LAB 1 presentation

Bolton Athit Davies 66340500029

19 Febuary 2026

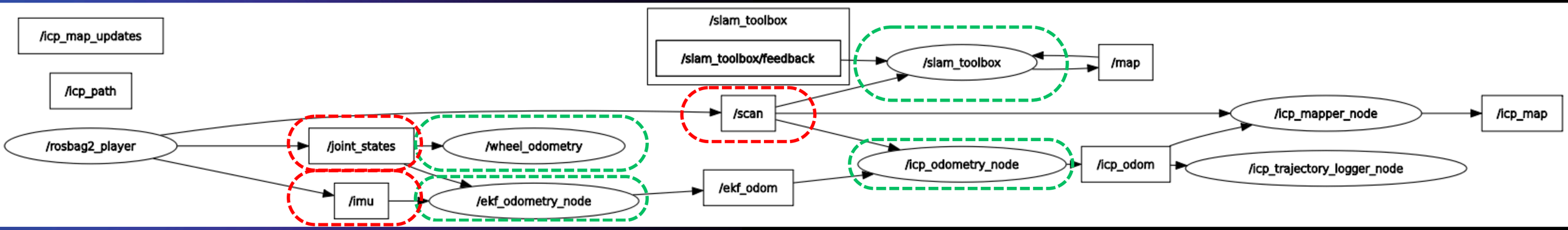
LAB 1 Objective

1. Fuse wheel odometry and IMU measurements using an Extended Kalman Filter (**EKF**)
2. Refine odometry using **LiDAR-based ICP** scan matching
3. Perform full **SLAM using slam_toolbox (Nav2)**
4. **Quantitatively and qualitatively compare** odometry estimation and mapping performance **across different methods**

LAB1 Deliverables

1. Source code
2. README
3. **Plots of trajectories from:**
    - **Wheel odometry**
    - **EKF odometry**
    - **ICP odometry**
    - **SLAM pose output**
4. **Generated 2D maps from ICP odometry and slam_toolbox**
5. **A discussion comparing accuracy, drift, and robustness of each method**

LAB1 SYSTEM DESIGN



/Jointstate /imu —EKF→ /ekf_odom

/ekf_odom /scan —ICP→ /icp_odom

/wheel_odom /scan —SLAM→ /slam_odom

Extended Kalman filter (EKF)

1. Motion model : $f(x, y, yaw) = \begin{pmatrix} x_k + (d_k + v_{1,k}).\cos(\Theta_k + \Delta\Theta_k + v_{2,k}) \\ y_k + (d_k + v_{1,k}).\sin(\Theta_k + \Delta\Theta_k + v_{2,k}) \\ \Theta_k + \Delta\Theta_k + v_{2,k} \end{pmatrix}$

2. Jacobian Matrix of the state : $\nabla f_X = \begin{pmatrix} 1 & 0 & -(d_k + v_{1,k})\sin(\Theta_k + \Delta\Theta_k + v_{2,k}) \\ 0 & 1 & (d_k + v_{1,k})\cos(\Theta_k + \Delta\Theta_k + v_{2,k}) \\ 0 & 0 & 1 \end{pmatrix}$

3. Process noise : np.diag([0.05, 0.05, 0.02])

4. Sensor model : IMU and odom from wheel (map 1 : 1)

5. Measurement noise (odom) = np.diag([0.05, 0.05, 0.1])

6. Measurement noise (imu) = np.array([[0.5]])

	Equation	EKF Equation
Predict	State Extrapolation	$\hat{\boldsymbol{x}}_{n+1,n} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{n,n}) + \boldsymbol{G}\boldsymbol{u}_n$
	Covariance Extrapolation	$\boldsymbol{P}_{n+1,n} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}} \boldsymbol{P}_{n,n} \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right)^T + \boldsymbol{Q}$
Update	State Update	$\hat{\boldsymbol{x}}_{n,n} = \hat{\boldsymbol{x}}_{n,n-1} + \boldsymbol{K}_n\left(\boldsymbol{z}_n - \boldsymbol{h}(\hat{\boldsymbol{x}}_{n,n-1})\right)$
	Covariance Update	$\boldsymbol{P}_{n,n} = \left(\boldsymbol{I} - \boldsymbol{K}_n \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\right) \boldsymbol{P}_{n,n-1} \times \left(\boldsymbol{I} - \boldsymbol{K}_n \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\right)^T + \boldsymbol{K}_n \boldsymbol{R}_n \boldsymbol{K}_n^T$
	Kalman Gain	$\boldsymbol{K}_n = \boldsymbol{P}_{n,n-1} \left(\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\right)^T \times \left(\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}} \boldsymbol{P}_{n,n-1} \left(\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\right)^T + \boldsymbol{R}_n\right)^{-1}$

Table 13.1: *Kalman Filter equations.*

Ref : KALMAN FILTER from the Ground up : Alex Becker : First Edition,
Extended Kalman Filter Sensor Fusion in Practice for Mobile Robot Localization Alaa Aldeen Housein, Gao Xingyu*, Weiming Li, Yang Huang

Iterative Closest Point (ICP)

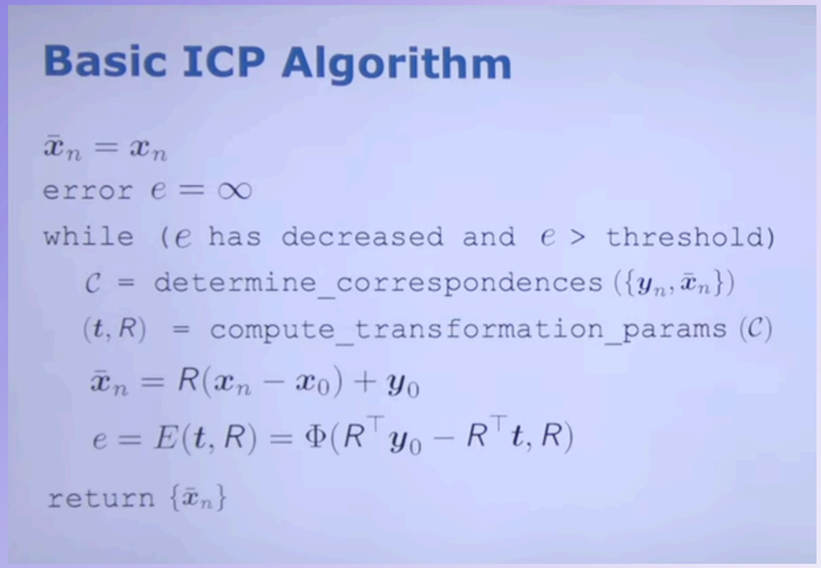
Default parameter from document

Example :

```
# ICP Odometry Parameters
icp_odometry_node:
  ros__parameters:
    # ICP Algorithm Parameters
    max_iterations: 100
    convergence_threshold: 0.001
    max_correspondence_distance: 0.005

    # Laser Scan Parameters
    max_range: 3.5
    min_range: 0.119
    angle_increment: 0.017453293  # ~1 degree in radians

    # Odometry Parameters
    publish_tf: true
    base_frame: "base_link"
    odom_frame: "odom"
```



Ref : ICP & Point Cloud Registration - Part 1: Known Data Association & SVD (Cyrill Stachniss, 2021)

Simultaneous Localization and Mapping (SLAM)

Default parameter from document

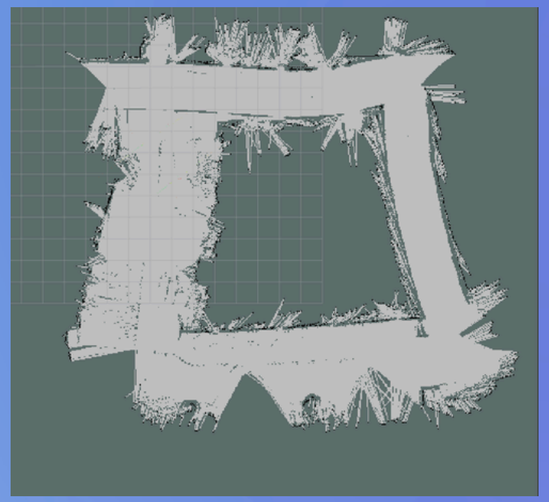
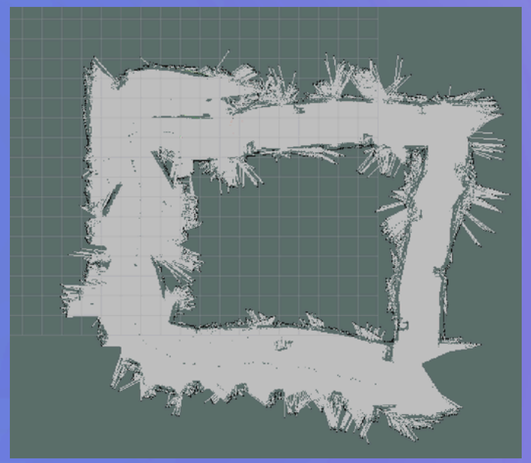
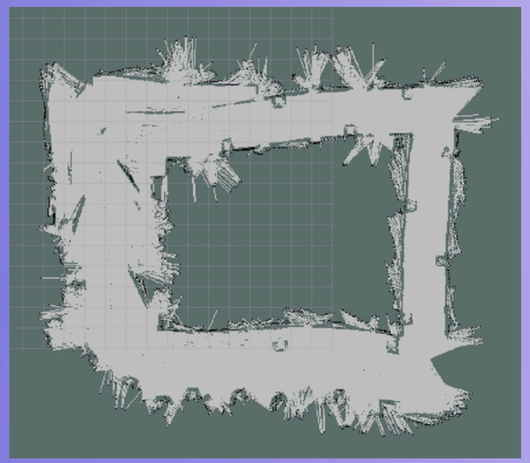
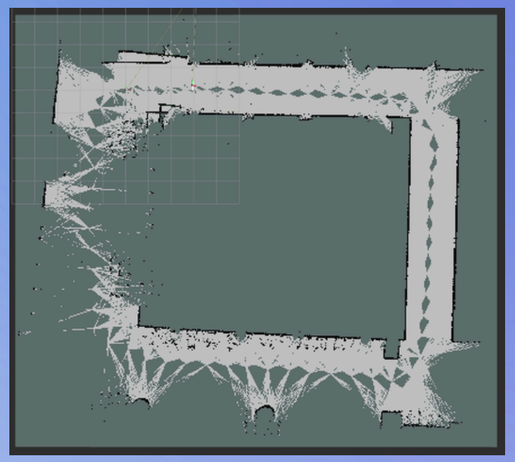
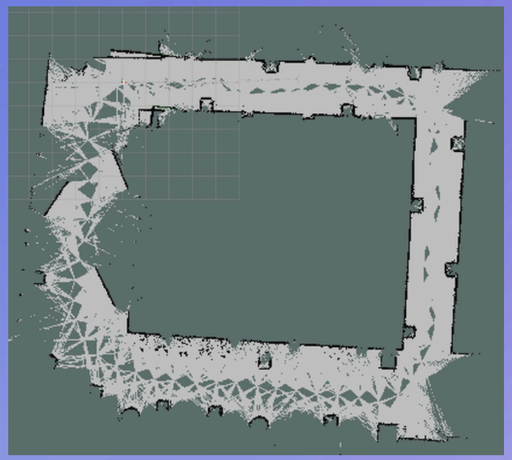
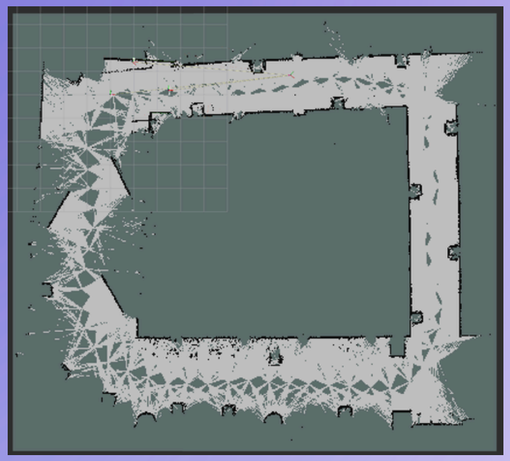
Example :

1. Correlation Parameters - Loop Closure Parameters
   - loop_search_space_dimension: 8.0
   - loop_search_space_resolution: 0.05
   - loop_search_space_smear_deviation: 0.03

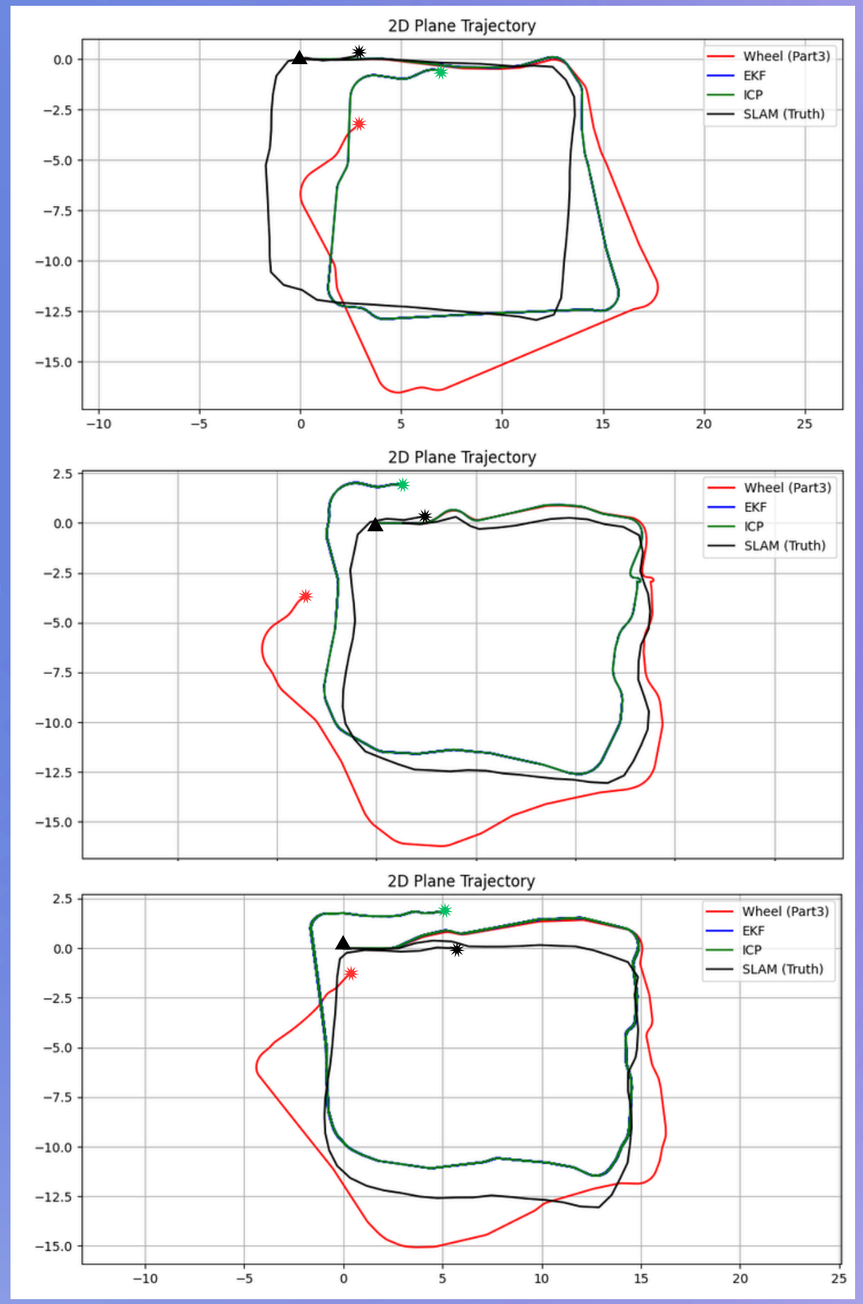
2. Scan Matcher Parameters
   - distance_variance_penalty: 0.5
   - angle_variance_penalty: 1.0
   - fine_search_angle_offset: 0.00349
   - coarse_search_angle_offset: 0.349
   - coarse_angle_resolution: 0.0349
   - minimum_angle_penalty: 0.9
   - minimum_distance_penalty: 0.5
   - use_response_expansion: true

Ref : https://docs.ros.org/en/humble/p/slam_toolbox/

Result : Mapping

	seq00	seq01	seq02
ICP			
SLAM			

Result : Path



seq00

seq01

seq02

y axis (m)

x axis (m)

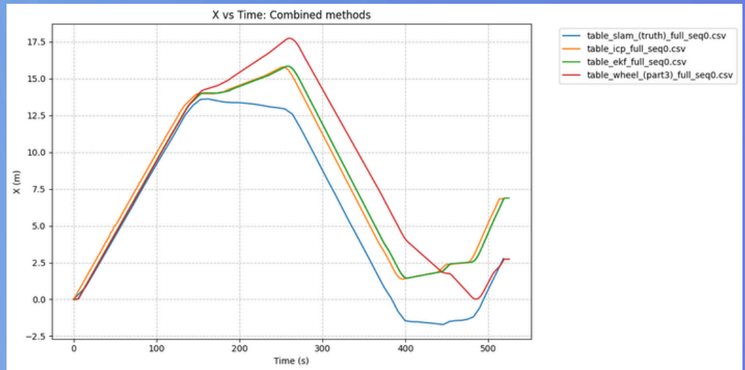
Result : trajectory

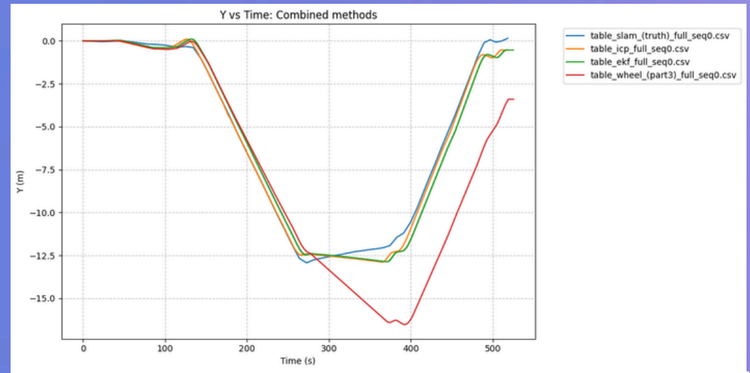
X (M)

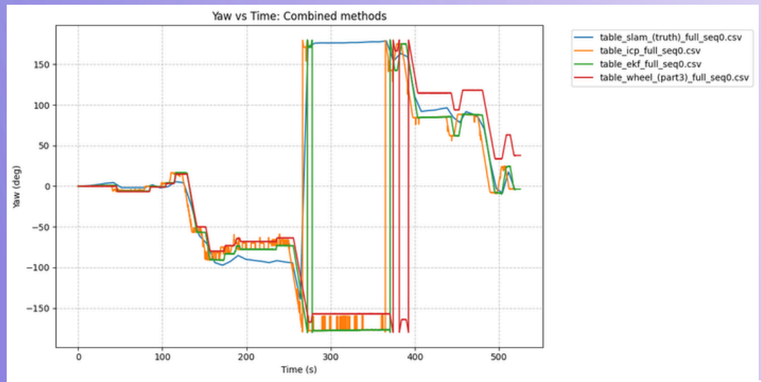
Y (M)

Yaw (rad)

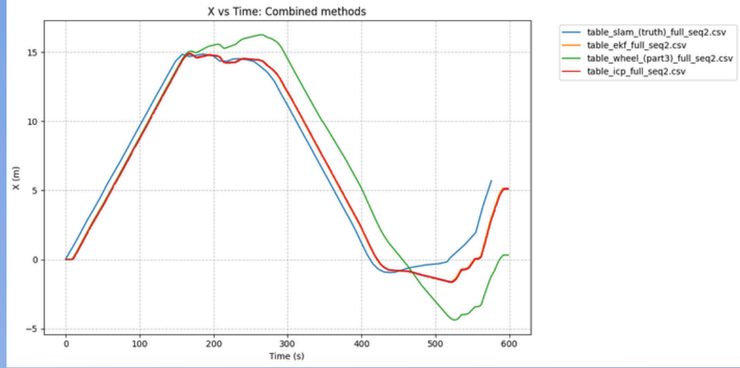
seq00

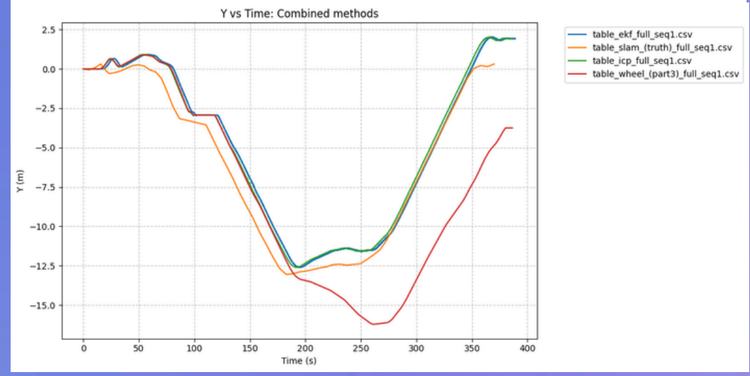


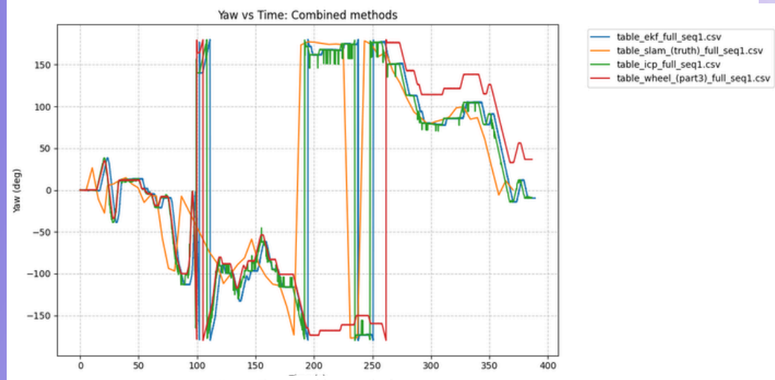




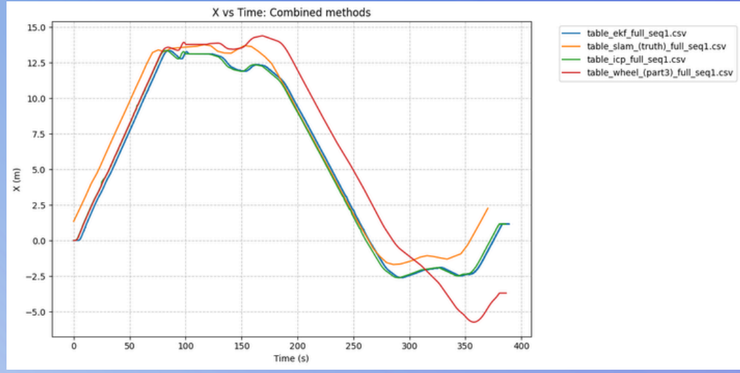
seq01

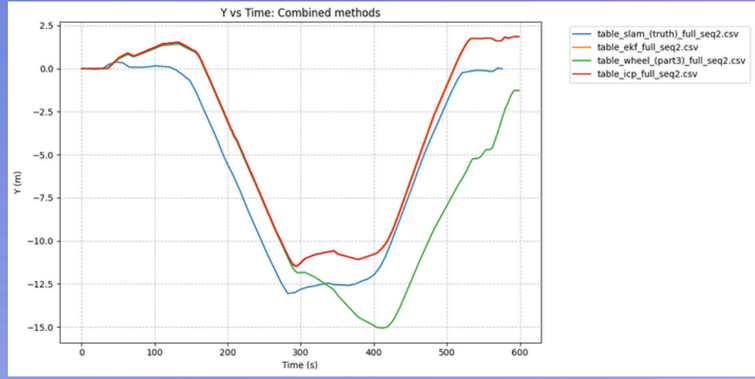


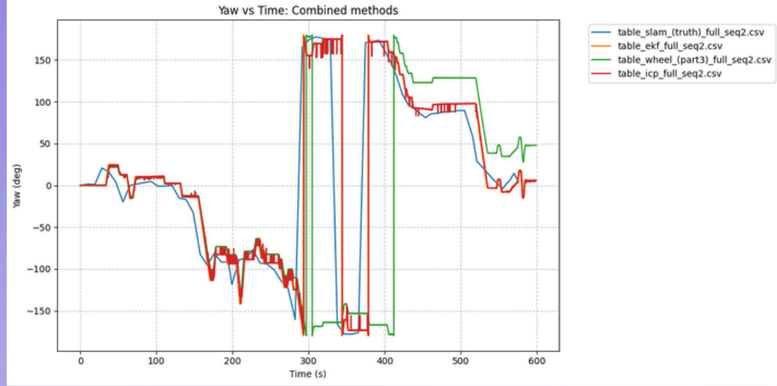




seq02







Discussion : Accuracy

Assuming Slam odometry value as ground truth : seq0

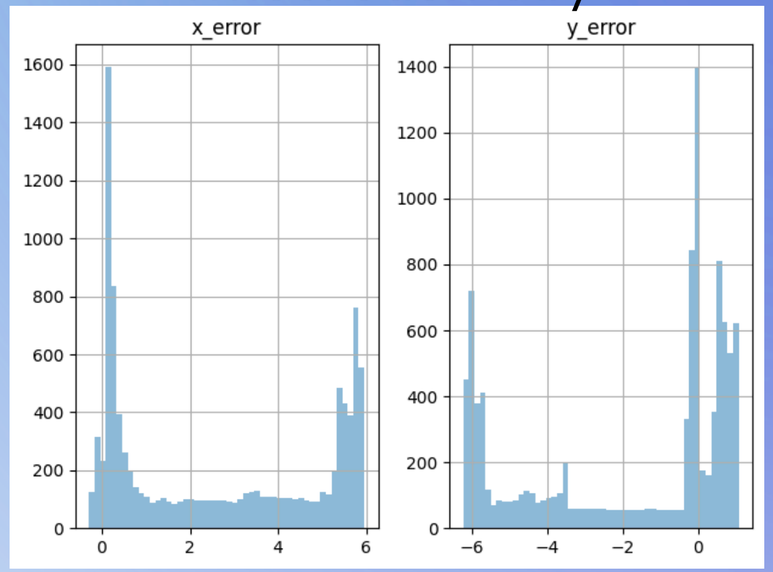
X (m)

Method	Min	Mean	S.D.	Max
Wheel odometry	-0.285	2.651	2.304	5.955
EKF odometry	-0.282	2.025	1.461	4.103
ICP odometry	0.009	2.027	1.268	4.640

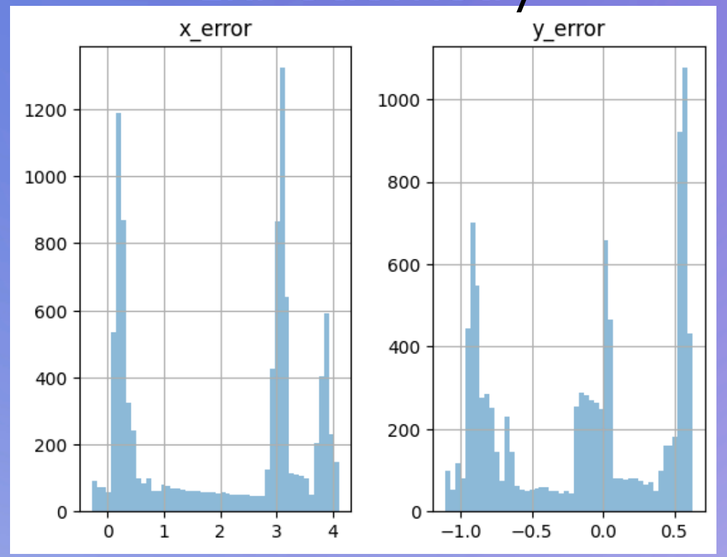
Y (m)

Method	Min	Mean	S.D.	Max
Wheel odometry	1.066	-1.776	2.648	-6.217
EKF odometry	-1.109	-0.149	0.576	0.628
ICP odometry	-1.016	-0.160	0.306	0.514

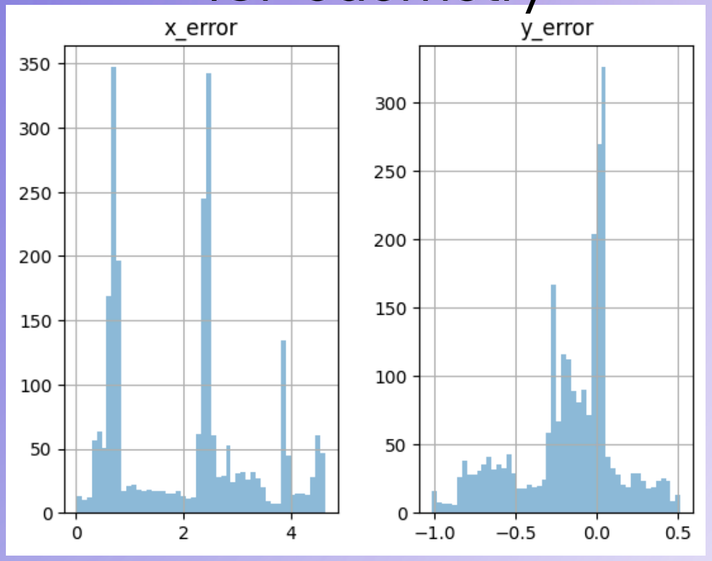
Wheel odometry



EKF odometry



ICP odometry



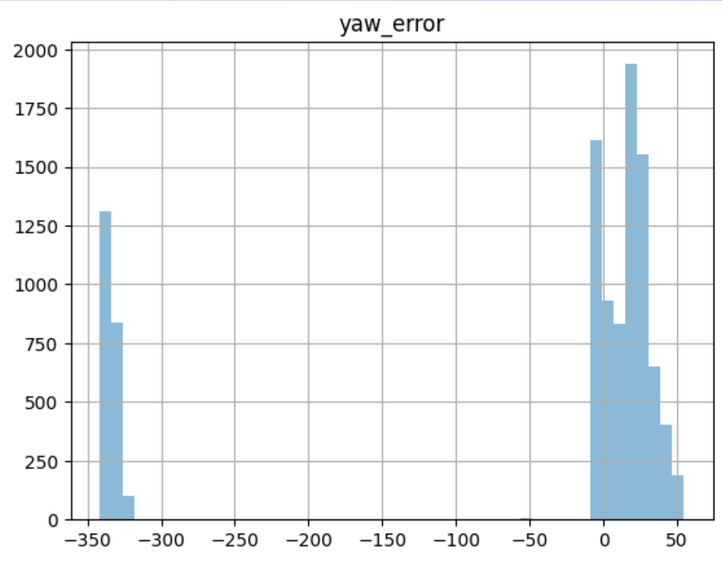
Discussion : Accuracy

Assuming Slam odometry value as ground truth : seq0

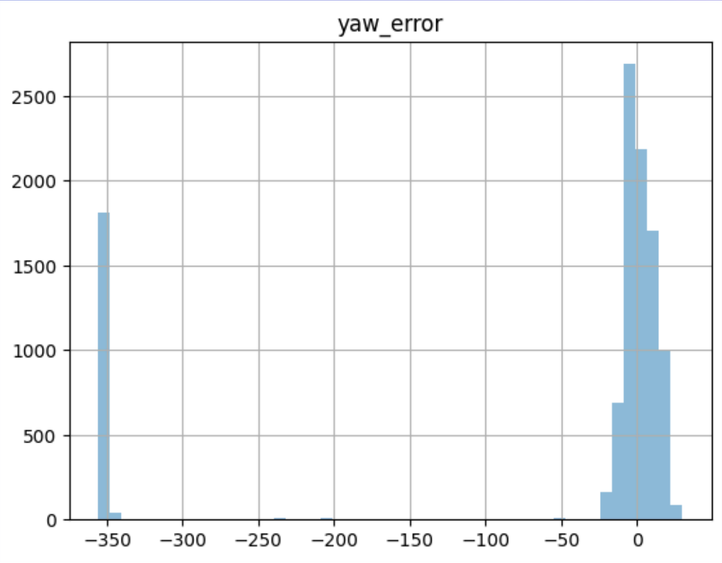
yaw (rad)

Method	Min	Mean	S.D.	Max
Wheel odometry	-341.474	-61.117	144.753	54.743
EKF odometry	-355.66	-62.929	136.96	29.916
ICP odometry	-357.976	-62.553	136.566	221.929

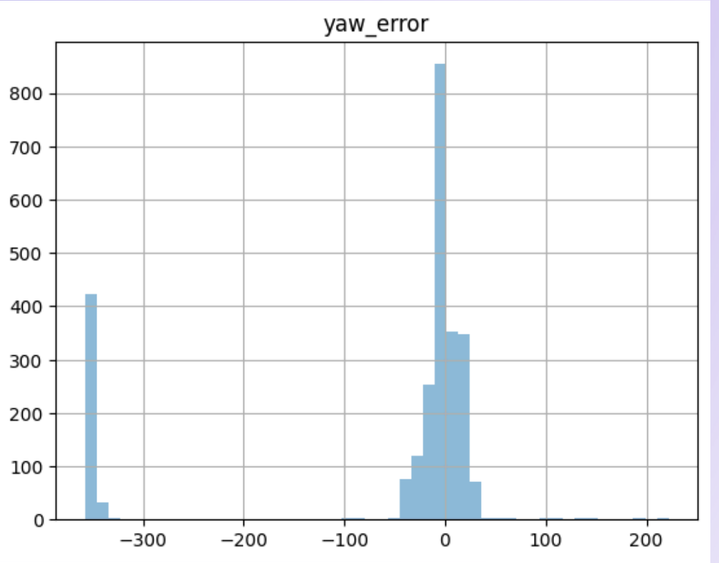
Wheel odometry



EKF odometry



ICP odometry



Discussion : Accuracy

Assuming Slam odometry value as ground truth : seq0

1. Position Error Distribution (X and Y)

The histograms for X and Y errors reveal how consistently each algorithm maintains the robot's position:

- Wheel Odometry (Bimodal/Skewed): The X and Y error distributions for wheel odometry are widely spread and exhibit multiple peaks (bimodal). The Y-error, in particular, shows a heavy concentration near -6.0 meters, indicating that the error is not just random noise but a severe systematic drift.
- EKF Odometry (Clustered Peaks): The EKF distributions are significantly more "tight" than the wheel-only data. In the Y-axis, the error is highly concentrated between -1.0 and 0.5 meters, showing that the sensor fusion effectively centers the robot's estimated path near the truth.
- ICP Odometry (High Precision): The ICP error distributions show the highest precision. The Y-error histogram is the most "peaked" (narrow) and is centered almost perfectly at 0.0 meters, demonstrating that scan-matching provides a superior zero-mean error distribution compared to the other methods.

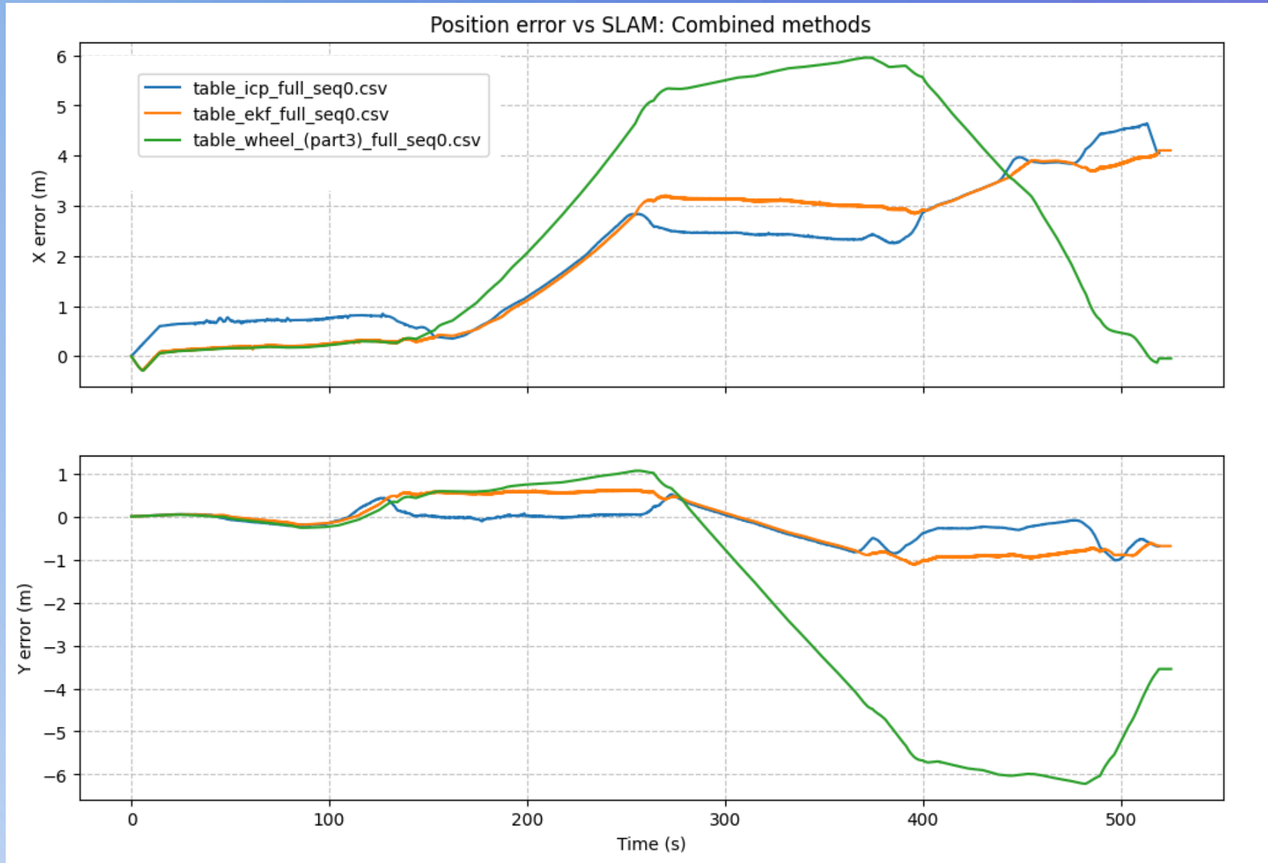
2. Yaw Heading Error Distribution

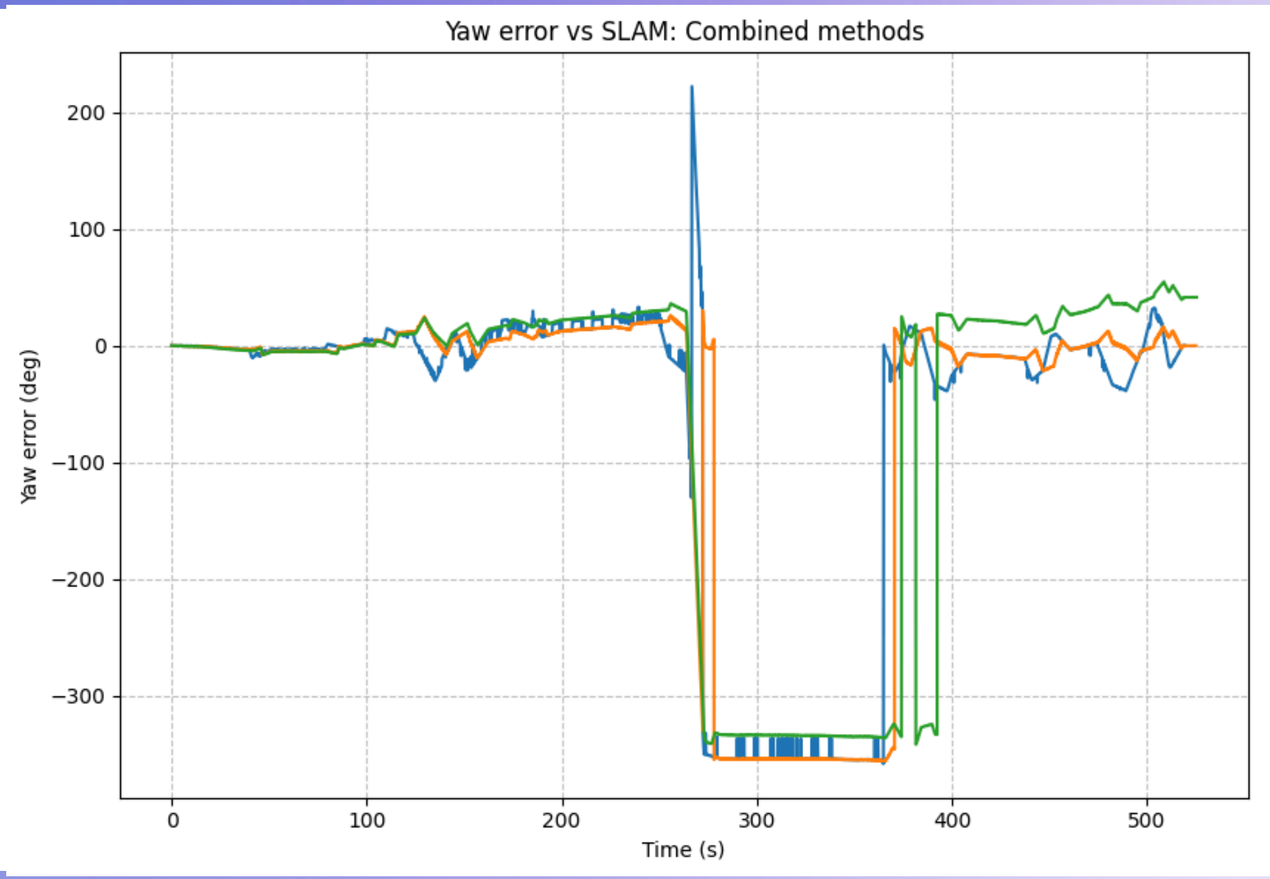
The yaw (orientation) error is the most critical factor for 2D accuracy, as heading errors lead to exponential position divergence.

- Broad Distribution (Wheel): The yaw error histogram for wheel odometry is the broadest, with a high Standard Deviation of 144.753 rad. This wide spread represents the "random walk" of the heading as the robot makes turns and the wheels slip.
- Targeted Correction (EKF/ICP): Both EKF and ICP show distributions with more distinct vertical peaks. The S.D. of 136.566 rad for ICP is the lowest recorded, confirming that environmental scan matching acts as a strong constraint on rotational drift.

Discussion : Drift

Assuming Slam odometry value as ground truth : seq 0





Discussion : Drift

Assuming Slam odometry value as ground truth : seq 0

1. Cumulative Position Drift (Time-Series Analysis)

The "Position error vs SLAM" plots provide a direct visualization of drift over the 500-second sequence:

- Wheel Odometry (Linear Growth): The green line in the position error plot shows a steep, nearly linear increase in error as time progresses. In the Y-axis, the drift accelerates after the first 250 seconds, plummeting to a maximum error of -6.217 meters. This is a classic example of unbounded dead-reckoning drift where every small slip of the wheel adds to the permanent error.
- EKF & ICP (Bounded Drift): Both the EKF (orange) and ICP (blue) exhibit much flatter error profiles. In the Y-axis, they manage to keep the drift within a +/- 1.0 meter range for the entire duration. While they still drift in the X-axis (reaching ~4.0m), the rate of accumulation is significantly slower than the raw wheel odometry.

2. Rotational Drift and Yaw Stability

Rotational drift is the "root cause" of positional drift; a 1-degree error in heading results in increasing lateral displacement the further the robot travels.

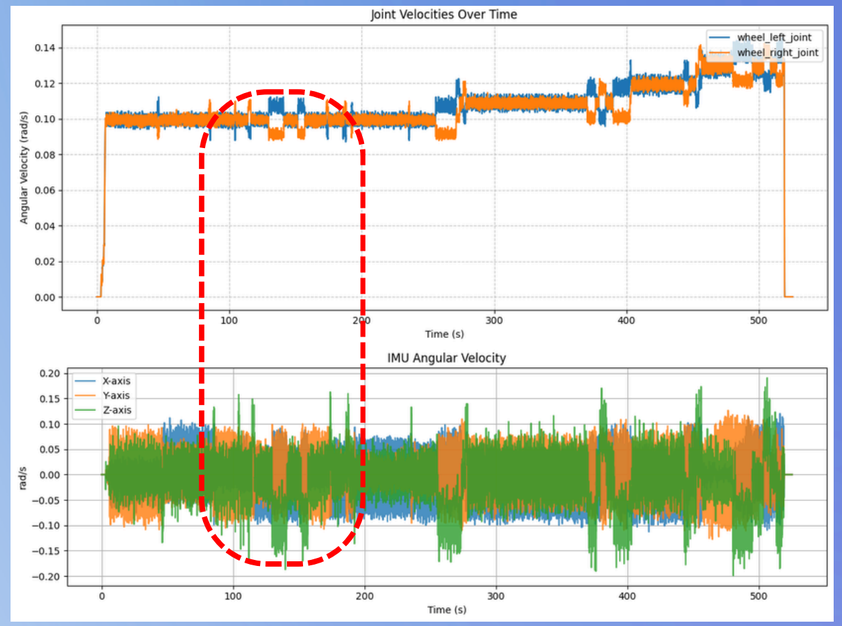
- Wheel Odometry Instability: The yaw error plot shows the wheel odometry heading diverging significantly, particularly during sharp turns. The high Standard Deviation (S.D.) of 144.753 rad indicates that the robot's "internal compass" is highly unstable.
- EKF/ICP Heading Correction: By fusing IMU data (EKF) and matching LiDAR scans (ICP), the rotational drift is periodically "reset" or constrained. The ICP method achieves the lowest yaw S.D. of 136.566 rad, proving that environmental referencing is the most effective way to combat heading drift in structured hallways.

Discussion : Robustness

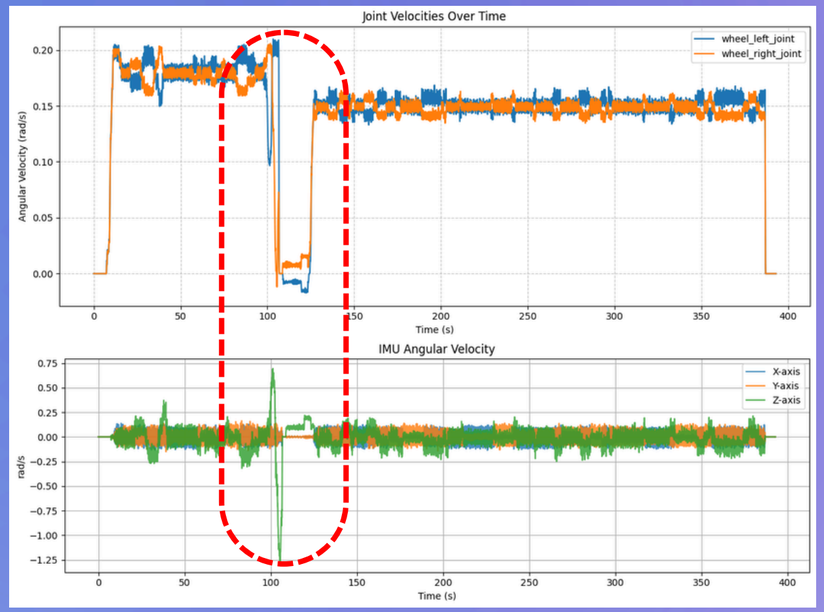
Assuming Slam odometry value as ground truth

Rosbag data

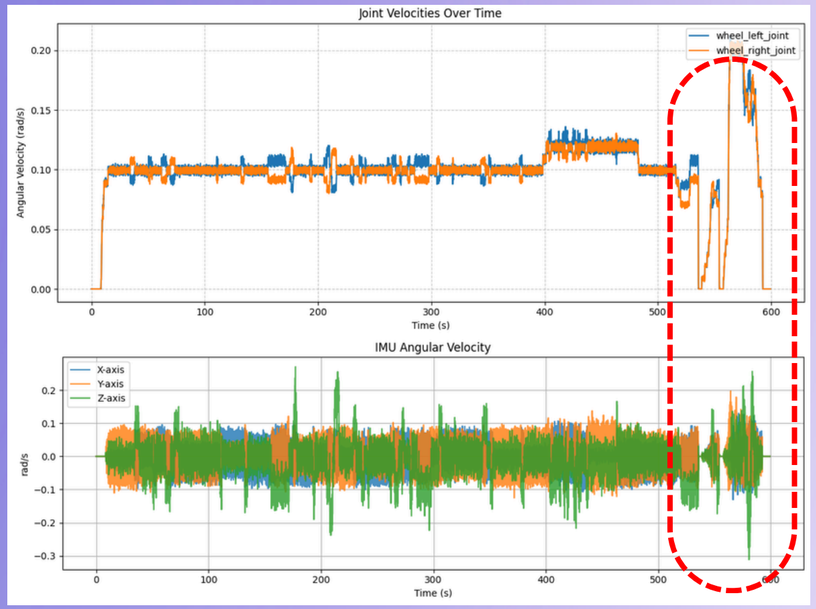
Seq0



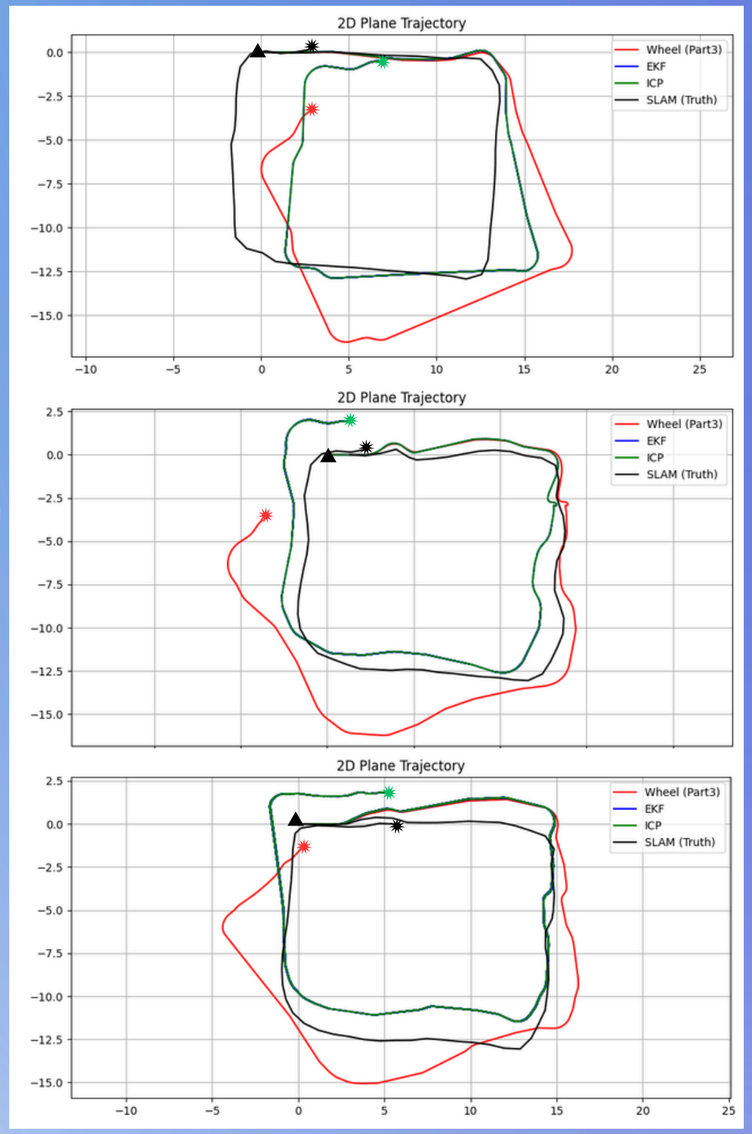
Seq1



Seq2



Discussion : Robustness



seq00

seq01

seq02

y axis (m)

x axis (m)

1. Handling Sharp Turns and Aggressive Motion

Sequence 01 was specifically designed to challenge the system with "sharp turning motion".

- Wheel Odometry (Low Robustness): In Sequence 01, the wheel odometry trajectory (red line) shows severe distortion immediately following the first sharp turn. This is caused by significant wheel slippage that occurs when angular velocity spikes, which the encoders cannot detect.
- EKF and ICP (High Robustness): Both EKF and ICP maintain high fidelity to the SLAM path even during these sharp maneuvers. The EKF is particularly robust here because the IMU provides high-frequency angular velocity data that overrides the slipping wheel data during the peak of the turn.

2. Resilience to Sensor Noise and Jitter

The "Sequence 00" trajectory reveals a specific robustness issue regarding the initial ICP implementation.

- ICP Sensitivity: In the initial results for Sequence 00, the ICP path (green) showed extreme "jitter" and jagged lines. This demonstrates a lack of robustness to Lidar sensor noise.
- EKF Smoothness: Throughout all sequences, the EKF (blue) provides the most consistently smooth trajectory. It is highly robust to the high-frequency "salt and pepper" noise found in raw Lidar scans because it uses a probabilistic model to filter out sudden, unrealistic jumps in position.

Thank you