

Нелинейные фильтры Калмана и их гибриды

Программная документация

1	Нелинейные фильтры Калмана и их гибриды (Nonlinear Kalman filters and their hybrids)	1
2	Алфавитный указатель групп	9
2.1	Группы	9
3	Алфавитный указатель пространств имен	11
3.1	Пространства имен	11
4	Иерархический список классов	13
4.1	Иерархия классов	13
5	Алфавитный указатель классов	15
5.1	Классы	15
6	Группы	17
6.1	Kalman_filters	17
6.1.1	Подробное описание	17
7	Пространства имен	19
7.1	Пространство имен KalmanFilters	19
7.1.1	Подробное описание	20
7.1.2	Функции	20
8	Классы	21
8.1	Шаблон класса KalmanFilters::CKalmanCKF< SizeX, SizeY >	21
8.1.1	Подробное описание	23
8.1.2	Конструктор(ы)	23
8.1.3	Методы	23
8.1.4	Данные класса	25
8.2	Шаблон класса KalmanFilters::CKalmanECKF< SizeX, SizeY >	27
8.2.1	Подробное описание	28
8.2.2	Конструктор(ы)	29
8.2.3	Методы	29
8.3	Шаблон класса KalmanFilters::CKalmanEKF< SizeX, SizeY >	30
8.3.1	Подробное описание	31
8.3.2	Конструктор(ы)	32
8.3.3	Методы	32
8.3.4	Данные класса	34
8.4	Шаблон класса KalmanFilters::CKalmanEUKF< SizeX, SizeY >	35
8.4.1	Подробное описание	37
8.4.2	Конструктор(ы)	37
8.4.3	Методы	38
8.5	Шаблон класса KalmanFilters::CKalmanLKF< SizeX, SizeY >	39
8.5.1	Подробное описание	42
8.5.2	Конструктор(ы)	42
8.5.3	Методы	42

8.5.4 Данные класса	50
8.6 Шаблон класса <code>KalmanFilters::CKalmanSRCKF< SizeX, SizeY ></code>	53
8.6.1 Подробное описание	54
8.6.2 Конструктор(ы)	55
8.6.3 Методы	55
8.7 Шаблон класса <code>KalmanFilters::CKalmanSRCKFB< SizeX, SizeY ></code>	56
8.7.1 Подробное описание	58
8.7.2 Конструктор(ы)	59
8.7.3 Методы	59
8.8 Шаблон класса <code>KalmanFilters::CKalmanSRECKF< SizeX, SizeY ></code>	59
8.8.1 Подробное описание	60
8.8.2 Конструктор(ы)	60
8.8.3 Методы	60
8.9 Шаблон класса <code>KalmanFilters::CKalmanSRECKFB< SizeX, SizeY ></code>	61
8.9.1 Подробное описание	62
8.9.2 Конструктор(ы)	62
8.9.3 Методы	62
8.10 Шаблон класса <code>KalmanFilters::CKalmanSREKF< SizeX, SizeY ></code>	62
8.10.1 Подробное описание	63
8.10.2 Конструктор(ы)	64
8.10.3 Методы	64
8.11 Шаблон класса <code>KalmanFilters::CKalmanSREUKF< SizeX, SizeY ></code>	65
8.11.1 Подробное описание	66
8.11.2 Конструктор(ы)	66
8.11.3 Методы	66
8.11.4 Данные класса	67
8.12 Шаблон класса <code>KalmanFilters::CKalmanSREUKFB< SizeX, SizeY ></code>	67
8.12.1 Подробное описание	68
8.12.2 Конструктор(ы)	68
8.12.3 Методы	69
8.12.4 Данные класса	69
8.13 Шаблон класса <code>KalmanFilters::CKalmanSRUKF< SizeX, SizeY ></code>	69
8.13.1 Подробное описание	71
8.13.2 Конструктор(ы)	71
8.13.3 Методы	71
8.13.4 Данные класса	74
8.14 Шаблон класса <code>KalmanFilters::CKalmanSRUKFB< SizeX, SizeY ></code>	74
8.14.1 Подробное описание	76
8.14.2 Конструктор(ы)	77
8.14.3 Методы	77
8.14.4 Данные класса	77
8.15 Шаблон класса <code>KalmanFilters::CKalmanUKF< SizeX, SizeY ></code>	78
8.15.1 Подробное описание	80

8.15.2 Конструктор(ы)	80
8.15.3 Методы	80
8.15.4 Данные класса	83
Предметный указатель	87

Раздел 1

Нелинейные фильтры Калмана и их гибриды (Nonlinear Kalman filters and their hybrids)

-
- 1) ЛФК - Линейный Фильтр Калмана (LKF - Linear Kalman Filter);
 - 2) РФК - Расширенный Фильтр Калмана (ЕКФ - Extended Kalman Filter);
 - 3) ККРФК - Квадратно-Корневой Расширенный Фильтр Калмана (SREKF - Square Root Extended Kalman Filter);
 - 4) СТФК - Сигма-точечный (ансцентный) Фильтр Калмана (UKF - Unscented Kalman Filter);
 - 5) КК-СТФК - Квадратно-Корневой Сигма-точечный Фильтр Калмана (SRUKF - Square Root Unscented Kalman Filter);
 - 6) КК-СТФКБ - Блочная реализация КК-СТФК (SRUKFB - Square Root Unscented Kalman Filter Block);
 - 7) КФК - Кубатурный Фильтр Калмана (СКФ - Cubature Kalman Filter);
 - 8) КК-КФК - Квадратно-Корневой Кубатурный Фильтр Калмана (SRCKF - Square Root Cubature Kalman Filter);
 - 9) КК-КФКБ - Блочная реализация КК-КФК (SRCKFB - Square Root Cubature Kalman Filter Block);
 - 10) РСТФК - Расширенно-Сигма-точечный Фильтр Калмана (EUKF - Extended Unscented Kalman Filter);
 - 11) КК-РСТФК - Квадратно-Корневой Расширенно-Сигма-точечный Фильтр Калмана (SREUKF - Square Root Extended Unscented Kalman Filter);
 - 12) КК-РСТФКБ - Блочная реализация КК-РСТФК (SREUKFB - Square Root Extended Unscented Kalman Filter Block);
 - 13) РКФК - Расширенно-Кубатурный Фильтр Калмана (ЕКФ - Extended Cubature Kalman Filter);
 - 14) КК-РКФК - Квадратно-Корневой Расширенно-Кубатурный Фильтр Калмана (SRECKF - Square Root Extended Cubature Kalman Filter);
 - 15) КК-РКФКБ - Блочная реализация КК-РКФК (SRECKFB - Square Root Extended Cubature Kalman Filter Block).
-

Выбранная система координат:

$$X = \{ X, Y, V, K, dK/dt \}, Y = \{ R, Az, Vr \},$$

где пространство состояния:

- X, Y - плоскостные декартовы координаты, км;
- V - полная скорость, м/с;
- K - курс, град;
- dK/dt - скорость изменения курса, град/с;

пространство измерений:

- R - дальность, км;
- Az - азимут, град;
- Vr - радиальная скорость, м/с;

ЛФК не может быть применен при выбранной системе координат напрямую, поэтому не приводится далее.

Начальное состояние объекта:

- X = 100 км;
- Y = 100 км;
- V = 100 м/с;
- K = 45 град;
- $dK/dt = 0$ град/с;

СКО измерений:

- RMS_R = 0.048 км;
- RMS_Az = 0.008 град;
- RMS_Vr = 0.016 м/с;

Рис.1 - RMSE вектора состояния при изменении параметра разброса сигма-точек $\alpha=1.0...0.35$ (от светлого к темному), $\beta=2$, $\kappa=-2$

<? <?

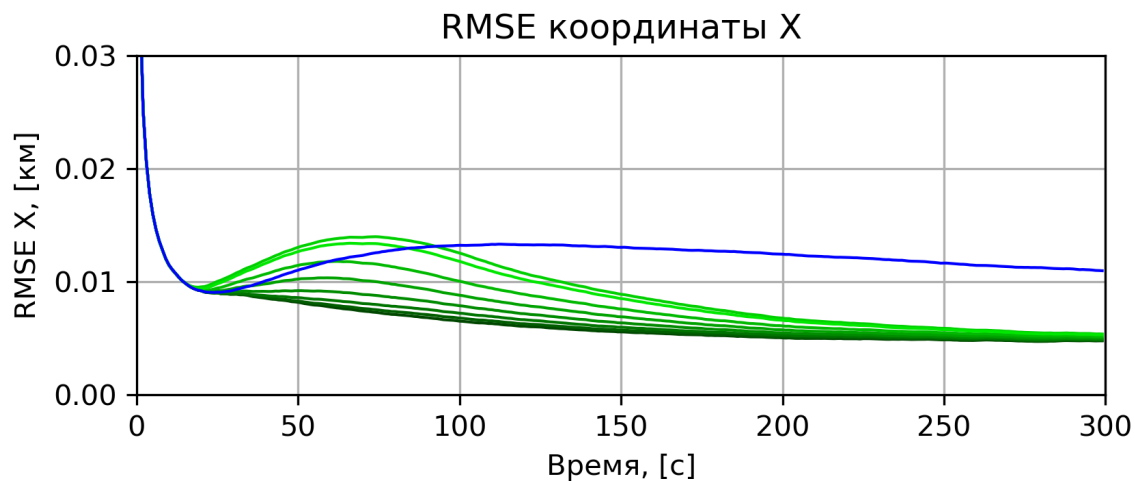


Рис. 1.1 ?>

<? <?

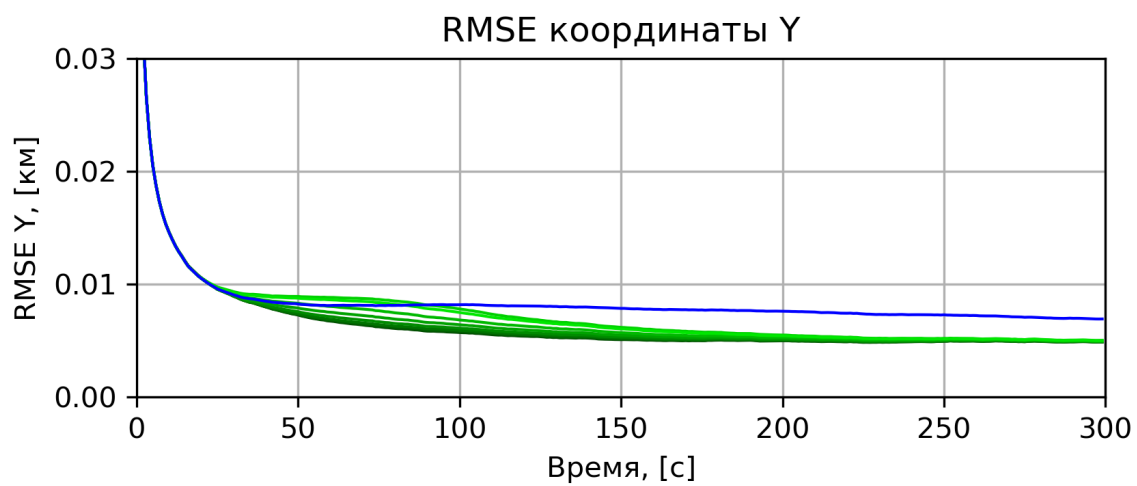


Рис. 1.2 ?>

<? <?

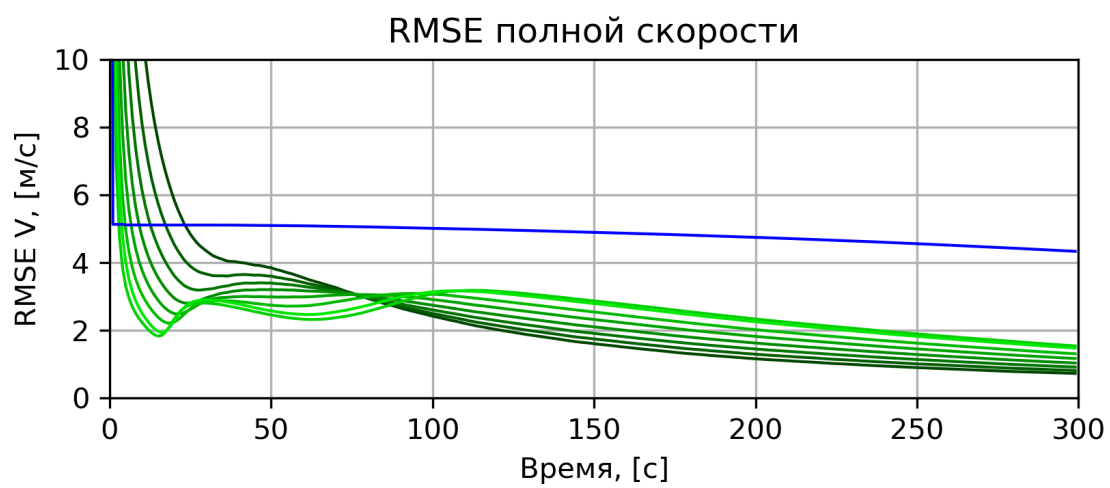


Рис. 1.3 ?>

<? <?

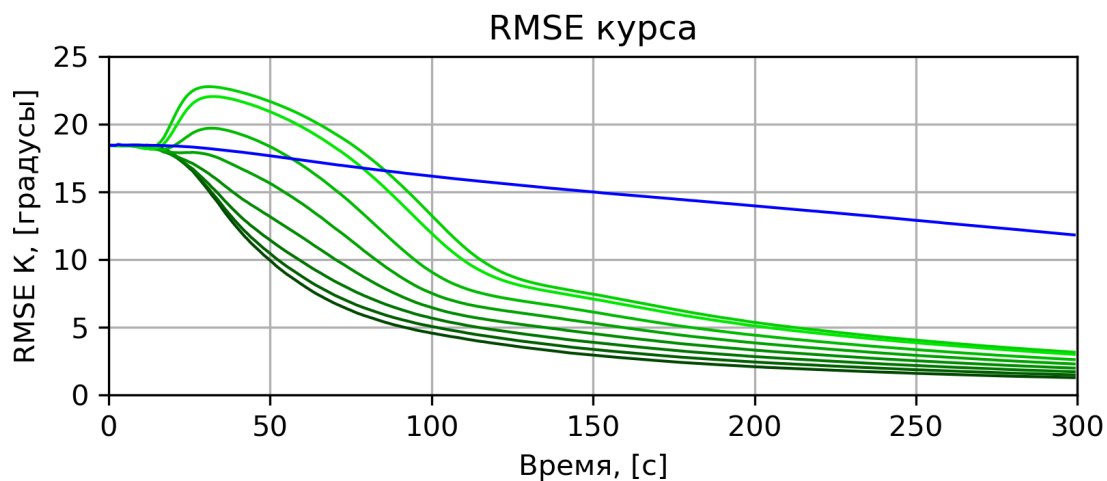


Рис. 1.4 ?>

<? <?

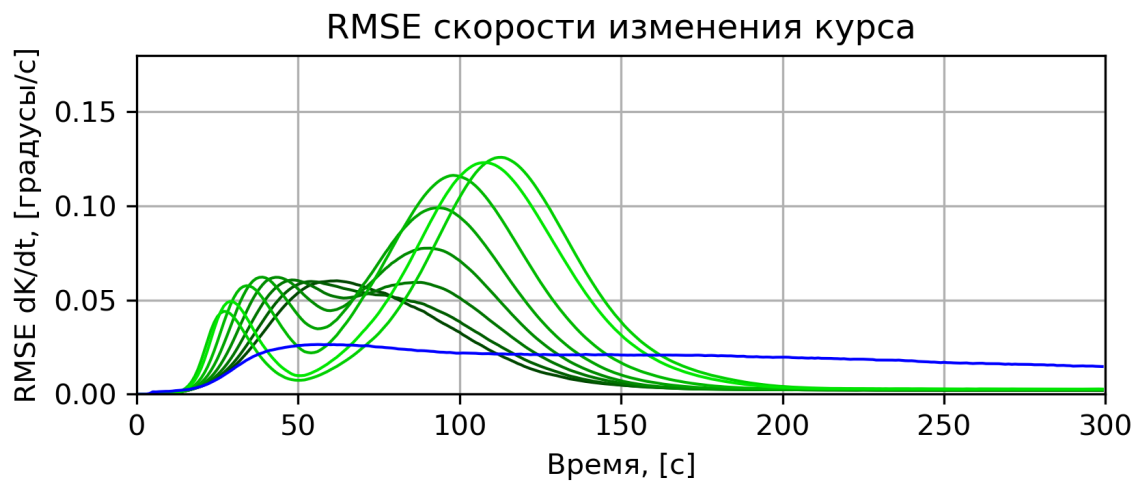


Рис. 1.5 ?>

Рис.2 - Сравнение RMSE вектора состояния

<? <?

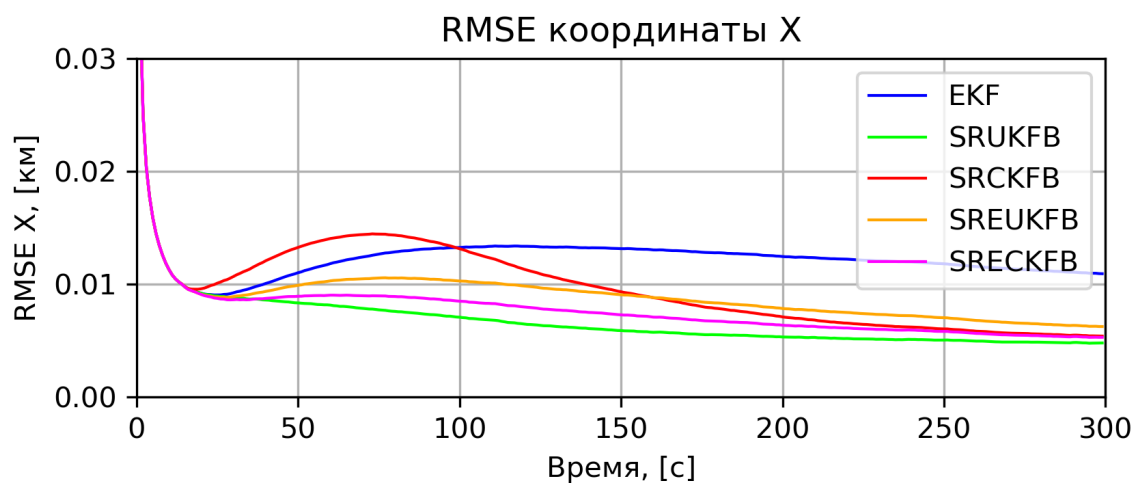


Рис. 1.6 ?>

<? <?

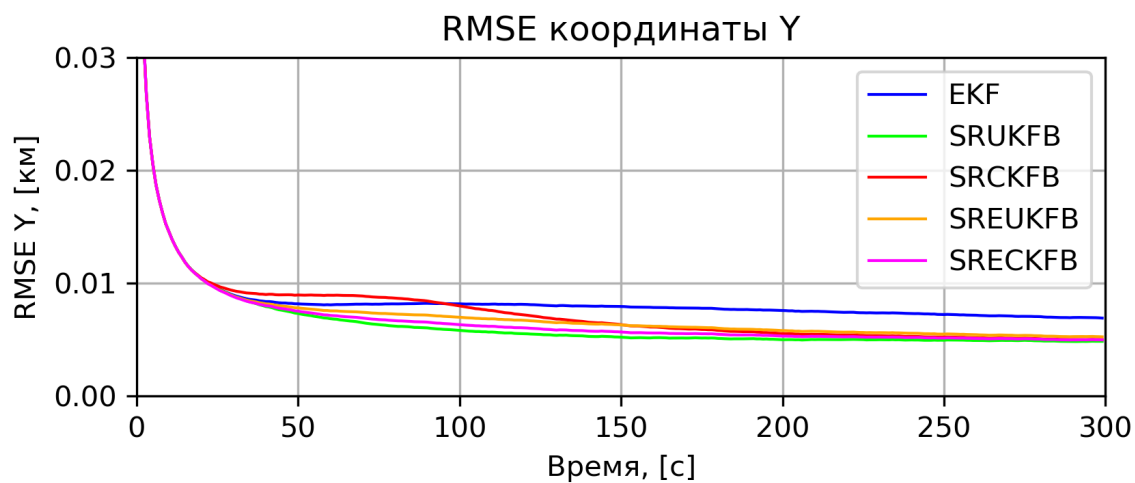


Рис. 1.7 ?>

<? <?

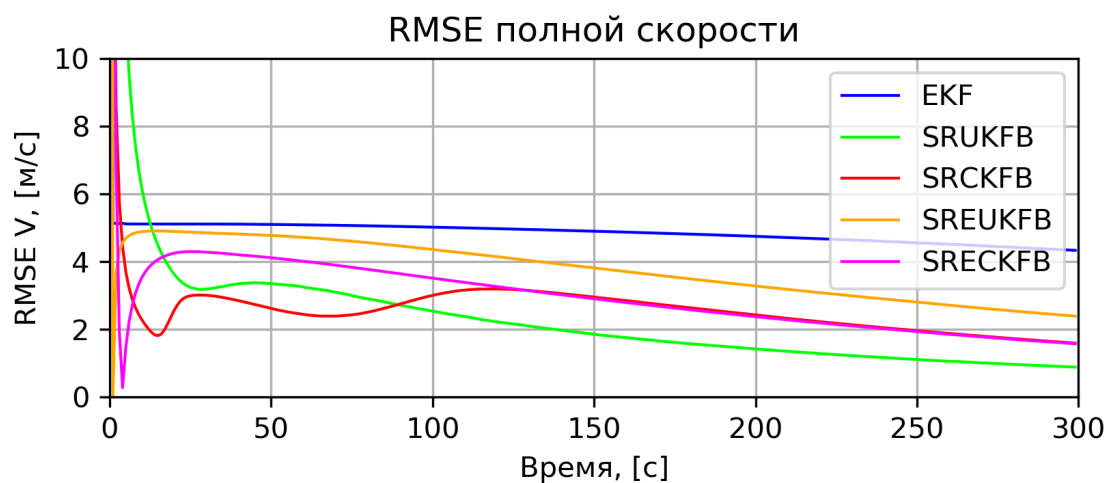


Рис. 1.8 ?>

<? <?

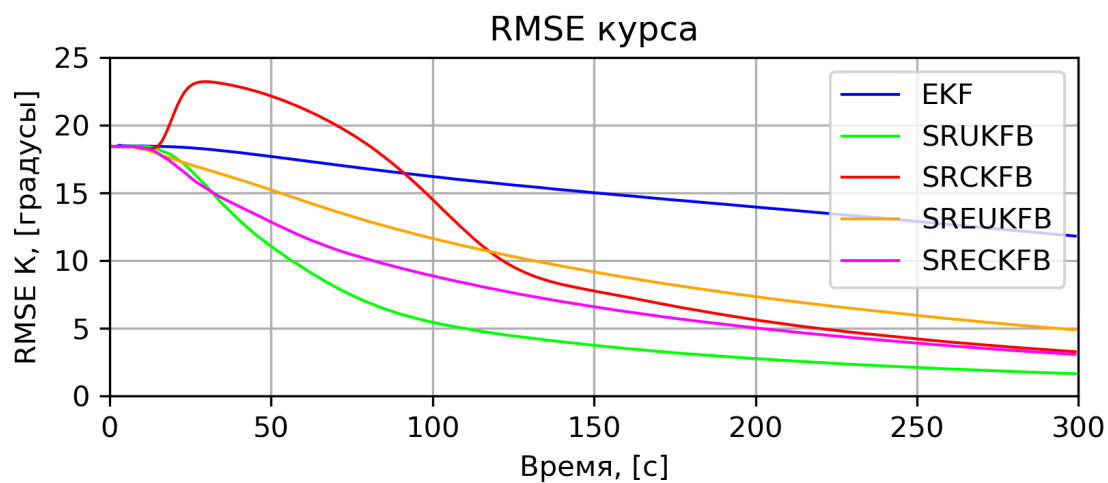


Рис. 1.9 ?>

<? <?

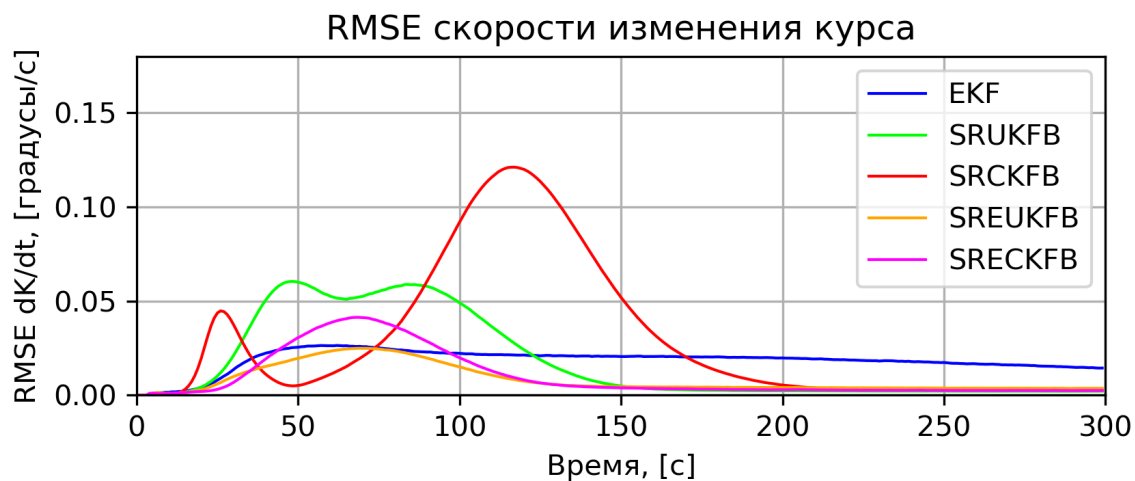


Рис.3 - Время выполнения относительно EKF при JQR разложении в SRUKF фильтрах

<? <?

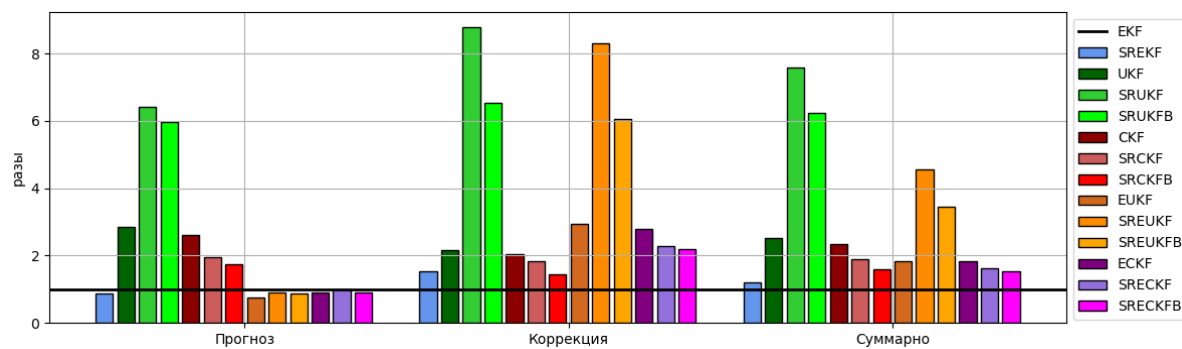
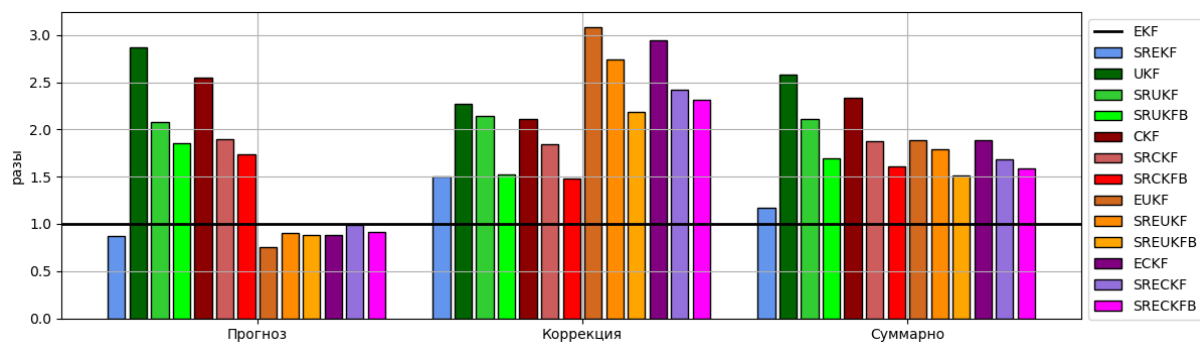


Рис.4 - Время выполнения относительно EKF при QR разложении в SRUKF фильтрах

<? <?



Раздел 2

Алфавитный указатель групп

2.1 Группы

Полный список групп.

`Kalman_filters` [17](#)

Раздел 3

Алфавитный указатель пространств имен

3.1 Пространства имен

Полный список пространств имен.

[KalmanFilters](#)

Фильтры Калмана 19

Раздел 4

Иерархический список классов

4.1 Иерархия классов

Иерархия классов.

| | |
|---|----|
| KalmanFilters::CKalmanLKF< SizeX, SizeY > | 39 |
| KalmanFilters::CKalmanEKF< SizeX, SizeY > | 30 |
| KalmanFilters::CKalmanCKF< SizeX, SizeY > | 21 |
| KalmanFilters::CKalmanECKF< SizeX, SizeY > | 27 |
| KalmanFilters::CKalmanSRCKF< SizeX, SizeY > | 53 |
| KalmanFilters::CKalmanSRCKFB< SizeX, SizeY > | 56 |
| KalmanFilters::CKalmanSRECKF< SizeX, SizeY > | 59 |
| KalmanFilters::CKalmanSRECKFB< SizeX, SizeY > | 61 |
| KalmanFilters::CKalmanSREKF< SizeX, SizeY > | 62 |
| KalmanFilters::CKalmanSRECKF< SizeX, SizeY > | 59 |
| KalmanFilters::CKalmanSREUKF< SizeX, SizeY > | 65 |
| KalmanFilters::CKalmanSREUKFB< SizeX, SizeY > | 67 |
| KalmanFilters::CKalmanUKF< SizeX, SizeY > | 78 |
| KalmanFilters::CKalmanEUKF< SizeX, SizeY > | 35 |
| KalmanFilters::CKalmanSRUKF< SizeX, SizeY > | 69 |
| KalmanFilters::CKalmanSREUKF< SizeX, SizeY > | 65 |
| KalmanFilters::CKalmanSRUKFB< SizeX, SizeY > | 74 |

Раздел 5

Алфавитный указатель классов

5.1 Классы

Классы с их кратким описанием.

| | | |
|---|--|----|
| KalmanFilters::CKalmanCKF< SizeX, SizeY > | Шаблонный класс кубатурного фильтра Калмана, КФК (Cubature Kalman Filter, CKF) | 21 |
| KalmanFilters::CKalmanECKF< SizeX, SizeY > | Шаблонный класс расширенного кубатурного фильтра Калмана, РКФК (Extended Cubature Kalman Filter, ECKF) | 27 |
| KalmanFilters::CKalmanEKF< SizeX, SizeY > | Класс расширенного фильтра Калмана, РФК (Extended Kalman Filter, EKF) . . | 30 |
| KalmanFilters::CKalmanEUKF< SizeX, SizeY > | Шаблонный класс расширенного сигма-точечного (ансцентного) фильтра Калмана, РСТФК (Extended Unscented Kalman Filter, EUKF) | 35 |
| KalmanFilters::CKalmanLKF< SizeX, SizeY > | Шаблонный класс линейного фильтра Калмана, ЛФК (Linear Kalman Filter, LKF) | 39 |
| KalmanFilters::CKalmanSRCKF< SizeX, SizeY > | Шаблонный класс квадратно-корневого кубатурного фильтра Калмана, КК-КФК (Square Root Cubature Kalman Filter, SR-CKF) | 53 |
| KalmanFilters::CKalmanSRCKFB< SizeX, SizeY > | Шаблонный класс квадратно-корневого кубатурного фильтра Калмана (блочная фильтрация), КК-КФКБ (Square Root Cubature Kalman Filter Block, SR-CKFB) . | 56 |
| KalmanFilters::CKalmanSRECKF< SizeX, SizeY > | Шаблонный класс квадратно-корневого расширенного кубатурного фильтра Калмана, КК-РКФК (Square Root Extended Cubature Kalman Filter, SR-ECKF) . . . | 59 |
| KalmanFilters::CKalmanSRECKFB< SizeX, SizeY > | Шаблонный класс квадратно-корневого расширенного кубатурного фильтра Калмана (блочная фильтрация), КК-РКФКБ (Square Root Extended Cubature Kalman Filter Block, SR-ECKFB) | 61 |
| KalmanFilters::CKalmanSREKF< SizeX, SizeY > | Класс квадратно-корневого расширенного фильтра Калмана, КК-РФК (Square Root Extended Kalman Filter, EKF) | 62 |
| KalmanFilters::CKalmanSREUKF< SizeX, SizeY > | Шаблонный класс квадратно-корневого расширенного сигма-точечного (ансцентного) фильтра Калмана, КК-РСТФК (Square Root Extended Unscented Kalman Filter, SR-EUKF) | 65 |
| KalmanFilters::CKalmanSREUKFB< SizeX, SizeY > | Шаблонный класс квадратно-корневого расширенного сигма-точечного (ансцентного) фильтра Калмана (блочная фильтрация), КК-РСТФК (Square Root Extended Unscented Kalman Filter Block, SR-EUKFB) | 67 |
| KalmanFilters::CKalmanSRUKF< SizeX, SizeY > | Шаблонный класс квадратно-корневого сигма-точечного (ансцентного) фильтра Калмана, КК-СТФК (КК-АФК) (Square Root Unscented Kalman Filter, SR-UKF) | 69 |

[KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >](#)

Шаблонный класс квадратно-корневого сигма-точечного (ансцентного) фильтра Калмана (блочная фильтрация), КК-СТФКБ (КК-АФКБ) (Square Root Unscented Kalman Filter Block, SR-UKFB) 74

[KalmanFilters::CKalmanUKF< SizeX, SizeY >](#)

Шаблонный класс сигма-точечного (ансцентного) фильтра Калмана, СТФК (АФК) (Unscented Kalman Filter, UKF) 78

Раздел 6

Группы

6.1 Kalman_filters

Пространства имен

- namespace [KalmanFilters](#)
Фильтры Калмана

6.1.1 Подробное описание

Раздел 7

Пространства имен

7.1 Пространство имен KalmanFilters

Фильтры Калмана

Классы

- class [CKalmanCKF](#)
Шаблонный класс кубатурного фильтра Калмана, КФК (Cubature Kalman Filter, CKF)
- class [CKalmanECKF](#)
Шаблонный класс расширенного кубатурного фильтра Калмана, РКФК (Extended Cubature Kalman Filter, ECKF)
- class [CKalmanEKF](#)
Класс расширенного фильтра Калмана, РКФ (Extended Kalman Filter, EKF)
- class [CKalmanEUKF](#)
Шаблонный класс расширенного сигма-точечного (ансцентного) фильтра Калмана, РСТФК (Extended Unscented Kalman Filter, EUKF)
- class [CKalmanLKF](#)
Шаблонный класс линейного фильтра Калмана, ЛФК (Linear Kalman Filter, LKF)
- class [CKalmanSRCKF](#)
Шаблонный класс квадратно-корневого кубатурного фильтра Калмана, КК-КФК (Square Root Cubature Kalman Filter, SR-CKF)
- class [CKalmanSRCKFB](#)
Шаблонный класс квадратно-корневого кубатурного фильтра Калмана (блочная фильтрация), КК-КФКБ (Square Root Cubature Kalman Filter Block, SR-CKFB)
- class [CKalmanSRECKF](#)
Шаблонный класс квадратно-корневого расширенного кубатурного фильтра Калмана, КК-РКФК (Square Root Extended Cubature Kalman Filter, SR-ECKF)
- class [CKalmanSRECKFB](#)
Шаблонный класс квадратно-корневого расширенного кубатурного фильтра Калмана (блочная фильтрация), КК-РКФКБ (Square Root Extended Cubature Kalman Filter Block, SR-ECKFB)
- class [CKalmanSREKF](#)
Класс квадратно-корневого расширенного фильтра Калмана, КК-РФК (Square Root Extended Kalman Filter, EKF)
- class [CKalmanSREUKF](#)
Шаблонный класс квадратно-корневого расширенного сигма-точечного (ансцентного) фильтра Калмана, КК-РСТФК (Square Root Extended Unscented Kalman Filter, SR-EUKF)
- class [CKalmanSREUKFB](#)
Шаблонный класс квадратно-корневого расширенного сигма-точечного (ансцентного) фильтра Калмана (блочная фильтрация), КК-РСТФК (Square Root Extended Unscented Kalman Filter Block, SR-EUKFB)

- class [CKalmanSRUKF](#)

Шаблонный класс квадратно-корневого сигма-точечного (ансцентного) фильтра Калмана, КК-СТФК (КК-АФК) (Square Root Unscented Kalman Filter, SR-UKF)

- class [CKalmanSRUKFB](#)

Шаблонный класс квадратно-корневого сигма-точечного (ансцентного) фильтра Калмана (блочная фильтрация), КК-СТФКБ (КК-АФКБ) (Square Root Unscented Kalman Filter Block, SR-UKFB)

- class [CKalmanUKF](#)

Шаблонный класс сигма-точечного (ансцентного) фильтра Калмана, СТФК (АФК) (Unscented Kalman Filter, UKF)

Функции

- std::string [GetVersion](#) ()

Возвращает строку, содержащую информацию о версии

7.1.1 Подробное описание

Фильтры Калмана

7.1.2 Функции

7.1.2.1 GetVersion()

std::string KalmanFilters::GetVersion ()

Возвращает строку, содержащую информацию о версии

Возвращает

Строка версии в формате DD-MM-YY-VV_COMMENTS, где DD - день, MM - месяц, YY - год, VV - версия, COMMENTS - комментарий(опционально)

Раздел 8

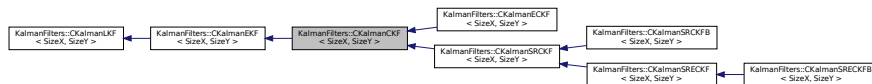
Классы

8.1 Шаблон класса KalmanFilters::CKalmanCKF< SizeX, SizeY >

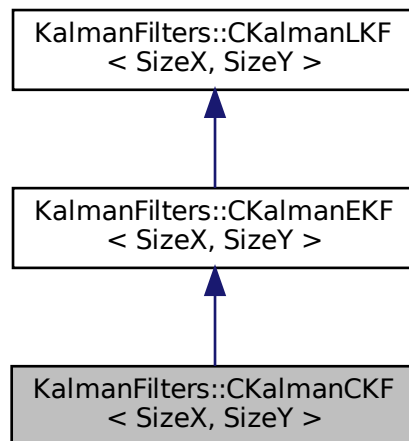
Шаблонный класс кубатурного фильтра Калмана, КФК (Cubature Kalman Filter, CKF)

#include <kalman_filter_cubature.h>

Граф наследования: KalmanFilters::CKalmanCKF< SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanCKF< SizeX, SizeY >:



Открытые члены

- [CKalmanCKF](#) ()

Конструктор по умолчанию

- void [SetWeightedSumStateSigmas](#) (std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> weightedSumStateSigmas)

- Установка функции вычисления взвешенной суммы сигма-точек пространства X
- void [SetWeightedSumMeasurementSigmas](#) (std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> weightedSumMeasurementSigmas)
Установка функции вычисления взвешенной суммы сигма-точек пространства Y.
- virtual void [Prediction](#) (double dt)
Прогноз
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция

Защищенные члены

- void [PredictionCKF](#) (double dt)
Прогноз CKF.
- void [CorrectionCKF](#) (const arma::vec &Y_msd)
Коррекция CKF.
- void [SetupDesignParametersCubatureBaseSet](#) ()
Установка кубатурных весов (базовый вариант ансцентного преобразования)
- void [SetStateTransitionJacobianF](#) ()
Запрет доступа
- void [SetObservationJacobianH](#) ()
Запрет доступа

Защищенные данные

- int [k_sigma_points_](#)
Число сигма-точек
- arma::vec [weights_mean_](#)
Веса среднего
- arma::vec [weights_covariance_](#)
Веса ковариации
- arma::mat [x_est_sigma_points_](#)
Матрица сигма-точек (сигма-точки - столбцы) в пространстве X на текущем такте, размерность [SizeX, k_sigma_points_].
- arma::mat [x_pred_sigma_points_](#)
Матрица сигма-точек (сигма-точки - столбцы) в пространстве X, экстраполированный на текущий такт, размерность [SizeX, k_sigma_points_].
- arma::mat [y_pred_sigma_points_](#)
Матрица сигма-точек (сигма-точки - столбцы) в пространстве Y, экстраполированный на текущий такт, размерность [SizeX, k_sigma_points_].
- arma::mat [dXcal_](#)
Матрица X-каллиграфическое (матрица сигма-точек - столбцов)
- arma::mat [dYcal_](#)
Матрица Y-каллиграфическое (матрица сигма-точек - столбцов)
- arma::mat [P_xy_](#)
Матрица кросс-ковариации векторов X и Y, размерность [SizeX * SizeY].
- arma::mat [sqrt_P_chol_](#)
Корень из матрицы P.
- double [gamma_](#)
Автоматически вычисляемый (в методах SetDesignParameters*) параметр (множитель при корне из P при создании сигма-точек)
- std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> [weightedSumStateSigmas_](#)
Вычисление взвешенной суммы сигма-точек пространства X
- std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> [weightedSumMeasurementSigmas_](#)
Вычисление взвешенной суммы сигма-точек пространства Y.

8.1.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanCKF< SizeX, SizeY >
```

Шаблонный класс кубатурного фильтра Калмана, КФК (Cubature Kalman Filter, CKF)

Частный случай сигма-точечного фильтра Калмана при параметрах разброса сигма-точек выбранных по рекомендации Merwe alpha=1.0, beta=0.0, kappa=0.0.

Источники:

[1] Cubature Kalman Filters, Ienkar Arasaratnam and Simon Haykin, Life Fellow, IEEE

[2] Sebastian Bitzer, Technische Universität Dresden, <https://github.com/sbitzer/UKF-exposed/blob/master/←UKF.pdf>

Внимание

Фильтр построен по классическому его варианту НИЖНЕтреугольного разложения Холецкого!

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.1.2 Конструктор(ы)

8.1.2.1 CKalmanCKF()

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanCKF< SizeX, SizeY >::CKalmanCKF ( ) [inline]
Конструктор по умолчанию
```

8.1.3 Методы

8.1.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanCKF< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanEKF< SizeX, SizeY >](#).

Переопределяется в [KalmanFilters::CKalmanSRCKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >](#), [KalmanFilters::CKalmanECKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRECKF< SizeX, SizeY >](#) и [KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >](#).

8.1.3.2 CorrectionCKF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanCKF< SizeX, SizeY >::CorrectionCKF (
    const arma::vec & Y_msd ) [inline], [protected]
```

Коррекция СКФ.

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

8.1.3.3 Prediction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanCKF< SizeX, SizeY >::Prediction (
    double dt ) [inline], [virtual]
```

Прогноз

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

Переопределяет метод предка [KalmanFilters::CKalmanEKF< SizeX, SizeY >](#).

Переопределяется в [KalmanFilters::CKalmanSRCKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanECKF< SizeX, SizeY >](#) и [KalmanFilters::CKalmanSRECKF< SizeX, SizeY >](#).

8.1.3.4 PredictionCKF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanCKF< SizeX, SizeY >::PredictionCKF (
    double dt ) [inline], [protected]
```

Прогноз СКФ.

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

8.1.3.5 SetObservationJacobianH()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanCKF< SizeX, SizeY >::SetObservationJacobianH ( ) [inline], [protected]
```

Запрет доступа

8.1.3.6 SetStateTransitionJacobianF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanCKF< SizeX, SizeY >::SetStateTransitionJacobianF ( ) [inline], [protected]
```

Запрет доступа

8.1.3.7 SetupDesignParametersCubatureBaseSet()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanCKF< SizeX, SizeY >::SetupDesignParametersCubatureBaseSet ( ) [inline], [protected]
```

Установка кубатурных весов (базовый вариант ансцентного преобразования)

Смотри [1] и [2]

8.1.3.8 SetWeightedSumMeasurementSigmas()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanCKF< SizeX, SizeY >::SetWeightedSumMeasurementSigmas (
    std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> weightedSum←
    MeasurementSigmas ) [inline]
```

Установка функции вычисления взвешенной суммы сигма-точек пространства Y.

Аргументы

| | |
|------------------------------|--|
| weightedSumMeasurementSigmas | - Функция вычисления взвешенной суммы сигма-точек пространства Y |
|------------------------------|--|

8.1.3.9 SetWeightedSumStateSigmas()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanCKF< SizeX, SizeY >::SetWeightedSumStateSigmas (
    std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> weightedSumState←
    Sigmas ) [inline]
```

Установка функции вычисления взвешенной суммы сигма-точек пространства X

Аргументы

| | |
|------------------------|--|
| weightedSumStateSigmas | - Функция вычисления взвешенной суммы сигма-точек пространства X |
|------------------------|--|

8.1.4 Данные класса

8.1.4.1 dXcal_

```
template<size_t SizeX, size_t SizeY>
arma::mat KalmanFilters::CKalmanCKF< SizeX, SizeY >::dXcal_ [protected]
```

Матрица X-каллиграфическое (матрица сигма-точек - столбцов)

8.1.4.2 dYcal_

```
template<size_t SizeX, size_t SizeY>
arma::mat KalmanFilters::CKalmanCKF< SizeX, SizeY >::dYcal_ [protected]
```

Матрица Y-каллиграфическое (матрица сигма-точек - столбцов)

8.1.4.3 gamma_

```
template<size_t SizeX, size_t SizeY>
double KalmanFilters::CKalmanCKF< SizeX, SizeY >::gamma_ [protected]
```

Автоматически вычисляемый (в методах SetDesignParameters*) параметр (множитель при корне из Р при создании сигма-точек)

8.1.4.4 k_sigma_points_

```
template<size_t SizeX, size_t SizeY>
int KalmanFilters::CKalmanCKF< SizeX, SizeY >::k_sigma_points_ [protected]
```

Число сигма-точек

8.1.4.5 P_xy_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanCKF< SizeX, SizeY >::P_xy_ [protected]
```

Матрица кросс-коварации векторов X и Y, размерность [SizeX * SizeY].

8.1.4.6 sqrt_P_chol_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanCKF< SizeX, SizeY >::sqrt_P_chol_ [protected]
```

Корень из матрицы P.

8.1.4.7 weightedSumMeasurementSigmas_

```
template<size_t SizeX, size_t SizeY>
```

```
std::function<arma::vec( const arma::vec &weights, const arma::mat &sigmaPoints )> KalmanFilters::CKalmanCKF<  
SizeX, SizeY >::weightedSumMeasurementSigmas_ [protected]
```

Вычисление взвешенной суммы сигма-точек пространства Y.

8.1.4.8 weightedSumStateSigmas_

```
template<size_t SizeX, size_t SizeY>
```

```
std::function<arma::vec( const arma::vec &weights, const arma::mat &sigmaPoints )> KalmanFilters::CKalmanCKF<  
SizeX, SizeY >::weightedSumStateSigmas_ [protected]
```

Вычисление взвешенной суммы сигма-точек пространства X

8.1.4.9 weights_covariance_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::vec KalmanFilters::CKalmanCKF< SizeX, SizeY >::weights_covariance_ [protected]
```

Веса ковариации

8.1.4.10 weights_mean_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::vec KalmanFilters::CKalmanCKF< SizeX, SizeY >::weights_mean_ [protected]
```

Веса среднего

8.1.4.11 x_est_sigma_points_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanCKF< SizeX, SizeY >::x_est_sigma_points_ [protected]
```

Матрица сигма-точек (сигма-точки - столбцы) в пространстве X на текущем такте, размерность [SizeX,k_sigma_points_].

8.1.4.12 x_pred_sigma_points_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanCKF< SizeX, SizeY >::x_pred_sigma_points_ [protected]
```

Матрица сигма-точек (сигма-точки - столбцы) в пространстве X, экстраполированный на текущий такт, размерность [SizeX,k_sigma_points_].

8.1.4.13 y_pred_sigma_points_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanCKF< SizeX, SizeY >::y_pred_sigma_points_ [protected]
```

Матрица сигма-точек (сигма-точки - столбцы) в пространстве Y, экстраполированный на текущий такт, размерность [SizeX,k_sigma_points_].

Объявления и описания членов класса находятся в файле:

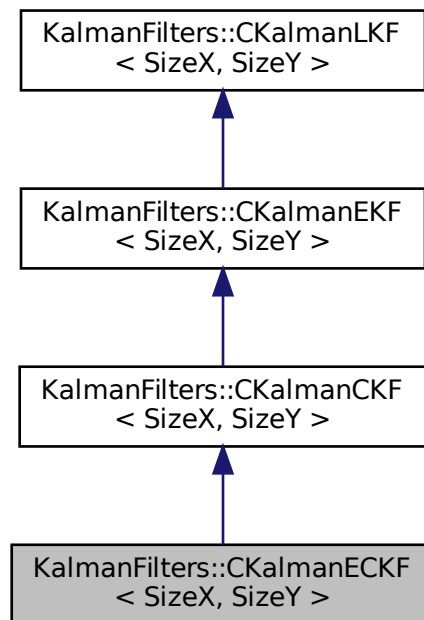
- kalman_filter_cubature.h

8.2 Шаблон класса KalmanFilters::CKalmanECKF< SizeX, SizeY >

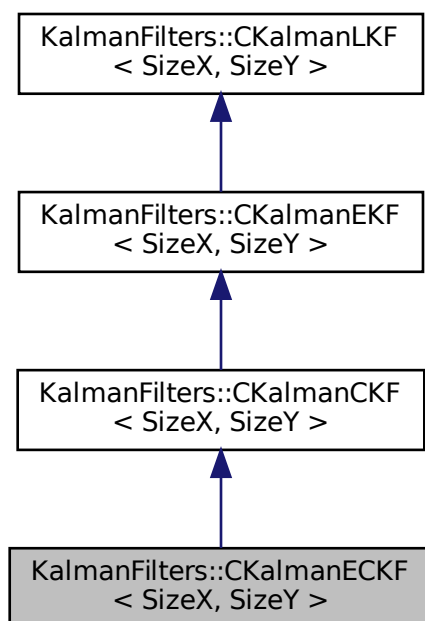
Шаблонный класс расширенного кубатурного фильтра Калмана, РКФК (Extended Cubature Kalman Filter, ECKF)

```
#include <kalman_filter_extended_cubature.h>
```

Граф наследования: KalmanFilters::CKalmanECKF< SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanECKF < SizeX, SizeY >:



Открытые члены

- [CKalmanECKF](#) ()
Конструктор по умолчанию
- void [SetStateTransitionJacobianF](#) (std::function< arma::mat(const arma::vec &X, double dt)> stateTransitionJacobianF)
Установка функции вычисления матрицы перехода состояния F (makeMatrixF)
- void [SetObservationJacobianH](#) (std::function< arma::mat(const arma::vec &X)> observationJacobianH)
Установка функции вычисления матрицы перехода измерений H (makeMatrixH)
- virtual void [Prediction](#) (double dt)
Прогноз расширенного кубатурного фильтра Калмана (РКФК, ЕСКФ)
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция расширенного кубатурного фильтра Калмана (РКФК, ЕСКФ)

Дополнительные унаследованные члены

8.2.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanECKF < SizeX, SizeY >
```

Шаблонный класс расширенного кубатурного фильтра Калмана, РКФК (Extended Cubature Kalman Filter, ECKF)

Источники:

[1] Cubature Kalman Filters, Ienkar Arasaratnam and Simon Haykin, Life Fellow, IEEE

Внимание

Фильтр построен по классическому его варианту НИЖНЕтреугольного разложения Холецкого!

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.2.2 Конструктор(ы)

8.2.2.1 CKalmanECKF()

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanECKF< SizeX, SizeY >::CKalmanECKF ( ) [inline]
Конструктор по умолчанию
```

8.2.3 Методы

8.2.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanECKF< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
Коррекция расширенного кубатурного фильтра Калмана (РКФК, ЕСКФ)
```

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanCKF< SizeX, SizeY >](#).

8.2.3.2 Prediction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanECKF< SizeX, SizeY >::Prediction (
    double dt ) [inline], [virtual]
Прогноз расширенного кубатурного фильтра Калмана (РКФК, ЕСКФ)
```

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

Переопределяет метод предка [KalmanFilters::CKalmanCKF< SizeX, SizeY >](#).

8.2.3.3 SetObservationJacobianH()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanECKF< SizeX, SizeY >::SetObservationJacobianH (
    std::function< arma::mat(const arma::vec &X)> observationJacobianH ) [inline]
```

Установка функции вычисления матрицы перехода измерений H (makeMatrixH)

См. также

[observationJacobianH_](#)

8.2.3.4 SetStateTransitionJacobianF()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanECKF< SizeX, SizeY >::SetStateTransitionJacobianF (
    std::function< arma::mat(const arma::vec &X, double dt)> stateTransitionJacobianF ) [inline]
```

Установка функции вычисления матрицы перехода состояния F (makeMatrixF)

См. также

[stateTransitionJacobianF_](#)

Объявления и описания членов класса находятся в файле:

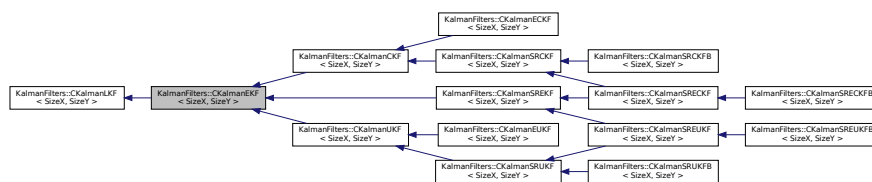
- kalman_filter_extended_cubature.h

8.3 Шаблон класса KalmanFilters::CKalmanEKF< SizeX, SizeY >

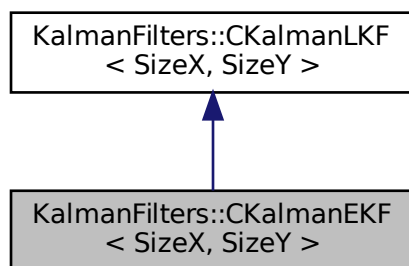
Класс расширенного фильтра Калмана, РФК (Extended Kalman Filter, EKF)

```
#include <kalman_filter_extended.h>
```

Граф наследования: KalmanFilters::CKalmanEKF< SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanEKF< SizeX, SizeY >:



Открытые члены

- [CKalmanEKF](#) ()

Конструктор по умолчанию

- virtual [~CKalmanEKF](#) ()=default
- void [SetStateTransitionModel](#) (std::function< arma::vec(const arma::vec &X, double dt)> state←TransitionModel)
Установка функции прогноза состояния (predictState)
- void [SetObservationModel](#) (std::function< arma::vec(const arma::vec &X)> observationModel)
Установка функции перевода состояния в измерение (XtoY)
- void [SetStateTransitionJacobianF](#) (std::function< arma::mat(const arma::vec &X, double dt)> stateTransitionJacobianF)
Установка функции вычисления матрицы перехода состояния F (makeMatrixF)
- void [SetObservationJacobianH](#) (std::function< arma::mat(const arma::vec &X)> observation←JacobianH)
Установка функции вычисления матрицы перехода измерений H (makeMatrixH)
- virtual void [Prediction](#) (double dt)
Прогноз
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция

Защищенные члены

- void [PredictionEKF](#) (double dt)
Прогноз EKF.

Защищенные данные

- std::function< arma::vec(const arma::vec &X, double dt)> [stateTransitionModel_](#)
Функция прогноза состояния (predictState)
- std::function< arma::vec(const arma::vec &X)> [observationModel_](#)
Функция перевода состояния в измерения (XtoY)
- std::function< arma::mat(const arma::vec &X, double dt)> [stateTransitionJacobianF_](#)
Функция вычисления матрицы перехода состояния F (makeMatrixF)
- std::function< arma::mat(const arma::vec &X)> [observationJacobianH_](#)
Функция вычисления матрицы перехода измерений H (makeMatrixH)

Закрытые члены

- void [SetStateTransitionJacobianLinearF](#) ()
- void [SetStateTransitionMatrixF](#) ()
Запрет доступа
- void [SetObservationMatrixH](#) ()
Запрет доступа

8.3.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanEKF< SizeX, SizeY >
```

Класс расширенного фильтра Калмана, РФК (Extended Kalman Filter, EKF)

Источник: NASA Technical report R-135, Application of statistical filter theory to the optimal estimation of position and velocity on board a circumlunar vehicle, Gerald L. Smith, Stanley F. Schmidt and Leonard A. McGee, 1962

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.3.2 Конструктор(ы)

8.3.2.1 CKalmanEKF()

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanEKF< SizeX, SizeY >::CKalmanEKF ( ) [inline]
Конструктор по умолчанию
```

8.3.2.2 ~CKalmanEKF()

```
template<size_t SizeX, size_t SizeY>
virtual KalmanFilters::CKalmanEKF< SizeX, SizeY >::~~CKalmanEKF ( ) [virtual], [default]
```

8.3.3 Методы

8.3.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanEKF< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanLKF< SizeX, SizeY >](#).

Переопределяется в [KalmanFilters::CKalmanCKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRCKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >](#), [KalmanFilters::CKalmanECKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRECKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanEUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >](#), [KalmanFilters::CKalmanUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRUKF< SizeX, SizeY >](#) и [KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >](#).

8.3.3.2 Prediction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanEKF< SizeX, SizeY >::Prediction (
    double dt ) [inline], [virtual]
```

Прогноз

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

Переопределяет метод предка [KalmanFilters::CKalmanLKF< SizeX, SizeY >](#).

Переопределяется в [KalmanFilters::CKalmanCKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRCKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanECKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRECKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanEUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanUKF< SizeX, SizeY >](#) и [KalmanFilters::CKalmanSRUKF< SizeX, SizeY >](#).

8.3.3.3 PredictionEKF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanEKF< SizeX, SizeY >::PredictionEKF (
    double dt ) [inline], [protected]
```

Прогноз EKF.

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

8.3.3.4 SetObservationJacobianH()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanEKF< SizeX, SizeY >::SetObservationJacobianH (
    std::function< arma::mat(const arma::vec &X)> observationJacobianH ) [inline]
```

Установка функции вычисления матрицы перехода измерений H (makeMatrixH)

См. также

[observationJacobianH_](#)

8.3.3.5 SetObservationMatrixH()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanEKF< SizeX, SizeY >::SetObservationMatrixH ( ) [inline], [private]
```

Запрет доступа

8.3.3.6 SetObservationModel()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanEKF< SizeX, SizeY >::SetObservationModel (
    std::function< arma::vec(const arma::vec &X)> observationModel ) [inline]
```

Установка функции перевода состояния в измерение (XtoY)

См. также

[observationModel_](#)

8.3.3.7 SetStateTransitionJacobianF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanEKF< SizeX, SizeY >::SetStateTransitionJacobianF (
    std::function< arma::mat(const arma::vec &X, double dt)> stateTransitionJacobianF ) [inline]
```

Установка функции вычисления матрицы перехода состояния F (makeMatrixF)

См. также

[stateTransitionJacobianF_](#)

8.3.3.8 SetStateTransitionJacobianLinearF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanEKF< SizeX, SizeY >::SetStateTransitionJacobianLinearF ( ) [inline], [private]
```

8.3.3.9 SetStateTransitionMatrixF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanEKF< SizeX, SizeY >::SetStateTransitionMatrixF ( ) [inline], [private]
Запрет доступа
```

8.3.3.10 SetStateTransitionModel()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanEKF< SizeX, SizeY >::SetStateTransitionModel (
    std::function< arma::vec(const arma::vec &X, double dt)> stateTransitionModel ) [inline]
Установка функции прогноза состояния (predictState)
```

См. также

[stateTransitionModel_](#)

8.3.4 Данные класса

8.3.4.1 observationJacobianH_

```
template<size_t SizeX, size_t SizeY>
std::function<arma::mat( const arma::vec &X )> KalmanFilters::CKalmanEKF< SizeX, SizeY >::observationJacobian↔
H_ [protected]
Функция вычисления матрицы перехода измерений H (makeMatrixH)
```

Аргументы

| | |
|---|---|
| X | - вектор состояния текущего момента времени |
|---|---|

См. также

[SetObservationJacobianH](#)

8.3.4.2 observationModel_

```
template<size_t SizeX, size_t SizeY>
std::function<arma::vec( const arma::vec &X )> KalmanFilters::CKalmanEKF< SizeX, SizeY >::observationModel_↔
[protected]
Функция перевода состояния в измерения (XtoY)
```

Аргументы

| | |
|---|---|
| X | - вектор состояния текущего момента времени |
|---|---|

См. также

[SetObservationModel](#)

8.3.4.3 stateTransitionJacobianF_

```
template<size_t SizeX, size_t SizeY>
std::function<arma::mat( const arma::vec &X, double dt )> KalmanFilters::CKalmanEKF< SizeX, SizeY >::state↔
TransitionJacobianF_ [protected]
```


Функция вычисления матрицы перехода состояния F (makeMatrixF)

Аргументы

| | |
|---|---|
| X | - вектор состояния с прошлого момента времени |
|---|---|

См. также

[SetStateTransitionJacobianF](#)

8.3.4.4 stateTransitionModel_

```
template<size_t SizeX, size_t SizeY>
std::function<arma::vec( const arma::vec &X, double dt )> KalmanFilters::CKalmanEKF< SizeX, SizeY >::state↔
TransitionModel_ [protected]
```

Функция прогноза состояния (predictState)

Аргументы

| | |
|---|---|
| X | - вектор состояния с прошлого момента времени |
|---|---|

См. также

[SetStateTransitionModel](#)

Объявления и описания членов класса находятся в файле:

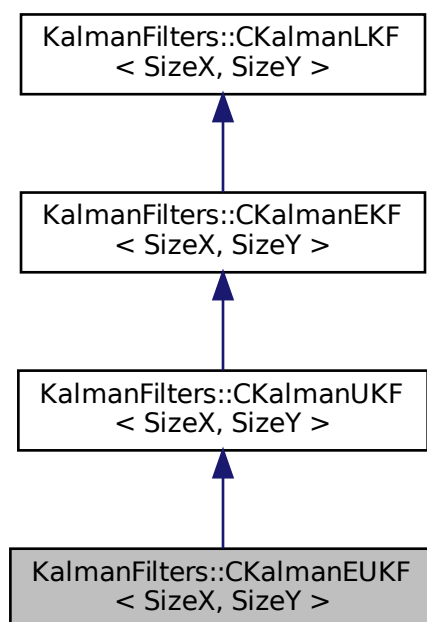
- kalman_filter_extended.h

8.4 Шаблон класса KalmanFilters::CKalmanEUKF< SizeX, SizeY >

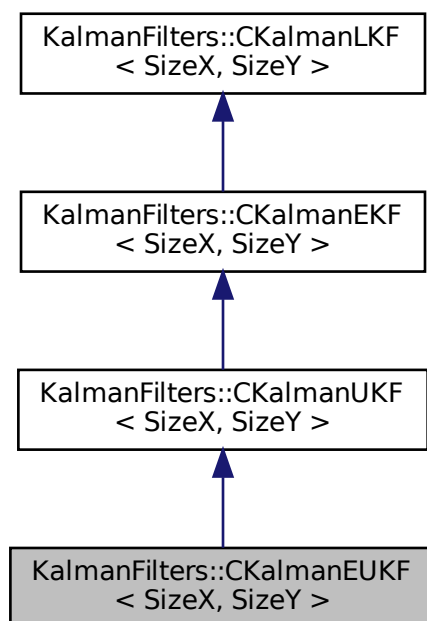
Шаблонный класс расширенного сигма-точечного (ансцентного) фильтра Калмана, РСТФК (Extended Unscented Kalman Filter, EUKF)

```
#include <kalman_filter_extended_unscented.h>
```

Граф наследования: KalmanFilters::CKalmanEUKF < SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanEUKF < SizeX, SizeY >:



Открытые члены

- [CKalmanEUKF](#) ()
Конструктор по умолчанию
- void [SetStateTransitionJacobianF](#) (std::function< arma::mat(const arma::vec &X, double dt)> stateTransitionJacobianF)
Установка функции вычисления матрицы перехода состояния F (makeMatrixF)
- void [SetObservationJacobianH](#) (std::function< arma::mat(const arma::vec &X)> observationJacobianH)
Установка функции вычисления матрицы перехода измерений H (makeMatrixH)
- virtual void [Prediction](#) (double dt)
Прогноз расширенного сигма-точечного фильтра Калмана (РСТФК, EUKF)
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция расширенного сигма-точечного фильтра Калмана (РСТФК, EUKF)

Дополнительные унаследованные члены

8.4.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanEUKF< SizeX, SizeY >
```

Шаблонный класс расширенного сигма-точечного (ансцентного) фильтра Калмана, РСТФК (Extended Unscented Kalman Filter, EUKF)

Источники:

- [1] A New Extension of the Kalman Filter to Nonlinear Systems, Simon J. Julier, Jeffrey K. Uhlmann, The Robotics Research Group, Department of Engineering Science, The University of Oxford, 1997
- [2] The Unscented Kalman Filter for Nonlinear Estimation, Eric A. Wan and Rudolph van der Merwe Oregon Graduate Institute of Science & Technology 20000 NW Walker Rd, Beaverton, Oregon 97006, 2000 ericwan[at]ece.ogi.edu, rvdmerwe[at]ece.ogi.edu
- [3] Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models, Rudolph van der Merwe & Eric Wan, OGI School of Science & Engineering Oregon Health & Science University Beaverton, Oregon, 97006, USA, 2003 {rvdmerwe,ericwan}[at]ece.ogi.edu
- [4] THE SQUARE-ROOT UNSCENTED KALMAN FILTER FOR STATE AND PARAMETER-ESTIMATION, Rudolph van der Merwe and Eric A. Wan, Oregon Graduate Institute of Science and Technology 20000 NW Walker Road, Beaverton, Oregon 97006, USA rvdmerwe,ericwan[at]ece.ogi.edu

Внимание

Внимание! В реализации UKF вес нулевой сигма-точки W_{cov} не может быть отрицательным! (W_{mean} - может)

Фильтр построен по классическому его варианту НИЖНЕтреугольного разложения Холецкого!

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.4.2 Конструктор(ы)

8.4.2.1 CKalmanEUKF()

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanEUKF< SizeX, SizeY >::CKalmanEUKF ( ) [inline]
```

Конструктор по умолчанию

8.4.3 Методы

8.4.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanEUKF< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция расширенного сигма-точечного фильтра Калмана (РСТФК, EUKF)

Аргументы

| | |
|--------------------|---|
| <code>Y_msd</code> | - вектор измерений, по которым производится коррекция |
|--------------------|---|

Переопределяет метод предка [KalmanFilters::CKalmanUKF](#)< SizeX, SizeY >.

8.4.3.2 Prediction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanEUKF< SizeX, SizeY >::Prediction (
    double dt ) [inline], [virtual]
```

Прогноз расширенного сигма-точечного фильтра Калмана (РСТФК, EUKF)

Аргументы

| | |
|-----------------|-----------------------|
| <code>dt</code> | - Время прогноза, [с] |
|-----------------|-----------------------|

Переопределяет метод предка [KalmanFilters::CKalmanUKF](#)< SizeX, SizeY >.

8.4.3.3 SetObservationJacobianH()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanEUKF< SizeX, SizeY >::SetObservationJacobianH (
    std::function< arma::mat(const arma::vec &X)> observationJacobianH ) [inline]
```

Установка функции вычисления матрицы перехода измерений H (makeMatrixH)

См. также

[observationJacobianH_](#)

8.4.3.4 SetStateTransitionJacobianF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanEUKF< SizeX, SizeY >::SetStateTransitionJacobianF (
    std::function< arma::mat(const arma::vec &X, double dt)> stateTransitionJacobianF ) [inline]
```

Установка функции вычисления матрицы перехода состояния F (makeMatrixF)

См. также

[stateTransitionJacobianF_](#)

Объявления и описания членов класса находятся в файле:

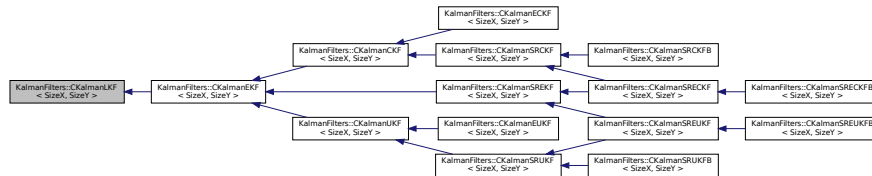
- `kalman_filter_extended_unscented.h`

8.5 Шаблон класса KalmanFilters::CKalmanLKF< SizeX, SizeY >

Шаблонный класс линейного фильтра Калмана, ЛФК (Linear Kalman Filter, LKF)

```
#include <kalman_filter_linear.h>
```

Граф наследования:KalmanFilters::CKalmanLKF< SizeX, SizeY >:



Открытые члены

- [CKalmanLKF](#) ()
Конструктор по умолчанию
- [CKalmanLKF](#) (const [CKalmanLKF](#) &)=default
- [CKalmanLKF](#) & operator= (const [CKalmanLKF](#) &)=default
- [CKalmanLKF](#) ([CKalmanLKF](#) &&)=default
- [CKalmanLKF](#) & operator= ([CKalmanLKF](#) &&)=default
- virtual ~[CKalmanLKF](#) ()=default
- void [SetStateTransitionMatrixF](#) (const arma::mat &F)
Установка матрицы перехода состояния F.
- void [SetObservationMatrixH](#) (const arma::mat &H)
Установка матрицы перехода измерений H.
- void [SetEstimateCovarianceMatrixP](#) (const arma::mat &P)
Установка ковариационной матрицы P состояния X.
- void [SetEstimateCovarianceMatrixPdiag](#) (const arma::vec &Pdiag)
Установка диагонали ковариационной матрицы P состояния X.
- void [SetProcessCovarianceMatrixQ](#) (const arma::mat &Q)
Установка ковариационной матрицы Q шумов состояния X.
- void [SetProcessCovarianceMatrixQdiag](#) (const arma::vec &Qdiag)
Установка диагонали ковариационной матрицы Q шумов состояния X.
- void [SetObservationCovarianceMatrixR](#) (const arma::mat &R)
Установка ковариационной матрицы R шумов измерений Y.
- void [SetObservationCovarianceMatrixRdiag](#) (const arma::vec &Rdiag)
Установка ковариационной матрицы R шумов измерений Y.
- void [SetEstimatedVectorX](#) (const arma::vec &X_est)
Установка оценки вектора состояния X.
- void [SetEstimatedVectorY](#) (const arma::vec &Y_est)
Установка оценки вектора состояния Y.
- void [SetMeasuredVectorY](#) (const arma::vec &Y_msd)
Установка измеренного вектора измерений Y.
- void [SetDeltaY](#) (const arma::vec &DeltaY)
Установка вектора невязки измерений DeltaY.
- void [SetCheckBordersStateAfterPrediction](#) (std::function< arma::vec(const arma::vec &X)> checkBordersStateAfterPrediction)
Установка функции проверки вектора состояния X после прогноза
- void [SetCheckBordersStateAfterCorrection](#) (std::function< arma::vec(const arma::vec &X)> checkBordersStateAfterCorrection)

- Установка функции проверки вектора состояния X после коррекции
- void [SetCheckBordersMeasurement](#) (std::function< arma::vec(const arma::vec &Y)> check← BordersMeasurement)
- Установка функции проверки вектора измерений Y.
- void [SetCheckDeltaState](#) (std::function< arma::vec(const arma::vec &DeltaX)> checkDeltaState)
- Установка функции проверки разности векторов состояний X.
- void [SetCheckDeltaMeasurement](#) (std::function< arma::vec(const arma::vec &DeltaY)> check← DeltaMeasurement)
- Установка функции проверки разности векторов измерений Y.
- void [SetStateTransitionJacobianLinearF](#) (std::function< arma::mat(double dt)> stateTransition← JacobianLinearF)
- Установка функции вычисления матрицы перехода состояния F в случае LKF (makeMatrixF)
- const int [GetSizeX](#) () const
- Получить размерность вектора состояния X.
- const int [GetSizeY](#) () const
- Получить размерность вектора состояния Y.
- const arma::mat & [GetEstimatedCovarianceMatrixP](#) () const
- Получить ковариационную матрицу P состояния X.
- const arma::mat & [GetInnovationCovarianceMatrixS](#) () const
- Получить ковариационную матрицу S вектора невязки DeltaY.
- const arma::mat & [GetDeltaY](#) () const
- Получить вектор невязки измерений DeltaY.
- const arma::mat & [GetKalmanGainMatrixK](#) () const
- Получить матрицу коэффициентов усиления фильтра K.
- const arma::mat & [GetEstimatedVectorX](#) () const
- Получить уточненный вектор состояния X.
- const arma::mat & [GetEstimatedVectorY](#) () const
- Получить уточненный вектор состояния Y.
- virtual void [Prediction](#) (double dt)
- Прогноз
- virtual void [Correction](#) (const arma::vec &Y_msd)
- Коррекция
- virtual void [CalculateInnovationCovarianceS](#) (const arma::vec &PdiagAdd, const arma::vec Rdiag)
- Отдельное вычисление ковариационной матрицы S невязки измерений

Защищенные члены

- void [fixMatrixMainDiagonalSymmetry](#) (arma::mat &A)
- Исправление симметричности матрицы относительно главной диагонали
- void [checkMatrixDiagPositive](#) (const arma::mat &A) const
- Проверка что в диагонали матрицы лежат только положительные элементы

Защищенные данные

- size_t [SizeX_](#)
- Размерность вектора состояния X (state)
- size_t [SizeY_](#)
- Размерность вектора измерений Y (measurement)
- arma::mat [F_](#)
- Матрица эволюции системы (перехода состояния) (state-transition model), размерность [SizeX * SizeX].
- arma::mat [H_](#)

- Матрица измерений (перехода измерений) (observation model), размерность [SizeY * SizeX].
- arma::mat [K_](#)
Коэффициент усиления фильтра Калмана (Kalman gain), размерность [SizeX * SizeY].
- arma::mat [I_](#)
Единичная матрица, размерность [SizeX * SizeX].
- arma::vec [DeltaY_](#)
Вектор невязки измерений, размерность [SizeY].
- arma::mat [P_](#)
Ковариационная матрица вектора состояния X (estimate covariance matrix), размерность [SizeX * SizeX].
- arma::mat [S_](#)
Ковариационная матрица вектора невязки (innovation covariance), размерность [SizeY * SizeY].
- arma::mat [Q_](#)
Ковариационная матрица (обычно диагональная) шумов вектора состояния X НА 1 СЕКУНДЕ (covariance of the process noise), размерность [SizeX * SizeX].
- arma::mat [R_](#)
Ковариационная матрица (обычно диагональная) шумов вектора измерений Y (covariance of the observation noise), размерность [SizeY * SizeY].
- arma::vec [X_pred_](#)
Экстраполированный (predicted) вектор состояния X, размерность [SizeX].
- arma::vec [Y_pred_](#)
Экстраполированный (predicted) вектор измерений Y, размерность [SizeY].
- arma::vec [X_est_](#)
Скорректированный (estimated) вектор состояния X, размерность [SizeX].
- arma::vec [Y_est_](#)
Скорректированный (estimated) вектор измерений Y, размерность [SizeY].
- arma::vec [Y_msd_](#)
Измеренный (measured) вектор Y (отметка), размерность [SizeY].
- bool [deltaY_isSet](#)
Признак установки вектора невязки DeltaY (нужно в методе Correction)
- bool [Y_msd_isSet](#)
Признак установки вектора измерений Y_msd;.
- bool [prediction_isDone](#)
Признак выполненного прогноза (без этого нельзя делать фильтрацию)
- std::function< arma::vec(const arma::vec &X)> [checkBordersStateAfterPrediction_](#)
Проверка границ вектора состояния X после прогноза
- std::function< arma::vec(const arma::vec &X)> [checkBordersStateAfterCorrection_](#)
Проверка границ вектора состояния X после фильтрации
- std::function< arma::vec(const arma::vec &Y)> [checkBordersMeasurement_](#)
Проверка границ вектора измерения Y.
- std::function< arma::vec(const arma::vec &DeltaX)> [checkDeltaState_](#)
Проверка разности векторов состояния X.
- std::function< arma::vec(const arma::vec &DeltaY)> [checkDeltaMeasurement_](#)
Проверка разности векторов измерения Y.

Закрытые данные

- std::function< arma::mat(double dt)> [stateTransitionJacobianLinearF_](#)
Функция вычисления матрицы перехода состояния F (makeMatrixF)

8.5.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanLKF< SizeX, SizeY >
```

Шаблонный класс линейного фильтра Калмана, ЛФК (Linear Kalman Filter, LKF)

Источник: A New Approach to Linear Filtering and Prediction Problems, R.E.KALMAN, Research Institute for Advanced Study, Baltimore, Md., 1960

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.5.2 Конструктор(ы)

8.5.2.1 CKalmanLKF() [1/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanLKF< SizeX, SizeY >::CKalmanLKF ( ) [inline]
Конструктор по умолчанию
```

8.5.2.2 CKalmanLKF() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanLKF< SizeX, SizeY >::CKalmanLKF (
    const CKalmanLKF< SizeX, SizeY > & ) [default]
```

8.5.2.3 CKalmanLKF() [3/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanLKF< SizeX, SizeY >::CKalmanLKF (
    CKalmanLKF< SizeX, SizeY > && ) [default]
```

8.5.2.4 ~CKalmanLKF()

```
template<size_t SizeX, size_t SizeY>
virtual KalmanFilters::CKalmanLKF< SizeX, SizeY >::~~CKalmanLKF ( ) [virtual], [default]
```

8.5.3 Методы

8.5.3.1 CalculateInnovationCovarianceS()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanLKF< SizeX, SizeY >::CalculateInnovationCovarianceS (
    const arma::vec & PdiagAdd,
    const arma::vec Rdiag ) [inline], [virtual]
```

Отдельное вычисление ковариационной матрицы S невязки измерений

Отдельное вычисление S применяется при стробировании

Внимание

Необходимо учесть 2 обстоятельства: 1) метод должен выполняться после вызова метода Prediction, где должна быть вычислена матрица H; 2) матрица R при вычислении S будет использована та, которая уже имеется в фильтре и если необходима иная матрица R, то следует её установить, вызвав метод SetObservationCovarianceMatrixR

Аргументы

| | |
|----------|---|
| PdiagAdd | - добавка в диагональ матрицы P, размерность [SizeX] |
| Rdiag | - диагональная матрица R априорных шумов измерений, размерность [SizeY] |

8.5.3.2 checkMatrixDiagPositive()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::checkMatrixDiagPositive (
    const arma::mat & A ) const [inline], [protected]
```

Проверка что в диагонали матрицы лежат только положительные элементы

Аргументы

| | |
|---|-----------------------|
| A | - проверяемая матрица |
|---|-----------------------|

Внимание

Метод вызывает assert!

8.5.3.3 Correction()

```
template<size_t SizeX, size_t SizeY>
```

```
virtual void KalmanFilters::CKalmanLKF< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяется в [KalmanFilters::CKalmanCKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRCKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >](#), [KalmanFilters::CKalmanEKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanECKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRECKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanEUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >](#), [KalmanFilters::CKalmanUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRUKF< SizeX, SizeY >](#) и [KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >](#).

8.5.3.4 fixMatrixMainDiagonalSymmetry()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::fixMatrixMainDiagonalSymmetry (
    arma::mat & A ) [inline], [protected]
```

Исправление симметричности матрицы относительно главной диагонали

Элементам вне главной диагонали присваивается полусумма между соответствующими элементами

Аргументы

| | |
|---|-----------------------|
| A | - проверяемая матрица |
|---|-----------------------|

8.5.3.5 GetDeltaY()

```
template<size_t SizeX, size_t SizeY>
```

```
const arma::mat & KalmanFilters::CKalmanLKF<SizeX, SizeY>::GetDeltaY ( ) const [inline]
```

Получить вектор невязки измерений DeltaY.

Возвращает

Вектор невязки измерений DeltaY

8.5.3.6 GetEstimatedCovarianceMatrixP()

```
template<size_t SizeX, size_t SizeY>
```

```
const arma::mat & KalmanFilters::CKalmanLKF<SizeX, SizeY>::GetEstimatedCovarianceMatrixP ( ) const [inline]
```

Получить ковариационную матрицу P состояния X.

Возвращает

Текущая ковариационная матрица P

8.5.3.7 GetEstimatedVectorX()

```
template<size_t SizeX, size_t SizeY>
```

```
const arma::mat & KalmanFilters::CKalmanLKF<SizeX, SizeY>::GetEstimatedVectorX ( ) const [inline]
```

Получить уточненный вектор состояния X.

Возвращает

Текущий уточненный вектор состояния X

8.5.3.8 GetEstimatedVectorY()

```
template<size_t SizeX, size_t SizeY>
```

```
const arma::mat & KalmanFilters::CKalmanLKF<SizeX, SizeY>::GetEstimatedVectorY ( ) const [inline]
```

Получить уточненный вектор состояния Y.

Возвращает

Текущий уточненный вектор состояния Y

8.5.3.9 GetInnovationCovarianceMatrixS()

```
template<size_t SizeX, size_t SizeY>
```

```
const arma::mat & KalmanFilters::CKalmanLKF<SizeX, SizeY>::GetInnovationCovarianceMatrixS ( ) const [inline]
```

Получить ковариационную матрицу S вектора невязки DeltaY.

Возвращает

Текущая ковариационная матрица S

8.5.3.10 GetKalmanGainMatrixK()

```
template<size_t SizeX, size_t SizeY>
const arma::mat & KalmanFilters::CKalmanLKF< SizeX, SizeY >::GetKalmanGainMatrixK ( ) const [inline]
Получить матрицу коэффициентов усиления фильтра K.
```

Возвращает

Текущая матрица коэффициентов усиления фильтра K

8.5.3.11 GetSizeX()

```
template<size_t SizeX, size_t SizeY>
const int KalmanFilters::CKalmanLKF< SizeX, SizeY >::GetSizeX ( ) const [inline]
Получить размерность вектора состояния X.
```

Возвращает

Размерность вектора состояния X

8.5.3.12 GetSizeY()

```
template<size_t SizeX, size_t SizeY>
const int KalmanFilters::CKalmanLKF< SizeX, SizeY >::GetSizeY ( ) const [inline]
Получить размерность вектора состояния Y.
```

Возвращает

Размерность вектора состояния Y

8.5.3.13 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanLKF & KalmanFilters::CKalmanLKF< SizeX, SizeY >::operator= (
    CKalmanLKF< SizeX, SizeY > && ) [default]
```

8.5.3.14 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanLKF & KalmanFilters::CKalmanLKF< SizeX, SizeY >::operator= (
    const CKalmanLKF< SizeX, SizeY > & ) [default]
```

8.5.3.15 Prediction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanLKF< SizeX, SizeY >::Prediction (
    double dt ) [inline], [virtual]
```

Прогноз

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

Переопределяется в [KalmanFilters::CKalmanCKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRCKF< SizeX, SizeY >](#)

[KalmanFilters::CKalmanEKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanECKF< SizeX, SizeY >](#),
[KalmanFilters::CKalmanSRECKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREKF< SizeX, SizeY >](#),
[KalmanFilters::CKalmanEUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREUKF< SizeX, SizeY >](#),
[KalmanFilters::CKalmanUKF< SizeX, SizeY >](#) и [KalmanFilters::CKalmanSRUKF< SizeX, SizeY >](#).

8.5.3.16 SetCheckBordersMeasurement()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetCheckBordersMeasurement (
    std::function< arma::vec(const arma::vec &Y)> checkBordersMeasurement ) [inline]
```

Установка функции проверки вектора измерений Y.

Аргументы

| | |
|-------------------------|--------------------------------------|
| checkBordersMeasurement | - Функция проверки вектора измерений |
|-------------------------|--------------------------------------|

8.5.3.17 SetCheckBordersStateAfterCorrection()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetCheckBordersStateAfterCorrection (
    std::function< arma::vec(const arma::vec &X)> checkBordersStateAfterCorrection ) [inline]
```

Установка функции проверки вектора состояния X после коррекции

Аргументы

| | |
|----------------------------------|--|
| checkBordersStateAfterCorrection | - Функция проверки вектора состояния после коррекции |
|----------------------------------|--|

8.5.3.18 SetCheckBordersStateAfterPrediction()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetCheckBordersStateAfterPrediction (
    std::function< arma::vec(const arma::vec &X)> checkBordersStateAfterPrediction ) [inline]
```

Установка функции проверки вектора состояния X после прогноза

Аргументы

| | |
|----------------------------------|---|
| checkBordersStateAfterPrediction | - Функция проверки вектора состояния после прогноза |
|----------------------------------|---|

8.5.3.19 SetCheckDeltaMeasurement()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetCheckDeltaMeasurement (
    std::function< arma::vec(const arma::vec &DeltaY)> checkDeltaMeasurement ) [inline]
```

Установка функции проверки разности векторов измерений Y.

Аргументы

| | |
|-----------------------|--|
| checkDeltaMeasurement | - Функция проверки разности векторов измерений |
|-----------------------|--|

8.5.3.20 SetCheckDeltaState()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetCheckDeltaState (
    std::function< arma::vec(const arma::vec &DeltaX)> checkDeltaState ) [inline]
```

Установка функции проверки разности векторов состояний X.

Аргументы

| | |
|-----------------|--|
| checkDeltaState | - Функция проверки разности векторов состояний |
|-----------------|--|

8.5.3.21 SetDeltaY()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetDeltaY (
    const arma::vec & DeltaY ) [inline]
```

Установка вектора невязки измерений DeltaY.

Аргументы

| | |
|--------|---|
| DeltaY | - вектор невязки измерений, размерность [SizeY] |
|--------|---|

8.5.3.22 SetEstimateCovarianceMatrixP()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetEstimateCovarianceMatrixP (
    const arma::mat & P ) [inline]
```

Установка ковариационной матрицы P состояния X.

Аргументы

| | |
|---|--|
| P | - матрица шумов состояния X, размерность [SizeX * SizeX] |
|---|--|

8.5.3.23 SetEstimateCovarianceMatrixPdiag()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetEstimateCovarianceMatrixPdiag (
    const arma::vec & Pdiag ) [inline]
```

Установка диагонали ковариационной матрицы P состояния X.

Аргументы

| | |
|-------|--|
| Pdiag | - вектор главной диагонали шумов состояния X (остальные элементы полагаются равными нулю), размерность [SizeX] |
|-------|--|

8.5.3.24 SetEstimatedVectorX()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetEstimatedVectorX (
    const arma::vec & X_est ) [inline]
```

Установка оценки вектора состояния X .

Требуется, например, при начальной установке фильтра

Аргументы

| | |
|--------------------|---|
| <code>X_est</code> | - вектор состояния, размерность [SizeX] |
|--------------------|---|

8.5.3.25 SetEstimatedVectorY()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetEstimatedVectorY (
    const arma::vec & Y_est ) [inline]
```

Установка оценки вектора состояния Y .

Требуется, например, при начальной установке фильтра

Аргументы

| | |
|--------------------|---|
| <code>Y_est</code> | - вектор состояния, размерность [SizeY] |
|--------------------|---|

8.5.3.26 SetMeasuredVectorY()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetMeasuredVectorY (
    const arma::vec & Y_msd ) [inline]
```

Установка измеренного вектора измерений Y .

Требуется после получения новых измерений

Аргументы

| | |
|--------------------|---|
| <code>Y_msd</code> | - вектор измерений, размерность [SizeY] |
|--------------------|---|

8.5.3.27 SetObservationCovarianceMatrixR()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetObservationCovarianceMatrixR (
    const arma::mat & R ) [inline]
```

Установка ковариационной матрицы R шумов измерений Y .

Аргументы

| | |
|----------------|--|
| <code>R</code> | - матрица R , размерность [SizeY, SizeY] |
|----------------|--|

8.5.3.28 SetObservationCovarianceMatrixRdiag()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetObservationCovarianceMatrixRdiag (
    const arma::vec & Rdiag ) [inline]
```

Установка ковариационной матрицы R шумов измерений Y .

Аргументы

| | |
|-------|--|
| Rdiag | - вектор элементов главной диагонали матрицы R (остальные элементы полагаются равными нулю), размерность [SizeY] |
|-------|--|

8.5.3.29 SetObservationMatrixH()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetObservationMatrixH (
    const arma::mat & H ) [inline]
```

Установка матрицы перехода измерений H.

Аргументы

| | |
|---|---|
| H | - матрица перехода измерений, размерность [SizeY * SizeX] |
|---|---|

8.5.3.30 SetProcessCovarianceMatrixQ()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetProcessCovarianceMatrixQ (
    const arma::mat & Q ) [inline]
```

Установка ковариационной матрицы Q шумов состояния X.

Аргументы

| | |
|---|---|
| Q | - матрица Q, размерность [SizeX, SizeX] |
|---|---|

8.5.3.31 SetProcessCovarianceMatrixQdiag()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetProcessCovarianceMatrixQdiag (
    const arma::vec & Qdiag ) [inline]
```

Установка диагонали ковариационной матрицы Q шумов состояния X.

Аргументы

| | |
|-------|--|
| Qdiag | - вектор элементов главной диагонали матрицы Q (остальные элементы полагаются равными нулю), размерность [SizeX] |
|-------|--|

8.5.3.32 SetStateTransitionJacobianLinearF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetStateTransitionJacobianLinearF (
    std::function< arma::mat(double dt)> stateTransitionJacobianLinearF ) [inline]
```

Установка функции вычисления матрицы перехода состояния F в случае LKF (makeMatrixF)

Аргументы

| | |
|--------------------------------|--|
| stateTransitionJacobianLinearF | - Функция вычисления матрицы перехода состояния F в случае LKF |
|--------------------------------|--|

8.5.3.33 SetStateTransitionMatrixF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanLKF< SizeX, SizeY >::SetStateTransitionMatrixF (
    const arma::mat & F ) [inline]
```

Установка матрицы перехода состояния F.

Аргументы

| | |
|---|---|
| F | - матрица перехода состояния, размерность [SizeX * SizeX] |
|---|---|

8.5.4 Данные класса

8.5.4.1 checkBordersMeasurement_

```
template<size_t SizeX, size_t SizeY>
std::function<arma::vec( const arma::vec &Y )> KalmanFilters::CKalmanLKF< SizeX, SizeY >::checkBorders↔
Measurement_ [protected]
```

Проверка границ вектора измерения Y.

8.5.4.2 checkBordersStateAfterCorrection_

```
template<size_t SizeX, size_t SizeY>
std::function<arma::vec( const arma::vec &X )> KalmanFilters::CKalmanLKF< SizeX, SizeY >::checkBordersStateAfter↔
Correction_ [protected]
```

Проверка границ вектора состояния X после фильтрации

8.5.4.3 checkBordersStateAfterPrediction_

```
template<size_t SizeX, size_t SizeY>
std::function<arma::vec( const arma::vec &X )> KalmanFilters::CKalmanLKF< SizeX, SizeY >::checkBordersStateAfter↔
Prediction_ [protected]
```

Проверка границ вектора состояния X после прогноза

8.5.4.4 checkDeltaMeasurement_

```
template<size_t SizeX, size_t SizeY>
std::function<arma::vec( const arma::vec &DeltaY )> KalmanFilters::CKalmanLKF< SizeX, SizeY >::checkDelta↔
Measurement_ [protected]
```

Проверка разности векторов измерения Y.

8.5.4.5 checkDeltaState_

```
template<size_t SizeX, size_t SizeY>
std::function<arma::vec( const arma::vec &DeltaX )> KalmanFilters::CKalmanLKF< SizeX, SizeY >::checkDeltaState_
[protected]
```

Проверка разности векторов состояния X.

8.5.4.6 DeltaY_

```
template<size_t SizeX, size_t SizeY>
arma::vec KalmanFilters::CKalmanLKF< SizeX, SizeY >::DeltaY_ [protected]
Вектор невязки измерений, размерность [SizeY].
```

8.5.4.7 deltaY_isSet

```
template<size_t SizeX, size_t SizeY>
bool KalmanFilters::CKalmanLKF< SizeX, SizeY >::deltaY_isSet [protected]
Признак установки вектора невязки DeltaY (нужно в методе Correction)
```

8.5.4.8 F_

```
template<size_t SizeX, size_t SizeY>
arma::mat KalmanFilters::CKalmanLKF< SizeX, SizeY >::F_ [protected]
Матрица эволюции системы (перехода состояния) (state-transition model), размерность [SizeX * SizeX].
```

8.5.4.9 H_

```
template<size_t SizeX, size_t SizeY>
arma::mat KalmanFilters::CKalmanLKF< SizeX, SizeY >::H_ [protected]
Матрица измерений (перехода измерений) (observation model), размерность [SizeY * SizeX].
```

8.5.4.10 I_

```
template<size_t SizeX, size_t SizeY>
arma::mat KalmanFilters::CKalmanLKF< SizeX, SizeY >::I_ [protected]
Единичная матрица, размерность [SizeX * SizeX].
```

8.5.4.11 K_

```
template<size_t SizeX, size_t SizeY>
arma::mat KalmanFilters::CKalmanLKF< SizeX, SizeY >::K_ [protected]
Коэффициент усиления фильтра Калмана (Kalman gain), размерность [SizeX * SizeY].
```

8.5.4.12 P_

```
template<size_t SizeX, size_t SizeY>
arma::mat KalmanFilters::CKalmanLKF< SizeX, SizeY >::P_ [protected]
Ковариационная матрица вектора состояния X (estimate covariance matrix), размерность [SizeX * SizeX].
```

8.5.4.13 prediction_isDone

```
template<size_t SizeX, size_t SizeY>
bool KalmanFilters::CKalmanLKF< SizeX, SizeY >::prediction_isDone [protected]
Признак выполненного прогноза (без этого нельзя делать фильтрацию)
```

8.5.4.14 Q_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanLKF< SizeX, SizeY >::Q_ [protected]
```

Ковариационная матрица (обычно диагональная) шумов вектора состояния X НА 1 СЕКУНДЕ (covariance of the process noise), размерность [SizeX * SizeX].

8.5.4.15 R_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanLKF< SizeX, SizeY >::R_ [protected]
```

Ковариационная матрица (обычно диагональная) шумов вектора измерений Y (covariance of the observation noise), размерность [SizeY * SizeY].

8.5.4.16 S_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanLKF< SizeX, SizeY >::S_ [protected]
```

Ковариационная матрица вектора невязки (innovation covariance), размерность [SizeY * SizeY].

8.5.4.17 SizeX_

```
template<size_t SizeX, size_t SizeY>
```

```
size_t KalmanFilters::CKalmanLKF< SizeX, SizeY >::SizeX_ [protected]
```

Размерность вектора состояния X (state)

8.5.4.18 SizeY_

```
template<size_t SizeX, size_t SizeY>
```

```
size_t KalmanFilters::CKalmanLKF< SizeX, SizeY >::SizeY_ [protected]
```

Размерность вектора измерений Y (measurement)

8.5.4.19 stateTransitionJacobianLinearF_

```
template<size_t SizeX, size_t SizeY>
```

```
std::function<arma::mat( double dt )> KalmanFilters::CKalmanLKF< SizeX, SizeY >::stateTransitionJacobianLinearF_ [private]
```

Функция вычисления матрицы перехода состояния F (makeMatrixF)

Внимание

Используется только в линейном фильтре Калмана

Аргументы

| | |
|---|---|
| X | - вектор состояния с прошлого момента времени |
|---|---|

8.5.4.20 X_est_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::vec KalmanFilters::CKalmanLKF< SizeX, SizeY >::X_est_ [protected]
```

Скорректированный (estimated) вектор состояния X, размерность [SizeX].

8.5.4.21 X_pred_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::vec KalmanFilters::CKalmanLKF< SizeX, SizeY >::X_pred_ [protected]
```

Экстраполированный (predicted) вектор состояния X, размерность [SizeX].

8.5.4.22 Y_est_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::vec KalmanFilters::CKalmanLKF< SizeX, SizeY >::Y_est_ [protected]
```

Скорректированный (estimated) вектор измерений Y, размерность [SizeY].

8.5.4.23 Y_msd_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::vec KalmanFilters::CKalmanLKF< SizeX, SizeY >::Y_msd_ [protected]
```

Измеренный (measured) вектор Y (отметка), размерность [SizeY].

8.5.4.24 Y_msd_isSet

```
template<size_t SizeX, size_t SizeY>
```

```
bool KalmanFilters::CKalmanLKF< SizeX, SizeY >::Y_msd_isSet [protected]
```

Признак установки вектора измерений Y_msd;.

8.5.4.25 Y_pred_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::vec KalmanFilters::CKalmanLKF< SizeX, SizeY >::Y_pred_ [protected]
```

Экстраполированный (predicted) вектор измерений Y, размерность [SizeY].

Объявления и описания членов класса находятся в файле:

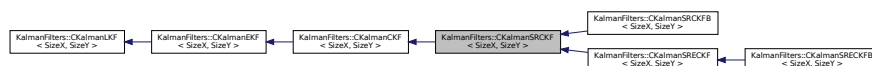
- kalman_filter_linear.h

8.6 Шаблон класса KalmanFilters::CKalmanSRCKF< SizeX, SizeY >

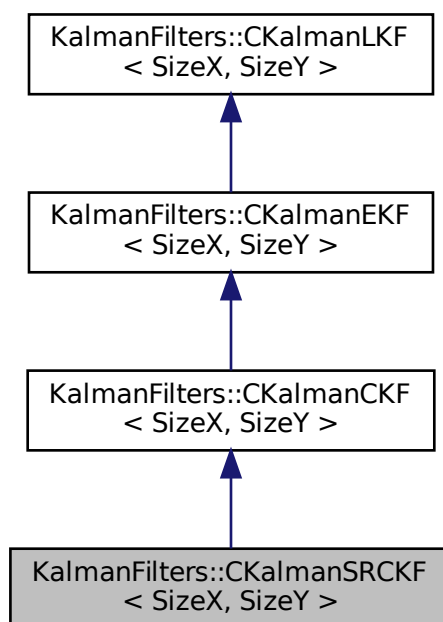
Шаблонный класс квадратно-корневого кубатурного фильтра Калмана, КК-КФК (Square Root Cubature Kalman Filter, SR-CKF)

```
#include <kalman_filter_cubature_square_root.h>
```

Граф наследования: KalmanFilters::CKalmanSRCKF< SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanSRCKF< SizeX, SizeY >:



Открытые члены

- [CKalmanSRCKF](#) ()
Конструктор по умолчанию
- void [SetStateTransitionJacobianF](#) (std::function< arma::mat(const arma::vec &X, double dt)> stateTransitionJacobianF)
Установка функции вычисления матрицы перехода состояния F (makeMatrixF)
- void [SetObservationJacobianH](#) (std::function< arma::mat(const arma::vec &X)> observationJacobianH)
Установка функции вычисления матрицы перехода измерений H (makeMatrixH)
- virtual void [Prediction](#) (double dt)
Прогноз
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция

Защищенные члены

- void [CorrectionSRCKF](#) (const arma::vec &Y_msd)
Коррекция SRCKF.

Дополнительные унаследованные члены

8.6.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanSRCKF< SizeX, SizeY >
```

Шаблонный класс квадратно-корневого кубатурного фильтра Калмана, КК-КФК (Square Root Cubature Kalman Filter, SR-CKF)

Источники:

[1] Cubature Kalman Filters, Ienkaran Arasaratnam and Simon Haykin, Life Fellow, IEEE

Внимание

Фильтр построен по классическому его варианту НИЖНЕтреугольного разложения Холецкого!

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.6.2 Конструктор(ы)

8.6.2.1 CKalmanSRCKF()

```
template<size_t SizeX, size_t SizeY>
```

```
KalmanFilters::CKalmanSRCKF< SizeX, SizeY >::CKalmanSRCKF ( ) [inline]
```

Конструктор по умолчанию

8.6.3 Методы

8.6.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
```

```
virtual void KalmanFilters::CKalmanSRCKF< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanCKF< SizeX, SizeY >](#).

Переопределяется в [KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRECKF< SizeX, SizeY >](#) и [KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >](#).

8.6.3.2 CorrectionSRCKF()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanSRCKF< SizeX, SizeY >::CorrectionSRCKF (
    const arma::vec & Y_msd ) [inline], [protected]
```

Коррекция SRCKF.

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

8.6.3.3 Prediction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanSRCKF< SizeX, SizeY >::Prediction (
    double dt ) [inline], [virtual]
```

Прогноз

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

Переопределяет метод предка [KalmanFilters::CKalmanCKF](#)< SizeX, SizeY >.

Переопределяется в [KalmanFilters::CKalmanSRECKF](#)< SizeX, SizeY >.

8.6.3.4 SetObservationJacobianH()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanSRCKF< SizeX, SizeY >::SetObservationJacobianH (
    std::function< arma::mat(const arma::vec &X)> observationJacobianH ) [inline]
```

Установка функции вычисления матрицы перехода измерений H (makeMatrixH)

См. также

[observationJacobianH_](#)

8.6.3.5 SetStateTransitionJacobianF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanSRCKF< SizeX, SizeY >::SetStateTransitionJacobianF (
    std::function< arma::mat(const arma::vec &X, double dt)> stateTransitionJacobianF ) [inline]
```

Установка функции вычисления матрицы перехода состояния F (makeMatrixF)

См. также

[stateTransitionJacobianF_](#)

Объявления и описания членов класса находятся в файле:

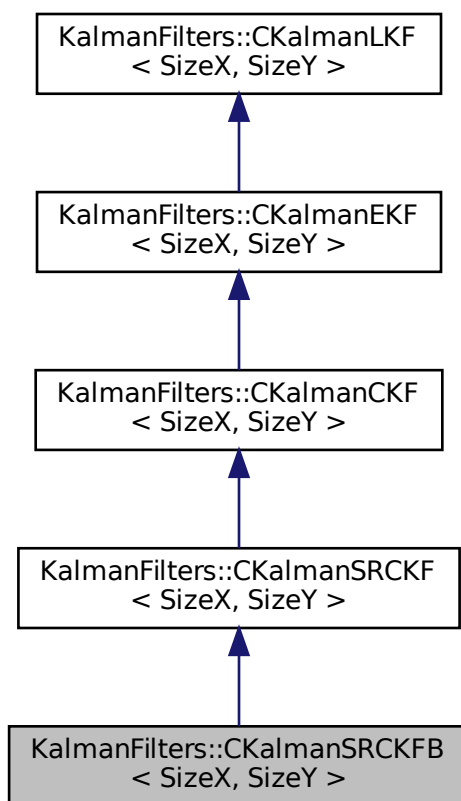
- `kalman_filter_cubature_square_root.h`

8.7 Шаблон класса [KalmanFilters::CKalmanSRCKFB](#)< SizeX, SizeY >

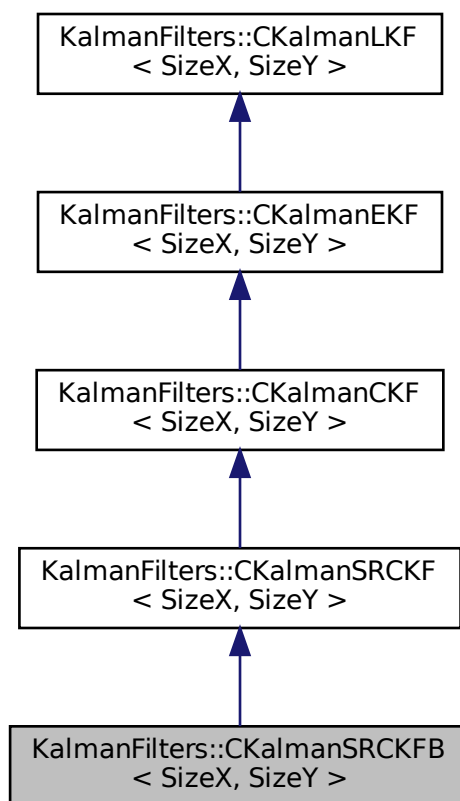
Шаблонный класс квадратно-корневого кубатурного фильтра Калмана (блочная фильтрация), КК-КФКБ (Square Root Cubature Kalman Filter Block, SR-CKFB)

```
#include <kalman_filter_cubature_square_root.h>
```

Граф наследования: KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >:



Открытые члены

- [CKalmanSRCKFB](#) ()
Конструктор по умолчанию
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция

Дополнительные унаследованные члены

8.7.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >
```

Шаблонный класс квадратно-корневого кубатурного фильтра Калмана (блочная фильтрация), КК-КФКБ (Square Root Cubature Kalman Filter Block, SR-CKFB)

См. [CKalmanSRCKF](#) и The J-Orthogonal Square-Root Euler-Maruyama-Based Unscented Kalman Filter for Nonlinear Stochastic Systems, Gennady Yu. Kulikov, Maria V. Kulikova, CEMAT, Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais 1, 1049-001 LISBOA, Portugal (emails: gennady.kulikov[at]tecnico.ulisboa.pt, maria.kulikova[at]ist.utl.pt)

Внимание

Фильтр построен по классическому его варианту НИЖНЕтреугольного разложения Холецкого!

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.7.2 Конструктор(ы)

8.7.2.1 CKalmanSRCKFB()

```
template<size_t SizeX, size_t SizeY>
```

```
KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >::CKalmanSRCKFB ( ) [inline]
```

Конструктор по умолчанию

8.7.3 Методы

8.7.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
```

```
virtual void KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanSRCKF< SizeX, SizeY >](#).

Объявления и описания членов класса находятся в файле:

- kalman_filter_cubature_square_root.h

8.8 Шаблон класса KalmanFilters::CKalmanSRECKF< SizeX, SizeY >

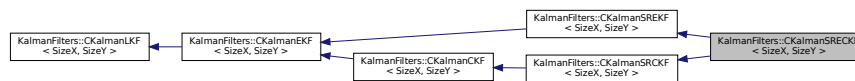
Шаблонный класс квадратно-корневого расширенного кубатурного фильтра Калмана, КК-РКФК (Square Root Extended Cubature Kalman Filter, SR-ECKF)

```
#include <kalman_filter_extended_cubature_square_root.h>
```

Граф наследования: KalmanFilters::CKalmanSRECKF< SizeX, SizeY >:



Граф связей класса `KalmanFilters::CKalmanSRECKF< SizeX, SizeY >`:



Открытые члены

- [CKalmanSRECKF](#) ()
Конструктор по умолчанию
- virtual void [Prediction](#) (double dt)
Прогноз квадратно-корневого расширенного кубатурного фильтра Калмана (КК-РКФК, SR-↔ECKF)
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция квадратно-корневого расширенного кубатурного фильтра Калмана (КК-РКФК, SR-↔ECKF)

Дополнительные унаследованные члены

8.8.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanSRECKF< SizeX, SizeY >
```

Шаблонный класс квадратно-корневого расширенного кубатурного фильтра Калмана, КК-РКФК (Square Root Extended Cubature Kalman Filter, SR-ECKF)

Источники:

[1] Cubature Kalman Filters, Ienkararan Arasaratnam and Simon Haykin, Life Fellow, IEEE

Внимание

Фильтр построен по классическому его варианту НИЖНЕТреугольного разложения Холецкого!

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.8.2 Конструктор(ы)

8.8.2.1 CKalmanSRECKF()

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSRECKF< SizeX, SizeY >::CKalmanSRECKF ( ) [inline]
Конструктор по умолчанию
```

8.8.3 Методы

8.8.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanSRECKF< SizeX, SizeY >::Correction (
    const arma::vec & Y msd ) [inline], [virtual]
```

Коррекция квадратно-корневого расширенного кубатурного фильтра Калмана (КК-РКФК, SR-ECKF)

Аргументы

| | |
|-----------|---|
| Y_{msd} | - вектор измерений, по которым производится коррекция |
|-----------|---|

Переопределяет метод предка `KalmanFilters::CKalmanSRCKF< SizeX, SizeY >`.

Переопределяется в `KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >`.

8.8.3.2 Prediction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanSRECKF< SizeX, SizeY >::Prediction (
    double dt )    [inline], [virtual]
```

Прогноз квадратно-корневого расширенного кубатурного фильтра Калмана (КК-РКФК, SR-ECKF)

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

Переопределяет метод предка `KalmanFilters::CKalmanSRCKF< SizeX, SizeY >`.

Объявления и описания членов класса находятся в файле:

- kalman filter extended cubature square root.h

8.9 Шаблон класса KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >

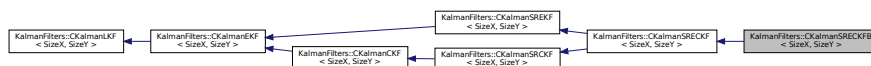
Шаблонный класс квадратно-корневого расширенного кубатурного фильтра Калмана (блочная фильтрация), КК-РКФКБ (Square Root Extended Cubature Kalman Filter Block, SR-ECKFB)

```
#include <kalman_filter_extended_cubature_square_root.h>
```

Граф наследования:KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >:



Открытые члены

- [CKalmanSRECKFB](#) ()
Конструктор по умолчанию
- `virtual void Correction (const arma::vec &Y_msd)`
Коррекция

Дополнительные унаследованные члены

8.9.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >
```

Шаблонный класс квадратно-корневого расширенного кубатурного фильтра Калмана (блочная фильтрация), КК-РКФКБ (Square Root Extended Cubature Kalman Filter Block, SR-ECKFB)

См. [CKalmanSRECKF](#) и The J-Orthogonal Square-Root Euler-Maruyama-Based Unscented Kalman Filter for Nonlinear Stochastic Systems, Gennady Yu. Kulikov, Maria V. Kulikova, CEMAT, Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais 1, 1049-001 LISBOA, Portugal (emails↔: gennady.kulikov[at]tecnico.ulisboa.pt, maria.kulikova[at]ist.utl.pt)

Внимание

Фильтр построен по классическому его варианту НИЖНЕтреугольного разложения Холецкого!

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.9.2 Конструктор(ы)

8.9.2.1 CKalmanSRECKFB()

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >::CKalmanSRECKFB ( ) [inline]
```

Конструктор по умолчанию

8.9.3 Методы

8.9.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanSRECKF< SizeX, SizeY >](#).

Объявления и описания членов класса находятся в файле:

- kalman_filter_extended_cubature_square_root.h

8.10 Шаблон класса KalmanFilters::CKalmanSREKF< SizeX, SizeY >

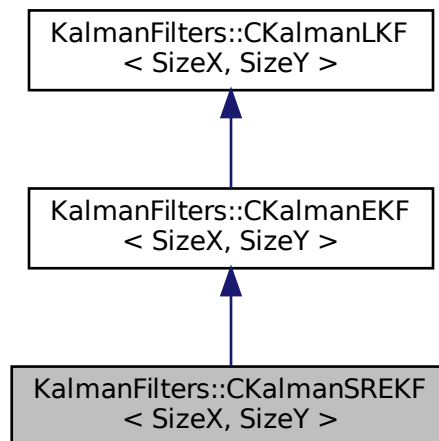
Класс квадратно-корневого расширенного фильтра Калмана, КК-РФК (Square Root Extended Kalman Filter, EKF)

```
#include <kalman_filter_extended_square_root.h>
```

Граф наследования: KalmanFilters::CKalmanSREKF< SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanSREKF< SizeX, SizeY >:



Открытые члены

- [CKalmanSREKF](#) ()
Конструктор по умолчанию
- virtual void [Prediction](#) (double dt)
Прогноз
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция

Защищенные члены

- void [PredictionSREKF](#) (double dt)
Прогноз SREKF.

Дополнительные унаследованные члены

8.10.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanSREKF< SizeX, SizeY >
```

Класс квадратно-корневого расширенного фильтра Калмана, КК-РФК (Square Root Extended Kalman Filter, EKF)

Источник: NASA Technical report R-135, Application of statistical filter theory to the optimal estimation of position and velocity on board a circumlunar vehicle, Gerald L. Smith, Stanley F. Schmidt and Leonard A. McGee, 1962

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.10.2 Конструктор(ы)

8.10.2.1 CKalmanSREKF()

```
template<size_t SizeX, size_t SizeY>
```

```
KalmanFilters::CKalmanSREKF< SizeX, SizeY >::CKalmanSREKF ( ) [inline]
```

Конструктор по умолчанию

8.10.3 Методы

8.10.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
```

```
virtual void KalmanFilters::CKalmanSREKF< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanEKF](#)< SizeX, SizeY >.

Переопределяется в [KalmanFilters::CKalmanSRECKF](#)< SizeX, SizeY >, [KalmanFilters::CKalmanSRECKFB](#)< SizeX, SizeY >, [KalmanFilters::CKalmanSREUKF](#)< SizeX, SizeY > и [KalmanFilters::CKalmanSREUKFB](#)< SizeX, SizeY >.

8.10.3.2 Prediction()

```
template<size_t SizeX, size_t SizeY>
```

```
virtual void KalmanFilters::CKalmanSREKF< SizeX, SizeY >::Prediction (
    double dt ) [inline], [virtual]
```

Прогноз

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

Переопределяет метод предка [KalmanFilters::CKalmanEKF](#)< SizeX, SizeY >.

Переопределяется в [KalmanFilters::CKalmanSRECKF](#)< SizeX, SizeY > и [KalmanFilters::CKalmanSREUKF](#)< SizeX, SizeY >.

8.10.3.3 PredictionSREKF()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanSREKF< SizeX, SizeY >::PredictionSREKF (
    double dt ) [inline], [protected]
```

Прогноз SREKF.

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

Объявления и описания членов класса находятся в файле:

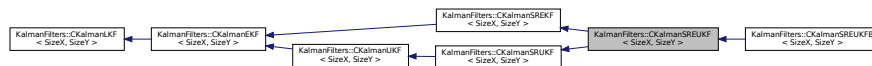
- kalman_filter_extended_square_root.h

8.11 Шаблон класса KalmanFilters::CKalmanSREUKF< SizeX, SizeY >

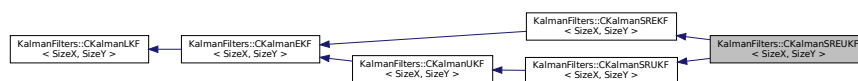
Шаблонный класс квадратно-корневого расширенного сигма-точечного (ансцентного) фильтра Калмана, КК-РСТФК (Square Root Extended Unscented Kalman Filter, SR-EUKF)

```
#include <kalman_filter_extended_unscented_square_root.h>
```

Граф наследования: KalmanFilters::CKalmanSREUKF< SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanSREUKF< SizeX, SizeY >:



Открытые члены

- [CKalmanSREUKF](#) ()
Конструктор по умолчанию
- virtual void [Prediction](#) (double dt)
Прогноз квадратно-корневого расширенного сигма-точечного фильтра Калмана (КК-РСТФК, SR-EUKF)
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция квадратно-корневого расширенного сигма-точечного фильтра Калмана (КК-РСТФК, SR-EUKF)

Защищенные члены

- void [createSignMatrices](#) ()
Создание матриц знаков

Защищенные данные

- arma::mat [J_](#)
Матрица знаков при Rx.
- arma::mat [Jcorrect_](#)
Матрица знаков при коррекции

8.11.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanSREUKF< SizeX, SizeY >
```

Шаблонный класс квадратно-корневого расширенного сигма-точечного (ансцентного) фильтра Калмана, КК-РСТФК (Square Root Extended Unscented Kalman Filter, SR-EUKF)

Источники:

- [1] A New Extension of the Kalman Filter to Nonlinear Systems, Simon J. Julier, Jeffrey K. Uhlmann, The Robotics Research Group, Department of Engineering Science, The University of Oxford, 1997
- [2] The Unscented Kalman Filter for Nonlinear Estimation, Eric A. Wan and Rudolph van der Merwe Oregon Graduate Institute of Science & Technology 20000 NW Walker Rd, Beaverton, Oregon 97006, 2000 ericwan[at]ece.ogi.edu, rvdmerwe[at]ece.ogi.edu
- [3] Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models, Rudolph van der Merwe & Eric Wan, OGI School of Science & Engineering Oregon Health & Science University Beaverton, Oregon, 97006, USA, 2003 {rvdmerwe,ericwan}[at]ece.ogi.edu
- [4] THE SQUARE-ROOT UNSCENTED KALMAN FILTER FOR STATE AND PARAMETER-ESTIMATION, Rudolph van der Merwe and Eric A. Wan, Oregon Graduate Institute of Science and Technology 20000 NW Walker Road, Beaverton, Oregon 97006, USA rvdmerwe,ericwan[at]ece.ogi.edu

Внимание

Внимание! В реализации UKF вес нулевой сигма-точки W_{cov} не может быть отрицательным! (W_{mean} - может)

Фильтр построен по классическому его варианту НИЖНЕтреугольного разложения Холецкого!

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.11.2 Конструктор(ы)

8.11.2.1 CKalmanSREUKF()

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSREUKF< SizeX, SizeY >::CKalmanSREUKF ( ) [inline]
Конструктор по умолчанию
```

8.11.3 Методы

8.11.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanSREUKF< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция квадратно-корневого расширенного сигма-точечного фильтра Калмана (КК-РСТФК, SR-EUKF)

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanSREKF< SizeX, SizeY >](#).
 Переопределяется в [KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >](#).

8.11.3.2 createSignMatrices()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanSREUKF< SizeX, SizeY >::createSignMatrices ( ) [inline], [protected]
```

Создание матриц знаков

8.11.3.3 Prediction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanSREUKF< SizeX, SizeY >::Prediction (
    double dt ) [inline], [virtual]
```

Прогноз квадратно-корневого расширенного сигма-точечного фильтра Калмана (КК-РСТФК, SR-EUKF)

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

Переопределяет метод предка [KalmanFilters::CKalmanSREKF< SizeX, SizeY >](#).

8.11.4 Данные класса

8.11.4.1 J_

```
template<size_t SizeX, size_t SizeY>
arma::mat KalmanFilters::CKalmanSREUKF< SizeX, SizeY >::J_ [protected]
```

Матрица знаков при Pху.

8.11.4.2 Jcorrect_

```
template<size_t SizeX, size_t SizeY>
arma::mat KalmanFilters::CKalmanSREUKF< SizeX, SizeY >::Jcorrect_ [protected]
```

Матрица знаков при коррекции

Объявления и описания членов класса находятся в файле:

- kalman_filter_extended_unscented_square_root.h

8.12 Шаблон класса KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >

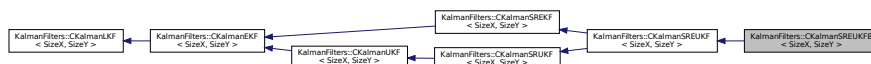
Шаблонный класс квадратно-корневого расширенного сигма-точечного (ансцентного) фильтра Калмана (блочная фильтрация), КК-РСТФК (Square Root Extended Unscented Kalman Filter Block, SR-EUKFB)

```
#include <kalman_filter_extended_unscented_square_root.h>
```

Граф наследования: KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >:



Открытые члены

- [CKalmanSREUKFB](#) ()
Конструктор по умолчанию
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция

Защищенные члены

- void [createSignMatricesBlock](#) ()
Создание матриц знаков

Защищенные данные

- arma::mat [JcorrectBlock_](#)
Матрица знаков для фильтра в блочном виде

8.12.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >
```

Шаблонный класс квадратно-корневого расширенного сигма-точечного (ансцентного) фильтра Калмана (блочная фильтрация), КК-РСТФК (Square Root Extended Unscented Kalman Filter Block, SR-EUKFB)

Источники:

- [1] A New Extension of the Kalman Filter to Nonlinear Systems, Simon J. Julier, Jeffrey K. Uhlmann, The Robotics Research Group, Department of Engineering Science, The University of Oxford, 1997
- [2] The Unscented Kalman Filter for Nonlinear Estimation, Eric A. Wan and Rudolph van der Merwe Oregon Graduate Institute of Science & Technology 20000 NW Walker Rd, Beaverton, Oregon 97006, 2000 ericwan[at]ece.ogi.edu, rvdmerwe[at]ece.ogi.edu
- [3] Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models, Rudolph van der Merwe & Eric Wan, OGI School of Science & Engineering Oregon Health & Science University Beaverton, Oregon, 97006, USA, 2003 {rvdmerwe,ericwan}[at]ece.ogi.edu
- [4] THE SQUARE-ROOT UNSCENTED KALMAN FILTER FOR STATE AND PARAMETER-ESTIMATION, Rudolph van der Merwe and Eric A. Wan, Oregon Graduate Institute of Science and Technology 20000 NW Walker Road, Beaverton, Oregon 97006, USA rvdmerwe,ericwan[at]ece.ogi.edu

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.12.2 Конструктор(ы)

8.12.2.1 CKalmanSREUKFB()

```
template<size_t SizeX, size_t SizeY>
```

```
KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >::CKalmanSREUKFB ( ) [inline]
```

Конструктор по умолчанию

8.12.3 Методы

8.12.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
```

```
virtual void KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanSREUKF< SizeX, SizeY >](#).

8.12.3.2 createSignMatricesBlock()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >::createSignMatricesBlock ( ) [inline], [protected]
```

Создание матриц знаков

8.12.4 Данные класса

8.12.4.1 JcorrectBlock_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >::JcorrectBlock_ [protected]
```

Матрица знаков для фильтра в блочном виде

Объявления и описания членов класса находятся в файле:

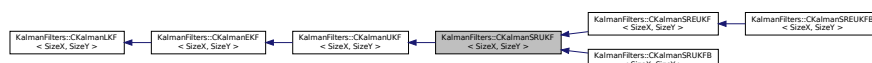
- kalman_filter_extended_unscented_square_root.h

8.13 Шаблон класса KalmanFilters::CKalmanSRUKF< SizeX, SizeY >

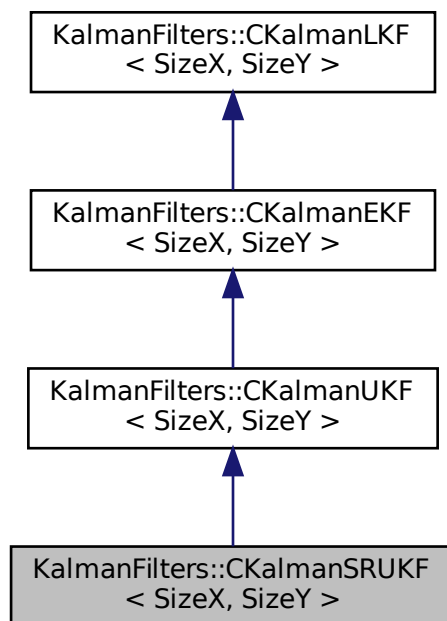
Шаблонный класс квадратно-корневого сигма-точечного (ансцентного) фильтра Калмана, КК-СТФК (КК-АФК) (Square Root Unscented Kalman Filter, SR-UKF)

```
#include <kalman_filter_unscented_square_root.h>
```

Граф наследования: KalmanFilters::CKalmanSRUKF< SizeX, SizeY >:



Граф связей класса `KalmanFilters::CKalmanSRUKF< SizeX, SizeY >`:



Открытые члены

- [CKalmanSRUKF](#) ()
Конструктор по умолчанию
- virtual void [SetupDesignParametersMeanSet](#) (double w0)
Установка параметра w0 апсцентного фильтра (MeanSet)
- virtual void [SetupDesignParametersScaledSet](#) (double alpha, double beta, double kappa)
Установка параметров масштабируемого апсцентного преобразования (Scaled UT)
- virtual void [SetupDesignParametersCDKF](#) (double h2)
Установка параметра фильтра по рекомендации Central Difference Kalman Filter (CDKF)
- void [SetStateTransitionJacobianF](#) (std::function< arma::mat(const arma::vec &X, double dt)> stateTransitionJacobianF)
Установка функции вычисления матрицы перехода состояния F (makeMatrixF)
- void [SetObservationJacobianH](#) (std::function< arma::mat(const arma::vec &X)> observationJacobianH)
Установка функции вычисления матрицы перехода измерений H (makeMatrixH)
- virtual void [Prediction](#) (double dt)
Прогноз
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция

Защищенные члены

- void [CorrectionSRUKF](#) (const arma::vec &Y_msd)
Коррекция SRUKF.
- void [createSignMatrices](#) ()
Создание матриц знаков

Защищенные данные

- arma::mat [J_](#)
Матрица знаков при Pxy.
- arma::mat [Jpredict_](#)
Матрица знаков при прогнозе
- arma::mat [Jcorrect_](#)
Матрица знаков при коррекции
- bool [negativeZeroCovWeight_](#)
Признак отрицательного веса "нулевой" сигма-точки Wcov.

8.13.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanSRUKF< SizeX, SizeY >
```

Шаблонный класс квадратно-корневого сигма-точечного (ансцентного) фильтра Калмана, КК-СТФК (КК-АФК) (Square Root Unscented Kalman Filter, SR-UKF)

Источники:

- [1] A New Extension of the Kalman Filter to Nonlinear Systems, Simon J. Julier, Jeffrey K. Uhlmann, The Robotics Research Group, Department of Engineering Science, The University of Oxford, 1997
- [2] The Unscented Kalman Filter for Nonlinear Estimation, Eric A. Wan and Rudolph van der Merwe Oregon Graduate Institute of Science & Technology 20000 NW Walker Rd, Beaverton, Oregon 97006, 2000 ericwan@ece.ogi.edu, rvdmerwe@ece.ogi.edu
- [3] Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models, Rudolph van der Merwe & Eric Wan, OGI School of Science & Engineering Oregon Health & Science University Beaverton, Oregon, 97006, USA, 2003 {rvdmerwe,ericwan}@at|ece.ogi.edu
- [4] THE SQUARE-ROOT UNSCENTED KALMAN FILTER FOR STATE AND PARAMETER-ESTIMATION, Rudolph van der Merwe and Eric A. Wan, Oregon Graduate Institute of Science and Technology 20000 NW Walker Road, Beaverton, Oregon 97006, USA rvdmerwe,ericwan [at]ece.ogi.edu
- [5] The J-Orthogonal Square-Root Euler-Maruyama-Based Unscented Kalman Filter for Nonlinear Stochastic Systems, Gennady Yu. Kulikov, Maria V. Kulikova, CEMAT, Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais 1, 1049-001 LISBOA, Portugal (emails: gennady.kulikov[at]tecnico.ulisboa.pt, maria.kulikova[at]ist.utl.pt)

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.13.2 Конструктор(ы)

8.13.2.1 CKalmanSRUKF()

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::CKalmanSRUKF ( ) [inline]
Конструктор по умолчанию
```

8.13.3 Методы

8.13.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::Correction (
```

```
const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanUKF< SizeX, SizeY >](#).

Переопределяется в [KalmanFilters::CKalmanSREUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >](#) и [KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >](#).

8.13.3.2 CorrectionSRUKF()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::CorrectionSRUKF (
```

```
const arma::vec & Y_msd ) [inline], [protected]
```

Коррекция SRUKF.

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

8.13.3.3 createSignMatrices()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::createSignMatrices ( ) [inline], [protected]
```

Создание матриц знаков

8.13.3.4 Prediction()

```
template<size_t SizeX, size_t SizeY>
```

```
virtual void KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::Prediction (
```

```
double dt ) [inline], [virtual]
```

Прогноз

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

Переопределяет метод предка [KalmanFilters::CKalmanUKF< SizeX, SizeY >](#).

Переопределяется в [KalmanFilters::CKalmanSREUKF< SizeX, SizeY >](#).

8.13.3.5 SetObservationJacobianH()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::SetObservationJacobianH (
```

```
std::function< arma::mat(const arma::vec &X)> observationJacobianH ) [inline]
```

Установка функции вычисления матрицы перехода измерений H (makeMatrixH)

См. также

[observationJacobianH_](#)

8.13.3.6 SetStateTransitionJacobianF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::SetStateTransitionJacobianF (
    std::function< arma::mat(const arma::vec &X, double dt)> stateTransitionJacobianF ) [inline]
Установка функции вычисления матрицы перехода состояния F (makeMatrixF)
```

См. также

[stateTransitionJacobianF_](#)

8.13.3.7 SetupDesignParametersCDKF()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::SetupDesignParametersCDKF (
    double h2 ) [inline], [virtual]
Установка параметра фильтра по рекомендации Central Difference Kalman Filter (CDKF)
Смотри [3]
```

Аргументы

| | |
|----|---|
| h2 | - параметр разброса сигма-точек ($h^2 = 3$ типичная рекомендация для гауссовых шумов, [3]) |
|----|---|

Переопределяет метод предка [KalmanFilters::CKalmanUKF< SizeX, SizeY >](#).

8.13.3.8 SetupDesignParametersMeanSet()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::SetupDesignParametersMeanSet (
    double w0 ) [inline], [virtual]
Установка параметра w0 ансцентного фильтра (MeanSet)
При выборе w0 = [0...1] обеспечиваются положительные веса. Смотри [1], [2]
```

Внимание

Нельзя выбирать w0 так, чтобы нулевой вес был > 0 , а остальные меньше нуля. Наоборот - МОЖНО, т.е. нулевой вес может быть отрицательным.

Аргументы

| | |
|----|---|
| w0 | - параметр разброса сигма точек ($w0 = [0...1]$ типичная рекомендация для положительных весов) |
|----|---|

Переопределяет метод предка [KalmanFilters::CKalmanUKF< SizeX, SizeY >](#).

8.13.3.9 SetupDesignParametersScaledSet()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::SetupDesignParametersScaledSet (
    double alpha,
    double beta,
    double kappa ) [inline], [virtual]
Установка параметров масштабируемого ансцентного преобразования (Scaled UT)
Смотри [2], [5], [6]
```

Аргументы

| | |
|-------|---|
| alpha | - параметр разброса сигма-точек ($\alpha = 10^{-3}$ - типичная рекомендация по van der Merwe) |
| beta | - параметр, отвечающий за характер распределения ($\beta = 2$ - нормальное) |
| карра | - параметр, отвечающий за разброс сигма-точек ($\kappa = 0$ или $(3 - \text{SizeX})$ - типичная рекомендация по van der Merwe) |

Переопределяет метод предка `KalmanFilters::CKalmanUKF< SizeX, SizeY >`.

8.13.4 Данные класса

8.13.4.1 J_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::J_ [protected]
```

Матрица знаков при P_{xy} .

8.13.4.2 Jcorrect_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::Jcorrect_ [protected]
```

Матрица знаков при коррекции

8.13.4.3 Jpredict_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::Jpredict_ [protected]
```

Матрица знаков при прогнозе

8.13.4.4 negativeZeroCovWeight_

```
template<size_t SizeX, size_t SizeY>
```

```
bool KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::negativeZeroCovWeight_ [protected]
```

Признак отрицательного веса "нулевой" сигма-точки W_{cov} .

Объявления и описания членов класса находятся в файле:

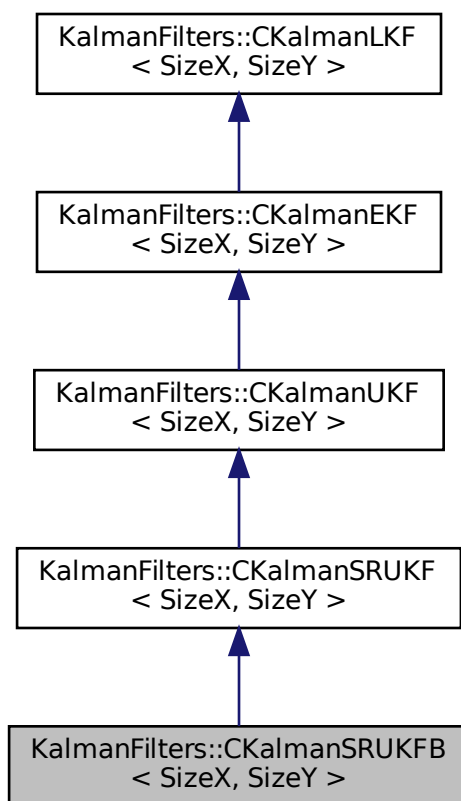
- `kalman_filter_unscented_square_root.h`

8.14 Шаблон класса `KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >`

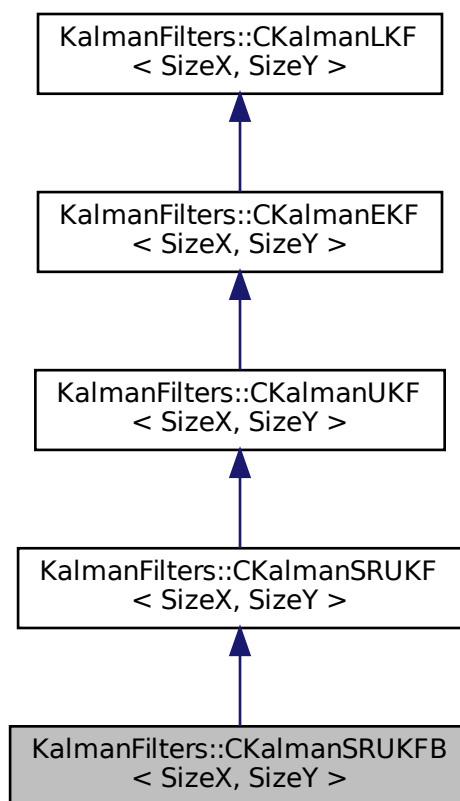
Шаблонный класс квадратно-корневого сигма-точечного (ансцентного) фильтра Калмана (блочная фильтрация), КК-СТФКБ (КК-АФКБ) (Square Root Unscented Kalman Filter Block, SR-UKFB)

```
#include <kalman_filter_unscented_square_root.h>
```


Граф наследования:KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >:



Открытые члены

- [CKalmanSRUKFB](#) ()
Конструктор по умолчанию
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция квадратно-корневого сигма-точечного фильтра Калмана (КК-СТФК, SR-UKF)

Защищенные члены

- void [createSignMatricesBlock](#) ()
Создание матриц знаков

Защищенные данные

- arma::mat [JcorrectBlock_](#)
Матрица знаков для фильтра в блочном виде

8.14.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >
```

Шаблонный класс квадратно-корневого сигма-точечного (ансцентного) фильтра Калмана (блочная фильтрация), КК-СТФКБ (КК-АФКБ) (Square Root Unscented Kalman Filter Block, SR-UKFB) см. [CKalmanSRUKF](#)

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.14.2 Конструктор(ы)

8.14.2.1 CKalmanSRUKFB()

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >::CKalmanSRUKFB ( ) [inline]
Конструктор по умолчанию
```

8.14.3 Методы

8.14.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция квадратно-корневого сигма-точечного фильтра Калмана (КК-СТФК, SR-UKF)

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanSRUKF< SizeX, SizeY >](#).

8.14.3.2 createSignMatricesBlock()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >::createSignMatricesBlock ( ) [inline], [protected]
Создание матриц знаков
```

8.14.4 Данные класса

8.14.4.1 JcorrectBlock_

```
template<size_t SizeX, size_t SizeY>
arma::mat KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >::JcorrectBlock_ [protected]
```

Матрица знаков для фильтра в блочном виде

Объявления и описания членов класса находятся в файле:

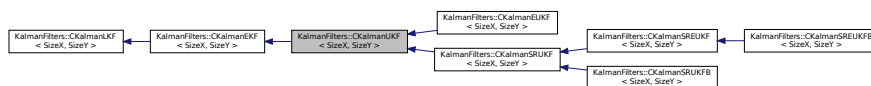
- `kalman_filter_unscented_square_root.h`

8.15 Шаблон класса KalmanFilters::CKalmanUKF< SizeX, SizeY >

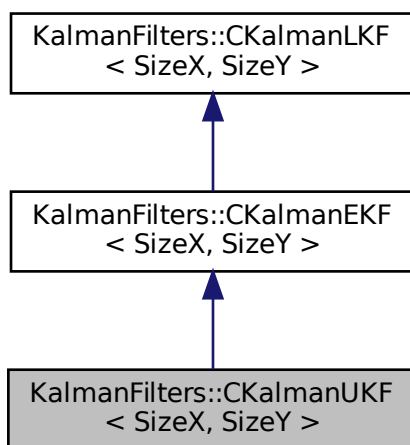
Шаблонный класс сигма-точечного (ансцентного) фильтра Калмана, СТФК (АФК) (Unscented Kalman Filter, UKF)

#include <kalman_filter_unscented.h>

Граф наследования: KalmanFilters::CKalmanUKF< SizeX, SizeY >:



Граф связей класса KalmanFilters::CKalmanUKF< SizeX, SizeY >:



Открытые члены

- [CKalmanUKF](#) ()
Конструктор по умолчанию
- virtual void [SetupDesignParametersMeanSet](#) (double w0)
Установка параметра w0 ансцентного фильтра (MeanSet)
- virtual void [SetupDesignParametersScaledSet](#) (double alpha, double beta, double kappa)
Установка параметров масштабируемого ансцентного преобразования (Scaled UT)
- virtual void [SetupDesignParametersCDKF](#) (double h2)
Установка параметра фильтра по рекомендации Central Difference Kalman Filter (CDKF)
- void [SetWeightedSumStateSigmas](#) (std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> weightedSumStateSigmas)
Установка функции вычисления взвешенной суммы сигма-точек пространства X
- void [SetWeightedSumMeasurementSigmas](#) (std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> weightedSumMeasurementSigmas)
Установка функции вычисления взвешенной суммы сигма-точек пространства Y.
- virtual void [Prediction](#) (double dt)
Прогноз
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция

Защищенные члены

- void [PredictionUKF](#) (double dt)
Прогноз UKF.
- void [CorrectionUKF](#) (const arma::vec &Y_msd)
Коррекция UKF.
- void [SetStateTransitionJacobianF](#) ()
Запрет доступа
- void [SetObservationJacobianH](#) ()
Запрет доступа

Защищенные данные

- int [k_sigma_points_](#)
Число сигма-точек
- arma::vec [weights_mean_](#)
Веса среднего
- arma::vec [weights_covariance_](#)
Веса ковариации
- arma::mat [x_est_sigma_points_](#)
Матрица сигма-точек (сигма-точки - столбцы) в пространстве X на текущем такте, размерность [SizeX,k_sigma_points_].
- arma::mat [x_pred_sigma_points_](#)
Матрица сигма-точек (сигма-точки - столбцы) в пространстве X, экстраполированный на текущий такт, размерность [SizeX,k_sigma_points_].
- arma::mat [y_pred_sigma_points_](#)
Матрица сигма-точек (сигма-точки - столбцы) в пространстве Y, экстраполированный на текущий такт, размерность [SizeX,k_sigma_points_].
- arma::mat [dXcal_](#)
Матрица X-каллиграфическое (матрица сигма-точек - столбцов)
- arma::mat [dYcal_](#)
Матрица Y-каллиграфическое (матрица сигма-точек - столбцов)
- arma::mat [P_xy_](#)
Матрица кросс-ковариации векторов X и Y, размерность [SizeX * SizeY].
- arma::mat [sqrt_P_chol_](#)
Корень из матрицы P.
- double [alpha_](#)
Параметр разброса сигма-точек ($\alpha = 10^{-3}$ типичная рекомендация)
- double [kappa_](#)
Параметр разброса сигма-точек ($\kappa = 3 - \text{SizeX}$ типичная рекомендация)
- double [beta_](#)
Параметр разброса сигма-точек ($\beta = 2$ - нормальное, 0 - нет сведений о распределении)
- double [lambda_](#)
Автоматически вычисляемый параметр, равный $(\alpha * \alpha) * (\text{SizeX} + \kappa) - \text{SizeX}$.
- double [w0_](#)
Параметр разброса сигма-точек (0..1)
- double [gamma_](#)
Автоматически вычисляемый (в методах SetDesignParameters*) параметр (множитель при корне из P при создании сигма-точек)
- std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> [weightedSumStateSigmas_](#)
Вычисление взвешенной суммы сигма-точек пространства X
- std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> [weightedSumMeasurementSigmas_](#)
Вычисление взвешенной суммы сигма-точек пространства Y.

8.15.1 Подробное описание

```
template<size_t SizeX, size_t SizeY>
class KalmanFilters::CKalmanUKF< SizeX, SizeY >
```

Шаблонный класс сигма-точечного (ансцентного) фильтра Калмана, СТФК (АФК) (Unscented Kalman Filter, UKF)

Источники:

- [1] A New Extension of the Kalman Filter to Nonlinear Systems, Simon J. Julier, Jeffrey K. Uhlmann, The Robotics Research Group, Department of Engineering Science, The University of Oxford, 1997
- [2] Julier S.J., Uhlmann J.K. Unscented filtering and nonlinear estimation // Proc. of the IEEE, 2004, №3.Р.401-422
- [3] The Unscented Kalman Filter for Nonlinear Estimation, Eric A. Wan and Rudolph van der Merwe Oregon Graduate Institute of Science & Technology 20000 NW Walker Rd, Beaverton, Oregon 97006, 2000 ericwan[at]ece.ogi.edu, rvdmerwe[at]ece.ogi.edu
- [4] Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models, Rudolph van der Merwe & Eric Wan, OGI School of Science & Engineering Oregon Health & Science University Beaverton, Oregon, 97006, USA, 2003 {rvdmerwe,ericwan}[at]ece.ogi.edu
- [5] THE SQUARE-ROOT UNSCENTED KALMAN FILTER FOR STATE AND PARAMETER-ESTIMATION, Rudolph van der Merwe and Eric A. Wan, Oregon Graduate Institute of Science and Technology 20000 NW Walker Road, Beaverton, Oregon 97006, USA rvdmerwe,ericwan[at]ece.ogi.edu
- [6] Sebastian Bitzer, Technische Universität Dresden, <https://github.com/sbitzer/UKF-exposed/blob/master/UKF.pdf>

Внимание

Внимание! В реализации UKF вес нулевой сигма-точки W_{cov} не может быть отрицательным! (W_{mean} - может)

Фильтр построен по классическому его варианту НИЖНЕтреугольного разложения Холецкого!

Параметры шаблона

| | |
|-------|--|
| SizeX | - размерность пространства состояния X |
| SizeY | - размерность пространства измерений Y |

8.15.2 Конструктор(ы)

8.15.2.1 CKalmanUKF()

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanUKF< SizeX, SizeY >::CKalmanUKF ( ) [inline]
Конструктор по умолчанию
```

8.15.3 Методы

8.15.3.1 Correction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanUKF< SizeX, SizeY >::Correction (
    const arma::vec & Y_msd ) [inline], [virtual]
```

Коррекция

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

Переопределяет метод предка [KalmanFilters::CKalmanEKF< SizeX, SizeY >](#).

Переопределяется в [KalmanFilters::CKalmanEUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >](#), [KalmanFilters::CKalmanSRUKF< SizeX, SizeY >](#) и [KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >](#).

8.15.3.2 CorrectionUKF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanUKF< SizeX, SizeY >::CorrectionUKF (
    const arma::vec & Y_msd ) [inline], [protected]
```

Коррекция UKF.

Аргументы

| | |
|-------|---|
| Y_msd | - вектор измерений, по которым производится коррекция |
|-------|---|

8.15.3.3 Prediction()

```
template<size_t SizeX, size_t SizeY>
virtual void KalmanFilters::CKalmanUKF< SizeX, SizeY >::Prediction (
    double dt ) [inline], [virtual]
```

Прогноз

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

Переопределяет метод предка [KalmanFilters::CKalmanEKF< SizeX, SizeY >](#).

Переопределяется в [KalmanFilters::CKalmanEUKF< SizeX, SizeY >](#), [KalmanFilters::CKalmanSREUKF< SizeX, SizeY >](#) и [KalmanFilters::CKalmanSRUKF< SizeX, SizeY >](#).

8.15.3.4 PredictionUKF()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanUKF< SizeX, SizeY >::PredictionUKF (
    double dt ) [inline], [protected]
```

Прогноз UKF.

Аргументы

| | |
|----|-----------------------|
| dt | - Время прогноза, [с] |
|----|-----------------------|

8.15.3.5 SetObservationJacobianH()

```
template<size_t SizeX, size_t SizeY>
void KalmanFilters::CKalmanUKF< SizeX, SizeY >::SetObservationJacobianH ( ) [inline], [protected]
```

Запрет доступа

8.15.3.6 SetStateTransitionJacobianF()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanUKF< SizeX, SizeY >::SetStateTransitionJacobianF ( ) [inline], [protected]
```

Запрет доступа

8.15.3.7 SetupDesignParametersCDKF()

```
template<size_t SizeX, size_t SizeY>
```

```
virtual void KalmanFilters::CKalmanUKF< SizeX, SizeY >::SetupDesignParametersCDKF (
    double h2 ) [inline], [virtual]
```

Установка параметра фильтра по рекомендации Central Difference Kalman Filter (CDKF)

Смотри [4]

Аргументы

| | |
|----|---|
| h2 | - параметр разброса сигма-точек ($h^2 = 3$ типичная рекомендация для гауссовых шумов, [3]) |
|----|---|

Переопределяется в [KalmanFilters::CKalmanSRUKF](#)< SizeX, SizeY >.

8.15.3.8 SetupDesignParametersMeanSet()

```
template<size_t SizeX, size_t SizeY>
```

```
virtual void KalmanFilters::CKalmanUKF< SizeX, SizeY >::SetupDesignParametersMeanSet (
    double w0 ) [inline], [virtual]
```

Установка параметра w0 ансцентного фильтра (MeanSet)

При выборе $w0 = [0...1]$ обеспечиваются положительные веса. Смотри [1], [2]

Внимание

Нельзя выбирать w0 так, чтобы нулевой вес был > 0 , а остальные меньше нуля. Наоборот - МОЖНО, т.е. нулевой вес может быть отрицательным.

Аргументы

| | |
|----|---|
| w0 | - параметр разброса сигма точек ($w0 = [0...1]$ типичная рекомендация для положительных весов) |
|----|---|

Переопределяется в [KalmanFilters::CKalmanSRUKF](#)< SizeX, SizeY >.

8.15.3.9 SetupDesignParametersScaledSet()

```
template<size_t SizeX, size_t SizeY>
```

```
virtual void KalmanFilters::CKalmanUKF< SizeX, SizeY >::SetupDesignParametersScaledSet (
    double alpha,
    double beta,
    double kappa ) [inline], [virtual]
```

Установка параметров масштабируемого ансцентного преобразования (Scaled UT)

Смотри [2], [5], [6]

Аргументы

| | |
|-------|--|
| alpha | - параметр разброса сигма-точек ($\alpha = 10^{-3}$ - типичная рекомендация по van der Merwe) |
| beta | - параметр, отвечающий за характер распределения ($\beta = 2$ - нормальное) |

Аргументы

| | |
|-------|--|
| карра | - параметр, отвечающий за разброс сигма-точек (карра = 0 или (3 - SizeX) - типичная рекомендация по van der Merwe) |
|-------|--|

Переопределяется в [KalmanFilters::CKalmanSRUKF< SizeX, SizeY >](#).

8.15.3.10 SetWeightedSumMeasurementSigmas()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanUKF< SizeX, SizeY >::SetWeightedSumMeasurementSigmas (
    std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> weightedSum←
    MeasurementSigmas ) [inline]
```

Установка функции вычисления взвешенной суммы сигма-точек пространства Y.

Аргументы

| | |
|------------------------------|--|
| weightedSumMeasurementSigmas | - Функция вычисления взвешенной суммы сигма-точек пространства Y |
|------------------------------|--|

8.15.3.11 SetWeightedSumStateSigmas()

```
template<size_t SizeX, size_t SizeY>
```

```
void KalmanFilters::CKalmanUKF< SizeX, SizeY >::SetWeightedSumStateSigmas (
    std::function< arma::vec(const arma::vec &weights, const arma::mat &sigmaPoints)> weightedSumState←
    Sigmas ) [inline]
```

Установка функции вычисления взвешенной суммы сигма-точек пространства X

Аргументы

| | |
|------------------------|--|
| weightedSumStateSigmas | - Функция вычисления взвешенной суммы сигма-точек пространства X |
|------------------------|--|

8.15.4 Данные класса

8.15.4.1 alpha_

```
template<size_t SizeX, size_t SizeY>
```

```
double KalmanFilters::CKalmanUKF< SizeX, SizeY >::alpha_ [protected]
```

Параметр разброса сигма-точек (alpha = 10⁻³ типичная рекомендация)

8.15.4.2 beta_

```
template<size_t SizeX, size_t SizeY>
```

```
double KalmanFilters::CKalmanUKF< SizeX, SizeY >::beta_ [protected]
```

Параметр разброса сигма-точек (beta = 2 - нормальное, 0 - нет сведений о распределении)

8.15.4.3 dXcal_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanUKF< SizeX, SizeY >::dXcal_ [protected]
```

Матрица X-калибрагическое (матрица сигма-точек - столбцов)

8.15.4.4 dYcal_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanUKF< SizeX, SizeY >::dYcal_ [protected]
```

Матрица Y-калибрагическое (матрица сигма-точек - столбцов)

8.15.4.5 gamma_

```
template<size_t SizeX, size_t SizeY>
```

```
double KalmanFilters::CKalmanUKF< SizeX, SizeY >::gamma_ [protected]
```

Автоматически вычисляемый (в методах SetDesignParameters*) параметр (множитель при корне из Р при создании сигма-точек)

8.15.4.6 k_sigma_points_

```
template<size_t SizeX, size_t SizeY>
```

```
int KalmanFilters::CKalmanUKF< SizeX, SizeY >::k_sigma_points_ [protected]
```

Число сигма-точек

8.15.4.7 kappa_

```
template<size_t SizeX, size_t SizeY>
```

```
double KalmanFilters::CKalmanUKF< SizeX, SizeY >::kappa_ [protected]
```

Параметр разброса сигма-точек (kappa = 3 - SizeX типичная рекомендация)

8.15.4.8 lambda_

```
template<size_t SizeX, size_t SizeY>
```

```
double KalmanFilters::CKalmanUKF< SizeX, SizeY >::lambda_ [protected]
```

Автоматически вычисляемый параметр, равный $(\alpha * \alpha) * (SizeX + \kappa) - SizeX$;

8.15.4.9 P_xy_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanUKF< SizeX, SizeY >::P_xy_ [protected]
```

Матрица кросс-коварации векторов X и Y, размерность [SizeX * SizeY].

8.15.4.10 sqrt_P_chol_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanUKF< SizeX, SizeY >::sqrt_P_chol_ [protected]
```

Корень из матрицы P.

8.15.4.11 w0_

```
template<size_t SizeX, size_t SizeY>
```

```
double KalmanFilters::CKalmanUKF< SizeX, SizeY >::w0_ [protected]
```

Параметр разброса сигма-точек (0..1)

8.15.4.12 weightedSumMeasurementSigmas_

```
template<size_t SizeX, size_t SizeY>
```

```
std::function<arma::vec( const arma::vec &weights, const arma::mat &sigmaPoints )> KalmanFilters::CKalmanUKF<
SizeX, SizeY >::weightedSumMeasurementSigmas_ [protected]
```

Вычисление взвешенной суммы сигма-точек пространства Y.

8.15.4.13 weightedSumStateSigmas_

```
template<size_t SizeX, size_t SizeY>
```

```
std::function<arma::vec( const arma::vec &weights, const arma::mat &sigmaPoints )> KalmanFilters::CKalmanUKF<
SizeX, SizeY >::weightedSumStateSigmas_ [protected]
```

Вычисление взвешенной суммы сигма-точек пространства X

8.15.4.14 weights_covariance_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::vec KalmanFilters::CKalmanUKF< SizeX, SizeY >::weights_covariance_ [protected]
```

Весы ковариации

8.15.4.15 weights_mean_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::vec KalmanFilters::CKalmanUKF< SizeX, SizeY >::weights_mean_ [protected]
```

Весы среднего

8.15.4.16 x_est_sigma_points_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanUKF< SizeX, SizeY >::x_est_sigma_points_ [protected]
```

Матрица сигма-точек (сигма-точки - столбцы) в пространстве X на текущем такте, размерность [SizeX,k_sigma_points_].

8.15.4.17 x_pred_sigma_points_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanUKF< SizeX, SizeY >::x_pred_sigma_points_ [protected]
```

Матрица сигма-точек (сигма-точки - столбцы) в пространстве X, экстраполированный на текущий такт, размерность [SizeX,k_sigma_points_].

8.15.4.18 y_pred_sigma_points_

```
template<size_t SizeX, size_t SizeY>
```

```
arma::mat KalmanFilters::CKalmanUKF< SizeX, SizeY >::y_pred_sigma_points_ [protected]
```

Матрица сигма-точек (сигма-точки - столбцы) в пространстве Y, экстраполированный на текущий такт, размерность [SizeX,k_sigma_points_].

Объявления и описания членов класса находятся в файле:

- kalman_filter_unscented.h

Предметный указатель

| | |
|---|--|
| ~CKalmanEKF
KalmanFilters::CKalmanEKF< SizeX, SizeY
>, 32 | KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 42 |
| ~CKalmanLKF
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 42 | CKalmanSRCKF
KalmanFilters::CKalmanSRCKF< SizeX,
SizeY >, 55 |
| alpha_
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, 83 | CKalmanSRCKFB
KalmanFilters::CKalmanSRCKFB< SizeX,
SizeY >, 59 |
| beta_
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, 83 | CKalmanSRECKF
KalmanFilters::CKalmanSRECKF< SizeX,
SizeY >, 60 |
| CalculateInnovationCovarianceS
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 42 | CKalmanSRECKFB
KalmanFilters::CKalmanSRECKFB< SizeX,
SizeY >, 62 |
| checkBordersMeasurement_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 50 | CKalmanSREKF
KalmanFilters::CKalmanSREKF< SizeX,
SizeY >, 64 |
| checkBordersStateAfterCorrection_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 50 | CKalmanSREUKF
KalmanFilters::CKalmanSREUKF< SizeX,
SizeY >, 66 |
| checkBordersStateAfterPrediction_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 50 | CKalmanSREUKFB
KalmanFilters::CKalmanSREUKFB< SizeX,
SizeY >, 68 |
| checkDeltaMeasurement_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 50 | CKalmanSRUKF
KalmanFilters::CKalmanSRUKF< SizeX,
SizeY >, 71 |
| checkDeltaState_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 50 | CKalmanSRUKFB
KalmanFilters::CKalmanSRUKFB< SizeX,
SizeY >, 77 |
| checkMatrixDiagPositive
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 43 | CKalmanUKF
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, 80 |
| CKalmanCKF
KalmanFilters::CKalmanCKF< SizeX, SizeY
>, 23 | Correction
KalmanFilters::CKalmanCKF< SizeX, SizeY
>, 23 |
| CKalmanECKF
KalmanFilters::CKalmanECKF< SizeX,
SizeY >, 29 | KalmanFilters::CKalmanECKF< SizeX,
SizeY >, 29 |
| CKalmanEKF
KalmanFilters::CKalmanEKF< SizeX, SizeY
>, 32 | KalmanFilters::CKalmanEKF< SizeX, SizeY
>, 32 |
| CKalmanEUKF
KalmanFilters::CKalmanEUKF< SizeX,
SizeY >, 37 | KalmanFilters::CKalmanEUKF< SizeX,
SizeY >, 38 |
| | KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 43 |
| | KalmanFilters::CKalmanSRCKF< SizeX,
SizeY >, 55 |
| | KalmanFilters::CKalmanSRCKFB< SizeX,
SizeY >, 59 |
| | KalmanFilters::CKalmanSRECKF< SizeX,
SizeY >, 60 |

- KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >, [62](#)
- KalmanFilters::CKalmanSREKF< SizeX, SizeY >, [64](#)
- KalmanFilters::CKalmanSREUKF< SizeX, SizeY >, [66](#)
- KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >, [69](#)
- KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, [71](#)
- KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >, [77](#)
- KalmanFilters::CKalmanUKF< SizeX, SizeY >, [80](#)
- CorrectionCKF
KalmanFilters::CKalmanCKF< SizeX, SizeY >, [23](#)
- CorrectionSRCKF
KalmanFilters::CKalmanSRCKF< SizeX, SizeY >, [55](#)
- CorrectionSRUKF
KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, [72](#)
- CorrectionUKF
KalmanFilters::CKalmanUKF< SizeX, SizeY >, [81](#)
- createSignMatrices
KalmanFilters::CKalmanSREUKF< SizeX, SizeY >, [67](#)
- KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, [72](#)
- createSignMatricesBlock
KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >, [69](#)
- KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >, [77](#)
- DeltaY_
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [50](#)
- deltaY_isSet
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [51](#)
- dXcal_
KalmanFilters::CKalmanCKF< SizeX, SizeY >, [25](#)
- KalmanFilters::CKalmanUKF< SizeX, SizeY >, [83](#)
- dYcal_
KalmanFilters::CKalmanCKF< SizeX, SizeY >, [25](#)
- KalmanFilters::CKalmanUKF< SizeX, SizeY >, [84](#)
- F_
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [51](#)
- fixMatrixMainDiagonalSymmetry
- KalmanFilters::CKalmanLKF< SizeX, SizeY >, [43](#)
- gamma_
KalmanFilters::CKalmanCKF< SizeX, SizeY >, [25](#)
- KalmanFilters::CKalmanUKF< SizeX, SizeY >, [84](#)
- GetDeltaY
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [44](#)
- GetEstimatedCovarianceMatrixP
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [44](#)
- GetEstimatedVectorX
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [44](#)
- GetEstimatedVectorY
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [44](#)
- GetInnovationCovarianceMatrixS
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [44](#)
- GetKalmanGainMatrixK
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [44](#)
- GetSizeX
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [45](#)
- GetSizeY
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [45](#)
- GetVersion
KalmanFilters, [20](#)
- H_
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [51](#)
- I_
KalmanFilters::CKalmanLKF< SizeX, SizeY >, [51](#)
- J_
KalmanFilters::CKalmanSREUKF< SizeX, SizeY >, [67](#)
- KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, [74](#)
- Jcorrect_
KalmanFilters::CKalmanSREUKF< SizeX, SizeY >, [67](#)
- KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, [74](#)
- JcorrectBlock_
KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >, [69](#)
- KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >, [77](#)
- Jpredict_

- KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, 74
- K_
 - KalmanFilters::CKalmanLKF< SizeX, SizeY >, 51
- k_sigma_points_
 - KalmanFilters::CKalmanCKF< SizeX, SizeY >, 25
 - KalmanFilters::CKalmanUKF< SizeX, SizeY >, 84
- Kalman_filters, 17
- KalmanFilters, 19
 - GetVersion, 20
- KalmanFilters::CKalmanCKF< SizeX, SizeY >, 21
 - CKalmanCKF, 23
 - Correction, 23
 - CorrectionCKF, 23
 - dXcal_, 25
 - dYcal_, 25
 - gamma_, 25
 - k_sigma_points_, 25
 - P_xy_, 25
 - Prediction, 24
 - PredictionCKF, 24
 - SetObservationJacobianH, 24
 - SetStateTransitionJacobianF, 24
 - SetupDesignParametersCubatureBaseSet, 24
 - SetWeightedSumMeasurementSigmas, 24
 - SetWeightedSumStateSigmas, 25
 - sqrt_P_chol_, 26
 - weightedSumMeasurementSigmas_, 26
 - weightedSumStateSigmas_, 26
 - weights_covariance_, 26
 - weights_mean_, 26
 - x_est_sigma_points_, 26
 - x_pred_sigma_points_, 26
 - y_pred_sigma_points_, 26
- KalmanFilters::CKalmanECKF< SizeX, SizeY >, 27
 - CKalmanECKF, 29
 - Correction, 29
 - Prediction, 29
 - SetObservationJacobianH, 29
 - SetStateTransitionJacobianF, 30
- KalmanFilters::CKalmanEKF< SizeX, SizeY >, 30
 - ~CKalmanEKF, 32
 - CKalmanEKF, 32
 - Correction, 32
 - observationJacobianH_, 34
 - observationModel_, 34
 - Prediction, 32
 - PredictionEKF, 32
 - SetObservationJacobianH, 33
 - SetObservationMatrixH, 33
 - SetObservationModel, 33
 - SetStateTransitionJacobianF, 33
 - SetStateTransitionJacobianLinearF, 33
 - SetStateTransitionMatrixF, 33
 - SetStateTransitionModel, 34
 - stateTransitionJacobianF_, 34
 - stateTransitionModel_, 35
- KalmanFilters::CKalmanEUKF< SizeX, SizeY >, 35
 - CKalmanEUKF, 37
 - Correction, 38
 - Prediction, 38
 - SetObservationJacobianH, 38
 - SetStateTransitionJacobianF, 38
- KalmanFilters::CKalmanLKF< SizeX, SizeY >, 39
 - ~CKalmanLKF, 42
 - CalculateInnovationCovarianceS, 42
 - checkBordersMeasurement_, 50
 - checkBordersStateAfterCorrection_, 50
 - checkBordersStateAfterPrediction_, 50
 - checkDeltaMeasurement_, 50
 - checkDeltaState_, 50
 - checkMatrixDiagPositive, 43
 - CKalmanLKF, 42
 - Correction, 43
 - DeltaY_, 50
 - deltaY_isSet, 51
 - F_, 51
 - fixMatrixMainDiagonalSymmetry, 43
 - GetDeltaY, 44
 - GetEstimatedCovarianceMatrixP, 44
 - GetEstimatedVectorX, 44
 - GetEstimatedVectorY, 44
 - GetInnovationCovarianceMatrixS, 44
 - GetKalmanGainMatrixK, 44
 - GetSizeX, 45
 - GetSizeY, 45
 - H_, 51
 - I_, 51
 - K_, 51
 - operator=, 45
 - P_, 51
 - Prediction, 45
 - prediction_isDone, 51
 - Q_, 51
 - R_, 52
 - S_, 52
 - SetCheckBordersMeasurement, 46
 - SetCheckBordersStateAfterCorrection, 46
 - SetCheckBordersStateAfterPrediction, 46
 - SetCheckDeltaMeasurement, 46
 - SetCheckDeltaState, 46
 - SetDeltaY, 47
 - SetEstimateCovarianceMatrixP, 47
 - SetEstimateCovarianceMatrixPdiag, 47
 - SetEstimatedVectorX, 47
 - SetEstimatedVectorY, 48
 - SetMeasuredVectorY, 48
 - SetObservationCovarianceMatrixR, 48
 - SetObservationCovarianceMatrixRdiag, 48
 - SetObservationMatrixH, 49
 - SetProcessCovarianceMatrixQ, 49

- SetProcessCovarianceMatrixQdiag, 49
- SetStateTransitionJacobianLinearF, 49
- SetStateTransitionMatrixF, 50
- SizeX_, 52
- SizeY_, 52
- stateTransitionJacobianLinearF_, 52
- X_est_, 52
- X_pred_, 52
- Y_est_, 53
- Y_msd_, 53
- Y_msd_isSet, 53
- Y_pred_, 53
- KalmanFilters::CKalmanSRCKF< SizeX, SizeY >, 53
 - CKalmanSRCKF, 55
 - Correction, 55
 - CorrectionSRCKF, 55
 - Prediction, 55
 - SetObservationJacobianH, 56
 - SetStateTransitionJacobianF, 56
- KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >, 56
 - CKalmanSRCKFB, 59
 - Correction, 59
- KalmanFilters::CKalmanSRECKF< SizeX, SizeY >, 59
 - CKalmanSRECKF, 60
 - Correction, 60
 - Prediction, 61
- KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >, 61
 - CKalmanSRECKFB, 62
 - Correction, 62
- KalmanFilters::CKalmanSREKF< SizeX, SizeY >, 62
 - CKalmanSREKF, 64
 - Correction, 64
 - Prediction, 64
 - PredictionSREKF, 64
- KalmanFilters::CKalmanSREUKF< SizeX, SizeY >, 65
 - CKalmanSREUKF, 66
 - Correction, 66
 - createSignMatrices, 67
 - J_, 67
 - Jcorrect_, 67
 - Prediction, 67
- KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >, 67
 - CKalmanSREUKFB, 68
 - Correction, 69
 - createSignMatricesBlock, 69
 - JcorrectBlock_, 69
- KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, 69
 - CKalmanSRUKF, 71
 - Correction, 71
 - CorrectionSRUKF, 72
 - createSignMatrices, 72
 - J_, 74
 - Jcorrect_, 74
 - Jpredict_, 74
 - negativeZeroCovWeight_, 74
 - Prediction, 72
 - SetObservationJacobianH, 72
 - SetStateTransitionJacobianF, 72
 - SetupDesignParametersCDKF, 73
 - SetupDesignParametersMeanSet, 73
 - SetupDesignParametersScaledSet, 73
- KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >, 74
 - CKalmanSRUKFB, 77
 - Correction, 77
 - createSignMatricesBlock, 77
 - JcorrectBlock_, 77
- KalmanFilters::CKalmanUKF< SizeX, SizeY >, 78
 - alpha_, 83
 - beta_, 83
 - CKalmanUKF, 80
 - Correction, 80
 - CorrectionUKF, 81
 - dXcal_, 83
 - dYcal_, 84
 - gamma_, 84
 - k_sigma_points_, 84
 - kappa_, 84
 - lambda_, 84
 - P_xy_, 84
 - Prediction, 81
 - PredictionUKF, 81
 - SetObservationJacobianH, 81
 - SetStateTransitionJacobianF, 81
 - SetupDesignParametersCDKF, 82
 - SetupDesignParametersMeanSet, 82
 - SetupDesignParametersScaledSet, 82
 - SetWeightedSumMeasurementSigmas, 83
 - SetWeightedSumStateSigmas, 83
 - sqrt_P_chol_, 84
 - w0_, 84
 - weightedSumMeasurementSigmas_, 84
 - weightedSumStateSigmas_, 85
 - weights_covariance_, 85
 - weights_mean_, 85
 - x_est_sigma_points_, 85
 - x_pred_sigma_points_, 85
 - y_pred_sigma_points_, 85
- kappa_
 - KalmanFilters::CKalmanUKF< SizeX, SizeY >, 84
- lambda_
 - KalmanFilters::CKalmanUKF< SizeX, SizeY >, 84
- negativeZeroCovWeight_
 - KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, 74

- observationJacobianH_
KalmanFilters::CKalmanEKF< SizeX, SizeY
>, 34
- observationModel_
KalmanFilters::CKalmanEKF< SizeX, SizeY
>, 34
- operator=
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 45
- P_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 51
- P_xy_
KalmanFilters::CKalmanCKF< SizeX, SizeY
>, 25
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, 84
- Prediction
KalmanFilters::CKalmanCKF< SizeX, SizeY
>, 24
KalmanFilters::CKalmanECKF< SizeX,
SizeY >, 29
KalmanFilters::CKalmanEKF< SizeX, SizeY
>, 32
KalmanFilters::CKalmanEUKF< SizeX,
SizeY >, 38
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 45
KalmanFilters::CKalmanSRCKF< SizeX,
SizeY >, 55
KalmanFilters::CKalmanSRECKF< SizeX,
SizeY >, 61
KalmanFilters::CKalmanSREKF< SizeX,
SizeY >, 64
KalmanFilters::CKalmanSREUKF< SizeX,
SizeY >, 67
KalmanFilters::CKalmanSRUKF< SizeX,
SizeY >, 72
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, 81
- prediction_isDone
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 51
- PredictionCKF
KalmanFilters::CKalmanCKF< SizeX, SizeY
>, 24
- PredictionEKF
KalmanFilters::CKalmanEKF< SizeX, SizeY
>, 32
- PredictionSREKF
KalmanFilters::CKalmanSREKF< SizeX,
SizeY >, 64
- PredictionUKF
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, 81
- Q_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 51
- R_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 52
- S_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 52
- SetCheckBordersMeasurement
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 46
- SetCheckBordersStateAfterCorrection
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 46
- SetCheckBordersStateAfterPrediction
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 46
- SetCheckDeltaMeasurement
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 46
- SetCheckDeltaState
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 46
- SetDeltaY
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 47
- SetEstimateCovarianceMatrixP
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 47
- SetEstimateCovarianceMatrixPdiag
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 47
- SetEstimatedVectorX
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 47
- SetEstimatedVectorY
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 48
- SetMeasuredVectorY
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 48
- SetObservationCovarianceMatrixR
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 48
- SetObservationCovarianceMatrixRdiag
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 48
- SetObservationJacobianH
KalmanFilters::CKalmanCKF< SizeX, SizeY
>, 24
KalmanFilters::CKalmanECKF< SizeX,
SizeY >, 29
KalmanFilters::CKalmanEKF< SizeX, SizeY
>, 33
KalmanFilters::CKalmanEUKF< SizeX,
SizeY >, 38

- KalmanFilters::CKalmanSRCKF< SizeX, SizeY >, 73
 KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, 72
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 81
 SetObservationMatrixH
 KalmanFilters::CKalmanEKF< SizeX, SizeY >, 33
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, 49
 SetObservationModel
 KalmanFilters::CKalmanEKF< SizeX, SizeY >, 33
 SetProcessCovarianceMatrixQ
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, 49
 SetProcessCovarianceMatrixQdiag
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, 49
 SetStateTransitionJacobianF
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, 24
 KalmanFilters::CKalmanECKF< SizeX, SizeY >, 30
 KalmanFilters::CKalmanEKF< SizeX, SizeY >, 33
 KalmanFilters::CKalmanEUKF< SizeX, SizeY >, 38
 KalmanFilters::CKalmanSRCKF< SizeX, SizeY >, 56
 KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, 72
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 81
 SetStateTransitionJacobianLinearF
 KalmanFilters::CKalmanEKF< SizeX, SizeY >, 33
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, 49
 SetStateTransitionMatrixF
 KalmanFilters::CKalmanEKF< SizeX, SizeY >, 33
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, 50
 SetStateTransitionModel
 KalmanFilters::CKalmanEKF< SizeX, SizeY >, 34
 SetupDesignParametersCDKF
 KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, 73
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 82
 SetupDesignParametersCubatureBaseSet
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, 24
 SetupDesignParametersMeanSet
 KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, 82
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 82
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 83
 SetWeightedSumMeasurementSigmas
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, 24
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 83
 SetWeightedSumStateSigmas
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, 25
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 83
 SizeX_
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, 52
 SizeY_
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, 52
 sqrt_P_chol_
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, 26
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 84
 stateTransitionJacobianF_
 KalmanFilters::CKalmanEKF< SizeX, SizeY >, 34
 stateTransitionJacobianLinearF_
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, 52
 stateTransitionModel_
 KalmanFilters::CKalmanEKF< SizeX, SizeY >, 35
 w0_
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 84
 weightedSumMeasurementSigmas_
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, 26
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 84
 weightedSumStateSigmas_
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, 26
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 85
 weights_covariance_
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, 26
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, 85
 weights_mean_

KalmanFilters::CKalmanCKF< SizeX, SizeY
>, [26](#)
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, [85](#)

X_est_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, [52](#)

x_est_sigma_points_
KalmanFilters::CKalmanCKF< SizeX, SizeY
>, [26](#)
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, [85](#)

X_pred_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, [52](#)

x_pred_sigma_points_
KalmanFilters::CKalmanCKF< SizeX, SizeY
>, [26](#)
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, [85](#)

Y_est_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, [53](#)

Y_msd_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, [53](#)

Y_msd_isSet
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, [53](#)

Y_pred_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, [53](#)

y_pred_sigma_points_
KalmanFilters::CKalmanCKF< SizeX, SizeY
>, [26](#)
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, [85](#)