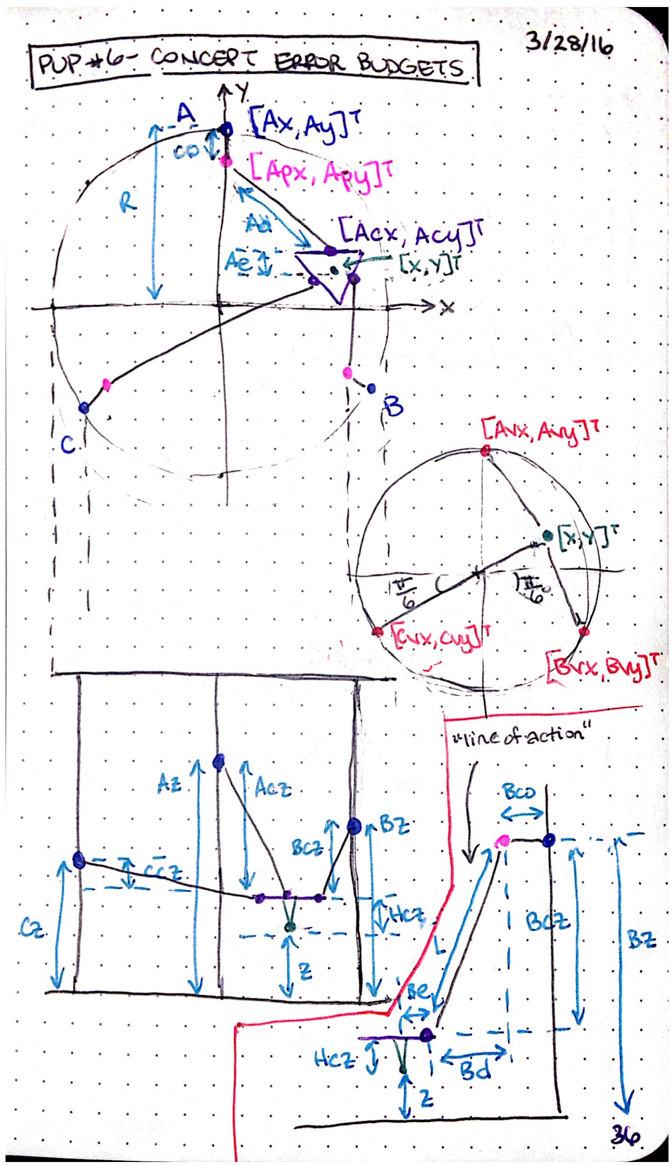
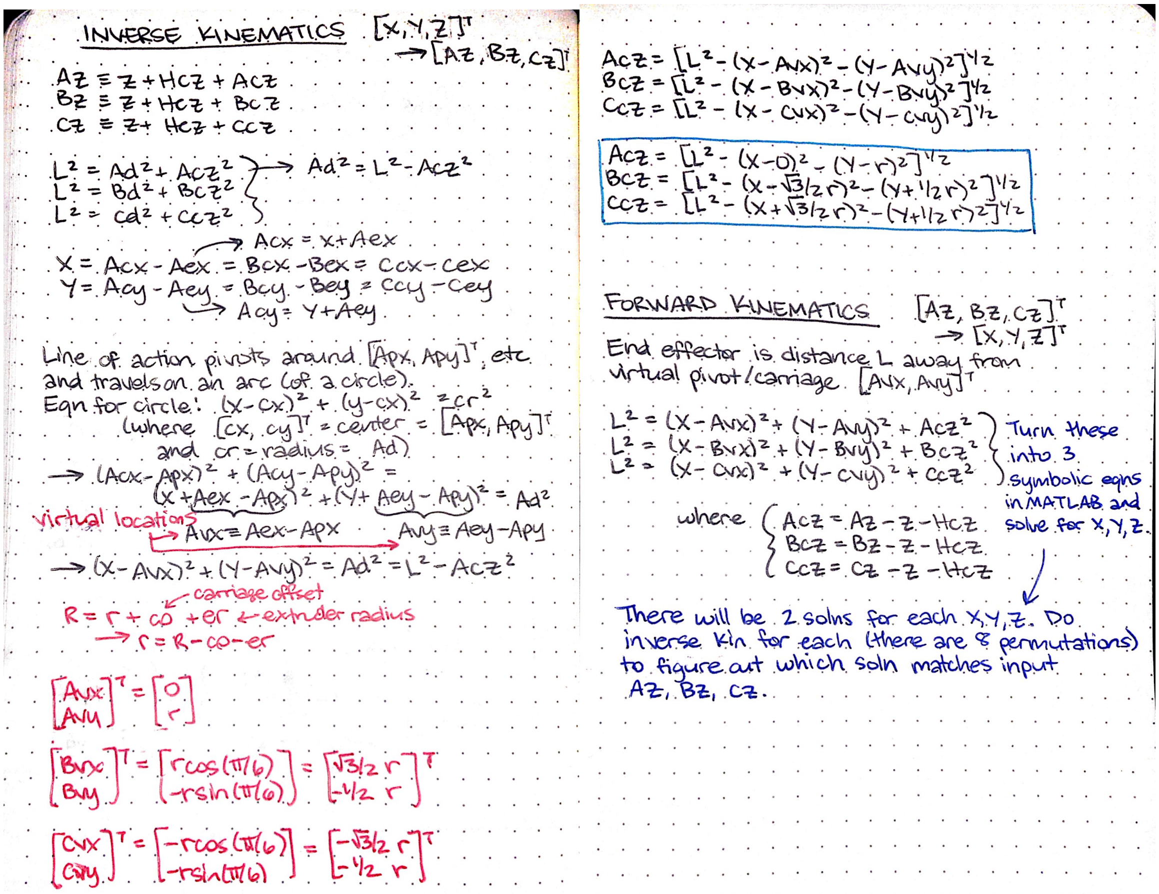
2.744 PUP #6: Concept Error Budget

I spent most of this PUP working on figuring out the forward and inverse kinematics of my delta linear 3d printer, and calculating/plotting/verifying it in MATLAB (I tried several approaches to modeling the kinematics, eventually settling on a method similar to that described in a [document by Steve Graves](#)). This is essential for determining geometric errors and help in error budgeting. It also helps me visualize the workspace and experiment with different part dimensions and trajectories, for example in [this video](https://www.youtube.com/watch?v=zfodeSEOl1k) (https://www.youtube.com/watch?v=zfodeSEOl1k) I made to simulate drawing a circle. In the video, you can see even though the circle is constrained to one plane, all three carriages move vertically in a fairly complicated "dance."

Below is a drawing showing different views (birds-eye and straight on) of the robot, with dimensions and coordinates added in. The dimensions in red (Avx, Avy, etc.) are virtual carriage locations that make the rods meet at the actual extruder position (instead of at the edge of the extruder platform). This makes the math simpler.



Forward and inverse kinematics calculations are in my notebook below. **Inverse kinematics** are simpler, because an input of desired $[X, Y, Z]^T$ produces only one possible solution of carriage heights $[Az, Bz, Cz]^T$. Thus, the computation time is shorter. This is fortunate, because inverse kinematics will be the direction that the printer will need to calculate to traverse a particular trajectory. It is still necessary to compute the **forward kinematics** for geometric error budgeting. Although the equations for forward kinematics are simple, they must be solved as a system in MATLAB. Solving for X, Y, and Z produces two results for each variable. We then have to run each XYZ permutation (there are eight permutations) into the inverse kinematics equations to figure out which one procures the original inputted carriage heights $[Az, Bz, Cz]^T$. My code can be found here.



INVERSE KINEMATICS $[X, Y, Z]^T \rightarrow [Az, Bz, Cz]^T$

$$Az \equiv Z + Hcz + Acz$$
$$Bz \equiv Z + Hcz + Bcz$$
$$Cz \equiv Z + Hcz + Ccz$$

$$\left.\begin{aligned} L^2 &= Ad^2 + Acz^2 \\ L^2 &= Bd^2 + Bcz^2 \\ L^2 &= Cd^2 + Ccz^2 \end{aligned}\right\} \rightarrow Ad^2 = L^2 - Acz^2$$

$$X = Acx - Aex = Bcx - Bex = Ccx - Cex \rightarrow Acx = X + Aex$$
$$Y = Acy - Aey = Bcy - Bey = Ccy - Cey \rightarrow Acy = Y + Aey$$

Line of action pivots around $[Apx, Apy]^T$, etc and travels on an arc (of a circle).
Eqn for circle: $(x-cx)^2 + (y-cx)^2 = cr^2$
(where $[cx, cy]^T$ = center = $[Apx, Apy]^T$ and cr = radius = Ad)

$$\rightarrow (Acx - Apx)^2 + (Acy - Apy)^2 = (X + \underbrace{Aex - Apx}) ^2 + (Y + \underbrace{Aey - Apy})^2 = Ad^2$$

virtual locations $Avx \equiv Aex - Apx$ $Avy \equiv Aey - Apy$

$$\rightarrow (X - Avx)^2 + (Y - Avy)^2 = Ad^2 = L^2 - Acz^2$$

$R = r + co + er$ (co: carriage offset; er: extruder radius)
$\rightarrow r = R - co - er$

$$\begin{bmatrix} Avx \\ Avy \end{bmatrix}^T = \begin{bmatrix} 0 \\ r \end{bmatrix}$$

$$\begin{bmatrix} Bvx \\ Bvy \end{bmatrix}^T = \begin{bmatrix} r\cos(\pi/6) \\ -r\sin(\pi/6) \end{bmatrix} = \begin{bmatrix} \sqrt{3}/2\, r \\ -1/2\, r \end{bmatrix}^T$$

$$\begin{bmatrix} Cvx \\ Cvy \end{bmatrix}^T = \begin{bmatrix} -r\cos(\pi/6) \\ -r\sin(\pi/6) \end{bmatrix} = \begin{bmatrix} -\sqrt{3}/2\, r \\ -1/2\, r \end{bmatrix}^T$$

$$Acz = [L^2 - (X - Avx)^2 - (Y - Avy)^2]^{1/2}$$
$$Bcz = [L^2 - (X - Bvx)^2 - (Y - Bvy)^2]^{1/2}$$
$$Ccz = [L^2 - (X - Cvx)^2 - (Y - Cvy)^2]^{1/2}$$

$$\boxed{\begin{aligned} Acz &= [L^2 - (X-0)^2 - (Y-r)^2]^{1/2} \\ Bcz &= [L^2 - (X - \sqrt{3}/2\, r)^2 - (Y + 1/2\, r)^2]^{1/2} \\ Ccz &= [L^2 - (X + \sqrt{3}/2\, r)^2 - (Y + 1/2\, r)^2]^{1/2} \end{aligned}}$$

FORWARD KINEMATICS $[Az, Bz, Cz]^T \rightarrow [X, Y, Z]^T$

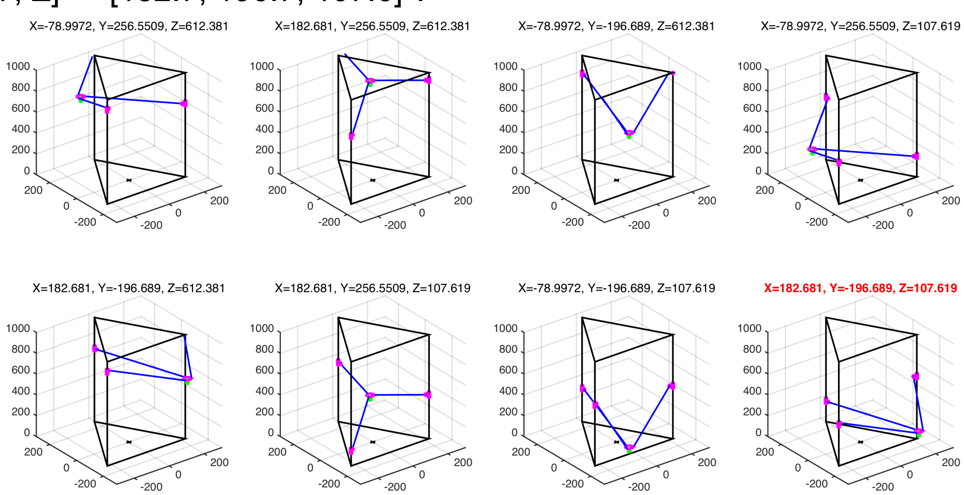
End effector is distance L away from virtual pivot/carriage $[Avx, Avy]^T$

$$\left.\begin{aligned} L^2 &= (X - Avx)^2 + (Y - Avy)^2 + Acz^2 \\ L^2 &= (X - Bvx)^2 + (Y - Bvy)^2 + Bcz^2 \\ L^2 &= (X - Cvx)^2 + (Y - Cvy)^2 + Ccz^2 \end{aligned}\right\}$$ Turn these into 3 symbolic eqns in MATLAB and solve for X, Y, Z.

where $\begin{cases} Acz = Az - Z - Hcz \\ Bcz = Bz - Z - Hcz \\ Ccz = Cz - Z - Hcz \end{cases}$

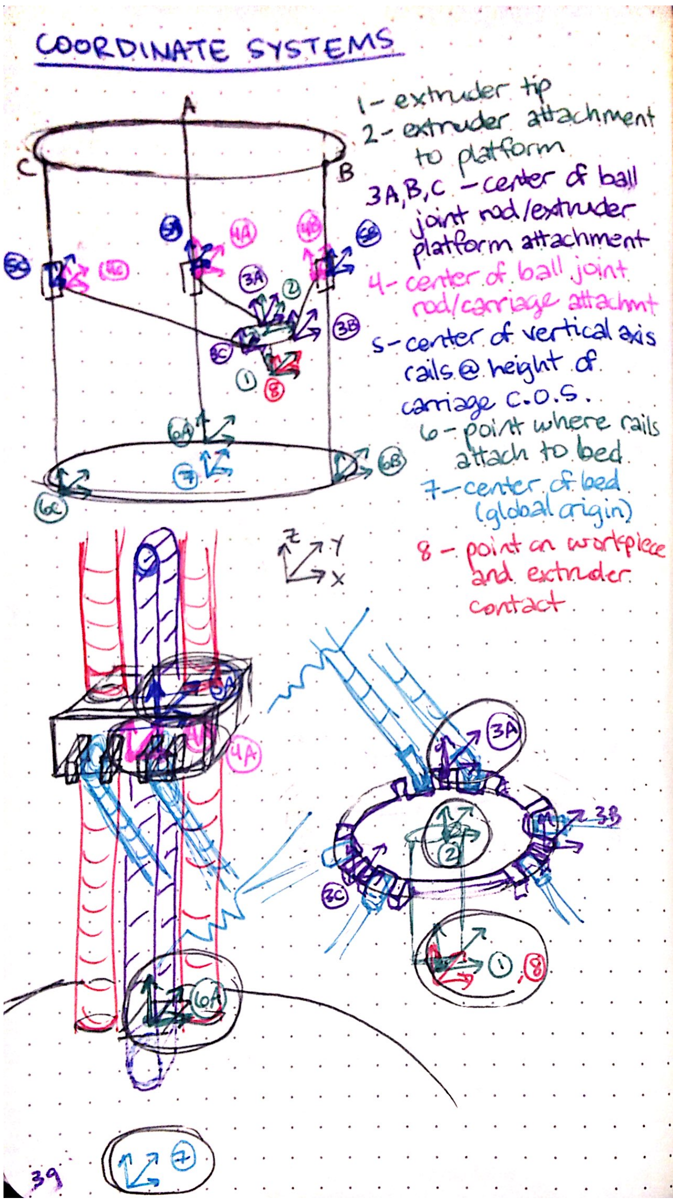
There will be 2 solns for each X, Y, Z. Do inverse Kin for each (there are 8 permutations) to figure out which soln matches input Az, Bz, Cz.

Below is an example of the 8 permutations of possible extruder locations based on a forward input of $[Az, Bz, Cz]^T = [200, 400, 600]^T$. The correct result is shown in red on the bottom right, with coordinates $[X, Y, Z]^T = [182.7, 196.7, 107.6]^T$.



Create first order error budgets for your favorite design.

Below is a schematic of my coordinate systems and their POI and POA locations, stiffnesses, and properties. I modeled the coordinate systems somewhat similar to the bridge machine in the example document. As opposed to my coordinate system assignment in PUP5 that used HTMS, all of these CS's have the same orientation and only differ in translation. CS 3-6 each have three sub-CS's for each of the vertical axes. However, in my error budgeting I chose to simplify the machine by only modeling one axis (the "A" rail, since it is located on the Y-axis).



SUMMARY OF COORDINATE POA/POI

ORIGIN [attachment stiffness to next CS's POI] POI [compliance CS's POI to its c.o.s.]

CS1 extruder tip
K [∞ b/c it's just a point] C [0]

CS2 extruder attachment to platform
K [nuts/bolts/screws of extruder to platform (LARGE, → ∞)] C [$\frac{1}{K_{extruder}}$]
model as cantilever beam, steel, 40mm, 16mm, F
Estimate forces due to acceleration (PUP 3)

CS3 center of ball jt/extr. platform attachment
K [screwed joints, ball jt translation, ball joint rot] C [extr. pltfrm]
Model as plate (wide thin beam) buckling

CS4 center of ball jt rod/carriage attachment
K [same] C [2 rods (~0)] buckling rods
(But rods will not buckle, b/c compliance of carriage axis >> c(rods))

CS5 center of vertical axis rails @ height of carriage c.o.s.
K [linear bearings to rails, vertical motion] C [small distance, very stiff → c≈0]

40

CS7 center of bed
K [large (can be clamped to table if needed)] C [buckling plate]

CS8 point on workpiece where extruder touches
K [large, encompassed by c] → C [plastic ABS part]

The next series of notebook pages goes over the changes I made in the spreadsheet, along with reasoning if necessary. I created a reference sheet with relevant calculations (mass, density, lengths, etc) and coordinates/values outputted from my MATLAB kinematics scripts.

ERROR BUDGET SPREADSHEET
(FOR RAIL (A))

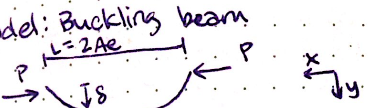
CS1
Coords of POI: [0, 0, 0]
Coords of this CS in next CS: [0, 0, −Hcz]
Applied loads: zeros
attach pts: 1
Translational error: all 0.00

CS2
Cd of this CS in nxt CS: [0, −Ae, 0]
Applied loads: F_z = − extruder weight
attach pts: 1 $F_x = m a_x = 8N$
Translational error: all 0.00
Model: cantilever beam

CS3
Cd of this CS in nxt CS: [0, −223, −193]
CALCULATE IN MATLAB
For [X, Y, Z] = [0, 0, 100]
Applied loads: assume none
Compliance: assume 0

Forces so low w/ stiffness so high that any compression/buckling of extruder platform is negligible

Model: Buckling beam



$$P_{yield} = \frac{\pi^2 EI}{L^2} = \frac{\pi^2 E w t^3}{3L^2} = \frac{\pi^2 E A t^2}{3L^2}$$

$$\sigma_y = \frac{P_y}{A} = \frac{\pi^2 E t^2}{3L^2}$$

check that $\sigma_y \ll \sigma_{y(ABS)}$
→ Don't need to worry abt buckl

Model: compression

$$E = \frac{\sigma}{\varepsilon} = \frac{\sigma}{(\delta/L)} = \frac{PwtL}{\delta} \rightarrow \delta = \frac{PwtL}{E}$$

$$\frac{P}{\delta} = k = \frac{E}{wtL}$$

(For now, assume $k = \infty$, $c = 0$).
Rod end stiffness not specified by mnfr

Attachment point locations: [±10, 0, 0]
Linear stiffness ↑, rot stiffness ↓
Add random rotational error 5E−4 X, Y, Z

Loads: negligible

CS4
Cd of this CS in next CS: [0, −co, 0] ← carriage offset
Loads: negligible
Model: simply supported aluminum beam
s×s square
Attachment points: [±10, 0, 0]
Linear stiffness ↑, rot stiffness
Loads: F_z = extruder weight
M_x = extruder weight · L

CS5
Cd of this CS in next CS: [0, 0, Cz]
(get Cz from MATLAB)
Stiffness ↑ in all dirs except Z
2 attachment points [±10, 0, 0]
Compliance: all zeros

CS6
Cd of this CS in next CS: [0, R, 0] ← from MATLAB
Loads: none
Model as simply supported beam
s×s square length H
Position along beam = Az
Linear/rot stiffness: ↑
Applied loads: none

CS7
Cd of this CS in next CS: [−X, −Y, −Z]
Loads: Approx weight of part
stiffness ↑ compliance ↓

CS8
Cd of this CS in next CS: [0, 0, 0]
Loads: none
Compliance: 0
Stiffness: ↑

All analysis and spreadsheet entries were made assuming the machine is in position to have the extruder at $[X, Y, Z]^T = [0, 0, 200]^T$ in the global coordinates, which is the same as CS7.

I modified the Error Budget spreadsheet so that I only need to change values in my reference sheet (the servo section in the summary sheet references my ref sheet) for different machine configurations and dimensions. Thus, I was able to play around with different end effector locations to see how that affects the errors in my machine (which change as the carriages move along axes). Below is an example of some of the calculations and dimensions I used in my reference sheet.

Hcz (mm)	40			ABS Young's modulus (N/m^2)	2000000000	2000	N/mm^2
Ae = Be = Ce (mm)	20			Extruder platform width (=2*Ae)	40		
Steel density (kg/m^3)	8050			Extruder platform length (=2*Ae)	40		
Extruder volume (mm^3)	8042.477193	8.04E-06	(m^3)	Extruder platform thickness (mm)	5		
Extruder mass (kg)	0.064741941	64.74194	(g)	Force req'd to buckle (N)	20561.67584		
Extruder weight (N)	0.647419414			Stress req'd to buckle (N/m^2)	102808379.2	102.8084	N/mm^2
Global coords of extruder (X, Y, Z)	0	0	200				
Coords of CS3 in CS4 (from MATLAB) (mm) (X, Y, Z) = (0, 0, 100)	0	-223	-193	Stress req'd to buckle (N/m^2)			
Max force on extruder head (N)	8						
Carriage offset co (mm)	20						
Rod length (mm)	460						
Az (mm)	543						
Bz (mm)	543						
Cz (mm)	543						
R (distance from center to rails) (mm)	258						

And this is the output found in the summary page. It shows that the average vector displacement is around 300 microns. This is quite large, but not too surprising or concerning, since I was super conservative in my models and chosen dimensions.

All Axes	For the Entire Machine				
	Number of axes	All axes' Geometric Errors			
	8	Random			Systematic
		Sum	RSS	Avg(SUM, RSS)	Sum
	deltaX	0.348275	0.226686	0.287480	0.000000
	deltaY	0.098113	0.049217	0.073665	0.000000
	deltaZ	0.084617	0.042822	0.063720	0.000000
	Vector displacement	0.371594	0.235887	0.303532	0.000000

Model preload method envisioned for the bearings and its effect on stiffness, bearing life, accuracy…especially given manufacturing limitations.

The biggest sources of backlash will be in the rod ends and carriage.

Rod ends: The rods are free to move/rotate about both ends (neither end is rigid), there is no way that I can stretch the rods to preload them. Thus, the best way to prevent wiggle and backlash is to tighten the nuts that constrain the rod ends to rods and use Loctite (making sure not to let it seep into the ball joint) to make sure that they don't come loose.

Carriage: If the timing belt is not sufficiently preloaded, the belt could either slip off the pulley, slide, or have backlash. The belt will need to be tightened to introduce preload into the axes. This can be done at the attachment point (clamp or bolt) to the carriage. Adding preload to the belt will also be a cheap and easy way to increase the linear bearing stiffness, life, and accuracy by further constraining the carriage to move vertically along the rails.

Assess risks and countermeasures and use to evolve designs and trim options to help converge on top two designs (or maybe even 1).

Risks	Countermeasures
Carriages jam on rails	St. Venant
Belt slides on pulley	Timing belt, with convex profile and/or flanges
Machine tips when extruder changes direction	Add weight or clamp machine to table

quickly/forcefully	
Vertical axis parts (rails, bars, carriages) are not identical	CNC machine / waterjet / laser cut / 3d print as many parts as possible (as opposed to machining by hand) or purchase standard parts
Not enough room for nuts/screws in extruder platform	Make platform large enough to accommodate screws and rod ends
Filament feed gets in the way	Route tube upwards through machine top, and then down on outside away from moving parts
Electronics get wet/dusty/short circuit	Properly constrain wires (tape), put case around electronics