Trajectory Generation
July 31, 2013

r	derivative to minimize in cost function
	implies r initial conditions at each keyframe (position and $r-1$ derivatives, indexed from 0 (constant term)
n	order of desired trajectory, minimum order is $2r-1$
	implies $n+1$ coefficients, indexed from 0 (constant term)
d	number of dimensions to optimize, indexed from 1
	for example, $d=3$ when optimizing $[xyz]$ position of a trajectory,
m	number of pieces in trajectory
	implies $m+1$ keyframes in trajectory, indexed from 1
t_{des}	vertical vector of desired arrival times at keyframes, indexed from 0
pos_{des}	matrix of desired positions, each row represents a derivative, each column represents a keyframe
	Inf represents unconstrained

1 Minimum order polynomial trajectories for the optimization problem

Suppose we want to find x(t) that minimizes the functional

$$J = \int_{t_0}^{t_1} F(x, x', ..., x^{(r)})dt = \int_{t_0}^{t_1} \| \frac{d^r x(t)}{dt} \|^2 dt,$$

where $x(t)$ has a polynomial basis:

$$x(t) = \sum_{i=0}^{n} c_i t^i$$

The functional is minimized by the Lagrangian:

$$\begin{aligned}
\mathcal{L} &= \frac{\partial F}{\partial x} - \frac{d}{dt}(\frac{\partial F}{\partial x'}) + \frac{d^2}{dt^2}(\frac{\partial F}{\partial x''}) - ... + \frac{d^r}{dt^r}(\frac{\partial F}{\partial x^{(r)}}) \\
&= 0 - 0 + 0 - ... + \frac{d^r}{dt^r}(2x^{(r)}) \\
&= 2x^{(2r)} \\
&= 0 \\
x^{(2r)} &= 0
\end{aligned}$$

To satisfy $x^{(2r)} = 0$, $x(t)$ must have its first $2r - 1$ derivatives defined. For a polynomial basis, this implies that $x(t)$ must be at least of order $2r - 1$.

2 Non-dimensionalization in time

For a trajectory $X(t)$ from t_0 to t_1, we can alternatively find the non-dimensionalized $x(\tau)$ from $\tau_0 = 0$ to $\tau_1 = 1$ and scale the result to any t_0 and t_1 value. For position, this is simply:

$$\begin{aligned}
\tau &= \frac{t - t_0}{t_1 - t_0} \\
X(t) &= x(\tau) = x(\frac{t - t_0}{t_1 - t_0}) \\
\frac{d}{dt}X(t) &= \frac{d}{dt}x(\tau) \\
&= \frac{d}{d\tau}\frac{d\tau}{dt}x(\tau) \\
&= \frac{1}{t_1 - t_0}\frac{d}{d\tau}x(\tau) \\
&... \\
\frac{d^r}{dt^r}X(t) &= \frac{1}{(t_1 - t_0)^r}\frac{d^r}{d\tau^r}x(\tau)
\end{aligned}$$

We can thus solve for each piece of any piece-wise trajectory from $\tau = 0 - 1$, then scale it for any t_0 to t_1 without recalculating the trajectory itself.

3 Optimization of a trajectory between two keyframes

We seek the desired trajectory $x(t) = c_n t^n + c_{n-1} t^{n-1} + ...c_1 t + c_0$. Let $x = [c_n \ \ c_{n-1} \ \ c_{n-2} \ \ ... \ \ c_1 \ \ c_0]^T$. Here, $d = 1$ and $m = 1$. We want to minimize the cost function:

$$\begin{aligned} J &= \int_{t_0}^{t_1} \| \frac{d^r x(t)}{dt} \|^2 dt \\ &= x^T Q x \\ \text{subject to: } & Ax = b \end{aligned}$$

TO FIND Q:
When $x = [c_0 \ \ c_1 \ \ ... \ \ c_{n-1} \ \ c_n]^T$:

$$Q[i,j] = \begin{cases} \prod_{k=0}^{r-1} (i-k)(l-k) \frac{t_1^{i+l-2r+1} - t_0^{i_l - 2r+1}}{i+l-2r+1}, & i \geq r \wedge l \geq r \\ 0, & i < r \vee l < r \end{cases}, i = 0...n, j = 0...n \tag{1}$$

Reflecting Q horizontally and vertically will give us the desired Q for the form of x we desire.

TO FIND A:

$$\begin{aligned} Ax &= \begin{bmatrix} x(t_0) \\ x'(t_0) \\ ... \\ x^{(r)}(t_0) \\ x(t_f) \\ x'(t_f) \\ ... \\ x^{(r)}(t_f) \end{bmatrix} \\ &= \begin{bmatrix} A(t_0) \\ A(t_1) \end{bmatrix} x \end{aligned}$$

Note that Ax only contains rows where constraints are specified - omit rows where a condition is unconstrained. Assuming that every condition is constrained, the general form of A is:

$$A[i,j](t) = \begin{cases} \prod_{k=0}^{i-1} (n-k-j) t^{n-j-i}, & n-j \geq i \\ 0, & n-j < i \end{cases}, i = 0...(r-1), j = 0...n \tag{2}$$

where $A[i,j]$ represents the $n-j$th coefficient of the ith derivative.

4 Optimization of a trajectory between $m+1$ keyframes in one dimension

We seek the piece-wise trajectory:

$$x(t) = \begin{cases} x_1(t) = c_{1,n}t^n + c_{1,n-1}t^{n-1} + ...c_{1,1}t + c_{1,0}, & t_0 \le t < t_1 \\ x_2(t) = c_{2,n}t^n + c_{2,n-1}t^{n-1} + ...c_{2,1}t + c_{2,0}, & t_1 \le t < t_2 \\ ... \\ x_m(t) = c_{m,n}t^n + c_{m,n-1}t^{n-1} + ...c_{m,1}t + c_{m,0}, & t_{m-1} \le t < t_m \end{cases}$$

Let $x = [c_{1,n} \;\; c_{1,n-1} \;\; c_{1,n-2} \;\; ... \;\; c_{1,1} \;\; c_{1,0} \;\; c_{2,n} \;\; c_{2,n-1} \;\; ... \;\; c_{m,1} \;\; c_{m,0}]^T$. Here, $d = 1$. We continue to minimize the cost function:

$$J = \int_{t_0}^{t_m} \| \frac{d^r x(t)}{dt} \|^2 dt = x^T Q x$$
$$\text{subject to: } Ax = b$$

TO FIND Q:
Recall that for each $x_k = [c_{k,0} \;\; c_{k,1} \;\; ... \;\; c_{k,n-1} \;\; c_{k,n}]^T$, where $k = 1...m$:

$$Q[i,j]_k = \begin{cases} \prod_{k=0}^{r-1} (i-k)(l-k) \frac{t_k^{i+l-2r+1} - t_{k-1}^{i_l - 2r+1}}{i+l-2r+1}, & i \ge r \wedge l \ge r \\ 0, & i < r \vee l < r \end{cases}, i = 0...n, j = 0...n, k = 1...m$$

Reflecting Q_k horizontally and vertically will give us the desired Q_k for the form of $x_k(t)$ we desire. We can then create the block diagonal matrix:

$$Q = \begin{bmatrix} Q_1 & 0 & 0 & ... & 0 \\ 0 & Q_2 & 0 & ... & 0 \\ & & ... & & \\ 0 & ... & 0 & Q_{m-1} & 0 \\ 0 & ... & 0 & 0 & Q_m \end{bmatrix} \tag{3}$$

TO FIND A:

First, we need to account for endpoint constraints:

$$A_{endpoint}x = \begin{bmatrix} x_1(t_0) \\ x_1'(t_0) \\ ... \\ x_1^{(r)}(t_0) \\ x_1(t_1) \\ x_1'(t_1) \\ ... \\ x_1^{(r)}(t_1) \\ x_2(t_1) \\ x_2'(t_1) \\ ... \\ x_2^{(r)}(t_1) \\ x_2(t_2) \\ x_2'(t_2) \\ ... \\ x_2^{(r)}(t_2) \\ ... \\ x_m(t_{m-1}) \\ x_m'(t_{m-1}) \\ ... \\ x_m^{(r)}(t_{m-1}) \\ x_m(t_m) \\ x_m'(t_m) \\ ... \\ x_m^{(r)}(t_m) \end{bmatrix} = \begin{bmatrix} A(t_0) & 0 & 0 & ... & 0 \\ A(t_1) & 0 & 0 & ... & 0 \\ 0 & A(t_1) & 0 & ... & 0 \\ 0 & A(t_2) & 0 & ... & 0 \\ 0 & 0 & A(t_2) & ... & 0 \\ 0 & 0 & A(t_3) & ... & 0 \\ & & ... & & \\ 0 & 0 & ... & 0 & A(t_{m-1}) \\ 0 & 0 & ... & 0 & A(t_m) \end{bmatrix} x$$

Note that again, we omit rows where a condition is unconstrained. Also, except for constraints at t_0 and t_m, every other constraint must be included twice - a constraint at t_k must be applied as a final condition to x_k and an initial condition x_{k+1}. The equation for $A[i,j](t)$ is given in Eq. 2.

We must also account for continuity constraints, which ensure that when the trajectory switches from one piece to another at the keyframes, position and all derivatives lower than r remain continuous, for a smooth path. In other words, we require:

$$A_{cont}x = \begin{bmatrix} x_1(t_1) - x_2(t_1) \\ x_1'(t_1) - x_2'(t_1) \\ ... \\ x_1^{(r)}(t_1) - x_2^{(r)}(t_1) \\ ... \\ x_{m-1}(t_{m-1}) - x_m(t_{m-1}) \\ x_{m-1}'(t_{m-1}) - x_m'(t_{m-1}) \\ ... \\ x_{m-1}^{(r)}(t_{m-1}) - x_m^{(r)}(t_{m-1}) \end{bmatrix} x = \begin{bmatrix} A_{cont}(t_1) & 0 & 0 & ... & 0 \\ 0 & A_{cont}(t_2) & 0 & ... & 0 \\ 0 & 0 & A_{cont}(t_3) & ... & 0 \\ & & ... & & \\ 0 & 0 & ... & 0 & A_{cont}(t_{m-1}) \end{bmatrix} x = 0$$

where:

$$A_{cont}[i,j](t) = \begin{cases} \prod_{k=0}^{i-1}(n-k-j)t^{n-j-i}, & n-j \geq i \wedge j \leq n \\ 0, & n-j < i \wedge j \leq n \\ -\prod_{k=0}^{i-1}(1-k-j)t^{1-j-i}, & 1-j \geq i \wedge j > n \\ 0, & 1-j < i \wedge j > n \end{cases}, i = 0...(r-1), j = 0...2(n+1) \tag{4}$$

Our constraints, $Ax = b$, take the form:

$$Ax = \begin{bmatrix} A_{endpoint} \\ A_{cont} \end{bmatrix} x = \begin{bmatrix} b_{endpoint} \\ 0 \end{bmatrix} = b \tag{5}$$

5 Optimization of a trajectory between $m+1$ keyframes in d dimensions, with corridor constraints

We seek the piece-wise multi-dimension trajectory:

$$\mathbf{X}(t) = [X_1(t) \;\; X_2(t) \;\; X_3(t) \;\; .... \;\; X_d(t)],$$
$$\text{where: } X_p(t) = \begin{cases} x_{p,1}(t) = c_{p,1,n}t^n + c_{p,1,n-1}t^{n-1} + ...c_{p,1,1}t + c_{p,1,0}, & t_0 \leq t < t_1 \\ x_{p,2}(t) = c_{p,2,n}t^n + c_{p,2,n-1}t^{n-1} + ...c_{p,2,1}t + c_{p,2,0}, & t_1 \leq t < t_2 \\ ... \\ x_{p,m}(t) = c_{p,m,n}t^n + c_{p,m,n-1}t^{n-1} + ...c_{p,m,1}t + c_{p,m,0}, & t_{m-1} \leq t < t_m \end{cases}$$

Let $x = [c_{1,1,n} \;\; ... \;\; c_{1,1,0} \;\; c_{1,2,n} \;\; c_{1,2,n-1} \;\; ... \;\; c_{1,m,0} \;\; c_{2,1,n} \;\; c_{2,1,n-1} \;\; ... \;\; c_{2,m,0} \;\; ... \;\; c_{d,m,0}]^T$. We continue to minimize the cost function:

$$J = \int_{t_0}^{t_m} \| \frac{d^r \mathbf{X}(t)}{dt} \|^2 dt = x^T Q x$$
$$\text{subject to: } A_{eq}x = b_{eq}, A_{ineq}x \leq b_{ineq}$$

TO FIND Q:
For each dimension, we define Q_p using Eq. 3. We then create the block diagonal matrix:

$$Q = \begin{bmatrix} Q_1 & 0 & 0 & ... & 0 \\ 0 & Q_2 & 0 & ... & 0 \\ & & ... & & \\ 0 & ... & 0 & Q_{p-1} & 0 \\ 0 & ... & 0 & 0 & Q_p \end{bmatrix} \tag{6}$$

TO FIND A_{eq}:

We simply use Eq. 5 to find $A_{eq_p}x = b_{eq}$ for each dimension and create the block diagonal matrix:

$$A_{eq} = \begin{bmatrix} A_{eq_1} & 0 & 0 & ... & 0 \\ 0 & A_{eq_2} & 0 & ... & 0 \\ & & ... & & \\ 0 & ... & 0 & A_{eq_{p-1}} & 0 \\ 0 & ... & 0 & 0 & A_{eq_p} \end{bmatrix} \tag{7}$$

$$b_{eq} = \begin{bmatrix} b_{eq_1} \\ b_{eq_2} \\ ... \\ b_{eq_{p-1}} \\ b_{eq_p} \end{bmatrix} \tag{8}$$

TO FIND A_{ineq}:

We can add corridor, or inequality constraints, to the paths between keyframes as well. Let the constraint i be between keyframe i and $i+1$ and applied to dimensions a, b, and c (for example, if the trajectory dimensions were $[\psi \ \ x \ \ y \ \ \phi \ \ z]^T$, the dimensions x, y, and z position would be $a=2$, $b=3$, $c=4$). Let $\mathbf{r}_i = [X_{a,i} \ \ X_{b,i} \ \ X_{c,i}]^T$, or the position vector of keyframe i and $\mathbf{X}(t) = [X_a(t) \ \ X_b(t) \ \ X_c(t)]^T$. We want position to stay within a corridor of width δ_i and imposed using n_c intermediate points.

$$\mathbf{t}_i = \frac{\mathbf{r}_{i+1} - \mathbf{r}_i}{\|\mathbf{r}_{i+1} - \mathbf{r}_i\|}$$

$$\mathbf{d}_i(t) = (\mathbf{X}(t) - \mathbf{r}_i) - ((\mathbf{X}(t) - \mathbf{r}_i) \cdot \mathbf{t}_i)\mathbf{t}_i$$

We want to satisfy the constraint:

$$\|\mathbf{d}_i\|_\infty \leq \delta_i \text{ for } t_i \leq t \leq t_{i+1}$$

$$|\mathbf{e}_p \cdot \mathbf{d}_i(t_i + \frac{j}{1+n_c}(t_{i+1} - t_i)| \leq \delta_i, p = a, b, c, j = 1...n_c$$

The inequality breaks down into:

$$(\mathbf{e}_p \cdot \mathbf{d}_i(t_i + \frac{j}{1+n_c}(t_{i+1} - t_i)) \leq \delta_i$$

$$-(\mathbf{e}_p \cdot \mathbf{d}_i(t_i + \frac{j}{1+n_c}(t_{i+1} - t_i)) \leq \delta_i$$

This results in a total of $2(p)(n_c)$ constraints for each corridor constraint.

$$\begin{aligned}
\mathbf{d}_i(t) &= (\mathbf{X}(t) - \mathbf{r}_i) - ((\mathbf{X}(t) - \mathbf{r}_i) \cdot \mathbf{t}_i)\mathbf{t}_i \\
&= (\mathbf{X}(t) - \mathbf{r}_i) - \left(\sum_{j=1}^{p} ((\mathbf{X}(t) - \mathbf{r}_i) \cdot \mathbf{e}_j)(\mathbf{t}_i \cdot \mathbf{e}_j) \right) \mathbf{t}_i \\
&= (\mathbf{X}(t) - \mathbf{r}_i) - \left(\sum_{j=1}^{p} ((\mathbf{X}(t) - \mathbf{r}_i) \cdot \mathbf{e}_j)((\frac{\mathbf{r}_{i+1} - \mathbf{r}_i}{\|\mathbf{r}_{i+1} - \mathbf{r}_i\|}) \cdot \mathbf{e}_j) \right) (\frac{\mathbf{r}_{i+1} - \mathbf{r}_i}{\|\mathbf{r}_{i+1} - \mathbf{r}_i\|}) \\
&= (\mathbf{X}(t) - \mathbf{r}_i) - \left(\sum_{j=1}^{p} ((\mathbf{X}(t) - \mathbf{r}_i) \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j) \right) (\frac{\mathbf{r}_{i+1} - \mathbf{r}_i}{\|\mathbf{r}_{i+1} - \mathbf{r}_i\|^2})
\end{aligned}$$

For dimension $\mathbf{e}_a$:

$$(\mathbf{X}(t)-\mathbf{r}_i)\cdot\mathbf{e}_a-\left(\sum_{j=1}^{p}((\mathbf{X}(t)-\mathbf{r}_i)\cdot\mathbf{e}_j)((\mathbf{r}_{i+1}-\mathbf{r}_i)\cdot\mathbf{e}_j)\right)(\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a\leq\delta_i$$

$$\mathbf{X}(t)\cdot\mathbf{e}_a-\mathbf{r}_i\cdot\mathbf{e}_a-\left(\sum_{j=1}^{p}(\mathbf{X}(t)\cdot\mathbf{e}_j)((\mathbf{r}_{i+1}-\mathbf{r}_i)\cdot\mathbf{e}_j)\right)(\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a$$
$$-\left(\sum_{j=1}^{p}(\mathbf{r}_i\cdot\mathbf{e}_j)((\mathbf{r}_{i+1}-\mathbf{r}_i)\cdot\mathbf{e}_j)\right)(\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a\leq\delta_i$$

$$\mathbf{X}(t)\cdot\mathbf{e}_a-\mathbf{r}_i\cdot\mathbf{e}_a-(\mathbf{X}(t)\cdot\mathbf{e}_a)(\frac{(\mathbf{r}_{i+1}-\mathbf{r}_i)^2}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a$$
$$-\left(\sum_{j=1,j\neq a}^{p}(\mathbf{X}(t)\cdot\mathbf{e}_j)((\mathbf{r}_{i+1}-\mathbf{r}_i)\cdot\mathbf{e}_j)\right)(\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a$$
$$-\left(\sum_{j=1}^{p}(\mathbf{r}_i\cdot\mathbf{e}_j)((\mathbf{r}_{i+1}-\mathbf{r}_i)\cdot\mathbf{e}_j)\right)(\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a\leq\delta_i$$

$$\mathbf{X}(t)\cdot\mathbf{e}_a\left(1-(\frac{(\mathbf{r}_{i+1}-\mathbf{r}_i)^2}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a\right)-\left(\sum_{j=1,j\neq a}^{p}(\mathbf{X}(t)\cdot\mathbf{e}_j)((\mathbf{r}_{i+1}-\mathbf{r}_i)\cdot\mathbf{e}_j)\right)(\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a$$
$$\leq\delta_i+\mathbf{r}_i\cdot\mathbf{e}_a+\left(\sum_{j=1}^{p}(\mathbf{r}_i\cdot\mathbf{e}_j)((\mathbf{r}_{i+1}-\mathbf{r}_i)\cdot\mathbf{e}_j)\right)(\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a$$

$$-\left((\mathbf{X}(t)-\mathbf{r}_i)\cdot\mathbf{e}_a-\left(\sum_{j=1}^{p}((\mathbf{X}(t)-\mathbf{r}_i)\cdot\mathbf{e}_j)((\mathbf{r}_{i+1}-\mathbf{r}_i)\cdot\mathbf{e}_j)\right)(\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a\right)\leq\delta_i$$

$$-\mathbf{X}(t)\cdot\mathbf{e}_a\left(1-(\frac{(\mathbf{r}_{i+1}-\mathbf{r}_i)^2}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a\right)+\left(\sum_{j=1,j\neq a}^{p}(\mathbf{X}(t)\cdot\mathbf{e}_j)((\mathbf{r}_{i+1}-\mathbf{r}_i)\cdot\mathbf{e}_j)\right)(\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a$$
$$\leq\delta_i-\mathbf{r}_i\cdot\mathbf{e}_a-\left(\sum_{j=1}^{p}(\mathbf{r}_i\cdot\mathbf{e}_j)((\mathbf{r}_{i+1}-\mathbf{r}_i)\cdot\mathbf{e}_j)\right)(\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2})\cdot\mathbf{e}_a$$

We can then construct the A_{ineq} and b_{ineq} matrices:

$$A_{ineq}x \leq b_{ineq}$$

$$\begin{bmatrix} (\mathbf{e}_a \cdot \mathbf{d}_i(t_i + \frac{1}{1+n_c}(t_{i+1} - t_i))) \\ -(\mathbf{e}_a \cdot \mathbf{d}_i(t_i + \frac{1}{1+n_c}(t_{i+1} - t_i))) \\ (\mathbf{e}_a \cdot \mathbf{d}_i(t_i + \frac{2}{1+n_c}(t_{i+1} - t_i))) \\ -(\mathbf{e}_a \cdot \mathbf{d}_i(t_i + \frac{2}{1+n_c}(t_{i+1} - t_i))) \\ ... \\ (\mathbf{e}_a \cdot \mathbf{d}_i(t_i + \frac{n_c}{1+n_c}(t_{i+1} - t_i))) \\ -(\mathbf{e}_a \cdot \mathbf{d}_i(t_i + \frac{n_c}{1+n_c}(t_{i+1} - t_i))) \\ (\mathbf{e}_b \cdot \mathbf{d}_i(t_i + \frac{1}{1+n_c}(t_{i+1} - t_i))) \\ -(\mathbf{e}_b \cdot \mathbf{d}_i(t_i + \frac{1}{1+n_c}(t_{i+1} - t_i))) \\ ... \\ (\mathbf{e}_c \cdot \mathbf{d}_i(t_i + \frac{n_c}{1+n_c}(t_{i+1} - t_i))) \\ -(\mathbf{e}_c \cdot \mathbf{d}_i(t_i + \frac{n_c}{1+n_c}(t_{i+1} - t_i))) \end{bmatrix} \leq \begin{bmatrix} \delta_i \\ \delta_i \\ \delta_i \\ \delta_i \\ ... \\ \delta_i \\ \delta_i \\ \delta_i \\ \delta_i \\ ... \\ \delta_i \\ \delta_i \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{X}(t_1) \cdot \mathbf{e}_a \left(1 - (\frac{(\mathbf{r}_{i+1}-\mathbf{r}_i)^2}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a\right) - \left(\sum_{j=1, j\neq a}^{p} (\mathbf{X}(t_1) \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a \\ -\mathbf{X}(t_1) \cdot \mathbf{e}_a \left(1 - (\frac{(\mathbf{r}_{i+1}-\mathbf{r}_i)^2}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a\right) + \left(\sum_{j=1, j\neq a}^{p} (\mathbf{X}(t_1) \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a \\ ... \\ \mathbf{X}(t_{n_c}) \cdot \mathbf{e}_a \left(1 - (\frac{(\mathbf{r}_{i+1}-\mathbf{r}_i)^2}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a\right) - \left(\sum_{j=1, j\neq a}^{p} (\mathbf{X}(t_{n_c}) \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a \\ -\mathbf{X}(t_{n_c}) \cdot \mathbf{e}_a \left(1 - (\frac{(\mathbf{r}_{i+1}-\mathbf{r}_i)^2}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a\right) + \left(\sum_{j=1, j\neq a}^{p} (\mathbf{X}(t_{n_c}) \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a \\ \mathbf{X}(t_1) \cdot \mathbf{e}_b \left(1 - (\frac{(\mathbf{r}_{i+1}-\mathbf{r}_i)^2}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_b\right) - \left(\sum_{j=1, j\neq a}^{p} (\mathbf{X}(t_1) \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_b \\ -\mathbf{X}(t_1) \cdot \mathbf{e}_b \left(1 - (\frac{(\mathbf{r}_{i+1}-\mathbf{r}_i)^2}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_b\right) + \left(\sum_{j=1, j\neq a}^{p} (\mathbf{X}(t_1) \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_b \\ ... \\ \mathbf{X}(t_{n_c}) \cdot \mathbf{e}_c \left(1 - (\frac{(\mathbf{r}_{i+1}-\mathbf{r}_i)^2}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_c\right) - \left(\sum_{j=1, j\neq a}^{p} (\mathbf{X}(t_{n_c}) \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_c \\ -\mathbf{X}(t_{n_c}) \cdot \mathbf{e}_c \left(1 - (\frac{(\mathbf{r}_{i+1}-\mathbf{r}_i)^2}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_c\right) + \left(\sum_{j=1, j\neq a}^{p} (\mathbf{X}(t_{n_c}) \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_c \end{bmatrix}$$

$$\leq \begin{bmatrix} \delta_i + \mathbf{r}_i \cdot \mathbf{e}_a + \left(\sum_{j=1}^{p} (\mathbf{r}_i \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a \\ \delta_i - \mathbf{r}_i \cdot \mathbf{e}_a - \left(\sum_{j=1}^{p} (\mathbf{r}_i \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a \\ ... \\ \delta_i + \mathbf{r}_i \cdot \mathbf{e}_a + \left(\sum_{j=1}^{p} (\mathbf{r}_i \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a \\ \delta_i - \mathbf{r}_i \cdot \mathbf{e}_a - \left(\sum_{j=1}^{p} (\mathbf{r}_i \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_a \\ \delta_i + \mathbf{r}_i \cdot \mathbf{e}_b + \left(\sum_{j=1}^{p} (\mathbf{r}_i \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_b \\ \delta_i - \mathbf{r}_i \cdot \mathbf{e}_b - \left(\sum_{j=1}^{p} (\mathbf{r}_i \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_b \\ ... \\ \delta_i + \mathbf{r}_i \cdot \mathbf{e}_c + \left(\sum_{j=1}^{p} (\mathbf{r}_i \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_c \\ \delta_i - \mathbf{r}_i \cdot \mathbf{e}_c - \left(\sum_{j=1}^{p} (\mathbf{r}_i \cdot \mathbf{e}_j)((\mathbf{r}_{i+1} - \mathbf{r}_i) \cdot \mathbf{e}_j)\right) (\frac{\mathbf{r}_{i+1}-\mathbf{r}_i}{\|\mathbf{r}_{i+1}-\mathbf{r}_i\|^2}) \cdot \mathbf{e}_c \end{bmatrix}$$

6 Alternate formation of optimization of $m+1$ keyframes in one dimensions as a joint optimization problem

We seek the piece-wise trajectory:

$$x(t) = \begin{cases} x_1(t) = c_{1,n}t^n + c_{1,n-1}t^{n-1} + ...c_{1,1}t + c_{1,0}, & t_0 \leq t < t_1 \\ x_2(t) = c_{2,n}t^n + c_{2,n-1}t^{n-1} + ...c_{2,1}t + c_{2,0}, & t_1 \leq t < t_2 \\ ... \\ x_m(t) = c_{m,n}t^n + c_{m,n-1}t^{n-1} + ...c_{m,1}t + c_{m,0}, & t_{m-1} \leq t < t_m \end{cases}$$

Let $x = [c_{1,n} \;\; c_{1,n-1} \;\; c_{1,n-2} \;\; ... \;\; c_{1,1} \;\; c_{1,0} \;\; c_{2,n} \;\; c_{2,n-1} \;\; ... \;\; c_{m,1} \;\; c_{m,0}]^T$. Here, $d = 1$. We continue to minimize the cost function:

$$J = \int_{t_0}^{t_m} \|\frac{d^r x(t)}{dt}\|^2 dt = x^T Q x$$

TO FIND Q:
We define Q using the block diagonal matrix found in Eq. 3.

TO FIND A:
We create the alternate state $d_1 =$
$[x_1(t_0) \; x_1'(t_0) \; ... \; x_1^{(r-1)}(t_0) \; x_1(t_1) \; x_1'(t_1) \; ... \; x_1^{(r-1)}(t_1) \; x_2(t_2) \; x_2'(t_2) \; ... \; x_2^{(r-1)}(t_2) \; ... \; x_m(t_m) \; x_m'(t_m) \; ... \; x_m^{(r-1)}(t_m)]^T$.

Here, we define matrix A_k such that
$d_k = [x_k(t_{k-1}) \;\; x_k'(t_{k-1}) \;\; ... \;\; x_k^{(r-1)}(t_{k-1}) \;\; x_k(t_k) \;\; x_k'(t_k) \;\; ... \;\; x_k^{(r-1)}(t_k)]^T = A_k x_k =$
$A_k[c_{k,n} \;\; c_{k,n-1} \;\; c_{k,n-2} \;\; ... \;\; c_{k,1} \;\; c_{k,0}]^T$, where:

$$A[i,j]_k = \begin{cases} \prod_{k=0}^{i-1}(n-k-j)t_0^{n-j-i}, & n-j \geq i \wedge j \leq n \wedge i < r \\ 0, & n-j < i \wedge j \leq n \wedge i < r \\ -\prod_{k=0}^{(i-r)-1}(1-k-j)t_1^{1-j-(i-r)}, & 1-j \geq (i-r) \wedge j > n \wedge i \geq r \\ 0, & 1-j < i \wedge j > n \wedge i \geq r \end{cases}, i = 0...2r-1, j = 0...2(n+1)$$

We can then construct matrix A where $d = Ax$:

$$A = \begin{bmatrix} A_1 & 0 & 0 & ... & 0 \\ 0 & A_2 & 0 & ... & 0 \\ & & ... & & \\ 0 & ... & 0 & A_{m-1} & 0 \\ 0 & ... & 0 & 0 & A_m \end{bmatrix} \quad (9)$$

TO FIND M and D:

We define a matrix $D = [D_f \quad D_p]^T$. To begin, let:

$$D' = \begin{bmatrix} x_1(t_0) \\ x_1(t_1) \\ x_2(t_2) \\ ... \\ x_m(t_m) \\ x_1'(t_0) \\ x_1'(t_1) \\ x_2'(t_2) \\ ... \\ x_m'(t_m) \\ x_1''(t_0) \\ x_1''(t_1) \\ x_2''(t_2) \\ ... \\ x_m^{(r-1)}(t_m) \end{bmatrix}$$

We can then rearrange D' into the fixed constraints, D_f and the unfixed constraints, D_p. We seek a matrix M such that $d = MD$.
We can then find:

$$\begin{aligned} J &= x^T Q x \\ &= (A^{-1}d)^T Q (A^{-1}d) \\ &= d^T A^{-T} Q A^{-1} d \\ &= (MD)^T A^{-T} Q A^{-1} (MD) \\ &= D^T M^T A^{-T} Q A^{-1} M D \end{aligned}$$

Define:

$$\begin{aligned} R &= M^T A^{-T} Q A^{-1} M \\ &= \begin{bmatrix} R_{FF} & R_{FP} \\ R_{PF} & R_{PP} \end{bmatrix} \\ D &= \begin{bmatrix} D_f \\ D_p \end{bmatrix} \\ D^T &= \begin{bmatrix} D_f^T & D_p^T \end{bmatrix} \end{aligned}$$

We can then solve for:

$$\begin{aligned} J &= \begin{bmatrix} D_f^T & D_p^T \end{bmatrix} \begin{bmatrix} R_{FF} & R_{FP} \\ R_{PF} & R_{PP} \end{bmatrix} \begin{bmatrix} D_F \\ D_P \end{bmatrix} \\ &= D_f^T R_{FF} D_f + D_p^T R_{PF} D_f + D_f^T R_{FP} D_p + D_p^T R_{PP} D_p \\ \frac{dJ}{dD_p} &= D_f^T R_{PF}^T + D_f^T R_{FP} + D_p^T R_{PP} + D_p^T R_{PP}^T \end{aligned}$$

R is symmetric, since:

$$\begin{aligned} R^T &= (M^T A^{-T} Q A^{-1} M)^T \\ &= (A^{-1}M)^T Q^T (M^T A^{-T})^T \\ &= M^T A^{-T} Q A^{-1} M \\ &= R, \text{ note that Q is symmetric} \end{aligned}$$

This implies that:

$$
\begin{aligned}
R_{PF}^T = R_{FP}, R_{PP}^T &= R_{PP} \\
\frac{dJ}{dD_p} &= 2(D_f^T R_{FP} + D_p^T R_{PP})
\end{aligned}
$$

When optimized,

$$
\begin{aligned}
\frac{dJ}{dD_p} &= 2(D_f^T R_{FP} + D_p^T R_{PP}) = 0 \\
D_f^T R_{FP} + D_p^T R_{PP} = 0 \\
D_p^T &= -D_f^T R_{FP} R_{PP}^{-1} \\
D_p &= -R_{PP}^{-T} R_{FP}^T D_f
\end{aligned}
$$

Now that we have D_p, we can use $D = [D_f \quad D_p]^T$ to reconstruct the all values for D'. This gives us the equality constraint: $Ax = b = D'$, where A is defined in Eq. 9. We now have the minimization problem:

$$
f(x) = x^T Q x, \text{subject to: } g(x) = Ax - b = 0
$$

We can set up an equation using Lagrange multipliers:

$$
\begin{aligned}
\frac{\partial}{\partial x}(f(x) + \lambda g(x)) = 0 \\
\frac{\partial}{\partial x}(x^T Q x + \lambda(Ax - b)) &= x^T Q + x^T Q^T + \lambda A \\
&= 2x^T Q + \lambda A, \text{ (note that Q is symmetric)} \\
&= 0 \\
2Q^T x + A^T \lambda^T = 0
\end{aligned} \tag{10}
$$

Eq. 10 and the constraint $Ax = b$ gives two equations for two unknowns, x and λ.

$$
\begin{aligned}
\begin{bmatrix} 2Q & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda^T \end{bmatrix} &= \begin{bmatrix} 0 \\ b \end{bmatrix} \\
\begin{bmatrix} x \\ \lambda^T \end{bmatrix} &= \begin{bmatrix} 2Q & A^T \\ A & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ b \end{bmatrix}
\end{aligned}
$$