

# Lecture 20, Mar 2, 2026

## Dead Reckoning and Odometry

- Given a vehicle model  $\dot{\mathbf{x}} = \mathbf{A}(\mathbf{x})\mathbf{x} + \mathbf{B}(\mathbf{x})\mathbf{u}$ , we can integrate to predict future states as  $\mathbf{x}(t) = \mathbf{x}(0) + \int_0^t \dot{\mathbf{x}}(t) dt$ , given initial condition and inputs  $\mathbf{u}(t)$
- Consider a 1D example  $\dot{x} = u + w$  where  $w$  is some noise, and we integrate as  $x(t+h) = x(t) + hu(t) + hw(t) \implies x(Kh) = x(0) + h \sum_{k=0}^{K-1} u(kh) + h \sum_{k=0}^{K-1} w(kh)$ 
  - Assuming the noise is a zero mean Gaussian we can derive the mean and variance
  - $\mu(Kh) = E[x(0)] + h \sum_{k=0}^{K-1} u(kh)$  – this means that the mean of the estimate is unbiased
  - $\sigma(Kh) = Kh^2Q$  – the variance grows without bound
  - The fact that the uncertainty grows without bound is the key characteristic of dead reckoning
- We can perform dead reckoning with wheel odometry from encoders
- For the differential drive robot:

$$-\dot{\mathbf{q}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} r/2 & r/2 \\ r/(2b) & -r/(2b) \end{bmatrix} \begin{bmatrix} \dot{\varphi}_r \\ \dot{\varphi}_l \end{bmatrix}$$

- Assume we can measure the wheel speed using encoders

$$-\text{We discretize as } \mathbf{q}(t+h) = \mathbf{q}(t) \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} r/2 & r/2 \\ r/(2b) & -r/(2b) \end{bmatrix} h \begin{bmatrix} \dot{\varphi}_r \\ \dot{\varphi}_l \end{bmatrix}$$

$$-\text{Model the encoder measurements as } h \begin{bmatrix} \dot{\varphi}_r(t) \\ \dot{\varphi}_l(t) \end{bmatrix} \approx \begin{bmatrix} \Delta\varphi_r(t) + w_r(t) \\ \Delta\varphi_l(t) + w_l(t) \end{bmatrix}$$

- \* We model  $w(t) \sim \mathcal{U}\left(-\frac{\pi}{2^N}, \frac{\pi}{2^N}\right)$  for quantization error

- \* For convenience we approximate as a zero-mean Gaussian  $\mathbf{w}(t) = \begin{bmatrix} w_r(t) \\ w_l(t) \end{bmatrix} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$

$$-\mathbf{q}(t+h) = \mathbf{q}(t) \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} r/2 & r/2 \\ r/(2b) & -r/(2b) \end{bmatrix} \begin{bmatrix} \Delta\varphi_r(t) + w_r(t) \\ \Delta\varphi_l(t) + w_l(t) \end{bmatrix}$$

- We often denote  $\mathbf{q}(t)$  as the overall noisy state, which is composed of  $\bar{\mathbf{q}}(t)$ , the large and known mean, and  $\delta\mathbf{q}(t)$ , the small uncertain part (noise), so  $\mathbf{q}$  has mean  $\bar{\mathbf{q}}$  and covariance  $E[\delta\mathbf{q}(t)\delta\mathbf{q}^T(t)]$

- Linearize the update equation about the mean to propagate uncertainty

$$*\cos(\theta(t)) \approx \cos(\bar{\theta}(t)) - \sin(\bar{\theta}(t))\delta\theta(t)$$

$$*\sin(\theta(t)) \approx \sin(\bar{\theta}(t)) + \cos(\bar{\theta}(t))\delta\theta(t)$$

$$-\mathbf{q}(t+h) = \mathbf{q}(t) + \left( \begin{bmatrix} \cos \bar{\theta}(t) & 0 \\ \sin \bar{\theta}(t) & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} -\sin \bar{\theta}(t) & 0 \\ \cos \bar{\theta}(t) & 0 \\ 0 & 0 \end{bmatrix} \delta\theta(t) \right) \begin{bmatrix} r/2 & r/2 \\ r/(2b) & -r/(2b) \end{bmatrix} \begin{bmatrix} \Delta\varphi_r(t) + w_r(t) \\ \Delta\varphi_l(t) + w_l(t) \end{bmatrix}$$

- \* We drop all second order terms in our approximation to make it linear

$$*\text{The mean propagates as } \bar{\mathbf{q}}(t+h) = \bar{\mathbf{q}}(t) + \begin{bmatrix} \cos \bar{\theta}(t) & 0 \\ \sin \bar{\theta}(t) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} r/2 & r/2 \\ r/(2b) & -r/(2b) \end{bmatrix} \begin{bmatrix} \Delta\varphi_r(t) \\ \Delta\varphi_l(t) \end{bmatrix}$$

$$*\delta\mathbf{q}(t+h) = \delta\mathbf{q}(t) + \begin{bmatrix} -\sin \bar{\theta}(t) & 0 \\ \cos \bar{\theta}(t) & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} r/2 & r/2 \\ r/(2b) & -r/(2b) \end{bmatrix} \begin{bmatrix} \Delta\varphi_r(t) \\ \Delta\varphi_l(t) \end{bmatrix} \delta\theta(t)$$

$$+ \begin{bmatrix} \cos \bar{\theta}(t) & 0 \\ \sin \bar{\theta}(t) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} r/2 & r/2 \\ r/(2b) & -r/(2b) \end{bmatrix} \begin{bmatrix} w_r(t) \\ w_l(t) \end{bmatrix}$$

$$= \mathbf{A}(t)\delta\mathbf{q}(t) + \mathbf{B}(t)\mathbf{w}(t)$$

- \* The uncertainty propagates as  $\Sigma(t+h) = \mathbf{A}(t)\Sigma(t)\mathbf{A}(t)^T + \mathbf{B}(t)\mathbf{Q}\mathbf{B}(t)^T$ 
  - Once again the uncertainty grows without bound
- Our wheel odometry model depends on the wheel radius, baseline, and the update covariance; good calibration of these parameters is crucial for minimizing error
  - For wheel radius, we drive forward some distance in a straight line and divide the distance travelled by the average of the wheel encoder angle measurements
    - \*  $r = 2 \frac{x_{\text{true}}}{\sum_k (\Delta\varphi_r(kh) + \Delta\varphi_l(kh))}$
  - For wheel separation, we rotate some number of rotations (we can measure this using e.g. a laser pointer on the robot), and solve for the wheel separation using the distance travelled calculated from the wheel radius
    - \*  $b = \frac{r}{2} \frac{\sum_k (\Delta\varphi_r(kh) + \Delta\varphi_l(kh))}{2\pi N}$
- Wheel odometry is usually used as the predictive source for sensor fusion, since it uses cheap sensors and low amount of computation, and does not drift when stationary unlike IMU measurements