

УДК 518:51:62-50

ПРИБЛИЖЕННОЕ РЕШЕНИЕ НЕКОТОРЫХ ЗАДАЧ
ОПТИМАЛЬНОГО УПРАВЛЕНИЯ

Р. П. ФЕДОРЕНКО

(Москва)

Возникающие в настоящее время в приложениях вариационные задачи имеют существенно неклассический характер из-за необходимости учитывать некоторые ограничения на область изменения фазовых переменных и искомых управляющих функций. Теоретические основы таких задач разработаны Л. С. Понтрягиным и его учениками. Хотя использовать сформулированный Понтрягиным «принцип максимума» непосредственно для нахождения оптимального управления пока удается только в самых простых случаях, тем не менее то геометрически очень наглядное доказательство «принципа», которое приведено в [1], подсказало автору способ приближенного решения задач оптимального управления. Настоящая работа посвящена в основном описанию численного метода и иллюстрации его эффективности на задачах, возникающих в приложениях, однако в ходе практической работы оказалось полезным несколько изменить формализм, развитый в [1], с тем, чтобы охватить возникшие задачи. Эти изменения носят в основном формальный характер и касаются вопросов, перечисленных ниже.

1. Мы рассматриваем задачу с общим дифференциальным оператором (а не с оператором, соответствующим задаче Коши, как в [1]). Это обобщение было сделано для задачи о нахождении оптимальных композиций защит от излучения.

В этом случае оператор L_0 является дифференциальным выражением, в котором x имеет более 30 компонент, и — три компоненты. «Краевые» условия, обеспечивающие существование L_0^{-1} , задаются как при t_0 , так и при t_1 , а также в некоторой средней точке интервала (t_0, t_1) . Дополнительные условия на решение сформулированы с использованием значений x в разных точках интервала.

Некоторые свойства оператора L_0 в этой задаче иллюстрируются примером в § 1.

2. Так как «принцип максимума» используется лишь как критерий, отличающий оптимальное управление от улучшаемого, то нас удовлетворяет формулировка его в форме неравенства.

3. Мы ограничились бесконечно малыми вариациями управления: именно на этой основе строились вычисления.

4. В наших вычислениях очень полезным оказался введенный в [1] конус достижимости; в связи с этим во всех формулировках фигурирует

не одно решение сопряженного уравнения, а матрица, составленная из таких решений; эта матрица дает возможность вычислять векторы, из которых строится конус достижимости. Легко видеть, что такая матрица автоматически учитывает условия трансверсальности.

Из работ, посвященных численному решению задач оптимального управления, отметим [2], где предлагается неклассическую вариационную задачу свести к классической (без ограничений), добавляя к исходному минимизируемому функционалу члены, учитывающие невыполнение условий на правом конце траектории, а также члены, учитывающие степень нарушения ограничений на управление и фазовые траектории. (Это, разумеется, уже другая задача, с другим решением; отличие от решения исходной задачи определяется тем, как введен новый функционал.)

В настоящей работе сделана попытка решать задачу, сохраняя ее неклассический характер. В связи с этим приходится прибегать к идеям линейного программирования.

§ 1. Постановка задачи

Рассматриваются следующие задачи оптимального управления. Пусть $x(t) = \{x^1, x^2, \dots, x^n\}$ — вектор-функция, определенная на отрезке (t_0, t_1) ; вектор $u(t) = \{u_1, u_2, \dots, u_k\}$ определен на том же отрезке; $u(t)$ — искомое управление; на область его значений наложено условие

$$u(t) \in U, \quad (1)$$

где U — некоторая замкнутая область. Ищется решение уравнения

$$L_0(u)x = f(x, u), \quad (2)$$

где $L_0(u)$ — некоторое дифференциальное выражение, имеющее смысл при любом заданном управлении (1). На $x(t)$ наложены определенные условия вида

$$F_i(x) = 0, \quad i = 1, 2, \dots, m \quad (3)$$

(m никак не связано, вообще говоря, с n).

Мы будем рассматривать функционалы F_i вида

$$F_i(x) = F_i[x(t'_i), x(t''_i), \dots]; \quad t_0 \leq t'_i, t''_i, \dots \leq t_1, \quad (4)$$

т. е. F_i — функция от значений x в точках t'_i, t''_i, \dots , которую мы будем предполагать дифференцируемой.

Ставится задача о нахождении управления $u(t) \in U$, для которого $x(t)$, удовлетворяющий (2) и условиям (3), давал бы минимальное значение функционалу $F_0(x)$ вида (4). Ниже предлагается метод численного решения подобных задач. Этот метод применим и к задачам, в которых нужно минимизировать

$$\max_{t_0 \leq t \leq t_1} F_0[x(t)], \quad (5)$$

а на область значений $x(t)$ наложено условие

$$Q[x(t)] \leq 0 \quad \text{при } t \in (t_0, t_1). \quad (6)$$

Среди условий (3) выделим некоторую группу условий

$$F_{i_1} = F_{i_2} = \dots = 0 \quad (7)$$

таким образом, чтобы дифференциальный оператор, определяемый выражением (2) и «краевыми» условиями (7), имел обратный при любом $u \in U$. Формально такое выделение неоднозначно. Однако требование иметь в своем распоряжении эффективную и корректную процедуру реализации L_0^{-1} существенно сужает этот производ.

Пример:

$$L_0(u)x \equiv \begin{pmatrix} d/dt & \sigma_1(u) \\ \sigma_2(u) & d/dt \end{pmatrix} \begin{pmatrix} x^1 \\ x^2 \end{pmatrix} = 0;$$

$$\sigma_1(u), \sigma_2(u) \sim +30, +40; \quad t_0 = 0; \quad t_1 = 1.$$

Условия: а) $x'(0) - X_1 = 0$; б) $x^1(1) - X_1' = 0$; в) $x^2(0) - X_2 = 0$.

Формально можно, следуя [1], включить а) и в) в оператор $L_0(u)$ и реализовать L_0^{-1} процедурой Рунге — Кутты, например, а условие б) рассматривать как дополнительное условие на решение. Однако это привело бы нас в дальнейшем к вычислительно некорректным задачам. Правильным является определение оператора L_0 с помощью условий а) и б); условие же в) становится дополнительным; L_0^{-1} корректно реализуется методом прогонки. В дальнейшем мы будем предполагать, что такое выделение проделано; под $L_0(u)$ будем подразумевать оператор, имеющий при заданном $u(t)$ корректно и эффективно реализуемый обратный $L_0^{-1}(u)$, а условия (3) — оставшиеся дополнительные условия на решение. Относительно L_0 предположим также, что для него справедлива теория малых возмущений. Отметим еще, что иногда полезно часть условий, необходимых для существования L_0^{-1} , считать элементами управления (сосредоточенными параметрами) и проводить по ним процесс варьирования, хотя формально можно было бы получить существование L_0^{-1} с помощью части дополнительных условий (3).

Суть метода состоит в построении перехода от $u(t)$ к близкому к нему $u^*(t)$. При этом переходе всегда выполняются (1); условия (3) выполнены с необходимой точностью, а значение F_0 понижается (если $u(t)$ не оптимальное управление). Для построения такого процесса нам необходимо знать связь между произвольной вариацией управления $\delta u(t)$ и вариациями функционалов F_i , $i = 0, 1, \dots, m$.

Проварьировав (2), получим

$$L(u, x) \delta x = M(x, u) \delta u. \quad (8)$$

Здесь $L(u, x)$ — линейный дифференциальный оператор, определенный на невозмущенной «траектории»; M — аналитичная матрица, действующая из пространства векторов u в пространство векторов x .

Варьируя (4), получим

$$\delta F = f' \delta x(t') + f'' \delta x(t'') + \dots \quad (9)$$

Определим L^* формулой *)

$$\int_{t_0}^{t_1} (\psi L \delta x - \delta x L^* \psi) dt = 0. \quad (10)$$

*) Подобным способом удобно рассматривать и некоторые задачи с операторами в частных производных ($\partial/\partial t - \partial^2/\partial x^2$; $\partial^2/\partial x^2 + \partial^2/\partial y^2$; и т. д.).

Беря в качестве $\psi(t)$ решение (операторного) уравнения*

$$L^* \phi_i = \bar{r}_i \delta(t - t_i) + \bar{r}_i'' \delta(t - t_i'') + \dots \quad (11)$$

(здесь $\delta(t - t')$ есть δ -функция с полюсом в t'), из (8) — (11) получаем

$$\delta F_i = \int_{t_0}^{t_1} \phi_i M \delta u \, dt. \quad (12)$$

Решив $(m + 1)$ раз уравнение (11) для $i = 0, 1, \dots, m$, получим матрицу $\Psi(t)$. Рассмотрим $(m + 1)$ -мерное пространство R_{m+1} функционалов. Тогда следствием перехода от u к $u^* = u + \delta u$ будет смещение точки $F(u)$ в точку

$$F(u^*) = F(u) + \int_{t_0}^{t_1} \Psi M \delta u \, dt$$

(так как $u(t)$ определяет $x(t)$ однозначно, то формально можно считать F_i функционалами от u ; в дальнейшем члены порядка $(\delta u)^2$ опущены).

Матрица $\Psi(t)$ даст возможность эффективно строить конус достижимости. Принцип максимума можно сформулировать, потребовав для оптимальной траектории существования вектора $r \in R_{m+1}$ (точнее, R_{m+1}^*) такого, что $r \Psi M \delta u \leq 0$ для всех t и δu , совместных с (1), и $r_0 < 0$.

Предполагая решать задачи с минимизацией (5) и с условиями (6), заметим, что точки t_i' , t_i'' , ... не должны обязательно задаваться заранее; они могут определяться некоторым свойством траектории, например как точки максимума функции $F_0[x(t)]$. Если F_0 достигает максимума в нескольких изолированных точках t_0', t_0'', \dots , то каждой из них соответствует своя строка матрицы $\Psi(t)$, а в формулировке «принципа» придется потребовать выполнение условий $r_0' < 0$; $r_0'' < 0$, ... (Размерность R соответственно возрастает.) Если траектория касается поверхности $Q = 0$ в нескольких точках, то каждой такой точке соответствует своя строка Ψ , а на соответствующие компоненты r накладывается условие неотрицательности. Однако практически важен тот случай, когда F_0 достигает максимума на интервале и когда на некотором интервале $Q = 0$. Исходя из сказанного, разберем случай, когда $Q[x(t)] = 0$ на отрезке (t_α, t_ω) .

Пусть $n(t')$ — нормаль к $Q = 0$ в точке t' . В качестве последней строки Ψ возьмем функцию $\psi(t, t')$, определенную на $(t_0, t_1) \times (t_\alpha, t_\omega)$ и являющуюся решением уравнения $L^* \psi(t, t') = n(t') \delta(t - t')$, $t' \in (t_\alpha, t_\omega)$. Тогда возмущение управления $\delta u(t)$ приведет к возмущению «условий»

$$h(t') = \int_{t_0}^{t_1} \Psi(t, t') M(t) \delta u(t) \, dt, \quad t' \in (t_\alpha, t_\omega).$$

От t' зависит только последняя компонента h , имеющая смысл проекции смещения $\delta x(t')$ на $n(t')$. Полагая, что положительное направление на n есть направление внутрь разрешенной для траектории области, критерий

* Граничные условия для L получаются вариацией условий для L^0 ; L^{-1} и $(L^*)^{-1}$ существуют.

оптимальности можно получить, потребовав существования вектора

$$\mathbf{r} = \{r_0, r_1, \dots, r_m, r(t')\}; \quad r_0 < 0; \quad r(t') \geq 0; \quad (13)$$

$$(\mathbf{r}, \Psi(t, t') M(t) \delta \mathbf{u}(t)) \leq 0,$$

где

$$(\mathbf{r}, \mathbf{h}) = \sum_{i=0}^m r_i h^i + \int_{t_\alpha}^{t_\omega} r(t') h(t') dt'$$

для всех t и $\delta \mathbf{u}$, совместимых с (1).

Несложные выкладки показывают, что если на некоторой траектории имеет место (13), то она стационарна. Т. е. для всех $\delta \mathbf{u}(t)$, совместимых с (1) и не приводящих (в первом порядке) к нарушению (3) и (6), должно быть $\delta F_0 \geq 0$.

Действительно, пусть $\delta \mathbf{u}(t)$ и есть такая вариация. Т. е. последствия ее таковы, что $\delta F_i = 0$, $i = 1, 2, \dots, m$,

$$\mathbf{n}(t') \delta \mathbf{x}(t') \geq 0, \quad t' \in (t_\alpha, t_\omega).$$

Интегрируя (13), получим

$$0 \leq \int_{t_0}^{t_1} (\mathbf{r}, \Psi(t, t') M \delta \mathbf{u}) dt = (\mathbf{r}, \int_{t_0}^{t_1} \Psi(t, t') M \delta \mathbf{u} dt) =$$

$$= \sum_{i=0}^m r_i \delta F_i + \int_{t_\alpha}^{t_\omega} r(t') dt' \int_{t_0}^{t_1} \phi(t, t') M \delta \mathbf{u} dt = r_0 \delta F_0 + \int_{t_\alpha}^{t_\omega} r(t') \mathbf{n}(t') \delta \mathbf{x}(t') dt'.$$

При этом было использовано то, что по определению $\phi(t, t')$

$$\int_{t_0}^{t_1} \phi(t, t') M(t) \delta \mathbf{u}(t) dt = \int_{t_0}^{t_1} \phi(t, t') L \delta \mathbf{x} dt =$$

$$= \int_{t_0}^{t_1} \delta \mathbf{x}(t) L^* \phi(t, t') dt = \int_{t_0}^{t_1} \delta \mathbf{x}(t) \mathbf{n}(t') \delta(t-t') dt = \delta \mathbf{x}(t') \mathbf{n}(t').$$

Так как по предположению $r(t') \geq 0$, $\delta \mathbf{x} \mathbf{n} \geq 0$ и $r_0 < 0$, то

$$\delta F_0 \geq -\frac{1}{r_0} \int_{t_\alpha}^{t_\omega} r(t') \delta \mathbf{x}(t') \mathbf{n}(t') dt' \geq 0.$$

Желая сблизить (13) с соответствующим результатом из [1], сделаем следующие выкладки: $(\mathbf{r}, \Psi(t, t') M \delta \mathbf{u}) = (\Psi^* \mathbf{r}, M \delta \mathbf{u})$. Но

$$\Psi^* \mathbf{r}(t') = \sum_{i=0}^m r_i \phi_i(t) + \int_{t_\alpha}^{t_\omega} \phi(t, t') r(t') dt'. \quad (14)$$

Применяя L^* к (14), получим, что $\psi(t) = \Psi^* \mathbf{r}(t')$ на (t_α, t_ω) удовлетворяет уравнению

$$L^* \psi = r(t) \mathbf{n}(t), \quad r(t) \geq 0$$

(ради простоты полагаем, что все t_i', t_i'', \dots лежат вне (t_α, t_ω)). Опираясь на вышесказанное, можно предложить следующий способ учета (6): интервал (t_α, t_ω) разбивается на конечное (не очень большое) число частей, причем каждой части соответствует своя строка матрицы Ψ , «ответственная» за выполнение (6) на своем «участке». Подробнее об этом ниже. Аналогичный подход используется и в задаче с минимизацией (5).

§ 2. Расчетная схема

Разобьем (t_0, t_1) на I интервалов точками

$$t_0 = \tau_0 < \tau_1 < \tau_2 < \dots < \tau_I = t_1$$

и в дальнейшем будем рассматривать кусочно-постоянные $u(t)$, т. е.

$$u(t) = u_j, \quad t \in (\tau_{j-1}, \tau_j) \quad (15)$$

(обычно в наших расчетах $I \sim 100$).

Возьмем некоторое управление u вида (15) и решим уравнение (2), не учитывая пока дополнительных условий (3).

Для каждого интервала (τ_{j-1}, τ_j) вычислим k векторов $h_{q,j} \in R_{m+1}$:

$$h_{q,j} = \Psi(\tau_j)M(\tau_j)e_q\Delta t_j \quad (q = 1, 2, \dots, k), \quad \Delta t_j = (\tau_j - \tau_{j-1}), \quad (16)$$

где e_q есть q -й орт в пространстве векторов u .

Для реализации этих вычислений нужно $m+1$ раз решить уравнение (11) (практически вычисление векторов h происходит отдельно по компонентам).

Смысл $h_{q,j}$ следующий. Если на j -м интервале перейти от u_j к $u_j^* = u_j + s e_q$, то следствием этого будет смещение «функционалов»

$$F(u^*) = F(u) + s h_{q,j}.$$

Совокупность векторов $h_{q,j}$ является основной информацией, используя которую можно строить разумный процесс варьирования u . Из дальнейшего будет видно, что к h будет очень частое обращение; поэтому желательно, чтобы все h разместились в оперативной памяти машины. Таким образом, для хорошей реализации предлагаемого метода желательно иметь возможность одновременно иметь в оперативной памяти $[I(m+1)k + 2kI]$ чисел ($2kI$ чисел носят служебный характер).

Для того чтобы следить за условием (1), будем считать, что e_q не орты, но некоторая система линейно-независимых векторов, обладающая следующим свойством: можно выбрать пару чисел $s_{q,j}^- \leq 0 \leq s_{q,j}^+$ таких, что переход от u_j к

$$u_j^* = u_j + \sum_{q=1}^k s_{q,j} e_q, \quad \text{где } s_{q,j}^- \leq s_{q,j} \leq s_{q,j}^+,$$

всегда совместим с (1), и при этом любое $u_j^* \in U$, близкое к u_j , может быть таким образом получено. Такой набор «псевдоортов» может быть своим для каждого интервала. Необходимый пересчет $h_{q,j}$ тривиален.

Пусть теперь для исходного $u(t)$ выполнены условия (3). Естественно поставить следующую задачу A :

найти такие числа $s_{q,j}$, чтобы минимизировать форму

$$H = \sum_{q,j} s_{q,j} h_{q,j}^0 \quad (A_1)$$

при условиях

$$\sum_{q,j} s_{q,j} h_{q,j}^i = 0 \quad (i = 1, 2, \dots, m), \quad (A_2)$$

$$s_{q,j}^- \leq s_{q,j} \leq s_{q,j}^+ \quad (A_3)$$

Это типичная задача линейного программирования. Ниже предлагается итерационное ее решение.

Замечание. Величины чисел $s_{q,j}^{+,-}$ определяются не столько расстояниями до границ U , сколько пределами применимости линейного приближения. Судить о «правильности» величин $s^{+,-}$ можно следующим образом. Решив задачу A и перейдя к управлению

$$u_j^* = u_j + \sum_q s_{q,j} e_{q,j} \quad (j = 1, 2, \dots, I),$$

мы можем предсказать на основе теории возмущений вариации функционалов

$$\delta F \approx \sum_q \sum_j s_{q,j} h_{q,j}. \quad (17)$$

Далее весь описанный выше цикл повторяется, отправляясь уже от u^* . Сравнение (17) с фактическими $\delta F = F(u^*) - F(u)$ покажет, следует ли уменьшить $s^{+,-}$, или можно их увеличить (что, разумеется, желательно).

При решении задачи с функционалом (5) используется следующий прием. После решения (2) находилось несколько точек t', t'', \dots с наибольшими значениями $F_0(t)$ (t', t'', \dots — «контрольные» точки); вместо H получаем формы вида

$$H' = F_0(t') + \sum_{q,j} s_{q,j} (h_{q,j}^0)^2;$$

требовалось минимизировать

$$H = \max(H', H'', \dots). \quad (A_1')$$

При этом, естественно, не контролируется поведение F^0 в других точках, поэтому на следующей итерации «контрольные» точки соответственно переместятся. Опыт показал, что две «контрольные» точки при достаточном числе итераций (30–40) с успехом «следят» за 20–30 интервалами Δt_j . Аналогично реализуется решение задач с условием вида (6).

Случай, когда некоторые из условий (3) имеют вид неравенств, сводится к рассматриваемому стандартным приемом линейного программирования.

Пусть теперь (3) не выполнены по тем или иным причинам. Тогда простой итерационный процесс (см. ниже) позволяет найти числа $s_{q,j}$ такие, что

$$F_i(u) + \sum_{q,j} s_{q,j} h_{q,j}^i \approx 0 \quad (i = 1, 2, \dots, m).$$

После этого решаем задачу A , очевидным образом пересчитав $s_{q,j}^-, s_{q,j}^+$. (В начале расчета, когда нарушения (3) велики, на выполнение (3) приходится затратить несколько итераций.)

Задаче A можно придать следующий геометрический смысл: рассмотрим выпуклый многогранник P в R_{m+1} («область достижимости») — множество точек

$$y = \sum_n s_n h_n, \quad s_n^- \leq s_n \leq s_n^+$$

(двойной индекс заменен одинарным) и его границу Γ .

Задача состоит в нахождении пересечения Γ с лучом $(-1, 0, 0, \dots, 0)$.

Задача с H вида A_1' сводится к решению задачи A , в которой ищется пересечение Γ с лучом $(-1, 1, \dots, 1, 0, \dots, 0)$ и к совокупности векторов h_n добавлено

l векторов вида $(1, 0, \dots, 0, \dots, 0), (0, 1, 0, \dots, 0), \dots$ с границами $-\infty \leq s \leq 0$ (l — число форм H', H'', \dots).

В тех случаях, когда в задаче имеются некоторые варьируемые параметры (например, величина t_1 , часть граничных значений x , которые удобно задавать для возможно более простой реализации L_0^{-1} , и т. д.), их вариации порождают некоторые векторы h , включающиеся в общую схему.

Рассматривая вопрос о точности приближенного решения, естественно выделить следующие источники ошибок.

1. Ошибка, связанная с переходом к управлению вида (15); она мала уже при не очень больших I .

2. Ошибка аппроксимации. Так как векторы h вычислены в (16) грубо, с точностью до $(\Delta t)^2$, то для управления, оптимального в классе (15), «конус достижимости», построенный по h , будет содержать направление $(-1, 0, 0, \dots, 0)$ и вычислительный процесс «уведет» решение от управления оптимального по F_0 к оптимальному в смысле численного аналога «принципа максимума». Естественно, ошибки такого рода особенно сильны в тех точках, где происходит резкое изменение величин (например, рвется $u(t)$). При расчете с равномерным шагом Δt такие участки обычно выявляются. Затем можно уточнить расчет, взяв на этих участках более подробную сетку. С этой ошибкой связано, по-видимому, «размазывание» разрывов в $u(t)$. Все это хорошо видно на примере решения задачи о подъеме ракеты.

3. Ошибка локальности. Чтобы не получить локальный экстремум, необходимо повторить расчет, отправляясь от других $u(t)$.

4. Ошибка поиска. Стабилизация значения F_0 в ходе итераций обычно создает впечатление, что получено u , достаточно близкое к оптимальному. Но это может быть следствием неэффективности метода. Поэтому важно уметь анализировать полученное решение. Совокупность векторов h содержит необходимую для этого информацию. Примеры такого анализа приведены ниже.

§ 3. Итерационное решение задачи A

Излагаемый ниже метод есть, по существу, поиск «опорной плоскости». В этом смысле нижеследующее близко к методу в [3], однако многогранники не обладают широко используемым в [3] свойством «строгой выпуклости».

Предварительно решим задачу B:

дана область P точек

$$x = x_0 + \sum s_n h_n, \quad s_n^- \leq s_n \leq s_n^+$$

найти s_n , минимизирующее $\|x\|$ в P .

Возьмем какой-либо вектор h_n и переходом из x_0 в $x_0 + s h_n$ минимизируем $\|x\|$.

Пусть $\tilde{s} = -(h, x_0) / \|h\|^2$; тогда

$$s_n = \begin{cases} \min(s_n^+, \tilde{s}) & \text{при } \tilde{s} \geq 0, \\ \max(s_n^-, \tilde{s}) & \text{при } \tilde{s} < 0. \end{cases}$$

Выберем то n , для которого минимальна $\|x_0 + s_n h_n\|$, и переместимся в точку $x_1 = x_0 + s_n h_n$. В дальнейшем это n может быть снова