Kinematics of Mobile Robots

Corina Barbalata

Mechanical and Industrial Engineering

iCORE Laboratory

cbarbalata@lsu.edu

Dynamics vs Kinematics

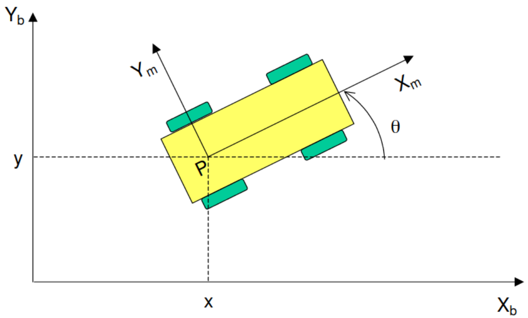
- Dynamics - the study of motion in which forces are modeled
  - includes the energies and speeds associated with these motions
- Kinematics - the study of the mathematics of motion without considering the forces that affect the motion.
  - deals with the geometric relationships that govern the system.
  - deals with the relationship between control parameters and the behavior of a system in state space.

Holonomic vs Non-holonomic

- A system is holonomic if the number of controllable degrees of freedom is equal to the total degrees of freedom.

Example: A holonomic robot can drive straight to a goal that is not in-line with its orientation whereas a non-holonomic robot must either rotate to the desired orientation before moving forward or rotate as it moves.

Notations for a 3DOF mobile robot



$$\{X_m, Y_m\} - \text{moving frame} \qquad \{X_b, Y_b\} - \text{base frame}$$

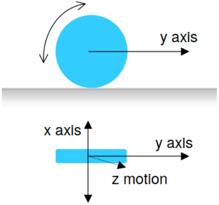
$$q = [x, y, \theta]^T - \text{robot pose in base frame}$$

$$R(\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

R – orientation of the base frame with respect to the moving frame

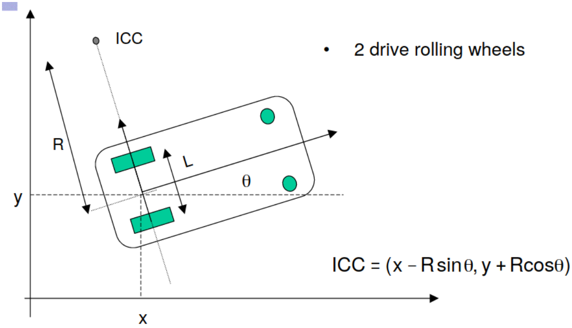
Wheeled Mobile Robots (WMR)

- Idealizing Rolling Wheels



- If the wheel is free to rotate about its axis (x axis), the robot exhibits preferential rolling motion in one direction (y axis) and a certain amount of lateral slip.
- For low velocities, rolling is a reasonable wheel model.
    - This is the model that will be considered in the kinematics models of WMR.

Differential Drive



ICC – instantenous center of curvature

v_r – linear velocity of right wheel (control variable)

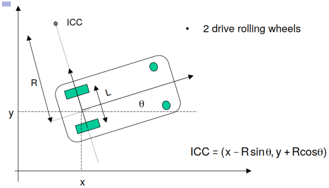
v_l – linear velocity of left wheel (control variable)

r – nominal radius of each wheel

R – instantaneous curvature radius of the robot trajectory

ω – rate of rotation about the ICC (same for both wheels)

Given the left wheel linear velocity and the right wheel linear velocity, compute the rate of rotation about ICC and the instantaneous curvature.



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

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

$$\omega(R - \frac{L}{2}) = v_l$$

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

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

Differential Drive

Forward Kinematic model in the robot frame:

$$\begin{bmatrix} v_x(t) \\ v_y(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} R/2 & R/2 \\ 0 & 0 \\ -R/L & R/L \end{bmatrix} \begin{bmatrix} w_l(t) \\ w_r(t) \end{bmatrix}$$

- $w_r(t)$ - angular velocity of right wheel
- $w_l(t)$ - angular velocity of left wheel

Forward Kinematic model in the world frame:

$$\begin{cases} v(t) = \omega(t)R = \frac{1}{2}(v_r(t) + v_l(t)) \\ \omega(t) = \frac{v_r(t) - v_l(t)}{L} \end{cases} \Rightarrow \begin{cases} \dot{x}(t) = v(t)\cos(\theta(t)) \\ \dot{y}(t) = v(t)\sin(\theta(t)) \\ \dot{\theta}(t) = \omega(t) \end{cases} \Rightarrow$$

$$\Rightarrow \begin{cases} x(t) = \int_0^t v(\tau)\cos(\theta(\tau))d\tau \\ y(t) = \int_0^t v(\tau)\sin(\theta(\tau))d\tau \\ \theta(t) = \int_0^t \omega(\tau)d\tau \end{cases}$$

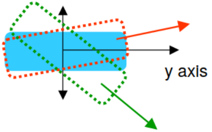
$$\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} \cos(\theta(t)) & 0 \\ \sin(\theta(t)) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v(t) \\ w(t) \end{bmatrix}$$

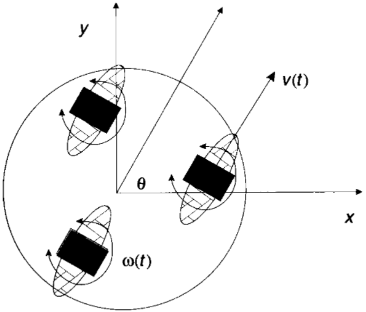
Synchronous drive

- In a synchronous drive robot (synchro drive) each wheel is capable of being driven and steered.
- Typical configurations:
  - Three steered wheels arranged as vertices of an equilateral triangle often surmounted by a cylindrical platform
  - All the wheels turn and drive in unison
  - This leads to a holonomic behavior

Synchronous drive

Steered wheel - the orientation of the rotation axis can be controlled.



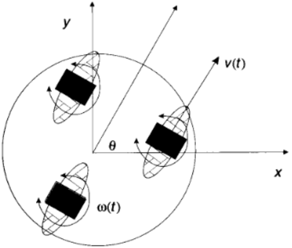


Synchronous drive

- All the wheels turn in unison
- All of the three wheels point in the same direction and turn at the same rate.
- The vehicle controls the direction in which the wheels point and the rate at which they roll
- Because all the wheels remain parallel the synchro drive always rotate about the center of the robot
- The synchro drive robot has the ability to control the orientation θ of their pose directly.

Synchronous drive

Control variables (independent): $v(t),\ w(t)$

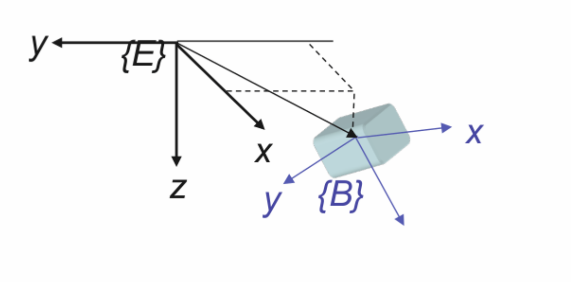


$$\begin{cases} x(t) = \int_0^t v(\tau)\cos(\theta(\tau))d\tau \\ y(t) = \int_0^t v(\tau)\sin(\theta(\tau))d\tau \\ \theta(t) = \int_0^t w(\tau)d\tau \end{cases}$$

- The ICC is always at infinity
- Changing the orientation of the wheels manipulates the direction of ICC

6DOF Mobile Robot

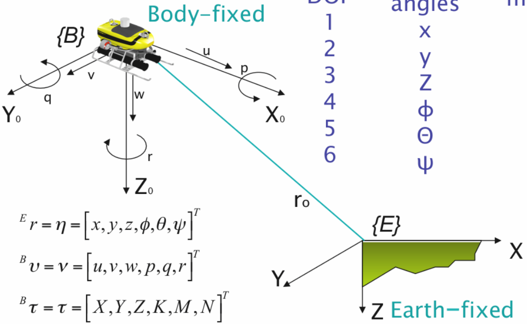
- Either Autonomous Underwater Vehicle (AUV) or Unmanned Aerial Vehicle (UAV)



- Kinematics using Homogeneous Matrix

$$T_B^E = \begin{bmatrix} n_x & o_x & a_x & x \\ n_y & o_y & a_y & y \\ n_z & o_z & a_z & z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

– **Remember**



$$^{E}r = \eta = \left[x, y, z, \phi, \theta, \psi\right]^{T}$$

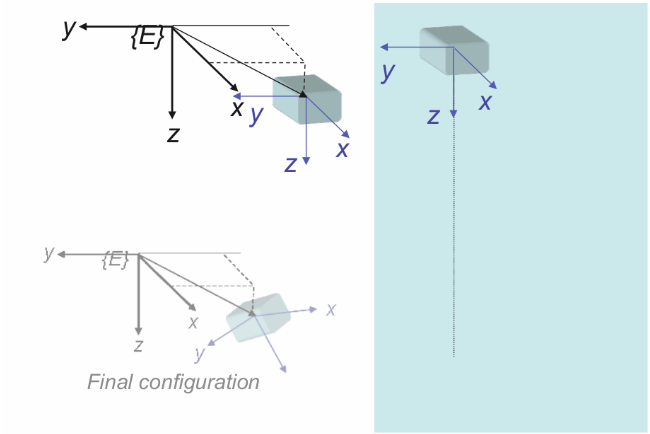
$$^{B}\upsilon = \nu = \left[u, v, w, p, q, r\right]^{T}$$

$$^{B}\tau = \tau = \left[X, Y, Z, K, M, N\right]^{T}$$

DOF	Position and Euler angles	Forces and moments	Lin. and ang. Vel.
1	x	X	u (surge)
2	y	Y	v (sway)
3	z	Z	w (heave)
4	ϕ	K	p (roll)
5	Θ	M	q (pitch)
6	ψ	N	r (yaw)

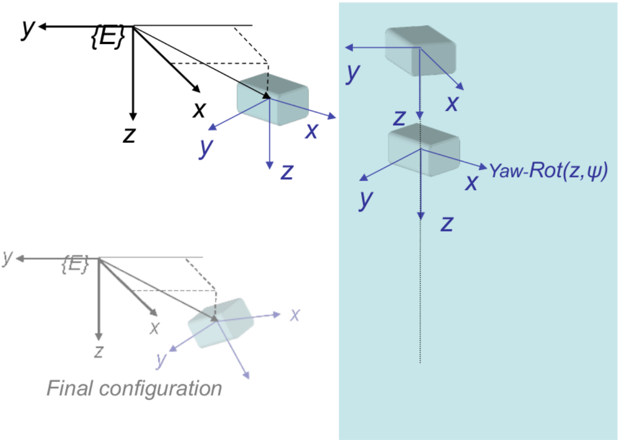
η relative to inertial frame

υ, τ relative to fixed-body frame

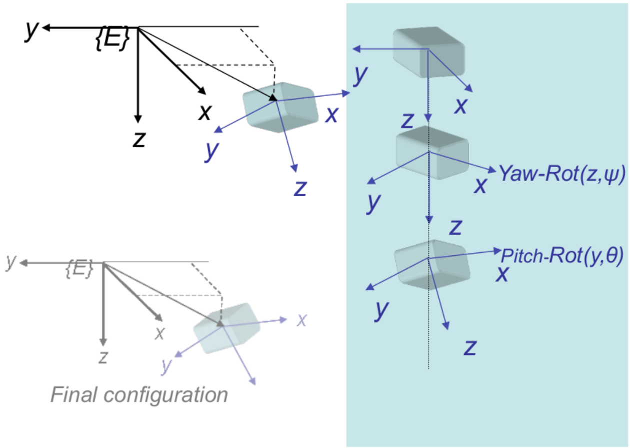


y
{E}
x
z
y
x
z
y
{E}
x
z
y
x
Final configuration
y
x
z

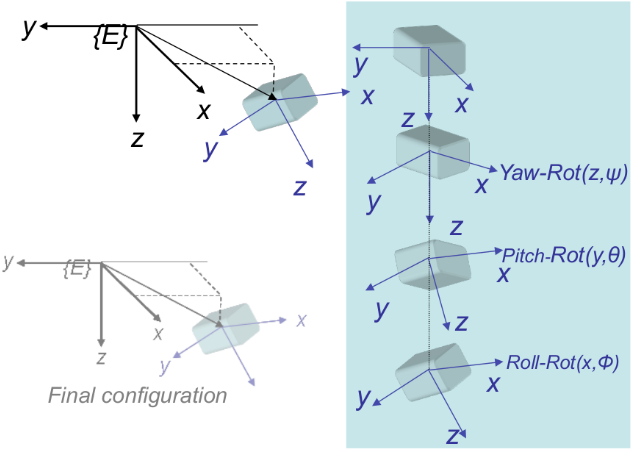
6DOF Mobile Robot



6DOF Mobile Robot



6DOF Mobile Robot



Final configuration

6DOF Mobile Robot

The robot's position is defined with the following notations

- UV Position & Altitude

$$r^E = (x, y, z, \phi, \theta, \psi)^T$$

- Homogeneous Matrix:

$$T_B^E = Trans(x, y, z)Rot(\psi, z)Rot(\theta, y)Rot(\phi, x)$$

$$T_B^E = \begin{bmatrix} . & . & . & x \\ . & R_B^E & . & y \\ . & . & . & z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad R_B^E = Rot(\psi, z)Rot(\theta, y)Rot(\phi, x)$$

$$R_B^E = \begin{bmatrix} \cos(\psi)\cos(\theta) & -\sin(\psi)\cos(\phi) + \cos(\psi)\sin(\theta)\sin(\phi) & \sin(\psi)\sin(\phi) + \cos(\psi)\cos(\phi)\sin(\theta) \\ \sin(\psi)\cos(\theta) & \cos(\psi)\cos(\phi) + \sin(\phi)\sin(\theta)\sin(\psi) & \cos(\psi)\sin(\phi) + \sin(\psi)\cos(\phi)\sin(\theta) \\ -sin(\theta) & \cos(\theta)\sin(\psi) & \cos(\theta)\cos(\psi) \end{bmatrix}$$

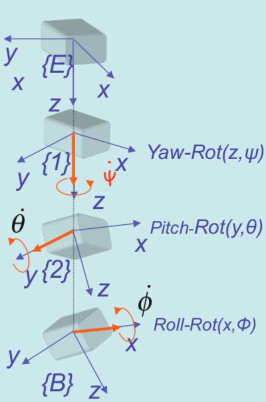
6DOF Mobile Robot Kinematics

- Linear Velocity:

$$(\dot{x}, \dot{y}, \dot{z}) = R_B^E (u, v, w)^T \Rightarrow \dot{\eta}_1 = R_B^E \nu$$

- Angular Velocity:

$$\omega = (\dot{\phi}, 0, 0)^T + R_2^B (0, \dot{\theta}, 0)^T + R_1^B (0, 0, \dot{\psi}) \Rightarrow \dot{\eta}_2 = J(\eta_2)^{-1} \omega$$

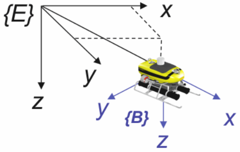


$$J(\eta_2)^{-1} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \frac{\sin(\phi)}{\cos(\theta)} & \frac{\cos(\phi)}{\cos(\theta)} \end{bmatrix}$$

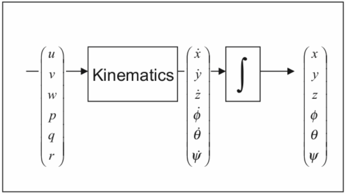
- Relation between Earth coordinates and Base (Vehicle) coordinates
  - Overall transformation:

$$\Rightarrow \begin{bmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{bmatrix} = \begin{bmatrix} R_B^E & 0_{3\times 3} \\ 0_{3\times 3} & J(\eta_2)^{-1} \end{bmatrix} \begin{bmatrix} \nu \\ \omega \end{bmatrix}$$

- **Kinematics Model of an Underwater Robot**



Kinematics *Simulation*



$$ {}^B\nu \longrightarrow \boxed{\text{Kinematics}} \longrightarrow {}^E\dot{\eta} $$

$$ {}^B\nu \longleftarrow \boxed{\text{Inverse Kinematics}} \longleftarrow {}^E\dot{\eta} $$

$$ {}^E\begin{pmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{pmatrix} = \begin{pmatrix} {}^E R_B & 0_{3x3} \\ 0_{3x3} & J(\eta_2)^{-1} \end{pmatrix} \cdot {}^B\begin{pmatrix} v \\ \omega \end{pmatrix} $$

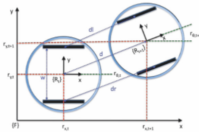
Direct Kinematics

$$ {}^B\begin{pmatrix} v \\ \omega \end{pmatrix} = \begin{pmatrix} {}^B R_E & 0_{3x3} \\ 0_{3x3} & J(\eta_2) \end{pmatrix} \cdot {}^E\begin{pmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{pmatrix} $$

Inverse Kinematics

Dead reckoning is the process of calculating current position of some moving object by using a previously determined position, or fix, by using estimations of speed, heading direction and course over elapsed time.

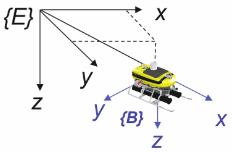
Dead Reckoning



```
[x_k y_k θ_k] function getOdometry()
    ΔN_L=ReadEncoder(LEFT);
    ΔN_R=ReadEncoder(RIGHT);

    dl = 2 * π * R * ((ΔN_L)/ζ) ;
    dr = 2 * π * R * ((ΔN_R)/ζ) ;
    d =(dr+dl)/2;

    Δθ_k=(dr-dl)/w;
    θ_k= θ_{k-1}+ Δθ_k;
    x_k= x_{k-1} + (d * cos(θ_k));
    y_k= y_{k-1} + (d * sin(θ_k));
Return [x_k y_k θ_k]
```



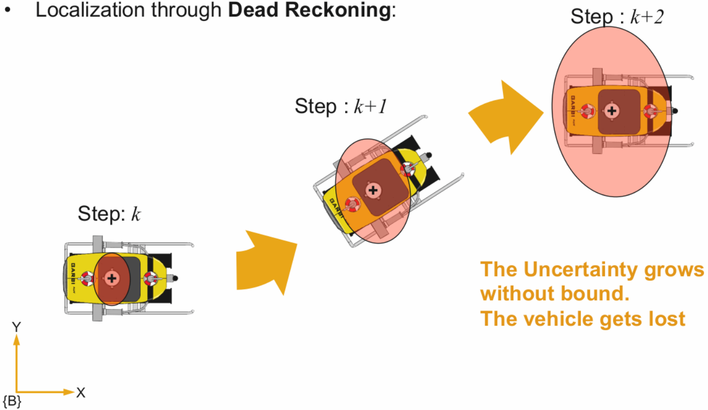
$$^{B}\begin{pmatrix} v \\ \omega \end{pmatrix} = get\_measurements()$$

$$^{E}\begin{pmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{pmatrix} = \begin{pmatrix} ^{E}R_B & 0_{3x3} \\ 0_{3x3} & J(\eta_2)^{-1} \end{pmatrix} \cdot {}^{B}\begin{pmatrix} v \\ \omega \end{pmatrix}$$

$$^{E}\begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix} = \int {}^{E}\begin{pmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{pmatrix} dt$$

Dead Reckoning

- Localization through **Dead Reckoning**:



Questions?