УДК 519.7

DOI: 10.15827/0236-235X.124.705-709

Дата подачи статьи: 11.09.18

2018. Т. 31. № 4. С. 705–709

Оптимальное управление с обратной связью одним классом нелинейных систем по квадратичному критерию

А.П. Афанасьев¹, д.ф.-м.н., профессор, apa@isa.ruС.М. Дзюба², д.ф.-м.н., профессор, sdzyuba@mail.ruА.Н. Пчелинцев³, к.ф.-м.н., доцент, pchelintsev.an@yandex.ruИ.И. Емельянова², ст. преподаватель, emelyanova-123@yandex.ru¹ Институт проблем передачи информации РАН, г. Москва, 127994, Россия² Тверской государственный технический университет, г. Тверь, 170026, Россия³ Тамбовский государственный технический университет, г. Тамбов, 392000, Россия

Статья посвящена проблеме синтеза оптимального управления нелинейными системами с выделенной линейной частью по квадратичному критерию. Данная проблема имеет достаточно большое значение, поскольку многие модели технических, физических, биологических и экономических систем описываются именно такими моделями.

Приведен метод синтеза оптимального управления с обратной связью. Он базируется на специальном методе последовательных приближений и является развитием результатов, полученных в более ранних работах авторов, в которых доказано: если время работы системы достаточно мало, то предложенный метод равномерно сходится. Это позволяет установить существование решения исходной задачи в виде закона управления с обратной связью и дает процедуру приближенного построения этого закона.

Если время работы системы велико, как это обычно бывает на практике, то доказать равномерную сходимость метода последовательных приближений не удастся. Это объясняется спецификой метода. Однако многочисленные вычислительные эксперименты показали, что простая сходимость метода наблюдается во многих случаях. Более того, можно показать, что из сходимости метода на произвольном промежутке времени следует существование решения исходной задачи. Это дополнительно дает ее (задачи) приближенное решение, причем в виде закона управления с обратной связью.

Таким образом, предложенный метод является обоснованным для решения задач во многих практических ситуациях. В статье приведен пример его использования для решения задачи стабилизации режимов работы регулятора Уатта. Данный пример имеет достаточно большое практическое значение, поскольку стабилизация работы этого регулятора является классической задачей теории управления. Для численной реализации примера использовались методы объектно-ориентированного программирования. Численный эксперимент показал, что метод позволяет стабилизировать работу регулятора Уатта в достаточно широком диапазоне извлечения параметров.

Ключевые слова: метод последовательных приближений, оптимальное управление с обратной связью, нелинейная система, объектно-ориентированное программирование.

Проблема синтеза оптимального управления нелинейными системами с выделенной линейной частью по квадратичному критерию является актуальной. Многие модели технических, физических, биологических и экономических систем описываются именно такими моделями.

Для решения исходной задачи в виде закона управления с обратной связью рассмотрим нелинейную динамическую систему, характеризуемую дифференциальным уравнением

$$\dot{x} = Ax + Bu + f(x, u), \quad (1)$$

где $x = (x^1, \dots, x^n)$ – n -мерный действительный вектор состояния; $u = (u^1, \dots, u^m)$ – m -мерный действительный вектор управления; A и B – действительные $(n \times n)$ и $(n \times m)$ матрицы; $f = (f^1, \dots, f^n)$ – векторная функция, определенная и непрерывная

вместе со своими частными производными $\frac{\partial f^i}{\partial x^j}$,

$i, j = 1, \dots, n$; $\frac{\partial f^i}{\partial u^j}$, $i = 1, \dots, n$; $j = 1, \dots, m$ в пространстве R^{n+m} .

Предположим, что начальное состояние

$$x(0) = c \quad (2)$$

задано, а задача управления системой (1) заключается в минимизации функционала

$$J(u) = \frac{1}{2} \int_0^T [\langle e(t), Qe(t) \rangle + \langle u(t), Ru(t) \rangle] dt + \frac{1}{2} \langle e(T), Pe(T) \rangle, \quad (3)$$

в котором T – фиксированное конечное время; Q и P – положительные полуопределенные $(n \times n)$ матрицы; R – положительно определенная $(m \times m)$ матрица; $e(t) = x(t) - z(t)$ – ошибка системы; $z = (z^1, \dots, z^n)$ – заданный режим функционирования.

Легко заметить, что непосредственное применение к рассматриваемой задаче принципа максимума Л.С. Понтрягина приводит к достаточно сложной краевой задаче, если только (1)–(3) не сводится к линейно-квадратичной задаче слежения (см., например, [1]). Однако во многих практических ситуациях режим $z(t)$ устроен так, что указанное сведение невозможно.

В общем случае для получения решения (или оценки решения) задачи используют самые разные методы (см., например, [2] и [3–9]). Одним из основных здесь является метод последовательных приближений, описанный в [2]. Внешне простой и понятный, он сводит исходную задачу к некоторой последовательности линейно-квадратичных задач. Вместе с тем он до сих пор не получил широкого распространения, поскольку его сходимость не доказана. Последнее, в частности, объясняется тем, что в схеме последовательных приближений оператор системы меняется от итерации к итерации.

Цель настоящей работы – получить решение задачи (1)–(3) в виде закона управления с обратной связью. Для получения искомого решения используется процедура, являющаяся модификацией метода [2] и заключающаяся в создании некоей специальным образом генерируемой последовательности вспомогательных линейно-квадратичных задач. Это позволяет в некоторых случаях установить сходимость метода, а из сходимости доказать существование оптимального управления и получить процедуру его построения.

Метод построения приближенных решений

Формально опишем метод последовательных приближений, на котором будут базироваться все дальнейшие построения.

Следуя [9], для всех $N = 0, 1, \dots$ рассмотрим вспомогательную задачу о минимизации функционала:

$$J_{N+1}(u) = \frac{1}{2} \int_0^T [\langle e(t), Qe(t) \rangle + \langle u(t), Ru(t) \rangle] dt + \frac{1}{2} \langle e(T), Pe(T) \rangle \quad (4)$$

с ограничением

$$x' = Ax + Bu + f(x_N, u_N), \quad x(0) = c, \quad (5)$$

где x_N и u_N – некоторые функции, определенные и непрерывные на отрезке $[0, T]$.

Для фиксированных x_N и u_N оптимальное управление $u_{N+1}(t)$ в задачах (4) и (5) дается законом управления с обратной связью

$$u_{N+1}(t) = R^{-1}B'[h_{N+1}(t) - K(t)x_{N+1}(t)], \quad (6)$$

в котором $x_{N+1}(t)$ – решение уравнения (5), соответствующее $u_{N+1}(t)$ и удовлетворяющее начальному условию

$$x_{N+1}(0) = c; \quad (7)$$

$K(t)$ – решение матричного дифференциального уравнения Риккати

$$K'(t) = -K(t)A - A'K(t) + K(t)BR^{-1}B'K(t) - Q \quad (8)$$

с граничным условием

$$K(T) = P; \quad (9)$$

$h_{N+1}(t)$ – решение линейного дифференциального уравнения

$$h'_{N+1}(t) = -[A - BR^{-1}B'K(t)]h_{N+1}(t) - Qz(t) + K(t)f(x_N(t), u_N(t)) \quad (10)$$

с граничным условием (см. [1])

$$h_{N+1}(T) = Pz(T) \quad (11)$$

Таким образом, если начальное приближение $x_0(t)$, $u_0(t)$ задано, то соотношения (4)–(11) определяют схему последовательных приближений, которая, как будет показано далее, при всех достаточно малых значениях T позволяет установить существование решения задачи (1)–(3) и дает эффективную процедуру построения этого решения. Отметим также, что для простоты начальное приближение здесь будет определено соотношениями

$$x_0(t) = c \quad (12)$$

$$\text{и } u_0(t) \equiv R^{-1}B'[Pz(T) - K(t)c]. \quad (13)$$

В [9] показано: если величина T достаточно мала, то решение задачи (1)–(3) существует и может быть получено сходящимся в данном случае методом последовательных приближений (4)–(13).

И наоборот, легко видеть, что в случае сходимости метода последовательных приближений (4)–(13) решение задачи (1)–(3) существует и может быть получено методом (4)–(13) вне зависимости от значения T .

Таким образом, метод (4)–(13) является обоснованным для решения задач во многих практических ситуациях. Далее приведен пример его использования для решения стабилизации режимов работы регулятора Уатта.

Иллюстративный пример

Рассмотрим систему дифференциальных уравнений

$$\begin{aligned} \varphi' &= \psi, \\ \psi' &= \alpha\omega^2 \sin \varphi \cos \varphi - g \sin \varphi - \beta\psi, \\ \omega' &= \gamma \cos \varphi - bu^1(t), \end{aligned} \quad (14)$$

где α , β , γ , b и g – параметры системы, а $u^1(t)$ – управление.

Система (14) может быть приведена к виду (1) следующим образом:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\beta & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ -b \end{bmatrix},$$

$$x(t) = \begin{bmatrix} x^1(t) \\ x^2(t) \\ x^3(t) \end{bmatrix} \equiv \begin{bmatrix} \phi(t) \\ \psi(t) \\ \omega(t) \end{bmatrix}, \quad u(t) = [u^1(t)],$$

$$f(x) = \begin{bmatrix} 0 \\ \alpha\omega^2 \sin \varphi \cos \varphi - g \sin \varphi \\ \gamma \cos \varphi \end{bmatrix}.$$

Для критерия (3)

$$Q = \begin{bmatrix} q^1 & 0 & 0 \\ 0 & q^2 & 0 \\ 0 & 0 & q^3 \end{bmatrix}, R = [r], R^{-1} = \begin{bmatrix} 1 \\ r \end{bmatrix}, P = k_p E,$$

где E – единичная матрица, а значения T (длина временного отрезка) q^1, q^2, q^3, r и k_p считаются заданными. Предположим, что вектор z постоянен и его элементы также заданы.

Пусть в формулах (5) и (7) $c = k_z Z + \Delta_z$,

$$\Delta_z = \begin{bmatrix} \delta_z \\ \delta_z \\ \delta_z \end{bmatrix}, \text{ где } k_z - \text{усиливающий коэффициент;}$$

δ_z – отклонение от рабочего режима z .

Для решения задачи (1)–(3) для заданного значения T разработана компьютерная программа на языке C++, основанная на описанном методе последовательных приближений (5)–(13). Чтобы реализовать его, вначале нужно численно решить матричное уравнение Риккати (8) в обратном направлении по времени (см. граничное условие (9)). Причем оно решается один раз, и массив-сетка матриц (в программе она относится к типу `vector<matrix>`) хранится в памяти для дальнейшего использования в схеме последовательных приближений (5)–(13) (см. в [10] строки 222–227). Для эффективной программной реализации алгоритма построения приближений к $u^1(t)$ решение уравнений (8), а также (5) и (10) на каждой итерации с точки зрения уменьшения дублирования кода нужно использовать принцип полиморфизма объектно-ориентированного программирования. Суть в том, что численное решение обыкновенных дифференциальных уравнений универсально по форме уравнения – подходит для решения и скалярного, и векторного, и матричного дифференциального уравнений. Такое возможно, если использовать шаблоны классов C++. Также сам численный алгоритм решения уравнения правильное вынести в отдельный шаблонный класс с чисто виртуальной функцией $f()$, перекрытие которой в потомке реализует вычисление конкретной правой части системы дифференциальных уравнений.

Заметим, что на каждой итерации с номером $N+1$ сначала численно решается уравнение (10) в обратном времени, так как его правая часть не зависит от $x_{N+1}(t)$. Далее полученная сетка для векторной функции $h_{N+1}(t)$ используется при численном интегрировании уравнения (5) в прямом времени.

Представим фрагмент кода (полная программа приведена в [11]), где объявлен универсальный метод численного решения дифференциальных уравнений для задачи (1)–(3) оптимального управления:

```
#ifndef _RK4_H_INCLUDED_
#define _RK4_H_INCLUDED_
```

```
#include <iostream>
#include <vector>

template <class TypeSol, class TypeK>
class rk4
{
public:
    void solve(const double &t0, int n, const
TypeSol &x0,
                std::vector<TypeSol> &x, const
double &h, int d = 1, bool info = true,
                std::vector<TypeSol> *x_prev_t =
NULL, std::vector<TypeSol> *u_prev_t = NULL,
                std::vector<TypeK> *K_t = NULL,
                std::vector<TypeSol> *H_t = NULL,
                std::vector<TypeSol> *u_t =
NULL)
    {
        x.push_back(x0);
        TypeSol k1, k2, k3, k4, x_c, x_new;
        double t = t0, s = d*h, s2 = s/2;
        TypeSol *x_prev, *u_prev, *H = NULL;
```

Сделаем пояснения: $t0$ – значение начального момента времени (в данном случае оно может быть равно или 0, или T); n – количество шагов численного интегрирования; $x0$ – значение начального условия; x – переменная (вектор библиотеки STL языка C++), куда будут добавляться узлы приближенного решения при выполнении численного интегрирования; h – шаг интегрирования; d – направление интегрирования (1 – прямое по времени, -1 – обратное по времени); $info$ – вывод информации о вычислениях (да/нет); x_prev_t – указатель на вектор, содержащий сетку приближенных значений $x_N(t)$ на предыдущей итерации (с номером N) (по умолчанию равен NULL, так как, например, для решения уравнения Риккати он не нужен); u_prev_t – указатель на вектор, содержащий сетку приближенных значений векторной функции управления $u_N(t)$ на предыдущей итерации (с номером N) (по умолчанию равен NULL); K_t – указатель на вектор, содержащий уже полученную сетку приближенных значений (матриц) решения уравнения Риккати (по умолчанию равен NULL); H_t – указатель на вектор, содержащий сетку приближенных значений решения $h_{N+1}(t)$ уравнения (10) (по умолчанию равен NULL, нужен при решении уравнения (5), когда вычисляется управление по формуле (6)); u_t – указатель на вектор (по умолчанию тоже равен NULL, нужен при решении уравнения (5)), содержащий пересчитанную по формуле (6) сетку приближенных значений текущего управления $u_{N+1}(t)$ по сетке $x_{N+1}(t)$ и сетке, описывающей решение уравнения (10) в обратном времени.

Представим фрагмент кода (полная программа приведена по ссылке [10]), в котором реализовано вычисление векторной функции f (см. правую часть уравнения (1)), а также решение векторного дифференциального уравнения (10):

```
inline double f1()
{
    return 0;
}
```

```

inline double f2(const double &phi, const
double &omega)
{
    return sys_alpha * omega * omega * sin(phi)
    * cos(phi) - sys_g * sin(phi);
}

inline double f3(const double &phi)
{
    return sys_gamma * cos(phi);
}

inline vector_col calc_f_vect(const
vector_col &x)
{
    vector_col f_vect(3);
    f_vect(0) = f1();
    f_vect(1) = f2(x(0), x(2));
    f_vect(2) = f3(x(0));
    return f_vect;
}

class eq_H: public rk4 <vector_col, matrix>
{
public:
    vector_col f(const double &t, const
vector_col &x, const vector_col *x_prev,
const vector_col *u_prev,
const matrix *K, const vector_col *H)
    {
        return -prod(trans(a - prod(prod_brb,
*K)), x) - prod(Q, calc_z_vect(t)) +
        prod(*K, calc_f_vect(*x_prev));
    }
};

```

Как видно, матричные операции в формуле (10) достаточно просто реализованы средствами языка C++, а точнее – использованы алгоритмы и шаблонные типы библиотеки uBLAS линейной алгебры собрания библиотек классов Boost. Было проведено профилирование кода решения задачи (1)–(3) с собственной реализацией матричных операций и с использованием uBLAS. Библиотека uBLAS дала большой выигрыш по времени решения данной задачи.

Результаты решения задачи оптимального управления для системы (14) в графической форме приведены на сайте (см. <http://www.swsys.ru/uploaded/image/2018-4/2018-4-dop/1.jpg>). При этом за-

даны следующие значения: $T = 6$, $q^1 = q^2 = q^3 = 3$, $k_p = 10$, $\beta = 15$, $\gamma = 1$, $b = 0,5$, $g = 9,8$, $k_z = 1$, $\delta_z = -1$.

Заключение

Ключевым для настоящей работы является метод последовательных приближений (5)–(13).

Для всех достаточно малых значений T метод (5)–(13) по построению равномерно сходится к решению задачи (1)–(3). При этом сходимость метода устанавливает существование решения задачи (1)–(3) для малых T и в пределе дает оптимальный закон управления с обратной связью.

Исследование выполнено при финансовой поддержке РНФ в рамках научного проекта № 16-11-10352.

Литература

1. Атанс М., Фалб П. Оптимальное управление. М.: Машиностроение, 1968. 764 с.
2. Беллман Р. Процессы регулирования с адаптацией. М.: Наука, 1964. 359 с.
3. Lukes D.L. Optimal regulation of nonlinear systems. SIAM J. Control Optim., 1969, vol. 7, pp. 75–100.
4. Yamamoto Y. Optimal control of nonlinear systems with quadratic performance. J. Math. Anal. Appl., 1978, vol. 64, pp. 348–353.
5. Dacka C. On the controllability of a class of nonlinear systems. IEEE Trans. Automat. Control, 1980, vol. 25, pp. 263–266.
6. Balachandran K., Somasundaram D. Existence of optimal control for nonlinear systems with quadratic performance. J. Austral. Math. Soc. ser. B., 1987, vol. 29, pp. 249–255.
7. Afanas'ev A.P., Dzyuba S.M., Lobanov S.M., Tyutyunik A.V. Successive approximation and suboptimal control of systems with separated linear part. Appl. Comp. Math., 2003, vol. 2, no. 1, pp. 48–56.
8. Afanas'ev A.P., Dzyuba S.M., Lobanov S.M., Tyutyunik A.V. On a suboptimal control of nonlinear systems via quadratic criteria. Appl. Comp. Math., 2004, vol. 3, no. 2, pp. 158–169.
9. Афанасьев А.П., Дзюба С.М. Об оптимальном управлении нелинейными системами по квадратичному критерию // Тр. ИСА РАН. 2008. Т. 32. С. 49–62.
10. Модуль stabil_task.cpp. 2018. URL: https://github.com/alpchelintsev/stabil_task/blob/master/stabil_task.cpp (дата обращения: 13.07.2018).
11. Модуль rk4.hpp. 2018. URL: https://github.com/alpchelintsev/stabil_task/blob/master/rk4.hpp (дата обращения: 13.07.2018).

Optimal feedback control for one class of nonlinear systems via quadratic criterion

A.P. Afanasyev¹, Dr.Sc. (Physics and Mathematics), Professor, apa@isa.ru

S.M. Dzyuba², Dr.Sc. (Physics and Mathematics), Professor, sdzyuba@mail.ru

A.N. Pchelintsev³, Ph.D. (Physics and Mathematics), Associate Professor, pchelintsev.an@yandex.ru

I.I. Emelyanova², Senior Lecturer, emelyanova-123@yandex.ru

¹ Institute for Information Transmission Problems of the Russian Academy of Sciences, IITP RAS, Moscow, 127994, Russian Federation

² Tver State Technical University, Tver, 170026, Russian Federation

³ Tambov State Technical University, Sovetskaya St. 106, Tambov, 392000, Russian Federation

Abstract. The paper considers the problem of synthesizing optimal control for non-linear systems with a separated linear part upon quadratic criterion. This problem is rather significant, since such models are used to described many models of technical, physical, biological and economical systems.

The paper presents a method of synthesizing optimal feedback control. This method is based on a special method of successive approximations. It is developed from the results described in earlier papers of the authors. These papers have proved that, if the system operation time is small enough, then the suggested method converges uniformly. This establishes the existence of a solution for the initial problem in the form of a feedback control law, and provides a procedure for approximate development of this law.

If the system operation time is long, as it usually happens, then it is impossible to prove the uniform convergence of the successive approximations method due to the specific nature of the method. However, numerous computational experiments have demonstrated that simple convergence of the method can be in many cases. Moreover, it can be demonstrated that, if the method converges within a random time interval, its convergence implies the existence of a solution for the initial problem. Thus, it additionally provides an approximate solution of the problem in the form of a feedback control law.

Therefore, the suggested method is justified for solving problems in many real world situations. The paper presents an example of applying this method for solving the problem of stabilization of Watt governor operation modes. This example has practical significance, as stabilization of Watt governor is a classical problem of the control theory. The numerical implementation of the example applies the methods of object-oriented programming. The numerical experiment has shown that the method in question allows stabilizing Watt governor operation within a rather wide range of parameter extraction.

Keywords: method of successive approximations, optimal feedback control, non-linear system, object-oriented programming.

Acknowledgements. The research has been financially supported by the Russian Science Foundation within the scientific project no. 16-11-10352.

References

1. Athans M., Falb P. *Optimal Control*. NY, McGraw-Hill Publ., 1966, 879 p.
2. Bellman R. *Adaptive Control Processes*. Princeton, Princeton Univ. Press, 1961, 276 p.
3. Lukes D.L. Optimal regulation of nonlinear systems. *SIAM J. Control Optim.* 1969, vol. 7, pp. 75–100.
4. Yamamoto Y. Optimal control of nonlinear systems with quadratic performance. *J. Math. Anal. Appl.* 1978, vol. 64, pp. 348–353.
5. Dacka C. On the controllability of a class of nonlinear systems. *IEEE Trans. Automat. Control.* 1980, vol. 25, pp. 263–266.
6. Balachandran K., Somasundaram D. Existence of optimal control for nonlinear systems with quadratic performance. *J. Austral. Math. Soc. Ser. B.* 1987, vol. 29, pp. 249–255.
7. Afanas'ev A.P., Dzyuba S.M., Lobanov S.M., Tyutyunnik A.V. Successive approximation and suboptimal control of systems with separated linear part. *Appl. Comp. Math.* 2003, vol. 2, no. 1, pp. 48–56.
8. Afanas'ev A.P., Dzyuba S.M., Lobanov S.M., Tyutyunnik A.V. On a suboptimal control of nonlinear systems via quadratic criteria. *Appl. Comp. Math.* 2004, vol. 3, no. 2, pp. 158–169.
9. Afanas'ev A.P., Dzyuba S.M. Optimal control of nonlinear systems via quadratic criteria. *Trans. of ISA RAS.* 2008, vol. 32, pp. 49–62 (in Russ.).
10. *The module stabil_task.cpp*. 2018. Available at: https://github.com/alpchelintsev/stabil_task/blob/master/stabil_task.cpp (accessed July 13, 2018).
11. *The module rk4.hpp*. 2018. Available at: https://github.com/alpchelintsev/stabil_task/blob/master/rk4.hpp (accessed July 13, 2018).

Примеры библиографического описания статьи

1. Афанасьев А.П., Дзюба С.М., Пчелинцев А.Н., Емельянова И.И. Оптимальное управление с обратной связью одним классом нелинейных систем по квадратичному критерию // Программные продукты и системы. 2018. Т. 31. № 4. С. 705–709. DOI: 10.15827/0236-235X.124.705-709.
2. Afanasyev A.P., Dzyuba S.M., Pchelintsev A.N., Emelyanova I.I. Optimal feedback control for one class of nonlinear systems via quadratic criterion. *Software & Systems*. 2018, vol. 31, no. 4, pp. 705–709 (in Russ.). DOI: 10.15827/0236-235X.124.705-709.