Date: 24th December 2025 IIT Dharwad

* SEIKO Implementation Workplan *

Task: Picking & Stacking of objects using SEIKO algorithm.

Bots: 2 Interbotix vx300s

Tech Stack: Python, qp solvers (quadratic programming)
pinnochio (Rigid Body Kinematics)
rclpy (ROS 2 Communication)
numpy (Calculation for matrix operations)

Optimization problem:

$$\min_{\Delta q} \frac{1}{2} \Delta q^T H \Delta q + g^T \Delta q$$

Subject to $q_{min} \leq q_{current} + \Delta q \leq q_{max}$ (Joint limits)

$$-\dot{q}_{max} \Delta t \leq \Delta q \leq \dot{q}_{max} \Delta t$$

parameters: $\Delta q \rightarrow$ change in joint angles

$H \rightarrow$ Hessian matrix $(H = J^T \cdot J + \lambda I)$

$J \rightarrow$ Jacobian matrix

$I \rightarrow$ Identity matrix

$\lambda \rightarrow$ Smoothness factor

$g \rightarrow$ gradient

'λ' decides smoothness of motion.

* Implementation *

Step. 1 → Creation of file handling physical math for calculation of Jacobian, manipulability index & forward Kinematics (FK)
→ Should return values of the joint limits.

Step. 2 → Defining a file class (SeikoSolver).
Input: Current state (q_c)
Target position (x_{target})
Jacobian (J)

Compute error between current pose & target pose. Let it be v_i

Construct matrix $H = J^T \cdot J + \lambda$
Construct gradient $g = -J^T v_i$
Defining inequality constraints based on joint limits
Ce.g: If approaching singularity, Δq must stop from hitting it).

Solve QP :- use qpsolverse. This will give Δq.

expected Output :- Should give next joint position $q_{next} = q_{current} + \Delta q$
(Note: leftsolver for leftarm & rightsolver for rightarm to Synchronise bimanual control).

Step 3 : Creation of ROS2 node

→ Creating a seperate file to subscribe to the joint states of Vx300s hardware joints.
→ Loading the urdf file (Robotmodel) & SeikoSolver from Step 2.

expected workflow: The Subscriber (ROS2 node) must listen to Vx300s joint states to get the real hardware's Current q.

- Get current q
- Define ~~target~~ pose (Coordinates of object)
- Run SeikoSolver (Step 2)
- Publish nent set of q to the Vx300s joint states.

execute Step 3 in loop (50Hz since all scripts ran at 50Hz)

— xx— xx —xx —