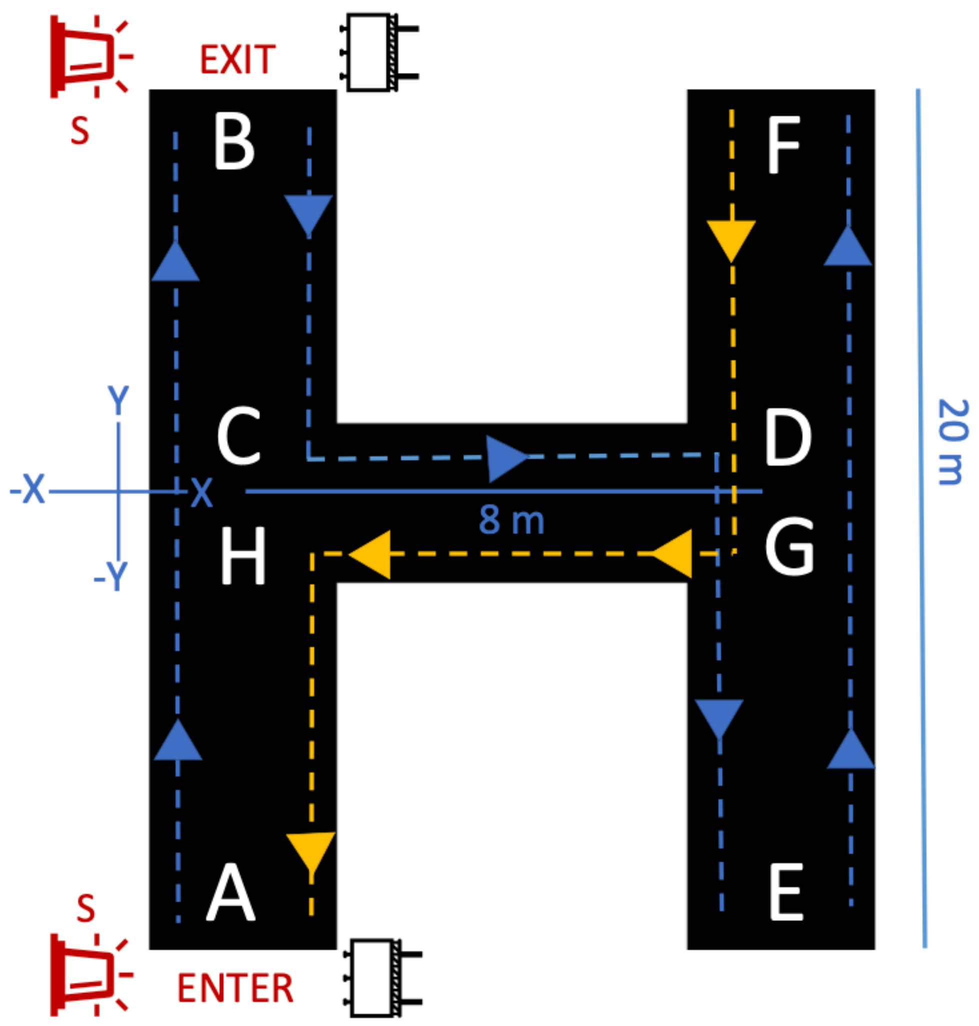
Vehicle Detection and Tracking for Automated Driving Test

Abijith J. Kamath
Department of Electrical Engineering
Indian Institute of Science, Bangalore
Email: abijithj@iisc.ac.in

This work is part of E1 244 Detection and Estimation Theory, Feb-May 2021 Final Project

Objective

Detection and Estimation Problems



- Detection: Vehicle at the entry and exit,
- Estimation: Vehicle position through the course.

Detector Specifications

- LDRs used at entry and exit,
- Measurements of the form:

$$x_m[n] = \begin{cases} A + B_t + w_m[n], & \text{vehicle absent,} \\ B_t + w_m[n], & \text{vehicle present,} \end{cases}$$

- The vehicle detection problem:

$$\begin{aligned} \text{(vehicle absent) } \mathcal{H}_0 &: x_m[n] = A + B_t + w_m[n], \\ \text{(vehicle present) } \mathcal{H}_1 &: x_m[n] = B_t + w_m[n], n = 0, 1, \cdots, N-1; m = 1, 2, \cdots, M. \end{aligned}$$

Detector at the Entry

- D_{entry} requires constant P_D and the smallest P_{FA},

$$\underset{R_1}{\text{minimise}}\ P_{FA},$$
$$\text{subject to } P_D = \beta.$$

- The solution is a ratio test, choose $\mathcal{H}_1$ if:

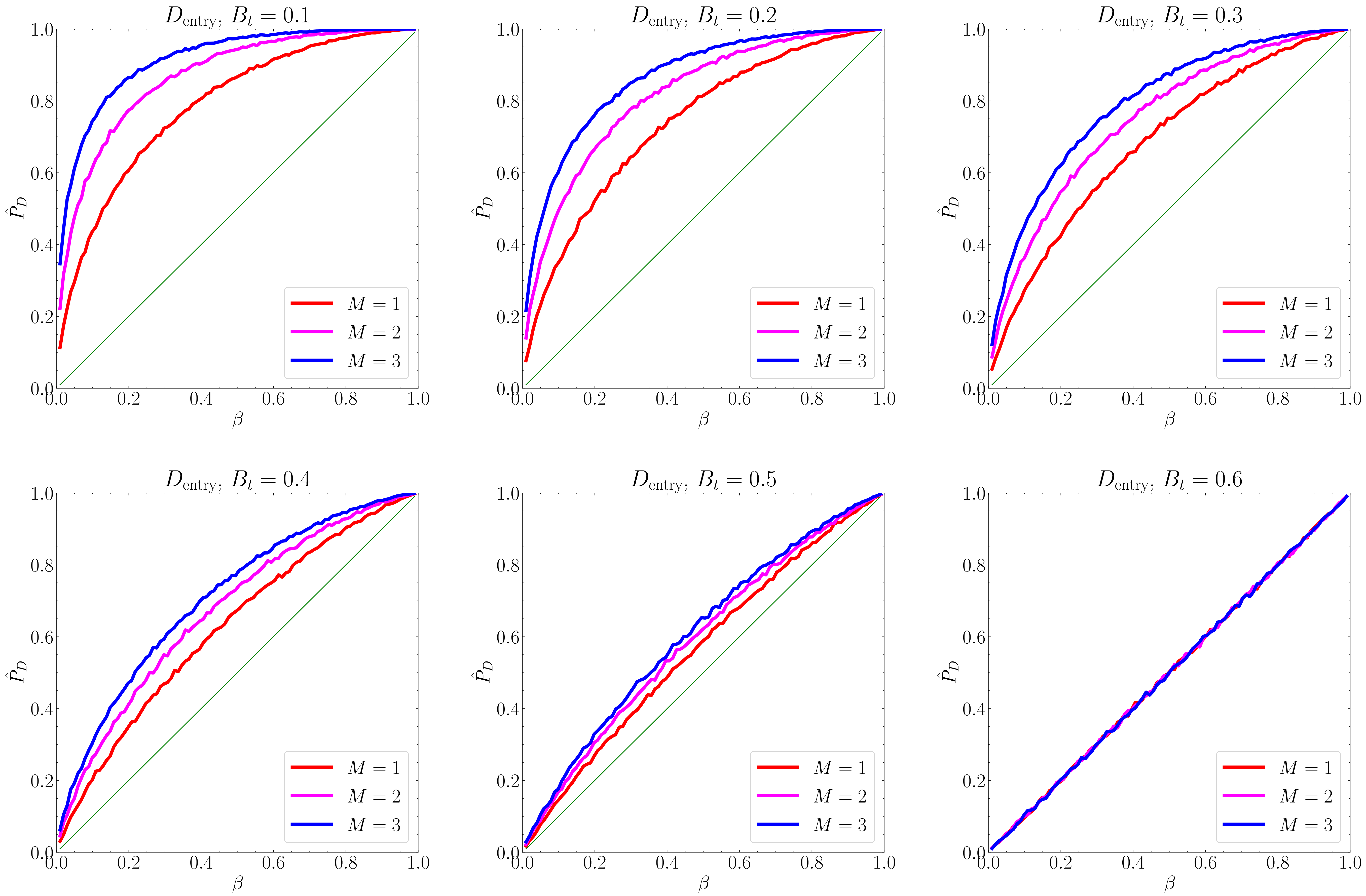
$$L_{ent}(\mathbf{x}) = \frac{p_X(\mathbf{x};\mathcal{H}_0)}{p_X(\mathbf{x};\mathcal{H}_1)} < \gamma.$$

- Under the given hypothesis,

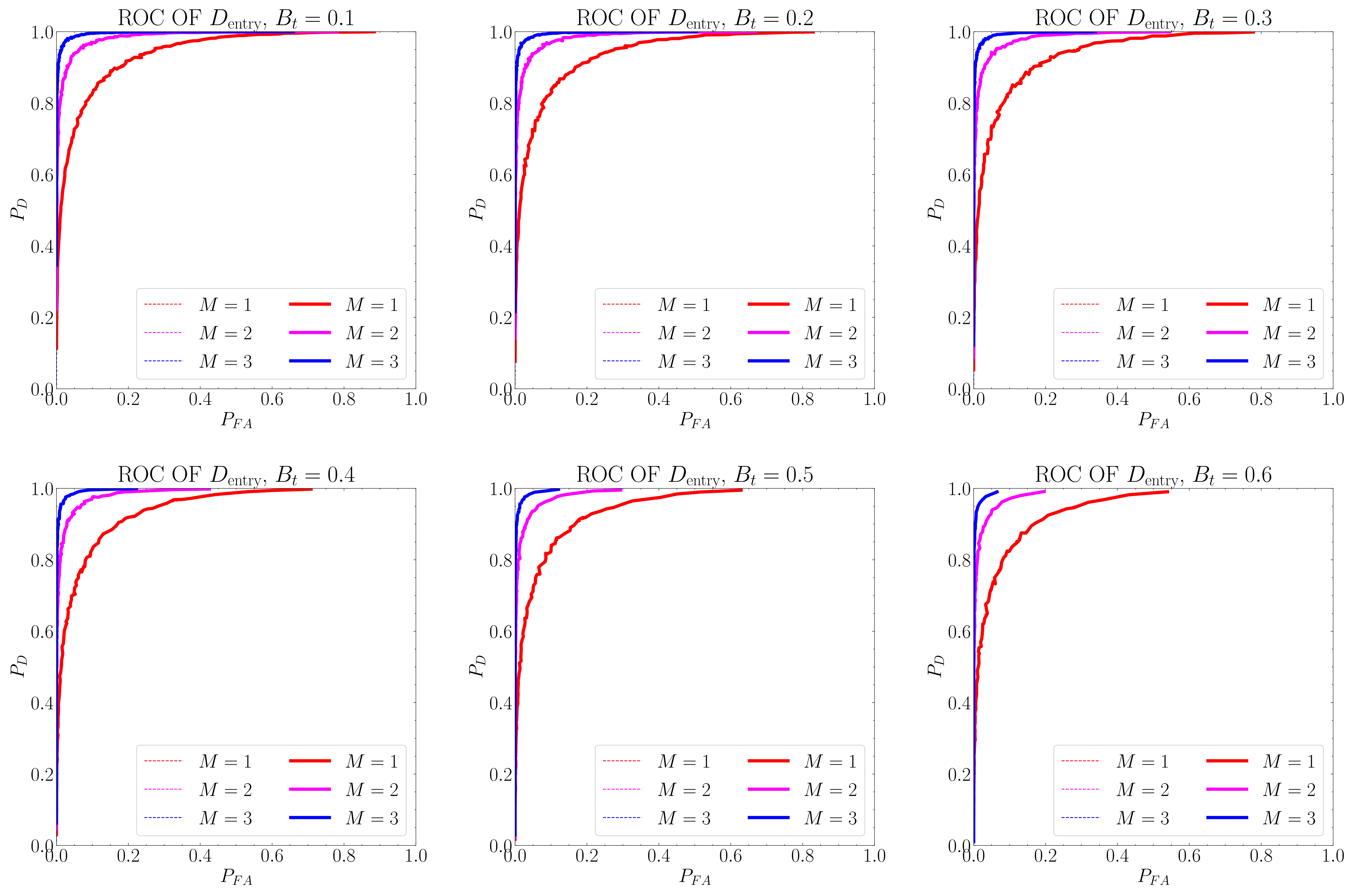
$$T(\mathbf{x}) = \frac{1}{N}\sum_{n=0}^{N-1} x_m[n] < \gamma'.$$

- The threshold of the detector $\gamma' = \sqrt{\frac{\sigma^2}{N}}Q^{-1}(1-\beta) + B_t$, and the probability of false alarm $P_{FA} = 1 - Q\left(Q^{-1}(1-\beta) - \sqrt{\frac{NA^2}{\sigma^2}}\right)$.

Detector at the Entry



Detector at the Entry



Detector at the Exit

- D_{exit} requires constant P_{FA} and the largest P_D,

$$\underset{R_1}{\text{maximise}}\ P_D,$$
$$\text{subject to } P_{FA} = \alpha.$$

- The solution is a ratio test, choose $\mathcal{H}_1$ if:

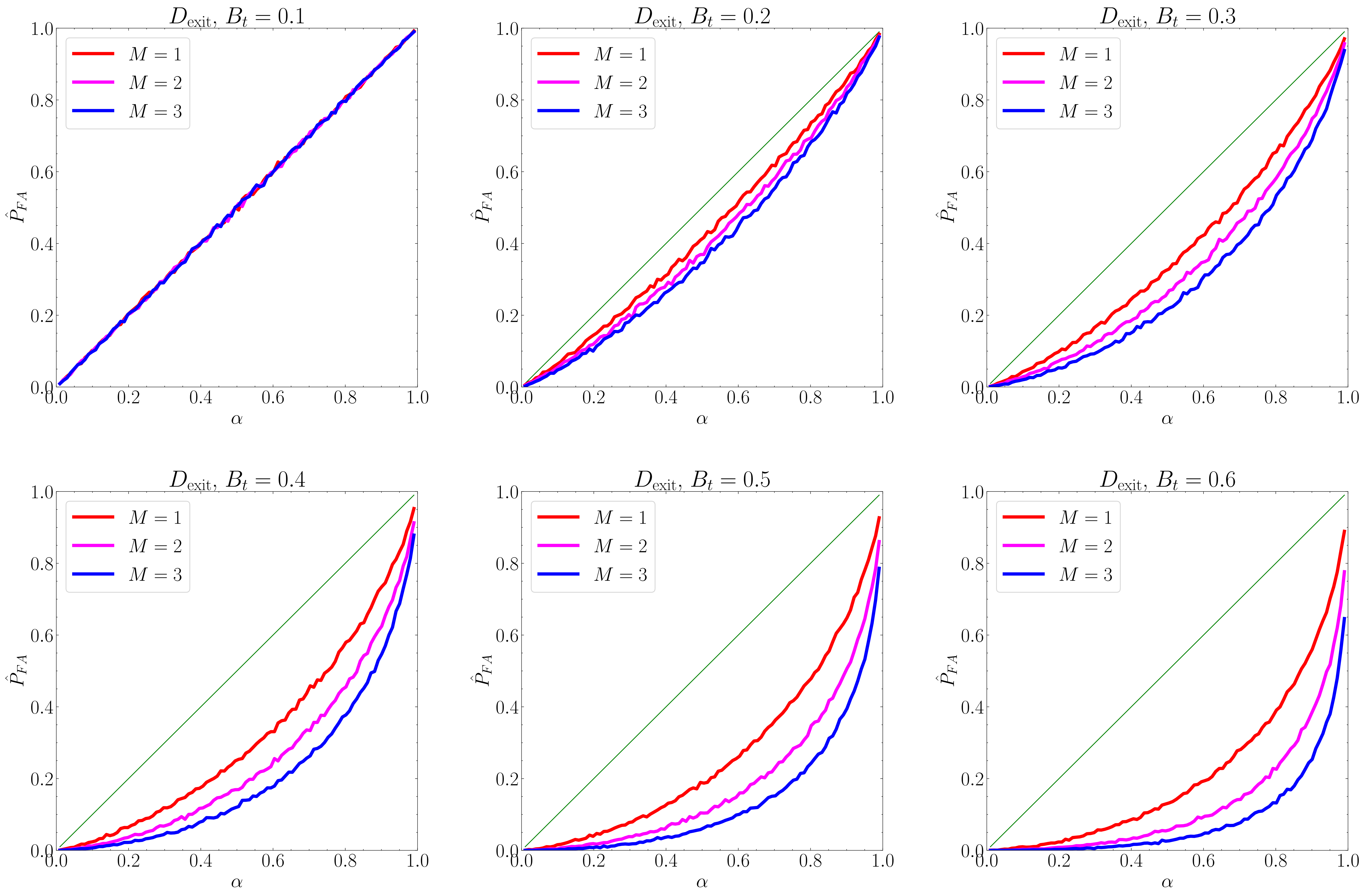
$$L_{ext}(\mathbf{x}) = \frac{p_X(\mathbf{x}; \mathcal{H}_1)}{p_X(\mathbf{x}; \mathcal{H}_0)} > \xi.$$

- Under the given hypothesis,

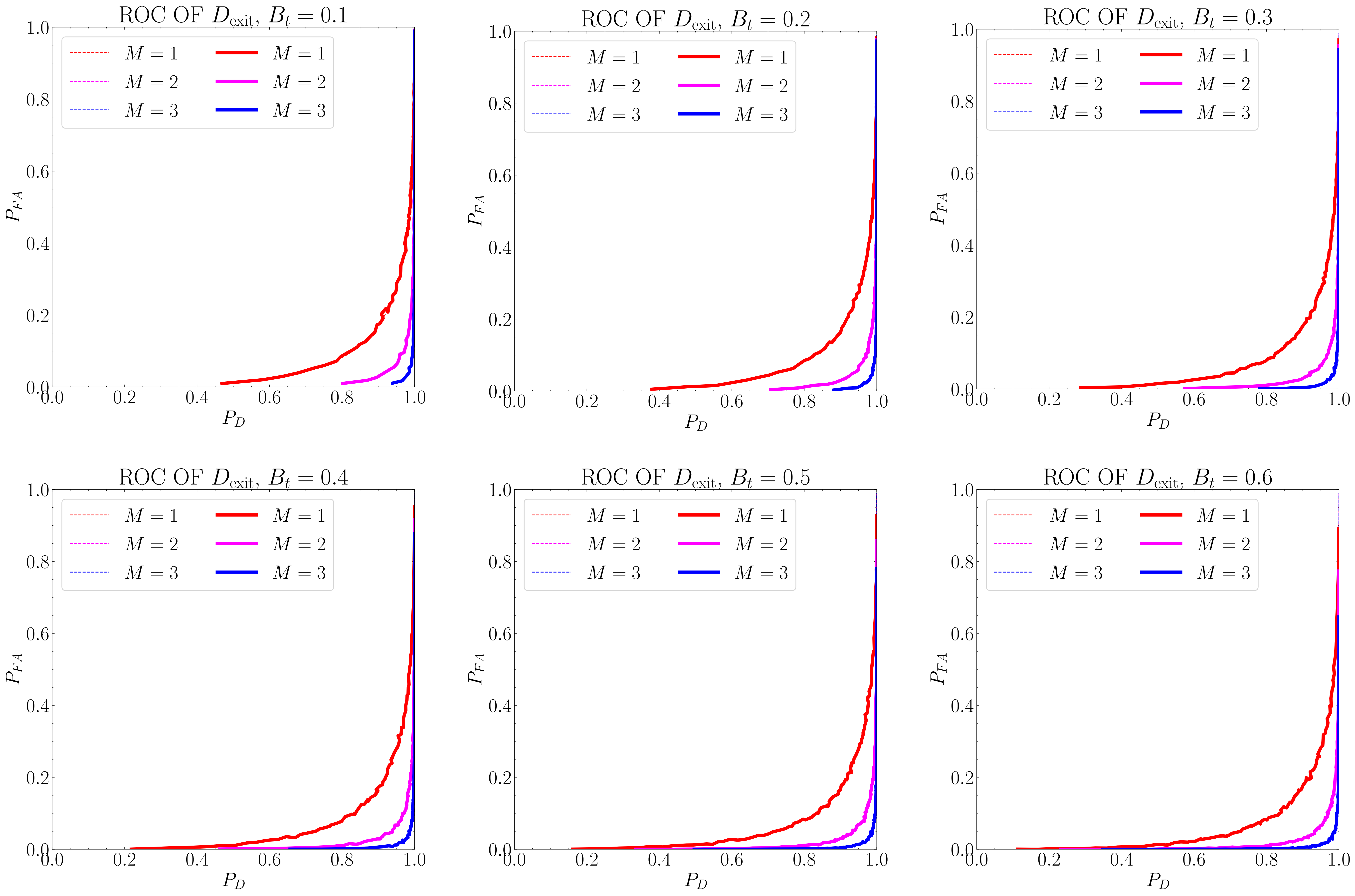
$$T(\mathbf{x}) = \frac{1}{N} \sum_{n=0}^{N-1} x_m[n] < \xi'.$$

- The threshold of the detector $\xi' = \sqrt{\frac{\sigma^2}{N}} Q^{-1}(1-\alpha) + A + B_t$, and the probability of detection $P_D = 1 - Q\left(Q^{-1}(1-\alpha) - \sqrt{\frac{NA^2}{\sigma^2}}\right)$.

Detector at the Exit



Detector at the Exit



Tracking Specification

- The RADAR provides noisy position and velocity in 2D. The state variable $\mathbf{x}[n] = [p_x[n] \ p_y[n] \ \dot{p}_x[n] \ \dot{p}_y[n]]^\mathsf{T}$,
- The state model,

$$\mathbf{x}[n] = \begin{bmatrix} 1 & 0 & \Delta & 0 \\ 0 & 1 & 0 & \Delta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}[n-1] + \mathbf{w}[n], a.e.$$

- Velocity-only measurements:

$$\mathbf{y}[n] = \underbrace{\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{C}[n]} \mathbf{x}[n-1] + \mathbf{v}[n], \ n = 0, 1, \cdots, N-1,$$

- Position and velocity measurements:

$$\mathbf{y}[n] = \mathbf{x}[n-1] + \mathbf{v}[n], \ n = 0, 1, \cdots, N-1.$$

Kalman Filter: Updates and Predictions

- With some initialisation $\hat{x}[0|0]$ and $P[0|0]$,
- The Kalman filter updates,

$$\hat{\mathbf{x}}[n|n-1] = \mathbf{A}[n-1]\hat{\mathbf{x}}[n-1|n-1],$$
$$\mathbf{P}[n|n-1] = \mathbf{A}[n-1]\mathbf{P}[n-1|n-1]\mathbf{A}^{\mathsf{T}}[n-1] + \mathbf{Q}_w[n],$$

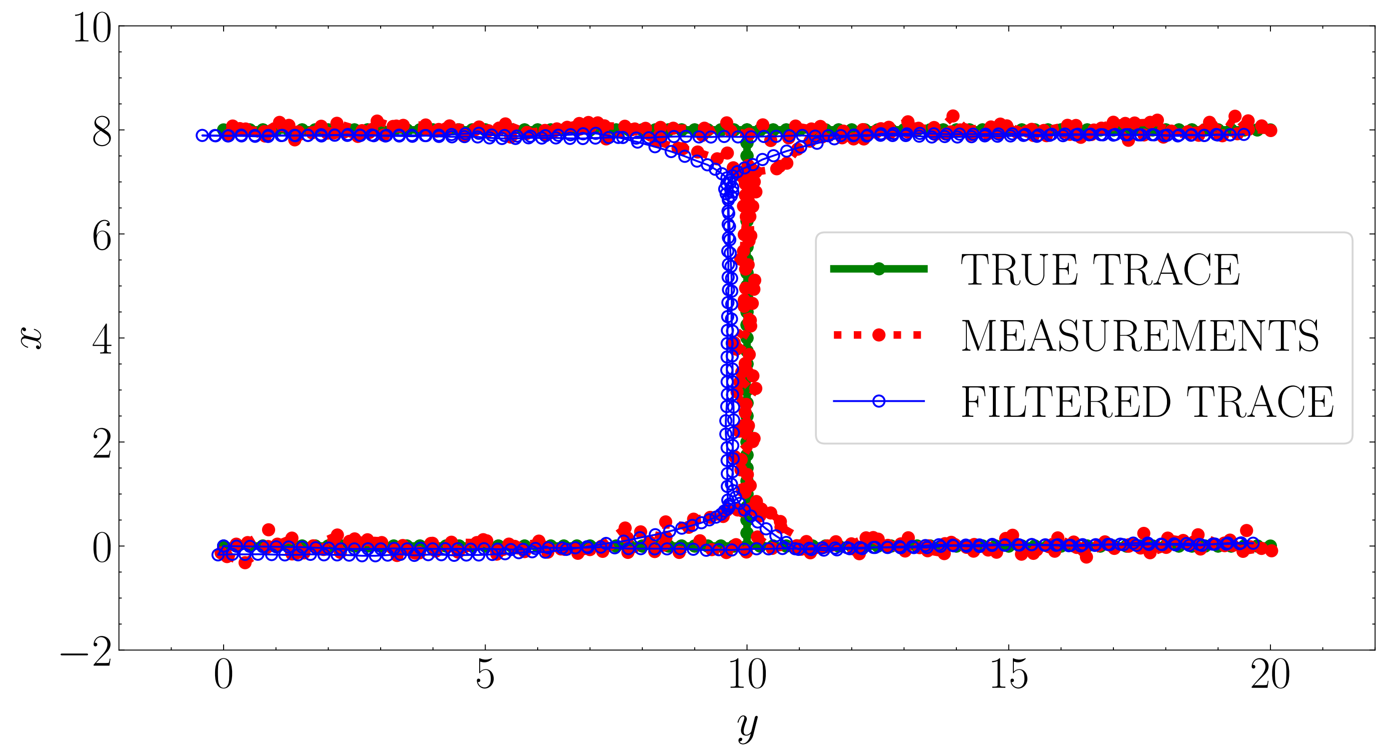
- The Kalman filter predictions,

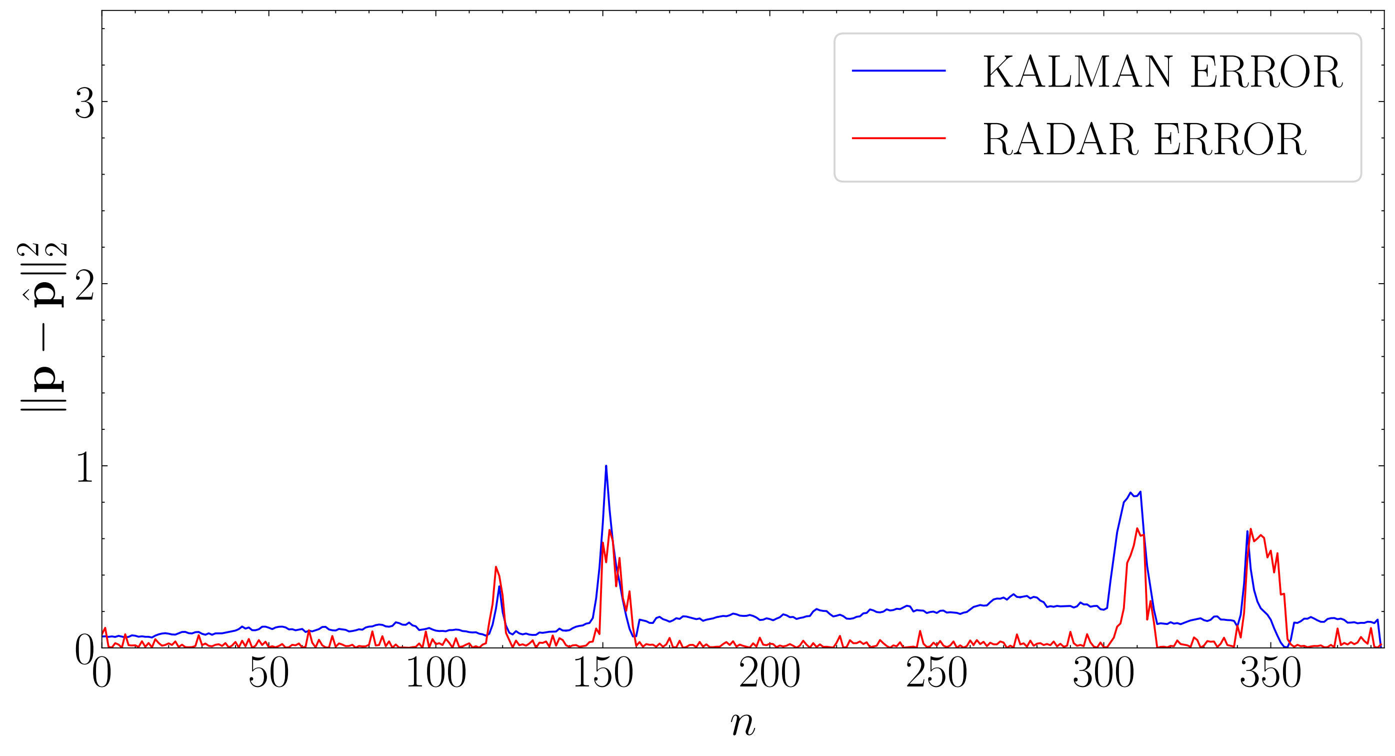
$$\hat{\mathbf{x}}[n|n] = \hat{\mathbf{x}}[n|n-1] + \mathbf{K}[n]\left(y[n] - \mathbf{C}[n]\hat{\mathbf{x}}[n|n-1]\right),$$
$$\mathbf{P}[n|n] = \left(\mathbf{I} - \mathbf{K}[n]\mathbf{C}[n]\right)\mathbf{P}[n|n-1],$$

where the Kalman gain is given by $\mathbf{K}[n] = \mathbf{P}[n|n-1]\mathbf{C}^{\mathsf{T}}[n]\left(\mathbf{Q}_v + \mathbf{C}[n]\mathbf{P}[n|n-1]\mathbf{C}^{\mathsf{T}}[n]\right)^{-1}$.

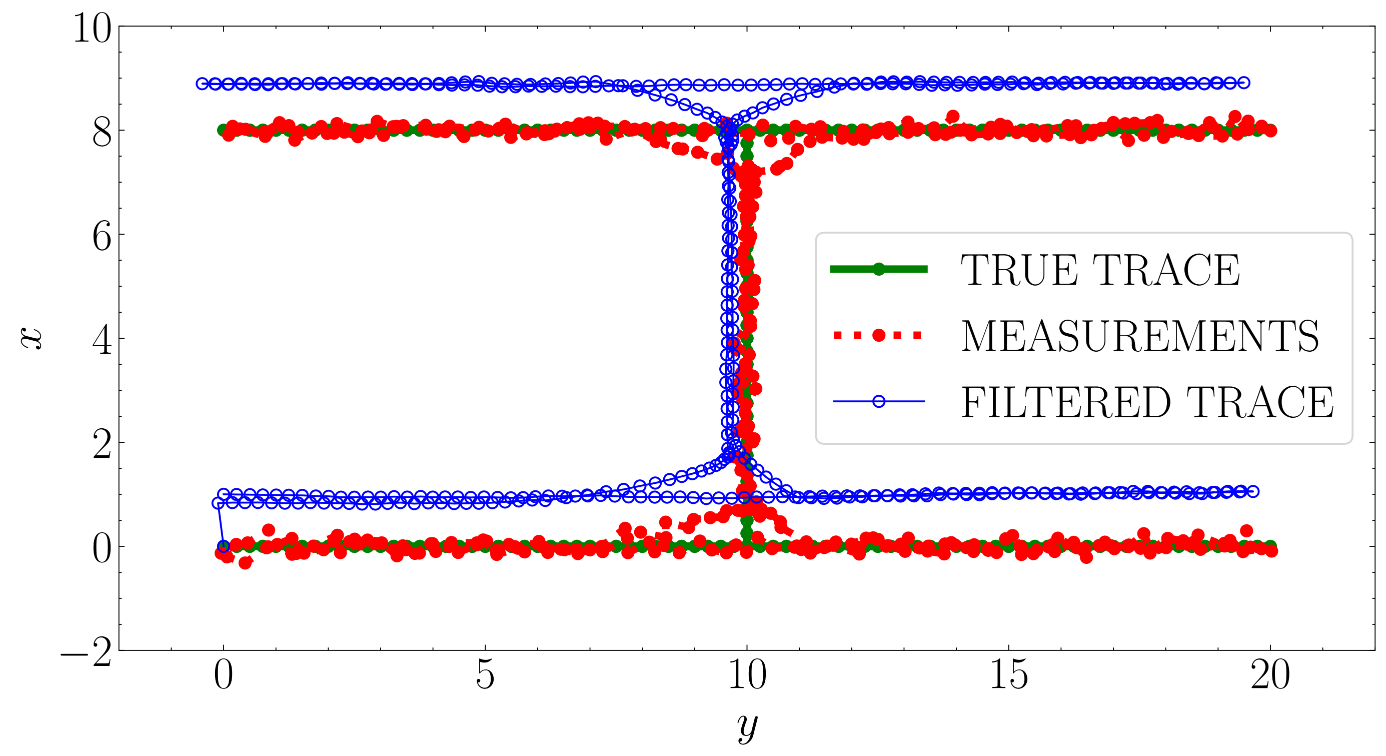
Kalman Filter with Velocity Only Measurements

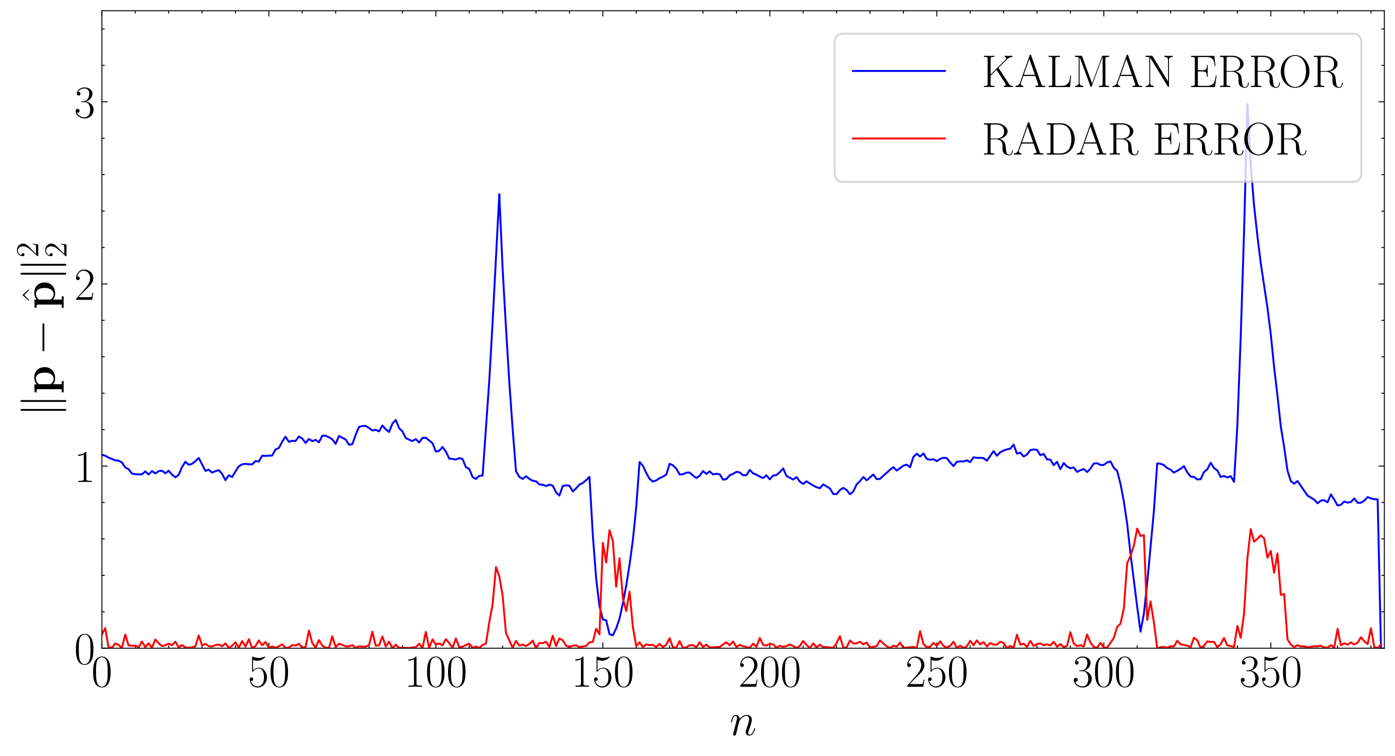
$$\hat{\mathbf{x}}(0|0) = \mathbf{0}$$



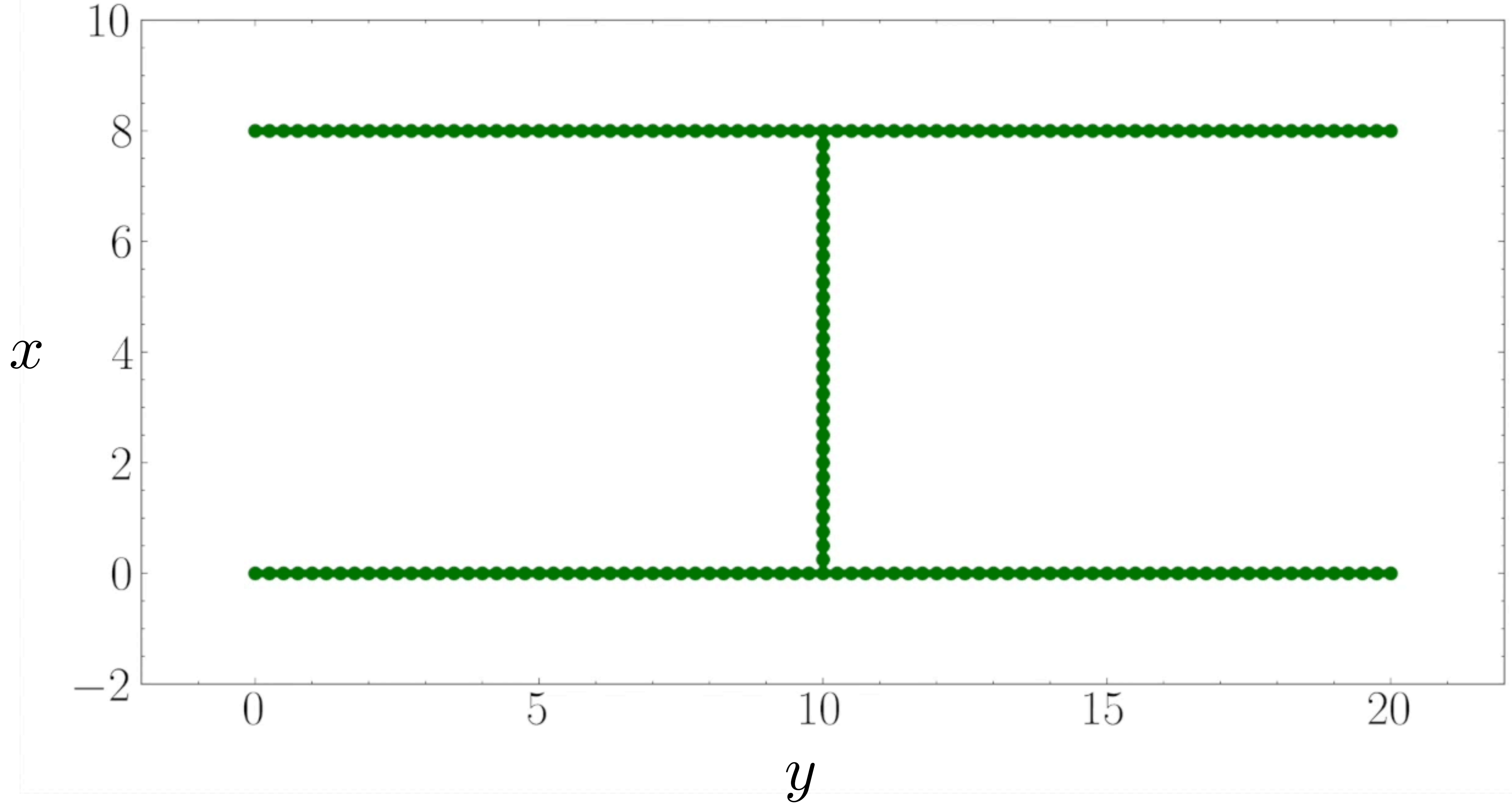


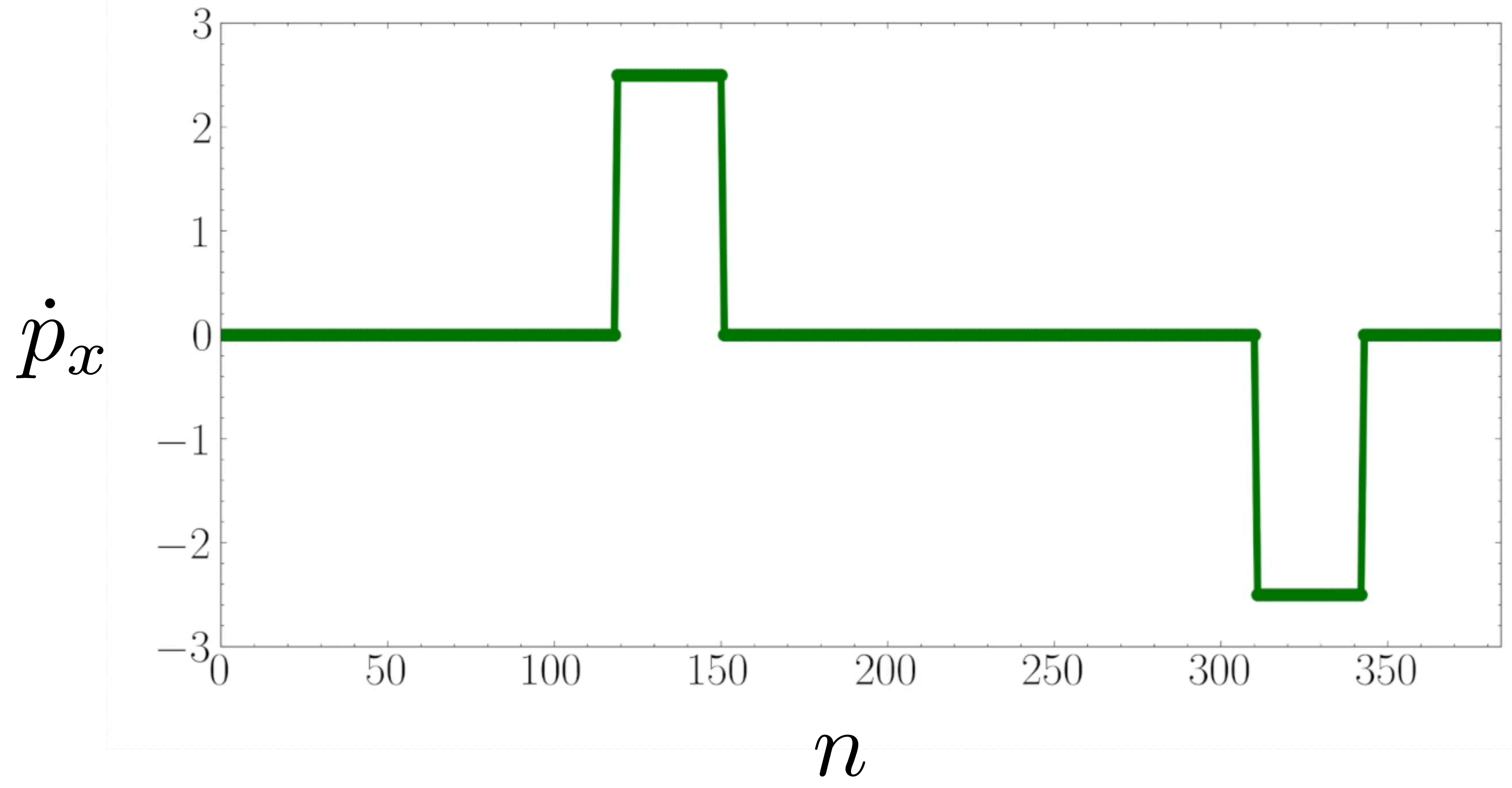
$$\hat{\mathbf{x}}(0|0) \neq \mathbf{0}$$

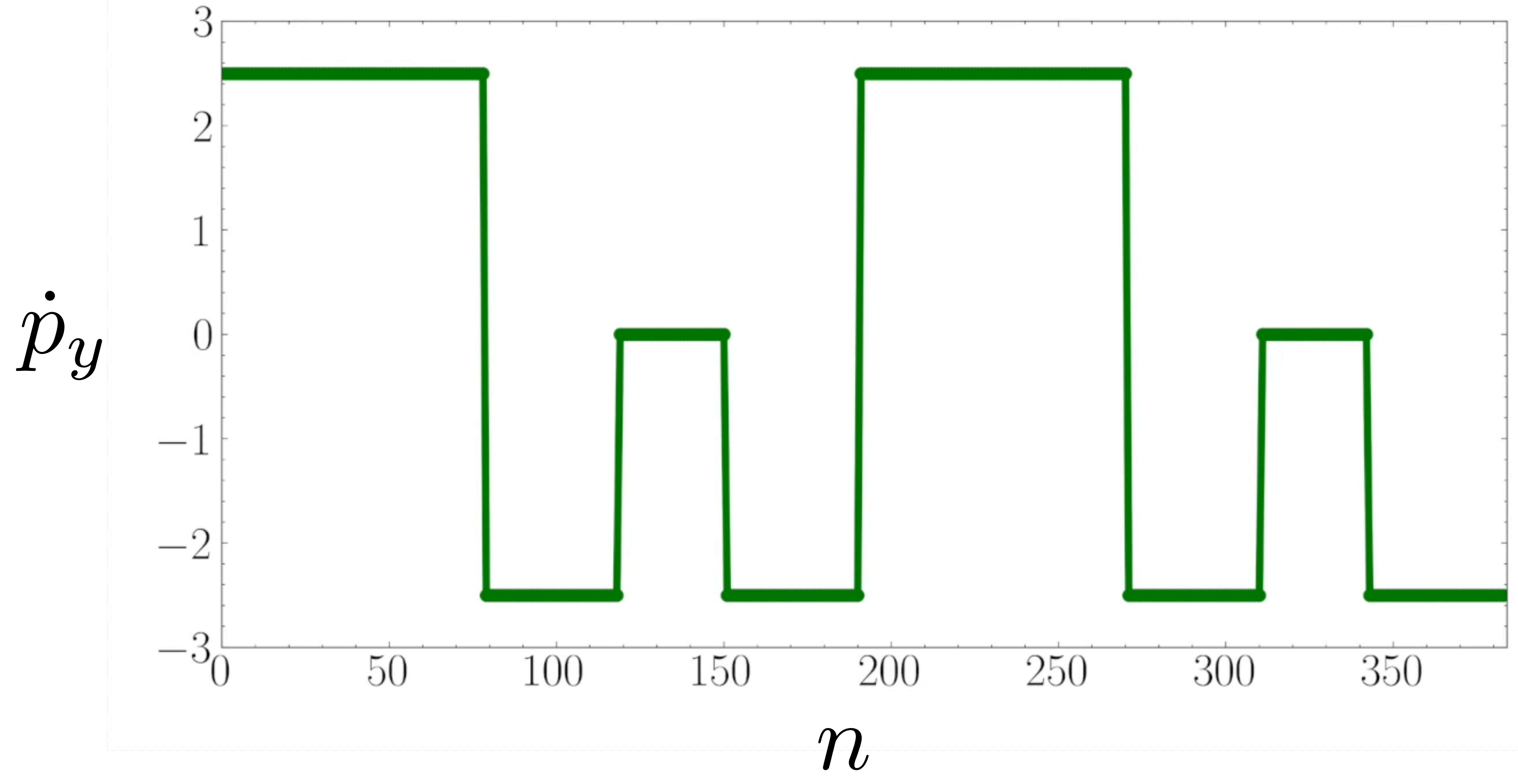




Kalman Filter with Position and Velocity Measurements

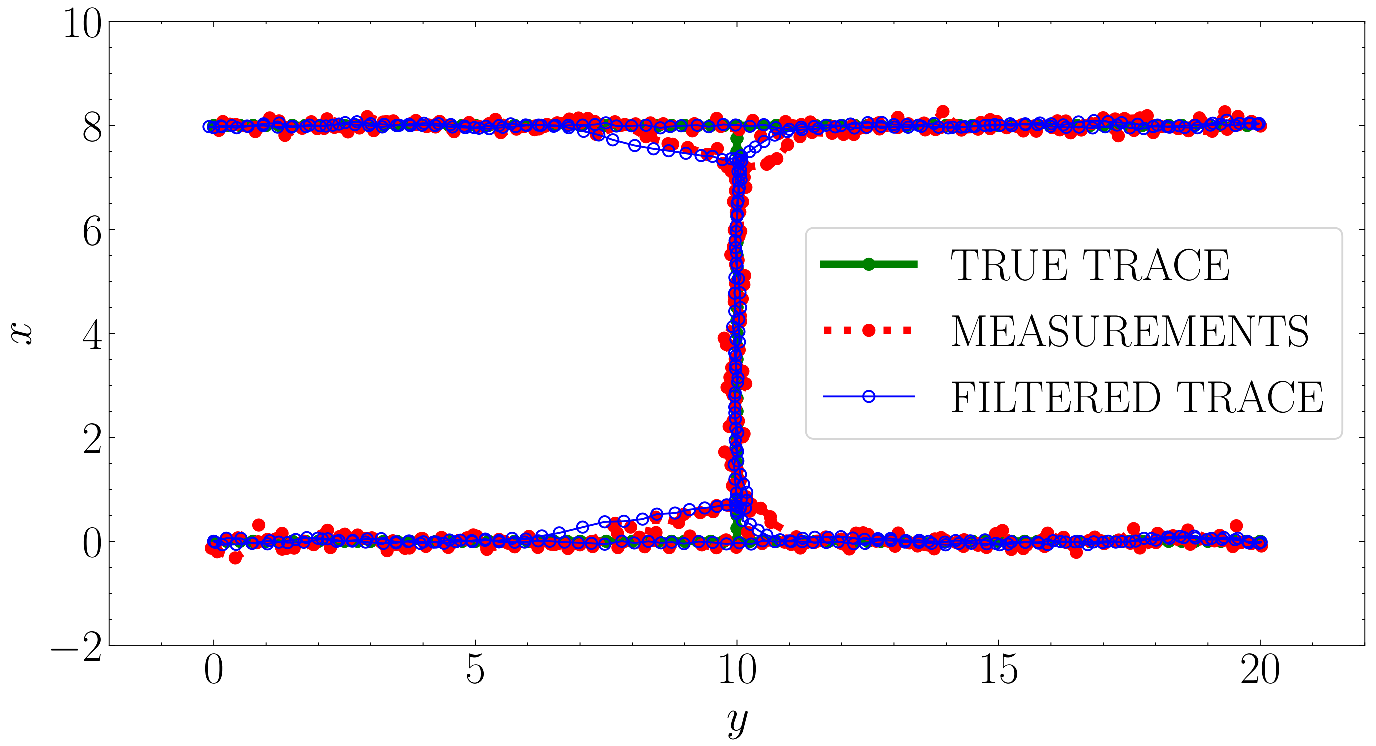


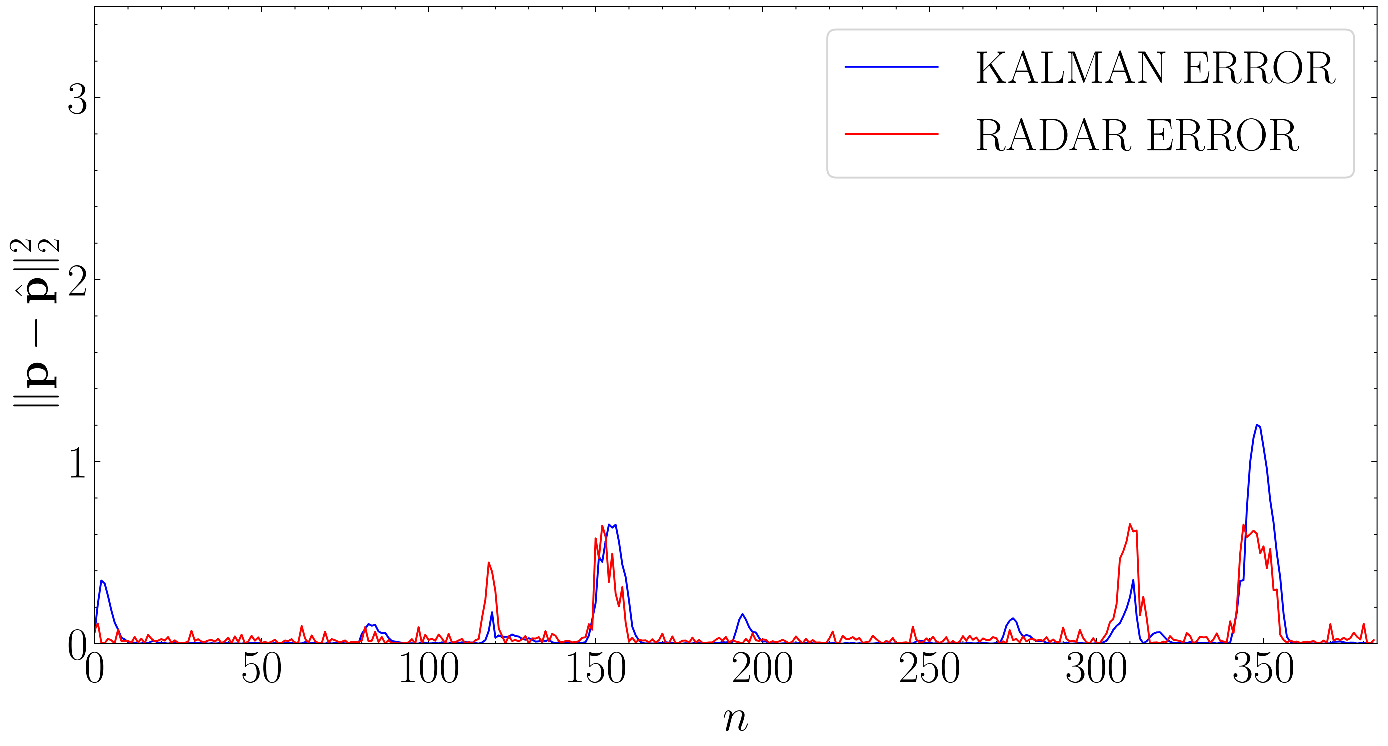




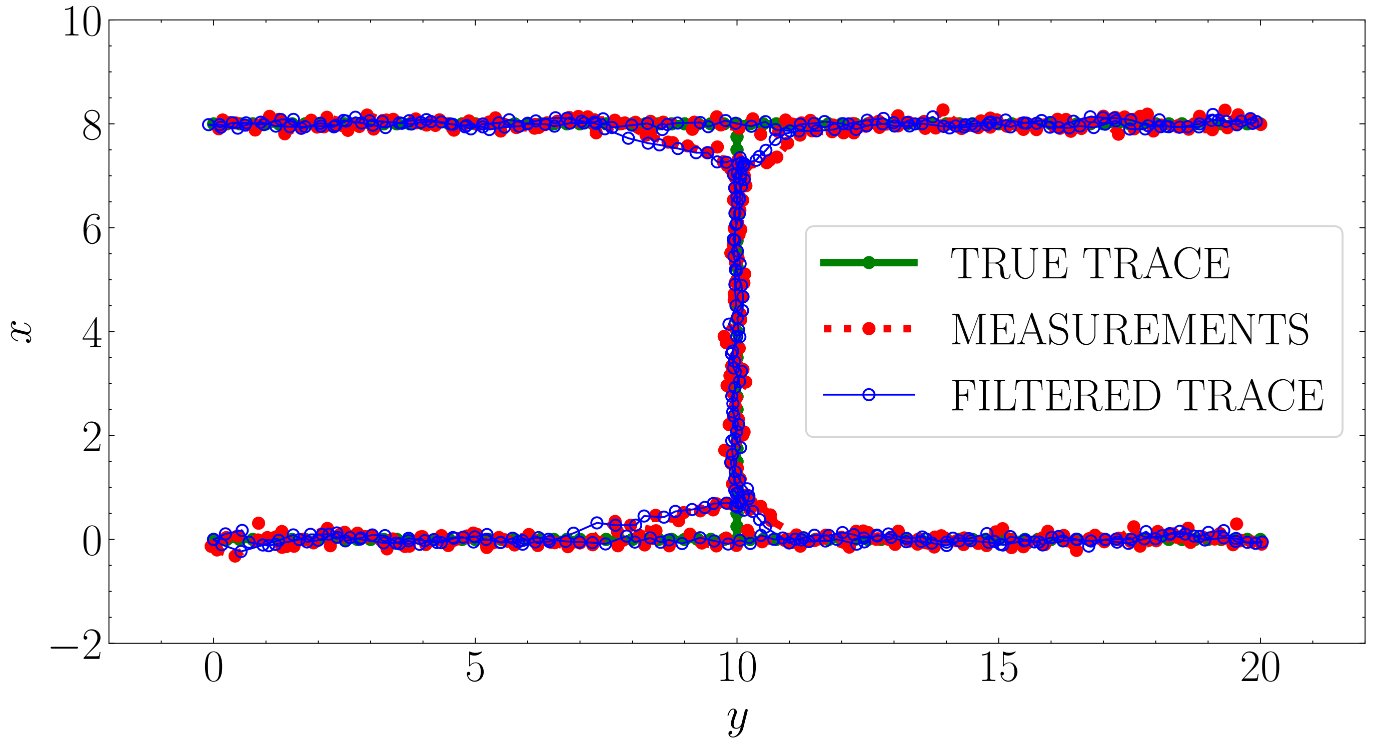
Kalman Filter with Position and Velocity Measurements

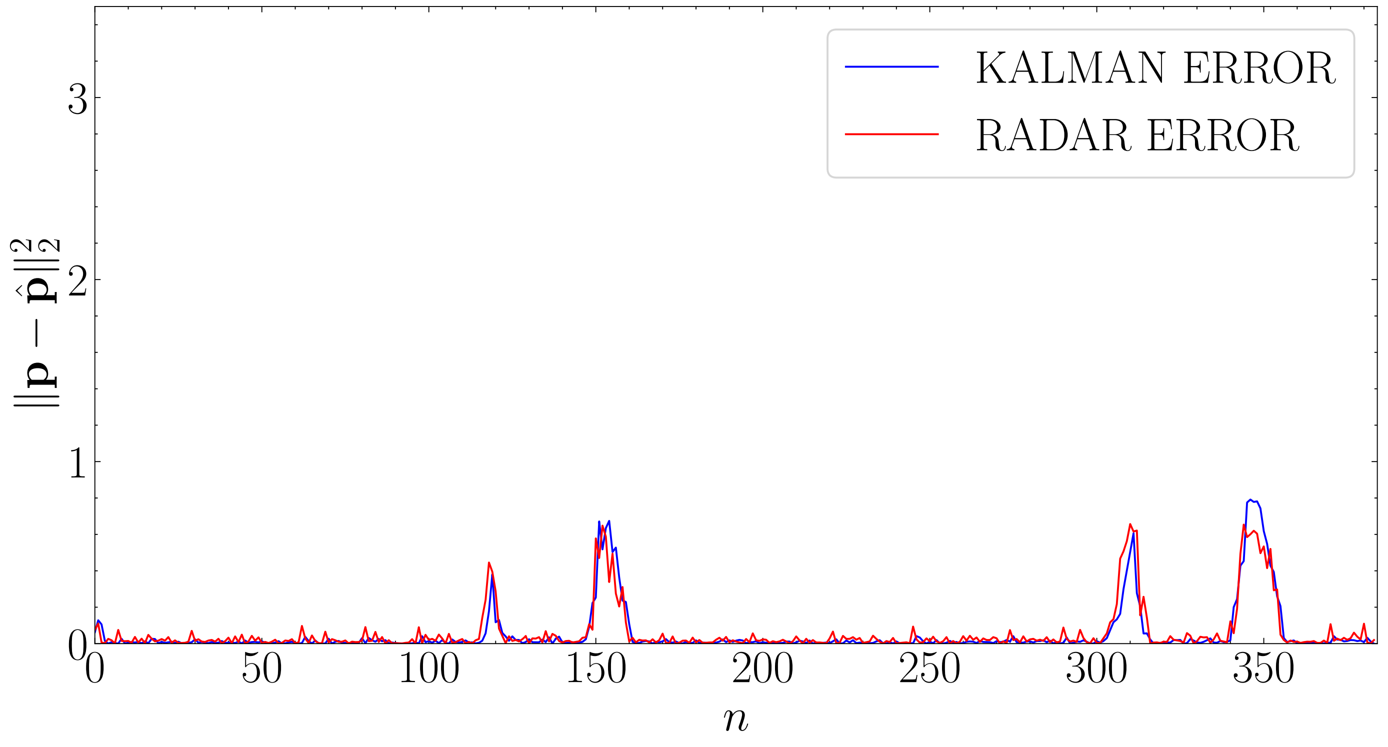
$\mathbf{Q}_w = 0.01\mathbf{I}$



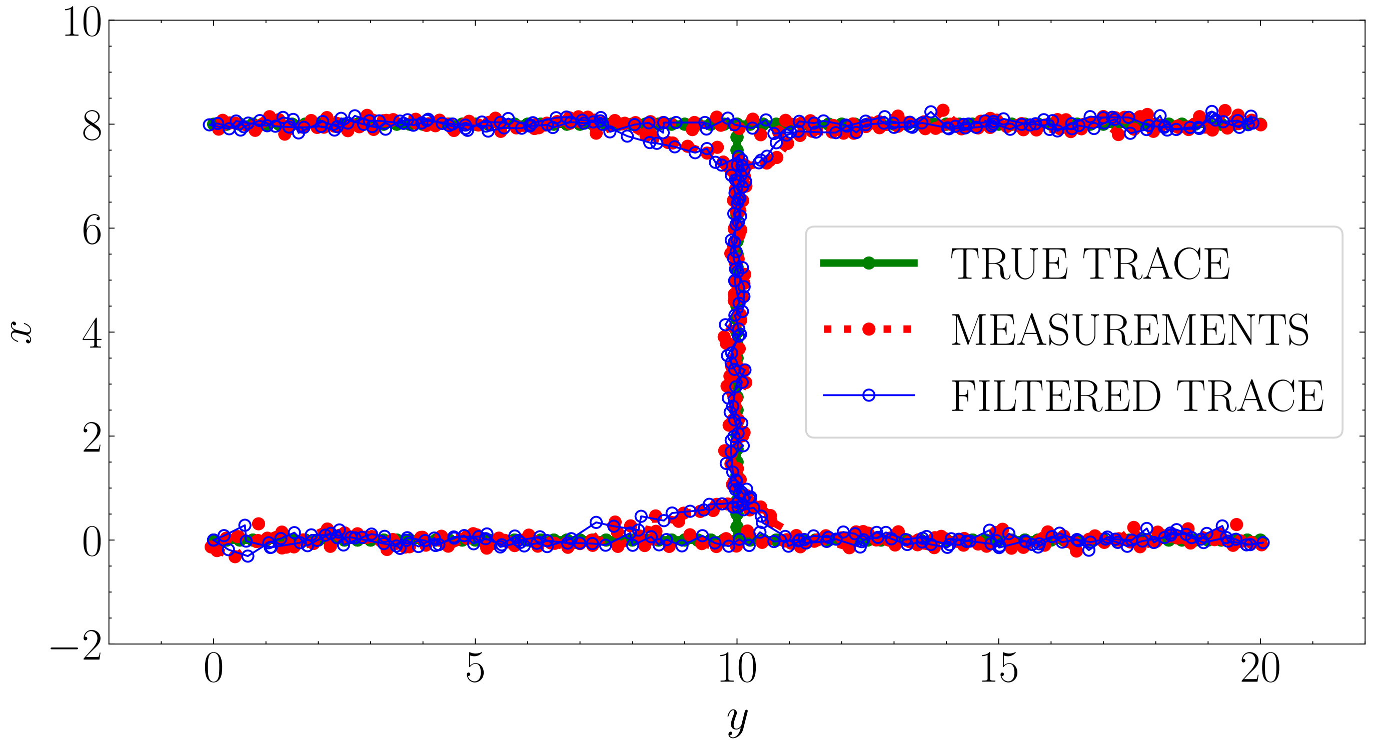


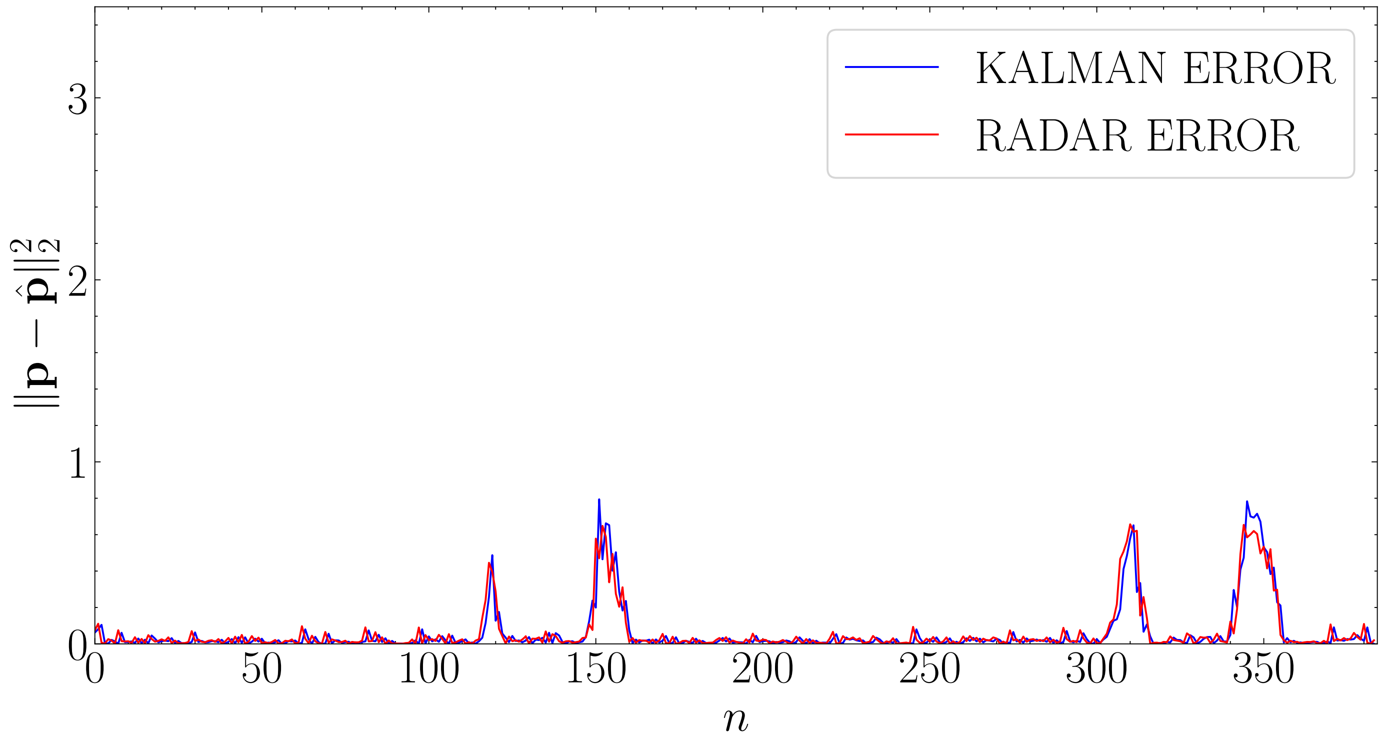
$\mathbf{Q}_w = 0.1\mathbf{I}$





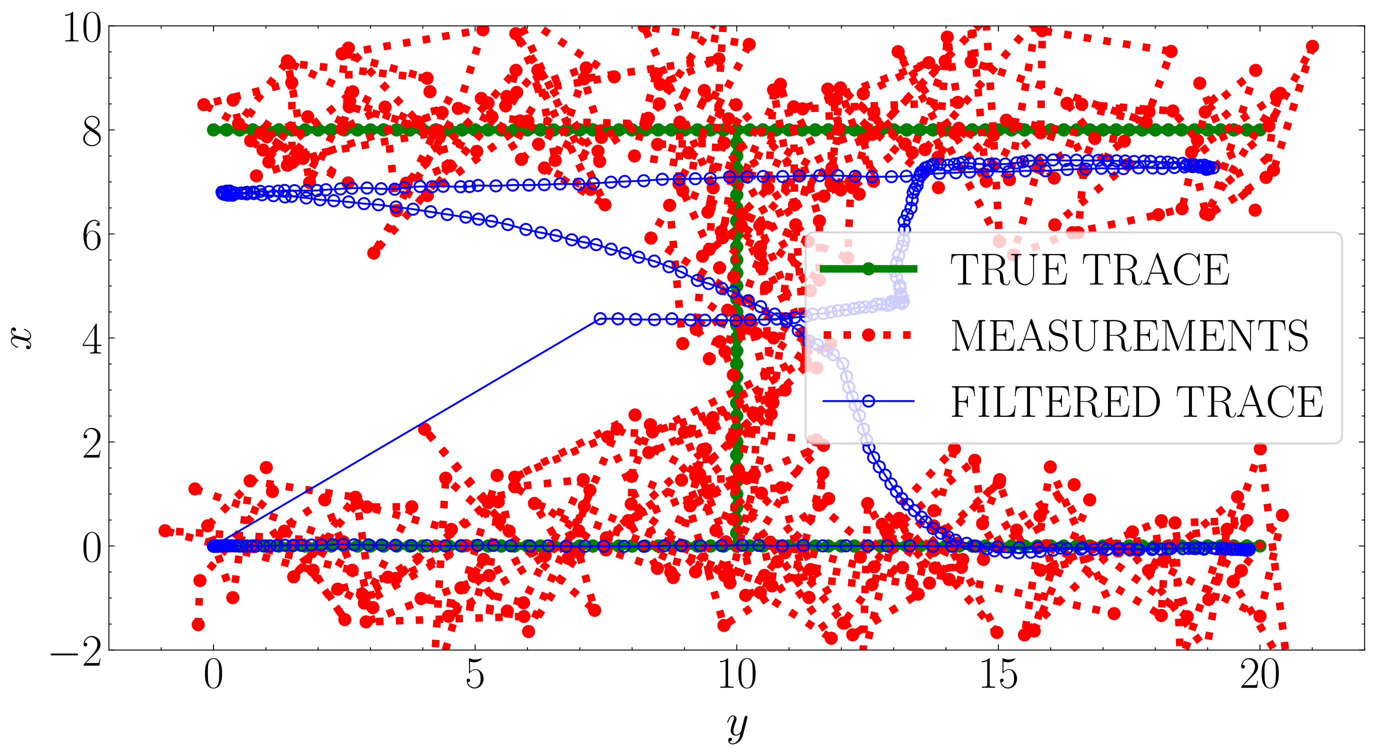
$\mathbf{Q}_w = \mathbf{I}$

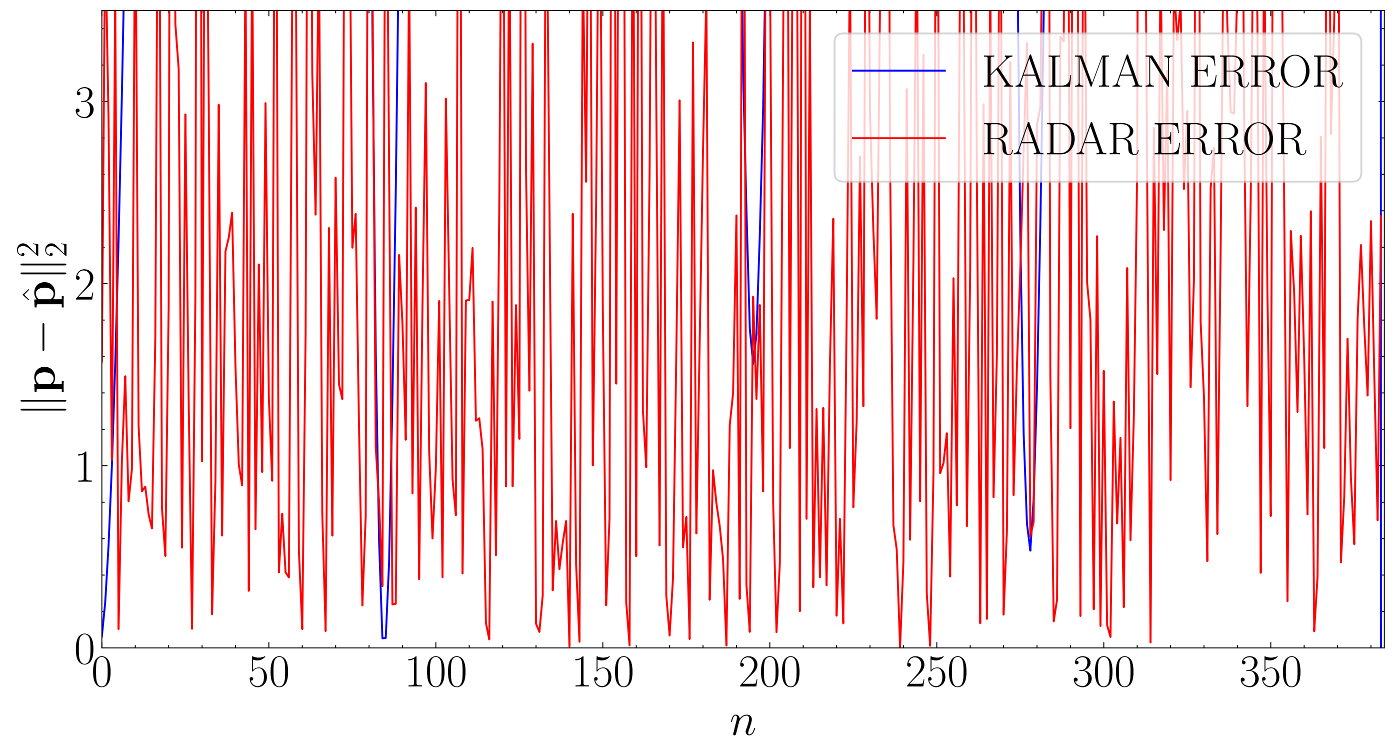




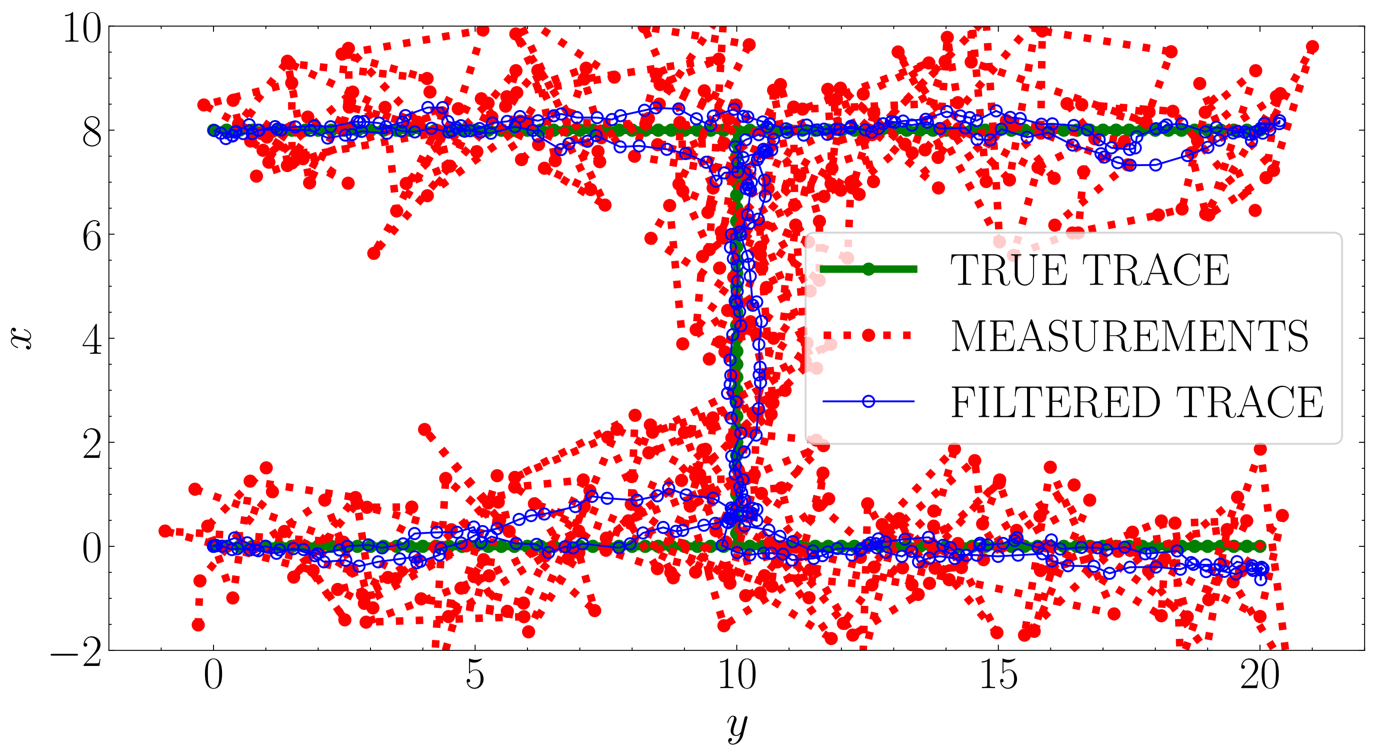
Kalman Filter with Noisy Measurements

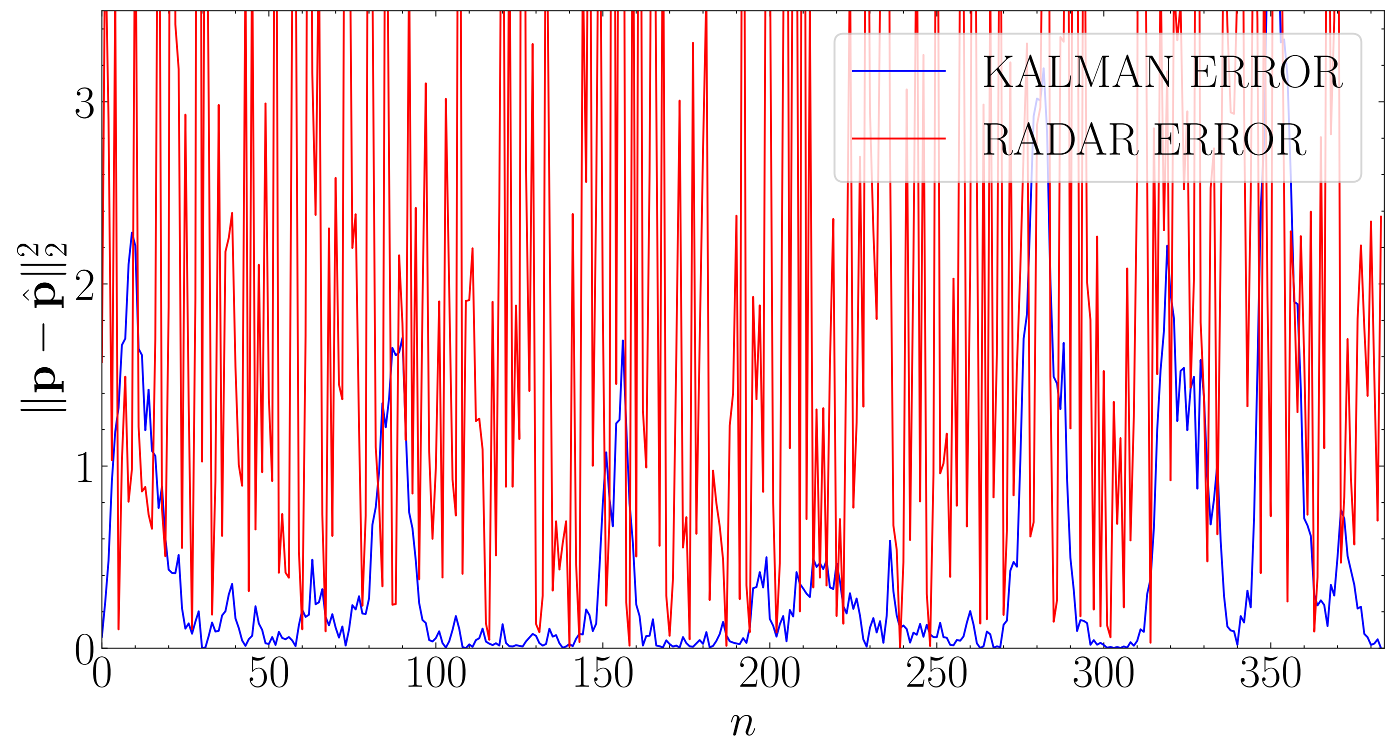
$\mathbf{Q}_w = 0.0001\mathbf{I}$





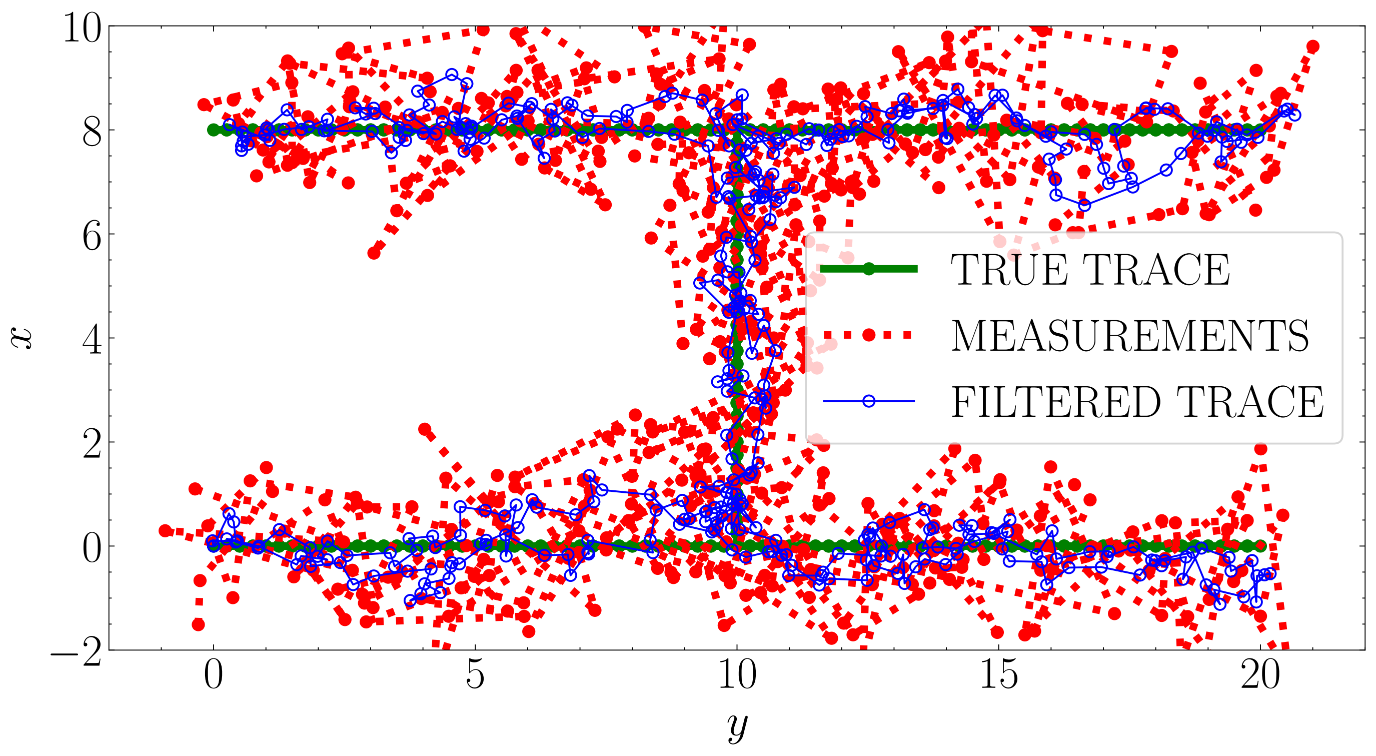
$\mathbf{Q}_w = 0.01\mathbf{I}$

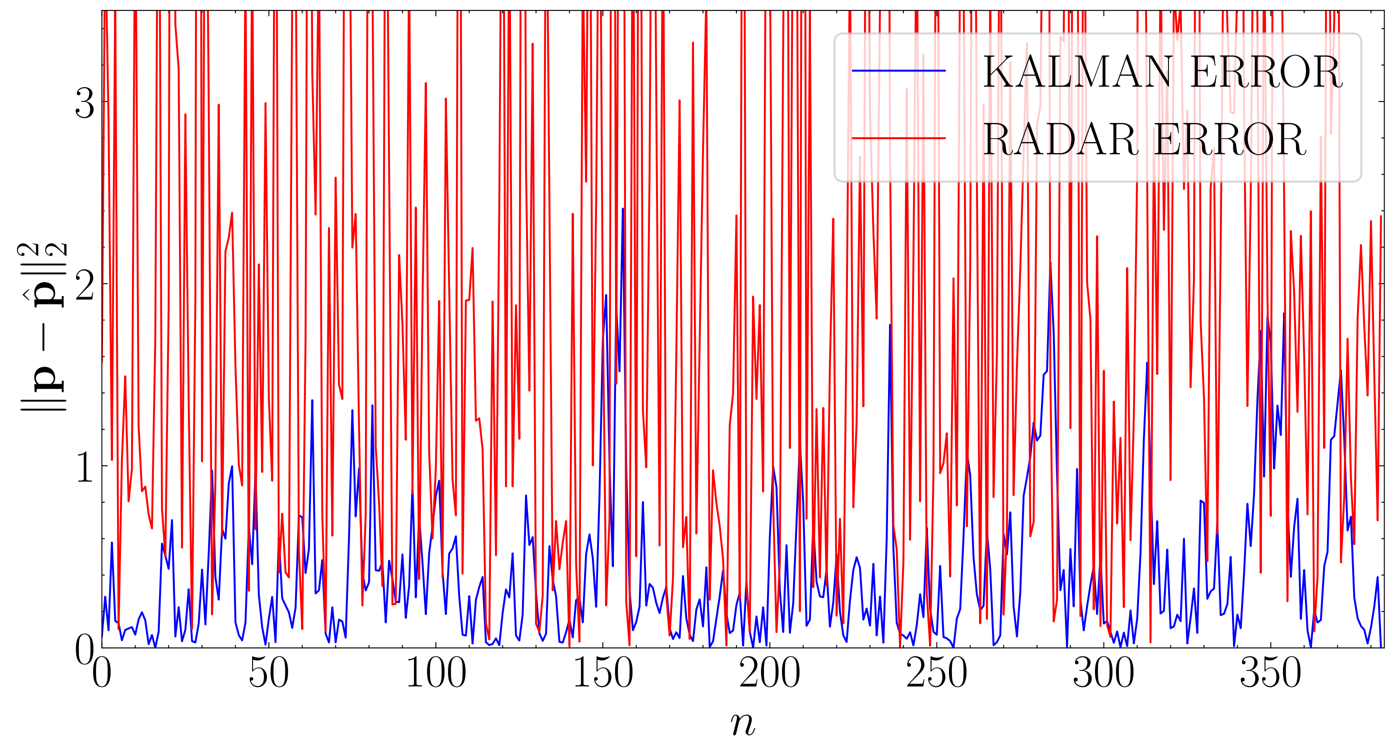




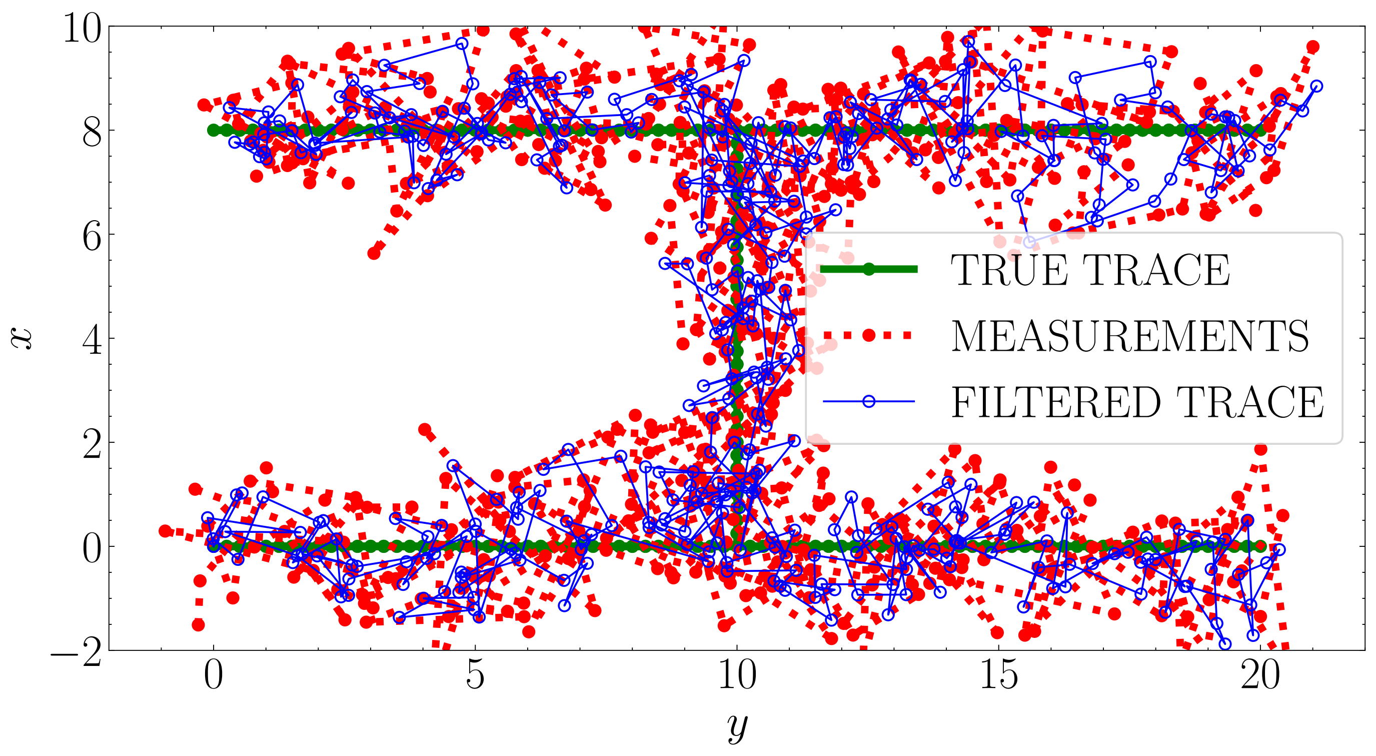
Kalman Filter with Noisy Measurements

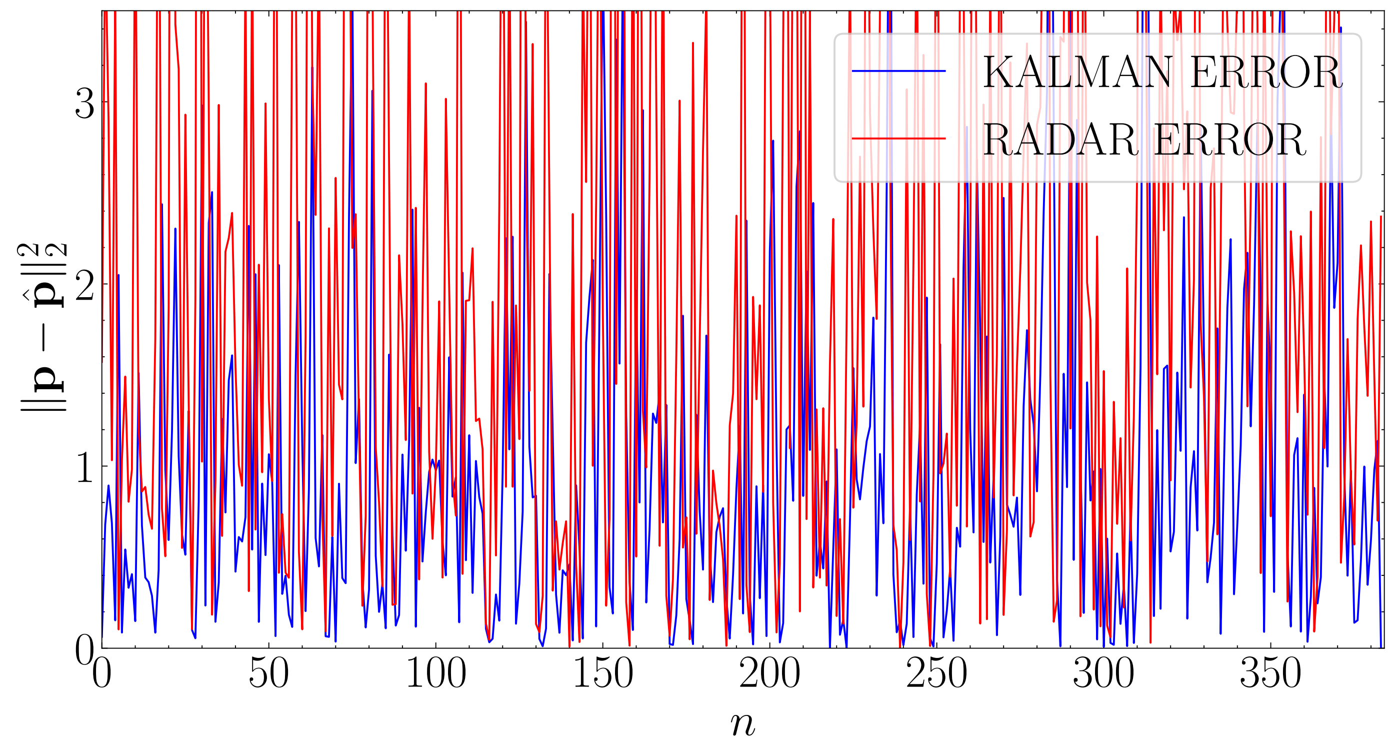
$\mathbf{Q}_w = 0.1\mathbf{I}$





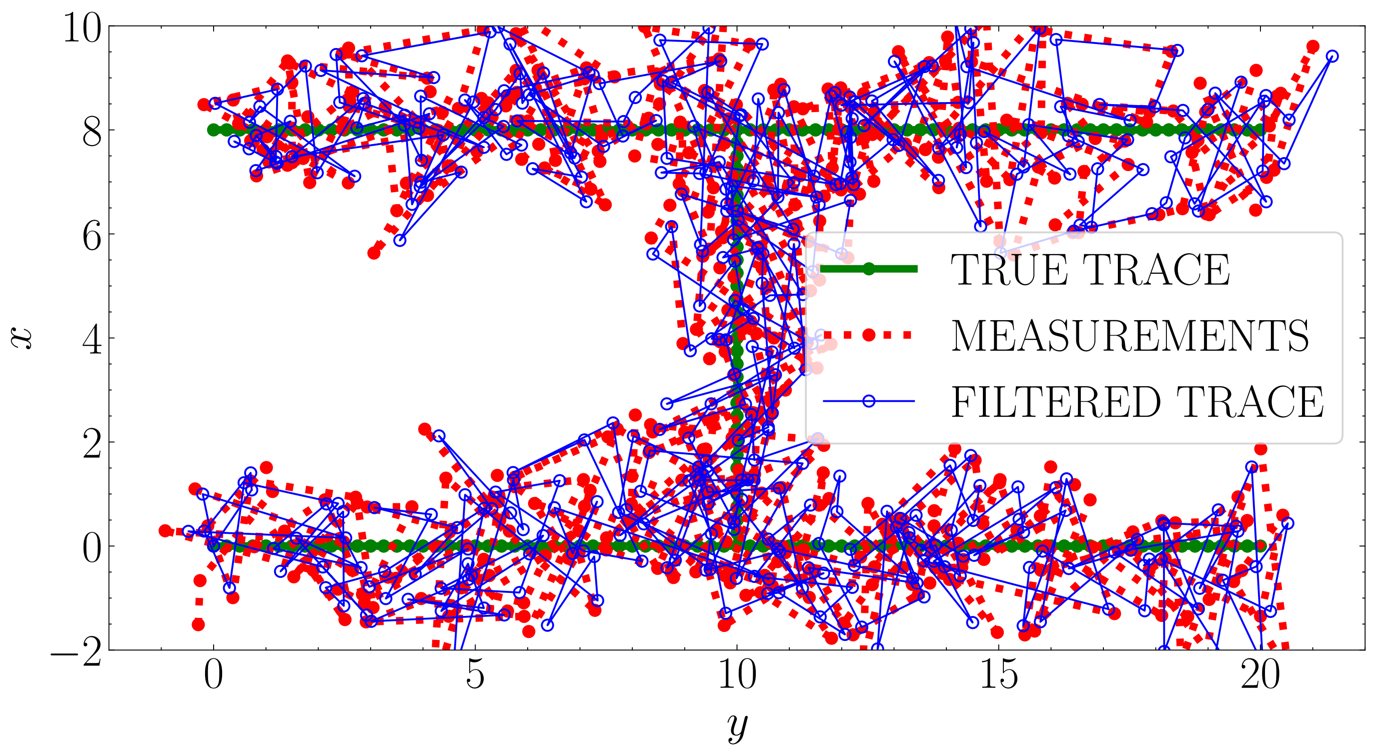
$\mathbf{Q}_w = \mathbf{I}$

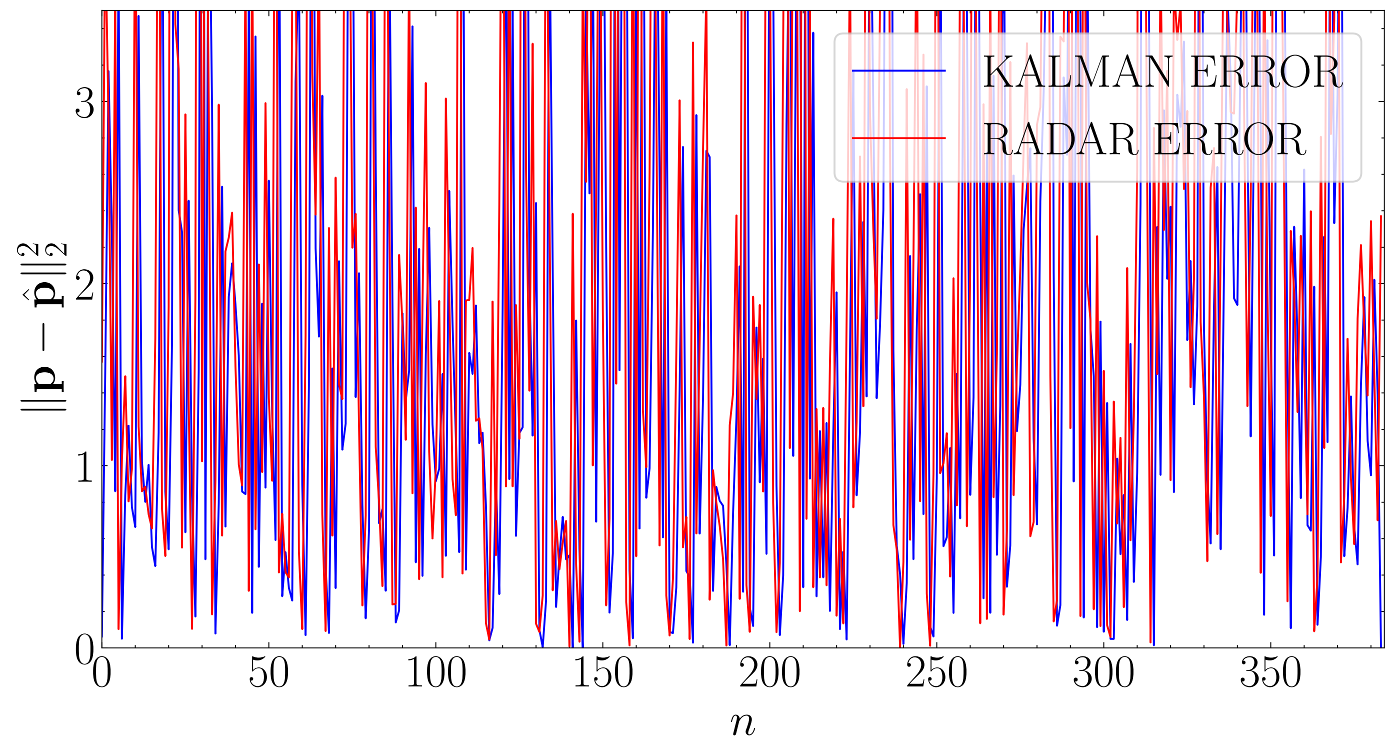




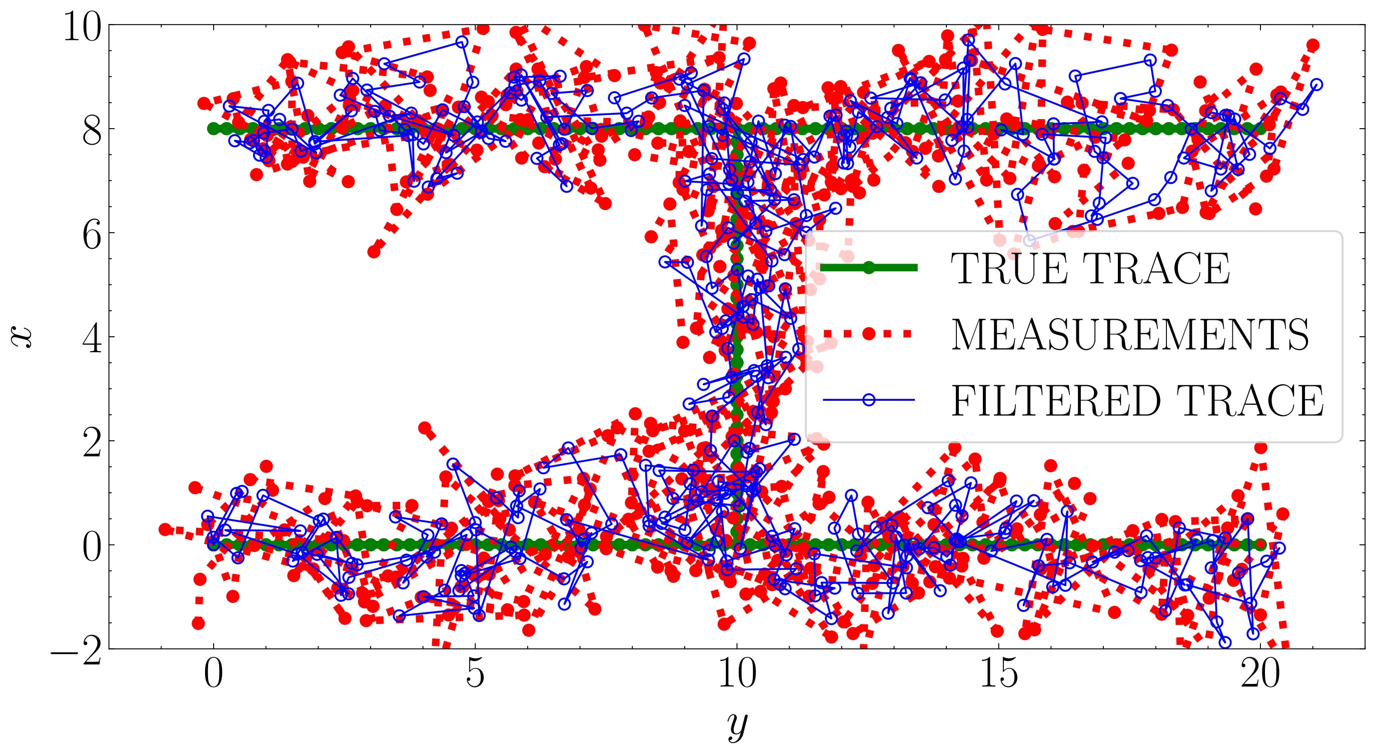
Kalman Filter with Mismatched Parameter

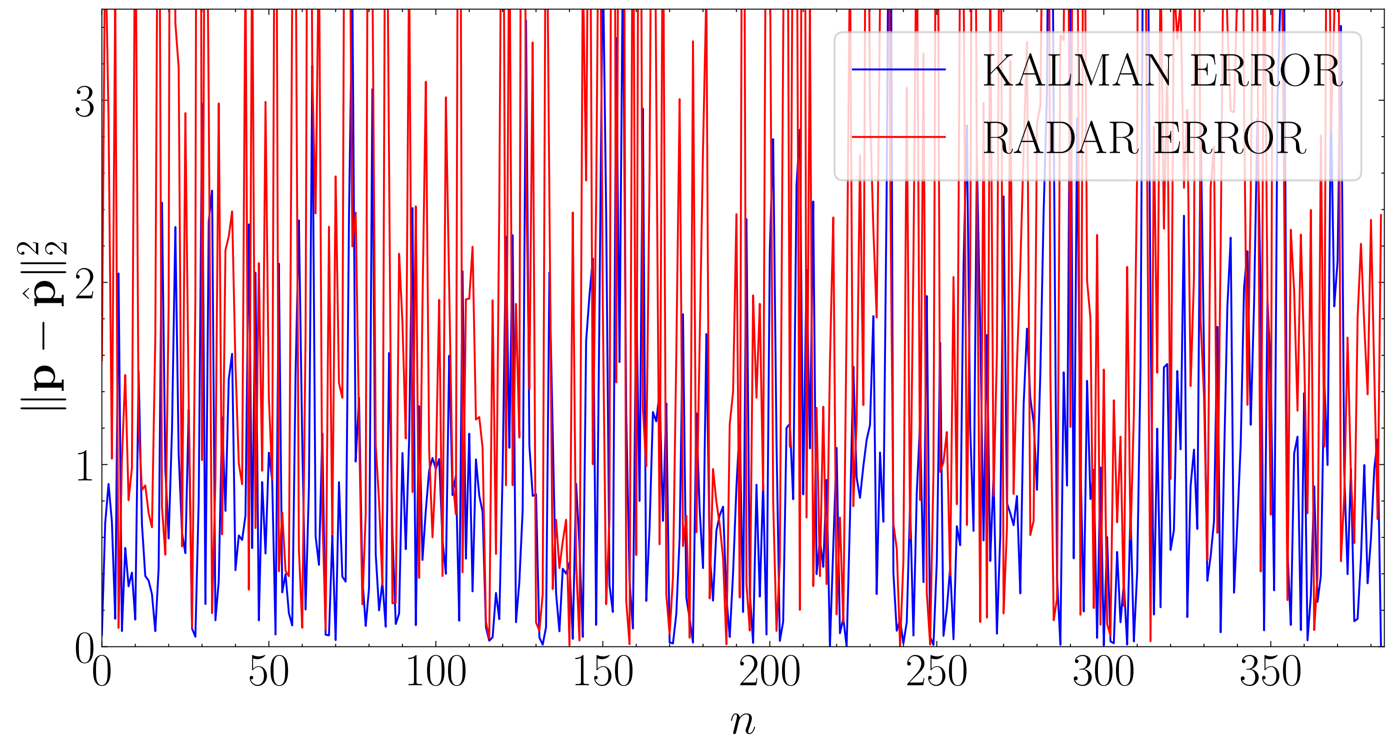
$\mathbf{Q}_v = 0.01\mathbf{I}$





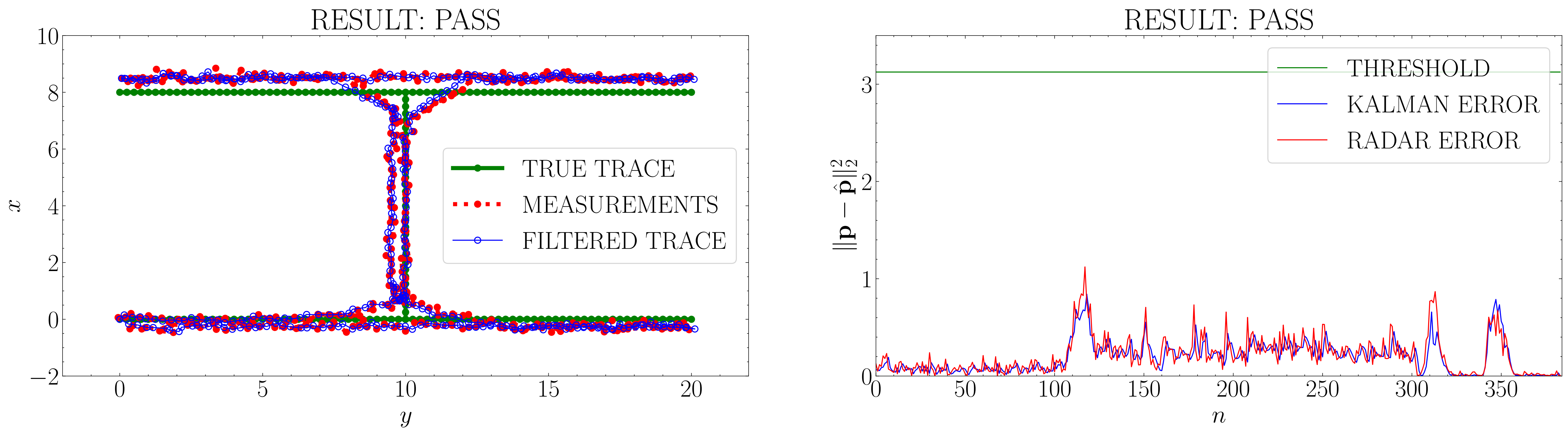
$\mathbf{Q}_v = 0.1\mathbf{I}$



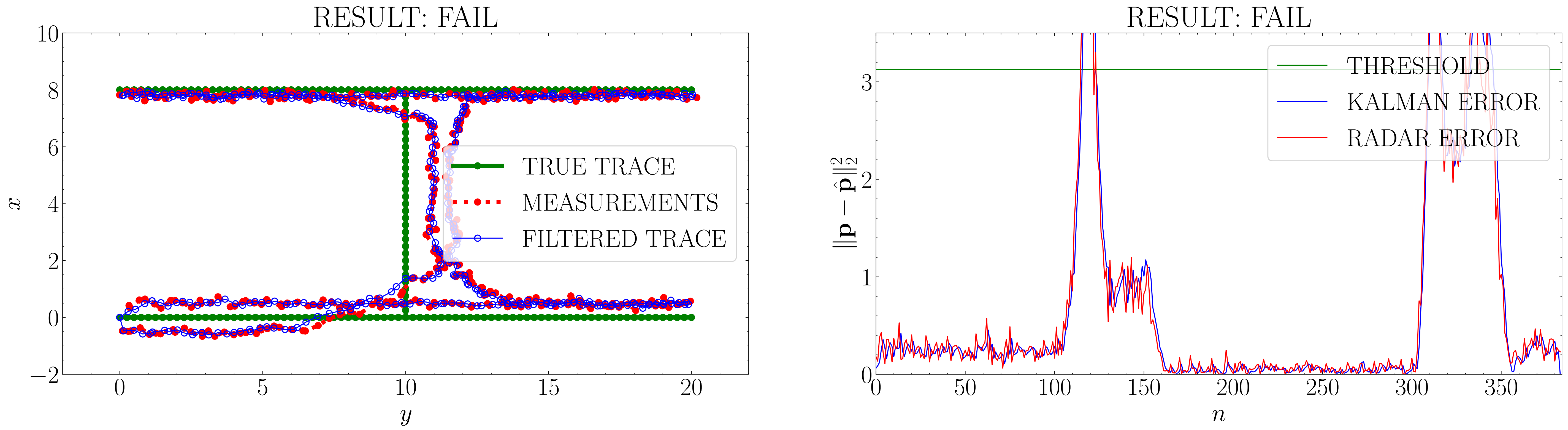


Kalman Filter for Automating Driving Test

Test 1

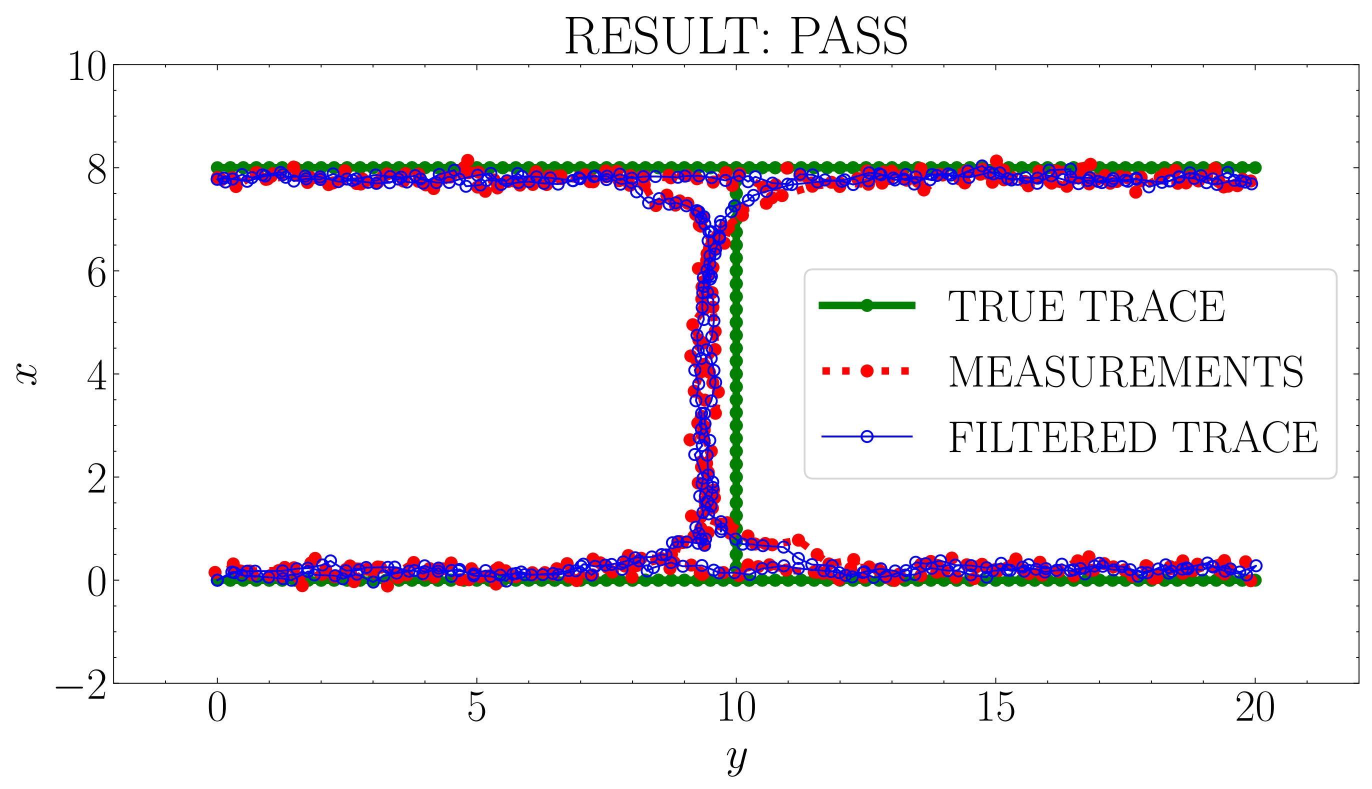


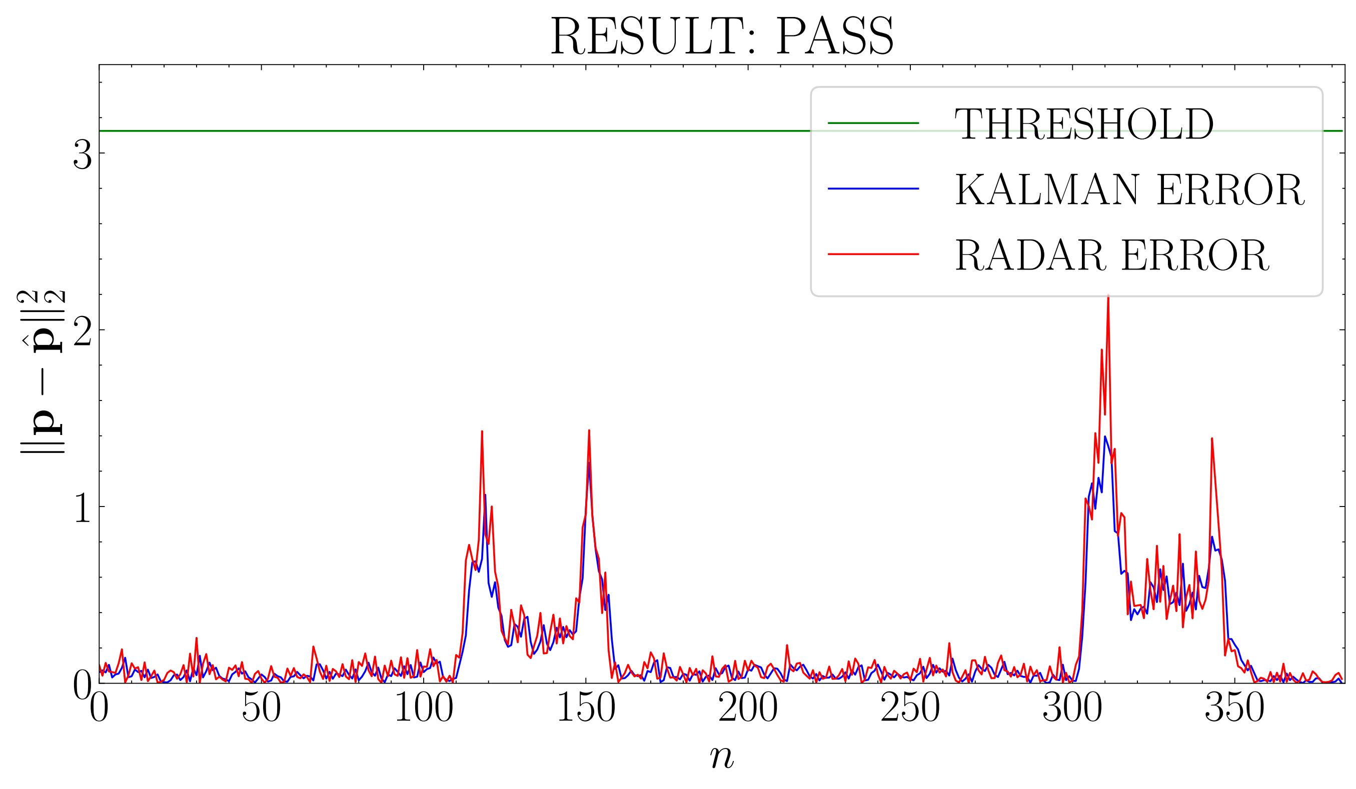
Test 2



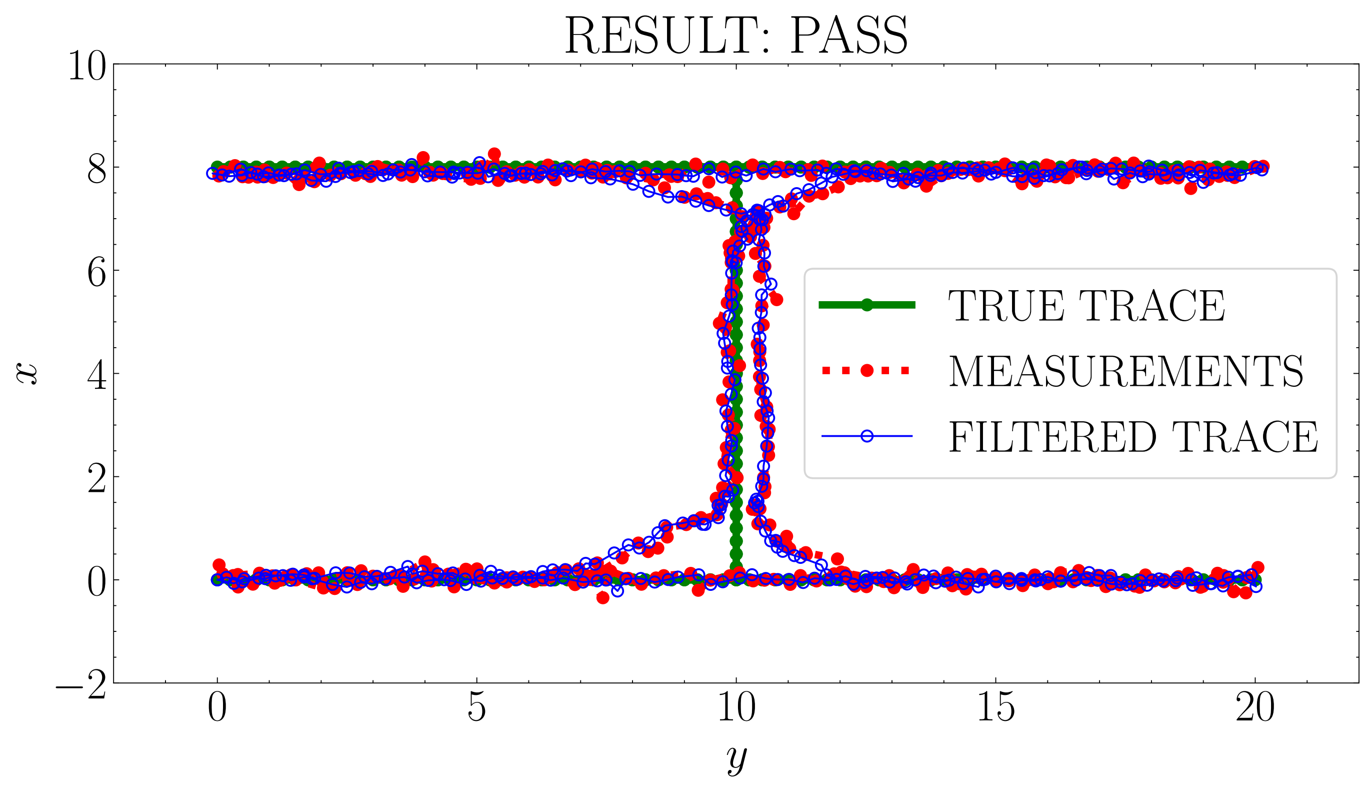
Kalman Filter for Automating Driving Test

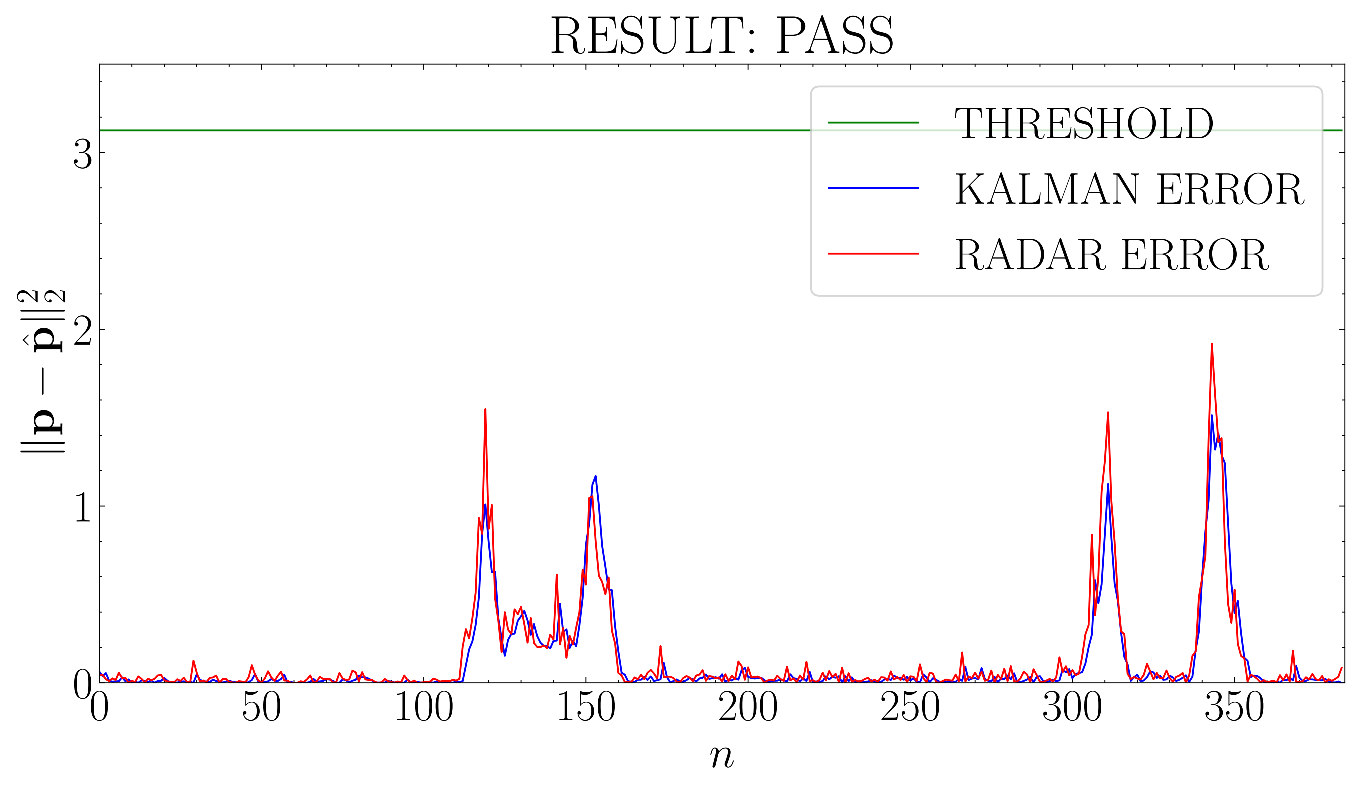
Test 3





Test 4





Thank You!

Please refer https://github.com/kamath-abhijith/Vehicle_Tracking *for reproducible Python codes.*