

## О МЕТОДАХ РЕАЛИЗАЦИИ UD-ФИЛЬТРА<sup>1</sup>

**Аннотация.** *Актуальность и цели.* Фильтр Калмана является математическим инструментом, завоевавшим широкую популярность среди специалистов в области оценивания и управления. Но он имеет один существенный недостаток – неустойчивость по отношению к ошибкам машинного округления при его практической реализации на ЭВМ. Проблема ошибок машинного округления является неустранимой ввиду ограниченной разрядности представления вещественных чисел с плавающей запятой на ЭВМ. Однако можно существенно уменьшить влияние ошибок машинного округления в алгебраически эквивалентных реализациях фильтра Калмана, которые называют численно эффективными реализациями. Они основаны на различных математических методах факторизации ковариационных матриц ошибок оценок, участвующих в уравнениях фильтра. Целью работы является изучение основных методов построения UD-реализаций дискретного фильтра Калмана, обладающих улучшенными вычислительными свойствами по сравнению со стандартной реализацией фильтра Калмана, а также построение новой расширенной формы ортогонализованного UD-фильтра, которая должна обладать следующими свойствами: устойчивость по отношению к ошибкам машинного округления; отсутствие операции извлечения квадратного корня; избавление от операции матричного обращения на каждой итерации алгоритма; компактность и удобство записи ортогонализованной формы UD-фильтра. *Материалы и методы.* Рассматриваются методы реализации UD-фильтров. Первой UD-реализацией фильтра Калмана является последовательный алгоритм Бирмана, а самыми современными являются ортогонализованные блочные алгоритмы. Подход к построению квадратно-корневых ортогонализованных блочных алгоритмов был предложен Кайлатом. В настоящей работе именно этот подход применяется для построения новой формы расширенного ортогонализованного UD-фильтра. *Результаты.* В работе изучены существующие к настоящему времени методы построения UD-фильтра. Наиболее эффективными в вычислительном плане и подходящими для реализации на современных вычислительных комплексах являются ортогонализованные формы UD-фильтра. Предложена новая форма расширенного ортогонализованного UD-фильтра, обладающая рядом преимуществ по сравнению с другими. *Выводы.* UD-алгоритмы в последовательной и матричной ортогонализованной формах являются эффективными в вычислительном плане реализациями дискретного фильтра Калмана. Их преимущества: 1) устойчивость по отношению к ошибкам машинного округления; 2) отсутствие операции извлечения квадратного корня; 3) избавление от операции матричного обращения на каждой итерации алгоритма; 4) компактность и удобство записи ортогонализованной формы UD-фильтра. Численные эксперименты показали работоспособность предложенной новой формы расширенного ортогонализованного UD-фильтра и ее устойчивость по отношению к ошибкам машинного округления на примерах плохо обусловленных задач.

**Ключевые слова:** стохастические дискретные линейные системы, оптимальная дискретная фильтрация, фильтр Калмана, устойчивые реализации дискретного фильтра, UD-фильтр.

<sup>1</sup> Работа выполнена в рамках Государственных заданий Министерства образования и науки РФ (шифры 1.919.2011 и 6.3072.2011), поддержана Грантом РФФИ (шифр 13-01-97035).

## ON THE UD FILTER IMPLEMENTATION METHODS

**Abstract.** *Background.* The Kalman filter (KF) is a mathematical tool that has won wide popularity among professionals in the field of estimation and control. However, it has a drawback – instability in respect to machine round off errors in its practical implementation on a computer. It is well known that the problem of machine round off errors is unavoidable due to the limited machine width of real floating-point numbers. However, one can significantly reduce the effect of round off errors in algebraically equivalent Kalman filter implementations (the so-called numerically efficient implementations). They are based on different mathematical factorization methods of the estimation error covariance matrices involved in the filter equations. The aim of the paper is to study the basic UD implementation methods of the discrete Kalman Filter with improved computational properties in comparison with the standard KF implementation, as well as the construction of the new extended orthogonal form of the UD filter which should have the following properties: robustness of computations against round off errors; lack of the square-root operation; deliverance from matrix inverse operation on each stage of the algorithm; compact and convenient orthogonal form of the UD filter. *Materials and methods.* The paper is based on the UD filter implementation methods. The first UD implementation of the KF was the Bierman's sequential algorithm. Matrix orthogonal algorithms are the most modern numerically efficient KF implementations. The approach to the construction of the square-root orthogonal algorithms was proposed by Kailath and it is used in the paper to construct a new form of the extended orthogonal UD filter. *Results.* In the paper the existing implementation methods of UD filter have been investigated. The orthogonal forms of the UD filter are the most computationally efficient and suitable for implementation in modern computer systems. A new form of the extended orthogonal UD filter, which has a number of advantages in comparison with others, has been proposed. *Conclusions.* The sequential and orthogonal forms of the UD algorithms are the most numerically efficient implementations of the discrete Kalman filter. Their benefits are: 1) robustness of computations against round off errors; 2) lack of the square-root operation; 3) deliverance from matrix inverse operation on each stage of the algorithm; 4) compact and convenient orthogonal form of the UD filter. The numerical experiments have shown the efficiency of the proposed new form of extended orthogonal UD filter and its computations robustness against round off errors while dealing with ill-conditioned tasks.

**Key words:** stochastic discrete linear systems, optimal discrete filtering, the Kalman filter, robust implementations of the discrete filter, the UD filter.

### Введение

В начале 1960-х гг. Рудольф Эмиль Калман [1] предложил свое знаменитое решение задачи оптимального линейного оценивания, и это решение в дальнейшем получило название *фильтр Калмана* (ФК). Первоначально ФК был построен для решения задач из области авиации, однако со временем он стал применяться в самых различных областях науки и техники, таких как эконометрика, молекулярная биология, метеорология, спутниковая геодезия, телекоммуникационные сети, обработка изображений и многих других.

Алгоритм фильтрации Калмана является изящным математическим инструментом, завоевавшим широкую популярность среди специалистов в области оценивания и управления, но, идеальный в теоретическом плане, он

имеет один существенный недостаток – неустойчивость по отношению к ошибкам машинного округления при его практической реализации на ЭВМ.

Хорошо известно, что проблема ошибок машинного округления является неустранимой ввиду ограниченной разрядности представления вещественных чисел с плавающей запятой на ЭВМ. Однако можно существенно уменьшить влияние ошибок машинного округления в алгебраически эквивалентных реализациях ФК, которые назовем *численно эффективными реализациями*. Первые численно эффективные реализации ФК появились уже вскоре после его открытия. Они основаны на различных математических методах факторизации ковариационных матриц ошибок оценок, участвующих в уравнениях фильтра. Первым таким алгоритмом явился квадратно-корневой фильтр Поттера [2], разработанный для программы Аполлон.

В дальнейшем эти и другие подобные методы стали отправной точкой для развития целого направления в теории линейного оценивания и управления, которое продолжает активно развиваться и в настоящее время. Существует огромное количество научных публикаций по данной теме. Здесь отметим только хорошо известные автору монографии, содержащие описание численно эффективных реализаций ФК: зарубежные [3–6] и отечественные [7–9].

В работе рассматриваются методы реализации UD-фильтров. Особенность данной группы численно эффективных реализаций дискретного ФК заключается в том, что все они основаны на представлении ковариационной матрицы ошибок оценок вектора состояния  $P$  в виде произведения  $U_P D_P U_P^T$ , где  $U$  – верхняя треугольная матрица с единицами на диагонали,  $D$  – диагональная матрица (для любой квадратной положительно определенной матрицы такое представление можно получить в результате модифицированного разложения Холецкого [3, 6]). Первой UD-реализацией ФК является последовательный алгоритм Дж. Бирмана [3]. Бирман не только доказал вычислительную эффективность UD-фильтра, но и показал, что при соответствующей программной реализации его алгоритм не сложнее, чем стандартный алгоритм Калмана [10].

Из всех численно эффективных реализаций ФК самыми современными являются ортогонализированные блочные алгоритмы. Подход к построению квадратно-корневых ортогонализированных блочных алгоритмов был предложен Т. Кайлатом (подробное описание см. в [5]). В настоящей работе именно этот подход применяется для построения новой формы расширенного ортогонализированного UD-фильтра.

### **1. Постановка задачи**

Рассмотрим стохастическую дискретную линейную систему:

$$x_{k+1} = \Phi_k x_k + G_k w_k, \quad w_k \in N(0, Q_k), \quad (1)$$

$$z_k = H_k x_k + v_k, \quad v_k \in N(0, R_k), \quad k \geq 0, \quad (2)$$

где  $x_k \in \mathbf{R}^n$  – вектор состояния системы,  $z_k \in \mathbf{R}^m$  – доступный вектор измерений. Последовательности  $\{w_0, w_1, \dots\}$  и  $\{v_1, v_2, \dots\}$  – независимые нормально распределенные белые последовательности шумов с нулевыми

средними и ковариационными матрицами  $Q_k \geq 0$  и  $R_k > 0$  соответственно,  $G_k \in \mathbf{R}^{n \times s}$ . Последовательности  $\{w_k\}$  и  $\{v_k\}$  не зависят от начального состояния  $x_0 \in N(\bar{x}_0, \Pi_0)$ . Другими словами, выполняется следующее условие:

$$\mathbf{E} \begin{bmatrix} w_k \\ v_k \\ x_0 \end{bmatrix} \begin{bmatrix} w_j^T & v_j^T & x_0^T & 1 \end{bmatrix} = \begin{bmatrix} Q_k & 0 & & \\ 0 & R_k & & \\ & & \delta_{kj} & 0 & 0 \\ 0 & 0 & & \Pi_0 & \bar{x}_0 \end{bmatrix},$$

где  $\delta_{kj}$  – символ Кронекера;  $\mathbf{E}[\cdot]$  – символ математического ожидания.

Для решения задач рекуррентного оценивания и адаптивной фильтрации [5, 7, 11], параметрической идентификации [12], стохастического управления [4, 13, 14] и многих других применяется широко известный со второй половины прошлого века математический инструмент – дискретный фильтр Калмана [1].

Стандартная реализация дискретного ФК состоит из двух этапов: экстраполяции (предсказания оценки вектора состояния и матрицы ковариации ошибки оценивания по доступным измерениям) и фильтрации (обновления предсказанной оценки вектора состояния и матрицы ковариации ошибки оценивания в соответствии с полученным в текущий момент времени измерением). Такую форму дискретного ФК назовем *двухстадийной формой* стандартного фильтра.

**Алгоритм 1** (двухстадийная форма стандартной реализации дискретного ФК).

0. НАЧАЛЬНЫЕ ДАННЫЕ:  $\hat{x}_{0|0} = \bar{x}_0$  и  $P_{0|0} = \Pi_0$ ,  $\Pi_0 > 0$ .

1. РЕКУРРЕНТНО ОБНОВЛЯТЬ ВЕЛИЧИНЫ ( $k \geq 0$ ).

1.1. НА ЭТАПЕ ФИЛЬТРАЦИИ:

$$K_k = P_k H_k^T R_{e,k}^{-1}, R_{e,k} = R_k + H_k P_k H_k^T \text{ (коэффициент Калмана),}$$

$$\hat{x}_{k|k} = \hat{x}_k + K_k e_k, e_k = z_k - H_k \hat{x}_k \text{ (оценка),}$$

$$P_{k|k} = P_k - K_k H_k P_k \text{ (матрица ковариации ошибки).}$$

1.2. НА ЭТАПЕ ЭКСТРАПОЛЯЦИИ:

$$\hat{x}_{k+1} = \Phi_k \hat{x}_{k|k} \text{ (оценка),}$$

$$P_{k+1} = \Phi_k P_{k|k} \Phi_k^T + G_k Q_k G_k^T \text{ (матрица ковариации ошибки),} \quad (3)$$

где  $\hat{x}_k \equiv \hat{x}_{k|k-1}$  – линейная среднеквадратичная экстраполированная оценка вектора состояния  $x_k$ , вычисленная по доступным измерениям  $\{z_0, \dots, z_{k-1}\}$ ,

$$P_k \equiv P_{k|k-1} = \mathbf{E}[(x_k - \hat{x}_{k|k-1})(x_k - \hat{x}_{k|k-1})^T] -$$

ковариационная матрица ошибки оценивания на этапе экстраполяции;  $\hat{x}_{k|k}$  – линейная среднеквадратичная отфильтрованная по текущему измерению  $z_k$

оценка вектора состояния,  $P_{k|k} = \mathbf{E}[(x_k - \hat{x}_{k|k})(x_k - \hat{x}_{k|k})^T]$  – ковариационная матрица ошибки оценивания на этапе фильтрации.

Если нас интересует только экстраполированная оценка вектора состояния и соответствующая ей ковариационная матрица ошибки оценивания, то алгоритм 1 можно переписать в удобной *одностадийной форме* [5].

**Алгоритм 2** (одностадийная форма стандартной реализации дискретного ФК).

0. НАЧАЛЬНЫЕ ДАННЫЕ:  $\hat{x}_{0|0} = \bar{x}_0$  и  $P_{0|0} = \Pi_0$ ,  $\Pi_0 > 0$ .

1. РЕКУРРЕНТНО ОБНОВЛЯТЬ ВЕЛИЧИНЫ ( $k \geq 0$ ):

$$K_{p,k} = \Phi_k P_k H_k^T R_{e,k}^{-1}, R_{e,k} = R_k + H_k P_k H_k^T \text{ (матрица обратной связи),} \quad (4)$$

$$P_{k+1} = \Phi_k P_k \Phi_k^T + G_k Q_k G_k^T - K_{p,k} R_{e,k} K_{p,k}^T \text{ (матрица ковариации ошибки),} \quad (5)$$

$$\hat{x}_{k+1} = \Phi_k \hat{x}_k + K_{p,k} e_k, e_k = z_k - H_k \hat{x}_k \text{ (оценка).} \quad (6)$$

Далее в работе будем использовать следующие обозначения: для невырожденной квадратной матрицы  $(A^{-1})^T = (A^T)^{-1} = A^{-T}$ ; через  $D$  обозначим диагональную матрицу, через  $U$  и  $L$  – соответственно верхнюю и нижнюю треугольную матрицу с единицами на диагонали. Используем модифицированное UD-разложение Холецкого [3, 6] для положительно определенной квадратной матрицы  $P = U_P D_P U_P^T$ .

Целью работы является изучение основных методов построения UD-реализаций дискретного ФК, обладающих улучшенными вычислительными свойствами по сравнению с алгоритмами 1 и 2, а также построение новой расширенной формы ортогонализованного UD-фильтра, которая должна обладать следующими свойствами:

- устойчивость по отношению к ошибкам машинного округления;
- отсутствие операции извлечения квадратного корня;
- избавление от операции матричного обращения на каждой итерации алгоритма;
- компактность и удобство записи ортогонализованной формы UD-фильтра.

## 2. Методы реализации UD-фильтра

Перечислим базовые идеи, на которых построены различные формы UD-реализации дискретного ФК:

1. Дж. Бирман (G. Bierman) [3] предложил использовать UD-факторы ковариационной матрицы ошибки оценивания вектора состояния для обновления разностного уравнения Риккати на этапе обработки измерений в фильтре Калмана.

2. К. Торнтон (C. Thornton) [15] применила алгоритм взвешенной ортогонализации Грама – Шмидта для вычисления UD-факторов ковариационной матрицы на этапе экстраполяции в фильтре Калмана.

3. Т. Кайлат (T. Kailath) [16] построил новый класс алгоритмов для реализации дискретного ФК – квадратно-корневые *ортогонализованные мат-*

ричные алгоритмы (*array algorithms*). Данные алгоритмы базируются на применении ортогональных преобразований к блочной матрице, называемой *pre-array*. В результате получается новая блочная матрица, называемая *post-array*, с готовыми результатами вычислений. Отличительной чертой в таких алгоритмах является удобная форма представления данных для реализации на ЭВМ, ориентированность на параллельные вычисления [17, 18]. Кроме того, ортогональные преобразования обладают численной устойчивостью по отношению к ошибкам машинного округления.

4. Т. Кайлат [17] предложил ортогонализированный алгоритм этапа обработки измерения в ФК, основанный на быстрых вращениях Гивенса. Было показано, что данный алгоритм эквивалентен алгоритму Бирмана для случая скалярных измерений.

5. В книге [6] авторами сформулирован одностадийный ортогонализированный алгоритм для UD-фильтра, основанный на модифицированной взвешенной ортогонализации Грама – Шмидта.

### 3.1. Последовательный алгоритм Бирмана

Пусть на этапе фильтрации в алгоритме 1 используются разложения

$$P_k = U_{P_k} D_{P_k} U_{P_k}^T \text{ и } P_{k|k} = U_{P_{k|k}} D_{P_{k|k}} U_{P_{k|k}}^T.$$

Предположим, что элементы матрицы  $D_{P_k}$  являются положительными, а матрица  $R_k$  – диагональная, т.е.  $R_k = \text{Diag}[r_k(1), \dots, r_k(m)]$ . Тогда этап фильтрации алгоритма 1 эквивалентен следующему алгоритму:

**Алгоритм 3** (последовательный алгоритм Бирмана).

0. НАЧАЛЬНОЕ ПРИСВАИВАНИЕ:  $\tilde{U} = U_{P_k}$ ,  $\tilde{D} = D_{P_k}$ ,  $\tilde{x} = \hat{x}_k$ .

1.  $m$ -КРАТНОЕ ПОВТОРЕНИЕ ПРОЦЕДУРЫ СКАЛЯРНОГО ОБНОВЛЕНИЯ.

Для  $j = 1, 2, \dots, m$  выполнять:

1.1. Вычислить векторы

$$f = [f_1, f_2, \dots, f_n]^T = \tilde{U}^T h;$$

$$v = [v_1, v_2, \dots, v_n]^T = \tilde{D}f.$$

1.2. Задать начальные значения  $\alpha' = r + v_1 f_1$ ,  $\hat{d}_1 = \tilde{d}_1 r / \alpha'$ ;  
 $K = [v_1 | 0 \dots 0]^T$ .

1.3. Для  $i = 2, \dots, n$  выполнять:

начало

$$\alpha := \alpha' + v_i f_i; \quad \gamma := 1 / \alpha;$$

$$\hat{d}_i := \tilde{d}_i \alpha' \gamma; \quad \lambda := -f_i \gamma;$$

$$\hat{u}_i := \tilde{u}_i + \lambda K; \quad K := K + v_i \tilde{u}_i;$$

$$\alpha' := \alpha.$$

конец

1.4. Вычислить векторы  $v := \gamma(z - h^T \tilde{x})$ ;  $\hat{x} := \tilde{x} + Kv$  с экстраполяцией между повторениями:  $\tilde{U} = \hat{U}$ ;  $\tilde{D} = \hat{D}$ ;  $\tilde{x} = \hat{x}$ .

2. ЗАВЕРШАЮЩЕЕ ПРИСВАИВАНИЕ:  $U_{P_{k|k}} = \hat{U}$ ;  $D_{P_{k|k}} = \hat{D}$ ;  $\hat{x}_{k|k} = \hat{x}$ .

Здесь  $h - j$ -й столбец матрицы  $H_k^T$ ;  $z - j$ -й элемент вектора  $z_k$ ;  $r - j$ -й элемент  $r_j(k)$  диагональной матрицы ковариаций шума измерений  $R_k$ ,  $j = 1, \dots, m$  – номер скалярного измерения в составе вектора измерений  $z_k$  в момент времени  $k$ . Доказательство алгоритма можно найти в [3, с. 77–80].

Алгоритм 3 не содержит операции извлечения квадратного корня, а работа с треугольными матрицами требует меньшего числа арифметических операций по сравнению с обычными. Однако он неудобен для реализации на многопроцессорных вычислительных системах в силу своей последовательной, а не параллельной структуры.

### 3.2. Алгоритм Торнтон на базе MWGS-ортогонализации

Предположим, что на этапе экстраполяции доступны UD-факторы ковариационных матриц  $P_{k|k} = U_{P_{k|k}} D_{P_{k|k}} U_{P_{k|k}}^T$  и  $Q_k = U_{Q_k} D_{Q_k} U_{Q_k}^T$ . Тогда уравнение (3) принимает вид

$$P_{k+1} = \Phi_k U_{P_{k|k}} D_{P_{k|k}} U_{P_{k|k}}^T \Phi_k^T + G_k U_{Q_k} D_{Q_k} U_{Q_k}^T G_k^T.$$

Представим матрицу  $P_{k+1}$  как

$$P_{k+1} = [\Phi_k U_{P_{k|k}} | G_k U_{Q_k}] \text{Diag}\{D_{P_{k|k}}, D_{Q_k}\} \begin{bmatrix} U_{P_{k|k}}^T & \Phi_k^T \\ U_{Q_k}^T & G_k^T \end{bmatrix},$$

т.е.  $P_{k+1} = W_k D_k W_k^T$ ,

$$W_k = [\Phi_k U_{P_{k|k}} | G_k U_{Q_k}] = \begin{bmatrix} w_1^T(k) \\ \dots \\ w_n^T(k) \end{bmatrix},$$

$$D_k = \text{Diag}\{D_{P_{k|k}}, D_{Q_k}\} = \text{Diag}\{d_1(k), \dots, d_M(k)\},$$

где  $W_k$  – матрица размера  $n \times (n + s)$  и  $M = n + s$ ,  $s$  – размер вектора шума  $w_k$  в уравнении состояния (1). Факторы  $U_{P_{k+1}}$  и  $D_{P_{k+1}}$  вычисляются методом модифицированной взвешенной ортогонализации Грама – Шмидта ([3], теорема VI.4.1) на этапе экстраполяции (описание метода MWGS приведено в приложении). Сформулируем алгоритм Торнтон.

**Алгоритм 4** (алгоритм Торнтон для этапа экстраполяции в ФК).

0. НАЧАЛЬНЫЕ ДАННЫЕ:  $\hat{x}_{k|k}$ ,  $\{U_{P_{k|k}}, D_{P_{k|k}}\}$ ,  $\{U_{Q_k}, D_{Q_k}\}$ .

1. ВЫЧИСЛИТЬ ФАКТОРЫ  $\{U_{P_{k+1}}, D_{P_{k+1}}\}$ :

1.1. Сформировать блочные матрицы

$$D_k = \text{Diag}\{D_{P_{k|k}}, D_{Q_k}\}, \quad W_k = [\Phi_{k|k} U_{P_{k|k}} \mid G_k U_{Q_k}].$$

1.2. Для пары  $\{W_k, D_k\}$  провести MWGS-ортогонализацию строк матрицы  $W_k$  относительно весовой матрицы  $D_k$  и получить в результате пару  $\{U_{P_{k+1}}, D_{P_{k+1}}\}$ .

1.3. Вычислить оценку вектора состояния:  $\hat{x}_{k+1} = \Phi_k \hat{x}_{k|k}$ .

Доказательство алгоритма Торнтон можно найти в [3, с. 127–128]. Этот алгоритм также не содержит операции извлечения квадратного корня, а однородная структура алгоритма подходит для реализации на современных вычислительных комплексах.

### 3.3. Одностадийная ортогонализованная форма UD-фильтра

Т. Кайлат [17] предложил ортогонализированный алгоритм этапа обработки измерения в ФК, основанный на быстрых вращениях Гивенса [19]. Алгоритм был сформулирован для LD-факторов ковариационных матриц.

Предположим, что на этапе обработки измерения доступны LD-факторы ковариационных матриц  $P_k = L_{P_k} D_{P_k} L_{P_k}^T$  и  $R_k = L_{R_k} L_{R_k} L_{R_k}^T$ . Тогда LD-факторы  $L_{P_{k|k}}$  и  $D_{P_{k|k}}$  ковариационной матрицы  $P_{k|k}$  можно получить в результате матричного ортогонального преобразования

$$\begin{bmatrix} L_{R_k} & H_k L_{P_k} \\ 0 & L_{P_k} \end{bmatrix} \begin{bmatrix} D_{R_k} & 0 \\ 0 & D_{P_k} \end{bmatrix}^{1/2} Q = \begin{bmatrix} L_{R_{e,k}} & 0 \\ K_k L_{R_{e,k}} & L_{P_{k|k}} \end{bmatrix} \begin{bmatrix} D_{R_{e,k}} & 0 \\ 0 & D_{P_{k|k}} \end{bmatrix}^{1/2}, \quad (7)$$

где  $Q$  – матрица ортогонального преобразования;  $L_{R_{e,k}}$  и  $D_{R_{e,k}}$  – LD-факторы ковариационной матрицы невязки измерений  $R_{e,k}$ ;  $K_k$  – матрица Калмана.

Если вместо  $Q$  использовать  $Q_D$  матрицу преобразования на основе быстрых вращений Гивенса, не содержащих операцию извлечения квадратного корня, то соотношение (7) можно переписать в виде

$$\begin{bmatrix} L_{R_k} & H_k L_{P_k} \\ 0 & L_{P_k} \end{bmatrix} Q_D = \begin{bmatrix} L_{R_{e,k}} & 0 \\ K_k L_{R_{e,k}} & L_{P_{k|k}} \end{bmatrix}.$$

В работе [17] было показано, что данный алгоритм эквивалентен LD-реализации алгоритма Бирмана для случая скалярных измерений.

На базе алгоритмов Т. Кайлата в книге [6, с. 262] сформулирован одностадийный ортогонализированный алгоритм для UD-фильтра. Вместо быстрых вращений Гивенса применяется модифицированная взвешенная



ортогонализация Грама – Шмидта. Запишем указанный алгоритм для системы (1), (2).

**Алгоритм 5** (одностадийная ортогонализованная форма UD-фильтра).

0. НАЧАЛЬНЫЕ ДАННЫЕ:  $\hat{x}_0 = 0$  и  $P_0 = \Pi_0 > 0$ . Вычислить UD-факторы Холецкого:  $\{U_{\Pi_0}, D_{\Pi_0}\}$ ,  $\{U_{R_0}, D_{R_0}\}$ ,  $\{U_{Q_0}, D_{Q_0}\}$ .

1. РЕКУРРЕНТНО ОБНОВЛЯТЬ  $\{U_{P_{k+1}}, D_{P_{k+1}}\}$  и  $\hat{x}_{k+1}$  ( $k > 0$ ):

1.1. Для  $\{U_{P_k}, D_{P_k}\}$ ,  $\{U_{R_k}, D_{R_k}\}$ ,  $\{U_{Q_k}, D_{Q_k}\}$  сформировать блочные матрицы

$$\mathbf{D}_k = \text{Diag}\{D_{Q_k}, D_{P_k}, D_{R_k}\}, \quad \mathbf{A}_k^T = \begin{bmatrix} G_k U_{Q_k} & \Phi_k U_{P_k} & 0 \\ 0 & H_k U_{P_k} & U_{R_k} \end{bmatrix}.$$

1.2. Для пары  $\{\mathbf{A}_k, \mathbf{D}_k\}$  провести MWGS-ортогонализацию столбцов матрицы  $\mathbf{A}_k$  относительно весовой матрицы  $\mathbf{D}_k$  и получить в результате пару блочных матриц  $\{\mathbf{A}_k^\dagger, \mathbf{D}_k^\dagger\}$ :

$$\mathbf{D}_k^\dagger = \text{Diag}\{D_{P_{k+1}}, D_{R_{e,k}}\}, \quad \mathbf{A}_k^\dagger = \begin{bmatrix} U_{P_{k+1}} & K_{p,k} U_{R_{e,k}} \\ 0 & U_{R_{e,k}} \end{bmatrix}$$

таких, что

$$\mathbf{A}_k^T = \mathbf{A}_k^\dagger B_k^T \quad \text{и} \quad \mathbf{A}_k^T \mathbf{D}_k \mathbf{A}_k = \mathbf{A}_k^\dagger \mathbf{D}_k^\dagger (\mathbf{A}_k^\dagger)^T,$$

где  $\mathbf{D}_k \in \mathbf{R}^{(n+m+s) \times (n+m+s)}$ ,  $\mathbf{A}_k \in \mathbf{R}^{(n+m+s) \times (n+m)}$ ,  $B_k \in \mathbf{R}^{(n+m+s) \times (n+m)}$  – матрица MWGS-преобразования, приводящего к верхней треугольной блочной матрице  $\mathbf{A}_k^\dagger \in \mathbf{R}^{(n+m) \times (n+m)}$  и диагональной матрице  $\mathbf{D}_k^\dagger \in \mathbf{R}^{(n+m) \times (n+m)}$ .

1.3. Вычислить оценку вектора состояния:

$$\hat{x}_{k+1} = \Phi_k \hat{x}_k + (K_{p,k} U_{R_{e,k}}) \bar{e}_k, \quad \text{где} \quad \bar{e}_k = U_{R_{e,k}}^{-1} e_k, \quad e_k = z_k - H \hat{x}_k. \quad (8)$$

Отметим, что величина  $K_{p,k} U_{R_{e,k}}$  (в круглых скобках в (8)) получена непосредственно из  $\mathbf{A}_k^\dagger$ .

Алгоритм 5 позволяет вычислить те же величины, что и алгоритм 2, но, в отличие от последнего, является устойчивым по отношению к ошибкам машинного округления. Вместе с тем он обладает следующими недостатками:

1) вычисление оценки вектора состояния по уравнению (8) происходит отдельно от обработки остальных данных, которые входят в блочные матрицы и обрабатываются с помощью MWGS-алгоритма. Таким образом, алгоритм имеет неоднородную структуру;

2) в уравнении (8) присутствует операция матричного обращения, которая может ухудшить качество оценок вектора состояния.

Для преодоления этих недостатков разработаем на базе алгоритма 5 расширенную ортогонализированную форму UD-фильтра, в которой все данные обрабатываются единообразно с помощью устойчивого к ошибкам машинного округления алгоритма MWGS-ортогонализации. Кроме того, новый алгоритм должен обладать всеми свойствами, указанными в разделе 2.

#### 4. Расширенная ортогонализированная форма UD-фильтра

Идея расширенного ортогонализированного алгоритма заключается в том, что все величины дискретного фильтра помещаются в блочную матрицу и обрабатываются с помощью ортогональных преобразований. Впервые эта идея была реализована в [20] для построения квадратно-корневого информационного фильтра (SRIF – square-root information filter). Для ковариационной формы ФК расширенные квадратно-корневые ортогонализированные алгоритмы построены в [16].

Рассмотрим снова представление ковариационной матрицы в виде

$$P_k = U_{P_k} D_{P_k} U_{P_k}^T.$$

Введем обозначения:

$$\hat{z}_k = (U_{P_k} D_{P_k})^{-1} \hat{x}_k,$$

$$b_k = -(U_{R_{e,k}} D_{R_{e,k}})^{-1} e_k.$$

Сформулируем алгоритм, реализующий расширенную ортогонализированную форму UD-фильтра.

**Алгоритм 6** (расширенная ортогонализированная форма UD-фильтра).

0. НАЧАЛЬНЫЕ ДАННЫЕ:  $\hat{z}_0 = 0$  и  $P_0 = \Pi_0 > 0$ . Вычислить UD-факторы Холецкого:  $\{U_{\Pi_0}, D_{\Pi_0}\}$ ,  $\{U_{R_0}, D_{R_0}\}$ ,  $\{U_{Q_0}, D_{Q_0}\}$ .

1. РЕКУРРЕНТНО ОБНОВЛЯТЬ  $\{U_{P_{k+1}}, D_{P_{k+1}}\}$  и  $\hat{z}_{k+1}$  ( $k > 0$ ):

1.1. Для  $\{U_{P_k}, D_{P_k}\}$ ,  $\{U_{R_k}, D_{R_k}\}$ ,  $\{U_{Q_k}, D_{Q_k}\}$ ,  $\hat{z}_k$  сформировать блочные матрицы

$$\mathbf{D}_k = \text{Diag}\{D_{Q_k}, D_{P_k}, D_{R_k}\},$$

$$\mathbf{A}_k^T = \begin{bmatrix} 0 & \hat{z}_k^T & -z_k^T (U_{R_k} D_{R_k})^{-T} \\ G_k U_{Q_k} & \Phi_k U_{P_k} & 0 \\ 0 & H_k U_{P_k} & U_{R_k} \end{bmatrix}. \quad (9)$$

1.2. Для пары  $\{\mathbf{A}_k, \mathbf{D}_k\}$  провести MWGS-ортогонализацию столбцов матрицы  $\mathbf{A}_k$  относительно весовой матрицы  $\mathbf{D}_k$  и получить в результате пару блочных матриц  $\{\mathbf{A}_k^\dagger, \mathbf{D}_k^\dagger\}$ :

$$\mathbf{D}_k^\dagger = \text{Diag}\{1, D_{P_{k+1}}, D_{R_{e,k}}\}, \quad \mathbf{A}_k^\dagger = \begin{bmatrix} (*) & \hat{z}_{k+1}^\top & b_k^\top \\ 0 & U_{P_{k+1}} & K_{p,k} U_{R_{e,k}} \\ 0 & 0 & U_{R_{e,k}} \end{bmatrix} \quad (10)$$

таких, что

$$\mathbf{A}_k^\top = \mathbf{A}_k^\dagger \mathbf{B}_k^\top$$

и

$$\mathbf{A}_k^\top \mathbf{D}_k \mathbf{A}_k = \mathbf{A}_k^\dagger \mathbf{D}_k^\dagger (\mathbf{A}_k^\dagger)^\top,$$

где  $\mathbf{D}_k \in \mathbf{R}^{(n+m+s) \times (n+m+s)}$ ,  $\mathbf{A}_k \in \mathbf{R}^{(n+m+s) \times (1+n+m)}$ ,  $\mathbf{B}_k \in \mathbf{R}^{(n+m+s) \times (1+n+m)}$  – матрица MWGS-преобразования, приводящего к верхней треугольной блочной матрице  $\mathbf{A}_k^\dagger \in \mathbf{R}^{(1+n+m) \times (1+n+m)}$  и диагональной матрице  $\mathbf{D}_k^\dagger \in \mathbf{R}^{(1+n+m) \times (1+n+m)}$ ; (\*) – величина, не представляющая для нас интерес.

*Замечание.* В любой момент времени  $k$  доступна оценка вектора состояния

$$\hat{x}_k = (U_{P_k} D_{P_k}) \hat{z}_k. \quad (11)$$

*Доказательство алгоритма б.* Представим произведение  $\mathbf{A}_k^\top \mathbf{D}_k \mathbf{A}_k$  в виде

$$\mathbf{A}_k^\top \mathbf{D}_k \mathbf{A}_k = \begin{bmatrix} (*) & \bar{A}_{21}^\top & \bar{A}_{31}^\top \\ \bar{A}_{21} & \bar{A}_{22} & \bar{A}_{32}^\top \\ \bar{A}_{31} & \bar{A}_{32} & \bar{A}_{33} \end{bmatrix}. \quad (12)$$

Определим элементы блочной матрицы (12):

$$\bar{A}_{21} = \Phi_k U_{P_k} D_{P_k} \hat{z}_k = \Phi \hat{x}_k;$$

$$\bar{A}_{22} = G_k U_{Q_k} D_{Q_k} U_{Q_k}^\top G_k^\top + \Phi_k U_{P_k} D_{P_k} U_{P_k}^\top \Phi_k^\top = G_k Q_k G_k^\top + \Phi_k P_k \Phi_k^\top;$$

$$\bar{A}_{31} = H_k U_{P_k} D_{P_k} \hat{z}_k - z_k = H_k \hat{x}_k - z_k = -e_k;$$

$$\bar{A}_{32} = H_k U_{P_k} D_{P_k} U_{P_k}^\top \Phi_k^\top = H_k P_k \Phi_k^\top;$$

$$\bar{A}_{33} = H_k U_{P_k} D_{P_k} U_{P_k}^\top + U_{R_k} D_{R_k} U_{R_k}^\top = H_k P_k H_k^\top + R_k.$$

Аналогично представим произведение  $\mathbf{A}_k^\dagger \mathbf{D}_k^\dagger (\mathbf{A}_k^\dagger)^\top$  в виде

$$\mathbf{A}_k^\dagger \mathbf{D}_k^\dagger (\mathbf{A}_k^\dagger)^\top = \begin{bmatrix} (*) & \tilde{A}_{21}^\top & \tilde{A}_{31}^\top \\ \tilde{A}_{21} & \tilde{A}_{22} & \tilde{A}_{32}^\top \\ \tilde{A}_{31} & \tilde{A}_{32} & \tilde{A}_{33} \end{bmatrix}. \quad (13)$$

Определим элементы блочной матрицы (13):

$$\begin{aligned}\tilde{A}_{21} &= U_{P_{k+1}} D_{P_{k+1}} \hat{z}_{k+1} + K_{p,k} U_{R_{e,k}} D_{R_{e,k}} b_k = \\ &= \hat{x}_{k+1} - K_{p,k} U_{R_{e,k}} D_{R_{e,k}} (U_{R_{e,k}} D_{R_{e,k}})^{-1} e_k = \hat{x}_{k+1} - K_{p,k} e_k;\end{aligned}$$

$$\begin{aligned}\tilde{A}_{22} &= U_{P_{k+1}} D_{P_{k+1}} U_{P_{k+1}}^T + K_{p,k} U_{R_{e,k}} D_{R_{e,k}} U_{R_{e,k}}^T K_{p,k}^T = \\ &= P_{k+1} + K_{p,k} R_{e,k} K_{p,k}^T;\end{aligned}$$

$$\tilde{A}_{31} = U_{R_{e,k}} D_{R_{e,k}} b_k = -e_k;$$

$$\tilde{A}_{32} = U_{R_{e,k}} D_{R_{e,k}} U_{R_{e,k}}^T K_{p,k}^T = R_{e,k} K_{p,k}^T;$$

$$\tilde{A}_{33} = U_{R_{e,k}} D_{R_{e,k}} U_{R_{e,k}}^T = R_{e,k}.$$

Сравнивая левую (12) и правую (13) части равенства  $\mathbf{A}_k^T \mathbf{D}_k \mathbf{A}_k = \mathbf{A}_k^\dagger \mathbf{D}_k^\dagger (\mathbf{A}_k^\dagger)^T$ , приходим к соотношениям:

$$\bar{A}_{21} = \tilde{A}_{21} \Rightarrow \Phi \hat{x}_k = \hat{x}_{k+1} - K_{p,k} e_k \Rightarrow (6);$$

$$\bar{A}_{22} = \tilde{A}_{22} \Rightarrow G_k Q_k G_k^T + \Phi_k P_k \Phi_k^T = P_{k+1} + K_{p,k} R_{e,k} K_{p,k}^T \Rightarrow (5);$$

$$\bar{A}_{31} = \tilde{A}_{31}; \quad \bar{A}_{32} = \tilde{A}_{32};$$

$$\bar{A}_{33} = \tilde{A}_{33} \Rightarrow H_k P_k H_k^T + R_k = R_{e,k} \Rightarrow (4),$$

что и доказывает справедливость алгоритма 6.

Построенный алгоритм 6 обладает заявленными в разделе 2 свойствами. Он имеет компактную и удобную форму записи, позволяющую единообразно обрабатывать все данные. Такая форма UD-фильтра позволяет предельно просто реализовать его в виде программного кода на ЭВМ. В алгоритме отсутствует операция извлечения квадратного корня, а вычисление оценки вектора состояния (11) теперь не требует операции матричного обращения.

*Замечание 2.* Для того чтобы избавиться от матричного обращения при вычислении элемента блочной матрицы  $-z_k^T (U_{R_k} D_{R_k})^{-T}$  в (10), можно заменить его на  $-(z_k^*)^T$ , где  $z_k^*$  – декоррелированные измерения (см. [9, с. 240]).

## 5. Численные примеры

Рассмотрим прикладную задачу [9] получения оценок высоты и вертикальной скорости летательного аппарата (ЛА) по показаниям двух приборов: инерциального датчика вертикального ускорения  $a_y(t)$  и барометрического датчика высоты, который обладает собственной инерционностью в своих показаниях  $h(t)$ , характеризуемой постоянной времени  $\tau$ .

Введем обозначения физических величин, участвующих в математической модели движения объекта (ЛА) по высоте и в модели наблюдения за этим

движением. В соответствии с принципиальной моделью – вторым законом Ньютона, движение центра масс объекта характеризуем тремя величинами:  $y = y(t)$  – вертикальная координата (высота над Землей),  $v_y = v_y(t)$  – вертикальная скорость,  $a_y = a_y(t)$  – вертикальное ускорение.

Будем рассматривать случай движения с постоянной силой тяги двигателя ЛА. В этом случае следует считать, что ускорение ЛА тоже постоянно (пренебрегая изменением массы ЛА), однако оно неизвестно, т.е. подлежит оцениванию по показаниям приборов. В роли оценителя наряду со стандартным фильтром Калмана используем рассмотренные выше UD-реализации.

Полученные показания приборов (измеренные величины) обозначим:  $z_a = z_a(t)$  – измеренное ускорение  $a(t)$ ,  $z_h = z_h(t)$  – измеренная барометрическая высота  $h(t)$ .

Считывание этих показаний происходит в дискретные моменты времени  $t_i$  с аддитивными погрешностями  $v_1(t_i)$  и  $v_2(t_i)$  соответственно. Это означает, что  $z_a(t) = a(t) + v_1(t)$ ,  $z_h(t) = h(t) + v_2(t)$ . Погрешности датчиков между собой независимы и в отдельные моменты времени  $t_i$  являются гауссовыми случайными величинами с нулевыми средними значениями и с постоянными дисперсиями  $\sigma_1^2$ ,  $\sigma_2^2$  соответственно. Истинная высота  $y(t)$ , на которой находится ЛА, воздействует на барометрический датчик (БД) высоты так, что барометрическая высота  $h(t)$  подчиняется уравнению

$$\tau \frac{d}{dt} h(t) + h(t) = y(t).$$

Запишем уравнения состояния обобщенного динамического объекта, включающего ЛА и БД. Обозначим вектор состояния как

$$x = x(t) = [x_1 \equiv y(t) | x_2 \equiv v_y(t) | x_3 \equiv a_y(t) | x_4 \equiv h(t)]^T$$

и вектор измерения как

$$z = z(t) = [z_1 \equiv z_a(t) | z_2 \equiv z_h(t)]^T$$

с погрешностью  $v(t) = [v_1(t) | v_2(t)]^T$ . Из принятых предположений получаем непрерывную модель состояния

$$\frac{d}{dt} x(t) = Fx(t), \quad F = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ \alpha & 0 & 0 & -\alpha \end{bmatrix}, \quad \alpha = \frac{1}{\tau},$$

и дискретную модель наблюдения

$$z_k = Hx_k + v_k, \quad H = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Далее, переходя к дискретной модели состояния, запишем переходную матрицу состояния на интервале времени  $t$  :

$$\Phi(t) = \begin{bmatrix} 1 & t & t^2/2 & 0 \\ 0 & 1 & t & 0 \\ 0 & 0 & 1 & 0 \\ a_1(t) & a_2(t) & a_3(t) & a_4(t) \end{bmatrix}, \quad \begin{aligned} a_1(t) &= 1 - e^{-\alpha t}, \\ a_2(t) &= (\alpha t - 1 + e^{-\alpha t}) / \alpha, \\ a_3(t) &= (1 - \alpha t + \frac{(\alpha t)^2}{2} - e^{-\alpha t}) / \alpha^2, \\ a_4(t) &= e^{-\alpha t}. \end{aligned}$$

Зададим  $\tau_s$  – постоянный интервал выборки, т.е. темп считывания данных с датчиков; при этом считаем, что  $\gamma = \tau_s / \tau < 1$ , например,  $\gamma = 1/10$  или меньше. Подстановкой  $t = \tau_s$  в  $\Phi(t)$  определим постоянную  $\Phi = \Phi(\tau_s)$  – переходную матрицу состояния дискретной модели

$$\text{Модель ЛА + БД: } \begin{cases} x_{k+1} = \Phi x_k + G w_k, & z_k = H x_k + v_k \\ G = [0 \ 1 \ 0 \ 0]^T = \text{const} \end{cases}$$

где  $G w_k$  – «шумовая» составляющая с независимым дискретным белым шумом  $w_k \in N(0, Q = q \tau_s = \text{const})$ ,  $q$  – коэффициент диффузии соответствующего винеровского процесса,  $\text{м}^2 / \text{с}^5$  (учитываем нестационарность силы тяги двигателя ЛА).

Для заданной модели «ЛА+БД» проведем вычислительные эксперименты. Экспериментальные условия приведены в табл. 1.

Таблица 1

Экспериментальные условия для модели «ЛА+БД»

№	Параметр	Вариант исходных данных					
		1	2	3	4	5	6
1	Диффузия $q, \text{м}^2/\text{с}^5$	3000	340	400	300	3500	3350
2	Постоянная времени $\tau, \text{с}$	0,05	0,65	0,80	0,90	0,10	0,12
3	Дисперсия $\sigma_1^2, (\text{м}/\text{с}^2)^2$	1,00	2,25	4,00	6,25	6,00	5,50
4	Дисперсия $\sigma_2^2, \text{м}^2$	40	30	25	35	45	50
5	Дисперсия $P_{11}(0), \text{м}^2$	10	20	30	40	50	60
6	Дисперсия $P_{22}(0), (\text{м}/\text{с})^2$	60	50	40	30	20	10
7	Дисперсия $P_{33}(0), (\text{м}/\text{с}^2)^2$	15	20	25	30	35	40
8	Дисперсия $P_{44}(0), \text{м}^2$	45	40	35	25	30	15

Выберем количество измерений  $N$  равным 100. Сначала экспериментально оценим разницу в вычислениях по следующим алгоритмам:

- 1) SKF – стандартная реализация дискретного ФК (алгоритм 2);
- 2) TBF – дискретный фильтр Торнтон – Бирмана (алгоритмы 3 и 4);
- 3) eUD – расширенный ортогонализированный UD-фильтр (алгоритм 6).

Обозначим через  $\Delta_{A,B}(\hat{x})$  разницу в вычислениях оценки вектора состояния по алгоритмам  $A$  и  $B$ ; при этом

$$\Delta_{A,B}(\hat{x}) = \max_{1 \leq k \leq N} \|\hat{x}_k^A - \hat{x}_k^B\|_{\infty},$$

где  $\hat{x}_k^A$  и  $\hat{x}_k^B$  – оценки вектора состояния  $x_k$ , вычисленные по алгоритмам  $A$  и  $B$  в равных экспериментальных условиях. Аналогично обозначим через  $\Delta_{A,B}(P)$  разницу в вычислениях ковариационной матрицы ошибки оценивания по алгоритмам  $A$  и  $B$ ; при этом

$$\Delta_{A,B}(P) = \max_{1 \leq k \leq N} \|P_k^A - P_k^B\|_{\infty},$$

где  $P_k^A$  и  $P_k^B$  – ковариационные матрицы ошибки оценивания, вычисленные по алгоритмам  $A$  и  $B$  в равных экспериментальных условиях. Результаты вычислительных экспериментов приведены в табл. 2 и 3. Анализируя полученные данные, можно сделать вывод о том, что разница в вычислениях по алгоритмам SKF, TBF и eUD не превышает  $10^{-12}$ , т.е. в условиях рассмотренной прикладной задачи данные алгоритмы практически эквивалентны. Этот факт подтверждает работоспособность и правильность построения алгоритма eUD.

Таблица 2  
Разница в вычислениях оценки вектора состояния  $x$  для модели «ЛА+БД»

Алгоритм		$\Delta_{A,B}(\hat{x})$					
$A$	$B$	1	2	3	4	5	6
SKF	TBF	$2,13 \cdot 10^{-14}$	$6,39 \cdot 10^{-14}$	$1,99 \cdot 10^{-13}$	$5,86 \cdot 10^{-14}$	$3,55 \cdot 10^{-14}$	$7,11 \cdot 10^{-14}$
SKF	eUD	$2,13 \cdot 10^{-14}$	$3,62 \cdot 10^{-13}$	$7,18 \cdot 10^{-13}$	$2,42 \cdot 10^{-13}$	$2,27 \cdot 10^{-13}$	$1,28 \cdot 10^{-13}$
eUD	TBF	$1,73 \cdot 10^{-14}$	$3,87 \cdot 10^{-13}$	$7,39 \cdot 10^{-13}$	$2,49 \cdot 10^{-13}$	$2,20 \cdot 10^{-13}$	$1,71 \cdot 10^{-13}$

Таблица 3  
Разница в вычислениях ковариационной матрицы ошибки оценивания  $P$  для модели «ЛА+БД»

Алгоритм		$\Delta_{A,B}(P)$					
$A$	$B$	1	2	3	4	5	6
SKF	TBF	$1,02 \cdot 10^{-12}$	$7,39 \cdot 10^{-13}$	$2,27 \cdot 10^{-13}$	$3,98 \cdot 10^{-13}$	$2,27 \cdot 10^{-13}$	$2,05 \cdot 10^{-12}$
SKF	eUD	$9,09 \cdot 10^{-13}$	$7,39 \cdot 10^{-13}$	$2,27 \cdot 10^{-13}$	$3,98 \cdot 10^{-13}$	$5,68 \cdot 10^{-13}$	$2,05 \cdot 10^{-12}$
eUD	TBF	$7,96 \cdot 10^{-13}$	$2,27 \cdot 10^{-13}$	$3,41 \cdot 10^{-13}$	$2,27 \cdot 10^{-13}$	$6,82 \cdot 10^{-13}$	$1,14 \cdot 10^{-12}$

Теперь покажем, что на множестве плохо обусловленных задач рассмотренные UD-реализации дискретного фильтра Калмана остаются работоспособными и позволяют получить приемлемые результаты, несмотря на наличие ошибок машинного округления, в то время как стандартный алгоритм Калмана не справляется с поставленной задачей. Для этого рассмотрим известный пример из [6].

Рассмотрим задачу оптимальной дискретной фильтрации. Матрицы-параметры системы (1), (2) заданы следующим образом:

$$\Phi_k = I_3, Q_k = 0, G_k = 0, R_k = \delta^2 I_2, H_k = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 + \delta \end{bmatrix},$$

начальные условия  $x_0 \in N(0, I_3)$ , где  $\delta^2 < \epsilon_{\text{окр}}$ , но  $\delta > \epsilon_{\text{окр}}$ ;  $\epsilon_{\text{окр}}$  – параметр машинного округления, т.е.  $\epsilon_{\text{окр}} + 1 \neq 1$ , но  $\epsilon_{\text{окр}} / 2 + 1 = 1$ . Требуется вычислить значение матрицы ковариации ошибки оценивания  $P$  после обработки одного измерения, т.е.  $P_{||}$ . Данный тестовый пример показывает, как влияют ошибки машинного округления на работоспособность ковариационных фильтров.

Легко видеть, что ранг матрицы  $H_k$  равен 2. Можно показать, что при малых значениях  $\delta$  матрица ковариации невязки измерений  $R_{e,1}$  является плохо обусловленной, а при  $\delta \rightarrow \epsilon_{\text{окр}}$  в результате округления она имеет ранг, равный 1. В работе [21] проведен строгий теоретический анализ влияния ошибок округления на некоторые из реализаций дискретного фильтра Калмана. Авторами доказано, что число обусловленности матрицы ковариации невязки измерений  $R_{e,k}$  является определяющим фактором, влияющим на численное поведение фильтров ковариационного типа.

Чтобы проиллюстрировать на практике численную эффективность UD-реализаций дискретного фильтра, проведем следующие эксперименты. Вычислим матрицу ковариации ошибки оценивания  $P_{||}$  по алгоритмам SKF, TBF, eUD и квадратно-корневому алгоритму Поттера [3, с. 22–24; 7, с. 79]. Для каждого значения  $\delta$  будем вычислять максимальную относительную ошибку среди элементов матрицы  $P_{||}$  (обозначим  $\Delta P_1$ ), сравнивая точное решение задачи с численным, полученным с использованием ЭВМ.

Результаты экспериментов представлены в виде графиков на рис. 1. Для каждой реализации дискретного фильтра изображена зависимость  $\Delta P_1$  от параметра обусловленности  $\delta$ , где  $\delta \rightarrow \epsilon_{\text{окр}}$ .

Численные расчеты проводились на персональном компьютере, обеспечивающем представление чисел с относительной погрешностью  $10^{-16}$ . Коды программ написаны на языке MATLAB, где параметр машинного округления  $\epsilon_{\text{окр}}$  хранится в переменной `eps` со значением  $\epsilon_{\text{окр}} = \text{eps}/2 = 1,1102 \cdot 10^{-16}$ . Проведем расчеты для  $\delta \in [10^{-9} \text{eps}^{2/3}, 10^9 \text{eps}^{2/3}]$ .

Проанализируем результаты эксперимента. Легко видеть, что стандартный алгоритм Калмана (сплошная линия с маркером \*) для рассмотренной задачи при  $\delta \rightarrow \epsilon_{\text{окр}}$  быстро теряет точность вычислений. Так, например, при  $\delta = 10^{-8}$  стандартная реализация фильтра обеспечивает вычисление  $P_{||}$  с точностью  $10^{-1}$ , в то время как UD-реализации (сплошные линии с маркерами  $\times$  и  $\circ$ ) обеспечивают точность  $10^{-9}$ . Таким образом, уже при  $\delta = 10^{-8}$  и меньше алгоритм SKF не обеспечивает даже одной правильной цифры после запятой, т.е. теряет работоспособность. UD-реализациям удается поддерживать приемлемую точность вычислений вплоть до граничного значения  $\delta = \epsilon_{\text{окр}}$ .



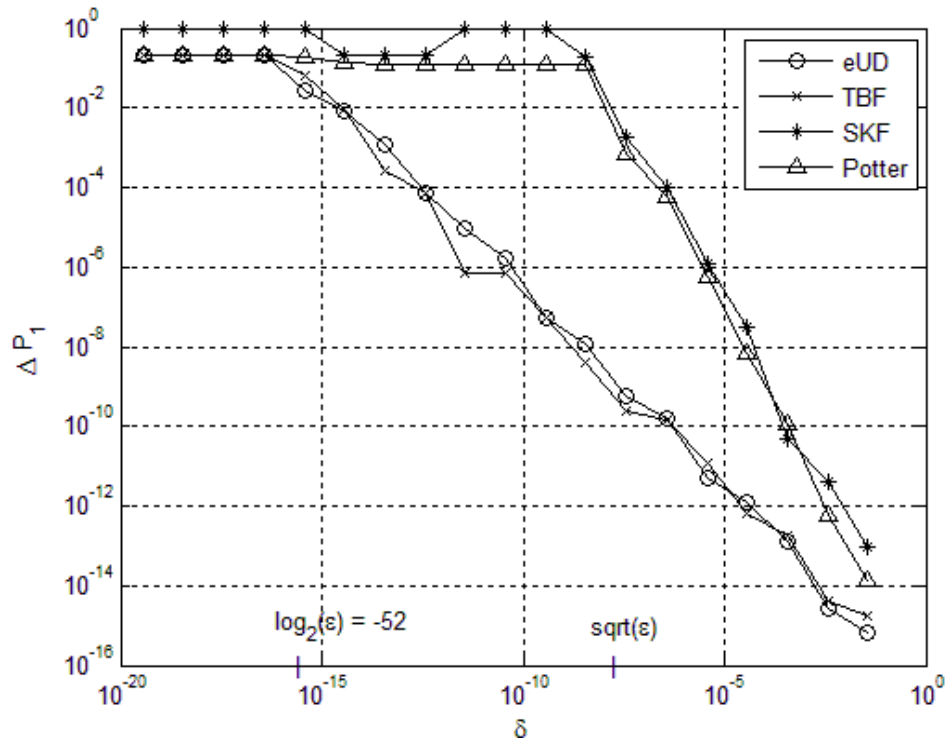


Рис. 1. Сравнение точности вычислений по четырем алгоритмам: стандартный алгоритм Калмана (SKF), UD-реализация Торнтон – Бирмана (TBF), расширенная UD-реализация (eUD), квадратно-корневой алгоритм Поттера (Potter)

Для сравнения здесь также приведен график, построенный с помощью квадратно-корневого последовательного алгоритма Поттера. Этот алгоритм выдал результаты, несколько лучшие по сравнению со стандартным алгоритмом Калмана, но в то же время существенно худшие, чем рассмотренные UD-реализации.

### Заключение

В работе изучены существующие к настоящему времени методы построения UD-фильтров. Наиболее эффективными в вычислительном плане и подходящими для реализации на современных вычислительных комплексах являются ортогонализированные формы UD-фильтра. Предложена новая форма расширенного ортогонализированного UD-фильтра (алгоритм 6), обладающая рядом преимуществ по сравнению с другими. Сравнительный анализ представленных алгоритмов проведен на примере решения одной практической задачи, а также на примере тестовой плохо обусловленной задачи.

Основная цель дальнейших исследований заключается в построении адаптивного расширенного ортогонализированного UD-фильтра для эффективного в вычислительном плане решения задачи параметрической идентификации дискретных моделей линейных стохастических систем. Подобные методы построения адаптивных ортогонализированных фильтров разработаны в [22–24].

## Приложение

*Модифицированная взвешенная ортогонализация Грама – Шмидта*

Алгоритм MWGS-ортогонализации был построен А. Бьерк (Å. Björck [25]). Доказано [26], что данный алгоритм эффективнее в вычислительном плане, чем классическая ортогонализация Грама – Шмидта, и точность вычислений по нему сравнима с точностью вычислений по известному алгоритму триангуляризации Хаусхолдера.

В алгоритме взвешенной ортогонализации рассматривается ортогональность  $m$ -векторов  $b_i$  и  $b_j$  относительно весовой матрицы  $D_w$ :

$$b_i^T D_w b_j = \begin{cases} \beta_i > 0, & \text{если } i = j, \\ 0, & \text{если } i \neq j. \end{cases}$$

Процесс MWGS-ортогонализации набора  $n$  линейно независимых  $m$ -векторов  $a_1, a_2, \dots, a_m$  относительно весовой матрицы  $D_w$  приводит к верхней треугольной  $(n \times n)$ -матрице  $U$  такой, что  $A^T = UB^T$ , т.е.

$$\begin{bmatrix} a_1^T \\ a_2^T \\ \dots \\ a_n^T \end{bmatrix} = \begin{bmatrix} 1 & u_{12} & \dots & u_{1n} \\ 0 & 1 & \dots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} \cdot \begin{bmatrix} b_1^T \\ b_2^T \\ \dots \\ b_n^T \end{bmatrix},$$

где вектора  $b_i$  есть столбцы матрицы MWGS-преобразования  $B$  и

$$B^T D_w B = \text{Diag}\{\beta_i\}_{1 \leq i \leq n} = D_\beta.$$

**Список литературы**

1. **Kalman, R. E.** A New Approach to Linear Filtering and Prediction Problems / R. E. Kalman. // Trans. of the ASME–Journal of Basic Engineering. – 1960. – Vol. 82, Series D. – P. 35–45.
2. **Potter, J. E.** Statistical Filtering of Space Navigation Measurements / J. E. Potter, R. G. Stern // Proceedings of 1963 AIAA Guidance and Control Conference. – New York : AIAA, 1963.
3. **Bierman, G. J.** Factorization methods for discrete sequential estimation / G. J. Bierman : New York : Academic Press, 1977. – 255 p.
4. **Maybeck, P. S.** Stochastic models, estimation and control, Vol. 1. / P. S. Maybeck. – New York : Academic Press, Inc., 1979. – 423 p.
5. **Kailath, T.** Linear estimation / T. Kailath, A. H. Sayed, B. Hassibi. – New Jersey : Prentice Hall, 2000. – 854 p.
6. **Grewal, M. S.** Kalman filtering: theory and practice / M. S. Grewal, A. P. Andrews. – New Jersey : Prentice-Hall, 2001. – 410 p.
7. **Огарков, М. А.** Методы статистического оценивания параметров случайных процессов / М. А. Огарков. – М. : Энергоатомиздат, 1990. – 208 с.
8. **Семущин, И. В.** Адаптивные системы фильтрации, управления и обнаружения : коллективная монография / И. В. Семущин, Ю. В. Цыганова, М. В. Куликова и др.; под. ред. проф. И. В. Семущина. – Ульяновск : Изд-во УлГУ, 2011. – 298 с.

9. **Семушин, И. В.** Вычислительные методы алгебры и оценивания : учеб. пособие / И. В. Семушин. – Ульяновск : УлГТУ, 2011. – 366 с.
10. **Bierman, G. J.** Numerical Comparison of Kalman Filter Algorithms : Orbit Determination Case Study / G. J. Bierman, C. L. Thornton // *Automatica*. – 1977. – Vol. 13. – P. 23–35.
11. **Фомин, В. Н.** Рекуррентное оценивание и адаптивная фильтрация / В. Н. Фомин. – М. : Наука, 1984. – 288 с.
12. **Льюнг, Л.** Идентификация систем. Теория для пользователя : пер. с англ. / Л. Льюнг ; под. ред. Я. З. Цыпкина. – М. : Наука, 1991. – 432 с.
13. **Саридис, Дж.** Самоорганизующиеся стохастические системы управления : пер. с англ. / Дж. Саридис ; под. ред. Я. З. Цыпкина. – М. : Наука, 1980. – 400 с.
14. **Mosca, E.** Optimal, predictive and adaptive control / E. Mosca. – New Jersey : Prentice-Hall, Inc., 1995. – 477 p.
15. **Thornton, C. L.** Triangular Covariance Factorizations for Kalman Filtering : Ph. D. Thesis / C. L. Thornton. – University of California at Los Angeles, School of Engineering, 1976.
16. **Park, P.** New square-root algorithms for Kalman filtering / P. Park, T. Kailath // *IEEE Trans. Automat. Control*. – 1995. – V. 40, № 5. – P. 895–899.
17. **Jover, J. M.** A Parallel Architecture for Kalman Filter Measurement Update and Parameter Estimation / J. M. Jover, T. Kailath // *Automatica*. – 1986. – Vol. 22, № 1. – P. 43–57.
18. **Hotop, H.-J.** New Kalman filter algorithms based on orthogonal transformations for serial and vector computers / H.-J. Hotop // *Parallel Computing*. – 1989. – № 12. – P. 233–247.
19. **Gentleman, W. M.** Least squares computations by Givens transformations without square roots / W. M. Gentleman // *J. Inst. Math. Appl.* – 1973. – № 12. – P. 329–336.
20. **Dyer, P.** Extension of square-root filtering to include process noise / P. Dyer, S. McReynolds // *J. Optim. Theory Appl.* – 1969. – № 3. – P. 444–459.
21. **Verhaegen, M.** Numerical aspects of different Kalman filter implementations / M. Verhaegen, P. Van Dooren // *IEEE Trans. Automat. Contr.* – 1986. – Vol. 31, № 10. – P. 907–917.
22. **Цыганова, Ю. В.** Вычисление градиента вспомогательного функционала качества в задаче параметрической идентификации стохастических систем / Ю. В. Цыганова // *Автоматика и телемеханика*. – 2011. – № 9. – С. 142–160.
23. **Цыганова, Ю. В.** Об эффективных методах параметрической идентификации линейных дискретных стохастических систем / Ю. В. Цыганова, М. В. Куликова // *Автоматика и телемеханика*. – 2012. – № 6. – С. 34–51.
24. **Tsyganova, J. V.** State sensitivity evaluation within UD based array covariance filters / J. V. Tsyganova, M. V. Kulikova // *IEEE Transactions on Automatic Control*. – 2013. – Vol. 58, № 11. – P. 2938–2944.
25. **Björck, Å.** Solving least squares problems by orthogonalization / Å. Björck // *BIT*. – 1967. – Vol. 7. – P. 1–21.
26. **Jordan, T. L.** Experiments on error growth associated with some linear least-squares procedures / T. L. Jordan // *Math. Comp.* – 1968. – Vol. 22. – P. 579–588.

### **References**

1. Kalman R. E. *Trans. of the ASME–Journal of Basic Engineering*. 1960, vol. 82, Series D, pp. 35–45.
2. Potter J. E., Stern R. G. *Proceedings of 1963 AIAA Guidance and Control Conference*. New York: AIAA, 1963.
3. Bierman G. J. *Factorization methods for discrete sequential estimation*. New York: Academic Press, 1977, 255 p.

4. Maybeck P. S. *Stochastic models, estimation and control, Vol. 1*. New York: Academic Press, Inc., 1979, 423 p.
5. Kailath T., Sayed A. H., Hassibi B. *Linear estimation*. New Jersey: Prentice Hall, 2000, 854 p.
6. Grewal M. S., Andrews A. P. *Kalman filtering: theory and practice*. New Jersey: Prentice-Hall, 2001, 410 p.
7. Ogarkov M. A. *Metody statisticheskogo otsenivaniya parametrov sluchaynykh protsessov* [Methods of statistical estimation of stochastic process parameters]. Moscow: Energoatomizdat, 1990, 208 p.
8. Semushin I. V., Tsyganova Yu. V., Kulikova M. V. et al. *Adaptivnye sistemy fil'tratsii, upravleniya i obnaruzheniya: kollektivnaya monografiya* [Adaptive systems of filtration, control and detection: collaborative monograph]. Ulyanovsk: Izd-vo UIGU, 2011, 298 p.
9. Semushin I. V. *Vychislitel'nye metody algebrы i otsenivaniya: uchebnoe posobie* [Computational methods of algebra and estimation: tutorial]. Ulyanovsk: UIGTU, 2011, 366 p.
10. Bierman G. J., Thornton C. L. *Automatica*. 1977, vol. 13, pp. 23–35.
11. Fomin V. N. *Rekurrentnoe otsenivanie i adaptivnaya fil'tratsiya* [Recurrent estimation and adaptive filtration]. Moscow: Nauka, 1984, 288 p.
12. L'yunг L. *Identifikatsiya sistem. Teoriya dlya pol'zovatelya: per. s angl.* [System identification. Theory for users: translation from English]. Moscow: Nauka, 1991, 432 p.
13. Saridis Dzh. *Samoorganizuyushchiesya stokhasticheskie sistemy upravleniya: per. s angl.* [Self-organizing stochastic systems of control: translation from English]. Moscow: Nauka, 1980, 400 p.
14. Mosca E. *Optimal, predictive and adaptive control*. New Jersey: Prentice-Hall, Inc., 1995, 477 p.
15. Thornton C. L. *Triangular Covariance Factorizations for Kalman Filtering: Ph. D. Thesis*. University of California at Los Angeles, School of Engineering, 1976.
16. Park P., Kailath T. *IEEE Trans. Automat. Control*. 1995, vol. 40, no. 5, pp. 895–899.
17. Jover J. M., Kailath T. *Automatica*. 1986, vol. 22, no. 1, pp. 43–57.
18. Hotop H.-J. *Parallel Computing*. 1989, no. 12, pp. 233–247.
19. Gentleman W. M. *J. Inst. Math. Appl.* 1973, no. 12, pp. 329–336.
20. Dyer P., McReynolds S. J. *Optim. Theory Appl.* 1969, no. 3, pp. 444–459.
21. Verhaegen M., P. Van Dooren *IEEE Trans. Automat. Contr.* 1986, vol. 31, no. 10, pp. 907–917.
22. Tsyganova Yu. V. *Avtomatika i telemekhanika* [Automatic and telemechanics]. 2011, no. 9, pp. 142–160.
23. Tsyganova Yu. V., Kulikova M. V. *Avtomatika i telemekhanika* [Automatic and telemechanics]. 2012, no. 6, pp. 34–51.
24. Tsyganova J. V., Kulikova M. V. *IEEE Transactions on Automatic Control*. 2013, vol. 58, no. 11, pp. 2938–2944.
25. Björck Å. *BIT*. 1967, vol. 7, pp. 1–21.
26. Jordan T. L. *Math. Comp.* 1968, vol. 22, pp. 579–588.

**Цыганова Юлия Владимировна**

кандидат физико-математических наук,  
доцент, кафедра информационных  
технологий, Ульяновский  
государственный университет  
(Россия, Ульяновск, ул. Льва Толстого, 42)  
E-mail: jvt.ulsu@gmail.com

**Tsyganova Yuliya Vladimirovna**

Candidate of physical and mathematical  
sciences, associate professor,  
sub-department of information  
technologies, Ulyanovsk State University  
(42 Lva Tolstogo street, Ulyanovsk, Russia)

УДК [519.254+519.654]:629.5.05

**Цыганова, Ю. В.**

**О методах реализации UD-фильтра** / Ю. В. Цыганова // Известия высших учебных заведений. Поволжский регион. Физико-математические науки. – 2013. – № 3 (27). – С. 84–104.