

# Метод использования памяти микроконтроллера при реализации нелинейных регуляторов, с коэффициентами, зависящими от состояния<sup>1</sup>

А.А. Семион

**Аннотация.** В статье рассматривается задача управления нелинейной системой, позволяющей описать её системой нелинейных дифференциальных уравнений с линейной структурой и параметрами, зависящими от состояния. Показана возможность построения регулятора на основе решения матричных уравнений типа Риккати с параметрами, зависящими от состояния. Предложен способ, позволяющий уменьшить нагрузку на процессор в случае, если множество допустимых состояний системы ограничено. Приведен расчет количества коэффициентов усиления и объема памяти микроконтроллера при заданной точности вычислений на примере модели летательного аппарата. На основе предложенного метода синтезировано управление беспилотным летательным аппаратом, которое затем было проверено путем моделирования в MatLab Simulink.

**Ключевые слова:** кватернионная алгебра, уравнение Риккати, нелинейные системы управления.

## Введение

Летательные аппараты являются объектами, описываемыми сложными нелинейными динамическими системами, требующими высокой скорости обработки данных датчиков и расчета управляющего воздействия. Линейные регуляторы не всегда способны корректно управлять таким объектом в предельных режимах. Эти системы, однако, можно описать системой нелинейных дифференциальных уравнений с линейной структурой и параметрами, зависящими от состояния объекта. Таким образом, возможно применение метода State Dependent Coefficients. Впервые такой подход к синтезу управления был сформулирован в начале 60-х годов [1]. Реализуемые результаты могут быть получены при использовании квадратичного

функционала качества, который позволяет перейти от уравнения Гамильтона-Якоби-Беллмана к уравнению типа Риккати с параметрами, зависящими от состояния (State Dependent Riccati Equation) (SDRE) [2].

Использование метода SDRE усложняется некоторыми проблемами, связанными с неоднозначностью эквивалентных преобразований исходной системы, а также созданием эффективного и быстрого алгоритма решения матричного уравнения типа Риккати с параметрами, зависящими от состояния, в темпе функционирования объекта [3].

В работе [4] показано, что для работы регулятора в реальном времени достаточно находить решения уравнения типа Риккати за 14 мс. Отмечено, что для реализации регулятора по-

<sup>1</sup> Работа выполнена при поддержке гранта РФФИ №16-08-00522 А и в рамках работ по проекту по договору УМНИК №5940ГУ/2015

требовалось использовать компьютер с процессором с частотой 300 МГц, для чего использовалась авионика, вес которой составил 3 кг, что не всегда возможно установить на произвольный летательный аппарат.

В данной статье предлагается приближенный метод, который позволяет смягчить требования к производительности управляющего устройства и облегчить его вес, однако его применение доступно только для определенного класса систем. Метод заключается в расчете коэффициентов усиления нелинейного регулятора с использованием значений состояний системы в конкретных точках траектории. Описываемый метод объединяет идеи синтеза регуляторов для систем с коэффициентами, зависящими от состояния, gain-scheduled регуляторов и метод кэширования.

### Постановка задачи

Будем рассматривать в качестве объекта управления вертолет с четырьмя моторами, расположенными в вершинах квадрата с диагональю длины  $2l$  (квадрокоптер). Моторы квадрокоптера нумеруются против часовой стрелки, начиная с переднего правого (Рис. 1).

Мотор может вращать пропеллер только в направлении, указанном стрелкой. Пропеллер не имеет автомата перекося и не изменяет угол атаки. Наклон аппарата осуществляется за счет разности тяг моторов. Предполагается, что сумма сил тяги всех винто-моторных групп способна поднимать аппарат. При моделировании считается, что отношение максимальной суммы тяги моторов к силе тяжести, действующей на аппарат, равно двум, что примерно соответствует реальным характеристикам аппаратов.

В работе предполагается, что текущее состояние системы точно известно. В реальных аппаратах для этого используется инерционно-измерительное устройство, состоящее в общем случае из трехосевого акселерометра, трехосевого гироскопа и трехосевого магнетометра. Полученные данные с сенсоров поступают на фильтр, а после на управляющее устройство.

Необходимо синтезировать управление, способное стабилизировать аппарат горизонтально в полете.

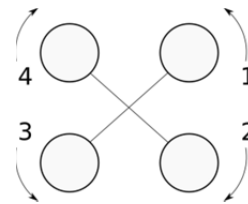


Рис. 1. Нумерация моторов

### Уравнения движения

Для дальнейших рассуждений нам понадобится математическая модель квадрокоптера, которая приведена в [5], где для описания вращения аппарата используется алгебра кватернионов [6].

Воспользуемся уравнениями Эйлера для вращательного движения (основной закон динамики вращательного движения):

$$\tau = I\dot{\omega} + \omega \times I\omega,$$

где  $\tau$  - моменты, действующие на тело,  $I$  - тензор инерции,  $\omega$  - угловые скорости по главным осям (в связанной системе координат).

Выражая угловые ускорения, получаем:

$$\dot{\omega} = I^{-1}(\tau - \omega \times I\omega). \quad (1)$$

Тензор инерции представим в диагональном виде:  $I = \begin{pmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{pmatrix}$ .

Внешние моменты определим через управление:  $\tau = (U_1 \ U_2 \ U_3)^T$ , где

$$U_1 = \frac{\sqrt{2}}{2} l (F_1 + F_2 - F_3 - F_4),$$

$$U_2 = \frac{\sqrt{2}}{2} l (-F_1 + F_2 + F_3 - F_4),$$

$U_3 = \alpha (F_1 - F_2 + F_3 - F_4)$ , а  $F_i$  - силы тяги соответствующих моторов. Коэффициент  $\alpha$  вычисляется эмпирически.

Для получения всех сил необходимо ввести управление  $U_0 = F_1 + F_2 + F_3 + F_4$ , которое должно определяться из задачи стабилизации по высоте, что выходит за рамки данной статьи.

Так же необходимо выписать уравнение для кватерниона вращения, получаемое из свойств кватернионов [6]:

$$\dot{\vec{\lambda}} = \frac{1}{2} \vec{\lambda} \circ \omega. \quad (2)$$

Объединяя (1) и (2), получаем математическую модель летательного аппарата, которая представляет собой систему обыкновенных дифференциальных уравнений

$$\begin{aligned} \frac{d}{dt}\dot{\varphi} &= \frac{1}{I_x}[U_1 - (I_z - I_y)\dot{\theta}\dot{\psi}], \\ \frac{d}{dt}\dot{\theta} &= \frac{1}{I_y}[U_2 - (I_x - I_z)\dot{\varphi}\dot{\psi}], \\ \frac{d}{dt}\dot{\psi} &= \frac{1}{I_z}[U_3 - (I_y - I_x)\dot{\theta}\dot{\varphi}], \\ \frac{d}{dt}\lambda_0 &= \frac{1}{2}[-\lambda_1\dot{\varphi} - \lambda_2\dot{\theta} - \lambda_3\dot{\psi}], \\ \frac{d}{dt}\lambda_1 &= \frac{1}{2}[\lambda_0\dot{\varphi} - \lambda_3\dot{\theta} + \lambda_2\dot{\psi}], \\ \frac{d}{dt}\lambda_2 &= \frac{1}{2}[\lambda_3\dot{\varphi} + \lambda_0\dot{\theta} - \lambda_1\dot{\psi}], \\ \frac{d}{dt}\lambda_3 &= \frac{1}{2}[-\lambda_2\dot{\varphi} + \lambda_1\dot{\theta} + \lambda_0\dot{\psi}], \end{aligned} \quad (3)$$

где  $\dot{\varphi}$ ,  $\dot{\theta}$ ,  $\dot{\psi}$  - угловые скорости в системе координат, связанной с аппаратом,

$\lambda_0, \lambda_1, \lambda_2, \lambda_3$  - компоненты кватерниона, описывающего поворот квадрокоптера в пространстве.

Необходимо отметить, что в таком представлении система является неуправляемой. Однако Yaguan Yang в [7] показывает способ редукции модели, описывающей положение летательного аппарата в алгебре кватернионов. Он доказал, что скалярный компонент кватерниона можно не включать в вектор состояний, заменив  $\lambda_0$  на  $\lambda_0 = f(\vec{\lambda}) = \sqrt{1 - \lambda_1^2 - \lambda_2^2 - \lambda_3^2}$ .

Этот способ позволяет сделать систему (3) управляемой.

Итоговая система представима в стандартном виде:

$$\dot{X} = A(X)X + BU, \quad (4)$$

где  $X = (\dot{\varphi} \ \dot{\theta} \ \dot{\psi} \ \lambda_1 \ \lambda_2 \ \lambda_3)^T$  - вектор пространства состояний,

$U = (U_1 \ U_2 \ U_3)^T$  - вектор управлений,

$$A(X) = \begin{pmatrix} 0 & 0 & \frac{I_y - I_z}{I_x}\dot{\theta} & 0 & 0 & 0 \\ \frac{I_z - I_x}{I_y}\dot{\psi} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{I_x - I_y}{I_z}\dot{\varphi} & 0 & 0 & 0 & 0 \\ \frac{1}{2}f(\vec{\lambda}) & -\frac{1}{2}\lambda_3 & \frac{1}{2}\lambda_2 & 0 & 0 & 0 \\ \frac{1}{2}\lambda_3 & \frac{1}{2}f(\vec{\lambda}) & -\frac{1}{2}\lambda_1 & 0 & 0 & 0 \\ -\frac{1}{2}\lambda_2 & \frac{1}{2}\lambda_1 & \frac{1}{2}f(\vec{\lambda}) & 0 & 0 & 0 \end{pmatrix},$$

$$B = \begin{pmatrix} \frac{1}{I_x} & 0 & 0 \\ 0 & \frac{1}{I_y} & 0 \\ 0 & 0 & \frac{1}{I_z} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

## Синтез управления для объекта

### Регулятор с дискретно изменяющимися параметрами

Регулятор с дискретно изменяющимися параметрами был подробно описан в [3;5]. Интервал функционирования разбивается на равные по длине промежутки. Управление на  $i+1$ -ом промежутке представляется в виде

$$U_i = K(X_i)X_i = -R^{-1}B^T(X_i)S(X_i)X_i, \quad (5)$$

где  $S(X_i)$  - решение уравнения Риккати вида

$$S(X_i)A(X_i) + A^T(X_i)S(X_i) - S(X_i)B(X_i)R^{-1}B^T(X_i)S(X_i) + Q = 0. \quad (6)$$

Матрицу  $K(X_i)$  будем называть коэффициентом усиления.

Использование предложенного управления требует решения уравнения (6) в темпе функционирования объекта. Вычислительных ресурсов современных микроконтроллеров порой может быть недостаточно.

Однако можно заметить, что матрица  $A(X)$  в (4) зависит только от компонент кватерниона и угловых скоростей, а матрица  $B$  постоянна. Считается, что кватернион нормирован, следовательно, все его компоненты лежат в пределах  $[-1;1]$ . Мы можем полагать, что угловые скорости лежат в пределах  $[-v; v]$ .

В действительности скорость вращения летательного аппарата ограничена аэродинамическими силами, мощностью силовых установок, моментом инерции и внешними силами. Предлагается выбирать максимальную угловую скорость такой, при которой аппарат будет в состоянии компенсировать отклонение тягой моторов за адекватное время. Исходя из приведенных выше соображений примем за максимальную угловую скорость:  $v = 20$  рад/с. Мы проведем расчеты для двух случаев: для случая, когда угловыми скоростями пренебрегают

(считают их достаточно близкими к нулю), и случае, когда необходимая точность угловых скоростей принята за 0.1 рад/с.

Исходя из вышесказанного, предлагается вычислить все необходимые решения уравнения (6) заранее и занести их в память управляющего устройства. Такой подход позволит ускорить получение следующего коэффициента, так как скорость чтения памяти современных носителей информации многократно превосходит скорость вычисления решения в темпе функционирования объекта.

### Метод использования памяти управляющего устройства

Примем, что необходимая точность компонент вектора пространства состояний, отвечающих за кватернион, равна двум значащим цифрам после запятой, что соответствует точности в  $\sim 1^\circ$  угла поворота аппарата. Обозначим точность как  $\epsilon$ , тогда можно заметить, что интервал  $[-1;1]$  разбивается на конечное количество точек с расстоянием  $\epsilon$  между ними. Очевидно, что количество возможностей для выбора  $a$ :  $2/\epsilon + 1$ .

При выбранной точности для одной компоненты кватерниона существует 201 различное значение. В векторе состояний содержится 3 компоненты кватерниона и количество различных значений этих компонент равно  $201^3 = 8120601$ , однако в силу нормированности кватерниона количество вариаций можно снизить, так как:  $1 - \lambda_1^2 - \lambda_2^2 - \lambda_3^2 \geq 0, \forall \lambda_i, i=1,2,3$ .

Можно представить, что все возможные значения компонент кватерниона укладываются на точки пересечений 3х-мерной сетки с размером ячейки  $\epsilon$ , ограниченной шаром единичного радиуса с центром в начале координат (Рис. 2).

Точный подсчет количества возможных троек значений компонент кватерниона проведен перебором всех точек внутри куба, описанного около единичного шара, так как использование формул объема не дает точный результат при относительно большом значении  $\epsilon$ .

Количество вариантов троек для нескольких выбранных значений  $\epsilon$  приведено в Табл. 1.

Для выбранной точности существует 4187707 вариантов. Руководствуясь стандартом IEEE.754, описывающим формат представления чисел с плавающей точкой, можно выбрать два наиболее

популярных типа данных для хранения состояния системы: float и double, занимающих 4 и 8 байт соответственно. Так как количество памяти управляющего устройства ограничено, прибежем к использованию формата float.

Для уменьшения необходимых вычислений и размера хранимой информации предлагается вычислить заранее и хранить в памяти микроконтроллера значения матриц  $K(X_i)$  из (5). Так как  $K(X_i)$  имеет размерность  $3 \times 6$ , то для хранения одной матрицы с использованием типа данных float требуется 72 байта. Объемы памяти для случая, когда угловые скорости учитываются, получены исходя из законов комбинаторики и аналогичных рассуждений.

Необходимое количество памяти приведено в Табл. 2.

Для ускорения доступа к коэффициентам предлагается добавить информацию в виде указателей на коэффициенты, вычисленные для

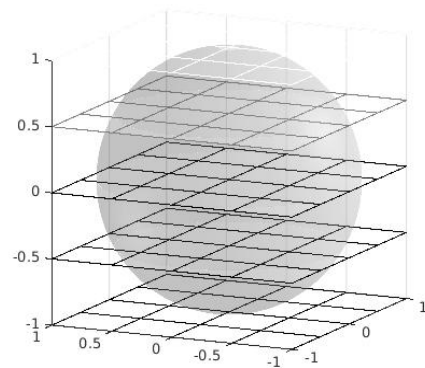


Рис. 2. Сетка в сфере

Табл. 1. Количество вариантов (троек) в зависимости от точности

Точность по компонентам кватерниона	Количество вариантов
0.1	4139
0.01	4187707
0.001	4188780761

Табл. 2. Необходимое количество памяти в зависимости от точности

Точность	Необходимое количество памяти без учета угловых скоростей	Необходимое количество памяти с учетом угловых скоростей
0.1	291КБ	113.96МБ
0.01	287.547МБ	112ГБ
0.001	280ГБ	110ТБ

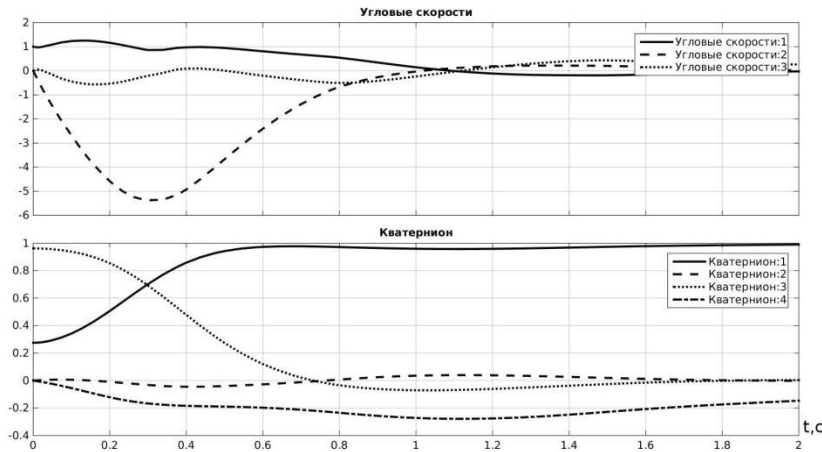


Рис. 3. Траектория движения аппарата

соседних (отличающихся на выбранную точность) состояний. Для каждой компоненты кватерниона существует два соседних состояния. Очевидно, для одной матрицы  $K(X_i)$  необходимо хранить дополнительно 24 байта информации. Для выбранной точности кватерниона потребуется дополнительно 95.8 МБ памяти. Для поиска следующей матрицы коэффициентов необходимо последовательно двигаться от прошлой матрицы по указателям вдоль каждой из компонент, пока разность между текущим состоянием и состоянием, для которого вычислен коэффициент, не станет меньше выбранной точности. Данный подход к расположению данных в памяти является шестисвязным списком и часто используется в прикладных задачах.

Для оценки скорости получения следующего коэффициента предположим, что за один такт состояние изменилось с  $(0; 0; 0; 1; 1; 1)^T$  на  $(0; 0; 0; -1; -1; -1)^T$  при точности 0.01. Для поиска следующей ячейки необходимо пройти по 603 указателям и считать 43КБ. У современных доступных на рынке твердотельных носителей информации максимальная скорость чтения достигает 30МБ/с. Без учета времени, затрачиваемого на операции сравнения, время получения новой матрицы  $K(X_i)$  составляет 1.4 мс.

### Математическое моделирование

Для проверки полученных результатов было проведено численное моделирование. Задачей управления было привести аппарат в горизонтальное положение. В качестве начальных

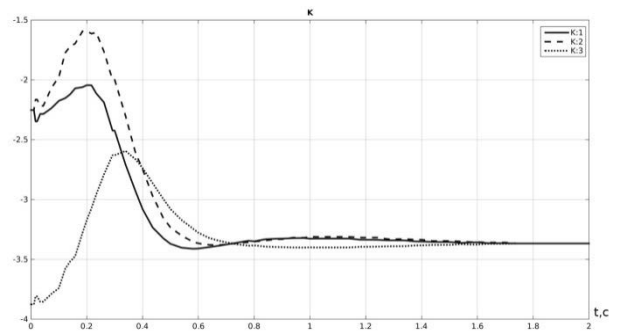


Рис. 4. Изменение трех элементов коэффициента усиления регулятора

условий был выбран вектор  $(\phi \ \theta \ \psi \ \lambda_0 \ \lambda_1 \ \lambda_2 \ \lambda_3)^T = (1; 0; 0; 0.274721; 0; 0.961524; 0)^T$ . Точность подсчета управления  $\epsilon=0.01$ .- Результаты моделирования приведены на Рис. 3

Возмущения по рысканию (последней координате) возникают в связи с тем, что при стабилизации по двум осям неизменно возникает крутящий момент по оставшейся оси.

Для наглядности на Рис. 4 представлен график изменения только трех компонентов матрицы  $K(X_i)$ .

На Рис. 5 проиллюстрирована последовательность выбора матрицы  $K(X_i)$  при изменении состояния системы.

### Заключение

В статье предложен способ реализации регулятора с параметрами, зависящими от состояния, для нелинейной динамической системы с

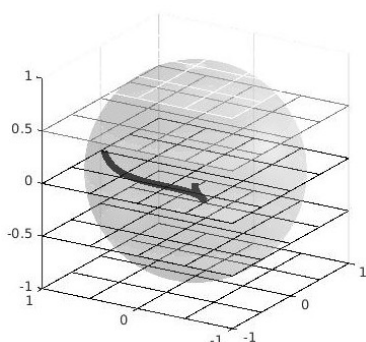


Рис. 5. Последовательность выбора матрицы  $K(X_i)$

хранением части коэффициентов усиления в памяти в условиях ограниченности множества допустимых состояний системы. Приведена оценка количества необходимых для расчета состояний системы и объем памяти, занимаемый коэффициентами усиления регулятора в зависимости от выбранной точности. Описан способ эффективного использования запоминающего устройства, обеспечивающий быстрое получение следующего коэффициента.

**Семион Александр Александрович.** Аспирант НИУ ВШЭ. Количество печатных работ: 2. Область научных интересов: управление нелинейными динамическими системами. E-mail: spin7ion@gmail.com

## Memory of microcontroller usage method for nonlinear regulators with state dependent coefficients realization

A. Semion

This article addresses the control task for nonlinear system that can be presented as nonlinear differential system with linear structure and parameters depending on object state. Usage of quadratic cost function allows developing control with coefficients that include solution of Riccati equation in real time. The rather common way is to solve Riccati equation at rate of object functioning which requires high performance controller what is not appropriate in some applications. The method, represented in this article, is useful when system state space is compact and performance or weight of control hardware is critical. It is offered to calculate regulator coefficients in advance and to keep them in memory of the control device. Calculation of coefficients quantity and memory size depending on accuracy of calculations is provided. The mathematical simulation of aircraft controlled by such regulator was made for verification.

**Keywords:** quaternion algebra, state dependent coefficients, Riccati equation, nonlinear dynamic systems.

### References

1. Pearson J.D. Approximation methods in optimal control // Journal of Electronics and Control, 1962
2. Mracek C.P., Cloutier J.R. Full envelope missile longitudinal autopilot design using the state-dependent Riccati equation method. // In Proc. of the AIAA Guidance, Navigation, and Control Conference. New Orleans, LA. 1997. pp. 1697-1705.
3. Afanasev V.N., Semion A.A. Regulyator s diskretno izmenyayuschimisya parametrami // Problemy upravleniya, May 2014. pp. 14-20.

В дальнейшей работе планируется рассмотреть влияние внешних возмущений.

## Литература

1. Pearson J.D. Approximation methods in optimal control // Journal of Electronics and Control, 1962
2. Mracek C.P., Cloutier J.R. Full envelope missile longitudinal autopilot design using the state-dependent Riccati equation method. // In Proc. of the AIAA Guidance, Navigation, and Control Conference. New Orleans, LA. 1997. pp. 1697-1705.
3. Афанасьев В.Н., Семион А.А. Регулятор с дискретно изменяющимися параметрами // Проблемы управления, May 2014. pp. 14-20.
4. Alexander Bogdanov, Magnus Carlsson, Geoff Harvey, John Hunt, Dick Kieburz, Rudolph van der Merwe, Eric Wan State-dependent Riccati equation control of a small unmanned helicopter // AIAA Guidance, Navigation, and Control Conference and Exhibit Austin, Texas 11-14 August 2003.
5. Семион А. А. Разработка автопилота для квадрокоптера // Качество. Инновации. Образование. 2016. № 6. С. 53-67.
6. Бранец В.Н., Шмыглевский И.П. Применение кватернионов в задачах ориентации твердого тела. Москва: Издательство "Наука", 1973.
7. Yang, Yaguang Analytic LQR Design for Spacecraft Control System Based on Quaternion Model, Journal of aerospace engineering, 25, 3, JULY 2012., p. 448-453.

4. Alexander Bogdanov , Magnus Carlsson, Geoff Harvey, John Hunt, Dick Kieburtz, Rudolph van der Merwe, Eric Wan State-dependent Riccati equation control of a small unmanned helicopter // AIAA Guidance, Navigation, and Control Conference and Exhibit Austin, Texas 11-14 August 2003.
5. Semion A. A. Razrabotka avtopilota dlya kvadroptera // Kachestvo. Innovatsii. Obrazovanie. 2016. № 6. С. 53-67.
6. Branets V.N., Shmyiglevskiy I.P. Primenenie kvaternionov v zadachah orientatsii tverdogo tela. Moskva: Izdatelstvo "Nauka", 1973.
7. Yang, Yaguang Analytic LQR Design for Spacecraft Control System Based on Quaternion Model, Journal of aerospace engineering, 25, 3, JULY 2012., p. 448-453

**Alexander Alexandrovich Semion** PhD student at National Research University Higher School of Economics. Has 2 publications. Science interests: nonlinear dynamic systems control. E-mail: spin7ion@gmail.com