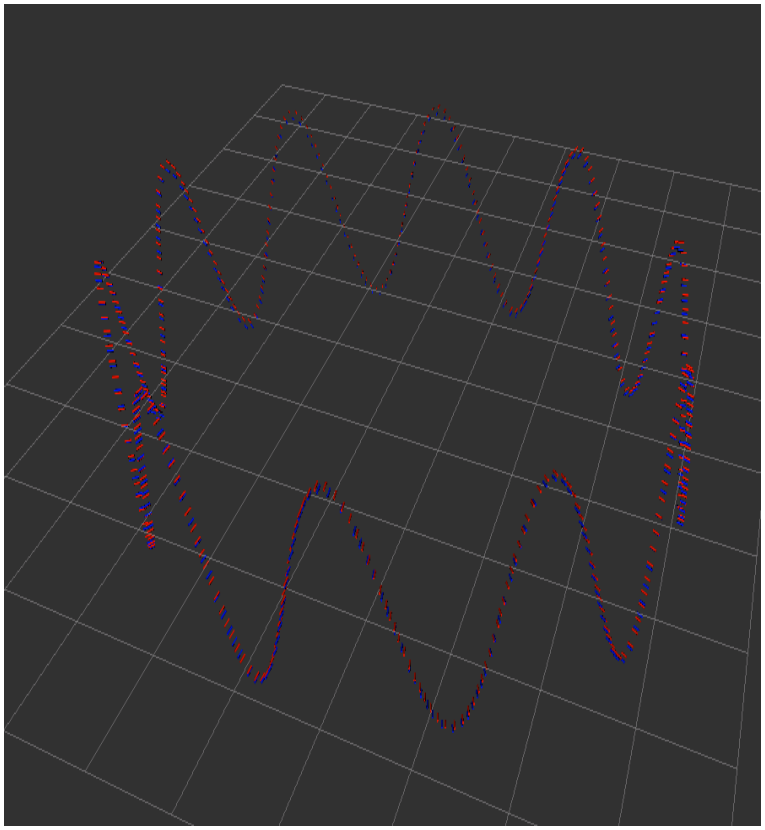
及格要求

初始化时间同步

精度评估时用 tum 格式的数据，可以不进行时间同步，但是初始化的时候，为了严谨起见，在原代码的基础上，参考前几节课的代码框架，增加了初始化时间同步的模块。主要修改了 readData 函数，HasData 函数以及数据文件 odom_data.hpp。

代码及仿真结果

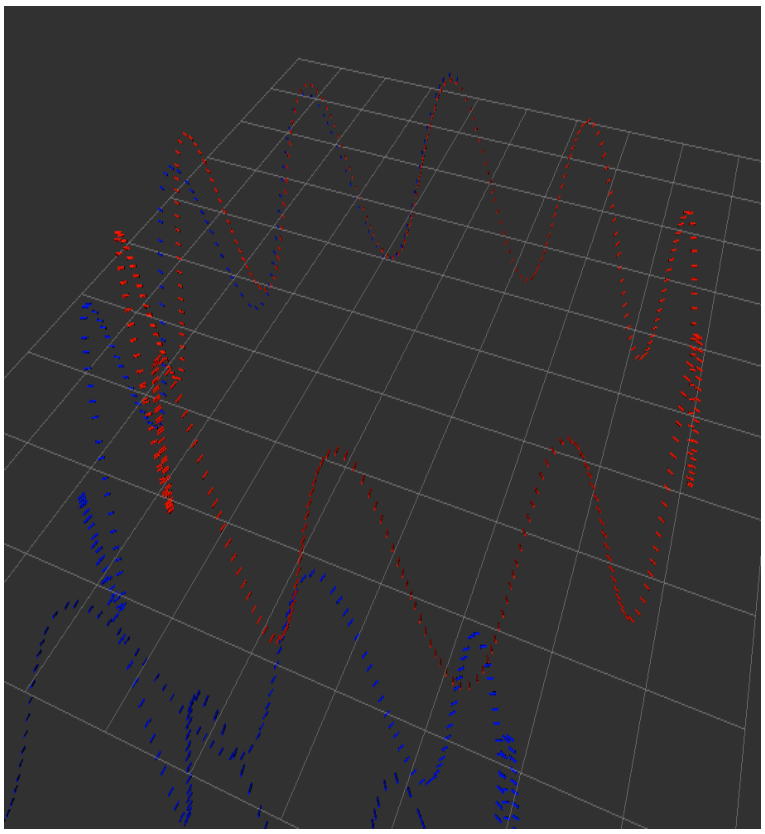
函数 UpdatePose 中 index_prev 和 index_curr 的选取策略为：index_prev 选取 imu_data_buff 中的第一个元素，index_curr 选取最后一个元素，并且每次清空 imu_data_buff 后，再把之前的最后一个元素放回，作为下一次的第一个元素，如此循环，仿真结果如下：



良好要求

代码及仿真结果

参考 estimator/activity.cpp 中的函数 GetAngularDelta 和函数 GetVelocityDelta，将其中的中值法公式换为欧拉法公式即可，新建的两个函数分别为 GetAngularDeltaEuler 和 GetVelocityDeltaEuler，仿真结果如下：



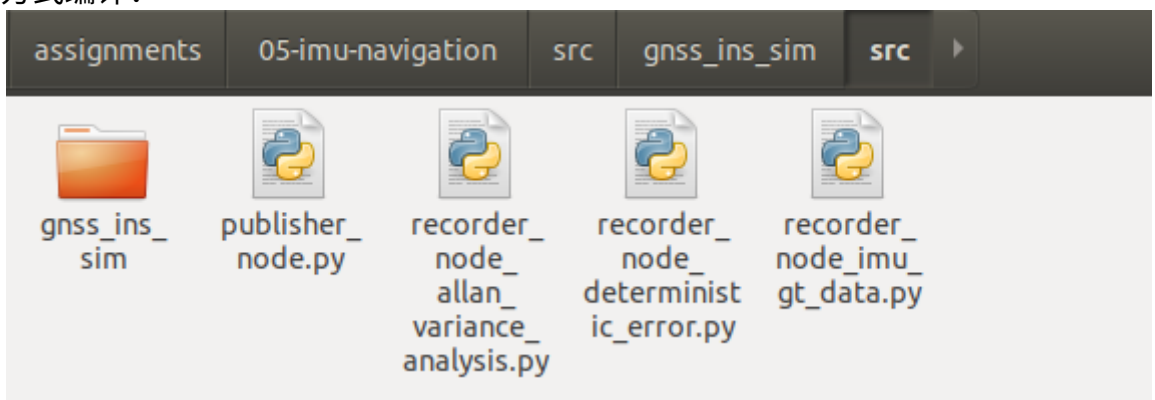
中值法和欧拉法对比

对于 VIO 课程中这种数据生成方式，运动相对剧烈和复杂，中值法的结果明显优于欧拉法，原因在于中值法用两个积分端点的平均加速度和平均角速度做积分，而欧拉法只用一个端点。这里不做定量分析，优秀要求中再做定量分析。

优秀要求

细节说明

为了使用 gnss_ins_sim 包，将其放置到如下图路径（目前还不理解），然后按照 readme 中的编译方式编译：



参考 recorder_node_allan_variance_analysis.py 写了程序 recorder_node_imu_gt_data.py，用于生成 bag 包，其中包含 ground truth 数据和 IMU 数据。

对于 ground truth 数据中的位置数据，每个数据减去初始的位置。

对于 IMU 数据，选用 gnss_ins_sim 中没有噪声的惯性器件数据 ref_gyro 和 ref_accel，排除噪声影响，抓住问题核心，直接对比中值法和欧拉法的性能。

用 gnss_ins_sim 中的函数 geoparams.py 由初始经纬高计算出的重力参数（9.7942164704）替换掉原程序中的重力参数，用-9.794216470，消除重力参数不统一的影响。

```
50      rn = Re / (math.sqrt(1.0 - E_SQR*sl_sqr))
51      g1 = normal_gravity * (1 + k*sl_sqr) / math.sqrt(1.0 - E_SQR*sl_sqr)
52      g = g1 * (1.0 - (2.0/Re) * (1.0 + FLATTENING + m - 2.0*FLATTENING*sl_sqr)*h + 3.0*h*h/Re/Re)
53      return rm, rn, g, sl, cl, W_IE
```

为了完成精度评估，参考之前的代码框架，增加了放置在 tools 文件夹下的文件管理类 file_manager.cpp，在 estimator/activity.cpp 中增加了两个存放轨迹的函数，SaveTrajectoryKitti 和 SaveTrajectoryTum，一个存放 kitti 格式数据，一个存放 tum 格式数据。

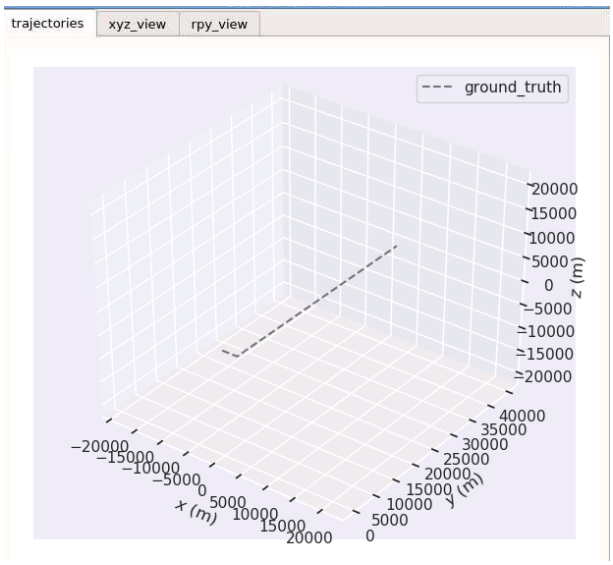
为了方便是使用 gnss_ins_sim 生成的 rosbag，新建 launch 文件 imu_integration_no_generator.launch，在 imu_integration.launch 基层上去掉 generator 节点，运行 imu_integration_no_generator.launch 后，再手动播放 rosbag 包。

对于不同的运动状态，可用一个 rosbag 包含，播放的时候用 rosbag play 中的-s 和-u 选项来指定使用的时间段。

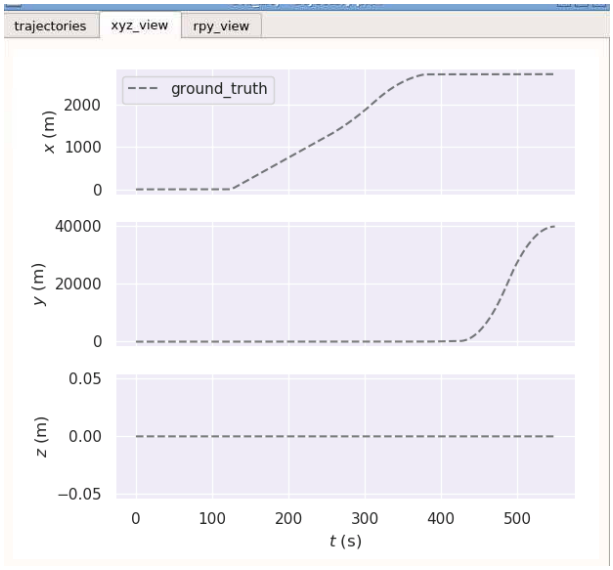
allan_variance_analysis.csv 中的运动定义如下表：

command	yaw	pitch	roll	vx_body	vy_body	vz_body	duration	GPS	Sum time
1	0	0	0	0	0	0	120	1	120
5	0	0	0	10	0	0	20	1	130.01
1	0	0	0	0	0	0	120	1	250.01
1	0	0	0	0.1	0	0	60	1	310.01
1	0	0	0	0	0	0	5	1	315.01
1	0	0	0	-0.2	0	0	60	1	375.01
3	90	0	0	0	0	0	60	1	394.09
1	0	0	0	0	0	0	30	1	424.09
1	0	0	0	10	0	0	60	1	484.09
1	0	0	0	0	0	0	5	1	489.09
1	0	0	0	-10	0	0	60	1	549.09

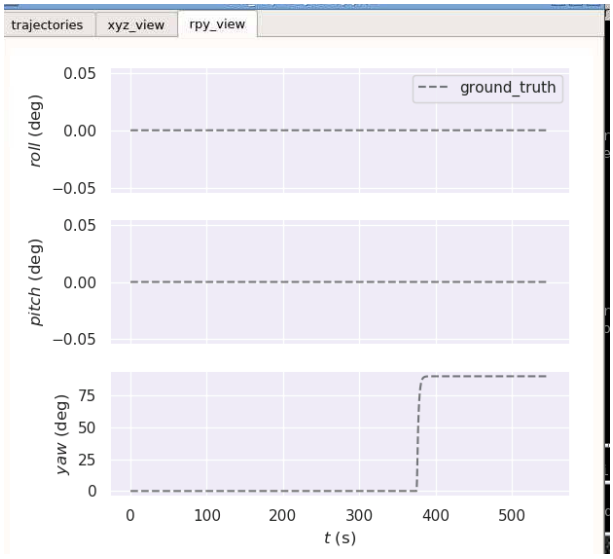
整个过程中的运行轨迹如下图：



三个方向上的移动距离如下图所示：



三个方向上的转动角度如下图所示：

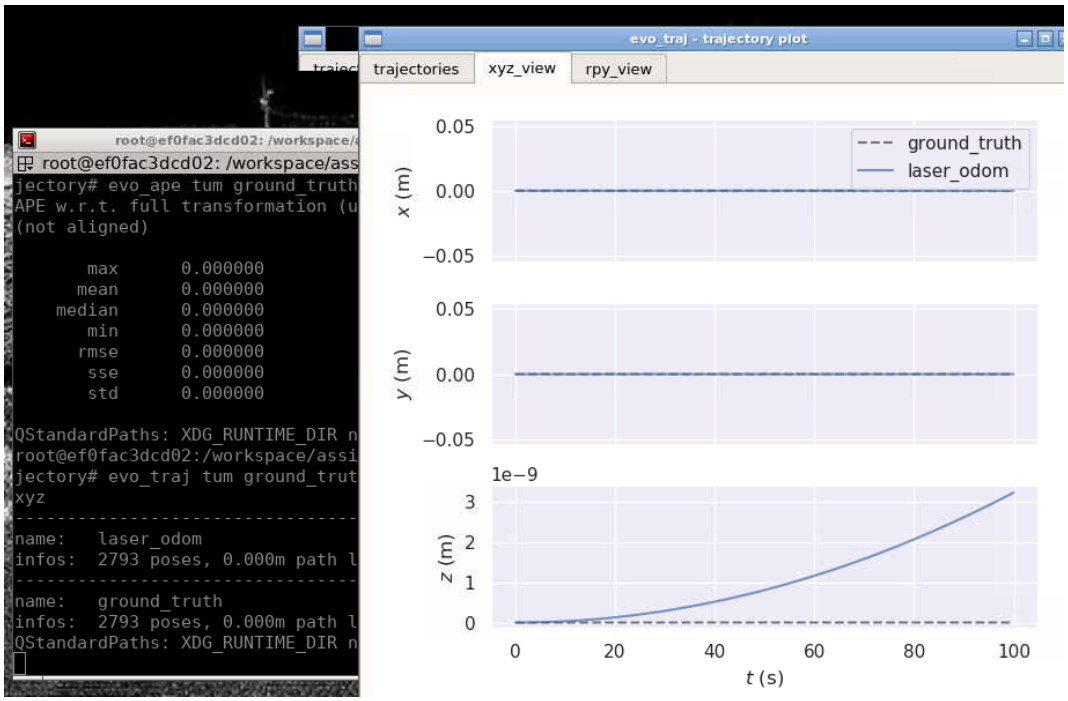


对应播放参数如下：

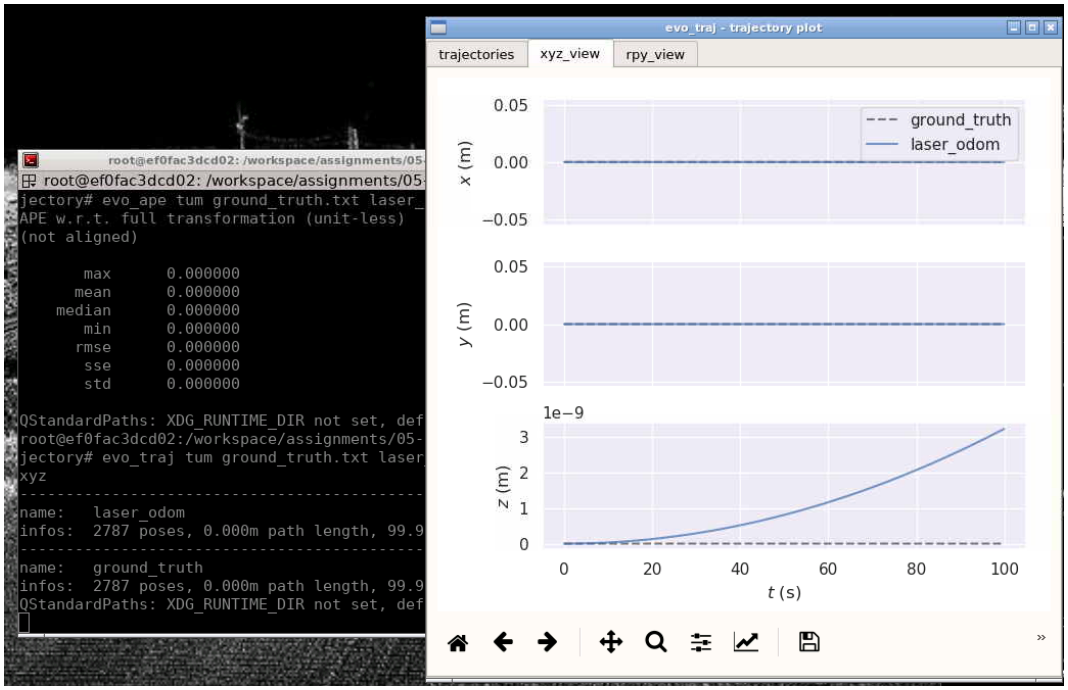
运动状况	播放开始时间	播放时长	播放结束时间
静止	10	100	110
匀速	140	100	240
0.1m/s/s 加速	260	40	300
0.2m/s/s 减速	320	40	360
快速转弯	376	40	416
10m/s/s 加速	430	40	470
10m/s/s 减速	495	40	535

静止

中值法

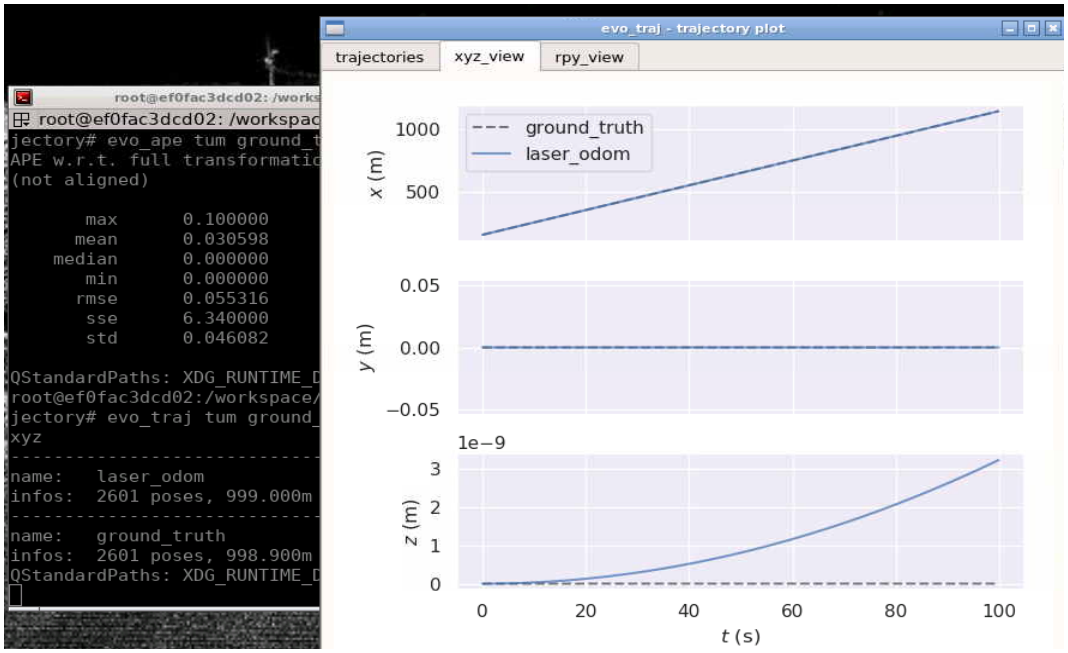


欧拉法

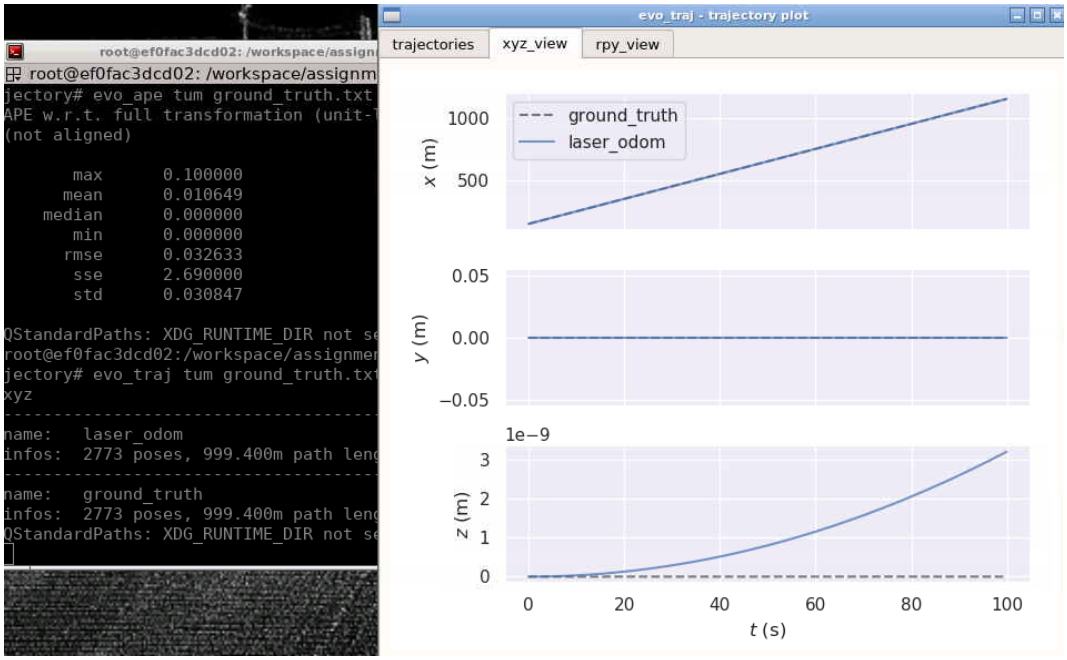

root@ef0fac3dcd02: /workspace/assignments/05-
root@ef0fac3dcd02: /workspace/assignments/05
jectory# evo_ape tum ground_truth.txt laser_
APE w.r.t. full transformation (unit-less)
(not aligned)
max 0.000000
mean 0.000000
median 0.000000
min 0.000000
rmse 0.000000
sse 0.000000
std 0.000000
QStandardPaths: XDG_RUNTIME_DIR not set, def
root@ef0fac3dcd02:/workspace/assignments/05-
jectory# evo_traj tum ground_truth.txt laser
xyz
name: laser_odom
infos: 2787 poses, 0.000m path length, 99.9
name: ground_truth
infos: 2787 poses, 0.000m path length, 99.9
QStandardPaths: XDG_RUNTIME_DIR not set, def
evo_traj - trajectory plot
trajectories
xyz_view
rpy_view
ground_truth
laser_odom
x (m)
y (m)
z (m)
1e−9
t (s)

匀速

中值法

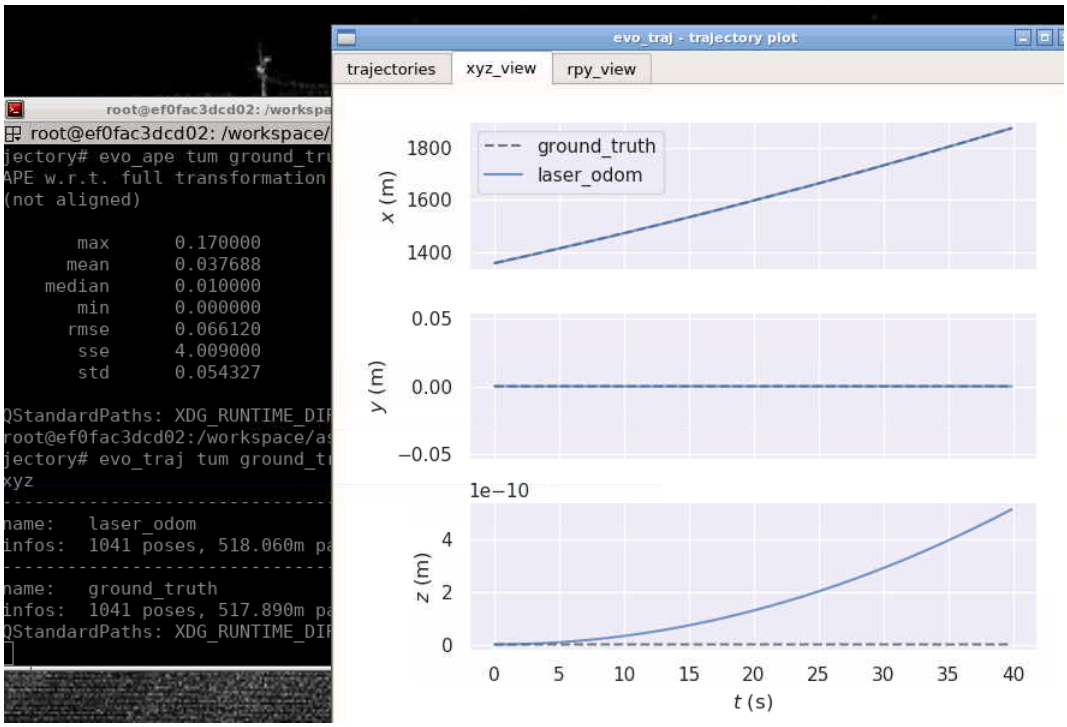

root@ef0fac3dcd02: /works
root@ef0fac3dcd02: /workspac
jectory# evo_ape tum ground_t
APE w.r.t. full transformatio
(not aligned)
max 0.100000
mean 0.030598
median 0.000000
min 0.000000
rmse 0.055316
sse 6.340000
std 0.046082
QStandardPaths: XDG_RUNTIME_D
root@ef0fac3dcd02:/workspace/
jectory# evo_traj tum ground_
xyz
name: laser_odom
infos: 2601 poses, 999.000m
name: ground_truth
infos: 2601 poses, 998.900m
QStandardPaths: XDG_RUNTIME_D
evo_traj - trajectory plot
trajectories
xyz_view
rpy_view
ground_truth
laser_odom
x (m)
y (m)
z (m)
1e−9
t (s)

欧拉法

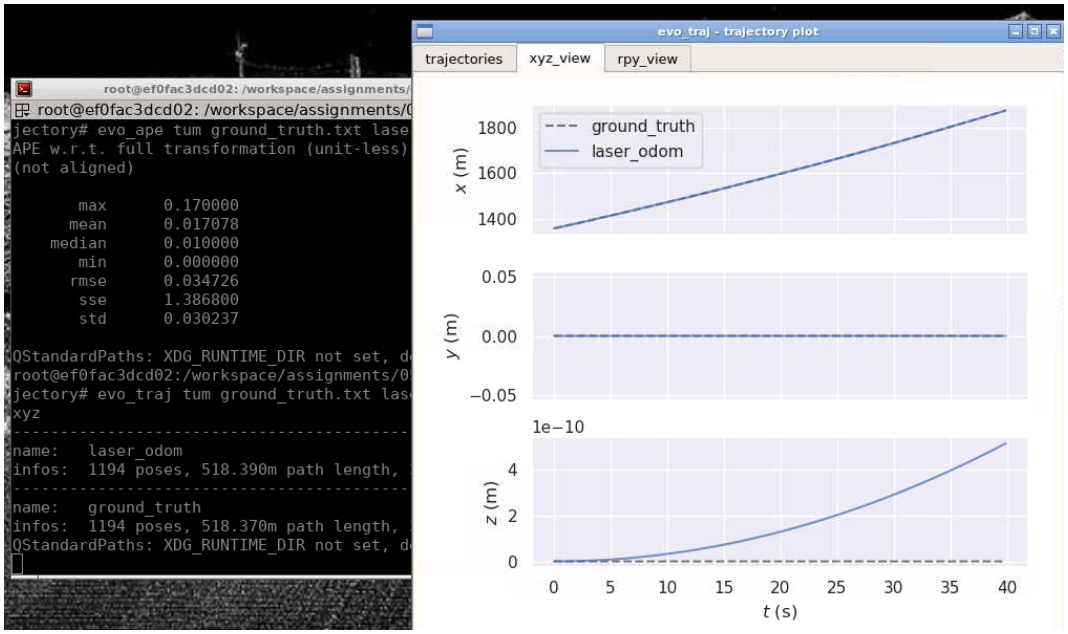


匀加速（加速度很小）

中值法

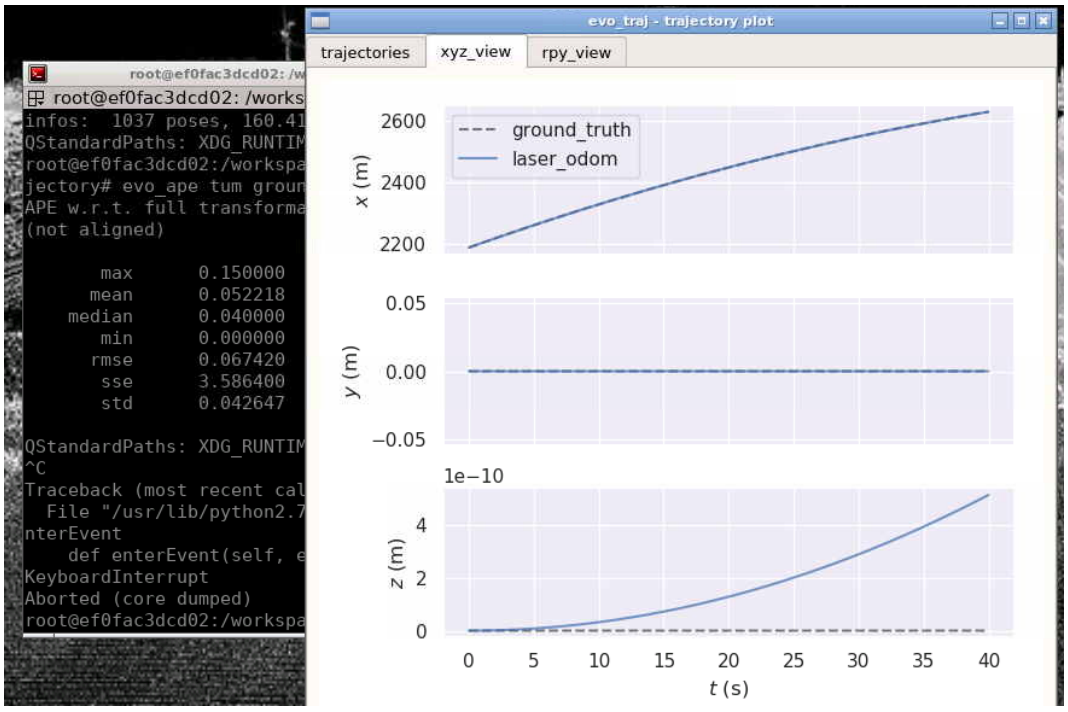


欧拉法

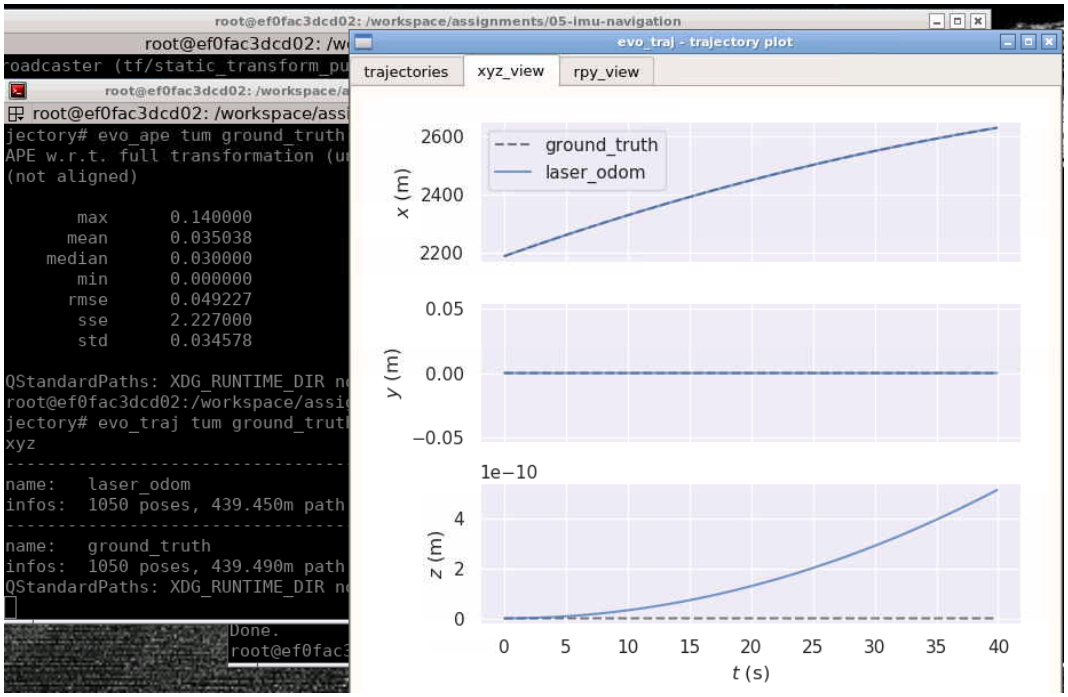


匀减速（加速度很小）

中值法

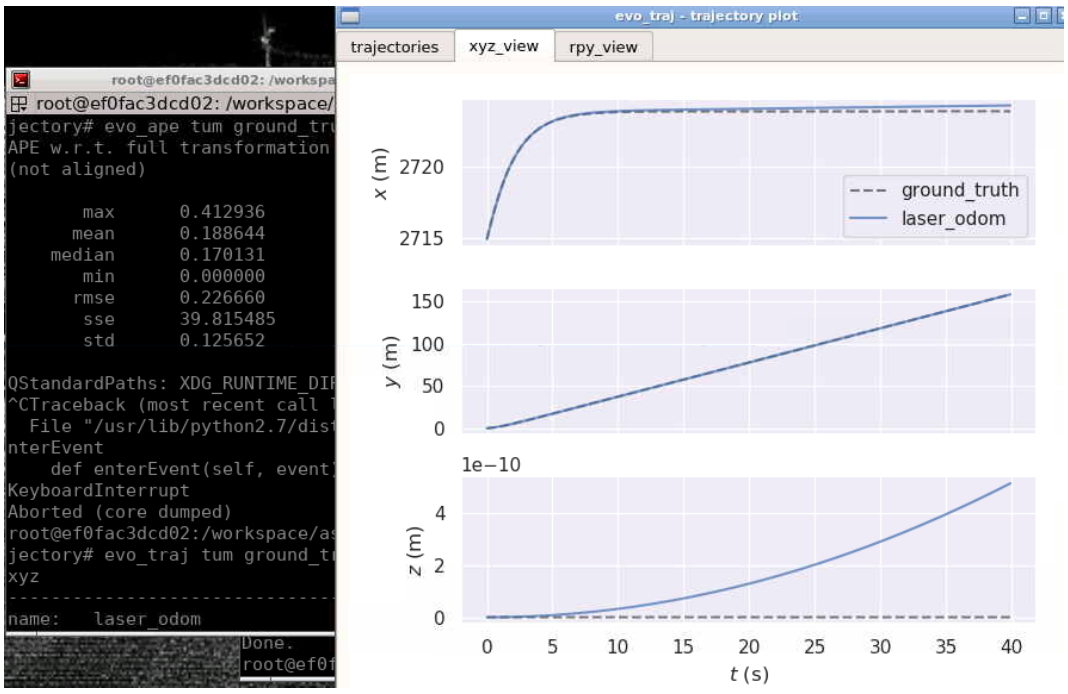


欧拉法

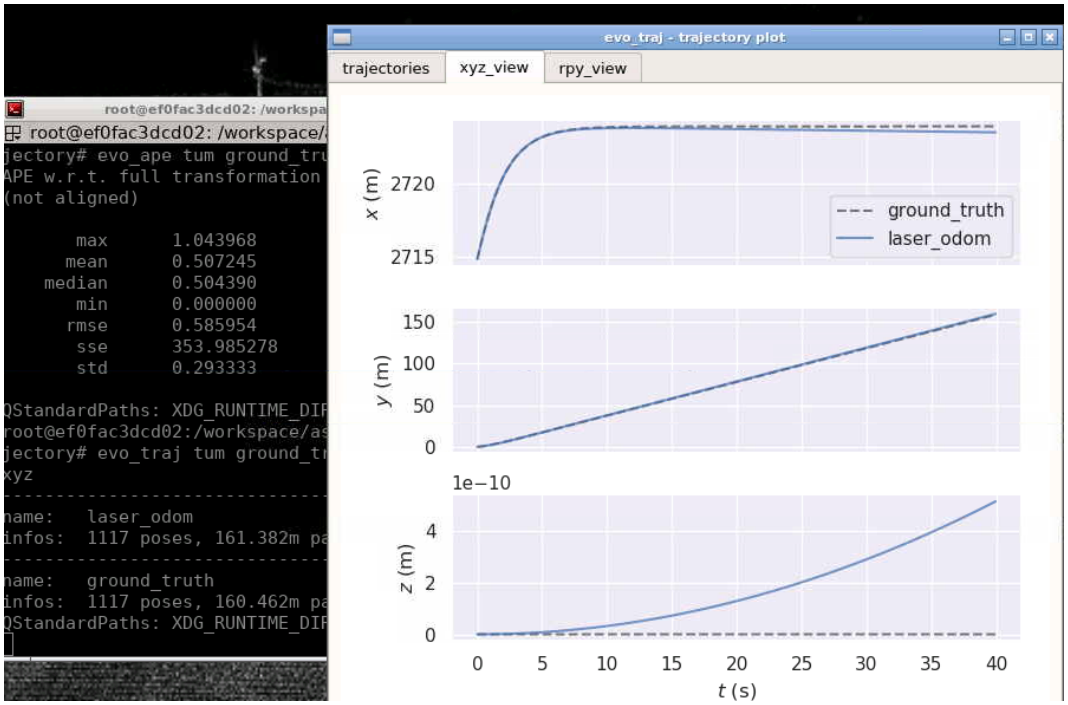


快速转弯（变加速）

中值法

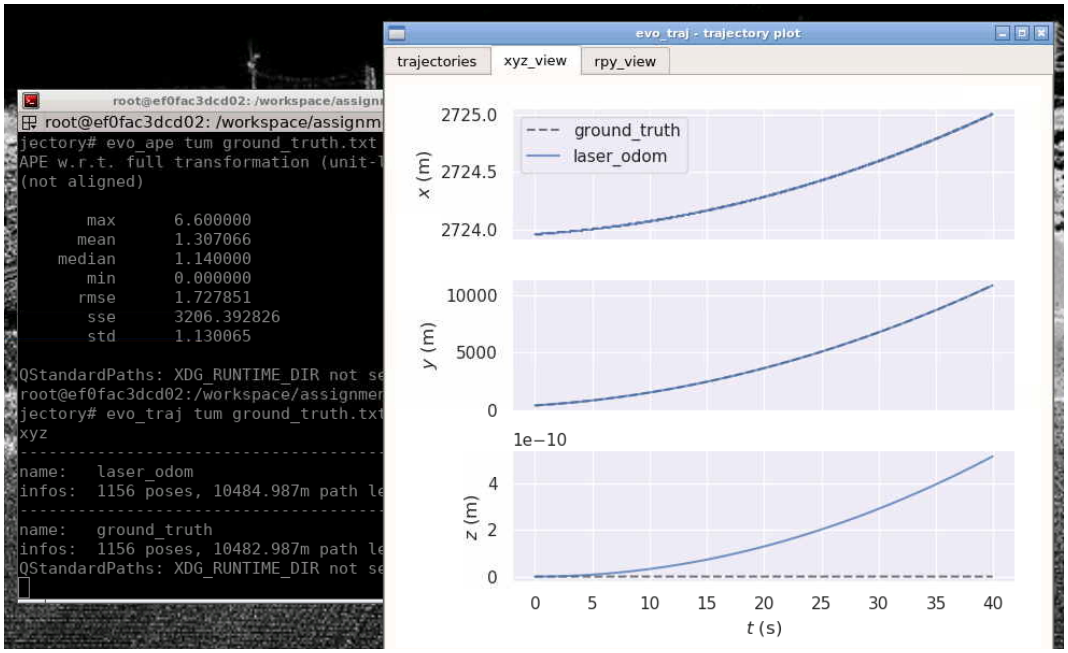


欧拉法

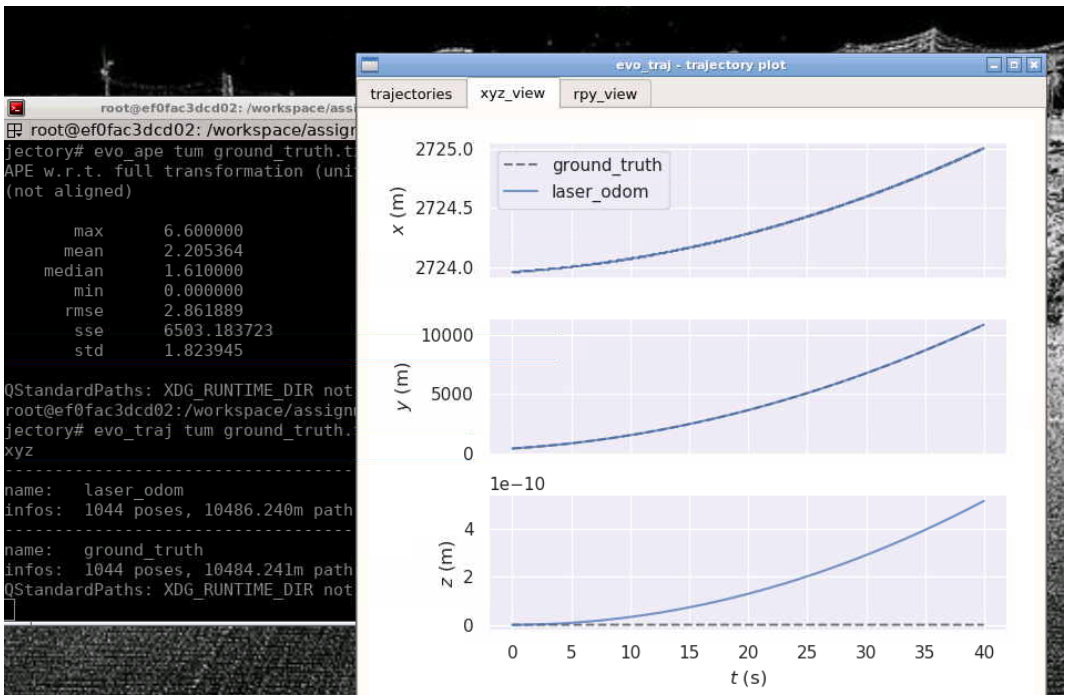


匀加速（加速度比较大）

中值法

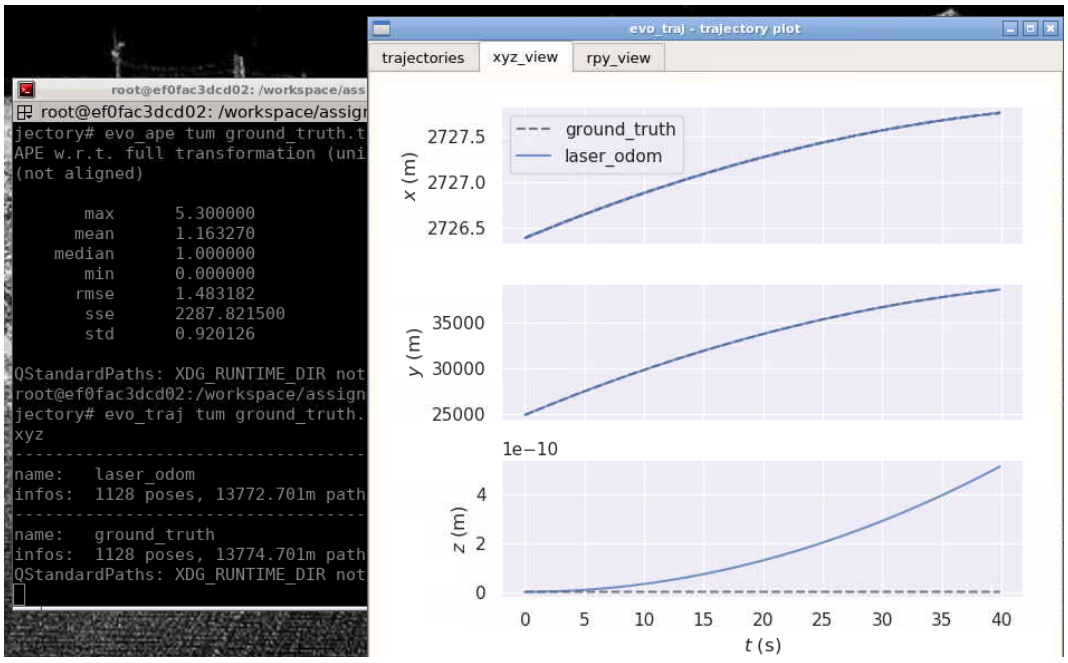


欧拉法

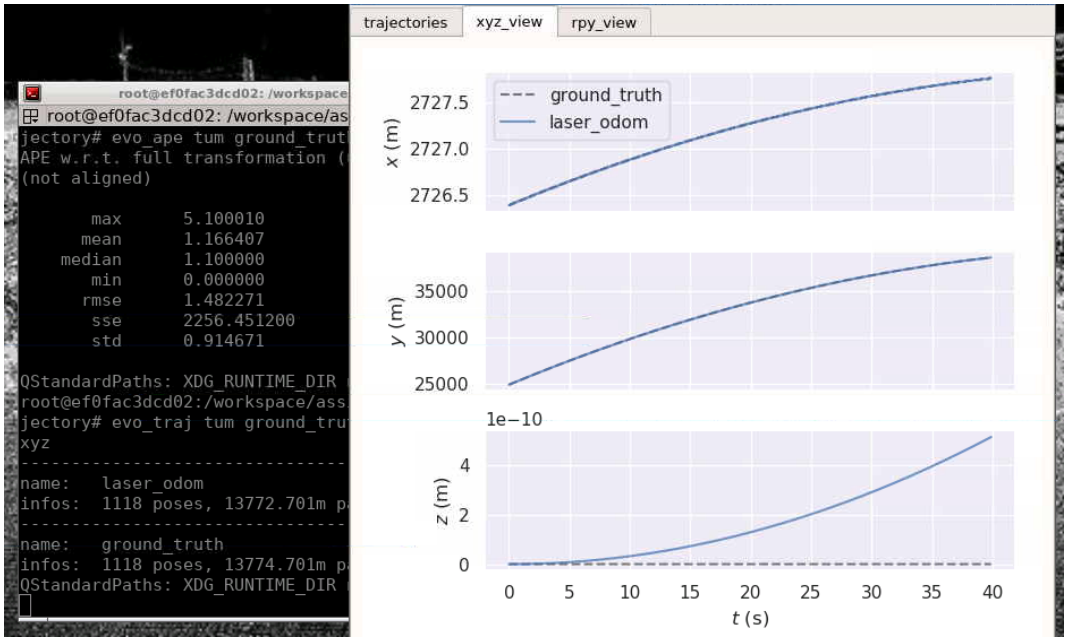


匀减速（加速度比较大）

中值法



欧拉法



精度评价分析

从以上仿真结果可以看出：在静止，匀速，匀加速的情况下都是欧拉法略优于中值法；只有在有转弯的情况下是中值法更性能更好。

理论上在静止，匀速，匀加速的情况下两种算法的结果应该一致，对于现在的仿真结果，我的理解是：变步长积分，从 buffer 里拿数据，那么运算量小的欧拉法采样频率会比中值法高一些，从而导致算法性能略高（还存在争议）。