



ITMO UNIVERSITY

Saint Petersburg, Russia

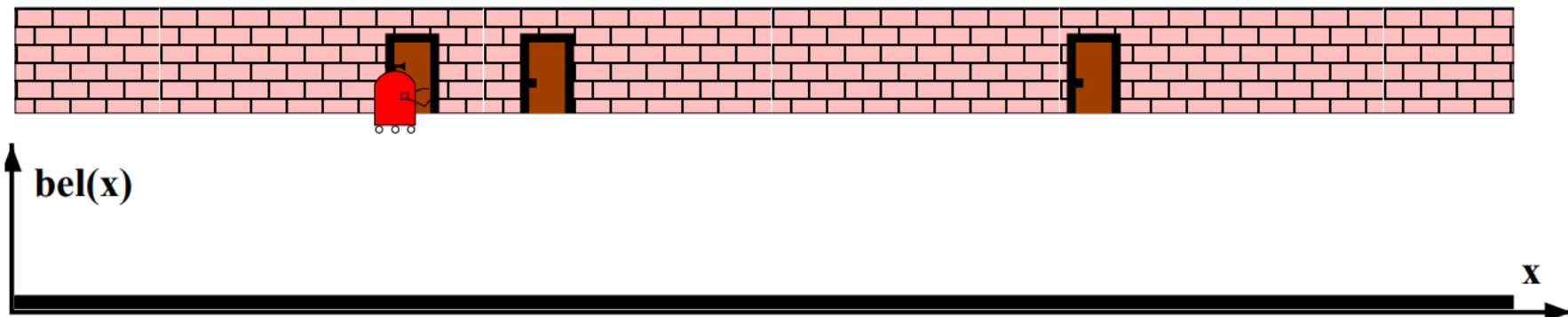
Системы оцувствления роботов

bia@itmo.ru

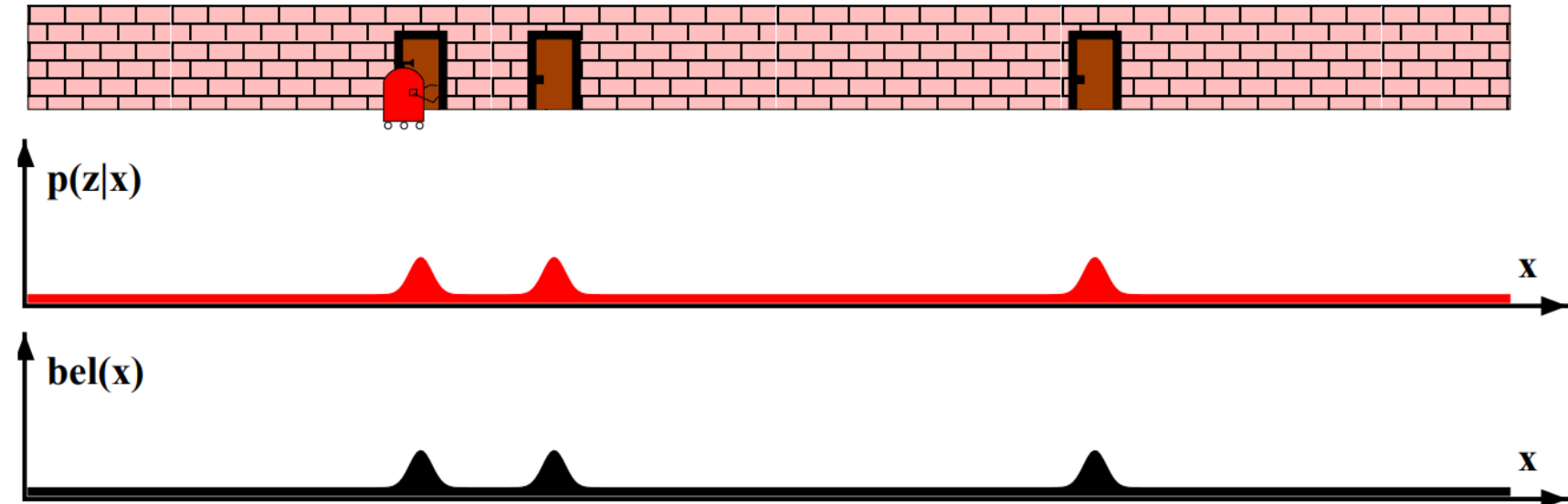
Оценка состояния ... рекурсивно?

Оцените состояние x системы по данным наблюдения z и управления u

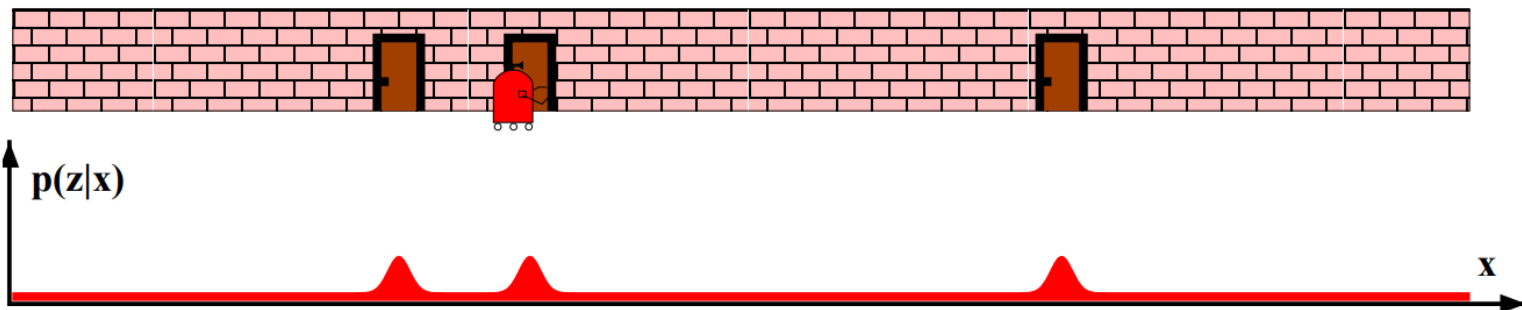
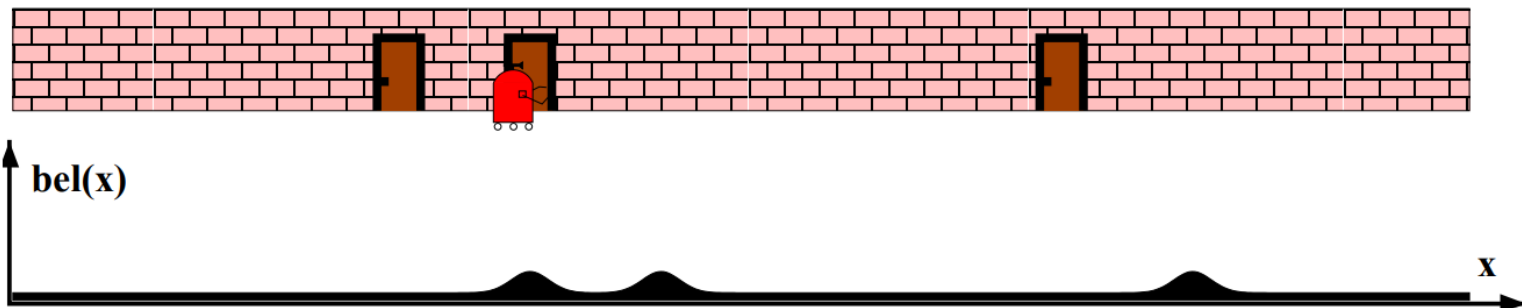
Цель: $bel(x_t) = p(x_t | z_{1:t}, u_{1:t})$



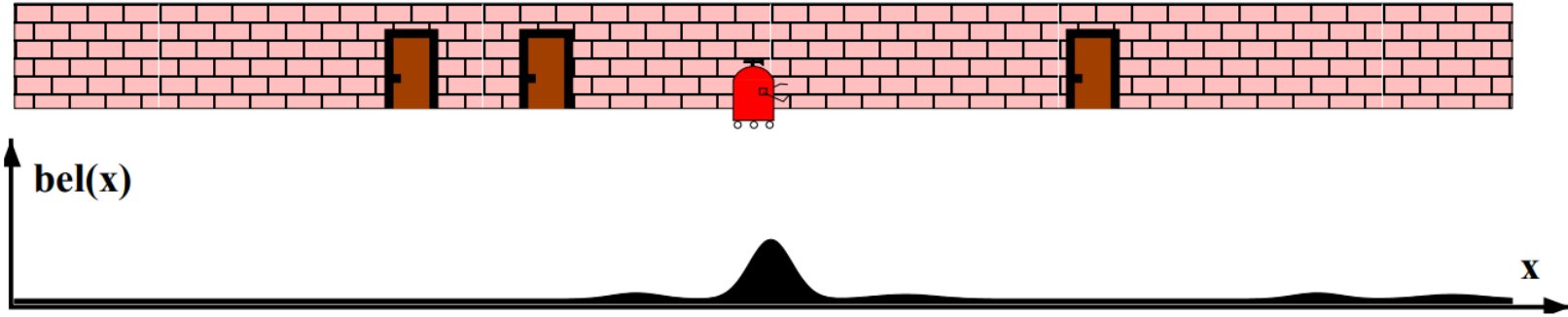
State estimation... recursively?



State estimation... recursively?



State estimation... recursively?



Оценка состояния

- ✓ Байесовская фильтрация – это общий метод для выполнения оценки состояния в рекурсивном режиме.

Может работать в режиме реального времени!

- ✓ Оценить состояние системы x при данных наблюдениях z и управлении u
- ✓ Цель: $bel(x_t) = p(x_t | z_{1:t}, u_{1:t})$

Формулы Байеса

✓ $P(x/y) = \frac{P(y/x)P(x)}{P(y)}; \Rightarrow P(y/x) = \frac{P(x/y)P(y)}{P(x)}$

✓ $p(x_t|z_{1:t}, u_{1:t}) = \frac{p(z_t|x_t, z_{1:t-1}, u_{1:t}) p(x_t|z_{1:t-1}, u_{1:t})}{p(z_t|z_{1:t-1}, u_{1:t})} =$



Скаляр, не зависит от x_t

$\mu p(z_t|x_t, z_{1:t-1}, u_{1:t}) p(x_t|z_{1:t}, u_{1:t})$

$\mu p(z_t|x_t) p(x_t|z_{1:t}, u_{1:t})$

Если мы знаем, где мы находимся в момент времени t , мы уже учли все до этого. Поэтому убираем... Марковское предположение

$\mu p(z_t|x_t) \int p(x_t|x_{t-1}, z_{1:t}, u_{1:t}) p(x_{t-1}|z_{1:t-1}, u_{1:t}) dx_{1-t}$

На основе закона полной вероятности

$\mu p(z_t|x_t) \int p(x_t|x_{t-1}, u_t) p(x_{t-1}|z_{1:t-1}, u_{1:t}) dx_{1-t}$

Марковское допущение

$\mu p(z_t|x_t) \int p(x_t|x_{t-1}, u_t) p(x_{t-1}|z_{1:t-1}, u_{1:t-1}) dx_{1-t}$

Предположение о независимости

$p(x_{t-1}|z_{1:t-1}, u_{1:t-1}) = bel(x_{t-1})$ - Рекурсивная часть

Байесовская фильтрация

Algorithm 1 The general algorithm for Bayes filtering

```
1: for each  $x_t$  do  
2:    $\overline{bel}(x_t) = \int p(x_t|u_t, x_{t-1}) bel(x_{t-1}) dx_{t-1}$             $\triangleright$  transition update  
3:    $bel(x_t) = \eta p(z_t|x_t) \overline{bel}(x_t)$                         $\triangleright$  measurement update  
4: end for each
```

x_t : state z_t : observation u_t : action $bel(x_t)$: belief

$p(x_t|u_t, x_{t-1})$: transition model (motion model)

$p(z_t|x_t)$: measurement model (observation model)

Линейный фильтр Калмана

- ✓ Фильтр Калмана - инструмент для объединения измерений нескольких датчиков, позволяющих в режиме реального времени оценивать состояние роботизированной системы.
- ✓ Задача - взять вероятностную оценку состояния и обновить ее в режиме реального времени, используя два шага: прогнозирование и коррекцию.

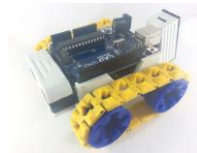
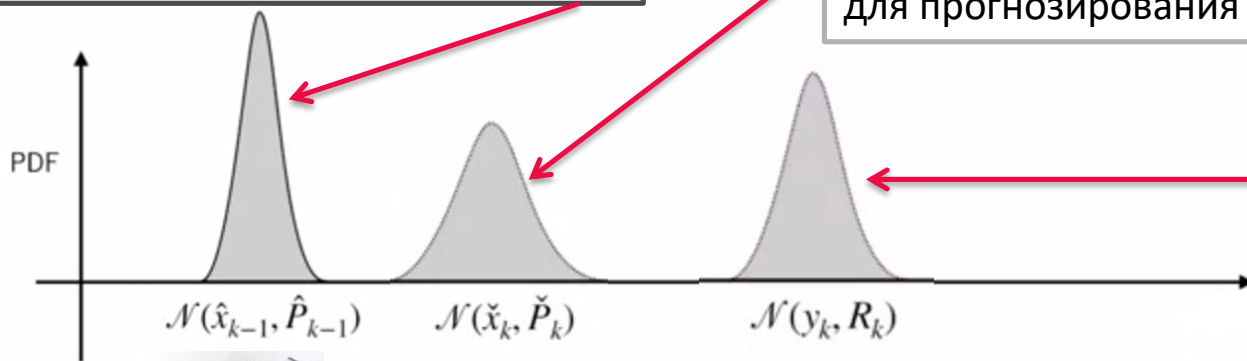
Рассмотрим подробно на примере



Пример 1: Оценить положение объекта в одномерном пространстве.

Берем начальную вероятностную оценку в момент времени $k-1$

Используем модель движения (может быть получена из измерений инерциального датчика), для прогнозирования нашего нового состояния.

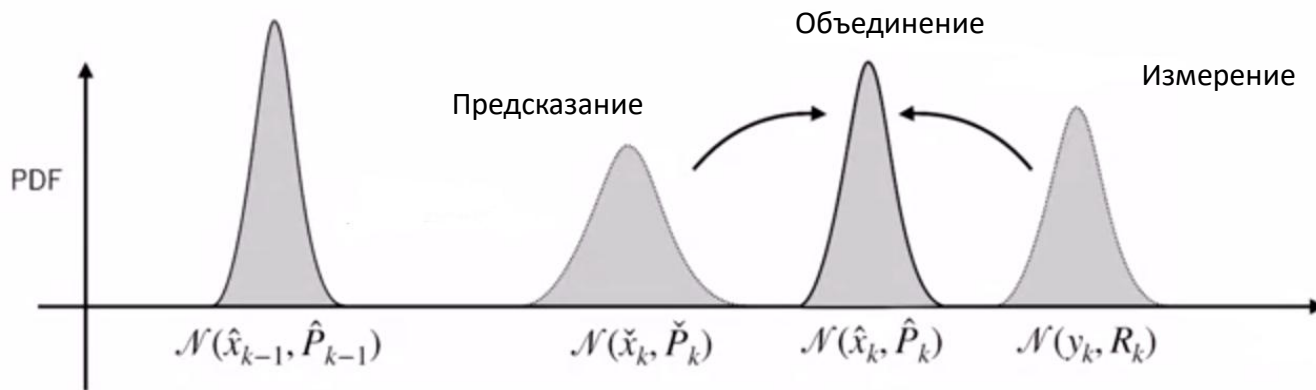


Используем модель наблюдения, полученную, например, из GPS, чтобы скорректировать этот прогноз положения в момент времени k .

Начальная оценка, прогнозируемое состояние и конечное исправленное состояние - это случайные величины, которые мы будем определять с помощью их средних и ковариаций.

Фильтр Калмана – метод объединения информации

от различных датчиков, чтобы получить окончательную оценку некоторого неизвестного состояния, принимая во внимание неопределенность в движении и в измерениях.



Фильтр Калмана

Для фильтра Калмана необходимо уметь строить модель движения и модель измерения:

$$\mathbf{x}_k = \mathbf{F}_{k-1}\mathbf{x}_{k-1} + \mathbf{G}_{k-1}\mathbf{u}_{k-1} + \mathbf{w}_{k-1} \text{ - модель движения}$$

вход

шум

$$\mathbf{y}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{v}_k \text{ - модель измерения}$$

$$\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$$

Шум измерения

$$\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$$

Шум движения или процесса

Рекурсивный МНК + модель процесса

✓ Предсказание

$$\check{\mathbf{x}}_k = \mathbf{F}_{k-1} \mathbf{x}_{k-1} + \mathbf{G}_{k-1} \mathbf{u}_{k-1}$$

$$\check{\mathbf{P}}_k = \mathbf{F}_{k-1} \hat{\mathbf{P}}_{k-1} \mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1}$$

✓ Оптимальное усиление

$$\mathbf{K}_k = \check{\mathbf{P}}_k \mathbf{H}_k^T (\mathbf{H}_k \check{\mathbf{P}}_k \mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

✓ Корректировка

$$\hat{\mathbf{x}}_k = \check{\mathbf{x}}_k + \mathbf{K}_k (\mathbf{y}_k - \mathbf{H}_k \check{\mathbf{x}}_k)$$

$$\hat{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \check{\mathbf{P}}_k$$

$\check{\mathbf{x}}_k$ - Предсказание
заданная модель
движения

$\hat{\mathbf{x}}_k$ - Скорректированное предсказание
(измерение в момент времени k)

Пример использования фильтра Калмана

Положение оценивается
положением и скоростью

$$\mathbf{x} = \begin{bmatrix} p \\ \frac{dp}{dt} = \dot{p} \end{bmatrix}$$

Движение задается через ускорение

$$\mathbf{u} = a = \frac{d^2 p}{dt^2}$$

Модель движения

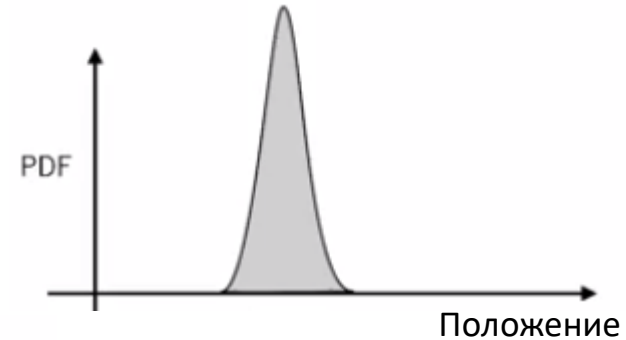
$$\mathbf{x}_k = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \mathbf{x}_{k-1} + \begin{bmatrix} 0 \\ \Delta t \end{bmatrix} \mathbf{u}_{k-1} + \mathbf{w}_{k-1}$$

Модель наблюдения

$$y_k = [1 \quad 0] \mathbf{x}_k + v_k$$

Плотность шума

$$v_k \sim \mathcal{N}(0, 0.05) \quad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, (0.1)\mathbf{I}_{2 \times 2})$$



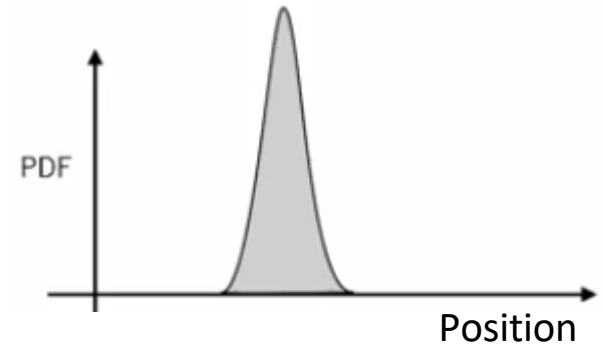
Пример использования фильтра Калмана

Наши данные

$$\hat{\mathbf{x}}_0 \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 5 \end{bmatrix}, \begin{bmatrix} 0.01 & 0 \\ 0 & 1 \end{bmatrix}\right)$$

$$\Delta t = 0.5\text{s}$$

$$u_0 = -2 \text{ [m/s}^2\text{]} \quad y_1 = 2.2 \text{ [m]}$$



Какое будет положение после одной итерации (одного предсказания и одной коррекции)?

Предсказание

$$\begin{aligned}\check{\mathbf{x}}_k &= \mathbf{F}_{k-1}\mathbf{x}_{k-1} + \mathbf{G}_{k-1}\mathbf{u}_{k-1} \\ \begin{bmatrix} \check{p}_1 \\ \check{\dot{p}}_1 \end{bmatrix} &= \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 5 \end{bmatrix} + \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}(-2) = \begin{bmatrix} 2.5 \\ 4 \end{bmatrix}\end{aligned}$$

$$\check{\mathbf{P}}_k = \mathbf{F}_{k-1}\hat{\mathbf{P}}_{k-1}\mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1}$$

$$\check{\mathbf{P}}_1 = \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0.01 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix}^T + \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix} = \begin{bmatrix} 0.36 & 0.5 \\ 0.5 & 1.1 \end{bmatrix}$$

Коррекция

$$\begin{aligned}\mathbf{K}_1 &= \check{\mathbf{P}}_1 \mathbf{H}_1^T (\mathbf{H}_1 \check{\mathbf{P}}_1 \mathbf{H}_1^T + \mathbf{R}_1)^{-1} \\ &= \begin{bmatrix} 0.36 & 0.5 \\ 0.5 & 1.1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \left([1 \ 0] \begin{bmatrix} 0.36 & 0.5 \\ 0.5 & 1.1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 0.05 \right)^{-1} \\ &= \begin{bmatrix} 0.88 \\ 1.22 \end{bmatrix}\end{aligned}$$

$$\begin{aligned}\hat{\mathbf{x}}_1 &= \check{\mathbf{x}}_1 + \mathbf{K}_1 (\mathbf{y}_1 - \mathbf{H}_1 \check{\mathbf{x}}_1) \\ \begin{bmatrix} \hat{p}_1 \\ \hat{\dot{p}}_1 \end{bmatrix} &= \begin{bmatrix} 2.5 \\ 4 \end{bmatrix} + \begin{bmatrix} 0.88 \\ 1.22 \end{bmatrix} (2.2 - [1 \ 0] \begin{bmatrix} 2.5 \\ 4 \end{bmatrix}) = \begin{bmatrix} 2.24 \\ 3.63 \end{bmatrix}\end{aligned}$$

$$\begin{aligned}\hat{\mathbf{P}}_1 &= (\mathbf{I} - \mathbf{K}_1 \mathbf{H}_1) \check{\mathbf{P}}_1 \\ &= \begin{bmatrix} 0.04 & 0.06 \\ 0.06 & 0.49 \end{bmatrix}\end{aligned}$$

Что делает таким «классным» фильтр Калмана?

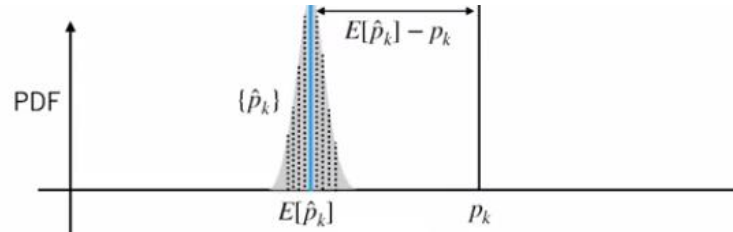
Почему фильтр Калмана так часто используется?

- ✓ Несмещенная и состоятельная оценка



Смещение в оценке состояния

- ✓ Оценка не смещенной является тогда, когда средняя ошибка равна нулю при большом количестве попыток.



- ✓ Если разница между средним и действительным значением не стремится к 0, то такая оценка называется смещенной.
- ✓ Оценка не смещена если $E[\hat{e}_k] = E[\hat{p}_k - p_k] = E[\hat{p}_k] - p_k = 0$

Смещение оценки состояния

Как мы можем проверить, является ли оценка смещенной или нет?

✓ Рассмотрим динамику ошибки.

Задавая наши предсказанные и исправленные ошибки

$$\check{\mathbf{e}}_k = \check{\mathbf{x}}_k - \mathbf{x}_k$$

$$\hat{\mathbf{e}}_k = \hat{\mathbf{x}}_k - \mathbf{x}_k$$

Мы можем воспользоваться уравнениями фильтра Калмана и получить следующее:

$$\check{\mathbf{e}}_k = \mathbf{F}_{k-1} \check{\mathbf{e}}_{k-1} - \mathbf{w}_k$$

$$\hat{\mathbf{e}}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \check{\mathbf{e}}_k + \mathbf{K}_k \mathbf{v}_k$$

Для фильтра Калмана мы можем показать, что ожидаемое значение (матожидание) этих ошибок точно равно нулю.

$$\begin{aligned} E[\check{\mathbf{e}}_k] &= E[\mathbf{F}_{k-1}\check{\mathbf{e}}_{k-1} - \mathbf{w}_k] \\ &= \mathbf{F}_{k-1}E[\check{\mathbf{e}}_{k-1}] - E[\mathbf{w}_k] \\ &= \mathbf{0} \end{aligned}$$

$$\begin{aligned} E[\hat{\mathbf{e}}_k] &= E[(1 - \mathbf{K}_k\mathbf{H}_k)\check{\mathbf{e}}_k + \mathbf{K}_k\mathbf{v}_k] \\ &= (1 - \mathbf{K}_k\mathbf{H}_k)E[\check{\mathbf{e}}_k] + \mathbf{K}_kE[\mathbf{v}_k] \\ &= \mathbf{0} \end{aligned}$$

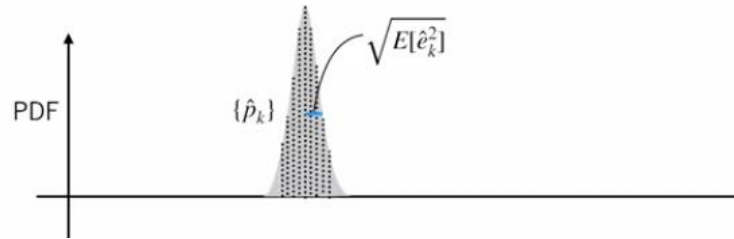
При условии, что наша начальная оценка состояния является несмещенной, а наш шум белым: компоненты некоррелированными и имеют нулевое среднее значение.

$$E[\hat{\mathbf{e}}_0] = \mathbf{0} \quad E[\mathbf{v}] = \mathbf{0} \quad E[\mathbf{w}] = \mathbf{0}$$

Это не гарантирует, что наши оценки будут безошибочными для конкретного случая, а только то, что ожидаемое значение ошибки равно нулю (т.е. при достаточном количестве попыток).

Состоятельность оценки состояния

- ✓ Под состоятельностью мы подразумеваем, что для всех моментов времени k , ковариация фильтра $\{\hat{p}_k\}$ соответствуют ожидаемому значению квадрата нашей ошибки.



- ✓ Для скалярных параметров это означает, что эмпирическая дисперсия нашей оценки должна соответствовать дисперсии, сообщаемой фильтром.

$$E[\hat{e}_k^2] = E[(\hat{p}_k - p_k)^2] = \hat{P}_k$$

Проще говоря, фильтр не является ни слишком уверенным, ни недостаточно уверенным в оценке, которую он произвел.

Состоятельность оценки состояния

Также можно показать (сильно больше математики потребует)

для любого k :

$$E[\check{\mathbf{e}}_k \check{\mathbf{e}}_k^T] = \check{\mathbf{P}}_k \quad E[\hat{\mathbf{e}}_k \hat{\mathbf{e}}_k^T] = \hat{\mathbf{P}}_k$$

Обеспечивается условие

$$E[\hat{\mathbf{e}}_0 \hat{\mathbf{e}}_0^T] = \check{\mathbf{P}}_0 \quad E[\mathbf{v}] = \mathbf{0} \quad E[\mathbf{w}] = \mathbf{0}$$

Т.е. до тех пор, пока наша первоначальная оценка состоятельна, и у нас имеется белый шум с нулевым средним, все оценки будут состоятельными.

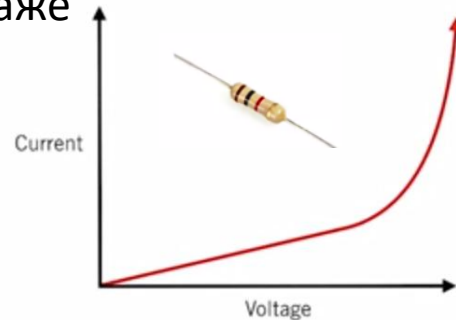
При заданной линейной формулировке, нулевом среднем с белым шумом фильтр Калмана является **несмещенным и состоятельным**.

Применение фильтра Калмана?

✓ Для чего можно применить?

Расширенный фильтр Калмана

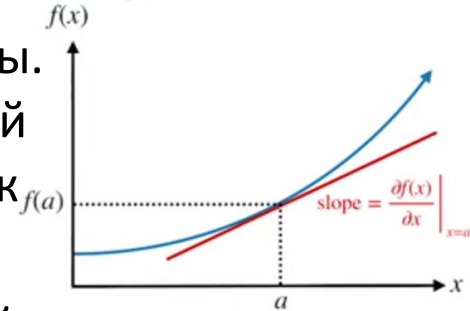
- ✓ Линейный фильтр Калмана нельзя использовать непосредственно для оценки состояний, которые являются нелинейными функциями либо измерений, либо входов управления.
- ✓ Например, ориентация, не является линейным измерением (сферическое).
- ✓ Грустно... В реальности не существует линейных систем. Даже резистор....



Линеаризация нелинейной системы

✓ Так адаптировать фильтр Калмана для нелинейных систем?

Ключевая концепция: линеаризация нелинейной системы. Линеаризация системы означает просто выбор некоторой рабочей точки a и нахождение линейного приближения к нелинейной функции в окрестности a .



В двух измерениях это означает нахождение касательной к функции $f(x)$, когда x равно a . Математически мы делаем это через разложение функции в ряд Тейлора.

$$f(x) \approx f(a) + \left. \frac{\partial f(x)}{\partial x} \right|_{x=a} (x - a) + \frac{1}{2!} \left. \frac{\partial^2 f(x)}{\partial x^2} \right|_{x=a} (x - a)^2 + \frac{1}{3!} \left. \frac{\partial^3 f(x)}{\partial x^3} \right|_{x=a} (x - a)^3 + \dots$$

Линеаризация нелинейной системы

Для линеаризации нужна только первая часть ряда

$$f(x) \approx f(a) + \left. \frac{\partial f(x)}{\partial x} \right|_{x=a} (x - a)$$

$$\mathbf{x}_k = \mathbf{F}_{k-1} \mathbf{x}_{k-1} + \mathbf{G}_{k-1} \mathbf{u}_{k-1} + \mathbf{w}_{k-1}$$



$$\mathbf{x}_k = \mathbf{f}_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1})$$

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k$$



$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k, \mathbf{v}_k)$$

Как нам выбрать рабочую точку для линеаризации?

В идеале, мы хотели бы сделать это около истинного значения состояния...

Но если бы мы знали истинное значение, нам бы не надо было все это...

Линеаризация нелинейной системы

Для нашей модели движения мы будем линеаризовать апостериорную оценку предыдущего состояния

$$\mathbf{x}_k = \mathbf{f}_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) \approx \underbrace{\mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}) + \left. \frac{\partial \mathbf{f}_{k-1}}{\partial \mathbf{x}_{k-1}} \right|_{\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}} (\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1})}_{\mathbf{F}_{k-1}} + \underbrace{\left. \frac{\partial \mathbf{f}_{k-1}}{\partial \mathbf{w}_{k-1}} \right|_{\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}} \mathbf{w}_{k-1}}_{\mathbf{L}_{k-1}}$$

а для модели измерения мы будем линеаризовать наше предсказание текущего состояния на основе модели движения.

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k, \mathbf{v}_k) \approx \underbrace{\mathbf{h}_k(\check{\mathbf{x}}_k, \mathbf{0}) + \left. \frac{\partial \mathbf{h}_k}{\partial \mathbf{x}_k} \right|_{\check{\mathbf{x}}_k, \mathbf{0}} (\mathbf{x}_k - \check{\mathbf{x}}_k)}_{\mathbf{H}_k} + \underbrace{\left. \frac{\partial \mathbf{h}_k}{\partial \mathbf{v}_k} \right|_{\check{\mathbf{x}}_k, \mathbf{0}} \mathbf{v}_k}_{\mathbf{M}_k}$$

Якобиан

Матрица отражает как быстро меняется каждый выход функции по каждому входному измерению. Смысл аналогичен тому, как производная скалярной функции говорит, как быстро меняется выход, когда мы меняем вход.

Якобиан - обобщение первой производной на множественные измерения.

Пример двумерной функции с двумя входами, какой Якобиан будет?

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} x_1 + x_2 \\ x_1^2 \end{bmatrix}$$

$$\frac{\partial \mathbf{f}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial \mathbf{f}}{\partial x_1} & \dots & \frac{\partial \mathbf{f}}{\partial x_n} \end{bmatrix} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \dots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

Теперь все вместе

С линеаризованными моделями и Якобианами можно теперь использовать уравнения фильтра Калмана.

$$\mathbf{x}_k = \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}) + \mathbf{F}_{k-1}(\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}) + \mathbf{L}_{k-1}\mathbf{w}_{k-1} \quad - \text{Линеаризованная модель движения}$$

$$\mathbf{y}_k = \mathbf{h}_k(\check{\mathbf{x}}_k, \mathbf{0}) + \mathbf{H}_k(\mathbf{x}_k - \check{\mathbf{x}}_k) + \mathbf{M}_k\mathbf{v}_k \quad - \text{Линеаризованная модель измерения}$$

Предсказание

$$\check{\mathbf{x}}_k = \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0})$$

$$\check{\mathbf{P}}_k = \mathbf{F}_{k-1}\hat{\mathbf{P}}_{k-1}\mathbf{F}_{k-1}^T + \mathbf{L}_{k-1}\mathbf{Q}_{k-1}\mathbf{L}_{k-1}^T$$

Расчет оптимального коэффициента

$$\mathbf{K}_k = \check{\mathbf{P}}_k\mathbf{H}_k^T(\mathbf{H}_k\check{\mathbf{P}}_k\mathbf{H}_k^T + \mathbf{M}_k\mathbf{R}_k\mathbf{M}_k^T)^{-1}$$

Коррекция

$$\hat{\mathbf{x}}_k = \check{\mathbf{x}}_k + \mathbf{K}_k(\mathbf{y}_k - \mathbf{h}_k(\check{\mathbf{x}}_k, \mathbf{0}))$$

$$\hat{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\check{\mathbf{P}}_k$$

$\check{\mathbf{x}}_k$ - Предсказание

$\hat{\mathbf{x}}_k$ - Исправленное
предсказание

Итого

Расширенная версия имеет несколько отличий:

- ✓ на этапах прогнозирования и коррекции мы по-прежнему используем нелинейные модели для наследования среднего значения оценки состояния и для вычисления поправки.

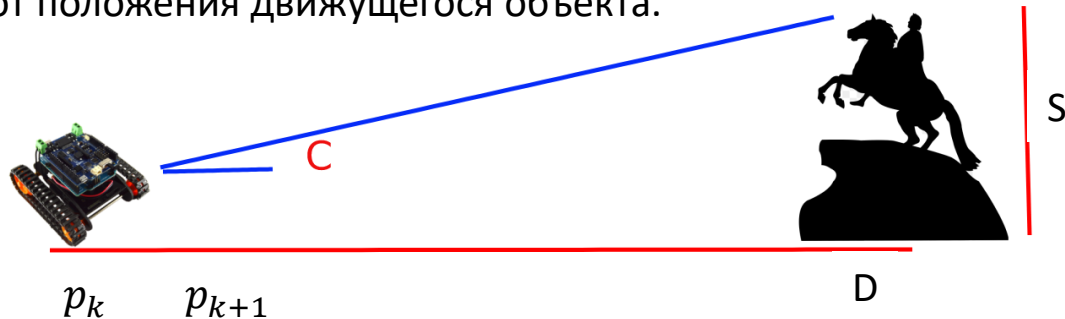
Мы линеаризовали нашу модель движения относительно предыдущей оценки состояния и линеаризуем модель измерения относительно предсказанного состояния. По определению, линеаризованная модель в точности совпадает с нелинейной моделью в рабочей точке.

- ✓ Появились матрицы F_{k-1} , L_{k-1} , H_k , M_k - якобианы системы, связанные с шумом процесса и шумом измерения.

Задача

Задача: найти положение в 1D пространстве, только в отличие от примера 1 (где измеряли изменение положения по данным GPS) используем данные с камеры, измеряя высоту отдаленных ориентиров относительно горизонта.

- ✓ Сохраним ту же модель линейного движения, что и в исходном примере, и предположим, что мы знаем высоту ориентира и его положение в глобальной системе координат.
- ✓ Поскольку наш датчик измеряет угол, наша измерительная модель имеет нелинейную зависимость от положения движущегося объекта.



Задача

Объект характеризуется положением и скоростью.
Управление: по ускорению.

$$\mathbf{x} = \begin{bmatrix} p \\ \dot{p} \end{bmatrix} \quad \mathbf{u} = \ddot{p}$$

Модель движения

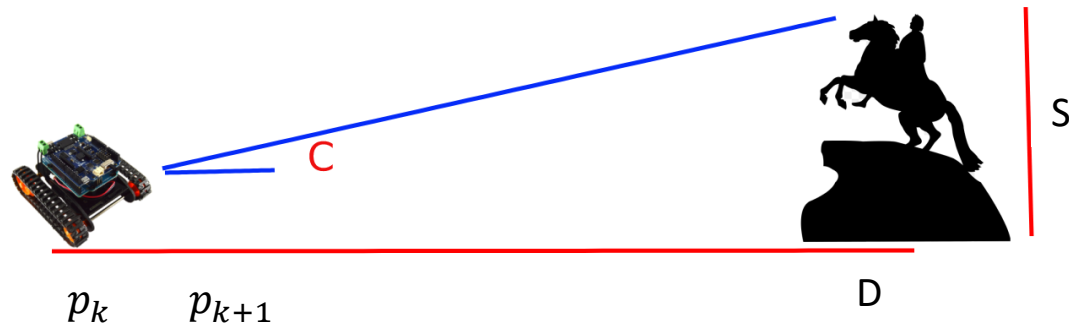
$$\begin{aligned} \mathbf{x}_k &= \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) \\ &= \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \mathbf{x}_{k-1} + \begin{bmatrix} 0 \\ \Delta t \end{bmatrix} \mathbf{u}_{k-1} + \mathbf{w}_{k-1} \end{aligned}$$

$$v_k \sim \mathcal{N}(0, 0.01) \quad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, (0.1)\mathbf{1}_{2 \times 2})$$

Модель измерения

$$\begin{aligned} y_k &= \phi_k = h(p_k, v_k) \\ &= \tan^{-1} \left(\frac{S}{D - p_k} \right) + v_k \end{aligned}$$

S (высота), D (расстояние) —
знаем заранее



Задача

Якобиан модели движения

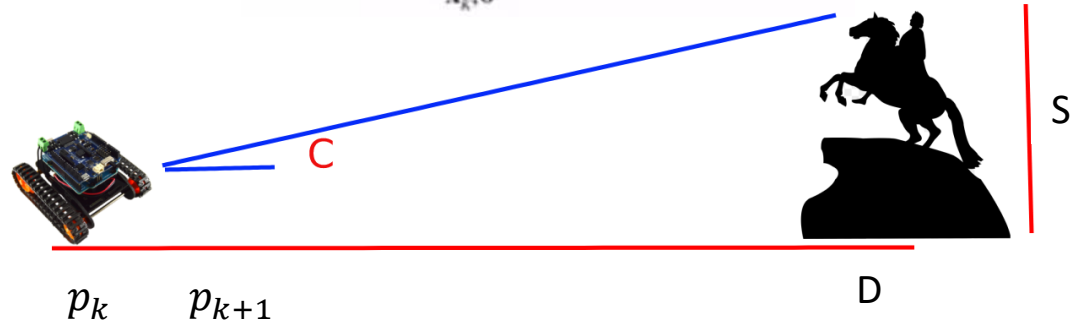
$$\mathbf{F}_{k-1} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}_{k-1}} \right|_{\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{L}_{k-1} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{w}_{k-1}} \right|_{\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, \mathbf{0}} = \mathbf{I}_{2 \times 2}$$

Якобиан модели измерения

$$\mathbf{H}_k = \left. \frac{\partial h}{\partial \mathbf{x}_k} \right|_{\check{\mathbf{x}}_k, \mathbf{0}} = \begin{bmatrix} \frac{S}{(D - \check{p}_k)^2 + S^2} & 0 \end{bmatrix}$$

$$M_k = \left. \frac{\partial h}{\partial v_k} \right|_{\check{\mathbf{x}}_k, \mathbf{0}} = 1$$



Задача

Необходимо найти положение объекта в момент времени $k + 1$, при $\Delta t = 0.5$ сек

$$\hat{\mathbf{x}}_0 \sim \mathcal{N} \left(\begin{bmatrix} 0 \\ 5 \end{bmatrix}, \begin{bmatrix} 0.01 & 0 \\ 0 & 1 \end{bmatrix} \right)$$

$$u_0 = -2 \text{ [m/s}^2\text{]} \quad y_1 = \pi/6 \text{ [rad]}$$

$$S = 20 \text{ [m]} \quad D = 40 \text{ [m]}$$

Задача.... Предсказание

$$\check{\mathbf{x}}_1 = \mathbf{f}_0(\hat{\mathbf{x}}_0, \mathbf{u}_0, \mathbf{0})$$

$$\begin{bmatrix} \check{p}_1 \\ \check{\dot{p}}_1 \end{bmatrix} = \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 5 \end{bmatrix} + \begin{bmatrix} 0 \\ 0.5 \end{bmatrix} (-2) = \begin{bmatrix} 2.5 \\ 4 \end{bmatrix}$$

$$\check{\mathbf{P}}_1 = \mathbf{F}_0 \hat{\mathbf{P}}_0 \mathbf{F}_0^T + \mathbf{L}_0 \mathbf{Q}_0 \mathbf{L}_0^T$$

$$\check{\mathbf{P}}_1 = \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0.01 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0.5 & 1 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.36 & 0.5 \\ 0.5 & 1.1 \end{bmatrix}$$

Такое же значение как с линейным фильтром Калмана, т.к. модель движения линейная.

Задача.... Корректировка

$$\mathbf{K}_1 = \check{\mathbf{P}}_1 \mathbf{H}_1^T (\mathbf{H}_1 \check{\mathbf{P}}_1 \mathbf{H}_1^T + \mathbf{M}_1 \mathbf{R}_1 \mathbf{M}_1^T)^{-1}$$

$$= \begin{bmatrix} 0.36 & 0.5 \\ 0.5 & 1.1 \end{bmatrix} \begin{bmatrix} 0.011 \\ 0 \end{bmatrix} \left([0.011 \quad 0] \begin{bmatrix} 0.36 & 0.5 \\ 0.5 & 1.1 \end{bmatrix} \begin{bmatrix} 0.011 \\ 0 \end{bmatrix} + 1(0.01)(1) \right)^{-1}$$

$$= \begin{bmatrix} 0.40 \\ 0.55 \end{bmatrix}$$

$$\hat{\mathbf{x}}_1 = \check{\mathbf{x}}_1 + \mathbf{K}_1 (\mathbf{y}_1 - \mathbf{h}_1(\check{\mathbf{x}}_1, \mathbf{0}))$$

$$\begin{bmatrix} \hat{p}_1 \\ \hat{\dot{p}}_1 \end{bmatrix} = \begin{bmatrix} 2.5 \\ 4 \end{bmatrix} + \begin{bmatrix} 0.40 \\ 0.55 \end{bmatrix} (0.52 - 0.49) = \begin{bmatrix} 2.51 \\ 4.02 \end{bmatrix}$$

$$\hat{\mathbf{P}}_1 = (\mathbf{I} - \mathbf{K}_1 \mathbf{H}_1) \check{\mathbf{P}}_1$$

$$= \begin{bmatrix} 0.36 & 0.50 \\ 0.50 & 1.1 \end{bmatrix}$$

Дополнительные материалы

- ✓ <https://www.embedded.com/using-nonlinear-kalman-filtering-to-estimate-signals/>
- ✓ Dan Simon, Optimal State Estimation, 2006.

Спасибо за внимание!

IT'sMOre than a
UNIVERSITY