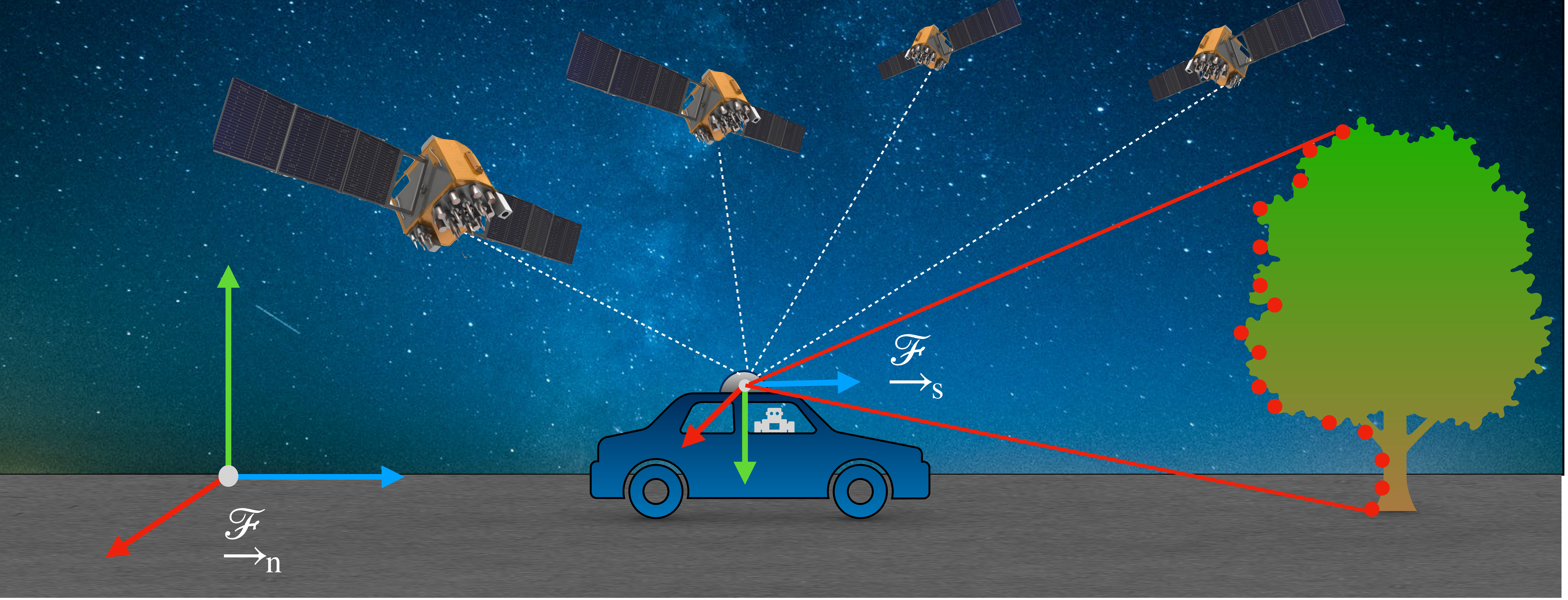
Module 5 | Lesson 2

EKF FOR VEHICULAR STATE

Multisensor Fusion for State Estimation



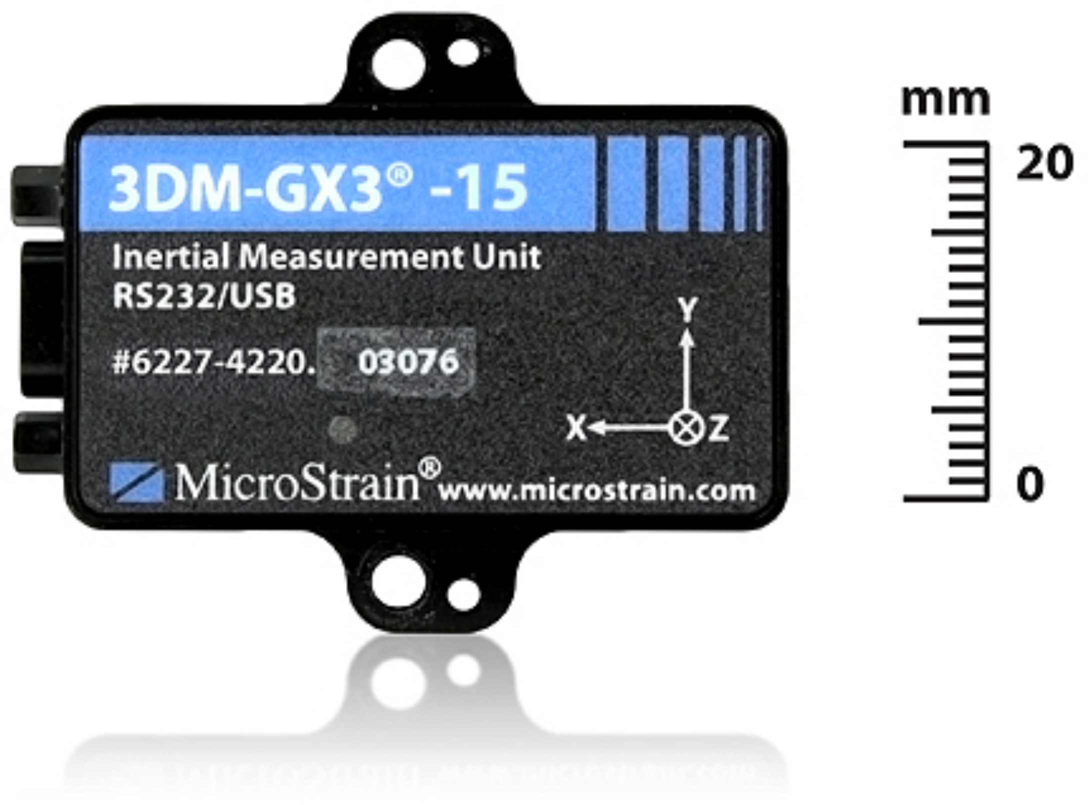
By the end of this video, you will be able to…

- Develop an error state extended Kalman Filter for estimating position, velocity and orientation using an IMU, GNSS sensor, and LIDAR.

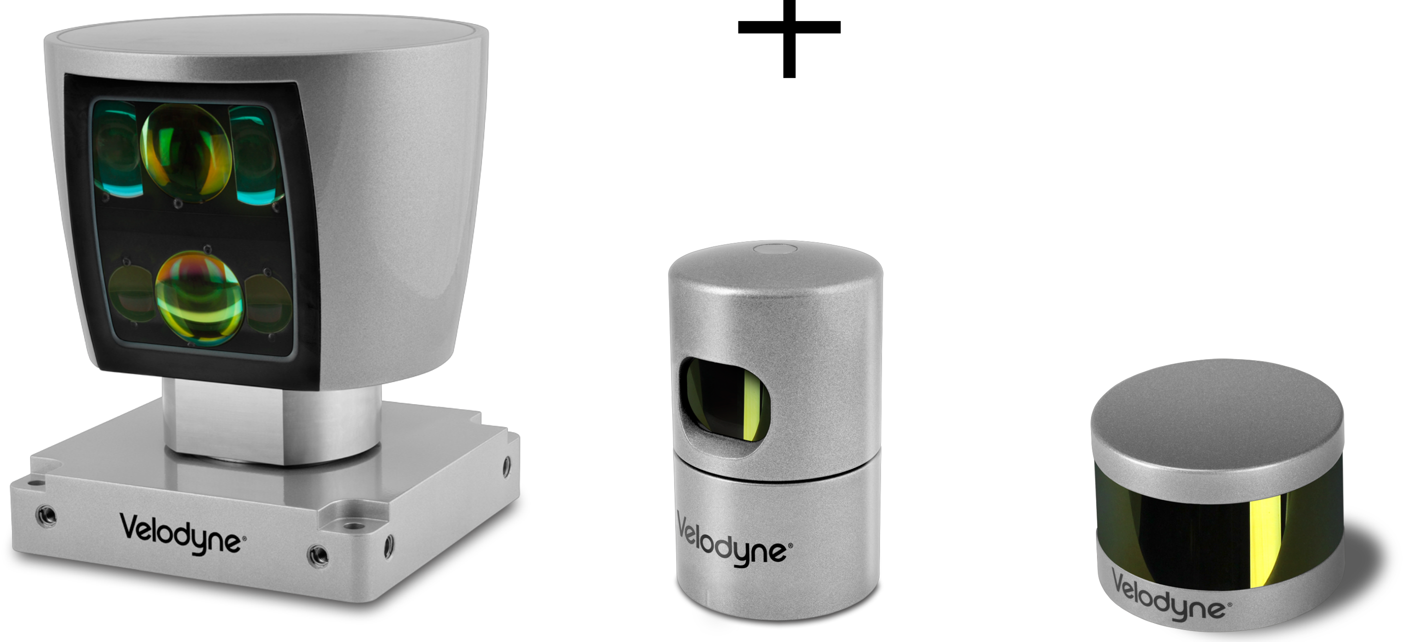
Why use GNSS with IMU & LIDAR?

- Error dynamics are completely different and uncorrelated
- IMU provides ‘smoothing’ of GNSS, fill-in during outages due to jamming or maneuvering
    - *Wheel odometry is also possible (if only 2D position orientation is desired)*
- GNSS provides absolute positioning information to mitigate IMU drift
- LIDAR provides accurate local positioning within known maps

+



+

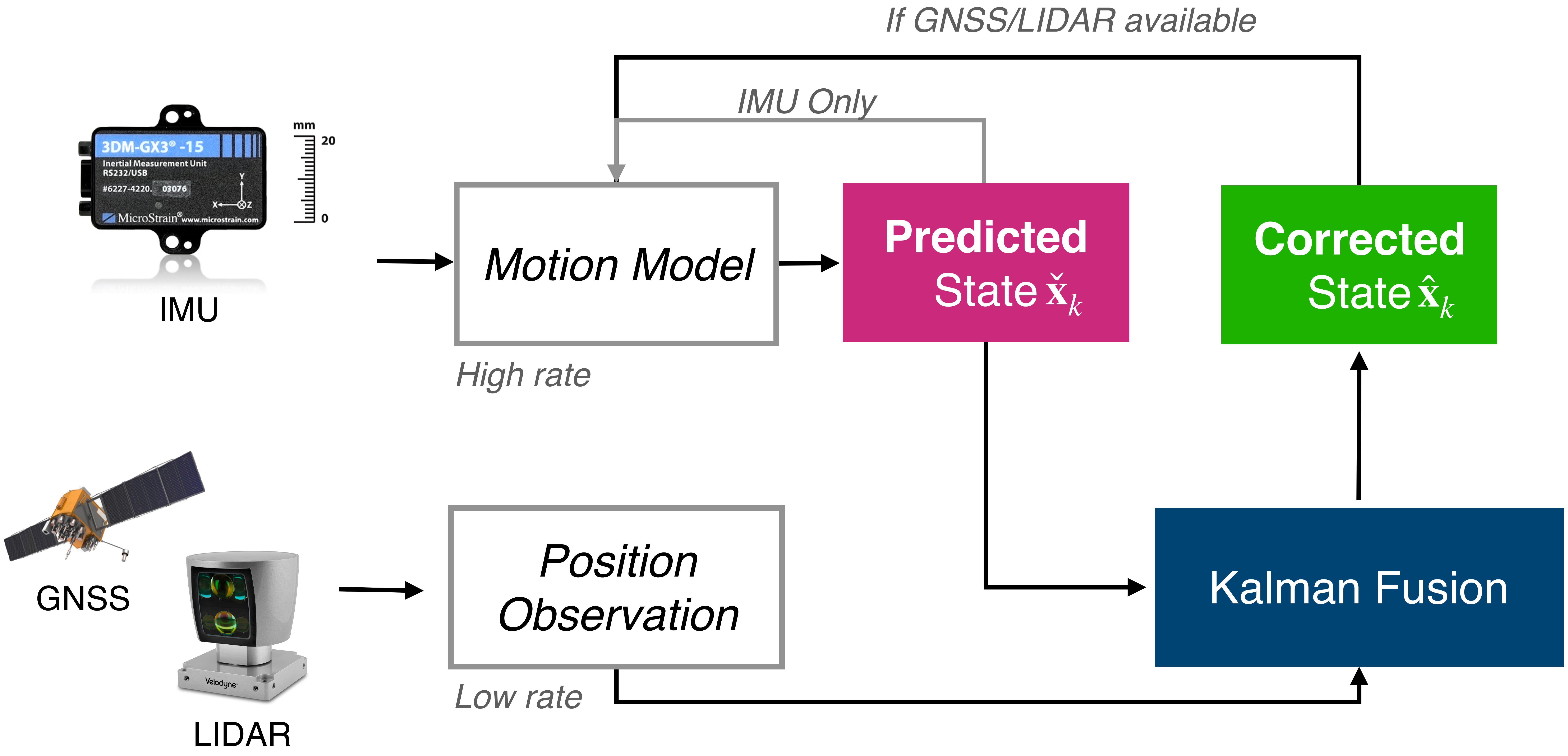


Tightly vs. Loosely Coupled

	EKF Coupling	
	Tight	Loose
GNSS/LIDAR Measurement	Pseudo-ranges to satellites LIDAR point clouds	Position
Accuracy	(Potentially) Higher	(Potentially) Lower
Complexity	Higher	Lower

Extended Kalman Filter

| IMU + GNSS + LIDAR



Some preliminaries

车辆状态包括位置，速度和使用单位四元数的方向参数化：

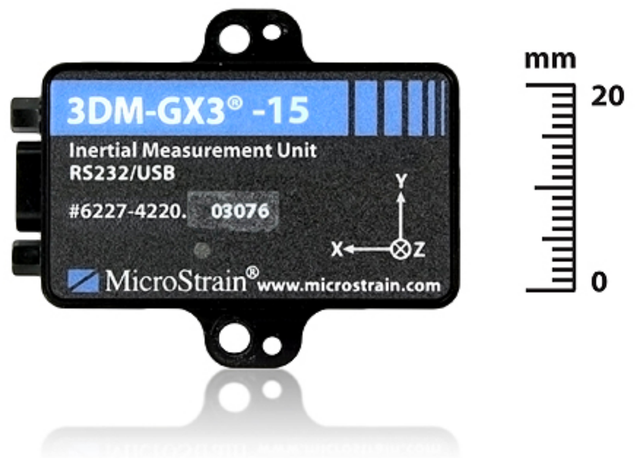
Vehicle state consists of position, velocity and parametrization of orientation using a unit quaternion:

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{p}_k \\ \mathbf{v}_k \\ \mathbf{q}_k \end{bmatrix} \in R^{10}$$

运动模型输入将包含来自IMU的特定力和转速：

Motion model input will consist of specific force and rotational rates from our IMU:

$$\mathbf{u}_k = \begin{bmatrix} \mathbf{f}_k \\ \boldsymbol{\omega}_k \end{bmatrix} \in R^6$$



Motion Model

Position $$\mathbf{p}_k = \mathbf{p}_{k-1} + \Delta t \mathbf{v}_{k-1} + \frac{\Delta t^2}{2}\left(\mathbf{C}_{ns}\mathbf{f}_{k-1} - \mathbf{g}\right)$$

Velocity $$\mathbf{v}_k = \mathbf{v}_{k-1} + \Delta t \left(\mathbf{C}_{ns}\mathbf{f}_{k-1} - \mathbf{g}\right)$$

Orientation $$\mathbf{q}_k = \boldsymbol{\Omega}\left(\mathbf{q}(\boldsymbol{\omega}_{k-1}\Delta t)\right)\mathbf{q}_{k-1}$$

where…

$$\mathbf{C}_{ns} = \mathbf{C}_{ns}(\mathbf{q}_{k-1}) \qquad \boldsymbol{\Omega}\left(\begin{bmatrix} q_w \\ \mathbf{q}_v \end{bmatrix}\right) = q_w \mathbf{1} + \begin{bmatrix} 0 & -\mathbf{q}_v^T \\ \mathbf{q}_v & -\{\mathbf{q}_v\}_\times \end{bmatrix} \qquad \mathbf{q}(\boldsymbol{\theta}) = \begin{bmatrix} \sin\frac{|\boldsymbol{\theta}|}{2} \\ \frac{\boldsymbol{\theta}}{|\boldsymbol{\theta}|}\cos\frac{|\boldsymbol{\theta}|}{2} \end{bmatrix}$$

Linearized **Motion Model**

Error State

$$\delta \mathbf{x}_k = \begin{bmatrix} \delta \mathbf{p}_k \\ \delta \mathbf{v}_k \\ \delta \boldsymbol{\phi}_k \end{bmatrix} \in R^9$$

3x1 rotation error

Error Dynamics

$$\delta \mathbf{x}_k = \mathbf{F}_{k-1} \delta \mathbf{x}_{k-1} + \mathbf{L}_{k-1} \mathbf{n}_{k-1}$$

measurement noise

where…

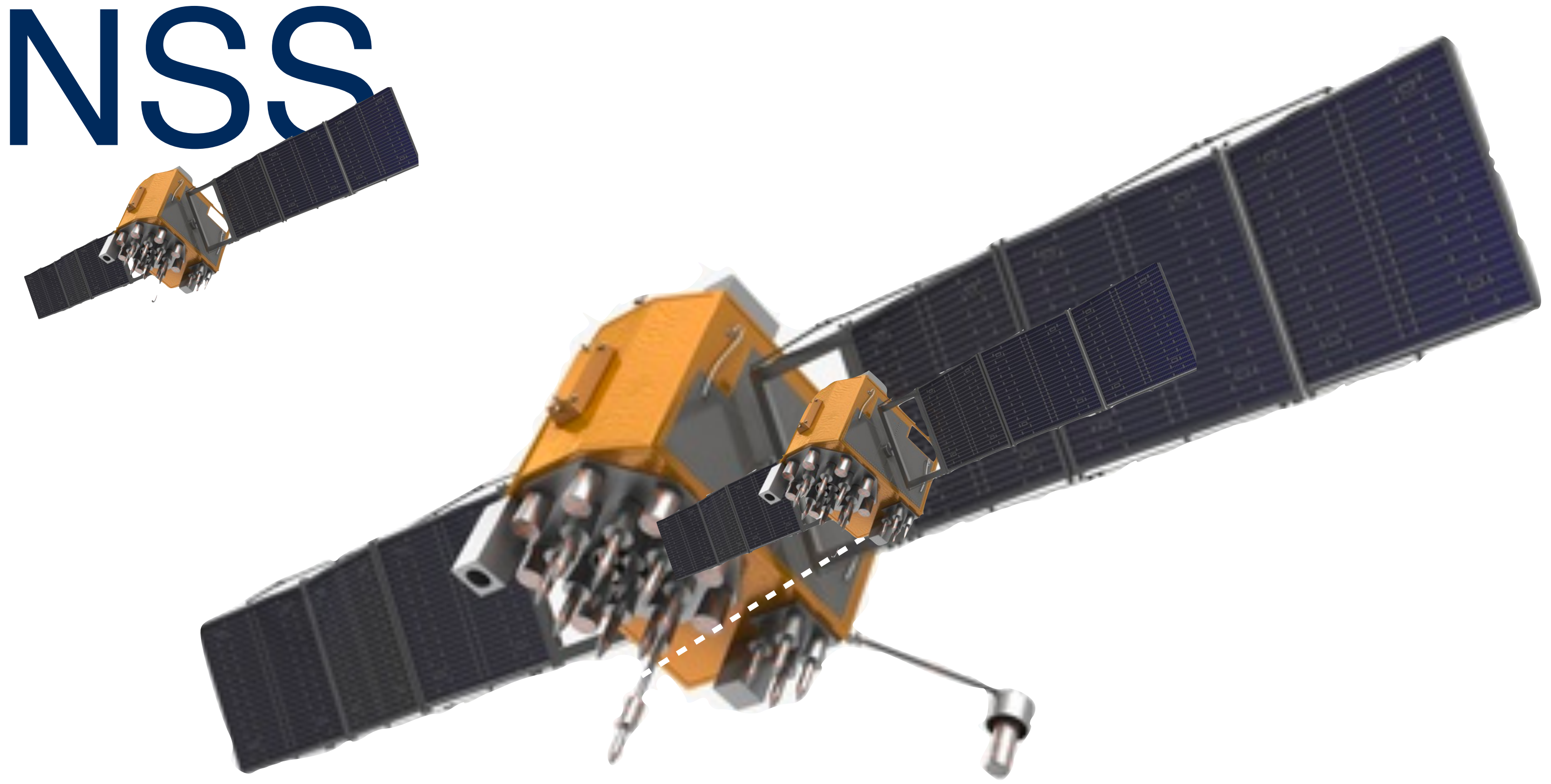
$$\mathbf{F}_{k-1} = \begin{bmatrix} \mathbf{1} & \mathbf{1}\Delta t & 0 \\ 0 & \mathbf{1} & -[\mathbf{C}_{ns} \mathbf{f}_{k-1}]_\times \Delta t \\ 0 & 0 & \mathbf{1} \end{bmatrix}$$

$$\mathbf{L}_{k-1} = \begin{bmatrix} 0 & 0 \\ \mathbf{1} & 0 \\ 0 & \mathbf{1} \end{bmatrix}$$

$$\mathbf{n}_k \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}_k\right) \sim \mathcal{N}\left(\mathbf{0}, \Delta t^2 \begin{bmatrix} \sigma^2_{\text{acc}} & \\ & \sigma^2_{\text{gyro}} \end{bmatrix}\right)$$

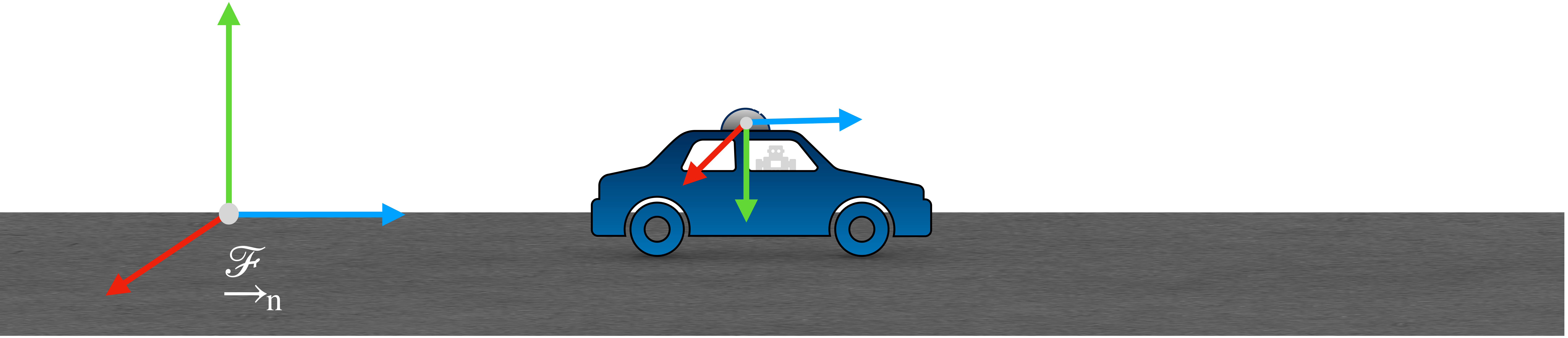
$\mathbf{1}$ is the 3×3 identity matrix

Measurement Model | GNSS



$$
\begin{aligned}
\mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k) + \boldsymbol{\nu}_k \\
&= \mathbf{H}_k \mathbf{x}_k + \boldsymbol{\nu}_k = [1 \;\; 0 \;\; 0]\, \mathbf{x}_k + \boldsymbol{\nu}_k \\
&= \mathbf{p}_k + \boldsymbol{\nu}_k
\end{aligned}
$$

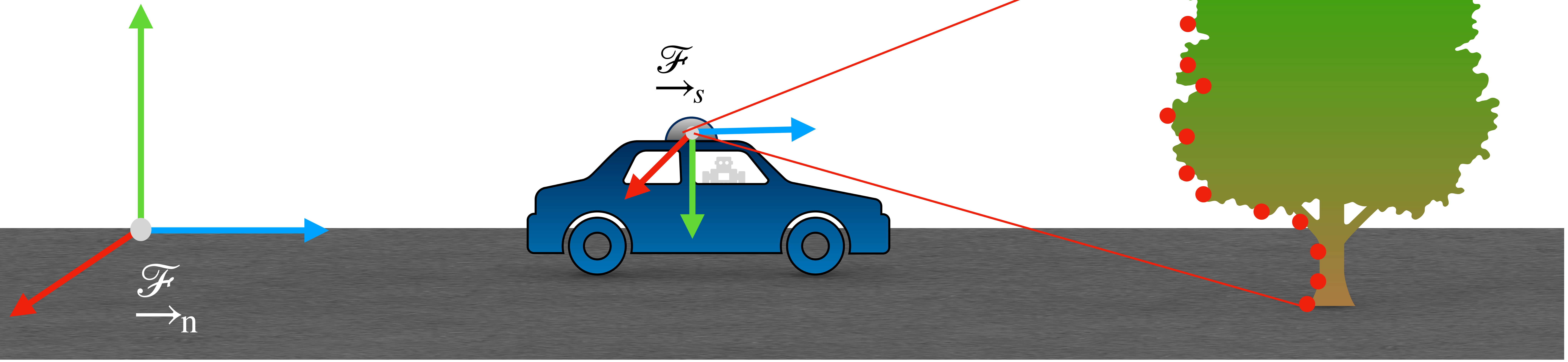
$$
\boldsymbol{\nu}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{\text{GNSS}})
$$



Measurement Model | LIDAR

$$
\begin{aligned}
\mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k) + \boldsymbol{\nu}_k \\
&= \mathbf{H}_k \mathbf{x}_k + \boldsymbol{\nu}_k = [1\ \ 0\ \ 0]\,\mathbf{x}_k + \boldsymbol{\nu}_k \\
&= \mathbf{p}_k + \boldsymbol{\nu}_k
\end{aligned}
$$

$$
\boldsymbol{\nu}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{\text{LIDAR}})
$$

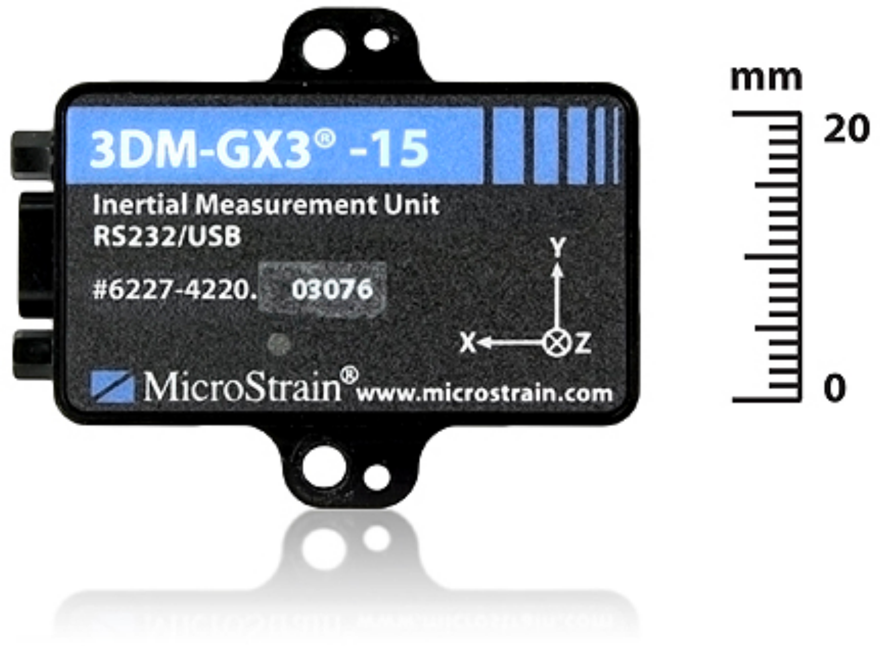


Known map

EKF | IMU + GNSS + LIDAR

Loop:

1. Update state with IMU inputs



$$\check{\mathbf{x}}_k = \begin{bmatrix} \check{\mathbf{p}}_k \\ \check{\mathbf{v}}_k \\ \check{\mathbf{q}}_k \end{bmatrix}$$

$$\check{\mathbf{p}}_k = \mathbf{p}_{k-1} + \Delta t \mathbf{v}_{k-1} + \frac{\Delta t^2}{2}\left(\mathbf{C}_{ns}\mathbf{f}_{k-1} + \mathbf{g}_n\right)$$

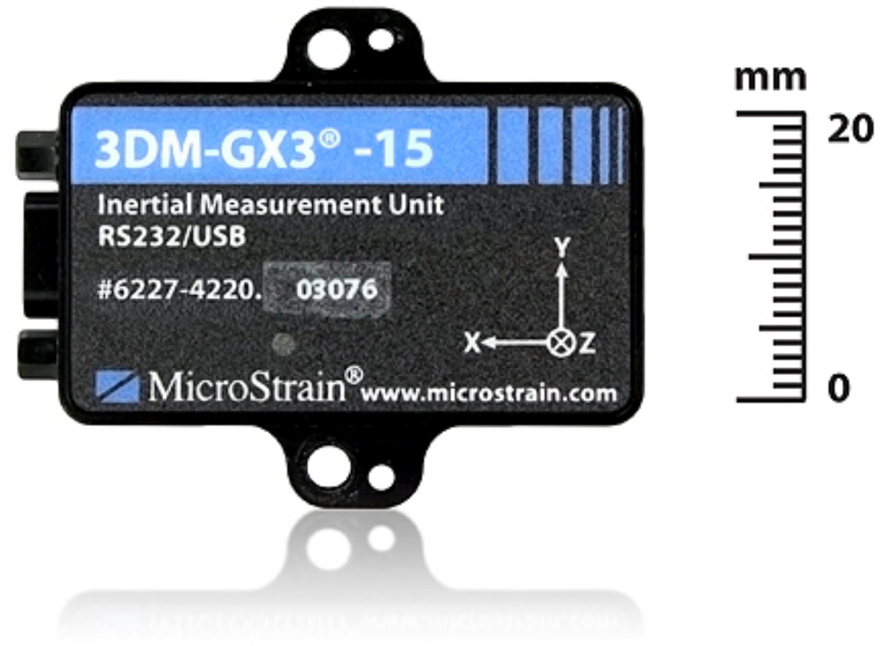
$$\check{\mathbf{v}}_k = \mathbf{v}_{k-1} + \Delta t \left(\mathbf{C}_{ns}\mathbf{f}_{k-1} + \mathbf{g}_n\right)$$

$$\check{\mathbf{q}}_k = \boldsymbol{\Omega}\left(\mathbf{q}(\omega_{k-1}\Delta t)\right)\mathbf{q}_{k-1}$$

$\mathbf{p}_{k-1}, \mathbf{v}_{k-1}, \mathbf{q}_{k-1}$

can be either be corrected or uncorrected depending on whether or not there was a GNSS/LIDAR measurement at time step *k - 1*

EKF | IMU + GNSS + LIDAR



Loop:

1. Update state with IMU inputs
2. Propagate uncertainty

$$\check{\mathbf{P}}_k = \mathbf{F}_{k-1}\mathbf{P}_{k-1}\mathbf{F}_{k-1}^T + \mathbf{L}_{k-1}\mathbf{Q}_{k-1}\mathbf{L}_{k-1}^T$$

Can either be
$\hat{\mathbf{P}}_{k-1}$ or $\check{\mathbf{P}}_{k-1}$

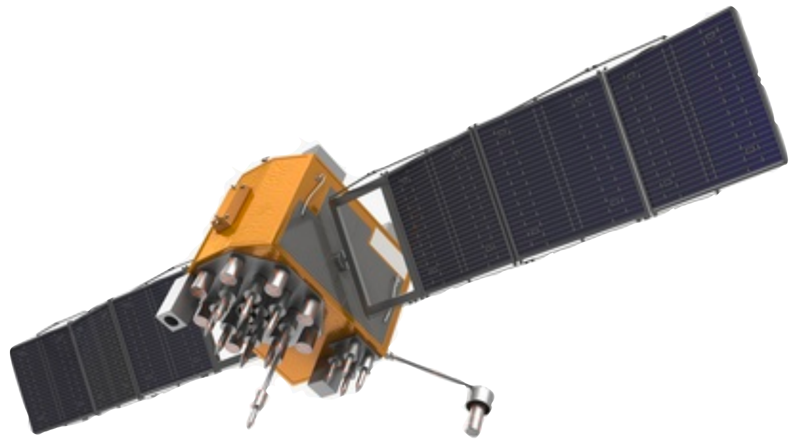
EKF | IMU + GNSS + LIDAR

Loop:

1. Update state with IMU inputs
2. Propagate uncertainty
3. If **GNSS** or **LIDAR** position available:
    1. Compute Kalman gain

$$\mathbf{K}_k = \check{\mathbf{P}}_k \mathbf{H}_k^T (\mathbf{H}_k \check{\mathbf{P}}_k \mathbf{H}_k^T + \mathbf{R})^{-1}$$

One of $\mathbf{R}_{\text{GNSS}}$ or $\mathbf{R}_{\text{LIDAR}}$

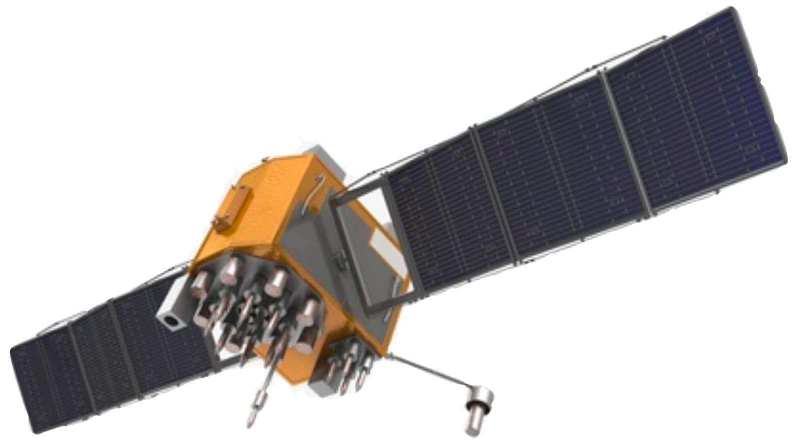


EKF | IMU + GNSS + LIDAR

Loop:

1. Update state with IMU inputs
2. Propagate uncertainty
3. If **GNSS** or **LIDAR** position available:
    1. Compute Kalman Gain
    2. Compute error state

$$\delta \mathbf{x}_k = \mathbf{K}_k (\mathbf{y}_k - \check{\mathbf{p}}_k)$$

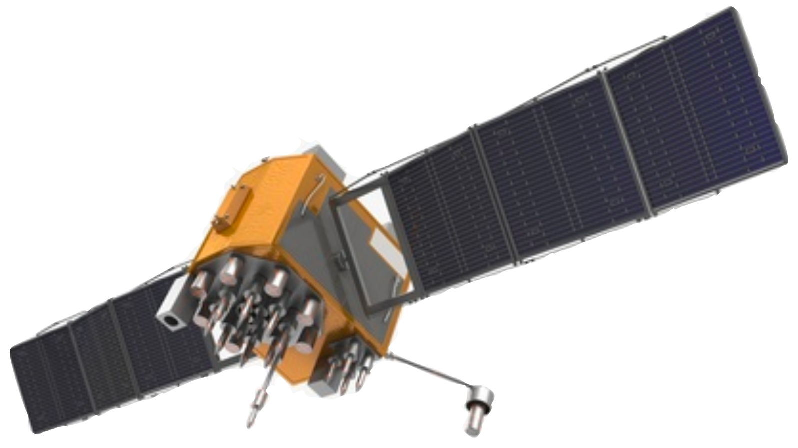


EKF | IMU + GNSS + LIDAR

Loop:

1. Update state with IMU inputs
2. Propagate uncertainty
3. If **GNSS** or **LIDAR** position available:
    1. Compute Kalman Gain
    2. Compute error state
    3. Correct predicted state

$$\hat{\mathbf{p}}_k = \check{\mathbf{p}}_k + \delta\mathbf{p}_k$$
$$\hat{\mathbf{v}}_k = \check{\mathbf{v}}_k + \delta\mathbf{v}_k$$
$$\hat{\mathbf{q}}_k = \mathbf{\Omega}(\mathbf{q}(\delta\boldsymbol{\phi}))\check{\mathbf{q}}_k$$

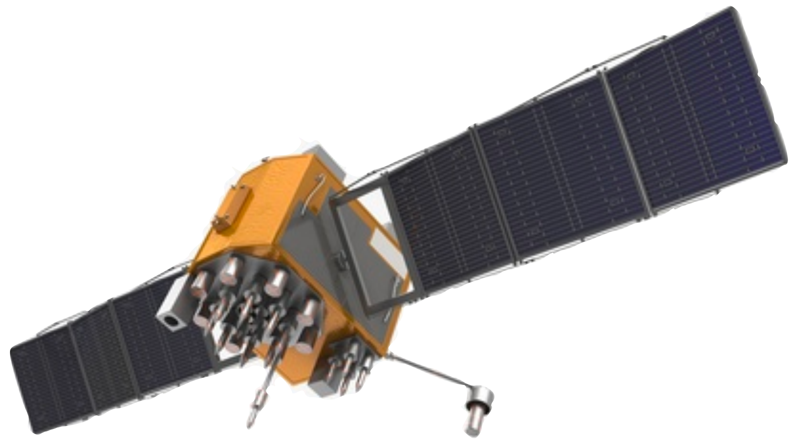


EKF | IMU + GNSS + LIDAR

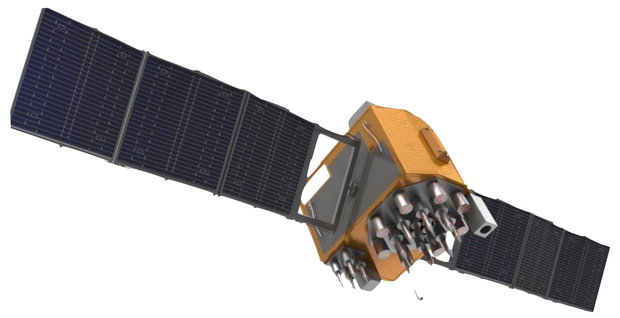
Loop:

1. Update state with IMU inputs
2. Propagate uncertainty
3. If **GNSS** or **LIDAR** position available:
    1. Compute Kalman Gain
    2. Compute error state
    3. Correct predicted state
    4. Computed corrected covariance

$$\hat{\mathbf{P}}_k = (\mathbf{1} - \mathbf{K}_k \mathbf{H}_k)\check{\mathbf{P}}_k$$



Summary | EKF for Vehicular State Estimation



- We used a loosely coupled EKF to fuse GNSS with IMU and LIDAR measurements
- *Assumptions*:
  1. LIDAR provides positions in the same reference frame as GNSS (possible)
  2. IMU has no biases (not realistic!)
  3. State initialization is provided (realistic)
  4. Our sensors are spatially and temporally aligned (somewhat realistic)