Homework Assignment 4 – 600.455/655 Fall 2025

Name Justin Wang	Name Alan You
Email jwang519@jhu.edu	Email ayou1@jhu.edu
Other contact information (optional)	Other contact information (optional)
Signature (required) I/We have followed the rules in completing this assignment JW	Signature (required) I/We have followed the rules in completing this assignment AY

1.Rememberthatthisisagradedhomeworkassignment.Itisthefunctionalequivalentofatake-homeexam.

2.Youaretowork**alone**orin**teams of two**andarenottodiscusstheproblemswithanyoneotherthan the TAs or the instructor.

3.**IMPORTANT NOTE:** Ifyouworkinteamsoftwo,youare**not**tosplitupthequestionsandeachanswer a subsetindividually.Youaretowork**together**~~.Iencour~~ageteamingontheseproblemsbecauseIbelieve thatitencourageslearning,notasawaytoreducetherequiredworkforstudentstakingthecourse. By signingthissheetyouareassertingthateachofyouhascontributedequallytoeachanswerand can individuallyexplaintheansweraswellasifyouhadansweredthequestionalone.Iviewthisasaquestion of trustandethics.

4. It is otherwiseopenbook,notes,andweb.Butyoushouldciteanyreferencesyouconsult.

5. Please refertothecourseorganizationalnotesforafullerlistingofalltherules.Iamnotrecitingthem all here, but theyarestillineffect.

6. Unless I sayotherwiseinclass,itisduebeforethestartofclassontheduedatepostedontheweb.

7. Submit theassignmentonGradeScopeasaneatandlegiblePDFfile.Wewillnotinsistontypesetting your answers, butwemustbeabletoreadthem.Wewillnotgotoextraordinarylengthstodecipherwhat you write. If thegraderscannotmakeoutananswer,thescorewillbe0.

8. Sign and handinthispageasthefirstsheetofyourassignment.Ifyouworkwithapartner,thenyou both should signthesheet,butyoushouldonlysubmitonePDFfileforbothofyou,usingtheGradeScopeteaming feature. Indicateclearlywhoitisfrom.

This assignment has more than 100 points, but the most that will be applied to your grade is 100

HW 4

$\vec{q}$: all joints $\quad F_{RW}(\vec{q}) = [R_{RW}(\vec{q}), \vec{P}_{RW}(\vec{q})] \quad F_{RW}(q+\Delta q) = F_{RW}(\vec{q})\,\Delta F_{RW}(\vec{q}, \Delta\vec{q})$

wrist

$$\Delta F_{RW}(\vec{q}, \Delta\vec{q}) \cong \Delta F_{RW}(\overset{\text{twist}}{\vec{\eta}_{RW}}) = [I + Sk(\vec{\alpha}_{RW}), \vec{\varepsilon}_{RW}]$$

$$\vec{\eta}_{RW} = [\vec{\alpha}_{RW}, \vec{\varepsilon}_{RW}]^T = J_{RW}(\vec{q})\,\Delta\vec{q}$$

F_{WT}: wrist → tool $\quad P_{tool}$: tool shaft length

wrench $\xi_W = [\underset{\text{forces}}{\phi_W^T}, \underset{\text{torques}}{\tau_W^T}]^T$ in F_W

~~Robot state: $[q, \dot{q}, \xi_W]$~~ $\qquad S(t) = [\underset{\text{joint}}{\vec{q}}, \underset{\text{d joint}}{\dot{\vec{q}}}, \underset{\text{wrist position}}{F_W}, \underset{\text{wrist twist}}{J_W}, \underset{\text{wrist meas. wrench}}{\vec{\xi}_W}]$

1. Each loop of this algorithm is executed every Δt. However, the exact timing between each step is not constant due to computing times. This means that to maintain deterministic timing to avoid gaps or spikes in commands, which could be incompatible with low-level controllers, we must output the command immediately in step 1, not later on where timing becomes nondeterministic.

2. Mechanical constraints: $q_{min} \le q \le q_{max}$, $|\dot{q}| \le \dot{q}_{max}$, $|\ddot{q}| \le \ddot{q}_{max}$

To ensure these will not be violated by the next command output Δq_{cmd}, Check:

$$\boxed{\begin{gathered} q_{min} \le q + \Delta q_{cmd} \le q_{max} \\ \left|\tfrac{1}{\Delta t}(\Delta q_{cmd})\right| \le \dot{q}_{max} \\ \left|\tfrac{1}{\Delta t^2}\Delta q_{cmd} - \tfrac{1}{\Delta t}\dot{q}\right| \le \ddot{q}_{max} \end{gathered}}$$

$\ddot{q}_{cmd} \cong \dfrac{\Delta\dot{q}_{cmd} - \dot{q}}{\Delta t} = \dfrac{\Delta q_{cmd}/\Delta t - \dot{q}}{\Delta t} = \dfrac{\Delta q_{cmd}}{\Delta t^2} - \dfrac{\dot{q}}{\Delta t}$

given robot state at time t and a known Δt

3. want $\vec{\eta}_{RT}(\vec{q}, \Delta q) = [\vec{\alpha}_{RT}, \vec{\varepsilon}_{RT}]$ and $J_{RT}(q, \Delta q)$

$F_{RT}(q) = F_{RW}(q)F_{WT} \qquad F_{RW}(q+\Delta q) = F_{RW}(q)\,\Delta F_{RW}(\vec{\eta}_{RW})$

First find J_{WT} s.t. $\eta_{RT} = J_{WT}\eta_{RW}$

$$F_{RT}(q+\Delta q) = F_{RW}(q+\Delta q)F_{WT} = F_{RW}(q)\,\Delta F_{RW}(\eta_{RW})F_{WT}$$

$$F_{RT}(q+\Delta q) = F_{RT}(q)\,\Delta F_{RT}(\eta_{RT}) = F_{RW}(q)F_{WT}\,\Delta F_{RT}(\eta_{RT})$$

The same motions, must be related by some adjoint matrix

3. Now, we have

$$F_{RW}(q)\,\Delta F_{RW}(\eta_{RW})F_{WT} = F_{RW}(q)\,F_{WT}\,\Delta F_{RT}(\eta_{RT})$$

$$\Delta F_{RW}(\eta_{RW})F_{WT} = F_{WT}\,\Delta F_{RT}(\eta_{RT})$$

$$\Delta F_{RW}(\eta_{RW}) = F_{WT}\,\Delta F_{RT}(\eta_{RT})\,F_{WT}^{-1}$$

Rewrite

$$\eta = \begin{bmatrix} d \\ \varepsilon \end{bmatrix} \to \hat{\eta} = \begin{bmatrix} Sk(d) & \varepsilon \\ 0 & 0 \end{bmatrix} \text{ so now, } \Delta F = I + \hat{\eta}$$

$$I + \hat{\eta}_{RW} = F_{WT}(I + \hat{\eta}_{RT})F_{WT}^{-1}$$

$$\begin{bmatrix} R_{WT} & P_{WT} \\ 0 & 1 \end{bmatrix}\begin{bmatrix} Sk(d_{RT}) & \varepsilon_{RT} \\ 0 & 0 \end{bmatrix}\begin{bmatrix} R_{WT}^{-1} & -R_{WT}^{-1}P \\ 0 & 1 \end{bmatrix}$$

$$I + \hat{\eta}_{RW} = I + F_{WT}\,\hat{\eta}_{RT}\,F_{WT}^{-1}$$

$$= \begin{bmatrix} R_{WT}Sk(d_{RT}) & -RSk(d) + R\varepsilon \\ 0 & 0 \end{bmatrix}$$

definition of adjoint

$$\hat{\eta}_{RW} = F_{WT}\,\hat{\eta}_{RT}\,F_{WT}^{-1} \Rightarrow \eta_{RW} = Adj_{F_{WT}}\,\eta_{RT}$$

$$Adj_{F_{WT}} = \begin{bmatrix} R_{WT} & R_{WT}Sk(P_{WT}) \\ 0 & R_{WT} \end{bmatrix}$$

$$\eta_{RT} = Adj_{F_{WT}}^{-1}\,\eta_{RW}, \text{ so } J_{WT}(q) = Adj_{F_{WT}}^{-1}$$

we know $J_{RT} = J_{WT}\,J_{RW}(q)$

$$\boxed{J_{RT} = Adj_{F_{WT}}^{-1}\,J_{RW}(q)}$$

where $Adj_{F_{WT}} = \begin{bmatrix} R_{WT} & R_{WT}Sk(P_{WT}) \\ 0 & R_{WT} \end{bmatrix}$

4. we want $|\vec{\eta}_{RT}| \le \eta_{RT}^{max}$

Since $\eta_{RT} = Adj_{F_{WT}}^{-1}\,\eta_{RW}$, $\;|Adj_{F_{WT}}^{-1}\,\eta_{RW}| \le \eta_{RT}^{max}$

where $Adj_{F_{WT}}$ is defined above, and $F_{WT} = F_{RW}^{-1}F_{RT}$

6.

If the robot is deficient, we should use a least-squares objective function and terminate when residuals stop decreasing.

So

instead of $\eta^{cmd}_{RT} = J_{RT}(q)\,\Delta q_{cmd}$,

$$\Delta q_{cmd} = \underset{\Delta q_{cmd}}{\operatorname{argmin}} \left| \eta^{desired}_{RT} - J_{RT}(q)\,\Delta q_{cmd} \right|^2$$

and terminate when $\eta^{desired}_{RT} - J_{RT}(q)\,\Delta q_{cmd}$ stops decreasing

7.

If the robot has redundant joints, we can add a term to the objective function to encourage joints to stay near their midpoint.

Let

$q_{mid} = \frac{1}{2}(q_{min} + q_{max})$

$w_{desired}, w_{mid}$ = weights, $w_{desired} >> w_{mid}$

Then we can do the same as before but objective function as:

$$\Delta q_{cmd} = \underset{\Delta q_{cmd}}{\operatorname{argmin}} \left(w_{desired} \left| \eta^{desired}_{RT} - J_{RT}(q)\,\Delta q_{cmd} \right|^2 + w_{mid} \left| (q + \Delta q_{cmd}) - q_{mid} \right|^2 \right)$$

Subject to same constraints as questions 2 and 4

8.

$$h\dot{\vec{\eta}}_{RW} = K\,\vec{\tau}_w/\Delta t = \begin{bmatrix} K^{rot}_{3\times3} & 0 \\ 0 & K^{vel}_{3\times3} \end{bmatrix} \begin{bmatrix} \vec{\tau}_w \\ \vec{\rho}_w \end{bmatrix} / \Delta t$$

$$\Delta F^{cmd}_{RW} = \left[I + Sk(d^{cmd}_{RW}),\ \epsilon^{cmd}_{RW} \right]$$

we want a Δq_{cmd} s.t.

$$J_{RW}(q)\,\Delta q_{cmd} = \eta^{cmd}_{RW}$$

So we want to optimize:

$$\Delta q_{cmd} = \underset{\Delta q_{cmd}}{\operatorname{argmin}} \left| \eta_{RW}^{cmd} - J_{RW}(q)\, \Delta q_{cmd} \right|^2$$

Subject to:

$$\boxed{q_{min} \le q + \Delta q \le q_{max} \quad \left|\tfrac{1}{\Delta t}\Delta q\right| \le \dot{q}_{max} \quad \left|\tfrac{1}{\Delta t^2}\Delta q - \tfrac{1}{\Delta t}\dot{q}\right| \le \ddot{q}_{max} \quad |\eta_{RW}| \le \eta_{RW}^{max}} \quad ☆$$

Pseudocode:

```
hand over hand (q, q-dot, J_RW, ξ_W, K_RW, Δt):
    η_RW_dot ← K_RW * ξ_W   // 6X1
    η_RW ← η_RW_dot * Δt    // 6X1

    Δq_cmd ← argmin  ( abs(η_RW - J_RW(q) * Δq_cmd))
             Δq_cmd
        subject to constraints in ☆
        using a quadratic programming solver
    Δq_cmd_dot ← Δq_cmd / Δt
    return Δq_cmd_dot
```

CIS HW4.9-13

Justin Wang, Alan You

November 2025

1 Question 9

To achieve constrained admittance, we need to modify step 3 of the algorithm for some cases. Observe that for admittance, we only care about the translation of the tip not the rotation, so we only care about the velocity admittance weights.
For each control cycle we compute the tool tip position in CT frame:

$$p_{CT}(t)$$

For each structure $X \in \{A, B, C\}$ we get its SDF and gradient:

$$d_X(t) = \text{GetMap}(p_{CT}(t), \text{“SDF\_X”}), \qquad g_X(t) = \nabla d_X(p_{CT}(t)) = \text{GetGrad}(p_{CT}(t), \text{SDF\_X})$$

Using Fig. 3, for each X we compute the distance–dependent virtual fixture admittance gains $\kappa_X^{(1)}(d_X)$ and $\kappa_X^{(2)}(d_X)$, leaving us with three possibilites we care about:

1. If $d_X(t) > d_X^{\text{thresh}}$, we use the unconstrained admittance from Question 8 (Step 3 is unmodified)
2. If $0 \leq d_X(t) \leq d_X^{\text{thresh}}$, we interpolate $\kappa_X^{(1)}(d_X)$ and $\kappa_X^{(2)}(d_X)$ according to Fig. 3.
3. If $d_X(t) < 0$, we additionally require that the commanded translational motion points outward.

If we hit possibility number 2, we have to calculate a new admittance weight matrix. We only care about constraining motion normal to the bodies, not tangentially.
The outward SDF normal is:

$$\mathbf{n}_X(t) = \mathbf{g}_X(t)/\|\mathbf{g}_X(t)\|$$

We define projection operators onto the normal direction and the tangent plane:

$$P_{\parallel}^{(X)}(t) = \mathbf{n}_X(t)\mathbf{n}_X^T(t), \qquad P_{\perp}^{(X)}(t) = I - P_{\parallel}^{(X)}(t).$$

Now that we have the tangential and normal directions, we can construct an effective admittance weight matrix for each of the bodies of interest ($X \in A, B, C$):

$$K_{\text{vel}}^{(X)}(t) = \kappa_X^{(1)}(d_X(t))\, P_{\parallel}^{(X)}(t) \; + \; \kappa_X^{(2)}(d_X(t))\, P_{\perp}^{(X)}(t).$$

Once we have $K_{\text{vel}}^{(A)}(t), K_{\text{vel}}^{(B)}(t), K_{\text{vel}}^{(C)}(t)$, we need to construct an effective admittance weight matrix because we need to consider the distance of the tip relative to each body concurrently. This can be achieved by constructing an admittance weight matrix such that the translational motion is affected by the strongest "push-back" from each body. Observe that smaller weight value means stronger perceived "push-back". Also observe that we are only ever modifying the velocity weights never the rotation weights. We can construct:

$$K_{\text{vel}}^{\text{eff}}(t) = \min\bigl(K_{\text{vel}}, K_{\text{vel}}^{(A)}(t), K_{\text{vel}}^{(B)}(t), K_{\text{vel}}^{(C)}(t)\bigr),$$

So now we can construct

$$K_{RW}^{\text{eff}}(t) = \begin{bmatrix} K_{\text{rot}} & 0 \\ 0 & K_{\text{vel}}^{\text{eff}}(t) \end{bmatrix}.$$

where K_{rot} is the same as form Question 8. If $K_{RW}^{\text{eff}}(t)$ is used, then the combined admittance law can now be expressed as:

$$\dot{\eta}_{RW}(t) = K_{RW}^{\text{eff}}(t)\, \zeta_W(t)/\Delta t, \qquad \eta_{RW}(t) = K_{RW}^{\text{eff}}(t)\, \zeta_W(t)$$

where $\eta_{RW}(t) = [\, \alpha_{RW}^T(t),\ \varepsilon_{RW}^T(t)\,]^T$. This can then be used in the same cost function as before:

$$\Delta\mathbf{q}_{\text{cmd}}(t) = \arg\min_{\Delta\mathbf{q}} \left\| \eta_{RW}(t) - J_{RW}(\mathbf{q}(t))\, \Delta\mathbf{q} \right\|^2,$$

If we hit possibility number 3, we still use $K_{RW}^{\text{eff}}(t)$ the same as calculated above, but now we have another constraint we have to follow while optimizing:

$$\dot{\mathbf{p}}_{CT}^{\text{cmd}} \cdot \nabla d_X > 0$$

If we use

$$\dot{\mathbf{p}}_{CT}^{\text{cmd}} = J_{p,CT}(\mathbf{q}(t))\, \Delta\mathbf{q}_{\text{cmd}}/\Delta t$$

We can rewrite it into this inequality

$$\nabla d_X\, J_{p,CT}(\mathbf{q}(t))\, \Delta\mathbf{q}_{\text{cmd}}(t) > 0$$

2 Question 10

To also constrain a portion of the tool shaft, we can modify Question 9 by running this algorithm not only for the tip of the tool, but also for sampled points going up as far along the tool shaft from the tip as we care about. Then when constructing $K_{RW}^{\text{eff}}(t)$, we must take the element-wise minimums of all the $K_{\text{vel}}^{(X)}(t), X \in (A, B, C)$ for each of the sampled points. So if we want to constrain the entire shaft of length ρ_{tool} we can run Step 3 described in Question 9 for a point sampled at every ρ_{tool}/n length along the shaft, where a good n would be 5 to 10.

3 Question 11

To ensure safety in this scenario, we can modify Question 9 by first incorporating all the modifications in Question 10 as well. Then, we can modify the calculations further by treating the areas as larger than they actually are to be safe. If there is some error in F_{RC},

$$F_{RC}^{*} = F_{RC}\,\Delta F_{RC}(\eta_{RC}), \qquad \text{cov}(\eta_{RC}) = C_{RC},$$

Let J_{RC}^{*} be the Jacobian that maps the effect of η_{RC} in F_{RC}^{*} to the resulting error in tool tip position calculation. This means the true tool tip in CT frame also has some error:

$$p_{CT}^{*} = p_{CT} + J_{RC}^{*}\,\eta_{RC}$$

Therefore the tool-tip position has covariance

$$C_{p_{CT}} = J_{RC}^{*}\,C_{RC}\,J_{RC}^{*T}$$

For each of the structures of interest $X \in \{A, B, C\}$ the true distance from each of the bodies to the tip now also has some error:

$$d_X^{*} = d_X + \nabla d_X\,(\mathbf{p}_{CT}^{*} - \mathbf{p}_{CT}) = d_X + \nabla d_X\,J_{RC}^{*}\,\eta_{RC}$$

So the variance of the distance error is

$$C_{d_X} = \nabla d_X\,C_p\,\nabla d_X^{T}$$

If we want to guarantee that the tool tip contacts any of the structures given this uncertainty, we replace the distance d_X used in Question 9 with a conservative "safe" distance for all $X \in (A.B.C)$ and each point along the tool shaft:

$$d_X^{\text{safe}} = d_X \;-\; k\,C_{d_X}$$

where k is a chosen safety factor like 4 or 5. This calculation should be done as soon as possible so that all following calculations that depend on distance use d_X^{safe}.

4 Question 12

If we have a distance sensor, we can use the Kalman filter. Let's first assume we are only concerned about the tool tip to make the explanation simpler. This can generalize to any point on the shaft very easily. We know the following:

$$F^*_{RC} = F_{RC}\,\Delta F_{RC}(\eta_{RC}), \qquad cov(\eta_{RC}) = C_{RC}$$

The twist η_{RC} represents uncertainty in F^*_{RC}. .

Let p_{CT} be nominal tool-tip position in CT coordinates, and let $d_{\min} = \text{GetMap}(p_{CT}, \text{“SDF\_X”})$ and $\nabla d_{min} = \text{GetGrad}(p_{CT}, \text{SDF\_X})$ for whatever X is associated with the smallest d_x. This is because it is safe to assume that whatever structure we think we are closest to based off of the preop CT scan, is most likely the structure that the distance sensor's measurement is based off of.

From Question 11,

$$p^*_{CT} = p_{CT} + J^*_{RC}\eta_{RC},$$
$$d^{true}_{\min} = d_{\min} + \nabla d_{min} J^*_{RC}\eta_{RC},$$

The sensor returns a measurement of the minimum distance to the nearest anatomic structure,

$$z = d^{true}_{min} + \delta \qquad \delta \sim \mathcal{N}(0, \sigma^2_\delta),$$

which can be used for cases when $d_X(p_{CT}) > d_{\text{thresh}}$ for all $X \in \{A, B, C, D\}$
We can now create:

$$z = d_{\min} + H\eta_{RC} + \delta, \qquad H = \nabla d^T_{min} J^*_{RC}$$

We can treat η_{RC} as the state we are trying to estimate using a Kalman filter. The measurement residual is

$$r = z - \dot{d}_{\min}$$

So the Kalman gain is

$$K = C^{prior}_{RC} H^T (H C^{prior}_{RC} H^T + \sigma^2_\delta)^{-1}$$

So then we can update $\eta_{\hat{RC}}$ and C_{RC}:

$$\hat{\eta}^{post}_{RC} = \hat{\eta}^{prior}_{RC} + Kr, \qquad C^{post}_{RC} = (I - KH)\,C^{prior}_{RC}.$$

Now we can also calculate an updated F_{RC}:

$$F^{post}_{RC} = F^{prior}_{RC}\,\Delta F_{RC}(\hat{\eta}^{post}_{RC}),$$

We can continue updating F_{RC} using this algorithm in each cycle.

5 Question 13

In this case, let's first assume we only care about the point at the tool tip. Points along the tool shaft can be generalized to easily based off of Question 10. The high level is that we now want admittances such that it is still very hard to move the tool tip into areas A, B, and C, but it is free to move within D, as well as to and from D and E but no other transition. To do this, we can add one high level condition to check first:

1. If the tool tip is inside of E and outside of D (and all other structures), proceed as normal.

2. If the tool tip is inside of D and outside E, still proceed as in Question 9 but instead of only finding K_{vel}^{X} for $X \in (A, B, C)$, also do it for D as well. Then, instead of calculating a twist to move the tool outwardly normal to structure D, instead calculate a twist to move the tool inwardly normal to structure D (flip the sign on its SDF gradient). However, if the tool is moving towards structure E, then do not move the tool inwardly normal to structure D.

References

[1] Johns Hopkins University, *"Courses: 455–655: 2025: Fall 2025 Schedule – CIIS Wiki"*. Accessed Nov. 18, 2025. Available: https://ciis.lcsr.jhu.edu/doku.php?id=courses:455-655:2025:fall-2025-schedule

[2] T. Lacey, *"Tutorial: The Kalman Filter"*. MIT. Available: https://web.mit.edu/kirtley/kirtley/binlustuff/literature/control/Kalman

[3] Wikipedia contributors, *"Screw theory"*, *Wikipedia, The Free Encyclopedia*. Accessed Apr. 18, 2023. Available: https://en.wikipedia.org/wiki/Screw$_{t}heory$

[4] T. DeWolf, *"Robot Control Part 2: Jacobians, Velocity, and Force"*. 2013. Available: https://studywolf.wordpress.com/2013/09/02/robot-control-jacobians-velocity-and-force/