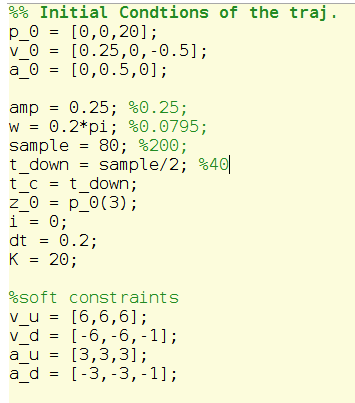
Hw8

**Homework1**

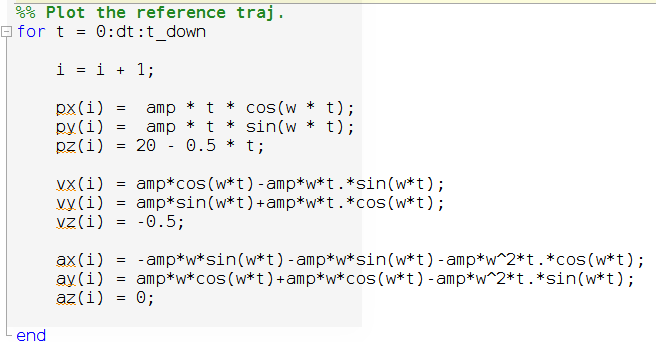
1. **主代码结构**
2. 确定跟踪轨迹的参数

注意到课件上的轨迹的频率w有问题，0.08 rad/s的螺旋线轨迹与高为20、底直径为10的要求不符，因此修正频率为0.2×pi rad/s。t\_down意为设定的追踪时间，因题目预设vz为-0.5m/s,下降高度为20m，故追踪时间定为40s。



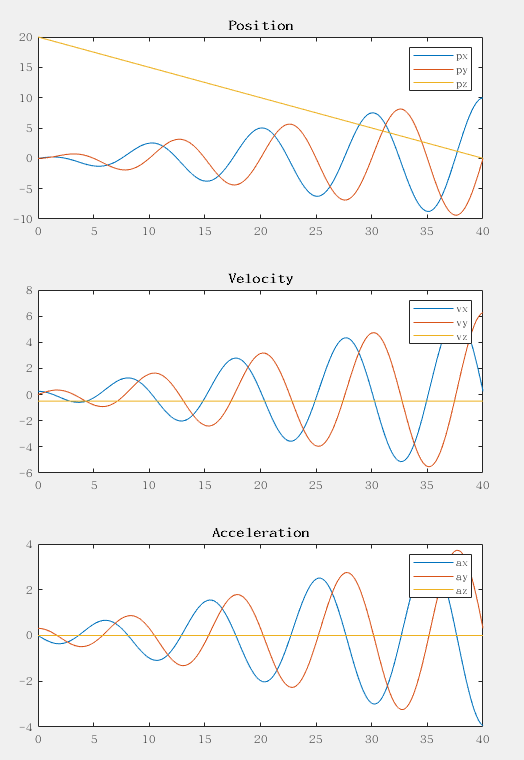
1. 确定要跟踪的轨迹

因为系统模型为三阶积分器，故若要跟踪位置轨迹，则速度与加速度的轨迹也必然需要跟踪，所需跟踪的轨迹分别如下：

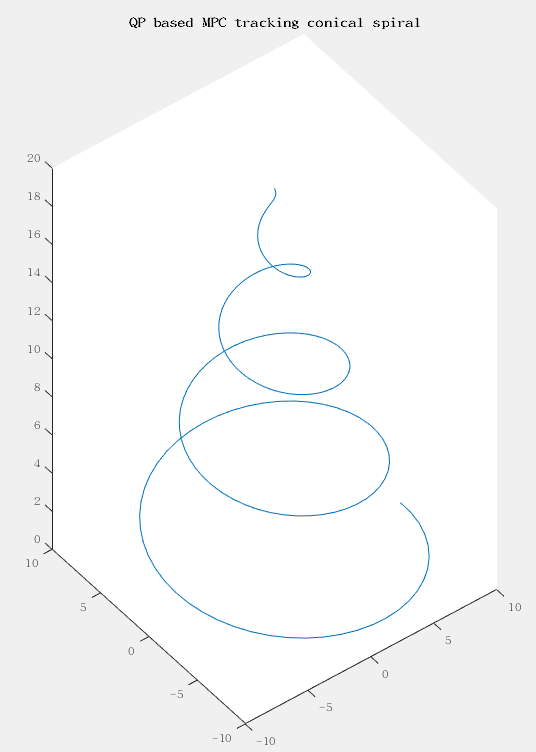


可视化：

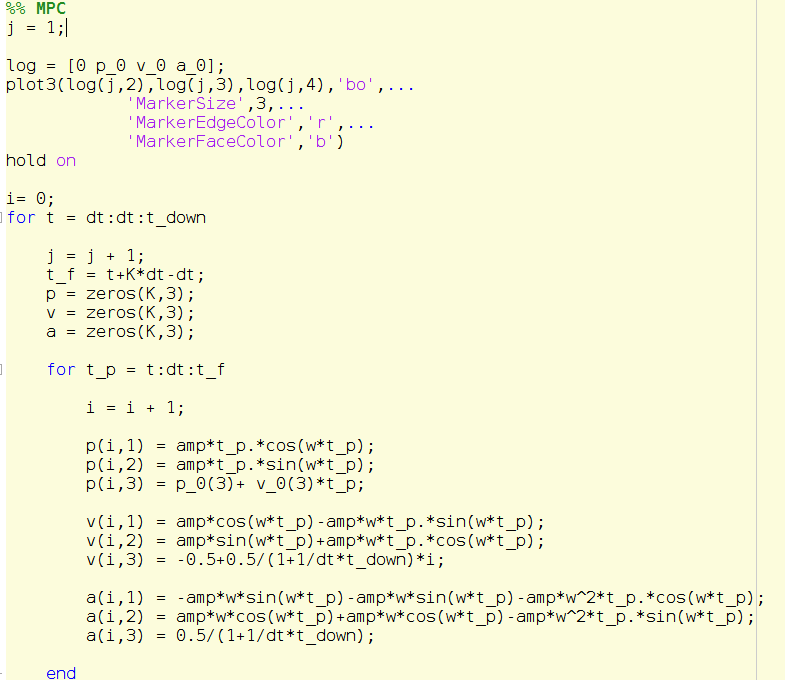
1. 一维上所需跟踪的位置、速度、加速度轨迹：



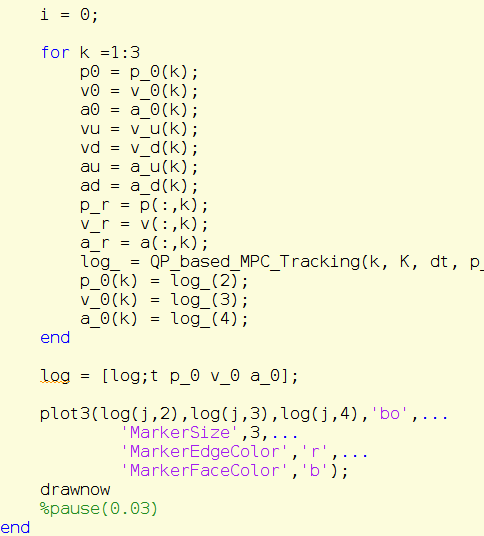
1. 三维位置轨迹：



1. 利用MPC跟踪轨迹

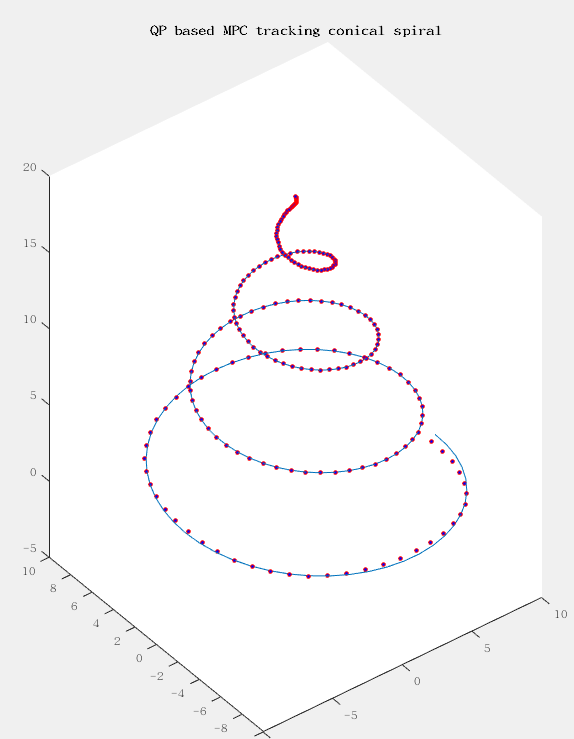


Log数组负责记录下所有已跟踪的信息。进入for循环后，先利用第一个for循环构建当前点下往前20s的跟踪轨迹;

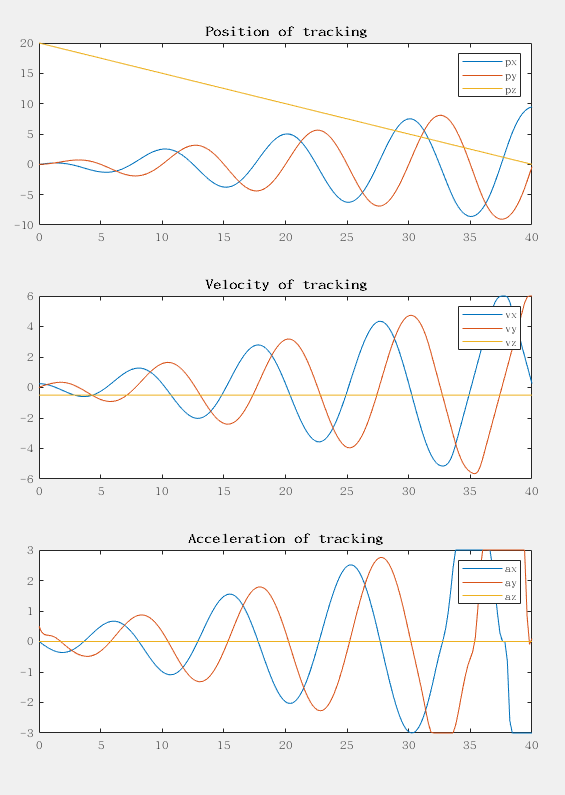


构造完成所需要的跟踪轨迹后，利用第二个for循环分别对x,y,z三个方向上的位置、速度以及加速进行MPC追踪：调用MPC函数QP\_Based\_MPC\_Tracking前，先确定对应方向的初值，软约束的上下限，利用MPC函数得出的结果更新当前初值p\_0、v\_0、a\_0的信息。之后把当前初值记录进log数组，并可视化。

1. 以Marker形式可视化当前所追踪的位置

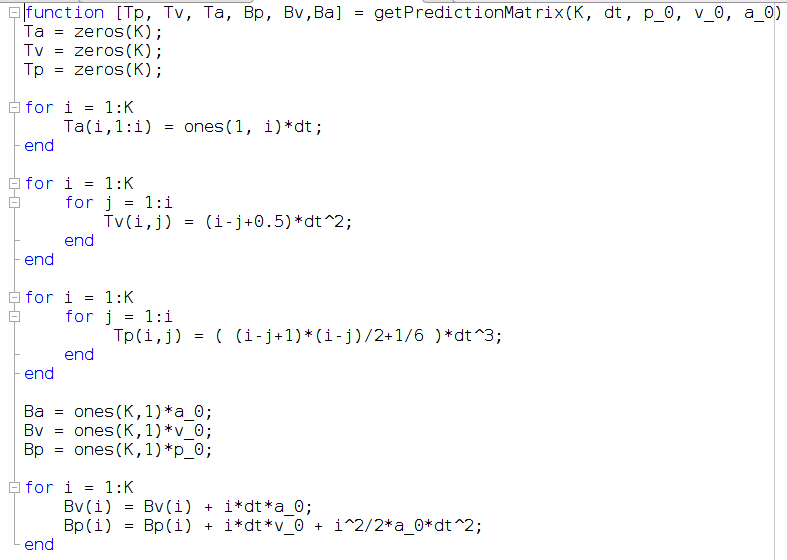


1. 一维上追踪所得的位置、速度、加速度轨迹

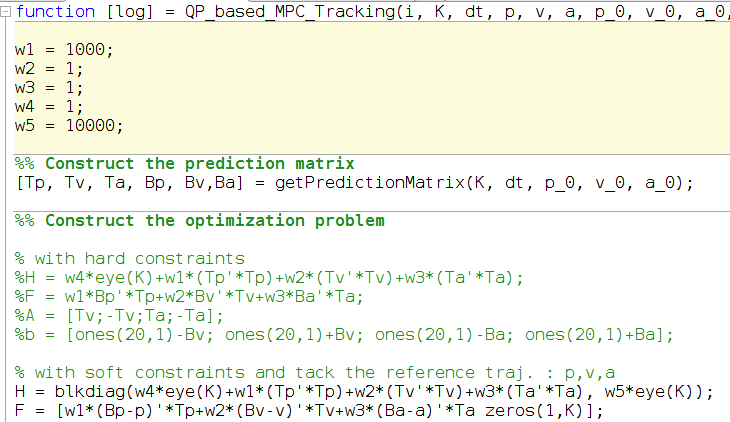


1. **所用函数**
2. getPredictionMatrix.m

与课件一致



1. QP\_Based\_MPC\_Tracking.m



通过在F矩阵中把Bp、Bv、Ba项修正为Bp-p、Bv-v、Ba-a以引入跟踪轨迹的要求。