AMRITA VISHWA VIDYAPEETHAM

TELESCOPE CONTROL SYSTEM

K. Nitya -CB.AI.U4AIM24123
B. Pavani Shreeya -CB.AI.U4AIM24106
Ardhra Vinod -CB.AI.U4AIM24105
Hari krishna Iyer -CB.AI.U4AIM24114

Data Acquisition:

- Imported JPL horizons from Astroquery python package.
- Imported SkyCoord (co-ordinates)from astropy to retrive positions of celestial objects.

DSA Implementation:

- Used arrays to store angles, angular velocities and feedforward terms.
- NumPy arrays is used to handle state space representation, matrix multiplications, simulation data and control input efficiently.

LQR ALGORITHM

Linear Quadratic Regulator (LQR) is a feedback control algorithm that utilizes all state variables for optimal control design, where each state variable is multiplied by a gain and summed to generate a single actuation value

ALGORITHM

-> **System Dynamics Modeling**

- State Space Representation:

$$\dot{x} = Ax + Bu, \qquad x = [\theta, \dot{\theta}, \phi, \dot{\phi}]^T$$

A: Defines how state variables change over time.
B: Control input matrix for torque application.

-> **Optimal LQR Control**

- computes u to minimize:

$$u = -kx \qquad J = \int_0^{\infty} (x^T Qx + u^T Ru)dt$$

Q: State weighting matrix (prioritizes angular precision).
R: Control effort weighting matrix.

Algebraic Riccati Equation (ARE):

- Solve $A^T P + PA - PBR^{-1} B^T P + Q = 0$

P = Solution matrix used to compute optimal control gain $K = R^{-1} B^T P$

-> **Feedforward Compensation**

- Velocity Feedforward: Enhance tracking by compensating for target motion:

$$ff_{\theta} = \frac{\Delta\theta\ target}{\Delta t}, \quad ff_{\varnothing} = \frac{\Delta\varnothing\ target}{\Delta t}$$

- Integrated into control law:

$$u = -k(x - x_{target}) + ff_{u}$$

-> **Real-Time Target Tracking Simulation:**

- Celestial Position Updates:

  Fetch real-time RA/Dec coordinates of target (e.g., Sirius) using astroquery.

- Time-Delay Simulation:

  Introduce latency with time.sleep() to mimic real-world operation.

-> **Numerical Integration of Dynamics**

Runge-Kutta 4th Order (RK4):

- Propagate system states over time:

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

- Solves nonlinear dynamics with time-varying feedforward terms.

-> **Performance Metric**

Root Mean Square Error (RMSE):

- Quantify tracking accuracy:

$$RMSE = \sqrt{\frac{1}{N}\sum(\theta_{target} - \theta_{actual})^2}$$

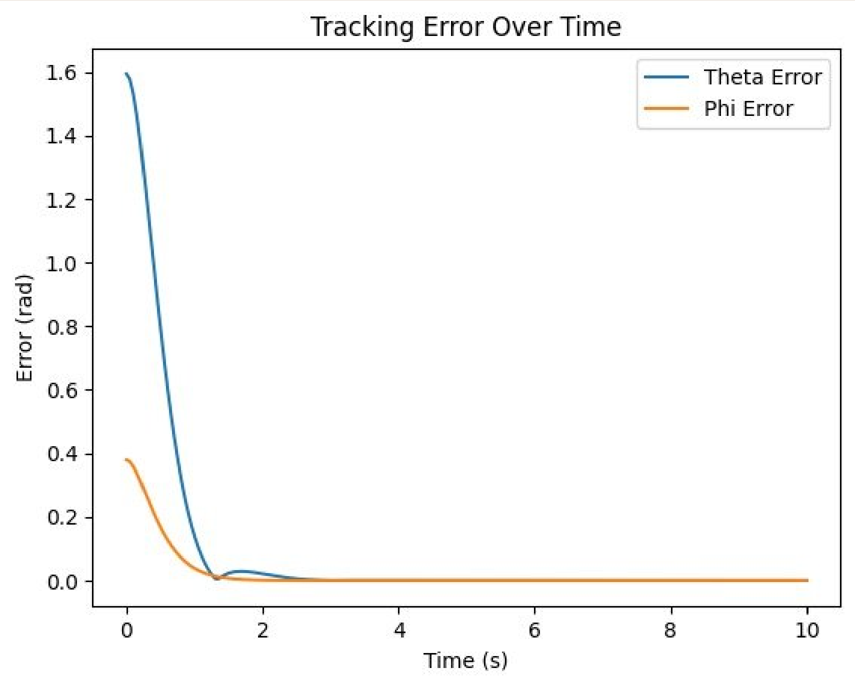
Tracking Error Over Time:

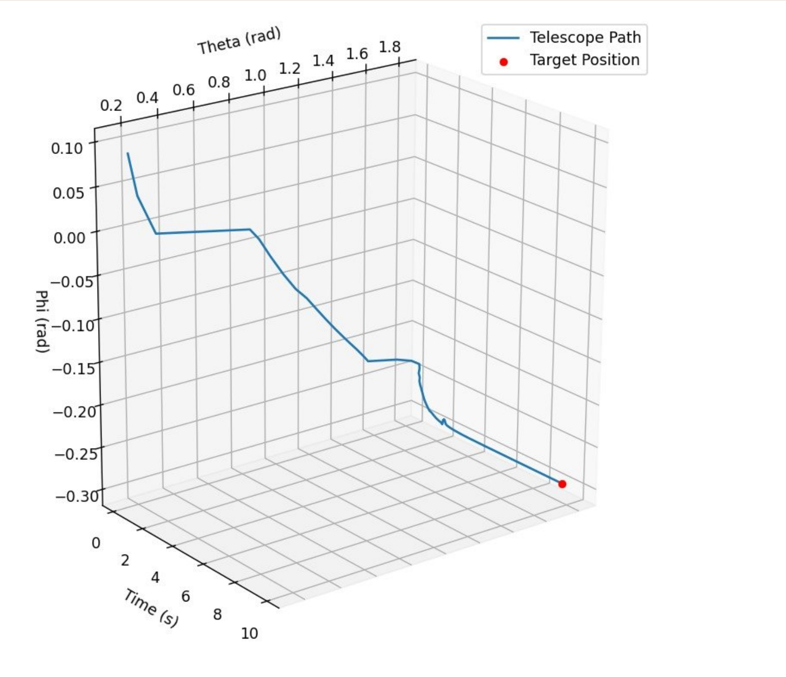
- The error in theta (blue) and phi (orange) decreases rapidly and stabilizes close to zero.

Telescope Path Over Time:

- The telescope starts from an initial state and gradually corrects its angles toward the target.

Observation: The LQR control effectively stabilizes the telescope and brings it to the desired target with minimal oscillations.





Thank You