**Joint Velocities from Gyrosopes**

- Computed using either relative link poses or link Jacobians $J^W_{Li}$ which provide constraints.
- Develop automatic IMU orientation calibration to account for unknown offsets.

**Motivation**

- Humans use one IMU for base pose estimation
- High-quality IMUs have become cheap and common
- What can we estimate with many IMUs?[1]

**Setup**

- Attached one Microstrain 3DM-GX3-25 IMU per link.
- 3-Axis Gyroscope: $\omega_{W} = R^W_{Ri} \omega_{W}^{Ri}$
- 3-Axis Accelerometer: $\lambda_{W} = R^W_{Ri} \lambda_{W}^{Ri} + \lambda$

**Goals**

- Joint derivatives computed numerically from noisy potentiometers, filtering induces delays $\rightarrow$ Compute derivatives directly from inertial sensors
- Account for unknown link IMU poses
- Increase control authority/damping using IMU signals

**IMU-Based Feedback Control**

- PD control sine task with both numerically-computed and IMU-based velocities
- Reduced RMS error from 0.0103 rad to 0.0099 rad in position and from 0.3786 rad/s to 0.0902 rad/s in velocity.
- Able to increase feedback gains by 50%
- Future work: use in a whole-body controller[2]

**Link IMU Pose Calibration**

- Compute orientation and position corrections in base frame (denoted $\hat{R}$ and $\hat{r}_{ij}$) assuming leg acts as a rigid body.
- Achieve this by "locking" leg and physically rotating robot.

**Orientation Calibration**

- All links have velocity $\hat{R} \hat{\omega} = R^W_{R_i} \hat{\omega}_i$
- Transpose, stack M measurements:
  $$\begin{bmatrix} \hat{\omega}_1^T \\ \vdots \\ \hat{\omega}_M^T \end{bmatrix} \hat{R} = \begin{bmatrix} \hat{\omega}_1^T R^W_{R_i} \hat{\omega}_i \\ \vdots \\ \hat{\omega}_M^T R^W_{R_i} \hat{\omega}_i \end{bmatrix}$$
- Solve $\hat{R} = \arg\min_{\hat{R}} ||AX - B||^2$ (Orthogonal Procrustes Problem)
- Compute $\text{SVD } \hat{A}^T = B \Sigma \hat{V}^T \rightarrow \hat{R} = \Sigma \hat{V}^T$, $\Sigma = \text{diag}(1,1,\text{sign}(\det(\Sigma)))$

**Position Calibration**

- Acceleration of link $i$ IMU with respect to base is $\hat{a}_i^W = \hat{a}_i^W - \hat{a}_i^W = \hat{a}_i^W \times \hat{\omega}_i + \omega_i \times \hat{r}_{ij}^W$
- Using $\hat{a}_i = R^W_{R_i} \hat{a}_i^W + g$, obtain
  $$\begin{bmatrix} \hat{\omega}^2 + \hat{\omega}_i^2 \\ \vdots \\ \hat{\omega}^2 + \hat{\omega}_i^2 \end{bmatrix} \hat{r}_{ij} = \begin{bmatrix} \hat{\omega}_1^T \hat{\omega}_1 \hat{R} \\ \vdots \\ \hat{\omega}_M^T \hat{\omega}_M \hat{R} \end{bmatrix}$$
- Stack $M$ measurements to form system and solve using SVD:
  $$\begin{bmatrix} \hat{\omega}^2 + \hat{\omega}_i^2 \\ \vdots \\ \hat{\omega}^2 + \hat{\omega}_i^2 \end{bmatrix} \hat{r}_{ij} = \begin{bmatrix} \hat{\omega}_1^T \hat{\omega}_1 \hat{R} \\ \vdots \\ \hat{\omega}_M^T \hat{\omega}_M \hat{R} \end{bmatrix}$$

**Joint State Estimators**

- Can we filter IMU-based velocities further using a process model? $\rightarrow$ Fuse joint accelerations with sensor data to smooth IMU-based joint velocities without significant delay.

**Gyroscope Bias Estimator**

- Consider effect of noise sources on joint velocity computation:
  $$\dot{\theta}_{ij}(\theta) = \ddot{\omega} - b - w$$
- Choose state $[\theta^T, \dot{\theta}^T]$, dynamics are
  $$\dot{\theta} = \dot{\theta}_{ij}(\theta) \cdot (\ddot{\omega} - b - w)$$
- Measure joint positions:
  $$y = [1 \ 0 \ \theta \ b \ \nu]$$

**References**