

Фильтры Калмана

Программная документация

1 Результаты моделирования фильтров Калмана	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	24
8.1.4 Документация по друзьям класса и функциям, относящимся к классу	26
8.1.5 Данные класса	27
8.2 Шаблон класса KalmanFilters::CKalmanECKF< SizeX, SizeY >	28
8.2.1 Подробное описание	31
8.2.2 Конструктор(ы)	31
8.2.3 Методы	32
8.2.4 Документация по друзьям класса и функциям, относящимся к классу	33
8.3 Шаблон класса KalmanFilters::CKalmanEKF< SizeX, SizeY >	34
8.3.1 Подробное описание	35
8.3.2 Конструктор(ы)	36
8.3.3 Методы	36
8.3.4 Документация по друзьям класса и функциям, относящимся к классу	39
8.3.5 Данные класса	39
8.4 Шаблон класса KalmanFilters::CKalmanEUKF< SizeX, SizeY >	41
8.4.1 Подробное описание	43
8.4.2 Конструктор(ы)	43
8.4.3 Методы	44
8.4.4 Документация по друзьям класса и функциям, относящимся к классу	45

8.5	Шаблон класса <code>KalmanFilters::CKalmanLKF< SizeX, SizeY ></code>	46
8.5.1	Подробное описание	49
8.5.2	Конструктор(ы)	49
8.5.3	Методы	50
8.5.4	Документация по друзьям класса и функциям, относящимся к классу	58
8.5.5	Данные класса	58
8.6	Шаблон класса <code>KalmanFilters::CKalmanSRCKF< SizeX, SizeY ></code>	61
8.6.1	Подробное описание	63
8.6.2	Конструктор(ы)	63
8.6.3	Методы	64
8.6.4	Документация по друзьям класса и функциям, относящимся к классу	66
8.7	Шаблон класса <code>KalmanFilters::CKalmanSRCKFB< SizeX, SizeY ></code>	66
8.7.1	Подробное описание	68
8.7.2	Конструктор(ы)	69
8.7.3	Методы	69
8.8	Шаблон класса <code>KalmanFilters::CKalmanSRECKF< SizeX, SizeY ></code>	69
8.8.1	Подробное описание	70
8.8.2	Конструктор(ы)	71
8.8.3	Методы	71
8.8.4	Документация по друзьям класса и функциям, относящимся к классу	72
8.9	Шаблон класса <code>KalmanFilters::CKalmanSRECKFB< SizeX, SizeY ></code>	73
8.9.1	Подробное описание	74
8.9.2	Конструктор(ы)	74
8.9.3	Методы	75
8.9.4	Документация по друзьям класса и функциям, относящимся к классу	76
8.10	Шаблон класса <code>KalmanFilters::CKalmanSREKF< SizeX, SizeY ></code>	76
8.10.1	Подробное описание	78
8.10.2	Конструктор(ы)	78
8.10.3	Методы	79
8.10.4	Документация по друзьям класса и функциям, относящимся к классу	80
8.11	Шаблон класса <code>KalmanFilters::CKalmanSREUKF< SizeX, SizeY ></code>	80
8.11.1	Подробное описание	82
8.11.2	Конструктор(ы)	82
8.11.3	Методы	83
8.11.4	Документация по друзьям класса и функциям, относящимся к классу	84
8.11.5	Данные класса	84
8.12	Шаблон класса <code>KalmanFilters::CKalmanSREUKFB< SizeX, SizeY ></code>	85
8.12.1	Подробное описание	86
8.12.2	Конструктор(ы)	86
8.12.3	Методы	87
8.12.4	Документация по друзьям класса и функциям, относящимся к классу	88
8.12.5	Данные класса	88
8.13	Шаблон класса <code>KalmanFilters::CKalmanSRUKF< SizeX, SizeY ></code>	89

8.13.1 Подробное описание	90
8.13.2 Конструктор(ы)	91
8.13.3 Методы	92
8.13.4 Документация по друзьям класса и функциям, относящимся к классу	94
8.13.5 Данные класса	95
8.14 Шаблон класса KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >	95
8.14.1 Подробное описание	98
8.14.2 Конструктор(ы)	98
8.14.3 Методы	99
8.14.4 Документация по друзьям класса и функциям, относящимся к классу	100
8.14.5 Данные класса	100
8.15 Шаблон класса KalmanFilters::CKalmanUKF< SizeX, SizeY >	100
8.15.1 Подробное описание	103
8.15.2 Конструктор(ы)	103
8.15.3 Методы	104
8.15.4 Документация по друзьям класса и функциям, относящимся к классу	107
8.15.5 Данные класса	108
Предметный указатель	111

Раздел 1

Результаты моделирования фильтров Калмана

-
- 1) ЛФК - Линейный Фильтр Калмана (LKF - Linear Kalman Filter);
 - 2) РФК - Расширенный Фильтр Калмана (EKF - 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 (CKF - 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 (ECKF - 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 $\alpha=1.0\dots0.35$ (от светлого к темному)

<? <?

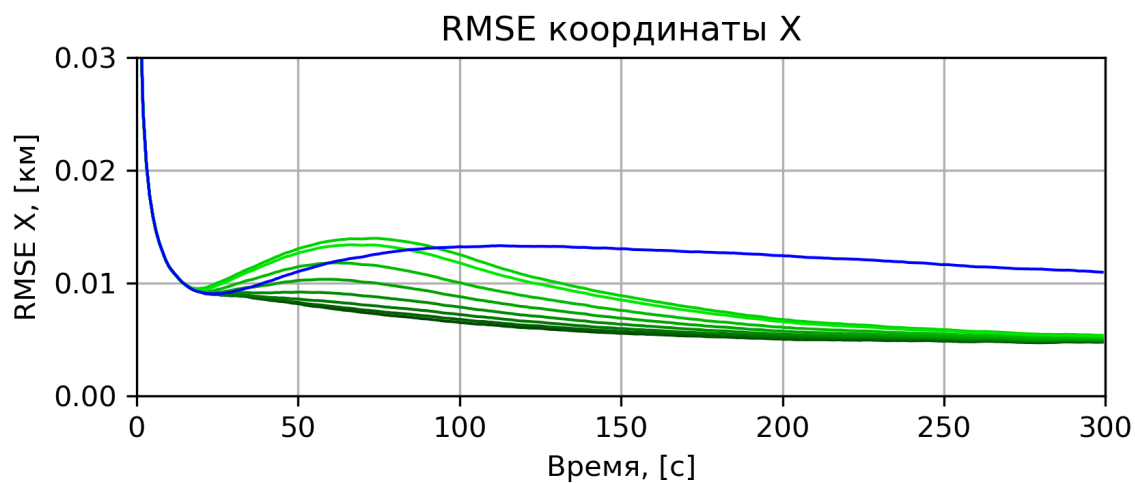


Рис. 1.1 ?>

<? <?

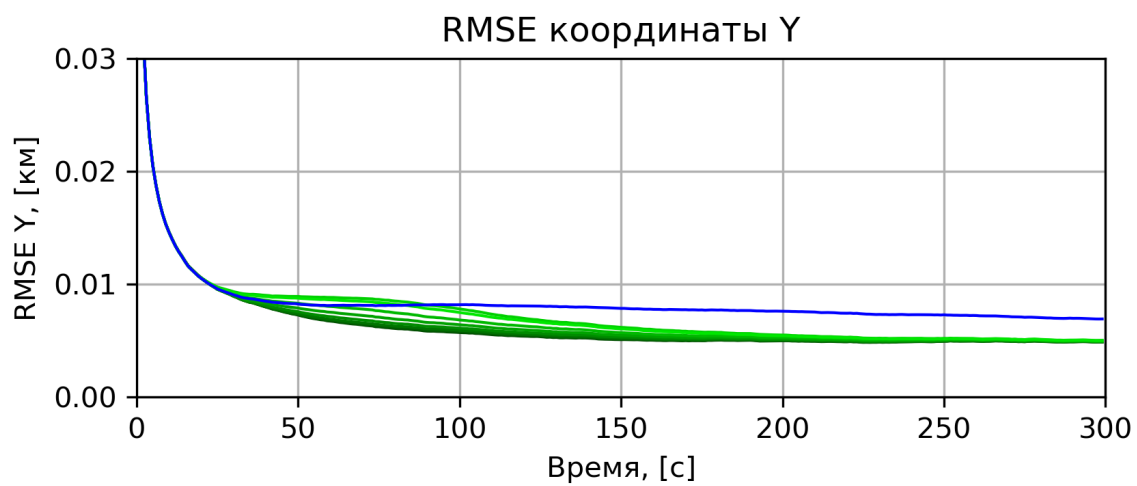


Рис. 1.2 ?>

<? <?

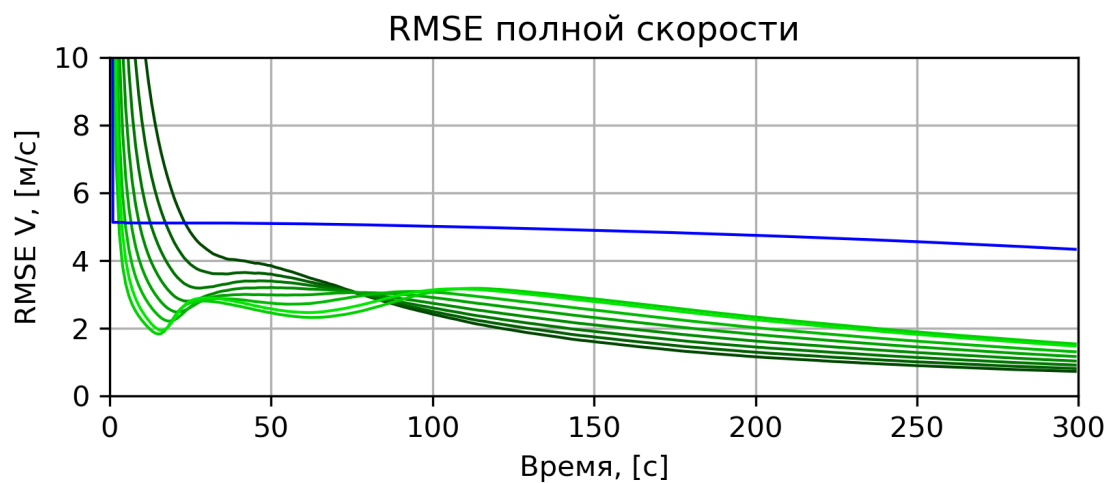


Рис. 1.3 ?>

<? <?

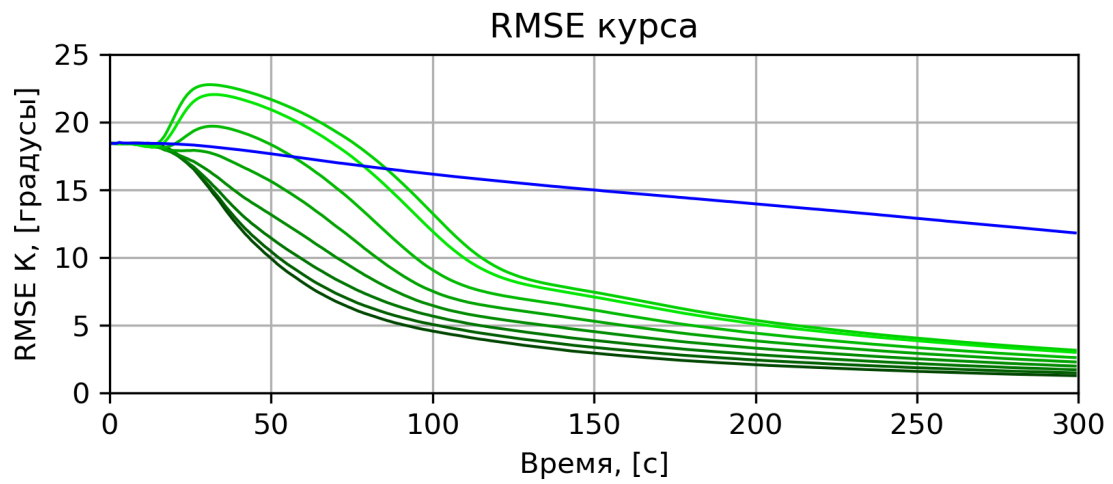


Рис. 1.4 ?>

<? <?

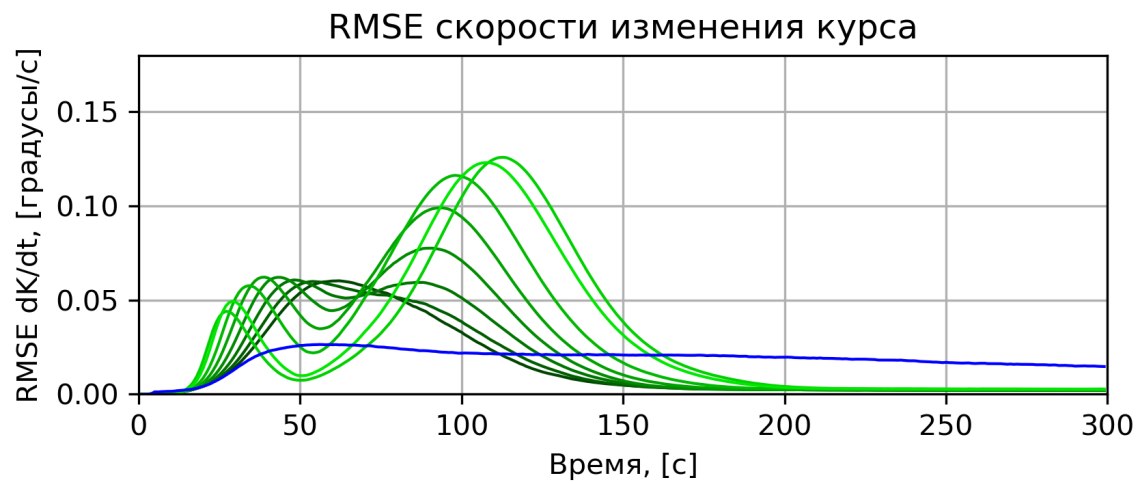


Рис. 1.5 ?>

Рис.2 - Сравнение RMSE вектора состояния

<? <?

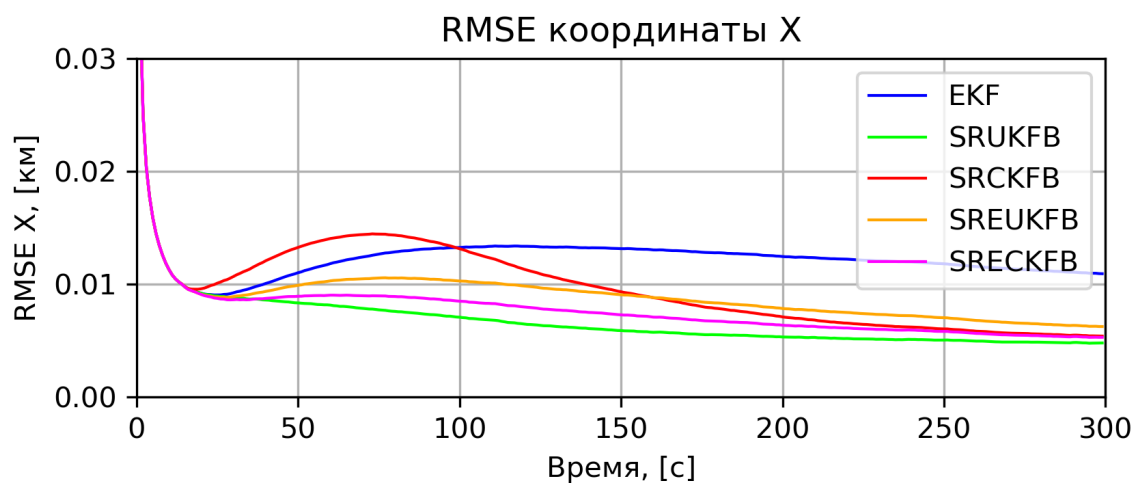


Рис. 1.6 ?>

<? <?

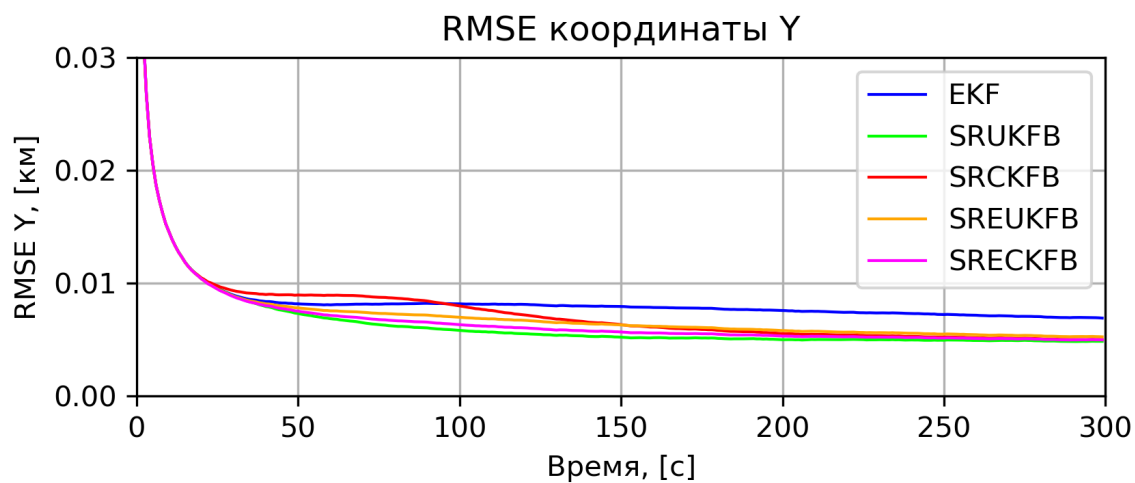


Рис. 1.7 ?>

<? <?

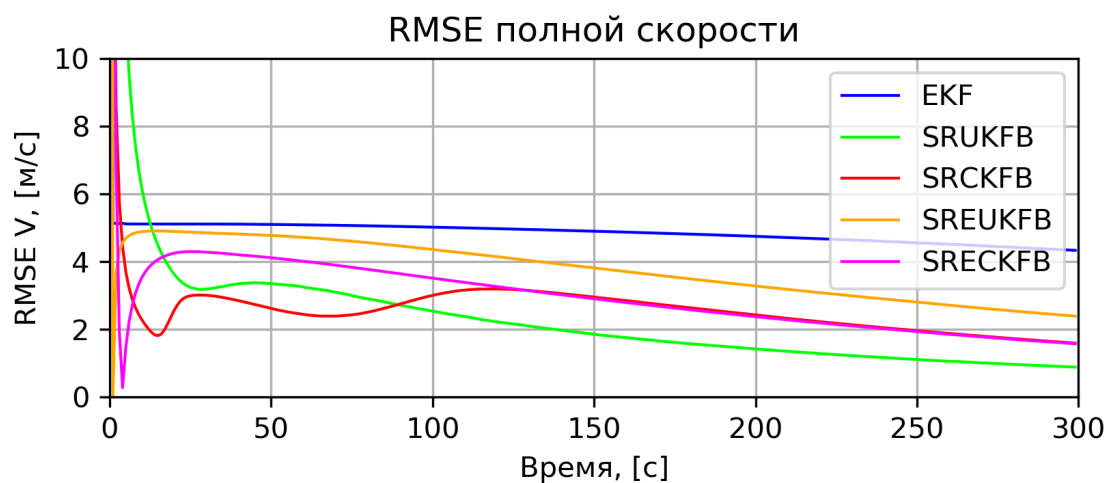


Рис. 1.8 ?>

<? <?

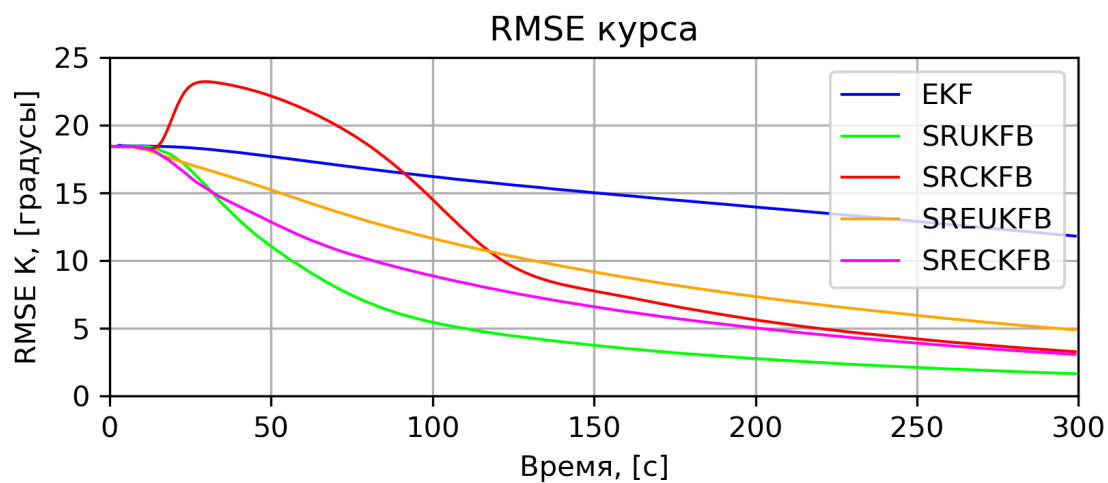


Рис. 1.9 ?>

<? <?

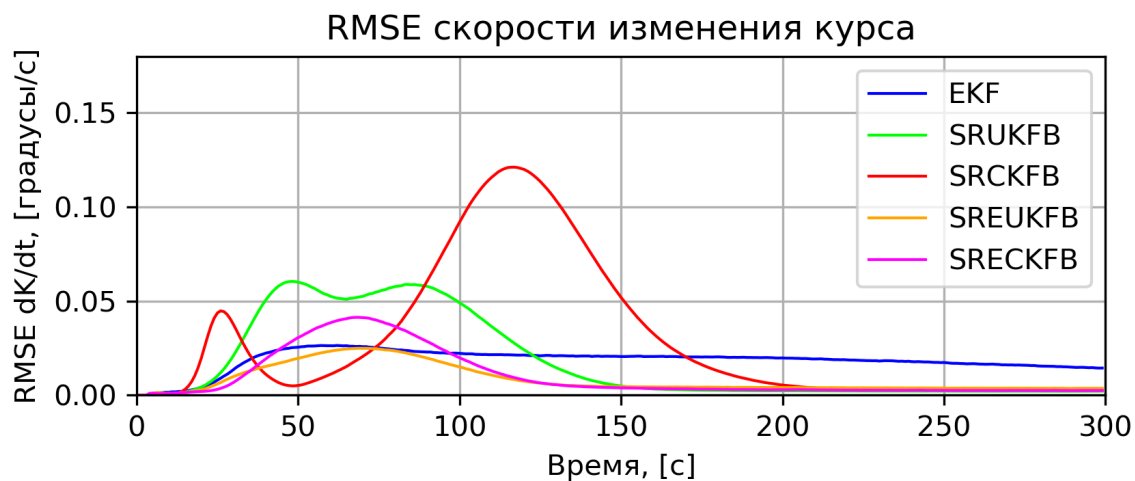


Рис. 1.10 ?>

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

<? <?

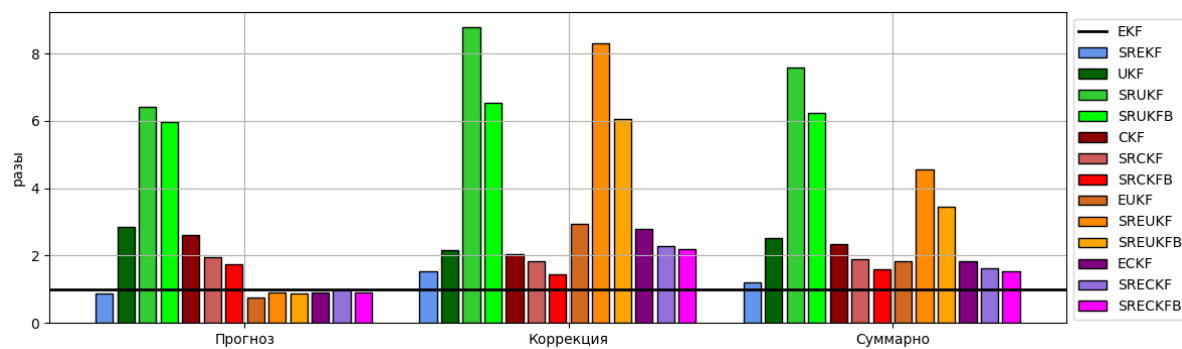


Рис. 1.11 ?>

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

<? <?

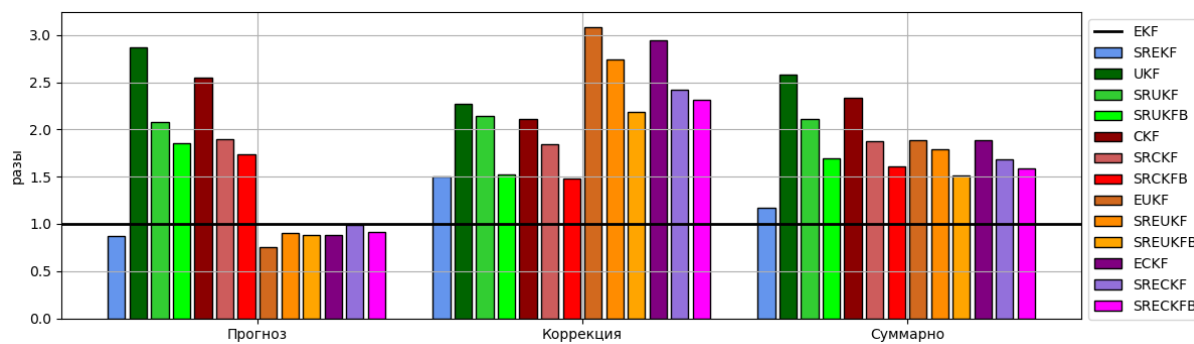


Рис. 1.12 ?>

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 > | 46 |
| KalmanFilters::CKalmanEKF< SizeX, SizeY > | 34 |
| KalmanFilters::CKalmanCKF< SizeX, SizeY > | 21 |
| KalmanFilters::CKalmanECKF< SizeX, SizeY > | 28 |
| KalmanFilters::CKalmanSRCKF< SizeX, SizeY > | 61 |
| KalmanFilters::CKalmanSRCKFB< SizeX, SizeY > | 66 |
| KalmanFilters::CKalmanSRECKF< SizeX, SizeY > | 69 |
| KalmanFilters::CKalmanSRECKFB< SizeX, SizeY > | 73 |
| KalmanFilters::CKalmanSREKF< SizeX, SizeY > | 76 |
| KalmanFilters::CKalmanSRECKF< SizeX, SizeY > | 69 |
| KalmanFilters::CKalmanSREUKF< SizeX, SizeY > | 80 |
| KalmanFilters::CKalmanSREUKFB< SizeX, SizeY > | 85 |
| KalmanFilters::CKalmanUKF< SizeX, SizeY > | 100 |
| KalmanFilters::CKalmanEUKF< SizeX, SizeY > | 41 |
| KalmanFilters::CKalmanSRUKF< SizeX, SizeY > | 89 |
| KalmanFilters::CKalmanSREUKF< SizeX, SizeY > | 80 |
| KalmanFilters::CKalmanSRUKFB< SizeX, SizeY > | 95 |

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) | 28 |
| KalmanFilters::CKalmanEKF< SizeX, SizeY > | Класс расширенного фильтра Калмана, РФК (Extended Kalman Filter, EKF) . . | 34 |
| KalmanFilters::CKalmanEUKF< SizeX, SizeY > | Шаблонный класс расширенного сигма-точечного (ансцентного) фильтра Калмана, РСТФК (Extended Unscented Kalman Filter, EUKF) | 41 |
| KalmanFilters::CKalmanLKF< SizeX, SizeY > | Шаблонный класс линейного фильтра Калмана, ЛФК (Linear Kalman Filter, LKF) | 46 |
| KalmanFilters::CKalmanSRCKF< SizeX, SizeY > | Шаблонный класс квадратно-корневого кубатурного фильтра Калмана, КК-КФК (Square Root Cubature Kalman Filter, SR-CKF) | 61 |
| KalmanFilters::CKalmanSRCKFB< SizeX, SizeY > | Шаблонный класс квадратно-корневого кубатурного фильтра Калмана (блочная фильтрация), КК-КФКБ (Square Root Cubature Kalman Filter Block, SR-CKFB) . | 66 |
| KalmanFilters::CKalmanSRECKF< SizeX, SizeY > | Шаблонный класс квадратно-корневого расширенного кубатурного фильтра Калмана, КК-РКФК (Square Root Extended Cubature Kalman Filter, SR-ECKF) . . . | 69 |
| 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) | 73 |
| KalmanFilters::CKalmanSREKF< SizeX, SizeY > | Класс квадратно-корневого расширенного фильтра Калмана, КК-РФК (Square Root Extended Kalman Filter, EKF) | 76 |
| KalmanFilters::CKalmanSREUKF< SizeX, SizeY > | Шаблонный класс квадратно-корневого расширенного сигма-точечного (ансцентного) фильтра Калмана, КК-РСТФК (Square Root Extended Unscented Kalman Filter, SR-EUKF) | 80 |
| 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) | 85 |
| KalmanFilters::CKalmanSRUKF< SizeX, SizeY > | Шаблонный класс квадратно-корневого сигма-точечного (ансцентного) фильтра Калмана, КК-СТФК (КК-АФК) (Square Root Unscented Kalman Filter, SR-UKF) | 89 |

[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) 95

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

Шаблонный класс сигма-точечного (ансцентного) фильтра Калмана, СТФК (АФК) (Unscented Kalman Filter, UKF) 100

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

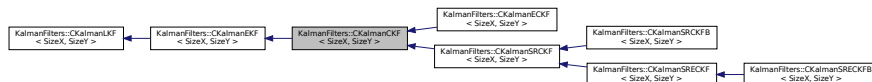
Классы

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

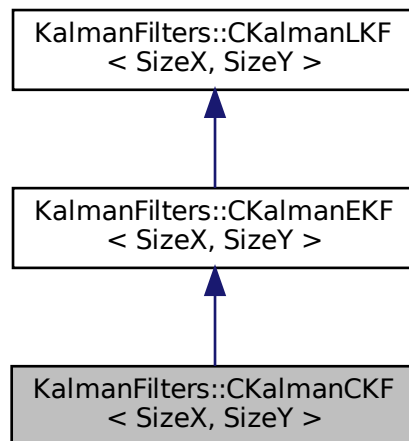
Шаблонный класс кубатурного фильтра Калмана, КФК (Cubature Kalman Filter, CKF)

#include <kalman_filter_cubature.h>

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



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



Открытые члены

- [CKalmanCKF](#) ()
Конструктор по умолчанию
- [CKalmanCKF](#) (const [CKalmanCKF](#) &other)
Конструктор копирования

- [CKalmanCKF](#) & [operator=](#) (const [CKalmanCKF](#) &other)
Перегрузка оператора присвоения
- [CKalmanCKF](#) ([CKalmanCKF](#) &&other) noexcept
Конструктор перемещения
- [CKalmanCKF](#) & [operator=](#) ([CKalmanCKF](#) &&other) noexcept
Перегрузка оператора перемещения
- virtual [~CKalmanCKF](#) ()=default
Деструктор
- 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-коварации 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.

Друзья

- void [swap](#) (CKalmanCKF< SizeX, SizeY > &lhs, CKalmanCKF< SizeX, SizeY > &rhs) noexcept
Метод свапа

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() [1/3]

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

8.1.2.2 CKalmanCKF() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanCKF< SizeX, SizeY >::CKalmanCKF (
    const CKalmanCKF< SizeX, SizeY > & other ) [inline]
Конструктор копирования
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.1.2.3 CKalmanCKF() [3/3]

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

```
KalmanFilters::CKalmanCKF< SizeX, SizeY >::CKalmanCKF (
    CKalmanCKF< SizeX, SizeY > && other ) [inline], [noexcept]
```

Конструктор перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.1.2.4 ~CKalmanCKF()

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

```
virtual KalmanFilters::CKalmanCKF< SizeX, SizeY >::~~CKalmanCKF ( ) [virtual], [default]
```

Деструктор

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 CKF.

Аргументы

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

8.1.3.3 operator=() [1/2]

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

```
CKalmanCKF & KalmanFilters::CKalmanCKF< SizeX, SizeY >::operator= (
    CKalmanCKF< SizeX, SizeY > && other ) [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.1.3.4 operator=() [2/2]

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

```
CKalmanCKF & KalmanFilters::CKalmanCKF< SizeX, SizeY >::operator= (
    const CKalmanCKF< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.1.3.5 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 >](#).

8.1.3.6 PredictionCKF()

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

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

Прогноз CKF.

Аргументы

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

8.1.3.7 SetObservationJacobianH()

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

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

Запрет доступа

8.1.3.8 SetStateTransitionJacobianF()

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

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

Запрет доступа

8.1.3.9 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]

8.1.3.10 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 |
|------------------------------|--|

8.1.3.11 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 Документация по друзьям класса и функциям, относящимся к классу

8.1.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    CKalmanCKF< SizeX, SizeY > & lhs,
    CKalmanCKF< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

8.1.5 Данные класса

8.1.5.1 dXcal_

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

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

8.1.5.2 dYcal_

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

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

8.1.5.3 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 P при создании сигма-точек)

8.1.5.4 k_sigma_points_

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

Число сигма-точек

8.1.5.5 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].

8.1.5.6 sqrt_P_chol_

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

Корень из матрицы P.

8.1.5.7 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.

8.1.5.8 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

8.1.5.9 weights_covariance_

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

Весы ковариации

8.1.5.10 weights_mean_

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

Весы среднего

8.1.5.11 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_].

8.1.5.12 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_].

8.1.5.13 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:

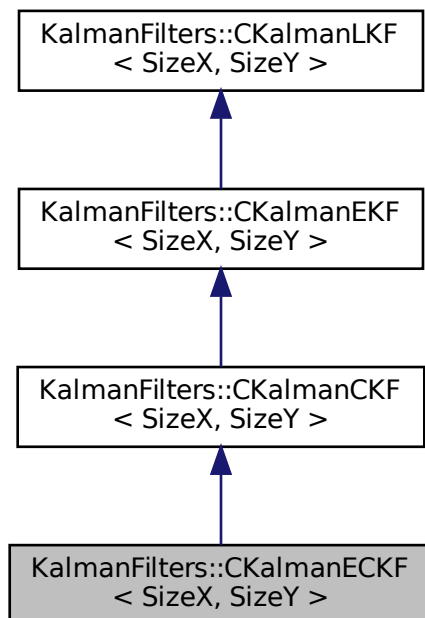
- kalman_filter_cubature.h

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

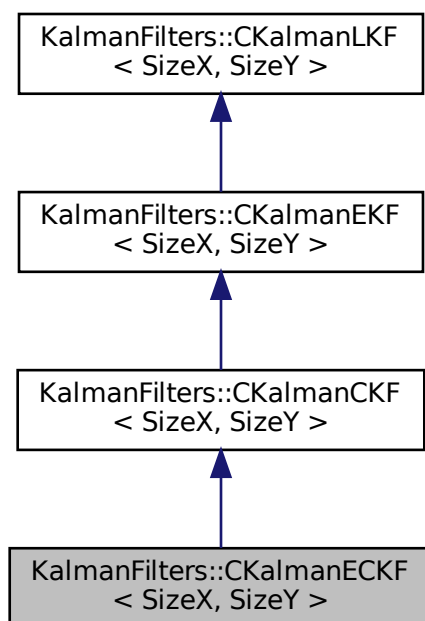
Шаблонный класс расширенного кубатурного фильтра Калмана, РКФК (Extended Cubature Kalman Filter, ECKF)

```
#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
- [CKalmanECKF](#) (const [CKalmanECKF](#) &other)
Конструктор копирования
- [CKalmanECKF](#) & [operator=](#) (const [CKalmanECKF](#) &other)
Перегрузка оператора присвоения
- [CKalmanECKF](#) ([CKalmanECKF](#) &&other) noexcept
Конструктор перемещения
- [CKalmanECKF](#) & [operator=](#) ([CKalmanECKF](#) &&other) noexcept
Перегрузка оператора перемещения
- virtual [~CKalmanECKF](#) ()=default
Деструктор
- 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)

Друзья

- void [swap](#) ([CKalmanECKF](#)< SizeX, SizeY > &lhs, [CKalmanECKF](#)< SizeX, SizeY > &rhs)
noexcept
Метод свапа

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, 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 |

8.2.2 Конструктор(ы)

8.2.2.1 CKalmanECKF() [1/3]

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

8.2.2.2 CKalmanECKF() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanECKF< SizeX, SizeY >::CKalmanECKF (
    const CKalmanECKF< SizeX, SizeY > & other ) [inline]
Конструктор копирования
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.2.2.3 CKalmanECKF() [3/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanECKF< SizeX, SizeY >::CKalmanECKF (
    CKalmanECKF< SizeX, SizeY > && other ) [inline], [noexcept]
Конструктор перемещения
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.2.2.4 ~CKalmanECKF()

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

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 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanECKF & KalmanFilters::CKalmanECKF< SizeX, SizeY >::operator= (
    CKalmanECKF< SizeX, SizeY > && other ) [inline], [noexcept]
Перегрузка оператора перемещения
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.2.3.3 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanECKF & KalmanFilters::CKalmanECKF< SizeX, SizeY >::operator= (
    const CKalmanECKF< SizeX, SizeY > & other ) [inline]
Перегрузка оператора присвоения
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.2.3.4 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 >](#).

8.2.3.5 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_](#)

8.2.3.6 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_](#)

8.2.4 Документация по друзьям класса и функциям, относящимся к классу

8.2.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    CKalmanECKF< SizeX, SizeY > & lhs,
    CKalmanECKF< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

Объявления и описания членов класса находятся в файле:

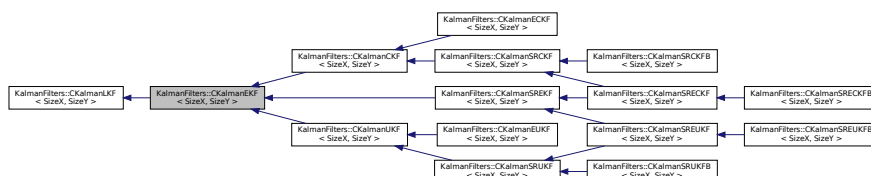
- `kalman_filter_extended_cubature.h`

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

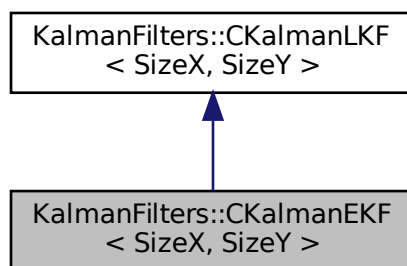
Класс расширенного фильтра Калмана, РФК (Extended Kalman Filter, EKF)

`#include <kalman_filter_extended.h>`

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



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



Открытые члены

- `CKalmanEKF ()`
Конструктор по умолчанию
- `CKalmanEKF (const CKalmanEKF &other)`
Конструктор копирования
- `CKalmanEKF & operator= (const CKalmanEKF &other)`
Перегрузка оператора присвоения
- `CKalmanEKF (CKalmanEKF &&other) noexcept`
Конструктор перемещения
- `CKalmanEKF & operator= (CKalmanEKF &&other) noexcept`
Перегрузка оператора перемещения
- `virtual ~CKalmanEKF ()=default`
Деструктор
- `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)> 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 [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

Друзья

- void [swap](#) (CKalmanEKF< SizeX, SizeY > &lhs, CKalmanEKF< SizeX, SizeY > &rhs) noexcept
- Метод свапа

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() [1/3]

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

Конструктор по умолчанию

8.3.2.2 CKalmanEKF() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanEKF< SizeX, SizeY >::CKalmanEKF (
    const CKalmanEKF< SizeX, SizeY > & other ) [inline]
```

Конструктор копирования

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.3.2.3 CKalmanEKF() [3/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanEKF< SizeX, SizeY >::CKalmanEKF (
    CKalmanEKF< SizeX, SizeY > && other ) [inline], [noexcept]
```

Конструктор перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.3.2.4 ~CKalmanEKF()

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

Деструктор

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 operator=() [1/2]

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

```
CKalmanEKF & KalmanFilters::CKalmanEKF< SizeX, SizeY >::operator= (
    CKalmanEKF< SizeX, SizeY > && other ) [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.3.3.3 operator=() [2/2]

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

```
CKalmanEKF & KalmanFilters::CKalmanEKF< SizeX, SizeY >::operator= (
    const CKalmanEKF< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.3.3.4 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 >](#).

8.3.3.5 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] |
|----|-----------------------|

8.3.3.6 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_](#)

8.3.3.7 SetObservationMatrixH()

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

Запрет доступа

8.3.3.8 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_](#)

8.3.3.9 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_](#)

8.3.3.10 SetStateTransitionJacobianLinearF()

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

8.3.3.11 SetStateTransitionMatrixF()

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

8.3.3.12 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 Документация по друзьям класса и функциям, относящимся к классу

8.3.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    CKalmanEKF< SizeX, SizeY > & lhs,
    CKalmanEKF< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

8.3.5 Данные класса

8.3.5.1 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](#)

8.3.5.2 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](#)

8.3.5.3 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](#)

8.3.5.4 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:

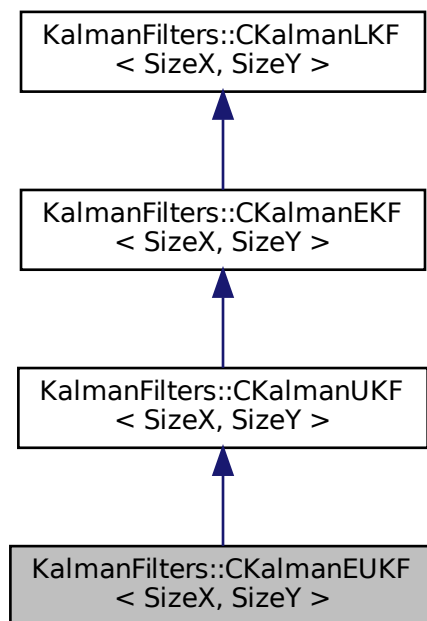
- kalman_filter_extended.h

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

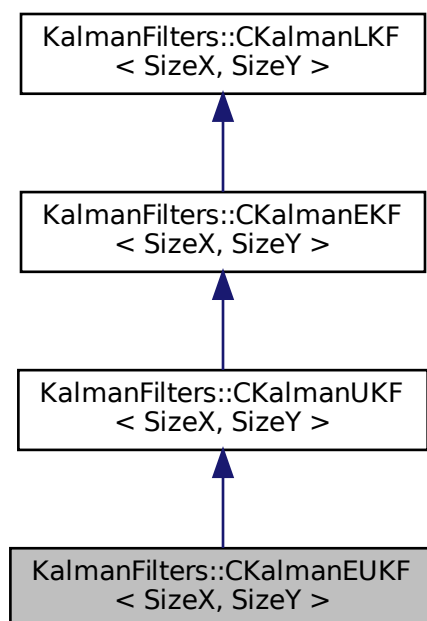
Шаблонный класс расширенного сигма-точечного (ансцентного) фильтра Калмана, РСТФК (Extended Unscented Kalman Filter, EUKF)

#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
- [CKalmanEUKF](#) (const [CKalmanEUKF](#) &other)
Конструктор копирования
- [CKalmanEUKF](#) & [operator=](#) (const [CKalmanEUKF](#) &other)
Перегрузка оператора присвоения
- [CKalmanEUKF](#) ([CKalmanEUKF](#) &&other) noexcept
Конструктор перемещения
- [CKalmanEUKF](#) & [operator=](#) ([CKalmanEUKF](#) &&other) noexcept
Перегрузка оператора перемещения
- virtual [~CKalmanEUKF](#) ()=default
Деструктор
- 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)

Друзья

- void [swap](#) ([CKalmanEUKF](#)< SizeX, SizeY > &lhs, [CKalmanEUKF](#)< SizeX, SizeY > &rhs)
noexcept
Метод свапа

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 Wcov не может быть отрицательным! (Wmean - 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() [1/3]

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

8.4.2.2 CKalmanEUKF() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanEUKF< SizeX, SizeY >::CKalmanEUKF (
    const CKalmanEUKF< SizeX, SizeY > & other ) [inline]
Конструктор копирования
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.4.2.3 CKalmanEUKF() [3/3]

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

Конструктор перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.4.2.4 ~CKalmanEUKF()

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

Деструктор

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

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

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

8.4.3.2 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanEUKF & KalmanFilters::CKalmanEUKF< SizeX, SizeY >::operator= (
    CKalmanEUKF< SizeX, SizeY > && other )    [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.4.3.3 operator=() [2/2]

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

```
CKalmanEUKF & KalmanFilters::CKalmanEUKF< SizeX, SizeY >::operator= (
    const CKalmanEUKF< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.4.3.4 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

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

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

8.4.3.5 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_](#)

8.4.3.6 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_](#)

8.4.4 Документация по друзьям класса и функциям, относящимся к классу

8.4.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    CKalmanEUKF< SizeX, SizeY > & lhs,
    CKalmanEUKF< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

Объявления и описания членов класса находятся в файле:

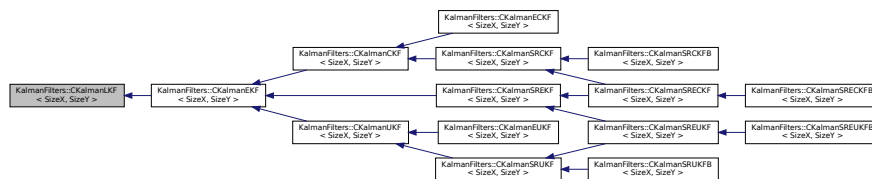
- kalman_filter_extended_unscented.h

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** &other)
Конструктор копирования
- **CKalmanLKF** & **operator=** (const **CKalmanLKF** &other)
Перегрузка оператора присвоения
- **CKalmanLKF** (**CKalmanLKF** &&other) noexcept
Конструктор перемещения
- **CKalmanLKF** & **operator=** (**CKalmanLKF** &&other) noexcept
Перегрузка оператора перемещения
- virtual ~**CKalmanLKF** ()=default
Деструктор
- 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)

Друзья

- `void swap (CKalmanLKF< SizeX, SizeY > &lhs, CKalmanLKF< SizeX, SizeY > &rhs)` `noexcept`
Метод свапа

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 > & other ) [inline]
Конструктор копирования
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.5.2.3 CKalmanLKF() [3/3]

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

```
  KalmanFilters::CKalmanLKF< SizeX, SizeY >::CKalmanLKF (
      CKalmanLKF< SizeX, SizeY > && other )    [inline], [noexcept]
```

Конструктор перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.5.2.4 ~CKalmanLKF()

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

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

Деструктор

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 > && other ) [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.5.3.14 operator=() [2/2]

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

```
CKalmanLKF & KalmanFilters::CKalmanLKF< SizeX, SizeY >::operator= (
    const CKalmanLKF< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

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

| | |
|-------|---|
| X_est | - 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

| | |
|----------|---|
| Y_est | - 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

| | |
|----------|---|
| Y_msd | - 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

| | |
|-----|--|
| R | - 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

| | |
|---|---|
| Н | - 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 Документация по друзьям класса и функциям, относящимся к классу

8.5.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    CKalmanLKF< SizeX, SizeY > & lhs,
    CKalmanLKF< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

8.5.5 Данные класса

8.5.5.1 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.

8.5.5.2 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

8.5.5.3 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

8.5.5.4 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.

8.5.5.5 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.

8.5.5.6 DeltaY_

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

8.5.5.7 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)
```

8.5.5.8 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].
```

8.5.5.9 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].
```

8.5.5.10 I_

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

8.5.5.11 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].
```

8.5.5.12 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].
```

8.5.5.13 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)
```

8.5.5.14 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].

8.5.5.15 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].

8.5.5.16 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].

8.5.5.17 SizeX_

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

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

Размерность вектора состояния X (state)

8.5.5.18 SizeY_

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

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

Размерность вектора измерений Y (measurement)

8.5.5.19 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 |
|---|---|

8.5.5.20 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].

8.5.5.21 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].

8.5.5.22 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].

8.5.5.23 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].

8.5.5.24 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;.

8.5.5.25 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:

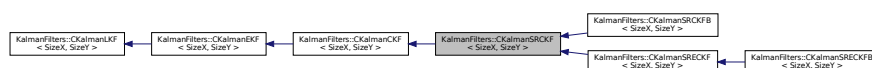
- kalman_filter_linear.h

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

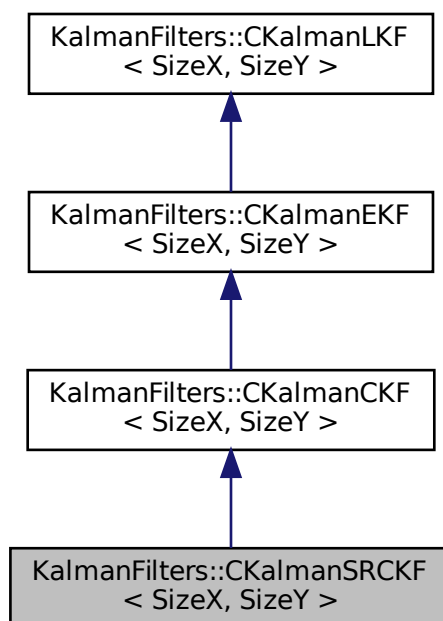
Шаблонный класс квадратно-корневого кубатурного фильтра Калмана, КК-КФК (Square Root Cubature Kalman Filter, SR-CKF)

```
#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
- [CKalmanSRCKF](#) (const [CKalmanSRCKF](#) &other)
Конструктор копирования
- [CKalmanSRCKF](#) & [operator=](#) (const [CKalmanSRCKF](#) &other)
Перегрузка оператора присвоения
- [CKalmanSRCKF](#) ([CKalmanSRCKF](#) &&other) noexcept
Конструктор перемещения
- [CKalmanSRCKF](#) & [operator=](#) ([CKalmanSRCKF](#) &&other) noexcept
Перегрузка оператора перемещения
- virtual [~CKalmanSRCKF](#) ()=default
Деструктор
- 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.

Друзья

- void [swap](#) ([CKalmanSRCKF](#)< SizeX, SizeY > &lhs, [CKalmanSRCKF](#)< SizeX, SizeY > &rhs)
поехсепт
Метод свапа

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() [1/3]

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

8.6.2.2 CKalmanSRCKF() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSRCKF< SizeX, SizeY >::CKalmanSRCKF (
    const CKalmanSRCKF< SizeX, SizeY > & other ) [inline]
```

Конструктор копирования

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.6.2.3 CKalmanSRCKF() [3/3]

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

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

Конструктор перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.6.2.4 ~CKalmanSRCKF()

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

```
virtual KalmanFilters::CKalmanSRCKF< SizeX, SizeY >::~~CKalmanSRCKF ( ) [virtual], [default]
```

Деструктор

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 operator=() [1/2]

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

```
CKalmanSRCKF & KalmanFilters::CKalmanSRCKF< SizeX, SizeY >::operator= (
    CKalmanSRCKF< SizeX, SizeY > && other ) [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.6.3.4 operator=() [2/2]

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

```
CKalmanSRCKF & KalmanFilters::CKalmanSRCKF< SizeX, SizeY >::operator= (
    const CKalmanSRCKF< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.6.3.5 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 >](#).

8.6.3.6 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_](#)

8.6.3.7 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_](#)

8.6.4 Документация по друзьям класса и функциям, относящимся к классу

8.6.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    CKalmanSRCKF< SizeX, SizeY > & lhs,
    CKalmanSRCKF< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

Объявления и описания членов класса находятся в файле:

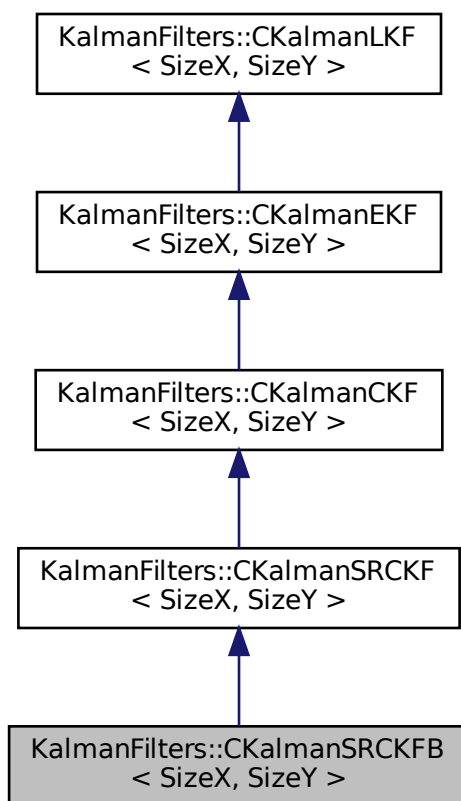
- kalman_filter_cubature_square_root.h

8.7 Шаблон класса KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >

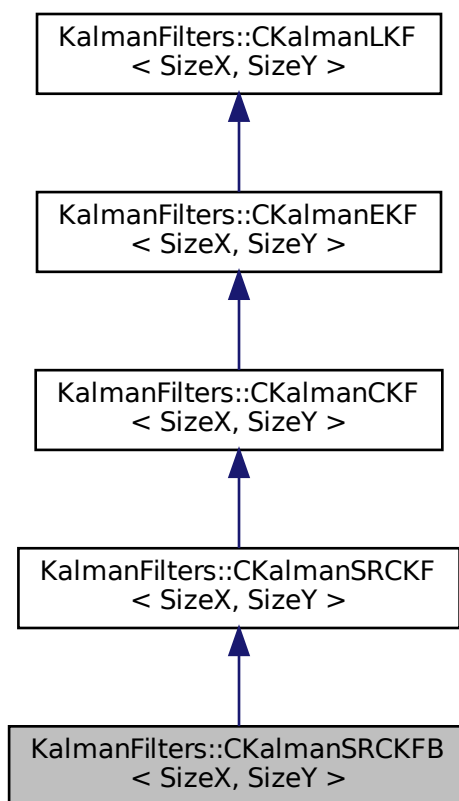
Шаблонный класс квадратно-корневого кубатурного фильтра Калмана (блочная фильтрация), КК-КФКБ (Square Root Cubature Kalman Filter Block, SR-CKFB)

#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 >

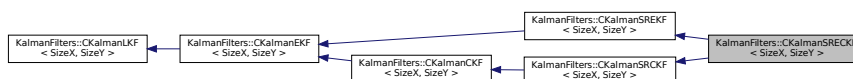
Шаблонный класс квадратно-корневого расширенного кубатурного фильтра Калмана, КК-РКФК (Square Root Extended Cubature Kalman Filter, SR-ECKF)

```
#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
- [CKalmanSRECKF](#) (const [CKalmanSRECKF](#) &other)
Конструктор копирования
- [CKalmanSRECKF](#) & [operator=](#) (const [CKalmanSRECKF](#) &other)
Перегрузка оператора присвоения
- [CKalmanSRECKF](#) ([CKalmanSRECKF](#) &&other) noexcept
Конструктор перемещения
- [CKalmanSRECKF](#) & [operator=](#) ([CKalmanSRECKF](#) &&other) noexcept
Перегрузка оператора перемещения
- virtual [~CKalmanSRECKF](#) ()=default
Деструктор
- 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)

Друзья

- void [swap](#) ([CKalmanSRECKF](#)< SizeX, SizeY > &lhs, [CKalmanSRECKF](#)< SizeX, SizeY > &rhs)
noexcept
Метод свапа

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, 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 |

8.8.2 Конструктор(ы)

8.8.2.1 CKalmanSRECKF() [1/3]

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

8.8.2.2 CKalmanSRECKF() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSRECKF< SizeX, SizeY >::CKalmanSRECKF (
    const CKalmanSRECKF< SizeX, SizeY > & other ) [inline]
Конструктор копирования
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.8.2.3 CKalmanSRECKF() [3/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSRECKF< SizeX, SizeY >::CKalmanSRECKF (
    CKalmanSRECKF< SizeX, SizeY > && other ) [inline], [noexcept]
Конструктор перемещения
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.8.2.4 ~CKalmanSRECKF()

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

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- \leftrightarrow 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 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanSRECKF & KalmanFilters::CKalmanSRECKF< SizeX, SizeY >::operator= (
    CKalmanSRECKF< SizeX, SizeY > && other ) [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.8.3.3 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanSRECKF & KalmanFilters::CKalmanSRECKF< SizeX, SizeY >::operator= (
    const CKalmanSRECKF< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.8.3.4 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 >](#).

8.8.4 Документация по друзьям класса и функциям, относящимся к классу

8.8.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    CKalmanSRECKFB< SizeX, SizeY > & lhs,
    CKalmanSRECKFB< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

Объявления и описания членов класса находятся в файле:

- kalman_filter_extended_cubature_square_root.h

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

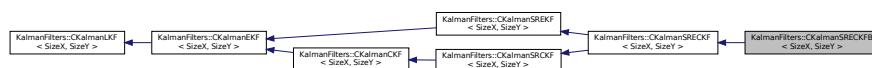
Шаблонный класс квадратно-корневого расширенного кубатурного фильтра Калмана (блочная фильтрация), КК-РКФКБ (Square Root Extended Cubature Kalman Filter Block, SR-ECKFB)

```
#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
- [CKalmanSRECKFB](#) (const [CKalmanSRECKFB](#) &other)
Конструктор копирования
- [CKalmanSRECKFB](#) & operator= (const [CKalmanSRECKFB](#) &other)
Перегрузка оператора присвоения
- [CKalmanSRECKFB](#) ([CKalmanSRECKFB](#) &&other) noexcept
Конструктор перемещения
- [CKalmanSRECKFB](#) & operator= ([CKalmanSRECKFB](#) &&other) noexcept
Перегрузка оператора перемещения
- virtual [~CKalmanSRECKFB](#) ()=default
Деструктор
- virtual void [Correction](#) (const arma::vec &Y_msd)
Коррекция

Друзья

- void [swap](#) ([CKalmanSRECKFB](#)< SizeX, SizeY > &lhs, [CKalmanSRECKFB](#)< SizeX, SizeY > &rhs) noexcept
Метод свапа

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() [1/3]

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

8.9.2.2 CKalmanSRECKFB() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >::CKalmanSRECKFB (
    const CKalmanSRECKFB< SizeX, SizeY > & other ) [inline]
Конструктор копирования
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.9.2.3 CKalmanSRECKFB() [3/3]

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


Конструктор перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.9.2.4 ~CKalmanSRECKFB()

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

```
virtual KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >::~~CKalmanSRECKFB ( ) [virtual], [default]
```

Деструктор

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 >](#).

8.9.3.2 operator=() [1/2]

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

```
CKalmanSRECKFB & KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >::operator= (
    CKalmanSRECKFB< SizeX, SizeY > && other ) [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.9.3.3 operator=() [2/2]

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

```
CKalmanSRECKFB & KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >::operator= (
    const CKalmanSRECKFB< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.9.4 Документация по друзьям класса и функциям, относящимся к классу

8.9.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    CKalmanSRECKFB< SizeX, SizeY > & lhs,
    CKalmanSRECKFB< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

Объявления и описания членов класса находятся в файле:

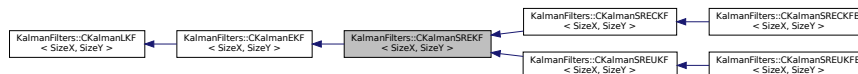
- kalman_filter_extended_cubature_square_root.h

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

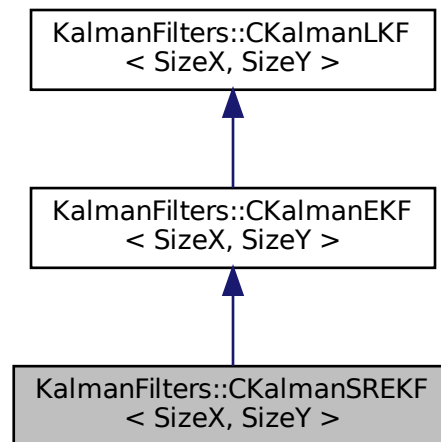
Класс квадратно-корневого расширенного фильтра Калмана, КК-РФК (Square Root Extended Kalman Filter, EKF)

#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
- `CKalmanSREKF` (const `CKalmanSREKF` &other)
Конструктор копирования
- `CKalmanSREKF` & `operator=` (const `CKalmanSREKF` &other)
Перегрузка оператора присвоения
- `CKalmanSREKF` (`CKalmanSREKF` &&other) noexcept
Конструктор перемещения
- `CKalmanSREKF` & `operator=` (`CKalmanSREKF` &&other) noexcept
Перегрузка оператора перемещения
- virtual `~CKalmanSREKF` ()=default
Деструктор
- virtual void `Prediction` (double dt)
Прогноз
- virtual void `Correction` (const arma::vec &Y_msd)
Коррекция

Защищенные члены

- void `PredictionSREKF` (double dt)
Прогноз SREKF.

Друзья

- void `swap` (`CKalmanSREKF` &lhs, `CKalmanSREKF` &rhs) noexcept
Метод свапа

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() [1/3]

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

8.10.2.2 CKalmanSREKF() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSREKF< SizeX, SizeY >::CKalmanSREKF (
    const CKalmanSREKF< SizeX, SizeY > & other ) [inline]
Конструктор копирования
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.10.2.3 CKalmanSREKF() [3/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSREKF< SizeX, SizeY >::CKalmanSREKF (
    CKalmanSREKF< SizeX, SizeY > && other ) [inline], [noexcept]
Конструктор перемещения
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.10.2.4 ~CKalmanSREKF()

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

virtual [KalmanFilters::CKalmanSREKF< SizeX, SizeY >::~CKalmanSREKF](#) () [virtual], [default]
 Деструктор

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 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanSREKF & KalmanFilters::CKalmanSREKF< SizeX, SizeY >::operator= (
    CKalmanSREKF< SizeX, SizeY > && other ) [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.10.3.3 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanSREKF & KalmanFilters::CKalmanSREKF< SizeX, SizeY >::operator= (
    const CKalmanSREKF< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

`*this`

8.10.3.4 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 >.

8.10.3.5 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] |
|----|-----------------------|

8.10.4 Документация по друзьям класса и функциям, относящимся к классу

8.10.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    CKalmanSREKF< SizeX, SizeY > & lhs,
    CKalmanSREKF< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

Объявления и описания членов класса находятся в файле:

- `kalman_filter_extended_square_root.h`

8.11 Шаблон класса [KalmanFilters::CKalmanSREUKF](#)< SizeX, SizeY >

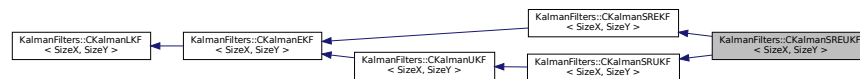
Шаблонный класс квадратно-корневого расширенного сигма-точечного (ансцентного) фильтра Калмана, КК-РСТФК (Square Root Extended Unscented Kalman Filter, SR-EUKF)

```
#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
- [CKalmanSREUKF](#) (const [CKalmanSREUKF](#) &other)
Конструктор копирования
- [CKalmanSREUKF](#) & [operator=](#) (const [CKalmanSREUKF](#) &other)
Перегрузка оператора присвоения
- [CKalmanSREUKF](#) ([CKalmanSREUKF](#) &&other) noexcept
Конструктор перемещения
- [CKalmanSREUKF](#) & [operator=](#) ([CKalmanSREUKF](#) &&other) noexcept
Перегрузка оператора перемещения
- virtual [~CKalmanSREUKF](#) ()=default
Деструктор
- 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 Pху.
- arma::mat [Jcorrect_](#)
Матрица знаков при коррекции

Друзья

- void [swap](#) ([CKalmanSREUKF](#)< SizeX, SizeY > &lhs, [CKalmanSREUKF](#)< SizeX, SizeY > &rhs)
noexcept
Метод свапа

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() [1/3]

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

8.11.2.2 CKalmanSREUKF() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSREUKF< SizeX, SizeY >::CKalmanSREUKF (
    const CKalmanSREUKF< SizeX, SizeY > & other ) [inline]
Конструктор копирования
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.11.2.3 CKalmanSREUKF() [3/3]

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

```
KalmanFilters::CKalmanSREUKF< SizeX, SizeY >::CKalmanSREUKF (
    CKalmanSREUKF< SizeX, SizeY > && other )    [inline], [noexcept]
```

Конструктор перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.11.2.4 ~CKalmanSREUKF()

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

```
virtual KalmanFilters::CKalmanSREUKF< SizeX, SizeY >::~~CKalmanSREUKF ( )    [virtual], [default]
```

Деструктор

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 operator=() [1/2]

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

```
CKalmanSREUKF & KalmanFilters::CKalmanSREUKF< SizeX, SizeY >::operator= (
    CKalmanSREUKF< SizeX, SizeY > && other )    [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

`*this`

8.11.3.4 `operator=()` [2/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanSREUKF & KalmanFilters::CKalmanSREUKF< SizeX, SizeY >::operator= (
    const CKalmanSREUKF< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

`*this`

8.11.3.5 `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 Документация по друзьям класса и функциям, относящимся к классу

8.11.4.1 `swap`

```
template<size_t SizeX, size_t SizeY>
void swap (
    CKalmanSREUKF< SizeX, SizeY > & lhs,
    CKalmanSREUKF< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

8.11.5 Данные класса

8.11.5.1 J_

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

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

Матрица знаков при Pxy.

8.11.5.2 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:

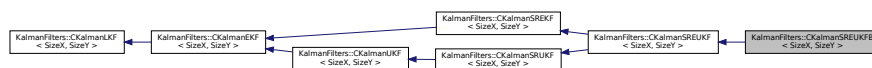
- kalman_filter_extended_unscented_square_root.h

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
- [CKalmanSREUKFB](#) (const [CKalmanSREUKFB](#) &other)
Конструктор копирования
- [CKalmanSREUKFB](#) & operator= (const [CKalmanSREUKFB](#) &other)
Перегрузка оператора присвоения
- [CKalmanSREUKFB](#) ([CKalmanSREUKFB](#) &&other) noexcept
Конструктор перемещения
- [CKalmanSREUKFB](#) & operator= ([CKalmanSREUKFB](#) &&other) noexcept
Перегрузка оператора перемещения
- virtual ~[CKalmanSREUKFB](#) ()=default
Деструктор
- 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

Друзья

- void [swap](#) (CKalmanSREUKFB< SizeX, SizeY > &lhs, CKalmanSREUKFB< SizeX, SizeY > &rhs) noexcept
Метод свапа

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() [1/3]

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

8.12.2.2 CKalmanSREUKFB() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >::CKalmanSREUKFB (
    const CKalmanSREUKFB< SizeX, SizeY > & other ) [inline]
```

Конструктор копирования

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.12.2.3 CKalmanSREUKFB() [3/3]

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

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

Конструктор перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.12.2.4 ~CKalmanSREUKFB()

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

```
virtual KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >::~~CKalmanSREUKFB ( )    [virtual], [default]
```

Деструктор

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

8.12.3.3 operator=() [1/2]

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

```
CKalmanSREUKFB & KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >::operator= (
    CKalmanSREUKFB< SizeX, SizeY > && other )    [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.12.3.4 operator=() [2/2]

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

```
CKalmanSREUKFB & KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >::operator= (
    const CKalmanSREUKFB< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.12.4 Документация по друзьям класса и функциям, относящимся к классу

8.12.4.1 swap

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

```
void swap (
    CKalmanSREUKFB< SizeX, SizeY > & lhs,
    CKalmanSREUKFB< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

8.12.5 Данные класса

8.12.5.1 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:

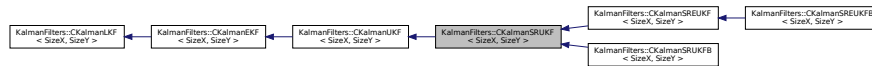
- kalman_filter_extended_unscented_square_root.h

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

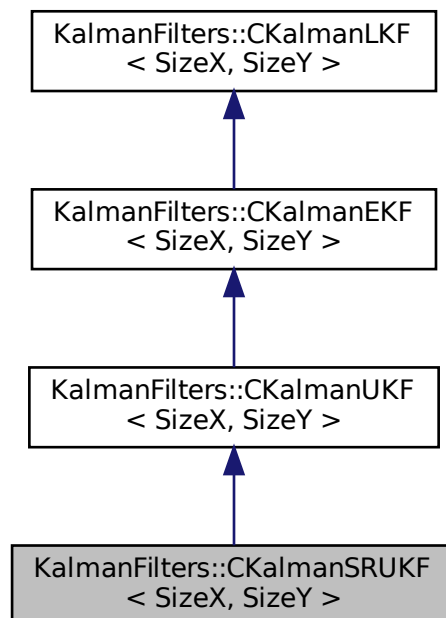
Шаблонный класс квадратно-корневого сигма-точечного (ансцентного) фильтра Калмана, КК-СТФК (КК-АФК) (Square Root Unscented Kalman Filter, SR-UKF)

#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
- [CKalmanSRUKF](#) (const [CKalmanSRUKF](#) &other)
Конструктор копирования
- [CKalmanSRUKF](#) & operator= (const [CKalmanSRUKF](#) &other)
Перегрузка оператора присвоения
- [CKalmanSRUKF](#) ([CKalmanSRUKF](#) &&other) noexcept
Конструктор перемещения
- [CKalmanSRUKF](#) & operator= ([CKalmanSRUKF](#) &&other) noexcept
Перегрузка оператора перемещения
- virtual [~CKalmanSRUKF](#) ()=default
Деструктор

- 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 [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.

Друзья

- void [swap](#) (CKalmanSRUKF< SizeX, SizeY > &lhs, CKalmanSRUKF< SizeX, SizeY > &rhs)
noexcept
Метод свапа

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() [1/3]

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

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

Конструктор по умолчанию

8.13.2.2 CKalmanSRUKF() [2/3]

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

```
KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::CKalmanSRUKF (
    const CKalmanSRUKF< SizeX, SizeY > & other ) [inline]
```

Конструктор копирования

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.13.2.3 CKalmanSRUKF() [3/3]

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

```
KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::CKalmanSRUKF (
    CKalmanSRUKF< SizeX, SizeY > && other ) [inline], [noexcept]
```

Конструктор перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.13.2.4 ~CKalmanSRUKF()

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

```
virtual KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::~~CKalmanSRUKF ( ) [virtual], [default]
```

Деструктор

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 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanSRUKF & KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::operator= (
    CKalmanSRUKF< SizeX, SizeY > && other ) [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.13.3.5 operator=() [2/2]

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

```
CKalmanSRUKF & KalmanFilters::CKalmanSRUKF< SizeX, SizeY >::operator= (
    const CKalmanSRUKF< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.13.3.6 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 >](#).

8.13.3.7 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_](#)

8.13.3.8 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_](#)

8.13.3.9 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 >](#).

8.13.3.10 SetupDesignParametersMeanSet()

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

Установка параметра w_0 апсцентного фильтра (MeanSet)

При выборе $w_0 = [0...1]$ обеспечиваются положительные веса. Смотри [1], [2]

Внимание

Нельзя выбирать w_0 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 ($w_0 = [0...1]$ типичная рекомендация для положительных весов) |
|----|--|

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

8.13.3.11 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 апсцентного 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) |
| kappa | - 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 Документация по друзьям класса и функциям, относящимся к классу

8.13.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
```

```

    CKalmanSRUKF< SizeX, SizeY > & lhs,
    CKalmanSRUKF< SizeX, SizeY > & rhs ) [friend]

```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

8.13.5 Данные класса

8.13.5.1 J_

```

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

```

Матрица знаков при Pxy.

8.13.5.2 Jcorrect_

```

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

```

Матрица знаков при коррекции

8.13.5.3 Jpredict_

```

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

```

Матрица знаков при прогнозе

8.13.5.4 negativeZeroCovWeight_

```

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

```

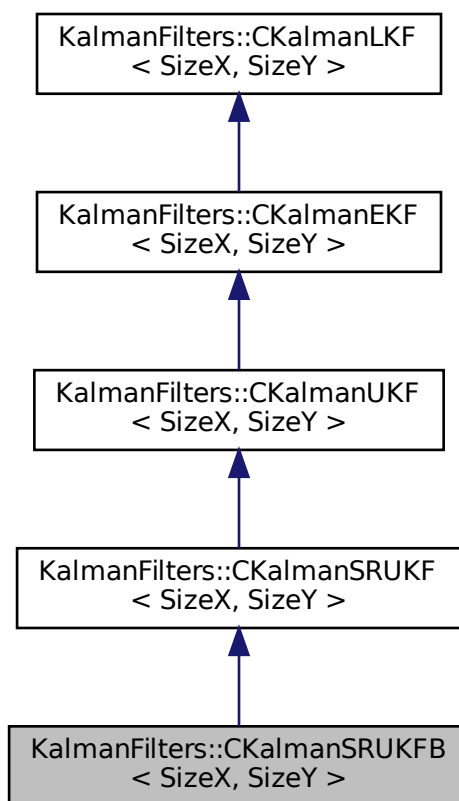
Признак отрицательного веса "нулевой" сигма-точки Wcov.
Объявления и описания членов класса находятся в файле:

- kalman_filter_unscented_square_root.h

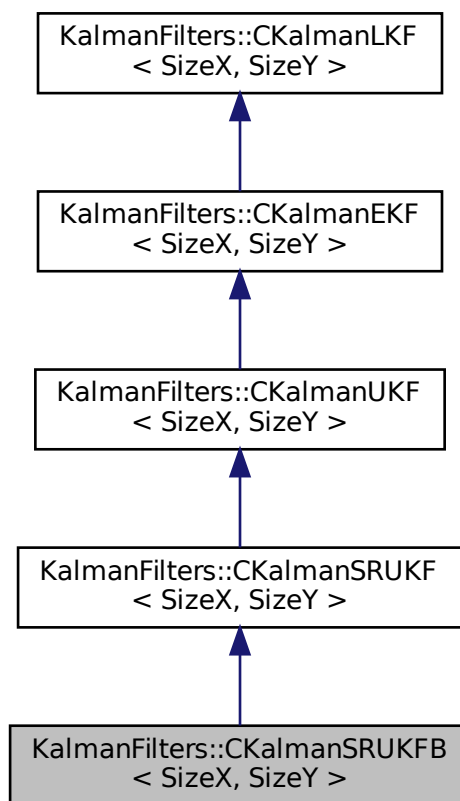
8.14 Шаблон класса KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >

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
- `CKalmanSRUKFB (const CKalmanSRUKFB &other)`
Конструктор копирования
- `CKalmanSRUKFB & operator= (const CKalmanSRUKFB &other)`
Перегрузка оператора присвоения
- `CKalmanSRUKFB (CKalmanSRUKFB &&other) noexcept`
Конструктор перемещения
- `CKalmanSRUKFB & operator= (CKalmanSRUKFB &&other) noexcept`
Перегрузка оператора перемещения
- `virtual ~CKalmanSRUKFB ()=default`
Деструктор
- `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

Друзья

- `void swap (CKalmanSRUKFB &lhs, CKalmanSRUKFB &rhs)` `noexcept`
Метод свапа

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() [1/3]

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

8.14.2.2 CKalmanSRUKFB() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >::CKalmanSRUKFB (
    const CKalmanSRUKFB< SizeX, SizeY > & other ) [inline]
Конструктор копирования
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.14.2.3 CKalmanSRUKFB() [3/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >::CKalmanSRUKFB (
    CKalmanSRUKFB< SizeX, SizeY > && other ) [inline], [noexcept]
Конструктор перемещения
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.14.2.4 ~CKalmanSRUKFB()

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

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
```

8.14.3.3 operator=() [1/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanSRUKFB & KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >::operator= (
    CKalmanSRUKFB< SizeX, SizeY > && other ) [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.14.3.4 operator=() [2/2]

```
template<size_t SizeX, size_t SizeY>
CKalmanSRUKFB & KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >::operator= (
    const CKalmanSRUKFB< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.14.4 Документация по друзьям класса и функциям, относящимся к классу

8.14.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
    CKalmanSRUKFB< SizeX, SizeY > & lhs,
    CKalmanSRUKFB< SizeX, SizeY > & rhs ) [friend]
```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

8.14.5 Данные класса

8.14.5.1 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:

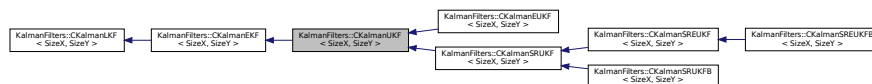
- kalman_filter_unscented_square_root.h

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

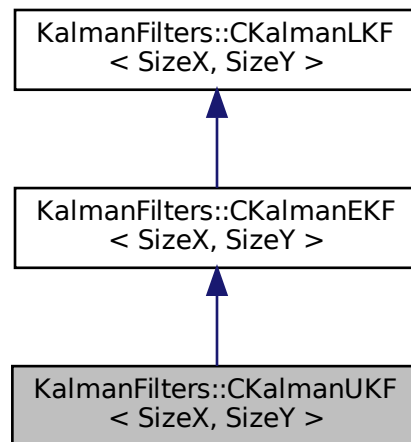
Шаблонный класс сигма-точечного (ансцентного) фильтра Калмана, СТФК (АФК) (Unscented Kalman Filter, UKF)

#include <kalman_filter_unscented.h>

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



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



Открытые члены

- [CKalmanUKF](#) ()
Конструктор по умолчанию
- [CKalmanUKF](#) (const [CKalmanUKF](#) &other)
Конструктор копирования
- [CKalmanUKF](#) & [operator=](#) (const [CKalmanUKF](#) &other)
Перегрузка оператора присвоения
- [CKalmanUKF](#) ([CKalmanUKF](#) &&other) noexcept
Конструктор перемещения
- [CKalmanUKF](#) & [operator=](#) ([CKalmanUKF](#) &&other) noexcept
Перегрузка оператора перемещения
- virtual [~CKalmanUKF](#) ()=default
Деструктор
- virtual void [SetupDesignParametersMeanSet](#) (double w0)
Установка параметра w0 апостериорного фильтра (MeanSet)
- virtual void [SetupDesignParametersScaledSet](#) (double alpha, double beta, double kappa)
Установка параметров масштабируемого апостериорного 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.

Друзья

- void **swap** (CKalmanUKF< SizeX, SizeY > &lhs, CKalmanUKF< SizeX, SizeY > &rhs) noexcept
Метод свапа

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.P.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 Wcov не может быть отрицательным! (Wmean - 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() [1/3]

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

8.15.2.2 CKalmanUKF() [2/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanUKF< SizeX, SizeY >::CKalmanUKF (
    const CKalmanUKF< SizeX, SizeY > & other ) [inline]
Конструктор копирования
```

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.15.2.3 CKalmanUKF() [3/3]

```
template<size_t SizeX, size_t SizeY>
KalmanFilters::CKalmanUKF< SizeX, SizeY >::CKalmanUKF (
    CKalmanUKF< SizeX, SizeY > && other ) [inline], [noexcept]
```

Конструктор перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

8.15.2.4 ~CKalmanUKF()

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

Деструктор

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 operator=() [1/2]

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

```
CKalmanUKF & KalmanFilters::CKalmanUKF< SizeX, SizeY >::operator= (
    CKalmanUKF< SizeX, SizeY > && other ) [inline], [noexcept]
```

Перегрузка оператора перемещения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.15.3.4 operator=() [2/2]

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

```
CKalmanUKF & KalmanFilters::CKalmanUKF< SizeX, SizeY >::operator= (
    const CKalmanUKF< SizeX, SizeY > & other ) [inline]
```

Перегрузка оператора присвоения

Аргументы

| | |
|-------|--|
| other | - экземпляр, с которого делается копия |
|-------|--|

Возвращает

*this

8.15.3.5 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 >](#).

8.15.3.6 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] |
|----|-----------------------|

8.15.3.7 SetObservationJacobianH()

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

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

Запрет доступа

8.15.3.8 SetStateTransitionJacobianF()

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

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

Запрет доступа

8.15.3.9 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 >](#).

8.15.3.10 SetupDesignParametersMeanSet()

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

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

Установка параметра w_0 апостериорного фильтра (MeanSet)

При выборе $w_0 = [0...1]$ обеспечиваются положительные веса. Смотри [1], [2]

Внимание

Нельзя выбирать w_0 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 ($w_0 = [0...1]$ типичная рекомендация для положительных весов) |
|----|--|

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

8.15.3.11 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 апостериорного 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) |
| каппа | - параметр, отвечающий за разброс сигма-точек ($\kappa = 0$ или $(3 - \text{SizeX})$ - типичная рекомендация по van der Merwe) |

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

8.15.3.12 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 |
|------------------------------|--|

8.15.3.13 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 Документация по друзьям класса и функциям, относящимся к классу

8.15.4.1 swap

```
template<size_t SizeX, size_t SizeY>
void swap (
```

```

    CKalmanUKF< SizeX, SizeY > & lhs,
    CKalmanUKF< SizeX, SizeY > & rhs )    [friend]

```

Метод свапа

Аргументы

| | |
|-----|----------------------------|
| lhs | - left hand side instance |
| rhs | - right hand side instance |

8.15.5 Данные класса

8.15.5.1 alpha_

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

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

Параметр разброса сигма-точек ($\alpha = 10^{-3}$ типичная рекомендация)

8.15.5.2 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)

8.15.5.3 dXcal_

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

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

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

8.15.5.4 dYcal_

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

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

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

8.15.5.5 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)

8.15.5.6 k_sigma_points_

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

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

Число сигма-точек

8.15.5.7 kappa_

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

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

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

8.15.5.8 lambda_

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

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

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

8.15.5.9 P_xy_

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

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

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

8.15.5.10 sqrt_P_chol_

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

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

Корень из матрицы P.

8.15.5.11 w0_

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

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

Параметр разброса сигма-точек (0..1)

8.15.5.12 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.

8.15.5.13 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

8.15.5.14 weights_covariance_

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

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

Веса ковариации

8.15.5.15 `weights_mean_`

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

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

Веса среднего

8.15.5.16 `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_].

8.15.5.17 `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_].

8.15.5.18 `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

| | |
|---|---|
| ~CKalmanCKF
KalmanFilters::CKalmanCKF< SizeX, SizeY
>, 24 | CalculateInnovationCovarianceS
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 50 |
| ~CKalmanECKF
KalmanFilters::CKalmanECKF< SizeX,
SizeY >, 32 | checkBordersMeasurement_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 58 |
| ~CKalmanEKF
KalmanFilters::CKalmanEKF< SizeX, SizeY
>, 36 | checkBordersStateAfterCorrection_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 58 |
| ~CKalmanEUKF
KalmanFilters::CKalmanEUKF< SizeX,
SizeY >, 44 | checkBordersStateAfterPrediction_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 58 |
| ~CKalmanLKF
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 50 | checkDeltaMeasurement_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 58 |
| ~CKalmanSRCKF
KalmanFilters::CKalmanSRCKF< SizeX,
SizeY >, 64 | checkDeltaState_
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 58 |
| ~CKalmanSRECKF
KalmanFilters::CKalmanSRECKF< SizeX,
SizeY >, 71 | checkMatrixDiagPositive
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 50 |
| ~CKalmanSRECKFB
KalmanFilters::CKalmanSRECKFB< SizeX,
SizeY >, 75 | CKalmanCKF
KalmanFilters::CKalmanCKF< SizeX, SizeY
>, 23 , 24 |
| ~CKalmanSREKF
KalmanFilters::CKalmanSREKF< SizeX,
SizeY >, 78 | CKalmanECKF
KalmanFilters::CKalmanECKF< SizeX,
SizeY >, 31 |
| ~CKalmanSREUKF
KalmanFilters::CKalmanSREUKF< SizeX,
SizeY >, 83 | CKalmanEKF
KalmanFilters::CKalmanEKF< SizeX, SizeY
>, 36 |
| ~CKalmanSREUKFB
KalmanFilters::CKalmanSREUKFB< SizeX,
SizeY >, 87 | CKalmanEUKF
KalmanFilters::CKalmanEUKF< SizeX,
SizeY >, 43 , 44 |
| ~CKalmanSRUKF
KalmanFilters::CKalmanSRUKF< SizeX,
SizeY >, 91 | CKalmanLKF
KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 49 |
| ~CKalmanSRUKFB
KalmanFilters::CKalmanSRUKFB< SizeX,
SizeY >, 99 | CKalmanSRCKF
KalmanFilters::CKalmanSRCKF< SizeX,
SizeY >, 63 |
| ~CKalmanUKF
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, 104 | CKalmanSRCKFB
KalmanFilters::CKalmanSRCKFB< SizeX,
SizeY >, 69 |
| alpha_
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, 108 | CKalmanSRECKF
KalmanFilters::CKalmanSRECKF< SizeX,
SizeY >, 71 |
| beta_
KalmanFilters::CKalmanUKF< SizeX, SizeY
>, 108 | CKalmanSRECKFB
KalmanFilters::CKalmanSRECKFB< SizeX,
SizeY >, 74 |
| | CKalmanSREKF |

- KalmanFilters::CKalmanSREKF< SizeX, SizeY >, [78](#)
 CKalmanSREUKF
 KalmanFilters::CKalmanSREUKF< SizeX, SizeY >, [82](#)
 CKalmanSREUKFB
 KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >, [86](#), [87](#)
 CKalmanSRUKF
 KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, [91](#)
 CKalmanSRUKFB
 KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >, [98](#)
 CKalmanUKF
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, [103](#), [104](#)
 Correction
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, [24](#)
 KalmanFilters::CKalmanECKF< SizeX, SizeY >, [32](#)
 KalmanFilters::CKalmanEKF< SizeX, SizeY >, [36](#)
 KalmanFilters::CKalmanEUKF< SizeX, SizeY >, [44](#)
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, [51](#)
 KalmanFilters::CKalmanSRCKF< SizeX, SizeY >, [64](#)
 KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >, [69](#)
 KalmanFilters::CKalmanSRECKF< SizeX, SizeY >, [71](#)
 KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >, [75](#)
 KalmanFilters::CKalmanSREKF< SizeX, SizeY >, [79](#)
 KalmanFilters::CKalmanSREUKF< SizeX, SizeY >, [83](#)
 KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >, [87](#)
 KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, [92](#)
 KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >, [99](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, [104](#)
 CorrectionCKF
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, [24](#)
 CorrectionSRCKF
 KalmanFilters::CKalmanSRCKF< SizeX, SizeY >, [64](#)
 CorrectionSRUKF
 KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, [92](#)
 CorrectionUKF
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, [104](#)
 createSignMatrices
 KalmanFilters::CKalmanSREUKF< SizeX, SizeY >, [83](#)
 KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, [92](#)
 createSignMatricesBlock
 KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >, [87](#)
 KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >, [99](#)
 DeltaY_
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, [58](#)
 deltaY_isSet
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, [59](#)
 dXcal_
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, [27](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, [108](#)
 dYcal_
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, [27](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, [108](#)
 F_
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, [59](#)
 fixMatrixMainDiagonalSymmetry
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, [51](#)
 gamma_
 KalmanFilters::CKalmanCKF< SizeX, SizeY >, [27](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY >, [108](#)
 GetDeltaY
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, [51](#)
 GetEstimatedCovarianceMatrixP
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, [51](#)
 GetEstimatedVectorX
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, [51](#)
 GetEstimatedVectorY
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, [52](#)
 GetInnovationCovarianceMatrixS
 KalmanFilters::CKalmanLKF< SizeX, SizeY >, [52](#)
 GetKalmanGainMatrixK

- KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 52
- GetSizeX
 - KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 52
- GetSizeY
 - KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 52
- GetVersion
 - KalmanFilters, 20
- H_
 - KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 59
- I_
 - KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 59
- J_
 - KalmanFilters::CKalmanSREUKF< SizeX,
SizeY >, 84
 - KalmanFilters::CKalmanSRUKF< SizeX,
SizeY >, 95
- Jcorrect_
 - KalmanFilters::CKalmanSREUKF< SizeX,
SizeY >, 85
 - KalmanFilters::CKalmanSRUKF< SizeX,
SizeY >, 95
- JcorrectBlock_
 - KalmanFilters::CKalmanSREUKFB< SizeX,
SizeY >, 88
 - KalmanFilters::CKalmanSRUKFB< SizeX,
SizeY >, 100
- Jpredict_
 - KalmanFilters::CKalmanSRUKF< SizeX,
SizeY >, 95
- K_
 - KalmanFilters::CKalmanLKF< SizeX, SizeY
>, 59
- k_sigma_points_
 - KalmanFilters::CKalmanCKF< SizeX, SizeY
>, 27
 - KalmanFilters::CKalmanUKF< SizeX, SizeY
>, 108
- Kalman_filters, 17
- KalmanFilters, 19
 - GetVersion, 20
- KalmanFilters::CKalmanCKF< SizeX, SizeY >, 21
 - ~CKalmanCKF, 24
 - CKalmanCKF, 23, 24
 - Correction, 24
 - CorrectionCKF, 24
 - dXcal_, 27
 - dYcal_, 27
 - gamma_, 27
 - k_sigma_points_, 27
 - operator=, 24, 25
- P_xy_, 27
- Prediction, 25
- PredictionCKF, 25
- SetObservationJacobianH, 26
- SetStateTransitionJacobianF, 26
- SetupDesignParametersCubatureBaseSet, 26
- SetWeightedSumMeasurementSigmas, 26
- SetWeightedSumStateSigmas, 26
- sqrt_P_chol_, 27
- swap, 26
- weightedSumMeasurementSigmas_, 27
- weightedSumStateSigmas_, 28
- weights_covariance_, 28
- weights_mean_, 28
- x_est_sigma_points_, 28
- x_pred_sigma_points_, 28
- y_pred_sigma_points_, 28
- KalmanFilters::CKalmanECKF< SizeX, SizeY >, 28
 - ~CKalmanECKF, 32
 - CKalmanECKF, 31
 - Correction, 32
 - operator=, 32
 - Prediction, 33
 - SetObservationJacobianH, 33
 - SetStateTransitionJacobianF, 33
 - swap, 33
- KalmanFilters::CKalmanEKF< SizeX, SizeY >, 34
 - ~CKalmanEKF, 36
 - CKalmanEKF, 36
 - Correction, 36
 - observationJacobianH_, 39
 - observationModel_, 40
 - operator=, 37
 - Prediction, 37
 - PredictionEKF, 38
 - SetObservationJacobianH, 38
 - SetObservationMatrixH, 38
 - SetObservationModel, 38
 - SetStateTransitionJacobianF, 38
 - SetStateTransitionJacobianLinearF, 38
 - SetStateTransitionMatrixF, 39
 - SetStateTransitionModel, 39
 - stateTransitionJacobianF_, 40
 - stateTransitionModel_, 40
 - swap, 39
- KalmanFilters::CKalmanEUKF< SizeX, SizeY >, 41
 - ~CKalmanEUKF, 44
 - CKalmanEUKF, 43, 44
 - Correction, 44
 - operator=, 44
 - Prediction, 45
 - SetObservationJacobianH, 45
 - SetStateTransitionJacobianF, 45
 - swap, 45
- KalmanFilters::CKalmanLKF< SizeX, SizeY >, 46
 - ~CKalmanLKF, 50

- CalculateInnovationCovarianceS, 50
- checkBordersMeasurement_, 58
- checkBordersStateAfterCorrection_, 58
- checkBordersStateAfterPrediction_, 58
- checkDeltaMeasurement_, 58
- checkDeltaState_, 58
- checkMatrixDiagPositive, 50
- CKalmanLKF, 49
- Correction, 51
- DeltaY_, 58
- deltaY_isSet, 59
- F_, 59
- fixMatrixMainDiagonalSymmetry, 51
- GetDeltaY, 51
- GetEstimatedCovarianceMatrixP, 51
- GetEstimatedVectorX, 51
- GetEstimatedVectorY, 52
- GetInnovationCovarianceMatrixS, 52
- GetKalmanGainMatrixK, 52
- GetSizeX, 52
- GetSizeY, 52
- H_, 59
- I_, 59
- K_, 59
- operator=, 52, 53
- P_, 59
- Prediction, 53
- prediction_isDone, 59
- Q_, 59
- R_, 60
- S_, 60
- SetCheckBordersMeasurement, 53
- SetCheckBordersStateAfterCorrection, 54
- SetCheckBordersStateAfterPrediction, 54
- SetCheckDeltaMeasurement, 54
- SetCheckDeltaState, 54
- SetDeltaY, 54
- SetEstimateCovarianceMatrixP, 55
- SetEstimateCovarianceMatrixPdiag, 55
- SetEstimatedVectorX, 55
- SetEstimatedVectorY, 55
- SetMeasuredVectorY, 56
- SetObservationCovarianceMatrixR, 56
- SetObservationCovarianceMatrixRdiag, 56
- SetObservationMatrixH, 56
- SetProcessCovarianceMatrixQ, 57
- SetProcessCovarianceMatrixQdiag, 57
- SetStateTransitionJacobianLinearF, 57
- SetStateTransitionMatrixF, 57
- SizeX_, 60
- SizeY_, 60
- stateTransitionJacobianLinearF_, 60
- swap, 58
- X_est_, 60
- X_pred_, 60
- Y_est_, 61
- Y_msd_, 61
- Y_msd_isSet, 61
- Y_pred_, 61
- KalmanFilters::CKalmanSRCKF< SizeX, SizeY >, 61
 - ~CKalmanSRCKF, 64
 - CKalmanSRCKF, 63
 - Correction, 64
 - CorrectionSRCKF, 64
 - operator=, 64, 65
 - Prediction, 65
 - SetObservationJacobianH, 65
 - SetStateTransitionJacobianF, 65
 - swap, 66
- KalmanFilters::CKalmanSRCKFB< SizeX, SizeY >, 66
 - CKalmanSRCKFB, 69
 - Correction, 69
- KalmanFilters::CKalmanSRECKF< SizeX, SizeY >, 69
 - ~CKalmanSRECKF, 71
 - CKalmanSRECKF, 71
 - Correction, 71
 - operator=, 72
 - Prediction, 72
 - swap, 72
- KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >, 73
 - ~CKalmanSRECKFB, 75
 - CKalmanSRECKFB, 74
 - Correction, 75
 - operator=, 75
 - swap, 76
- KalmanFilters::CKalmanSREKF< SizeX, SizeY >, 76
 - ~CKalmanSREKF, 78
 - CKalmanSREKF, 78
 - Correction, 79
 - operator=, 79
 - Prediction, 80
 - PredictionSREKF, 80
 - swap, 80
- KalmanFilters::CKalmanSREUKF< SizeX, SizeY >, 80
 - ~CKalmanSREUKF, 83
 - CKalmanSREUKF, 82
 - Correction, 83
 - createSignMatrices, 83
 - J_, 84
 - Jcorrect_, 85
 - operator=, 83, 84
 - Prediction, 84
 - swap, 84
- KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >, 85
 - ~CKalmanSREUKFB, 87
 - CKalmanSREUKFB, 86, 87
 - Correction, 87
 - createSignMatricesBlock, 87
 - JcorrectBlock_, 88

- operator=, 87, 88
- swap, 88
- KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, 89
 - ~CKalmanSRUKF, 91
 - CKalmanSRUKF, 91
 - Correction, 92
 - CorrectionSRUKF, 92
 - createSignMatrices, 92
 - J_, 95
 - Jcorrect_, 95
 - Jpredict_, 95
 - negativeZeroCovWeight_, 95
 - operator=, 92
 - Prediction, 93
 - SetObservationJacobianH, 93
 - SetStateTransitionJacobianF, 93
 - SetupDesignParametersCDKF, 93
 - SetupDesignParametersMeanSet, 94
 - SetupDesignParametersScaledSet, 94
 - swap, 94
- KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >, 95
 - ~CKalmanSRUKFB, 99
 - CKalmanSRUKFB, 98
 - Correction, 99
 - createSignMatricesBlock, 99
 - JcorrectBlock_, 100
 - operator=, 99
 - swap, 100
- KalmanFilters::CKalmanUKF< SizeX, SizeY >, 100
 - ~CKalmanUKF, 104
 - alpha_, 108
 - beta_, 108
 - CKalmanUKF, 103, 104
 - Correction, 104
 - CorrectionUKF, 104
 - dXcal_, 108
 - dYcal_, 108
 - gamma_, 108
 - k_sigma_points_, 108
 - kappa_, 108
 - lambda_, 109
 - operator=, 104, 105
 - P_xy_, 109
 - Prediction, 105
 - PredictionUKF, 105
 - SetObservationJacobianH, 106
 - SetStateTransitionJacobianF, 106
 - SetupDesignParametersCDKF, 106
 - SetupDesignParametersMeanSet, 106
 - SetupDesignParametersScaledSet, 106
 - SetWeightedSumMeasurementSigmas, 107
 - SetWeightedSumStateSigmas, 107
 - sqrt_P_chol_, 109
 - swap, 107
 - w0_, 109
 - weightedSumMeasurementSigmas_, 109
 - weightedSumStateSigmas_, 109
 - weights_covariance_, 109
 - weights_mean_, 109
 - x_est_sigma_points_, 110
 - x_pred_sigma_points_, 110
 - y_pred_sigma_points_, 110
- kappa_
 - KalmanFilters::CKalmanUKF< SizeX, SizeY >, 108
- lambda_
 - KalmanFilters::CKalmanUKF< SizeX, SizeY >, 109
- negativeZeroCovWeight_
 - KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, 95
- observationJacobianH_
 - KalmanFilters::CKalmanEKF< SizeX, SizeY >, 39
- observationModel_
 - KalmanFilters::CKalmanEKF< SizeX, SizeY >, 40
- operator=
 - KalmanFilters::CKalmanCKF< SizeX, SizeY >, 24, 25
 - KalmanFilters::CKalmanECKF< SizeX, SizeY >, 32
 - KalmanFilters::CKalmanEKF< SizeX, SizeY >, 37
 - KalmanFilters::CKalmanEUKF< SizeX, SizeY >, 44
 - KalmanFilters::CKalmanLKF< SizeX, SizeY >, 52, 53
 - KalmanFilters::CKalmanSRCKF< SizeX, SizeY >, 64, 65
 - KalmanFilters::CKalmanSRECKF< SizeX, SizeY >, 72
 - KalmanFilters::CKalmanSRECKFB< SizeX, SizeY >, 75
 - KalmanFilters::CKalmanSREKF< SizeX, SizeY >, 79
 - KalmanFilters::CKalmanSREUKF< SizeX, SizeY >, 83, 84
 - KalmanFilters::CKalmanSREUKFB< SizeX, SizeY >, 87, 88
 - KalmanFilters::CKalmanSRUKF< SizeX, SizeY >, 92
 - KalmanFilters::CKalmanSRUKFB< SizeX, SizeY >, 99
 - KalmanFilters::CKalmanUKF< SizeX, SizeY >, 104, 105
- P_
 - KalmanFilters::CKalmanLKF< SizeX, SizeY >, 59
- P_xy_

- KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [27](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [109](#)
 Prediction
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [25](#)
 KalmanFilters::CKalmanECKF< SizeX,
 SizeY >, [33](#)
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [37](#)
 KalmanFilters::CKalmanEUKF< SizeX,
 SizeY >, [45](#)
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [53](#)
 KalmanFilters::CKalmanSRCKF< SizeX,
 SizeY >, [65](#)
 KalmanFilters::CKalmanSRECKF< SizeX,
 SizeY >, [72](#)
 KalmanFilters::CKalmanSREKF< SizeX,
 SizeY >, [80](#)
 KalmanFilters::CKalmanSREUKF< SizeX,
 SizeY >, [84](#)
 KalmanFilters::CKalmanSRUKF< SizeX,
 SizeY >, [93](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [105](#)
 prediction_isDone
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [59](#)
 PredictionCKF
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [25](#)
 PredictionEKF
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [38](#)
 PredictionSREKF
 KalmanFilters::CKalmanSREKF< SizeX,
 SizeY >, [80](#)
 PredictionUKF
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [105](#)
 Q_
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [59](#)
 R_
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [60](#)
 S_
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [60](#)
 SetCheckBordersMeasurement
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [53](#)
 SetCheckBordersStateAfterCorrection
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [54](#)
 SetCheckBordersStateAfterPrediction
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [54](#)
 SetCheckDeltaMeasurement
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [54](#)
 SetCheckDeltaState
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [54](#)
 SetDeltaY
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [54](#)
 SetEstimateCovarianceMatrixP
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [55](#)
 SetEstimateCovarianceMatrixPdiag
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [55](#)
 SetEstimatedVectorX
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [55](#)
 SetEstimatedVectorY
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [55](#)
 SetMeasuredVectorY
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [56](#)
 SetObservationCovarianceMatrixR
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [56](#)
 SetObservationCovarianceMatrixRdiag
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [56](#)
 SetObservationJacobianH
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [26](#)
 KalmanFilters::CKalmanECKF< SizeX,
 SizeY >, [33](#)
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [38](#)
 KalmanFilters::CKalmanEUKF< SizeX,
 SizeY >, [45](#)
 KalmanFilters::CKalmanSRCKF< SizeX,
 SizeY >, [65](#)
 KalmanFilters::CKalmanSRUKF< SizeX,
 SizeY >, [93](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [106](#)
 SetObservationMatrixH
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [38](#)
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [56](#)
 SetObservationModel
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [38](#)

- SetProcessCovarianceMatrixQ
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [57](#)
- SetProcessCovarianceMatrixQdiag
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [57](#)
- SetStateTransitionJacobianF
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [26](#)
 KalmanFilters::CKalmanECKF< SizeX,
 SizeY >, [33](#)
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [38](#)
 KalmanFilters::CKalmanEUKF< SizeX,
 SizeY >, [45](#)
 KalmanFilters::CKalmanSRCKF< SizeX,
 SizeY >, [65](#)
 KalmanFilters::CKalmanSRUKF< SizeX,
 SizeY >, [93](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [106](#)
- SetStateTransitionJacobianLinearF
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [38](#)
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [57](#)
- SetStateTransitionMatrixF
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [39](#)
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [57](#)
- SetStateTransitionModel
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [39](#)
- SetupDesignParametersCDKF
 KalmanFilters::CKalmanSRUKF< SizeX,
 SizeY >, [93](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [106](#)
- SetupDesignParametersCubatureBaseSet
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [26](#)
- SetupDesignParametersMeanSet
 KalmanFilters::CKalmanSRUKF< SizeX,
 SizeY >, [94](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [106](#)
- SetupDesignParametersScaledSet
 KalmanFilters::CKalmanSRUKF< SizeX,
 SizeY >, [94](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [106](#)
- SetWeightedSumMeasurementSigmas
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [26](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [107](#)
- SetWeightedSumStateSigmas
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [26](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [107](#)
- KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [107](#)
- SizeX_
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [60](#)
- SizeY_
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [60](#)
- sqrt_P_chol_
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [27](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [109](#)
- stateTransitionJacobianF_
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [40](#)
- stateTransitionJacobianLinearF_
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [60](#)
- stateTransitionModel_
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [40](#)
- swap
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [26](#)
 KalmanFilters::CKalmanECKF< SizeX,
 SizeY >, [33](#)
 KalmanFilters::CKalmanEKF< SizeX, SizeY
 >, [39](#)
 KalmanFilters::CKalmanEUKF< SizeX,
 SizeY >, [45](#)
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [58](#)
 KalmanFilters::CKalmanSRCKF< SizeX,
 SizeY >, [66](#)
 KalmanFilters::CKalmanSRECKF< SizeX,
 SizeY >, [72](#)
 KalmanFilters::CKalmanSRECKFB< SizeX,
 SizeY >, [76](#)
 KalmanFilters::CKalmanSREKF< SizeX,
 SizeY >, [80](#)
 KalmanFilters::CKalmanSREUKF< SizeX,
 SizeY >, [84](#)
 KalmanFilters::CKalmanSREUKFB< SizeX,
 SizeY >, [88](#)
 KalmanFilters::CKalmanSRUKF< SizeX,
 SizeY >, [94](#)
 KalmanFilters::CKalmanSRUKFB< SizeX,
 SizeY >, [100](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [107](#)
- w0_
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [109](#)
- weightedSumMeasurementSigmas_
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [107](#)

KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [27](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [109](#)
 weightedSumStateSigmas_
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [28](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [109](#)
 weights_covariance_
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [28](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [109](#)
 weights_mean_
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [28](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [109](#)

 X_est_
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [60](#)
 x_est_sigma_points_
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [28](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [110](#)
 X_pred_
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [60](#)
 x_pred_sigma_points_
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [28](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [110](#)

 Y_est_
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [61](#)
 Y_msd_
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [61](#)
 Y_msd_isSet
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [61](#)
 Y_pred_
 KalmanFilters::CKalmanLKF< SizeX, SizeY
 >, [61](#)
 y_pred_sigma_points_
 KalmanFilters::CKalmanCKF< SizeX, SizeY
 >, [28](#)
 KalmanFilters::CKalmanUKF< SizeX, SizeY
 >, [110](#)