Technical Debugging Note: Warehouse Robot Failures

Date: January 31, 2026
Subject: Investigation of Localization Drift, Update Delays, and TF Instability

1. Problem Description

After 10-15 minutes of operation, the robot exhibits localization drift where pose estimate deviates from ground truth.

Pose updates in RViz appear delayed or jumpy.

TF warnings are logged regarding discarded messages or transform timeouts.

2. Root Cause Analysis

- Odometry Drift: Wheel encoders accumulate error due to floor material or acceleration bias.

- CPU Bottleneck: Robot processor saturated (e.g., >80% usage), causing AMCL update delays.

- Time Synchronization: Clock drift between the robot and remote monitoring machines.

- Feature Repetition: Warehouse aisles are too identical for LiDAR to distinguish (aliasing).

- IMU/Odom Mismatch: Discordance between sensor fusion inputs.

3. Isolation Strategies

- Isolation 1: Command a 1m square path. If it physical location != computed, it's Odometry.

- Isolation 2: Monitor 'top' or 'htop' during movement for CPU throttle.

- Isolation 3: Use 'tf2_monitor' to check for latency/jitter in the TF tree.

- Isolation 4: Check /amcl_pose covariance. If it grows in aisles, it's environment aliasing.

4. Recommended Debugging Tools

- TF Tree: 'ros2 run tf2_tools view_frames' for connection integrity.

- RQT Graph: Graph visualization for node communication checks.

- Topic Stats: 'ros2 topic hz' to verify sensor data consistency.

- Rosbag: Recording data for offline playback and detailed frame-by-frame analysis.

5. Proposed Solutions

- Fuse IMU data with Odometry using an EKF (Extended Kalman Filter).

- Lower LiDAR frequency or resolution to save CPU cycles.

- Use 'chrony' or 'ntp' to maintain microsecond accuracy between computers.

- Tune the 'transform_tolerance' in the localization YAML.