

# VIO - Homework 2

YaoGeFAD, e-mail: [alexgecontrol@qq.com](mailto:alexgecontrol@qq.com)

June 23, 2019

## 说明:

- 文本使用在线 L<sup>A</sup>T<sub>E</sub>X 编辑器 <https://www.overleaf.com>生成

1. **Allan 方差标定曲线生成** 设置 IMU 仿真代码中不同的参数, 生成 Allan 方差标定曲线:

\* **ANS:**

使用工具 `imu_utils` 生成 Allan 方差标定曲线流程如下:

- (a) 修改 ROS 版 `vio_data_simulation` 中 `src/gener_alldata.cpp` 文件中的输出路径, 使其与系统文件路径相匹配.
- (b) 在标准版 `vio_data_simulation` 中 `src/imu.cpp` 文件中实现两种积分方法, 编译, 并使用 `python_tool` 下 `draw_trajectory.py` 文件验证实现的正确性.
- (c) 将前述两积分实现移植到 ROS 版本中.
- (d) 设置 `src/param.h` 下的 IMU 参数, 使用 ROS 版本 `vio_data_simulation` 生成 ROS bag.
- (e) 首先加入 Package `code_utils` 进行编译, 随后再加入 `imu_utils` 进行编译.
- (f) 修复 `code_utils` 中头文件引用错误 [sumpixel\\_test.cpp](#)
- (g) 使用 `imu_utils` 分析所得的 ROS bag, 绘制 Allan variance curve.

为了不使用 MATLAB 的绘图功能, 首先使用 Python 将工具所提供的 `draw_allan.m` 重新实现. 具体实现细节请参考链接所指向的 **GitHub Repo**.

使用中值积分进行 IMU 仿真, 所得的 Allan 方差曲线如图1所示. 该图片已同步到 GitHub Repo, 可通过上述 Python 脚本工具复现:

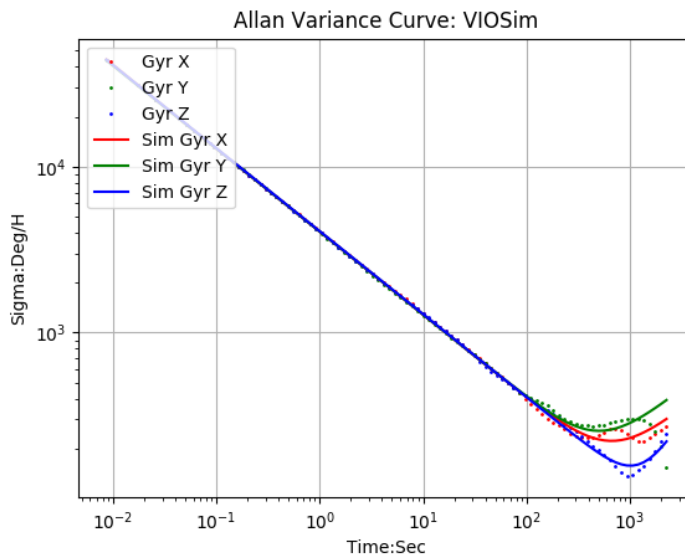


Figure 1: Allan Variance Curve for VIO Sim IMU @ 200Hz

相应的 YAML 文件请参考 [VIOSim\\_imu\\_param.yaml](#)

## 2. 中值积分的实现 将 IMU 仿真代码中的欧拉积分替换成中值积分

\* **ANS:** 中值积分的 C++ 实现如下. 请参考链接所指向的 **GitHub Repo** 链接以获取具体实现细节

(a) **Standard Version.**

(b) **ROS Version.**

实现时对参考代码进行了重构, 将欧拉积分与中值积分全部封装为函数, 同时依靠宏定义, 使具体实现可在编译时通过宏指定.

使用欧拉积分进行 IMU 仿真, 所得的 Trajectory 曲线如图2所示.

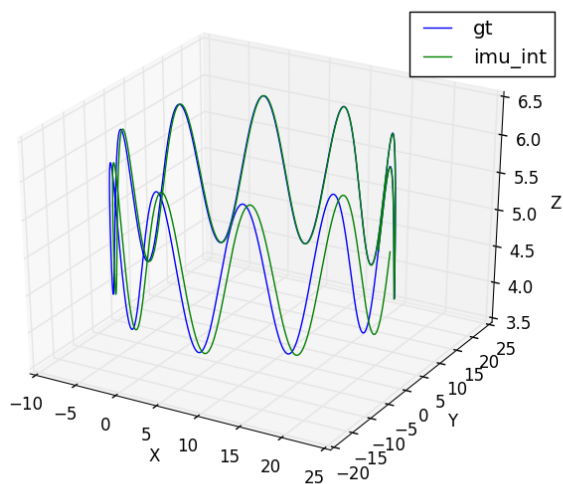


Figure 2: Trajectory using Euler Integration

使用中值积分进行 IMU 仿真, 所得的 Trajectory 曲线如图3所示.

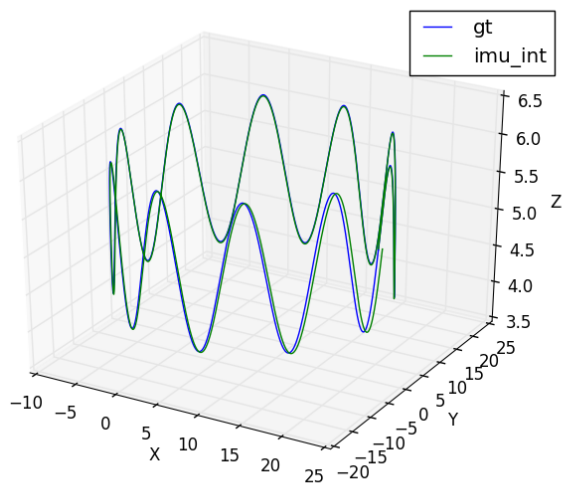


Figure 3: Trajectory using Euler Integration

## References

- [1] Steven Lovegrove (2013) *Spline Fusion: A Continuous-Time Representation for Visual Inertial Fusion with Application to Rolling Shutter Camera*, BMVC.