Motion Profiling

Alexander Koldy

Forest Hills Robotics League

alexanderkoldy.ak@gmail.com

September 18, 2024

Introduction

In this lecture, we will discuss motion profiling in the context of the *FIRST Tech Challenge* (FTC).

Generating motion profiles helps our drivetrain and other mechanisms achieve smoother, more controlled motions. For a drivetrain, motion profiles help our robot mitigate slipping, a behavior that occurs when rapid accelerations are introduced to the system.

We will discuss generation asymmetric motion profiles, i.e., profiles where the acceleration and deceleration values can be different!

Time-based profiles

To generate a motion profile in Java, we will first need to figure out a few constants, namely the acceleration times and distances. We will call our acceleration, cruise (constant velocity) and deceleration times Δt_{accel}, Δt_{cruise} and Δt_{decel}, respectively. Moreover, we will call out acceleration, cruise and deceleration distances d_{accel}, d_{cruise} and d_{decel}, respectively. These constants will come from tune-able variables that we set for each motion profile, namely

- Maximum velocity, v_{max}
- Constant (maximum) acceleration, a_{accel}
- Constant (maximum) deceleration, a_{decel}

We will also pass in a desired distance we would like to cover (d). We will use basic kinematics equations to generate the time-based motion profiles, namely

$$d_1 = d_0 + v_0\Delta t + \frac{1}{2}a\Delta t^2$$

$$a = \frac{\Delta v}{\Delta t}$$

where d_0 (typically set to zero) and d_1 are our initial and final distances, v_0 is some potential initial velocity, and a is some constant acceleration.

Note: we will treat a_{decel} as a positive constant and apply the negation in our math!

Time-based profiles

To start, we find the total time it will take to acceleration from a zero velocity to v_max as well as the time it will take to deceleration from v_max back down to zero. These two equations are:

$$\Delta t_{accel} = \frac{v_{max}}{a_{accel}}$$
$$\Delta t_{decel} = \frac{v_{max}}{a_{decel}}$$

Moreover, we find the total distance our acceleration and deceleration phases will cover as follows:

$$d_{accel} = \frac{1}{2} a_{accel} (\Delta t_{accel})^2$$
$$d_{decel} = \frac{1}{2} a_{decel} (\Delta t_{decel})^2$$

Time-based profiles

In some cases, our input distance d will be too short for our motion profile to reach our desired maximum velocity, i.e., $d < d_{accel} + d_{decel}$. If this is the case, we will calculate a new, lower maximum velocity. This will require us to recalculate our Δt and distance values, so we can assume that these are no longer known. To figure out these calculations, we will employ velocity continuity, i.e., the velocity after accelerating and decelerating should be zero. Recalling that $a = \frac{\Delta v}{\Delta t}$:

$$a_{accel} * \Delta t_{accel} - a_{decel} * \Delta t_{decel} = 0$$

Note: we negate the second term since technically deceleration is negative.
We also introduce the equation for the total distance of both phases:

$$d = \frac{1}{2} a_{accel} (\Delta t_{accel})^2 + \tilde{v}_{max} \Delta t_{decel} - \frac{1}{2} a_{decel} (\Delta t_{decel})^2$$

The first term is our total distance during our acceleration phase, and the second two terms is our total distance during our deceleration phase. In this case, we are attempting to solve for our acceleration and deceleration times. We know d since it is our set distance. $\tilde{v}_{max}$ is our new maximum velocity which is equivalent to $a_{accel} \Delta t_{accel}$.

Time-based profiles

Using the two equations on the previous slide, we can rearrange and solve for both Δt_{accel} and Δt_{decel}:

$$\Delta t_{decel} = \sqrt{\frac{2a_{accel}d}{(a_{decel})^2 + a_{accel}a_{decel}}}$$

$$\Delta t_{accel} = \frac{a_{decel}}{a_{accel}}\Delta t_{decel}$$

We now recalculate our acceleration and deceleration distances as:

$$d_{accel} = \frac{1}{2}a_{accel}(\Delta t_{accel})^2$$

$$d_{decel} = \frac{1}{2}a_{decel}(\Delta t_{decel})^2$$

Our new maximum velocity (what we denoted as $\tilde{v}_{max}$) is:

$$\tilde{v}_{max} = a_{accel}\Delta t_{accel}$$

Time-based profiles

Now, assuming we do not have issues with our desired distance, we can calculate our cruise distance as:

$$d_{cruise} = d - d_{accel} - d_{decel}$$

Since our velocity is constant when cruising, our cruise time is:

$$\Delta t_{cruise} = \frac{d_{cruise}}{v_{max}}$$

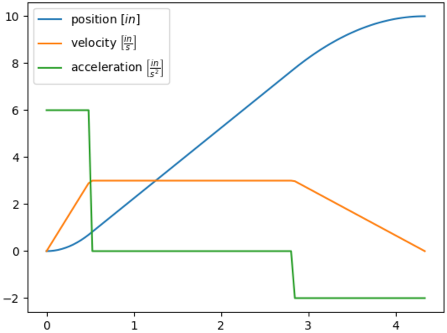
Our total motion time is defined as

$$\Delta t = \Delta t_{accel} + \Delta t_{cruise} + \Delta t_{decel}$$

We now have all of the details we need to fully define our motion profile!

In this example, we set

- $d = 10\text{in}$
- $v_{max} = 3\,\frac{\text{in}}{\text{s}}$
- $a_{accel} = 6\,\frac{\text{in}}{\text{s}^2}$
- $a_{decel} = 2\,\frac{\text{in}}{\text{s}^2}$



Time-based profiles

Now, to generate the position, velocity and acceleration at some time T, we can use the following conditionals. For position:

$$\text{position} = p(T)$$

$$= \begin{cases} d & T > \Delta t \\ \frac{1}{2} a_{accel} T^2 & T \leq \Delta t_{accel} \\ d_{accel} + v_{max} T_{cruise} & \Delta t_{accel} < T \leq \Delta t_{accel} + \Delta t_{cruise} \\ d_{accel} + d_{cruise} + v_{max} T_{decel} - \frac{1}{2} a_{decel} (T_{decel})^2 & t_{accel} + \Delta t_{cruise} < T \leq \Delta t \end{cases}$$

where $T_{accel} = T - \Delta t_{accel}$ and $T_{decel} = T - \Delta t_{accel} - \Delta t_{cruise}$. For velocity we have:

$$\text{velocity} = v(T)$$

$$= \begin{cases} 0 & T > t \\ a_{accel} T & T \leq \Delta t_{accel} \\ v_{max} & \Delta t_{accel} < T \leq \Delta t_{accel} + \Delta t_{cruise} \\ v_{max} - a_{decel} T_{decel} & t_{accel} + \Delta t_{cruise} < T \leq \Delta t \end{cases}$$

Time-based profiles

Finally for acceleration, we have:

$$
\begin{aligned}
\text{acceleration} &= a(T) \\
&= \begin{cases} 0 & T > t \\ a_{accel} & T \leq \Delta t_{accel} \\ 0 & \Delta t_{accel} < T \leq \Delta t_{accel} + \Delta t_{cruise} \\ -a_{decel} & t_{accel} + \Delta t_{cruise} < T \leq \Delta t \end{cases}
\end{aligned}
$$

When building a motion profile object in Java, these piecewise functions can be represented using conditionals (i.e., `if`, `else if`, and `else` statements). A simple naming convention could be `getPosition`, `getVelocity` and `getAcceleration`, with the current elapsed time (from a timer object) as an input to each.

Using motion profiles

Motion profiles can be used for many FTC mechanisms. For the drivetrain, we can approximate the total distance of our path by taking the distance between each waypoint. This distance, along with our other tune-able parameters, can be fed to our motion profile generator to generate a longitudinal motion profile.

Motion profiles can also be used on commonly-used linear slide mechanisms. This ensures that the extension and retraction is smooth.