УДК 519.711:681.5

И. В. Семушин, Ю.В. Цыганова, Н.Д. Старостина

АЛГОРИТМЫ РЕШЕНИЯ ОБРАТНОГО УРАВНЕНИЯ РИККАТИ ДЛЯ ДИСКРЕТНЫХ ЗАДАЧ УПРАВЛЕНИЯ

Семушин Иннокентий Васильевич, *доктор технических наук, профессор кафедры «Информационные технологии» Ульяновского государственного университета (УлГУ). Имеет монографии, статьи, учебные пособия и патенты на изобретения. Область научных интересов: фильтрация и управление в условиях неопределенности. [e-mail: kentvsem@yandex.ru].*

Цыганова Юлия Владимировна, *кандидат физико-математических наук, доцент кафедры «Информационные технологии» УлГУ. Имеет монографию, статьи, учебные и учебно-методические пособия. Область научных интересов: параметрическая идентификация, адаптивная фильтрация и численно эффективные алгоритмы для стохастических систем. [e-mail: tsyganovajv@mail.ru].*

Старостина Наталья Дмитриевна, *аспирант кафедры «Информационные технологии» УлГУ, окончила факультет математики и информационных технологий УлГУ по специальности «Прикладная математика». Область научных интересов: вычислительные методы оценивания и управления. [e-mail: kapelika88@mail. ru].*

Аннотация

Рассматриваются численно устойчивые алгоритмы линейно-квадратичного регулятора (LQR) в структуре управления со скользящей глубиной прогнозирования потерь на основе скаляризованных квадратно-корневых реализаций.

Ключевые слова: LQG-управление, скаляризованные квадратно-корневые алгоритмы, скользящая глубина прогнозирования потерь, ортогонализация.

Innokentiy Vasilyevich Semushin, *Doctor of Science in Engineering, Professor of Information Technology Department at Ulyanovsk State University (UlSU); author of papers, monographs and textbooks; holds patents for inventions; is interested in filtering and control under uncertainty. e-mail: kentvsem @yandex. ru.*

Yuliya Vladimirovna Tsyganova, *Doctor of Philosophy in Physics and Mathematics, Associate Professor of Information Technology Department at UlSU; author and co-author of monograph, papers, and textbooks; is interested in parameter identification, adaptive filtering, and numerically efficient algorithms for stochastic systems. e-mail: tsyganovajv@mail.ru.*

Natalia Dmitrievna Starostina, *post-graduate student of Information Technology Department at UlSU; graduated from Mathematics and Information Technology Faculty of UlSU with the specialty in «Applied Mathematics»; is interested in computational methods of estimation and control. e-mail: kapelika88@mail.ru.*

Abstract

Robust algorithms for linear-quadratic regulator (LQR) within the structure of receding horizon control are considered based on scalarized square-root implementations.

Key words: LQG-control, scalarized square-root algorithms, receding horizon control, orthogonalization.

Введение

В основе решения LQG-задачи управления, т.е. задачи с *линейными* моделями систем, с оптимизацией по *квадратическому* критерию и с *гауссовыми* возмущениями, лежит решение двух матричных нелинейных рекуррентных уравнений Риккати (Recurrent Riccati Equation, RRE), двойственных друг другу [1]. Прямое RRE включено в синтез оптимального LQG-оценивателя (Kalman Filter, KF), а обратное RRE — в решение задачи LQR (Linear-Quadratic Regulator). KF и LQR синтезируются независимо друг от друга (теорема разделения), и последний включается каскадно с первым, замыкая таким образом обратную связь (компенсатор). Регулятор в законе LQG-управления идентичен регулятору в законе детерминистского LQ (Linear-Quadratic) управления (принцип эквивалентности).

Любая литература по теории LQG-управления содержит эти хорошо известные факты, но не в каждой представлены вычислительные аспекты матричных уравнений Риккати. В этом вопросе обычно ограничиваются ссылками на функции care или dare системы MATLAB [1] (например, [2, 3]), имея таким образом в виду *алгебраические уравнения Риккати* (Algebraic Riccati Equations, ARE) [4]: дифференциальное (care, continuous ARE) либо разностное (dare, discrete ARE). Решение ARE актуально для синтеза *стабилизирующей* обратной связи системы. Для систем дискретного времени решение DARE является установившимся (предельным) решением RRE при стремлении к бесконечности глубины прогнозирования потерь (Infinite Horizon Control, IHC) [5]. Обобщенная теория Риккати применяется в задачах робастного управления [6].

В учебной литературе лишь иногда присутствуют не только запись или вывод RRE, но также некоторый обзор методов численного решения ARE. Отметим в этой связи [7], где говорится (с. 389), что (по данным на 1986 год) «количество опубликованных работ, посвященных решению и свойствам уравнения Риккати, может, по-видимому, составить целую книгу», упоминается итеративный метод и несколько подробнее демонстрируются рекуррентный (прямой) метод решения и метод собственных значений и собственных векторов.

Однако основной источник сведений по уравнениям Риккати — обширная научная литература [8]. Большое внимание, уделяемое ARE, вызвано тем, что метод прямых итераций RRE оказывается неприемлем, прежде всего, из-за низкой скорости сходимости к установившемуся режиму итераций.

Число оригинальных российских разработок в области ARE сравнительно невелико [9]. Вычислительные методы решения ARE разрабатываются за рубежом широким фронтом в течение многих лет. Среди множества зарубежных публикаций отметим лишь некоторые, например, [10, 11, 12] — для решения задач LQR и [13, 14, 15] — для задач LQG-оценивания. На базе этих методов построены решатели уравнений Риккати в математических пакетах *Maple* [16], *Mathematica* [17], MATLAB [18, 19, 20], в компьютерных библиотеках BLAS (level I-III), EISPACK и LINPACK, а также в их преемнике LAPACK [21, 22] на языке FORTRAN, но есть и на языке Python [23]. Число публикаций по решателям ARE продолжает расти [24, 25, 26].

Использование существующих методов, пакетов и библиотек рассчитано, в основном, на режим «offline». Для систем реального времени (в режиме «online») многие из этих эффективных методов оказываются слишком времязатратными. Кроме того, в ряде случаев решение DARE не требуется. К этой категории относятся системы управления со скользящей глубиной прогнозирования потерь (с «уходящим горизонтом управления» — Receding Horizon Control, RHC), предиктивное управление и адаптивное управление, оба основанные на RHC [1, 27].

Подход, принятый в данной работе, рассчитан именно на системы с RHC. Он заключается в прямом переносе вычислительных методов оценивания [13, 14, 15], недавний обзор которых содержится в [28], на задачи управления с RHC, где требуется решать не ARE, а RRE. Двойственные аналоги этих методов для задач управления могут составить продуктивный подход и заслуживают детального исследования, дополняя таким образом имеющиеся зарубежные работы этого плана [22].

Цель данной статьи — подвести данный подход к стадии детального сравнительного анализа и практической реализации численно устойчивых версий прямых итераций Риккати для LQR.

1 Задача LQG-управления

Исходная модель системы принадлежит классу линейных стохастических систем и включает: n-мерное стохастическое рекуррентное уравнение состояния

$$x(t_{i+1}) = \Phi(t_{i+1}, t_i)\, x(t_i) + B(t_i)\, u(t_i) + w(t_i),$$
$$i = 0, 1, \ldots;\; x(t_0) \sim \mathrm{N}(\bar{x}_0, P_0) \tag{1}$$

и m-мерное уравнение измерений

$$z(t_i) = H(t_i)\, x(t_i) + v(t_i), \quad i = 1, 2, \ldots, \tag{2}$$

в которых $\{w(t_0),\ w(t_1),\ \ldots\}$ и $\{v(t_1),\ v(t_2),\ \ldots\}$ представляют собой две независимые последовательности независимых нормально распределенных случайных векторов возмущений w и погрешностей v, каждая с нулевым средним значением, имеющие, соответственно, размерности q и m, обладающие в каждый момент времени t_i ковариациями $Q(t_i) \geq 0$ и $R(t_i) > 0$, независимые от случайного нормально распределенного начального состояния $x(t_0)$ со средним значением $\bar{x}_0$ и ковариацией P_0.

Управляющее воздействие $u_{[0\,N]} \triangleq \{u(t_i)\}_{i=0}^{N}$ предполагается отыскивать как ряд $\{u(t_0),\ u(t_1), \ldots,\ u(t_N)\}$ r-мерных векторов $u(t)$, вводимых в объект (1) с целью минимизировать среднеквадратический функционал качества (ожидаемую величину потерь с глубиной прогнозирования N),

$$J(x_0, P_0, u_{[0,N]}) = \mathbf{E}\left\{ \sum_{i=0}^{\mathbf{N}} \left(\left\| x(t_i) \right\|^2_{\Psi(t_i)} + \left\| u(t_i) \right\|^2_{\Sigma(t_i)} \right) + \left\| x(t_{N+1}) \right\|^2_{\Psi_f} \right\}. \tag{3}$$

---

1 MATLAB является зарегистрированной торговой маркой компании The MathWorks, Inc., MA.

Симметрические матрицы $\psi(t_i) \geq 0$, $\Sigma(t_i) > 0$, а также $\psi_f \geq 0$ (f = final) задают веса (удельные значимости) потерь из-за отклонений от нуля состояний и управлений на интервале прогнозирования потерь $[t_0, t_N]$, а также финального состояния $x(t_{N+1})$.

Таким образом, задача заключается в определении оптимального физически осуществимого закона управления $u^* = u^*_{[0\,N]}$, минимизирующего квадратический функционал (3).

2 Оптимальный компенсатор

Теорема 1 [1]. Оптимальный закон LQG-управления для задачи с критерием (3) разделяется на две части (часть I и часть II), соединенные последовательно (II вслед за I) и синтезируемые независимо друг от друга:

I. Оптимальный фильтр Калмана (KF).

А. Для $i = 0, 1, \ldots, N$ KF вычисляет *экстраполяционные оценки* $\hat{x}(t_{i+1}^-)$ состояния $x(t_{i+1})$, получаемые при экстраполяции отфильтрованных оценок $\hat{x}(t_i^+)$ от момента t_i к моменту t_{i+1}, в виде

$$\hat{x}(t_{i+1}^-) = \Phi(t_{i+1}, t_i)\hat{x}(t_i^+) + B(t_i)u^*(t_i) \tag{4}$$

с $\hat{x}(t_0^+) := \bar{x}_0 \triangleq \mathbf{E}\{x(t_0)\}$,

и также их *ковариации*

$$P(t_{i+1}^-) = Q(t_i) + \Phi(t_{i+1}, t_i)P(t_i^+)\Phi^T(t_{i+1}, t_i)$$

с $P(t_0^+) := P_0 \triangleq \mathbf{E}\left\{[x(t_0) - \bar{x}_0][x(t_0) - \bar{x}_0]^T\right\}$.

Б. Для $i = 1, 2, \ldots, N$ KF вычисляет *отфильтрованные оценки* $\hat{x}(t_i^+)$, обновляемые по измерениям $z_i \triangleq z(t_i)$ с ковариациями $R(t_i) > 0$ ошибок измерений в моменты t_i, в виде

$$\hat{x}(t_i^+) = \hat{x}(t_i^-) + K_f(t_i)\left[z_i - H(t_i)\hat{x}(t_i^-)\right] \tag{5}$$

с коэффициентом усиления фильтра (f ≡ filter)

$$K_f(t_i) = P(t_i^-)H^T(t_i)\left[R(t_i) + H(t_i)P(t_i)H^T(t_i)\right]^{-1}$$

и также *ковариации отфильтрованных оценок*

$$P(t_i^+) = P(t_i^-) - K_f(t_i)H(t_i)P(t_i^-).$$

II. Оптимальный линейный регулятор (LQR).

Минимальные ожидаемые потери обеспечивает оптимальный линейный регулятор (r = *r*egulator)

$$u^*(t_i) = -G_r(t_i)\hat{x}(t_i^+),\ i = 0, 1, \ldots, N. \tag{6}$$

Управляющая функция *стохастического* LQR

$$u^*[t_i, (.)] = -G_r(t_i)(.) \tag{7}$$

идентична управляющей функции детерминистского LQR, причем для матрицы $G_r(t_i)$ в (7) справедлив следующий алгоритм:

$$\left.\begin{array}{ll}
(a) & \Pi(t_{N+1}) = \psi_f,\\
(b) & \bar{A}(t_i) = \Sigma(t_i) + B^T(t_i)\Pi(t_{i+1})B(t_i),\\
(c) & \begin{cases}\Lambda(t_i) = \Phi^T(t_{i+1}, t_i)\Pi(t_{i+1})B(t_i)\times\\ \quad\times \bar{A}^{-1}(t_i)B^T(t_i)\Pi(t_{i+1})\Phi(t_{i+1}, t_i),\end{cases}\\
(d) & \begin{cases}M(t_i) = \Phi^T(t_{i+1}, t_i)\Pi(t_{i+1})\times\\ \qquad\qquad \times\Phi(t_{i+1}, t_i) - \Lambda(t_i),\end{cases}\\
(e) & K_r(t_i) = \bar{A}^{-1}(t_i)B^T(t_i)\Pi(t_{i+1}),\\
(f) & G_r(t_i) = K_r(t_i)\Phi(t_{i+1}, t_i),\\
(g) & \Pi(t_i) = \psi(t_i) + M(t_i).
\end{array}\right\} \tag{8}$$

В алгоритме (8) пп. (8b)–(8g) действуют циклически для $i = N, N-1, \ldots, 1, 0$, хотя $\Pi(t_0)$, найденное в п. (8g) при $i = 0$, далее не используется (конец вычислений).

3 Формальное RRE: неустойчивость

Рассмотрим одну итерацию RRE, записанную в формальных (отвлеченных) обозначениях с произвольной матрицей G согласованного размера:

$$\left.\begin{array}{l}\tilde{X}_- = V +\\ \quad + A^T\left[\tilde{X} - \tilde{X}G\left(C + G^T\tilde{X}G\right)^{-1}G^T\tilde{X}\right]A,\end{array}\right\} \tag{9}$$

где $C \in \mathbb{R}^{s\times s}$, $C = C^T > 0$, $\tilde{X} > 0$ и $V \geq 0$.

Итерации вида (9) повторяются как для KF, так и для LQR с операцией $\tilde{X} := \tilde{X}_-$ между повторениями. Случай KF опускаем из рассмотрения как известный [28]. Для случая LQR введем следующие соответствия между формальными и фактическими значениями матриц-параметров:

$$\left.\begin{array}{l}\tilde{X}_- \equiv \Pi(t_i), V \equiv \Psi(t_i), A \equiv \Phi(t_{i+1}, t_i),\\ \tilde{X} \equiv \Pi(t_{i+1}), G \equiv B(t_i), C \equiv \Sigma(t_i),\\ K_r \equiv K_r(t_i), G_r \equiv G_r(t_i), V_f \equiv \Psi_f.\end{array}\right\} \tag{10}$$

Подстановка (10) в (9) дает

$$\left.\begin{array}{l}\Pi(t_i) = \psi(t_i) + \Phi^T(t_{i+1}, t_i)\{\Pi(t_{i+1}) -\\ -\Pi(t_{i+1})B(t_i)\left[\Sigma(t_i) + B^T(t_i)\Pi(t_{i+1})\times\right.\\ \left.\times B(t_i)\right]^{-1}B^T(t_i)\Pi(t_{i+1})\}\Phi(t_{i+1}, t_i),\\ i = N, N-1, \ldots, 1, 0,\end{array}\right\} \tag{11}$$

т. е. *обратное* RRE, которому удовлетворяет матрица $\Pi(t_i)$ в алгоритме (8) с терминальным условием $\Pi(t_{N+1}) = \psi_f$ в момент $i = N$ начала счета в обращенном времени. Учитывая формальные обозначения K_r и G_r в (10), для (9) имеем эквивалентное представление:

$$K_r = \left(C + G^T\tilde{X}G\right)^{-1}G^T\tilde{X}, \tag{12}$$

$$\tilde{X}_- = V + A^T\left[\tilde{X} - \tilde{X}GK_r\right]A, \tag{13}$$

$$G_r = K_rA. \tag{14}$$

Оформим вычисления по алгоритму (12), (13), (14) в виде процедуры следующим образом:

$$\boxed{\mathrm{Ric}\ \big(\underbrace{C,G,\tilde{X},V,A}_{in} \mid \underbrace{\tilde{X}_{-},G_r}_{out}\big)}$$

Сделав для $i = N$ предварительную установку терминального значения $\tilde{X} := V_f$ и затем запуская эту процедуру в цикле $for\ i = N\ downto\ 0\ do$ так, чтобы входные параметры (in) были взяты в согласии с обозначениями (10), будем иметь выходные параметры (out) также в согласии с (10).

Замечание 1. В действительности последнее утверждение верно лишь теоретически, т.е. при отсутствии инструментальных погрешностей в вычислениях (в компьютерной арифметике вещественных чисел). Формула (13) содержит в себе реальную опасность потери положительной определенности матрицы $\tilde{X}_{-}$ на стадии вычитания в квадратных скобках. Именно это является причиной численной неустойчивости этой процедуры Ric.

4 Двухстадийная форма RRE

Представим процедуру Ric (12), (13), (14) в двухстадийной форме, выделяя вычисления:

$$\hat{X} = \tilde{X} - \tilde{X}GK_r, \tag{15}$$

$$\tilde{X}_{-} = V + A^T\hat{X}A. \tag{16}$$

Стадия I: $\boxed{\mathrm{Riciup}\big(C,G,\tilde{X} \mid \hat{X},K_r\big)}$

Riccati instant update: (12) ⇒(15).

Стадия II: $\boxed{\mathrm{Rictup}\big(V,A,\hat{X},K_r \mid \tilde{X}_{-},G_r\big)}$

Riccati temporal update: (16) ⇒(14).

Лемма 1. Для любых положительно определенных матриц $\tilde{X}$ и C алгоритм (12) ⇒(15) стадии I эквивалентен следующему алгоритму (17)

$$\hat{Z} = \tilde{Z} + GC^{-1}G^T \tag{17}$$

в том смысле, что $\hat{Z}^{-1} = \hat{X}$ с любой матрицей $G \in \mathbb{R}^{n\times s}$, когда $\tilde{Z}^{-1} = \tilde{X}$.

Доказательство. Известно ([13] с. 26–27). □

5 Скаляризация процедуры Riciup

В общем случае C – недиагональная матрица. Найдем разложение Холесского $C = LDL^T$ (без операции квадратного корня) с нижней треугольной матрицей L, имеющей единичную диагональ, и $D = diag\ [d_1, d_2, \ldots, d_s]$, $\forall\ d_k > 0$. Обозначим $Y \triangleq GL^{-T}$, т. е. Y^T – решение нижнетреугольной системы $LY^T = G^T$. Вместо алгоритма (12) ⇒ (15) для стадии I получаем его эквивалент

$$\hat{X} = \tilde{X} - \tilde{X}Y\big(D + Y^T\tilde{X}Y\big)^{-1}Y^T\tilde{X}. \tag{18}$$

Запишем $Y = [y_1 | y_2 | \ldots | y_s]$, где y_k — k-й столбец, и будем вводить матрицу Y постолбцово в следующих алгоритмах 1, 2.

Алгоритм 1 (скаляризованный, прямой).

А. Начальное присваивание. $X_0 := \tilde{X}$.

Б. Скаляризованный ввод. Для $k = 1, 2, \ldots, s$ выполнять:

$$\left.\begin{aligned} \bar{\alpha}_k &= d_k + y_k^T X_{k-1} y_k, \\ X_k &= X_{k-1} - X_{k-1} y_k \bar{\alpha}_k^{-1} y_k^T X_{k-1}. \end{aligned}\right\} \tag{19}$$

В. Завершающее присваивание. $\hat{X} := X_s$

Алгоритм 2 (скаляризованный, инверсный).

А. Начальное присваивание. $Z_0 := \tilde{Z} \triangleq \tilde{X}^{-1}$.

Б. Скаляризованный ввод. Для $k = 1, 2, \ldots, s$ выполнять

$$Z_k = Z_{k-1} + y_k d_k^{-1} y_k^T. \tag{20}$$

В. Завершающее присваивание. $\hat{Z} := Z_s$.

Лемма 2. Алгоритмы 1 и 2 эквивалентны.

Доказательство. Все X_j и Z_j в (19), (20) взаимно инверсны в силу леммы 1, и $\hat{Z} = \hat{X}^{-1}$. □

Теорема 2 *(верификация алгоритма 1 для уравнения (18))*. Алгоритм 1 верен, т. е. может быть применен вместо (18).

Доказательство. По лемме 1 равенство (18) эквивалентно равенству (17) при смене обозначений: $G \to Y,\ C \to D$. При этом (17) примет вид:

$$\hat{Z} = \tilde{Z} + [y_1 | \ldots | y_s] \times \\ \times \begin{bmatrix} d_1^{-1} & & 0 \\ & \ddots & \\ 0 & & d_s^{-1} \end{bmatrix} \begin{bmatrix} y_1^T \\ \vdots \\ y_s^T \end{bmatrix} = \tilde{Z} + \sum_{k=1}^{s} y_k d_k^{-1} y_k^T.$$

Алгоритм 2 дает это же значение $\hat{Z}$. Он эквивалентен алгоритму 1 (в силу леммы 2). Отсюда, алгоритм 1 дает результат $\hat{X}$ в алгоритме (12) ⇒ (15) (Riciup) и в (18). □

Введем скаляризованную процедуру Ricsiup:

Стадия I: $\boxed{\mathrm{Ricsiup}\big(C,G,\tilde{X},s \mid \hat{X},K_r\big)}$

Riccati scalarized instant update:

begin	
$C = LDL^T$	⊙ найти L и D
$LY^T = G^T$	⊙ найти Y
$k := 1$	⊙ начать
while $k \le s$ do	⊙ цикл
begin	⊙ продолжить
$p^T := y_k^T \tilde{X}$	⊙ строка
$\bar{\alpha} := d_k + p^T y_k$	⊙ скаляр
$\bar{K}_k := p^T / \bar{\alpha}$	⊙ строка
$\tilde{X} := \tilde{X} - \tilde{X} y_k \bar{K}_k$	⊙ матрица $\tilde{X}$
$k := k + 1$	⊙ инкремент
end	⊙ закончить
$\hat{X} := \tilde{X}$	⊙ вывести $\hat{X}$
$\bar{K}^T := \big[\bar{K}_1^T \mid \ldots \mid \bar{K}_s^T\big]$	⊙ собрать $\bar{K}^T$
$L^T K_r = \bar{K}$	⊙ найти K_r
end	

Теорема 3. Алгоритм Ricsiup эквивалентен алгоритму Riciup.

Доказательство. В цикле while реализован доказанный Алгоритм 1 формирования $\hat{X}$. Для завершения доказательства достаточно подставить $C = LDL^T$ в (12) и убедиться, что $K_r = L^{-T}\bar{K}$, где $\bar{K} = (D + Y^T\tilde{X}Y)^{-1}Y^T\tilde{X}$. □

Замечание 2. В переходе от Riciup к Ricsiup устранена операция обращения матрицы в формуле (12). Однако источник численной неустойчивости (вычисление матрицы $\tilde{X}$ в цикле while) требует модификации.

6 Квадратно-корневая модификация

Применим разложение Холесского к симметрическим матрицам $\hat{X}$, $\tilde{X}$, $\tilde{X}_-$, V и V_f, для определенности – нижнетреугольное [28]:

$$\left.\begin{aligned} &\hat{X} \equiv \hat{S}\hat{S}^T, \tilde{X} \equiv \tilde{S}\tilde{S}^T, \tilde{X}_- \equiv \tilde{S}_-\tilde{S}_-^T, \\ &V \equiv S_V S_V^T, V_f \equiv S_f S_f^T. \end{aligned}\right\} \quad (21)$$

Модифицированная процедура srRicsiup оперирует с квадратными корнями из (21) [29]:

Стадия I : $\boxed{\mathrm{srRicsiup}\left(C, G, \tilde{S}, s \mid \hat{S}, K_r\right)}$

square-root Riccati scalarized instant update:

begin

$C = LDL^T$ ⊙ найти L и D

$LY^T = G^T$ ⊙ найти Y

$k := 1$ ⊙ начать

while $k \le s$ do ⊙ цикл

begin ⊙ продолжить

$f := \tilde{S}^T y_k$ ⊙ столбец

$\bar{\alpha} := d_k + f^T f$ ⊙ скаляр

$\gamma := \left(1 + \sqrt{d_k / \bar{\alpha}}\right)^{-1}$ ⊙ скаляр

$\bar{K}_k := f^T \tilde{S}^T / \bar{\alpha}$ ⊙ строка

$\tilde{S} := \tilde{S} - \gamma \bar{K}_k^T f^T$ ⊙ матрица $\tilde{S}$

$k := k + 1$ ⊙ инкремент

end ⊙ закончить

$\hat{S} := \tilde{S}$ ⊙ вывести $\hat{S}$

$\bar{K}^T := \left[\bar{K}_1^T \mid \ldots \mid \bar{K}_s^T\right]$ ⊙ собрать $\bar{K}^T$

$L^T K_r = \bar{K}$ ⊙ найти K_r

end

Соответственно, заменим процедуру Rictup на srRictup с ортогональными преобразованиями:

Стадия II : $\boxed{\mathrm{srRictup}\left(S_V, A, \hat{S}, K_r \mid \tilde{S}_-, G_r\right)}$

square-root Riccati temporal update orthogonalized

$$\begin{bmatrix} \tilde{S}_-^T \\ 0 \end{bmatrix} = T \begin{bmatrix} \hat{S}^T A \\ S_V^T \end{bmatrix}, \quad G_r = -K_r A, \quad (22)$$

где T есть одно из ортогональных преобразований (Хаусхолдера, или Гивенса, или Грама-Шмидта), приводящее матрицу в правой части равенства (22) к верхнему треугольному виду [30].

Теорема 4. Алгоритм srRicsiup эквивалентен алгоритму Ricsiup, и алгоритм srRictup эквивалентен алгоритму Rictup.

Доказательство. Выбирая из (21) нужную подстановку для матрицы $\tilde{X}$ в алгоритме Ricsiup и факторизуя разность $\tilde{X} - \tilde{X}y_k\bar{K}_k$, получаем, что

$$\forall f : I - ff^T/\bar{\alpha} = \left(I - \beta ff^T\right)^2.$$

Это влечет квадратное уравнение

$$\beta^2 - 2\left(\bar{\alpha} - d_k\right)^{-1}\beta + \bar{\alpha}^{-1}\left(\bar{\alpha} - d_k\right)^{-1} = 0,$$

из двух решений которого выбираем

$$\beta = \bar{\alpha}^{-1}\left(1 + \sqrt{d_k / \bar{\alpha}}\right)^{-1}$$

как более устойчивое численно и вводим обозначение $\gamma \triangleq \bar{\alpha}\beta$. Справедливость первого равенства (22) проверяется умножением результата его транспонирования на само себя: результат совпадает с (16). □

Заключение

В статье представлены главные формулировки для задач LQG-оценивания и LQ-регулирования. Отталкиваясь от ранее решенных вопросов численно устойчивого оценивания, совершен переход к построению численно устойчивых алгоритмов оптимального линейного регулятора с постоянным и скользящим интервалом прогнозирования потерь (Receding Horizon Control). В качестве примера приведен аналог алгоритма Поттера, который совершает прямые численно устойчивые итерации обратного дискретного уравнения Риккати.

Список литературы

1. Mosca E. Optimal, Predictive, and Adaptive Control. — NJ: Prentice Hall, 1995.

2. Душин С.Е., Зотов Н.С., Имаев Д.Х., Кузьмин Н.Н., Яковлев В.Б. Теория автоматического управления. Под ред. В.Б. Яковлева. — 3-е изд. — М.: Высшая школа, 2009.

3. Александров А.Г. Оптимальные и адаптивные системы. — М.: Высшая школа, 1989 (2003).

4. Lancaster P., Rodman L. Algebraic Riccati Equations. — N-Y: Oxford University Press Inc., 1995.

5. Willems J.C. Least Squares Stationary Optimal Control and the Algebraic Riccati Equation // IEEE Trans. Automat. Contr. — 1971. — Vol. AC- 16. — Pp. 621-634.

6. Ionescu V., Oara C., Weiss M.D. Generalized Riccati Theory and Robust Control. A Popov Function Approach. — N-Y: John Wiley & Sons, Inc., 1999.

7. Куо Б. Теория и проектирование цифровых систем управления. Пер. с англ. Под ред. П.И. Попова. — М.: Машиностроение, 1986.

8. Riccati solvers at ScienceDirect: 485 articles found // URL: http://www.sciencedirect.com/ Visited: April 14, 2012.

9. Aliyev F.A., B. A. Bordyug, Larin B.V. Orthogonal Projections and Solution of Algebraic Riccati Equations // USSR Computational Mathematics and Mathematical Physics. — 1989. Vol. 29. — Iss. 3. — Pp. 104-108.

10. Laub A. J. A Schur Method for Solving Algebraic Riccati Equations // IEEE Trans. Automat.Contr.— 1979. — Vol. 24, — Pp. 913-921.

11. Pappas T., Laub A. J., Sandell Jr. N. R. On the Numerical Solution of the Discrete-time Algebraic Riccati Equation // IEEE Trans. Automat. Contr. — 1980. — Vol. AC-25. — Pp. 631-641.

12. Arnold W. F., Laub A.J. Generalized Eigenproblem Algorithms and Software for Algebraic Riccati Equations // Proceedings of the IEEE. — 1984. — Vol. 72, Iss. 12. — Pp. 1746-1754.

13. Bierman G.J. Factorization Methods for Discrete Sequential Estimation. — N-Y, San Francisco, London: Academic Press, 1977.

14. Kailath T., Sayed A. H., Hassibi B. Linear Estimation. — NJ: Prentice Hall, 1999.

15. Kaminski P. G., Bryson A. E., Schmidt S. F. Discrete Square Root Filtering: A Survey of Current Techniques // IEEE Trans. on Automat. Contr. — 1971. — Vol. AC-16. — Pp. 727-736. [Русский перевод: Каминский П.Г., Брайсон А.Е., Шмидт С.Ф. Обзор современных методов дискретной фильтрации, использующих квадратные корни матриц // Зарубежная радиоэлектроника. — 1973. — №6. — С. 37–53.]

16. New Features in Maple 15: Algebraic Riccati Equation Solvers // URL: http://www.maplesoft. com/products/maple/new_features/algebraic_ricc ati.aspx. — Visited: April 14, 2012.

17. Wolfram. Mathematica 8: Control System Professional Documentation. 10.3 Riccati Equations // URL: http://reference.wolfram.com/legacy/ applications/control/OptimalControlSystemsDesi gn/10.3.html. — Visited: April 14, 2012.

18. Matlab Toolboxes of SLICOT (Subroutine Library in Systems and Control Theory) // URL: http://www.slicot.org/index.php?site=home. — Visited: April 14, 2012.

19. SLICOT Basic Systems and Control Toolbox // URL: http://www.slicot.org/index.php?site=slbasic. — Visited: April 14, 2012.

20. Benner P., Sima V. Solving Algebraic Riccati Equations with SLICOT // In: CD-ROM Proc. of The 11th Mediterranean Conference on Control and Automation MED'03, June 18-20 2003, Rhodes, Greece, 2003, invited session IV01, Paper IV01-01.

21. Sima V. Computational Experience in Solving Algebraic Riccati Equations // In: Decision and Control, 2005 European Control Conference. CDC-ECC '05. 12-15 Dec. 2005. Pp. 7982-7987.

22. Sima V. Algorithms for Linear-Quadratic Optimization. Vol. 200 of Series Pure and Applied Mathematics: A series of Monographs and Textbooks. — N-Y: Marcel Dekker, 1996.

23. Armstrong J. Updated Discrete Algebraic Riccati Equation Solver in Python // URL: http://jeff.rainbow-100.com/?p=141. — Posted: December 31, 2010 by Jeff. — Visited: April 14, 2012.

24. Wook Hyun Kwon, Young Soo Moon, Sang Chul Ahn. Bounds in Algebraic Riccati and Lyapunov Equations: A Survey and Some New Results // International Journal of Control. — 1996. — Vol. 64. — Iss. 3. — Pp. 377-389. DOI: 10.1080/00207179608921634

25. Bunse-Gerstner A. Computational Solution of the Algebraic Riccati Equation // URL: http://www.math.uni-bremen.de/zetem/numerik/ Published/Report982.ps.gz. — Visited: April 14,

26. Hua Dai, Zhong-Zhi Bai. On Eigenvalue Bounds and Iteration Methods for Discrete Algebraic Riccati Equations // Journal of Computational Mathematics. — 2011. — Vol. 29(3). — Pp. 341366. URL: http://www.jcm.ac.cn/EN/10.4208/ jcm.l010-m3258 OR http://www.jcm.ac.cn/EN/Y2011 /V29/137341. — Visited: April 14, 2012.

27. Semushin I. V. Adaptation in Stochastic Dynamic Systems — Survey and New Results II // Int. J. Communications, Network and System Sciences.- 2011. — Vol.4, No. 4. — Pp. 266-285.

28. Semushin I.V. Computational Methods of Algebra and Estimation : textbook. — Ulyanovsk : UlSTU, 2011. [На русском: Семушин И.В. Вычислительные методы алгебры и оценивания: учеб. пособие. — Ульяновск: УлГТУ, 2011.]

29. Potter J. E., Stern R. G. Statistical Filtering of Space Navigation Measurements // In: Proceedings of 1963 AIAA Guidance and Control Conference. — New York: AIAA, 1963.

30. Schmidt S. F. The Kalman filter: recognition and development for aerospace applications // Journal of Guidance and Control. — 1981. — V. 4. — No. 1. — P. 4.