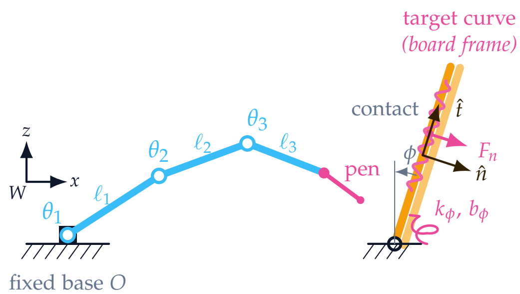
Chopstick Crane

JRF Selection Task — INTERFACE lab, IIT Madras

A planar 3-link arm holds a pen and must *trace a prescribed curve* onto a board. The catch: the board is not fixed. It is mounted on a single passive, spring-loaded hinge, so its tilt *responds* to where and how hard the pen presses. You must hit the curve in the board's *moving* frame while keeping the contact force inside a target band. This is the entire difficulty: the target lives in a frame that moves because of your own actions. World-frame IK will not work.



The arm ($\theta_1, \theta_2, \theta_3$; links ℓ_1, ℓ_2, ℓ_3) traces a curve in x–z plane onto a board pivoting about a spring–damped hinge (k_ϕ, b_ϕ). The tilt ϕ depends on contact force F_n, so the curve — fixed in the board frame ($\hat{t}, \hat{n}$) — moves in the world frame W as you draw it.

Problem statement

Build a planar 3-DOF arm and a passively tilting, spring-loaded board in MuJoCo. Everything lives in one vertical plane: the three revolute joints rotate about axes normal to the page, the pen tip moves in the x–z plane, and the board is seen edge-on as a line that pivots about a spring–damped hinge. The task is *not* to draw a 2-D shape — it is to drive the contact point along the board to a prescribed trajectory while regulating how hard the pen presses.

Concretely, the pen must track a supplied contact-point trajectory $u(s)$, $s \in [0, 1]$, measured *along the board surface* ($\hat{t}$, in the board's instantaneous frame), while holding the contact normal force F_n inside a target band $[F_{\min}, F_{\max}]$. The surface-normal offset v (penetration along $\hat{n}$) is held near zero so the pen stays in contact. As you press, the board tilts by $\phi(F_n)$ — so the line you are tracking along keeps rotating in the plane and your IK target moves every step. Compose the board's forward kinematics into the target each timestep; exploit the arm's redundancy (3 joints for a 1-D along-surface target + 1 force target) through a small optimization rather than a closed-form solve.

Profiles to track (pick one; u is contact position along $\hat{t}$, $F_n^\star$ the desired normal force — both functions of s):

- *Static hold:* $u(s) = u_0$ constant, $F_n^\star$ in band. The warm-up — pure contact-holding while the board settles.
- *Sweep:* $u(s) = u_0 + (u_1 - u_0)\,s$ at constant $F_n^\star$. Slide the contact up the board while the tilt drifts; tests moving-frame tracking.
- *Sinusoidal sweep:* $u(s) = u_0 + A\sin(2\pi s)$. Tests tracking of a smooth back-and-forth contact profile under shifting tilt.

Recommended environment

- **MuJoCo** (latest), driven from **Python** via the official `mujoco` bindings. Author the model yourself in **MJCF**.
- `numpy` for the kinematics and the IK/optimization; `matplotlib` for the result plots.
- Python `3.10+`. Any optimizer is fine (`scipy.optimize`, or a hand-rolled damped-least-squares / Gauss–Newton step).
- *Not* permitted as a black box: full IK stacks (`MoveIt`, `ikpy` one-liners) or stock arm environments — they will not handle the moving-frame coupling, and we will ask about your derivation. Cite anything you reuse.

Deliverables

- A **2-page report** covering: *(i)* problem formulation (frames, notation, objective, force constraint, board coupling); *(ii)* forward kinematics — derive the transform chain and the Jacobian; *(iii)* your inverse-kinematics / optimization method; *(iv)* how the board tilt enters the target each step; *(v)* results — tracking-error and contact-force plots, failure modes (loss of contact, board oscillation, singularities), and one thing you'd improve.
- A **30–60 s video** of the simulation clearly showing the pen tracing the curve while the board visibly tilts.
- **Runnable code** with a short README: dependencies and how to run.

What we look for: a correct FK derivation, a *justified* IK choice, clear writing, and honest discussion of what didn't work. The task is open-ended, so we are not looking for a particular error threshold or optimization method.