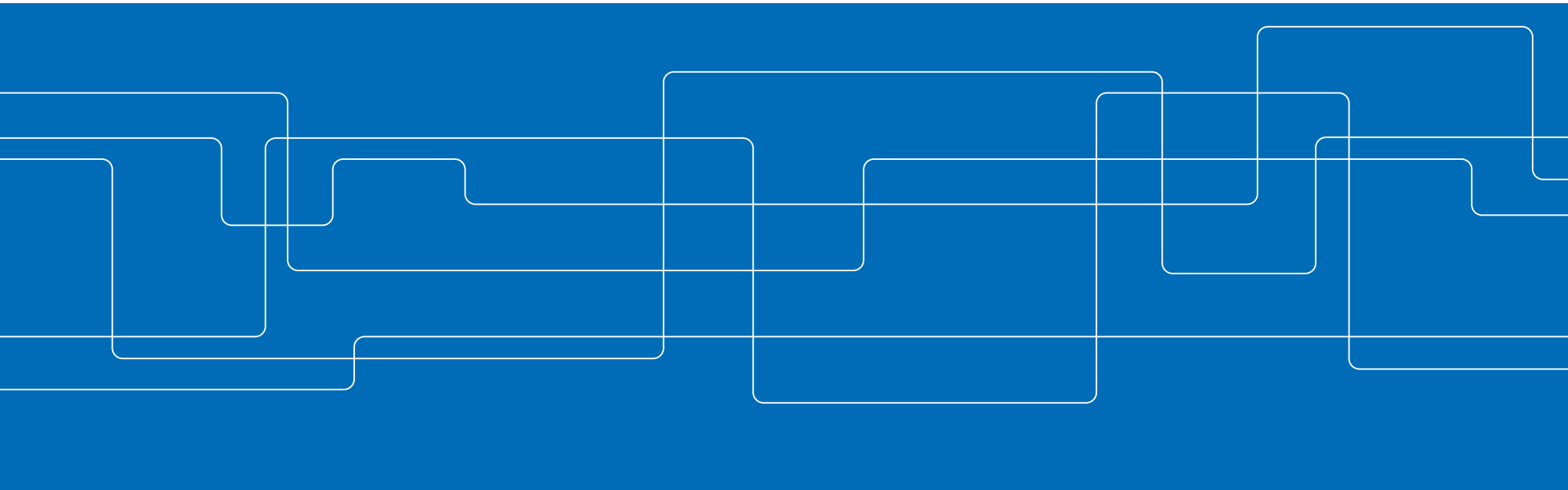
WASP workshop on Smart Localization Systems 2018 Seminar

Self-Calibration of Sensor Fusion Systems

Håkan Carlsson

September 4, 2018

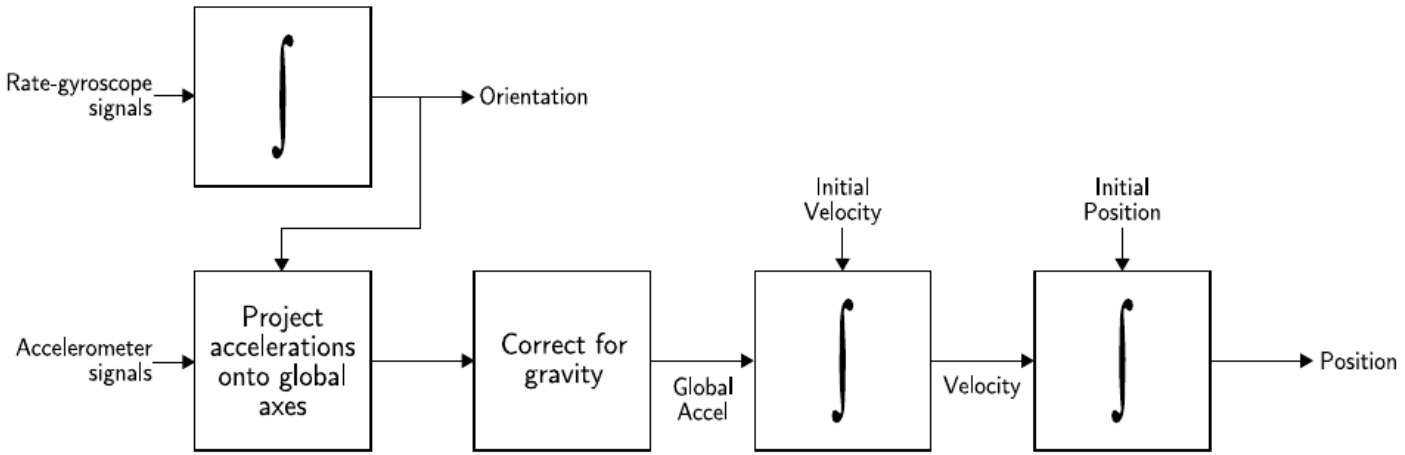


Introduction

- Sensors are **cheaper to manufacture** than before and **computing capabilities** are readily available.
- Instead of using expensive hardware, use an **redundant amount** of cheap sensors.
- **Sensor fusion algorithms** can be used to:
  - Estimate non-measurable quantities.
  - Noise reduction.
- The variance of the estimate often depends on the accuracy of the **static parameter values**.
- A separate **calibration** procedure for static parameter values may impose a high cost on the system.

Example: Dead Reckoning from Inertial Measurement Units

- IMU = 3 accelerometers (x, y, z) + 3 gyroscopes (x, y, z)



- Three integrations $\rightarrow$ error grows as $\mathcal{O}(N^3)$
- Using Xsens Mtx IMU for 60 seconds $\rightarrow$ position off by 150 m. [1]

[1] An introduction to inertial navigation, Oliver J. Woodman, 2007, ISSN 1476-2986

Example: Employ Sensor Fusion on Array of Accelerometers

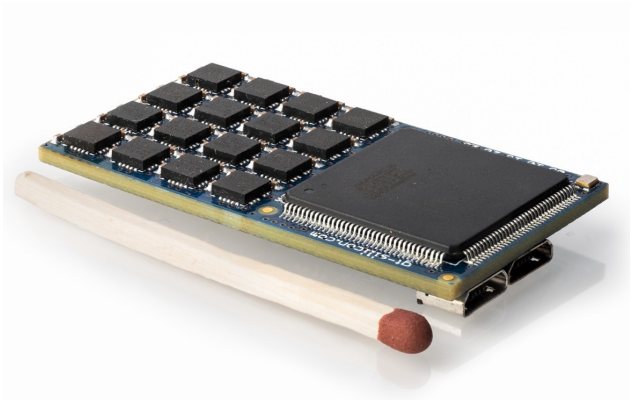
- Sensor array of 32 IMUs on a solid body
- Accelerometer and gyroscope model:

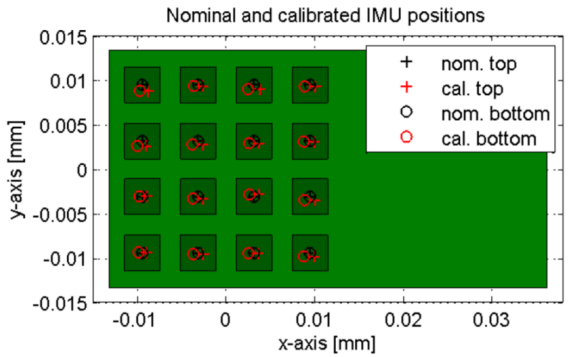
$$\boldsymbol{y}_a = \boldsymbol{s} + \underbrace{\boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{r})}_{\text{Centrifugal}} + \underbrace{\dot{\boldsymbol{\omega}} \times \boldsymbol{r}}_{\text{Euler}} + \boldsymbol{\epsilon}_a,$$

$$\boldsymbol{y}_g = \boldsymbol{\omega} + \boldsymbol{\epsilon}_g,$$

- Fusion via a ML Estimator[2]

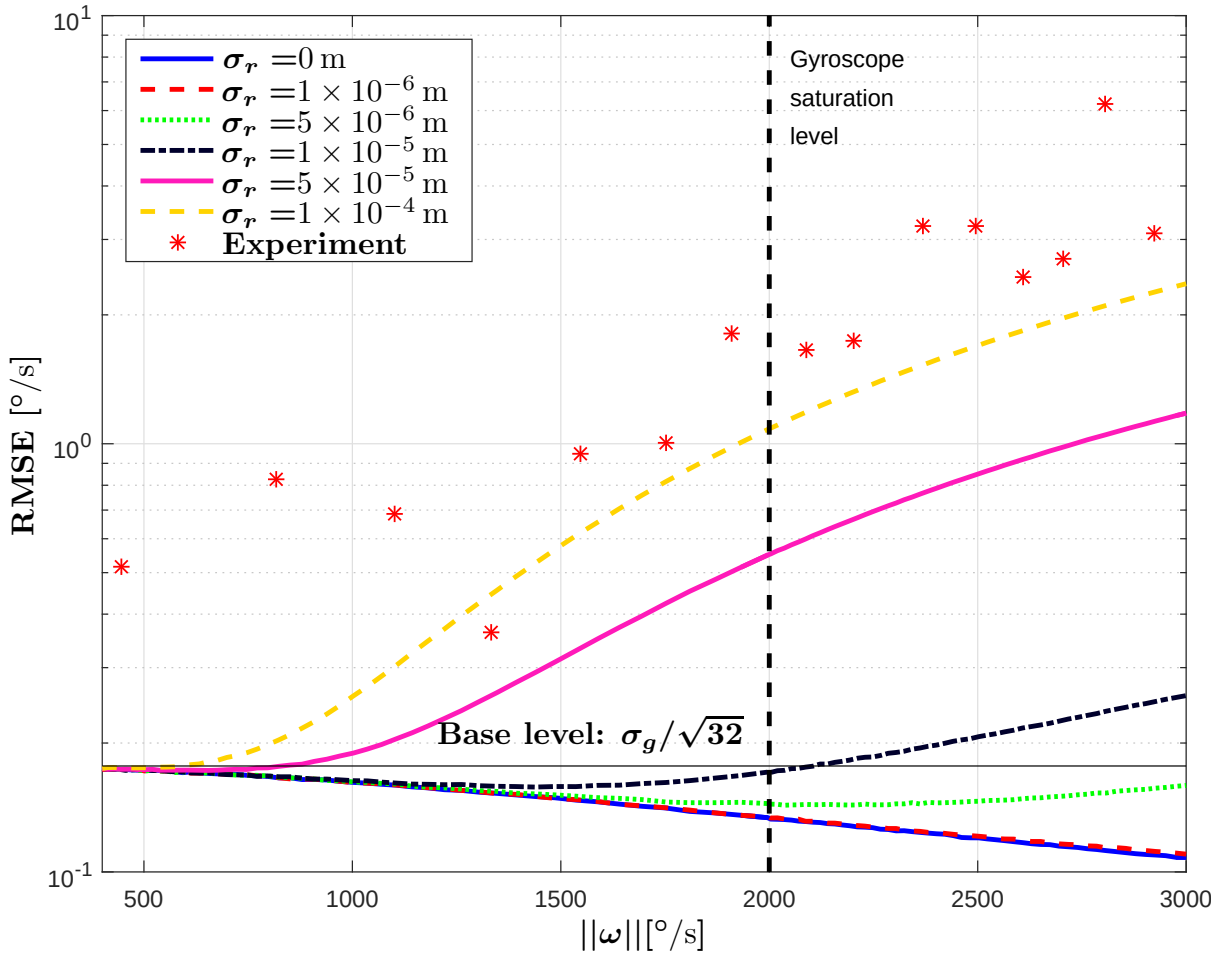
$$\left\{\hat{\boldsymbol{s}}, \hat{\boldsymbol{\omega}}, \hat{\dot{\boldsymbol{\omega}}}\right\} = \arg\max_{\boldsymbol{s},\boldsymbol{\omega},\dot{\boldsymbol{\omega}}} p(\boldsymbol{y}; \boldsymbol{s}, \boldsymbol{\omega}, \dot{\boldsymbol{\omega}}, \boldsymbol{r}_{1:A}).$$





[2] Skog et al., "Inertial Sensor Arrays, Maximum Likelihood, and Cramér-Rao Bound".

Maximum Likelihood IMU sensor fusion[2]



[3]Skog et al., "Inertial Sensor Arrays, Maximum Likelihood, and Cramér-Rao Bound".

Self-calibration of the IMU array

- Joint estimation for N time-samples with no time-dependence. $(\boldsymbol{x} = (\boldsymbol{s}, \boldsymbol{\omega}, \dot{\boldsymbol{\omega}}))$:

$$\{\hat{\boldsymbol{x}}_{1:N}, \hat{\boldsymbol{r}}_{1:A}\} = \arg\max_{\boldsymbol{r}_{1:A}} \prod_{n=1}^{N} \max_{\boldsymbol{x}_n} p(\boldsymbol{y}_n; \boldsymbol{x}_n, \boldsymbol{r}_{1:A})$$

- ML-estimator by using the Block Coordinate Descent[4] method.

$$\hat{\boldsymbol{x}}_n^{(i+1)} = \arg\max_{\boldsymbol{x}_n} p(\boldsymbol{y}_n; \boldsymbol{x}_n, \hat{\boldsymbol{r}}_{1:A}^{(i)}), \quad n = 1 \dots N$$

$$\hat{\boldsymbol{r}}_{1:A}^{(i+1)} = \arg\max_{\boldsymbol{r}_{1:A}} p(\boldsymbol{y}_{1:N}; \hat{\boldsymbol{x}}_{1:N}^{(i+1)}, \boldsymbol{r}_{1:A})$$

[4] Tseng, "Convergence of a Block Coordinate Descent Method for Nondifferentiable Minimization".

Estimation of static positions

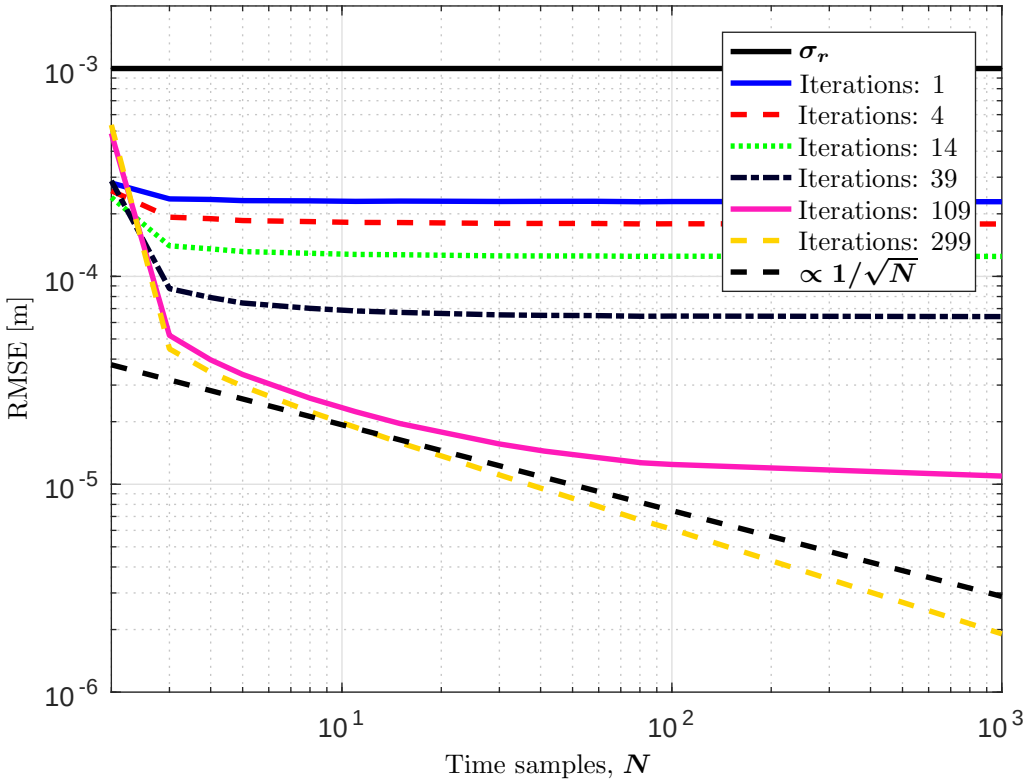


Figure: RMSE of $\boldsymbol{r}_{(1:A)}$, 10^4 realizations, $||\boldsymbol{\omega}|| = 2000^\circ/\text{s}$, $||\dot{\boldsymbol{\omega}}|| = 2000^\circ/\text{s}^2$, dependence on the number of time samples.[5]

[5] Carlsson, Skog, and Jaldén, "On-the-fly geometric calibration of inertial sensor arrays".

Current work: Bayesian self-calibration

- Usually, there is **prior knowledge** in the static parameter values. Motivation for Bayesian inference:

$$p(\boldsymbol{r}_{1:A}|\boldsymbol{y}_{1:N}) = \frac{p(\boldsymbol{y}_{1:N}|\boldsymbol{r}_{1:A})p(\boldsymbol{r}_{1:A})}{p(\boldsymbol{y}_{1:N})},$$

- $p(\boldsymbol{r}_{1:A}|\boldsymbol{y}_{1:N})$ is the posterior distribution of the relative distances.
- $p(\boldsymbol{y}_{1:N}|\boldsymbol{r}_{1:A})$ is the likelihood distribution.
- $p(\boldsymbol{y}_{1:N})$ is the marginalized likelihood.
- $p(\boldsymbol{r}_{1:A})$ is the prior distribution.
- Idea: sample the posterior using the **Metropolis** Algorithm[6]

[6] Metropolis et al., "Equation of State Calculations by Fast Computing Machines".

Metropolis algorithm

- Construct a Markov Chain that converges to $p(\boldsymbol{r}_{1:A}|\boldsymbol{y}_{1:N})$.
- Can be constructed using accept-reject sampling:
  - Sample a proposal $\boldsymbol{r}_{1:A}^{'} \sim q(\cdot|\boldsymbol{r}_{1:A}^{(i)})$
  - Set $\boldsymbol{r}_{1:A}^{i+1} = \boldsymbol{r}_{1:A}^{'}$ with probability:

$$\min\left(1, \frac{p(\boldsymbol{r}_{1:A}^{'}|\boldsymbol{y}_{1:N})}{p(\boldsymbol{r}_{1:A}^{(i)}|\boldsymbol{y}_{1:N})}\right) = \min\left(1, \frac{p(\boldsymbol{y}_{1:N}|\boldsymbol{r}_{1:A}^{'})p(\boldsymbol{r}_{1:A}^{'})}{p(\boldsymbol{y}_{1:N}|\boldsymbol{r}_{1:A}^{(i)})p(\boldsymbol{r}_{1:A}^{(i)})}\right)$$

  - Otherwise set $\boldsymbol{r}_{1:A}^{(i+1)} = \boldsymbol{r}_{1:A}^{(i)}$
- But $p(\boldsymbol{y}_{1:N}|\boldsymbol{r}_{1:A}^{'})$ is not analytically available, have to marginalize out the motion:

$$p(\boldsymbol{y}_{1:N}|\boldsymbol{r}_{1:A}^{'}) = \prod_{n=1}^{N} \int p(\boldsymbol{y}_n|\boldsymbol{x}_n, \boldsymbol{r}_{1:A}) p(\boldsymbol{x}_n) \mathrm{d}\boldsymbol{x}_n$$

Marginalization of x

- Replace with an Monte Carlo estimate:

$$\hat{p}(\boldsymbol{y}|\boldsymbol{r}_{1:A}) = \frac{1}{K}\sum_{k}^{K} p(\boldsymbol{y}|\boldsymbol{x}^{(k)}, \boldsymbol{r}_{1:A}), \quad \boldsymbol{x}^{(k)} \sim p(\boldsymbol{x})$$

- This is the **Pseudo-marginal Metropolis-Hastings** Algorithm[7]
- A proposal $p(\boldsymbol{x})$ can be a Gaussian with mean

$$\hat{\boldsymbol{x}} = \arg\max_{\boldsymbol{x}} p(\boldsymbol{y}; \boldsymbol{x}, \boldsymbol{r}_{1:A}^{(0)}) = h(\boldsymbol{y}, \boldsymbol{r}_{1:A}^{(0)}).$$

and variance

$$\operatorname{Var}\hat{\boldsymbol{x}} = \operatorname{Var} h(\boldsymbol{y}, \boldsymbol{r}_{1:A}^{(0)}) \approx \boldsymbol{A}_{\boldsymbol{y}} \operatorname{Var}(\boldsymbol{y}) \boldsymbol{A}_{\boldsymbol{y}}^{T} + \boldsymbol{A}_{\boldsymbol{r}_{1:A}} \operatorname{Var}(\boldsymbol{r}_{1:A}) \boldsymbol{A}_{\boldsymbol{r}_{1:A}}^{T}$$

[7] Andrieu and Roberts, "The pseudo-marginal approach for efficient Monte Carlo computations".

Preliminary Results

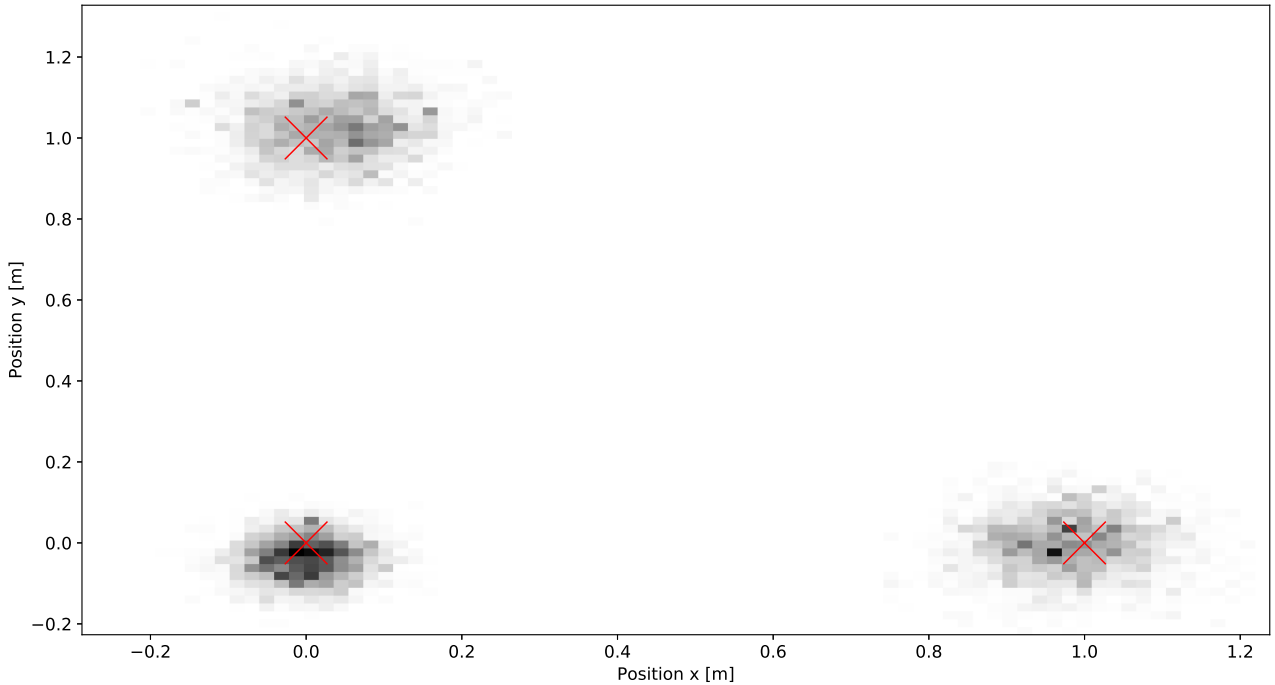


Figure: Samples of posterior distribution (3 IMUs), 100 time-steps, 10000 marginalization samples, and 50000 MCMC iterations.

Preliminary Results