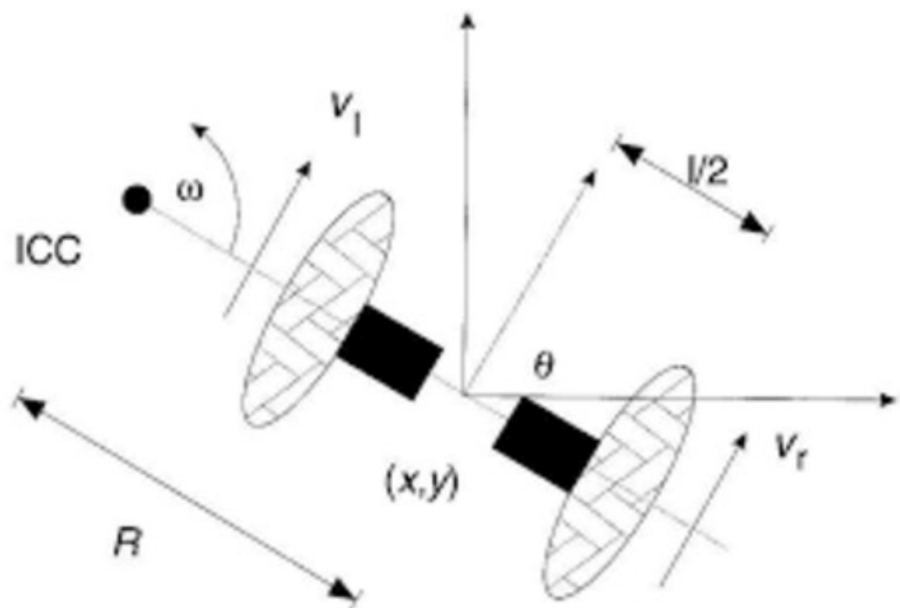
Examples





Differential drive

- Two active fixed wheels



Source: berkeley.edu

Differential drive

- Given a differential drive robot
  - State: position of the robot in the space $x, y, \theta \in \mathbb{R}$
  - Action: left and right linear velocities (e.g., m/s) and time interval $v_l, v_r, \Delta t \in \mathbb{R}$
    - In case the velocities are given in rad/s, it is sufficient to multiply by the velocity by the wheel radius

Differential drive model

- The rate of rotation ω around ICC must be the same for both wheels

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

where

– R is the distance from ICC to center of the axis between the wheels

– l is the distance between wheels

Differential drive model

- Then, it is possible to find R and ω at any time

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

$$\omega = \frac{v_r - v_l}{l}$$

Transition function

- Given the current state (x, y, θ) and executing action $(v_l, v_r, \Delta t)$
  - ICC location can be found using the equation for R

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

- Using some vector arithmetic we can compute the new state (x', y', θ')

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

Inverse kinematics

- Because of the non-holonomic constraint, an arbitrary target pose cannot be specified and used to find the velocities
- One solution, probably the easiest but not necessarily the most efficient, is
  1. Rotate in place until the robot is facing its destination
  2. Move forward to the destination
  3. Rotate to the correct orientation

  namely a *rotation-translation-rotation* motion

Special cases

- In particular, there are two important special cases
  - If $v_l = v_r$, then R is infinite. The robot moves in a straight line
  - If $v_l = -v_r$, then $R = 0$. The robot rotates in place
- In the general case, the robot moves along a circular arc.

Inverse kinematics

- Translation

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} x + v\,cos(\theta)\delta t \\ y + v\,sin(\theta)\delta t \\ \theta \end{bmatrix}$$

- Rotation

$$\begin{bmatrix} x' \\ y' \\ \theta' \end{bmatrix} = \begin{bmatrix} x \\ y \\ \theta + 2v\delta t/l \end{bmatrix}$$

- Use them to find the time, e.g., specifying the velocity

Example of inverse kinematics

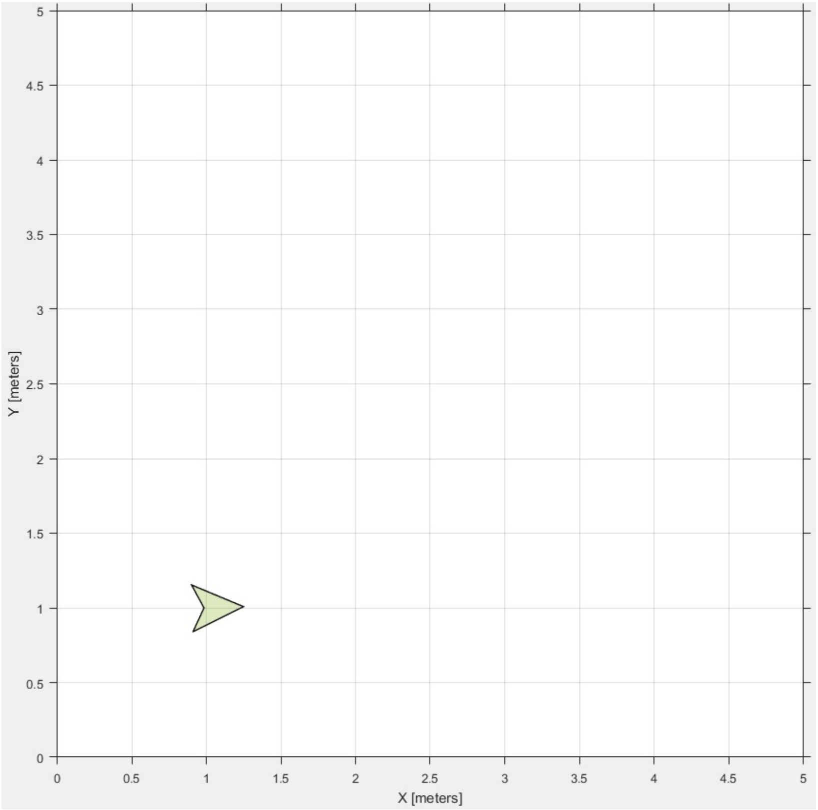
- Given the following differential drive robot with the following characteristics

$$l = 50\text{cm}$$
$$v_l \in [-1\frac{\text{m}}{\text{s}}, 1\frac{\text{m}}{\text{s}}]$$
$$v_r \in [-1\frac{\text{m}}{\text{s}}, 1\frac{\text{m}}{\text{s}}]$$

- and the start state and target state

$$\mathbf{x}_s = (1\text{m}, 1\text{m}, 0)$$
$$\mathbf{x}_t = (1\text{m}, 4\text{m}, 0)$$

- Find the control for the robot following the rotation-translation-rotation motion



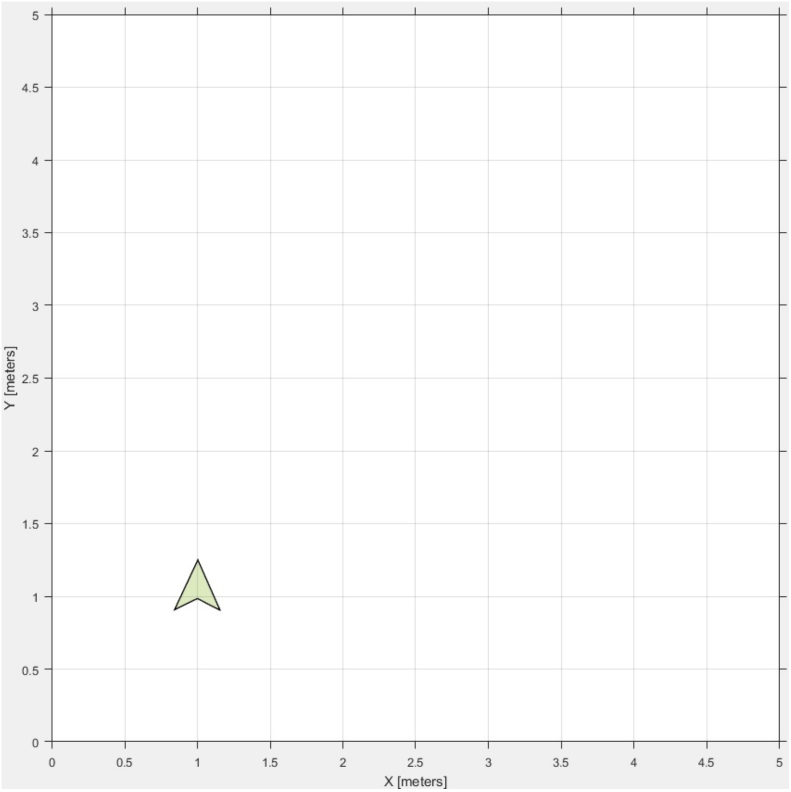
Example of inverse kinematics

- First the robot match the orientation of the target
- For a counterclockwise rotation in place

  $v_l = -v_r$ with $v_l < 0$
- Given that $\omega = \frac{v_r - v_l}{l}$

  and $\theta' = \omega \Delta t$
- By setting the velocity to the maximum, we can find the time necessary to rotate

$$v_l = -1$$
$$v_r = 1$$
$$\Delta t = 0.125\pi$$

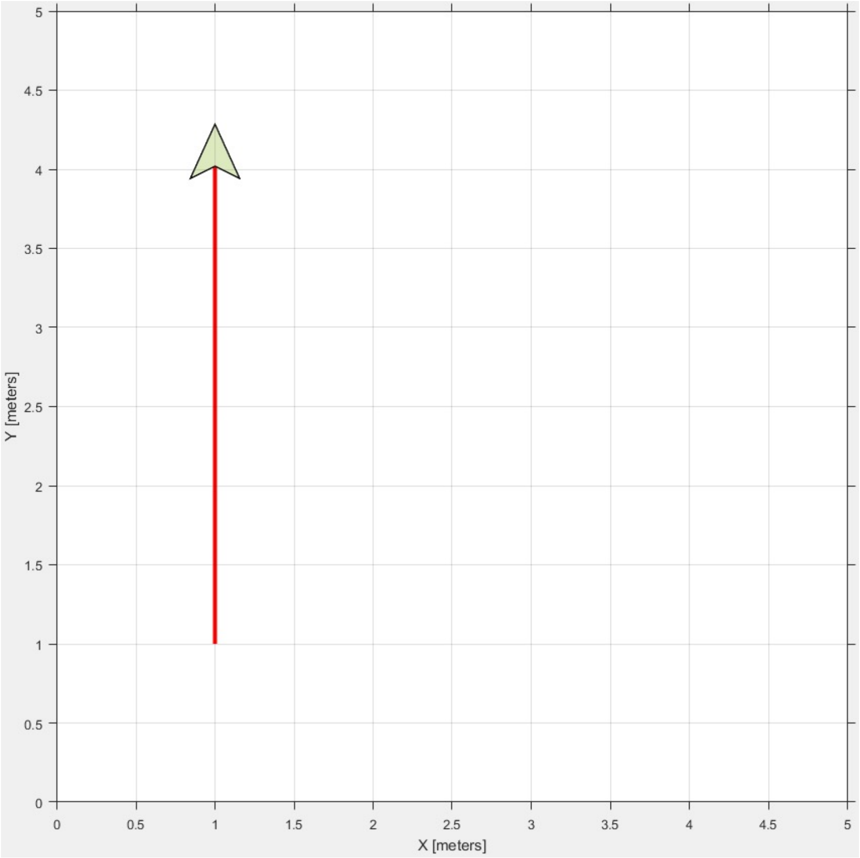


Example of inverse kinematics

- Then translate the robot

$$v_l = v_r \qquad v_l > 0$$

$$v_l = 1$$
$$v_r = 1$$
$$\Delta t = 3$$



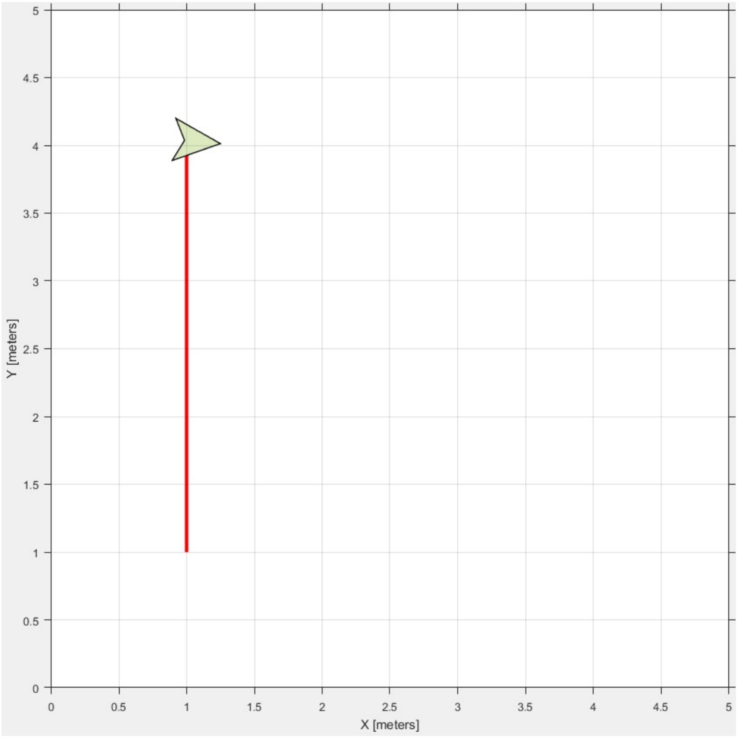
Example of inverse kinematics

- Finally rotate the robot to match the orientation

$$v_l = -v_r \quad v_l > 0$$

- Following the same reasoning as the first rotation, the solution is

$$v_l = 1$$
$$v_r = -1$$
$$\Delta t = 0.125\pi$$

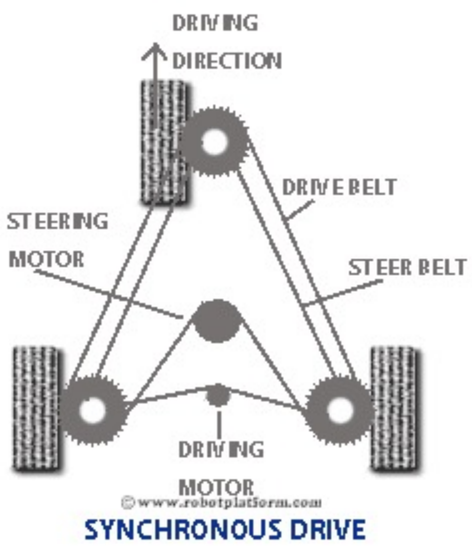


Differential drive issues

- Matching of drive mechanisms
  - Tire wear (the radius of the wheel is wrong)
  - Motors (the velocity is wrong)
  - Ground traction (specified rotation is not actual motion)
  - …
- Balance – Castor (caster) wheel

Synchronous drive

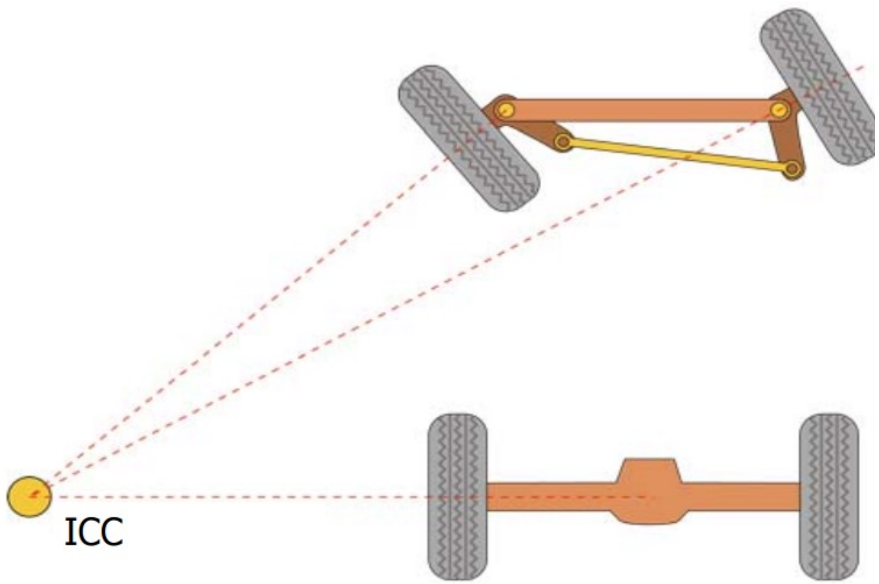
- A **synchronous drive** robot has usually three wheels
  - All the wheels always point in the same direction
  - All the wheels always drive at the same speed



Source: robotplatform.com

Ackerman

- If only some of the wheels are steerable, the robot has some direct control over where the ICC is



Source: Wikipedia.org