Continuous Integration over SO(3) for IMU Preintegration

Cedric Le Gentil *and* Teresa Vidal-Calleja

University of Technology Sydney

ROBOTICS SCIENCE AND SYSTEMS

UTS:CAS CENTRE FOR AUTONOMOUS SYSTEMS

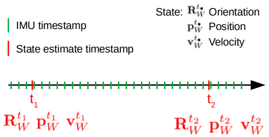
July 2021

Context: IMU Preintegration

Classic integration of inertial measurement from an IMU allows for the prediction of a system's pose from time t_1 to t_2 given a set of initial conditions at time t_1. Consequently, any change in the initial conditions implies the recomputation of the integrals.

Preintegration techniques combine the inertial data into pseudo-measurements that are **independent from the initial pose** of the system. It allows for efficient optimisation-based estimation by preventing numerous integral computations.

Inertial data



Preintegrated measurements

$$[t_1,t_2] \longrightarrow \Delta\mathbf{R}_{t_1}^{t_2}\ \Delta\mathbf{v}_{t_1}^{t_2}\ \Delta\mathbf{p}_{t_1}^{t_2}$$

$$\mathbf{R}_W^{t_2} = \mathbf{R}_W^{t_1}\Delta\mathbf{R}_{t_1}^{t_2}$$

$$\mathbf{v}_W^{t_2} = \mathbf{v}_W^{t_1} + \mathbf{g}\Delta t + \mathbf{R}_W^{t_1}\Delta\mathbf{v}_{t_1}^{t_2}$$

$$\mathbf{p}_W^{t_2} = \mathbf{p}_W^{t_1} + \mathbf{v}_W^{t_1}\Delta t + \frac{\mathbf{g}\Delta t^2}{2} + \mathbf{R}_W^{t_1}\Delta\mathbf{p}_{t_1}^{t_2}$$

Problematic: Dynamics of the System

A system's dynamics is ruled by a set of differential equations.

Unfortunately, the rotational part does not possess any known general analytical solution. Numerical motion-model-based approaches are generally found in the literature. But the drawback of these approaches is the accuracy. To address this issue, we propose a model-less method using **Gaussian Processes** (GP) to represent the system's dynamics.

Differential equations

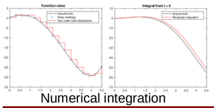
$$\dot{\mathbf{R}}_W^t(t) = \mathbf{R}_W^t(t)\boldsymbol{\omega}(t)^\wedge \qquad (1)$$

$$\dot{\mathbf{v}}_W^t(t) = \mathbf{f}_W(t) \qquad (2)$$

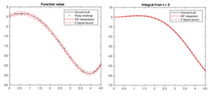
$$\dot{\mathbf{p}}_W^t(t) = \mathbf{v}_W^t(t) \qquad (3)$$

$\mathbf{f}_W(t)$: angular velocity
$\boldsymbol{\omega}(t)$: acceleration

Numerical integration vs. Gaussian Process integration



Numerical integration



Gaussian Process integration

Method: Continuous Integration over SO(3)

Our method estimates the inducing (or training) values of the GPs that represent the derivative of the system's orientation in its minimum representation.

Relying on equations (4), (5), and (6) it is possible to constrain the GPs' inducing values with respect to the observed instantaneous angular velocity readings of the IMU. It is a non-linear least-square optimisation problem solved with the Levenberg-Marquardt algorithm.

This continuous integration method is incorporated in a preintegration pipeline that first infers the rotational part of the preintegrated measurements. These are used to reproject the acceleration data in a unique referential frame. From there, the position and velocity components are analytically inferred using GPs as in to equation (5).

Gaussian Process inference and integral inference

$$\dot{\mathbf{r}}_{t_1}^t(t) = \boldsymbol{\beta}(t)^\top\boldsymbol{\rho} \qquad (4)$$

$$\mathbf{r}_{t_1}^t(t) = \left(\int_{t_1}^{t}\boldsymbol{\beta}(x)\partial x\right)^\top\boldsymbol{\rho} \qquad (5)$$

$\boldsymbol{\rho}$: estimated inducing values

$$\mathbf{r}_{t_1}^t(t) = \log(\mathbf{R}_{t_1}^t(t))^\vee$$

$$\boldsymbol{\beta}(t) = \mathbf{k}_{r_j}(t,\mathbf{t})\left[\mathbf{K}_{r_j}(\mathbf{t},\mathbf{t}) + \sigma_r^2\mathbf{I}\right]^{-1}$$

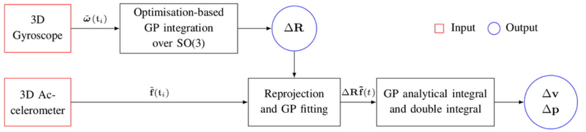
Relationship between orientation and angular velocity

$$\mathbf{J}_r(\mathbf{r}_{t_1}^t(t))\dot{\mathbf{r}}_{t_1}^t(t) = \boldsymbol{\omega}(t) \qquad (6)$$

$\mathbf{J}_r$: right Jacobian of SO(3)

Proposed preintegration pipeline to generate the UGPMs

(Unified Gaussian Preintegrated Measurements)



In real-world scenarios, the IMU data contain slow-varying additive biases. While these are not accurately known at the time of preintegration, the proposed method provides a postintegration correction mechanism that allows for both the correction of the biases and the potential inter-sensor time-shift for inertial-aided navigation systems. It is based on a first order Taylor expansion of the UGPMs.

Our paper contains the full derivation of the method as well as a succinct presentation of the required background knowledge.

Our code is available open-source at:

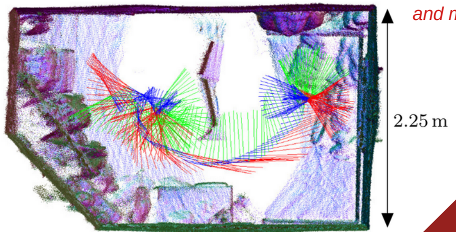
https://github.com/UTS-CAS/ugpm

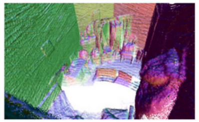
Real-World Validation

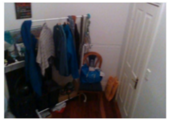
Atop the simulation experiments, we integrated the UGPMs in a lidar-inertial localisation and mapping framework called IN2LAAMA. Equipped with a 16-beam spinning lidar, the system is able to estimate its trajectory and the IMU biases while creating an accurate 3D map of the environment.

While GP regression has a cubic computational complexity, we introduce a per-chunk computation scheme that allows for linear **real-time** computations regardless of the size of the temporal integration window. Timing results are present in the paper.

Lidar-inertial localisation and mapping







C. Le Gentil, T. Vidal-Calleja, S. Huang. *"IN2LAAMA: Inertial Lidar Localization Autocalibration and Mapping". IEEE Transactions on Robotics, 2021.*

Simulation Results

Extensive experiments have been conducted. It includes analyses of our method's accuracy, robustness to noise, and the precision of the postintegration correction mechanism.

Here we show an extract of the accuracy experiments. One can see that our method significantly **outperforms the current state-of-the-art by four folds** and the standard preintegration by more than an order of magnitude.

Absolute rotation and position errors in fast motion scenarios

Our proposed method → UGPM; Current state-of-the-art → GPM; Standard preintegration → PM

Δt [ms] $(t_2 - t_1)$	Error type	PM	GPM	UGPM
0.5	Rot abs. er. [mrad]	66.7 ± 25	8.2 ± 2.61	**1.18** ± 0.485
	Pos abs. er. [mm]	300 ± 111	14.6 ± 8.16	**2.89** ± 1.5
1	Rot abs. er. [mrad]	49.3 ± 22.3	5.74 ± 2.23	**1.59** ± 0.759
	Pos abs. er. [mm]	574 ± 211	42.4 ± 20.3	**8.7** ± 3.62