

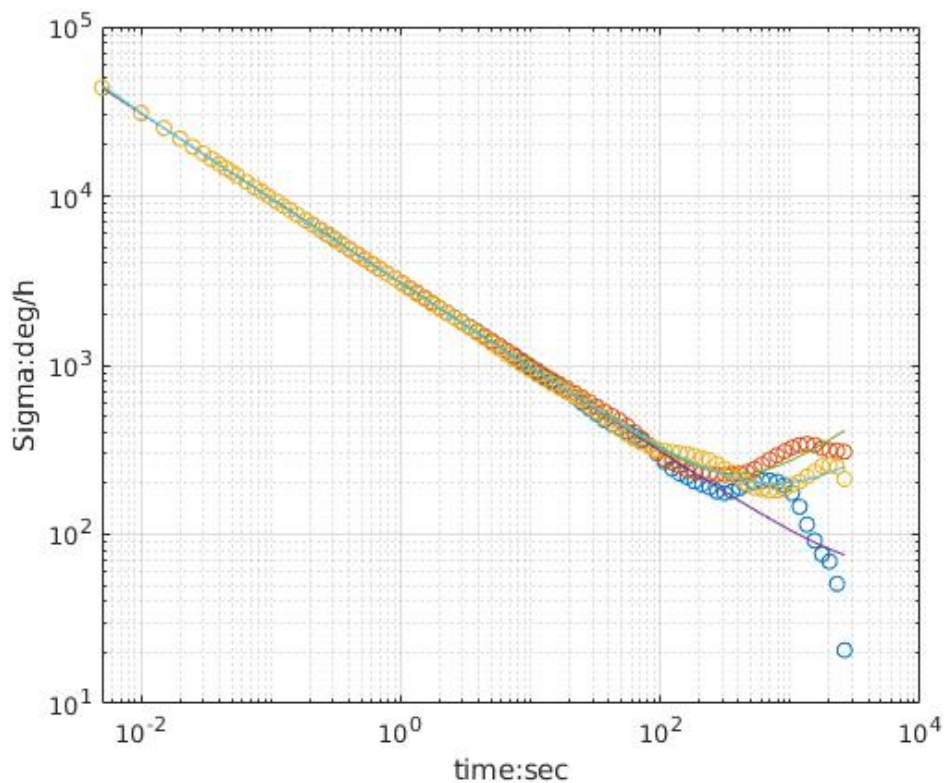
## 基础作业

### 第一题：

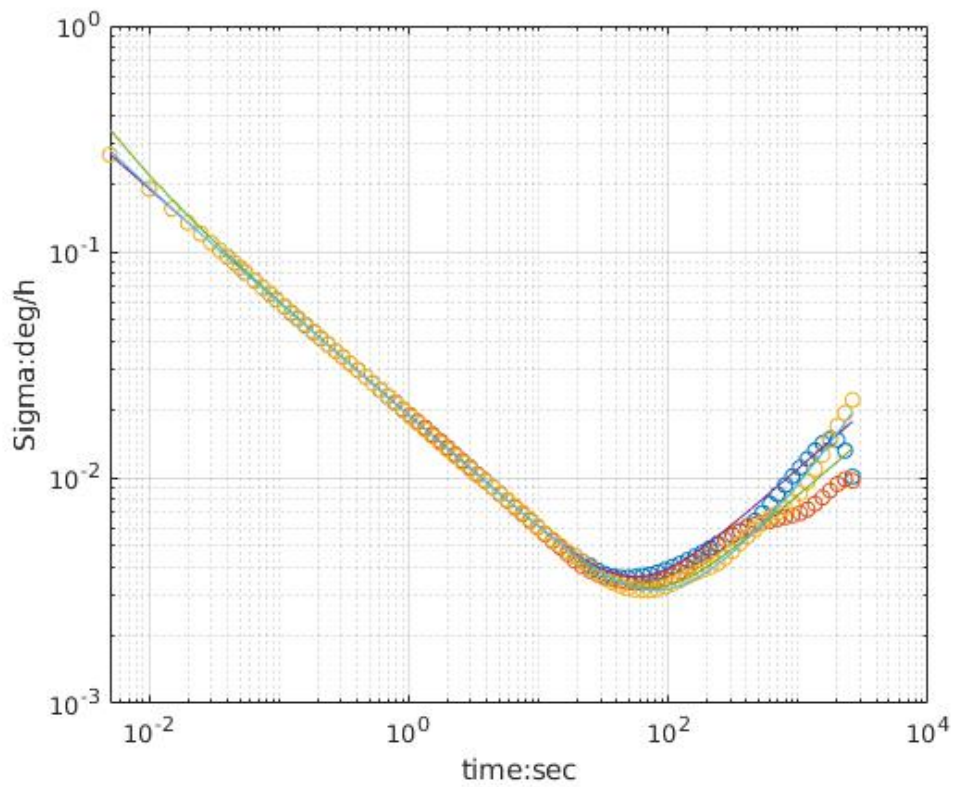
第一组噪声：noise (0, 1.0)

- (1) 首先创建 ros 工作空间 imu\_sim\_ws/src, 把 ros 版本代码放到 src 下, catkin\_make 编译, 运行 vio\_data\_simulation\_node 节点, 生成 imu.bag 数据;
- (2) 下载 code\_utils 到 ros 工作空间的 src 下, 重新 catkin\_make 编译;
- (3) 下载 imu\_utils 到工作空间的 src 中, 重新 catkin\_make 编译;
- (4) 在 imu\_utils/launch 下 roslaunch A3.launch, 然后运行 rosbag play imu.bag, 生成文件放在 imu\_utils/data1/下 (这里要修改 A3.launch 中的路径);
- (5) 最后用 matlab 运行 imu\_utils/scripts 下的 draw\_allan.m 文件, 注意要修改文件路径和文件名。生成曲线图片在 figure 下。

gyr1.jpg:



accl.jpg:

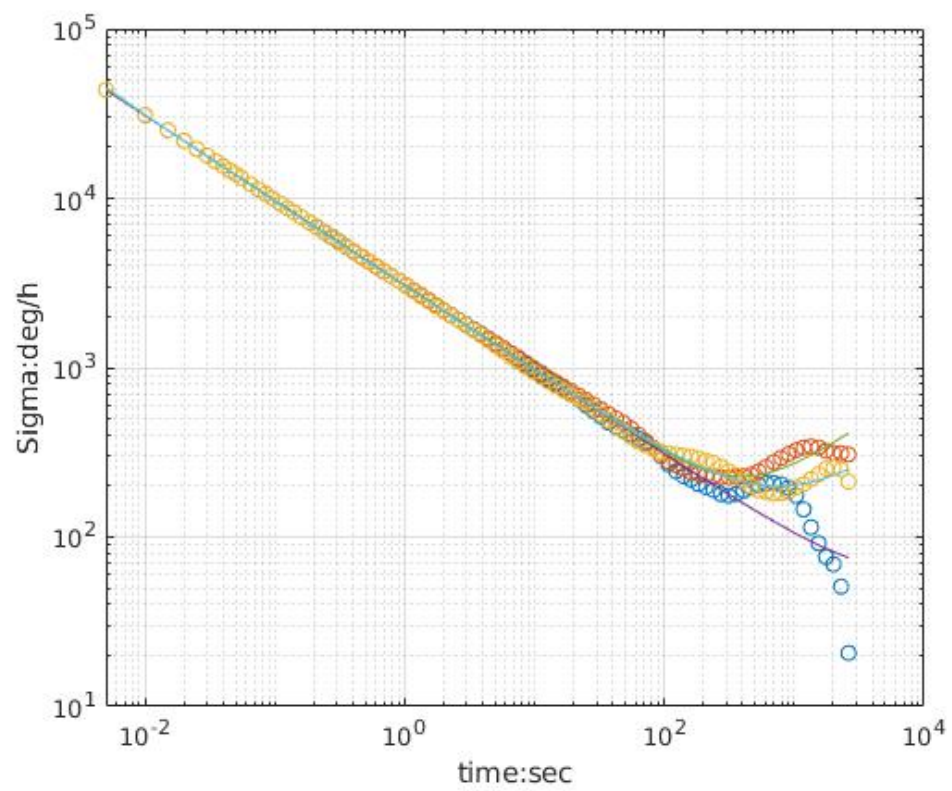


第二组噪声: noise (1.0, 2.0)

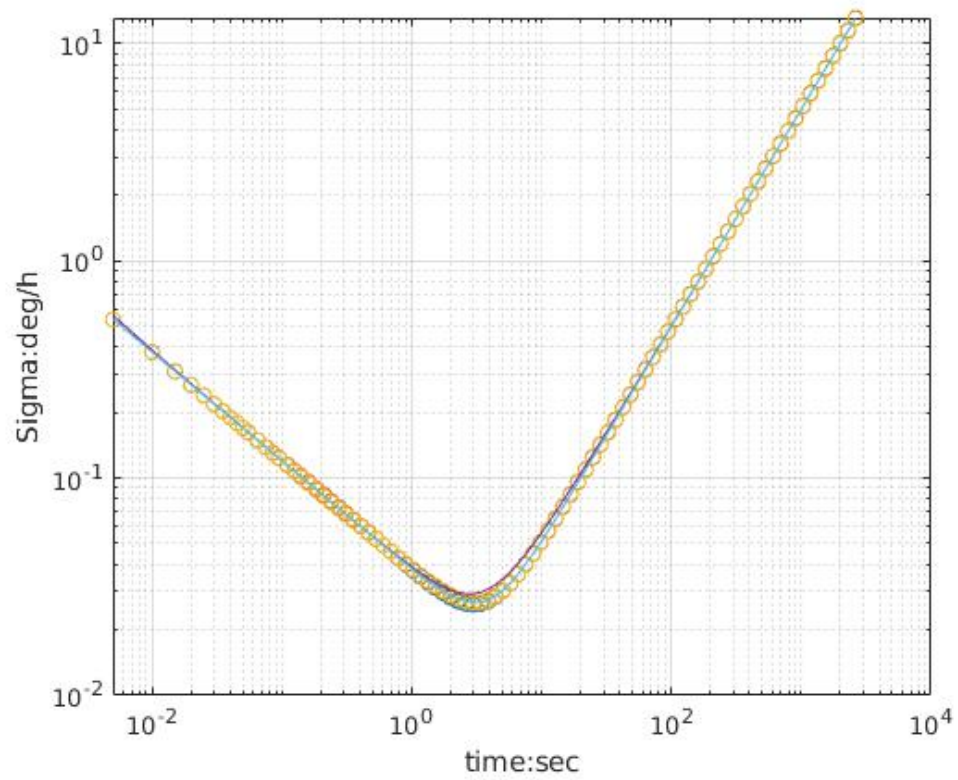
这里要修改源码 imu.cpp 中的噪声数据, 从 (0, 1.0) 修改为 (1.0, 2.0)。

其余按第一组操作即可, 画图如下:

gyr2.jpg:



acc2.jpg:



## 第二题：

把代码中的欧拉积分换成中值积分，代码修改见下图：

```
/**
/// imu 动力学模型 欧拉积分
Eigen::Vector3d acc_w = Qwb * (imupose.imu_acc) + gw; // aw = Rwb * ( acc_body - acc_bias ) + gw
Qwb = Qwb * dq;
Vw = Vw + acc_w * dt;
Pwb = Pwb + Vw * dt + 0.5 * dt * dt * acc_w;
**/

//中值积分
Eigen::Quaterniond Qwbk = Qwb;
Qwb = Qwb * dq;
Eigen::Vector3d acc_w = (Qwbk * (imupose.imu_acc) + gw + Qwb * (imudata[i+1].imu_acc) + gw) * 0.5;
Pwb = Pwb + Vw * dt + 0.5 * dt * dt * acc_w;
Vw = Vw + acc_w * dt;
```

修改后用 python-tool 画图对比（欧拉积分的曲线和中值积分的曲线很接近）：

