Vehicle Commands

There are three ways to control CPM vehicles. They take precedence in the following order:

1. Direct Commands
2. Path Tracking Commands
3. Trajectory Commands

That means if a vehicle receives both Direct Commands and Trajectory Commands, it will perform the Direct Commands.

Direct Command

See https://github.com/embedded-software-laboratory/cpm_lab/blob/master/cpm_lib/dds_idl/VehicleCommandDirect.idl

The direct command uses dimensionless, uncalibrated inputs and applies them directly to the motor and servo.

	motor_throttle	**steering_servo**
-1.0	Maximum reverse acceleration	Full right
0.0	Brake, stop	Steering roughly centered, but may have an offset
+1.0	Maximum forward acceleration	Full left

Path Tracking Command

The vehicles are controlled in path tracking mode with PathTracking-Messages.

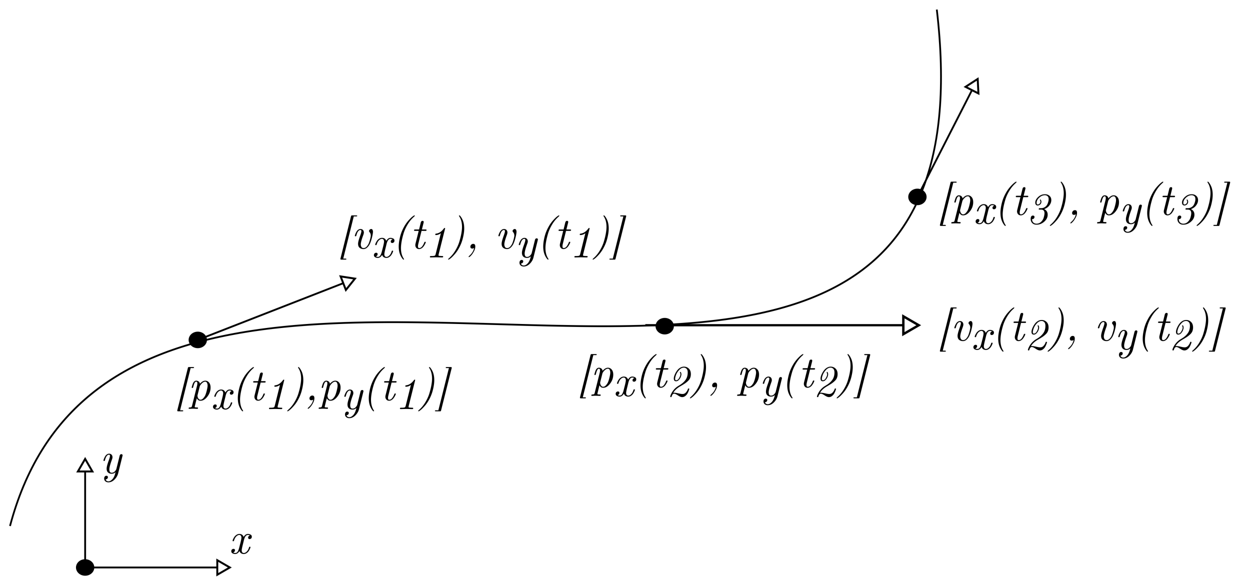
The inputs in Path Tracking Mode are a path consisting of poses and distances. The distance describes the distance along the path from the first path point to the current path point. Between path points, an interpolation with cubic splines is done on the basis of the poses and the distances. If you understood the Trajectory Command structure, a path is basically a trajectory with the speed of 1.

Trajectory Command

See https://github.com/embedded-software-laboratory/cpm_lab/blob/master/cpm_lib/dds_idl/VehicleCommandTrajectory.idl

The reference trajectory is defined as a **Cubic Hermite spline**. A spline is a function that is defined piecewise in time. Thus it can be extended in real time, which (metaphorically) looks like this:

The points defining the spline must be given as a set of five numbers `[ti, px(ti), py(ti), vx(ti), vy(ti)]`, where `ti` is the absolute time in nanoseconds of the trajectory point, `[px, py]` is the position vector (meters) and `[vx, vy]` is the velocity vector (m/s).



The time between trajectory points `t(i+1) - t(i)` is not fixed, but should be between 100 ms and 400 ms. Due to network latency and the controller structure, the trajectory points need to be created and sent ahead of time. The exact required lead time is TBD, but 1000 ms should be enough.

> **Troubleshooting**
>
> Check the *Logs (Tab)*.
>
> The vehicle does not start driving.
>
> When first taking control of the vehicle using a reference trajectory, the reference trajectory must be consistent with the current position and orientation of the vehicle. Small deviations (~0.5m, ~30 degrees) are allowed. For large deviations the trajectory controller will deactivate and stop the vehicle.
>
> The vehicle stops driving.
>
> The reference trajectory is a function of continuous time. Thus it also implicitly defines other kinematic quantities, such as yaw, yaw rate, curvature, acceleration and jerk, for any point in time. The user must ensure that the reference trajectory is within the limits of the vehicle's capabilities. The curvature should be less than 3/m ("three per meter"), and the acceleration should be less than 5 m/s². If these limits are significantly exceeded, the trajectory controller will deactivate and stop the vehicle.