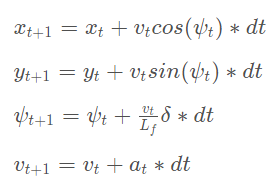
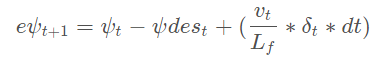
**Rubric Points**

**Model:**

The kinematic behavior is modeled in terms of the state space equation below. The state variables are position in x, y, velocity (vt), orientation (ψ), cross track error (cte) and orientation error (eψ). This model is used to find the response of the state variables for given actuator values like acceleration and steering (at and δ).





**Timestep Length and Elapsed Duration (N & dt):**

I have first experimented the code with the values N=20 and dt=0.1, this combination is giving the optimization over a large horizon, because of this more oscillation is seen in tight corners. Then I have decided to decrease N to 10 and increase dt=0.2. Because of this combination there is more lag between the computations and the behavior of the car in this extended interval is unknown and the car behaves erratically. Then I have chosen N=10 and dt=0.1 to make the car drive within the lines.

Previous tried values – N=20 and dt=0.1 & N=10 and dt=0.2

**Polynomial Fitting and MPC Preprocessing:**

In order to make the polynomial fitting easy the waypoints have to converted based on the coordinate system fitted to the car. Hence the position of the car is considered as origin and orientation is considered as zero. Hence the new point coordinates are calculated using the rotation and translation matrices, i.e., subtracting the x and y values from the current position value of the car.

**Model Predictive Control with Latency:**

Latency is considered in the constraints formulation where the actuations values are taken from the previous step. Since the timesteps are 100 milliseconds it was easy to incorporate in the model. In addition, a cost expression which considers the reduction of speed with increase in steering angle is considered and it is nothing but the product of speed and steering angle value. This helps in the smooth cornering and avoids the vehicle from going out of the track.