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

PROBLEM STATEMENT

Modern telescope systems require precise pointing accuracy to track celestial objects efficiently. This work introduces an interactive simulation setup that controls theory, real-time visualization, and user interaction to address these challenges.

Data Acquisition:

- Imported SkyCoord (co-ordinates), EarthLocation and Time from astropy to retrive positions of celestial objects.

DSA Implementation:

- Used arrays to store angles, angular velocities and feedforward terms.

- Dynamic arrays are used to handle state space representation, matrix multiplications, simulation data and control input efficiently.

LITERATURE REVIEW

Sl. no.	**Title**	Year	**Methodology**	**Key Contributions**
1.	A Systematic Methodology for Modeling and Attitude Control of Multibody Space Telescopes	2024	Kane's method for multibody modeling, Linear Quadratic Regulator (LQR) design.	Developed a systematic method for modelling multibody space telescope.
2.	Execution of Queue-Scheduled Observations with the Gemini 8m Telescopes	1997	Queue scheduling, dynamic simulation models.	Proposed an efficient queue-based scheduling model for telescope operations.

3	Pointing Accuracy Improvements for the South Pole Telescope with Machine Learning	**2024**	Two XGBoost models were trained using historical telescope observation data, and weather conditions to predict real-time corrections.	Achieved a 33% reduction in pointing errors, enabling more precise observations.
4	Model Predictive Star Tracking Control for Ground-Based Telescopes: The Telescopio Nazionale Galileo Case	**2024**	A two-layer MPC-based tracking system is designed, to track astronomical targets while mitigating disturbances.	Achieved superior tracking accuracy, robustness to wind disturbances, and improved performance over traditional PID and LQG-PI controllers, validated on the Telescopio Nazionale Galileo (TNG).
5	Model Control Law by William Brogan	**1990**	Centers on designing optimal control laws by minimizing a quadratic cost function that balances state regulation (matrix Q) and control effort (matrix R) We solve the Algebraic Riccati Equation to compute a feedback gain matrix K, ensuring asymptotic stability for stabilizable systems.	used for its systematic trade-off tuning and inherent robustness margins
6	Modern Control Engineering by Katsuhiko Ogata	**1970**	The optimal feedback gain is computed by solving the algebraic Riccati equation, resulting in a control law that ensures system stability and a systematic trade-off between performance and energy use.	highlights LQR’s practicality for modern control system design, especially for multi-variable and state-space models

7	Modern Control of Dynamic Systems by Gene F. Franklin J David Powell	1986	The approach formulates a quadratic cost function that penalizes both deviations in system states and excessive control effort, leading to a control law where the optimal gain matrix is found by solving the algebraic Riccati equation.	This ensures a balance between performance and energy use, providing robust and stable closed-loop behavior.
8	Modern Control Systems by Richard C Dorf	1967	Minimizes a quadratic cost function reflecting both state deviations and control effort, weighted matrices Q and R. The optimal feedback gain is determined by solving the algebraic Riccati equation, resulting in a control law $u=-Kx$ that ensures system stability and a systematic trade-off between performance and actuator usage. Dorf emphasizes LQR's value for its mathematical rigor, robustness, and practical applicability to modern state-space control system design.	Dorf emphasizes LQR's value for its mathematical rigor, robustness, and practical applicability to modern state-space control system design.

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:

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

3D - VISUALIZATION

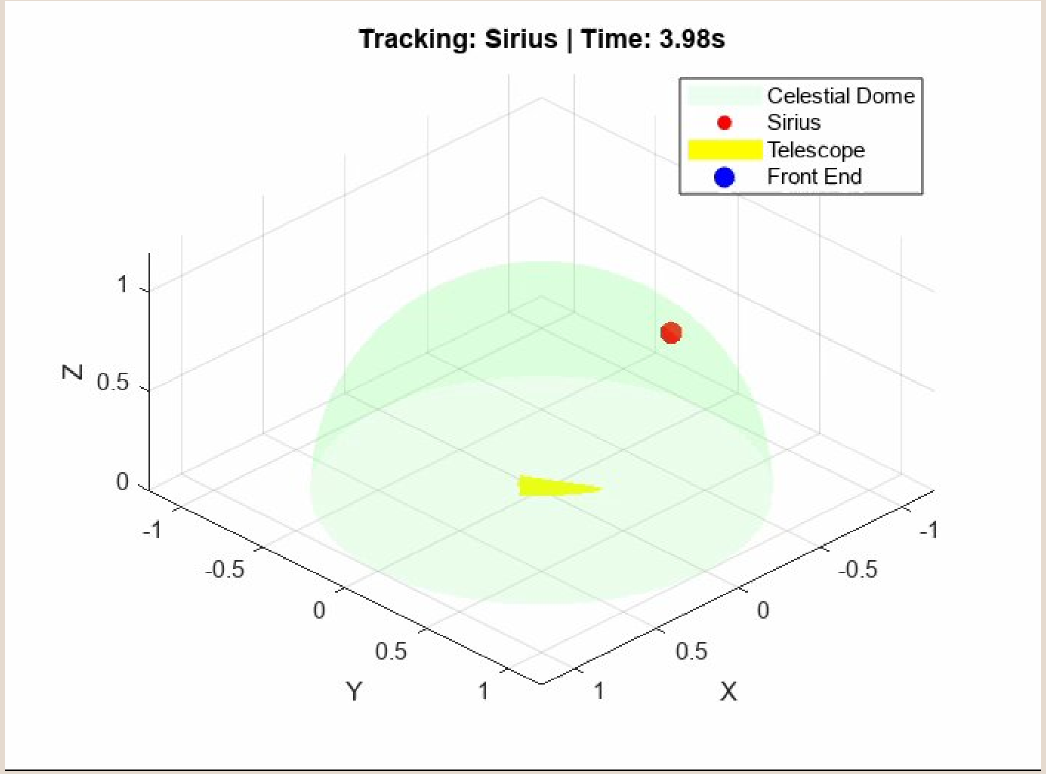
- A celestial dome representing the sky and a red dot at Siriu's position in 3D Cartesian coordinates is the Traget. The telescope model is shown as a cylinder oriented using the simulated angles.
- Spherical Linear Interpolation: A technique used in computer graphics to smoothly interpolate between two rotations or orientations. It's particularly useful for animating 3D rotations, ensuring a smooth and constant-speed motion along the shortest path between two rotations.
- Rodrigues' Rotation Formula: This computes a 3D rotation matrix which rotates the telescope in the target direction. When the initial direction vector, the target direction vector, the rotation axis k, and the angle θ are given, the rotation matrix R is calculated as
    - $R = I + sin\theta\, K + (1 - cos\theta)\, K^2$

  where, K is the skew-symmetric matrix of cross product vector k.

RESULTS

Observation:

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



Thank You