

С. УЛЬМ

## О РЕШЕНИИ КРАЕВЫХ ЗАДАЧ, ВЫТЕКАЮЩИХ ИЗ ПРИНЦИПА МАКСИМУМА

При решении задач оптимального управления принципом максимума Понтрягина главная трудность состоит в нахождении решений двухточечных краевых задач. Одним из возможных путей решения последних является метод, названный методом организованного подбора недостающих начальных условий (см. [1]). Основу этого метода составляет сведение краевой задачи к повторному решению задач Коши с последующим подбором недостающих условий по какому-либо алгоритму решения нелинейных алгебраических и трансцендентных уравнений. При этом для решения последних используется главным образом обобщенный метод Ньютона (см., напр., [1, 2]). Но в последнее время [3-5] для решения систем нелинейных алгебраических и трансцендентных уравнений разработан ряд методов интерполяционного типа, которые, по нашему мнению, можно часто с большим успехом применять при решении краевых задач принципа максимума. Целью настоящей статьи и является описание этих методов и их сравнение с методом Ньютона. Отдельно рассматривается приближенное решение линейных задач по быстродействию.

1. Применение вышеупомянутых методов мы демонстрируем на основании следующей задачи:  
минимизировать

$$I = \sum_{i=1}^n c_i x_i(T) = (c, x(T)) \quad (1)$$

при условиях

$$\begin{cases} \dot{x} = f(x, u, t) \\ x(0) = x^0, \end{cases} \quad (2)$$

где  $x = (x_1, x_2, \dots, x_n)$ ;  $u = (u_1, u_2, \dots, u_m)$ ;  $f = (f_1, f_2, \dots, f_n)$ ;  $c = (c_1, c_2, \dots, c_n)$ ;  $u \in U$  ( $U$  — некоторое множество в  $m$ -мерном пространстве).

Конечное время  $T$  считаем фиксированным. По принципу максимума [6] оптимальное управление  $u$  максимизирует гамильтониан

$$H(x, p, u, t) = \sum_{i=1}^n p_i f_i(x, u, t), \quad (3)$$

причем импульсы  $p_i$  удовлетворяют системе сопряженных уравнений

$$\begin{cases} \dot{p}_i = - \sum_{k=1}^n p_k (\partial f_k(x, u, t) / \partial x_i) \\ p_i(T) = -c_i \quad (i = 1, 2, \dots, n). \end{cases} \quad (4)$$

Из условия максимума гамильтониана можно в принципе выразить  $u = u(x, p, t)$ .

Обозначив  $y = (x, p) = (x_1, \dots, x_n, p_1, \dots, p_n) = (y_1, \dots, y_n, y_{n+1}, \dots, y_{2n})$ , получим на основании (2) и (4) следующую краевую задачу:

$$\begin{cases} \dot{y}_i = g_i(y_1, \dots, y_{2n}, t) & (i = 1, \dots, 2n) \end{cases} \quad (5)$$

$$\begin{cases} y_i(0) = x_i^0 & (i = 1, \dots, n) \end{cases} \quad (6)$$

$$\begin{cases} y_{n+i}(T) = -c_i & (i = 1, \dots, n). \end{cases} \quad (7)$$

Величины  $y_{n+i}(0)$  ( $i = 1, \dots, n$ ) мы не знаем, (в противном случае мы могли бы решить для системы (5) задачу Коши). Но каждому выбору значений  $y_{n+i}(0)$  соответствуют определенные значения  $y_{n+i}(T)$ . Нашей задачей является нахождение таких значений  $y_{n+i}(0)$ , при которых условия  $y_{n+i}(T) = -c_i$  были бы удовлетворены. Обозначим

$$\psi_i(y_{n+1}(0), \dots, y_{2n}(0)) = y_{n+i}(T) + c_i \quad (i = 1, \dots, n). \quad (8)$$

Проблема решения краевой задачи сводится к нахождению таких значений  $y_{n+i}(0)$ , при которых  $\psi_i = 0$ , т. е. мы должны решить следующую систему нелинейных уравнений:

$$\psi_i(y_{n+1}(0), \dots, y_{2n}(0)) = 0 \quad (i = 1, \dots, n).$$

Обозначив  $y_{n+i}(0) = \lambda_i$ , получим систему

$$\psi_i(\lambda_1, \lambda_2, \dots, \lambda_n) = 0 \quad (i = 1, \dots, n). \quad (9)$$

Чтобы найти значения функций  $\psi_i = y_{n+i}(T) + c_i$  ( $i = 1, \dots, n$ ) в точке  $(\lambda_1, \dots, \lambda_n)$ , решим для системы (5) задачу Коши (например, методом Рунге-Кутты) при начальных условиях

$$y_i(0) = x_i^0 \quad (i = 1, \dots, n); \quad y_{n+i}(0) = \lambda_i \quad (i = 1, \dots, n).$$

2. Применение метода Ньютона. Пусть выбрано некоторое приближение  $\lambda^{(0)} = (\lambda_1^{(0)}, \dots, \lambda_n^{(0)})$ . Если приближение  $\lambda^{(k)} = (\lambda_1^{(k)}, \dots, \lambda_n^{(k)})$  найдено, то по методу Ньютона так наз. вектор поправки  $\Delta\lambda^{(k)} = \lambda^{(k+1)} - \lambda^{(k)} = (\Delta\lambda_1^{(k)}, \dots, \Delta\lambda_n^{(k)})$  определяется из следующей системы уравнений:

$$J^{(k)} \Delta\lambda^{(k)} = -\psi^{(k)} \quad (k = 0, 1, \dots), \quad (10)$$

где

$$J^{(k)} = (\partial\psi_i / \partial\lambda_j) \Big|_{\lambda = \lambda^{(k)}} \quad \begin{matrix} i=1, \dots, n \\ j=1, \dots, n \end{matrix}$$

и

$$\psi^{(k)} = (\psi_1(\lambda_1^{(k)}, \dots, \lambda_n^{(k)}), \dots, \psi_n(\lambda_1^{(k)}, \dots, \lambda_n^{(k)})).$$

Новое приближение  $\lambda^{(k+1)}$  найдется из соотношений

$$\lambda_i^{(k+1)} = \lambda_i^{(k)} + \Delta\lambda_i^{(k)} \quad (i = 1, \dots, n).$$

Итак, метод Ньютона требует, кроме вычисления значений  $\psi_i$ , также вычисления частных производных  $\partial\psi_i / \partial\lambda_j$ . Для нахождения последних

в [1, 2] действуют следующим образом. Дифференцируя систему (5) по  $y_{n+j}(0)$ , получим

$$\dot{\partial y_i} / \partial y_{n+j}(0) = \partial [g_i(y_1, \dots, y_{2n}, t)] / \partial y_{n+j}(0)$$

или

$$d(\partial y_i / \partial y_{n+j}(0)) / dt = \sum_{s=1}^{2n} (\partial g_i / \partial y_s) (\partial y_s / \partial y_{n+j}(0)) \quad (11)$$

$$(i = 1, \dots, 2n; j = 1, \dots, n).$$

Так как  $\partial \psi_i / \partial \lambda_j = \partial y_{n+i}(T) / \partial y_{n+j}(0)$ , из (11) можно определить нужные нам частные производные. В качестве начальных условий для системы (11) получим:

$$\partial y_i / \partial y_{n+j}(0) = \begin{cases} 0, & \text{если } i \neq n+j \\ 1, & \text{если } i = n+j. \end{cases} \quad (12)$$

В зависимости от размерности задачи и от памяти машины здесь имеются три возможности.

**А.** Интегрируем систему (11) при начальных условиях (12) совместно с системой

$$\begin{cases} \dot{y}_i = g_i(y_1, \dots, y_{2n}, t) \\ y_i(0) = x_i^0 & (i = 1, \dots, n) \\ y_{n+i}(0) = \lambda_i^{(k)} & (i = 1, \dots, n). \end{cases} \quad (13)$$

Итак, для выполнения каждого итерационного шага требуется итерирование  $2n(n+1)$ -мерной системы дифференциальных уравнений.

**Б.** Решим численно систему (13), сохраняя в памяти значения  $y_1, y_2, \dots, y_{2n}$ . Разбивая систему (11) на подсистемы ( $j = 1, \dots, n$ ), интегрируем каждую подсистему отдельно. Всего требуется на каждом итерационном шаге интегрировать  $n+1$  раз  $2n$ -мерные системы.

**В.** Чтобы избежать чрезмерной нагрузки памяти, интегрируем систему (13) повторно совместно с каждой подсистемой (11) ( $j = 1, \dots, n$ ). На каждом шаге требуется  $n$  раз интегрировать  $4n$ -мерные системы.

Рассмотрим еще одну методику для вычисления частных производных  $\partial \psi_i / \partial \lambda_j$ . Исходя из соотношений

$$\psi_i(y_{n+1}(0), \dots, y_{2n}(0)) = y_{n+i}(T) + c_i,$$

получим

$$\delta y_{n+i}(T) = \sum_{j=1}^n (\partial \psi_i / \partial y_{n+j}(0)) \delta y_{n+j}(0). \quad (14)$$

С другой стороны, варьируя начальные условия  $y_{n+j}(0)$ , можем  $\delta y_i(t)$  вычислить из линеаризованной системы

$$\dot{\delta y} = (\partial g / \partial y) \delta y, \quad (15)$$

где

$$\partial g / \partial y = (\partial g_i / \partial y_k)_{\substack{i=1, \dots, 2n \\ k=1, \dots, 2n}}; \quad \delta y = (\delta y_1, \dots, \delta y_{2n}).$$

Рассмотрим сопряженную к системе (15) систему

$$\dot{z} = -(\partial g / \partial y)^* z \quad (16)$$

при конечных условиях

$$z(T) = \partial y_{n+i}(T) / \partial y(T), \quad (17)$$

т. е.

$$z(T) = (0, \dots, 0, \underbrace{1}_{n+i}, 0, \dots, 0).$$

Так как справедливо соотношение

$$d(z, \delta y) / dt = 0,$$

то получим

$$(z, \delta y)_{t=T} = (z, \delta y)_{t=0}$$

или

$$\delta y_{n+i}(T) = \sum_{j=1}^n z_{n+j}(0) \delta y_{n+i}(0). \quad (18)$$

Сравнивая (14) и (18), получим

$$z_{n+j}(0) = \partial \psi_i / \partial y_{n+j}(0) = \partial \psi_i / \partial \lambda_j.$$

Для вычисления значений  $\psi_i^{(k)}$  и  $\partial \psi_i / \partial \lambda_j |_{\lambda=\lambda^{(k)}}$  интегрируем численно систему (13), сохраняя в памяти машины значения  $y_1, y_2, \dots, y_{2n}$ . Интегрируем систему (16) при конечных условиях (17)  $n$  раз ( $i = 1, \dots, n$ ) справа налево. Итак, для выполнения каждого шага итерационного метода требуется  $n+1$  раз интегрировать  $2n$ -мерные системы. Если не сохранять в памяти значения  $y_1(t), \dots, y_{2n}(t)$  (сохраняя только значения  $y_1(T), \dots, y_{2n}(T)$ ), то придется один раз интегрировать систему (13) слева направо и  $n$  раз систему (13) совместно с системой (16) справа налево (т. е.  $4n$ -мерную систему).

3. Чтобы избежать в вычислении частных производных, при применении метода Ньютона используются формулы численного дифференцирования, т. е. примем

$$\begin{aligned} \partial \psi_i / \partial \lambda_j |_{\lambda=\lambda^{(k)}} \approx & [\lambda_j^{(k)} - \lambda_j^{(k)}]^{-1} [\psi_i(\lambda_1^{(k)}, \dots, \lambda_{j-1}^{(k)}, \lambda_j^{(k)}, \lambda_{j+1}^{(k)}, \dots, \lambda_n^{(k)}) - \\ & - \psi_i(\lambda_1^{(k)}, \dots, \lambda_j^{(k)}, \dots, \lambda_n^{(k)})]. \end{aligned} \quad (19)$$

На каждом шаге итерации требуется  $n+1$  раз интегрировать систему (5) (для вычисления величин  $\psi_i(\lambda_1^{(k)}, \dots, \lambda_n^{(k)})$ ;  $\psi_i(\lambda_1^{(k)}, \dots, \lambda_{j-1}^{(k)}, \lambda_j^{(k)}, \lambda_{j+1}^{(k)}, \dots, \lambda_n^{(k)})$  ( $j = 1, \dots, n$ )). Отметим, что при этом свойства сходимости метода Ньютона могут существенно ухудшиться, так как не имеется критерия для оптимального выбора  $\lambda_j^{(k)}$ . Поэтому более целесообразно использовать так наз. интерполяционные методы, выработанные в последнее время, — метод хорд и метод Стеффенсена. Кроме этих методов, ниже рассматривается некоторый разностный аналог

метода градиентов, имеющий более низкий порядок сходимости, чем методы хорд и Стеффенсена, но не требующий решения системы линейных алгебраических уравнений.

Метод хорд [3]. Выберем начальные приближения  $\lambda^{(0)} = (\lambda_1^{(0)}, \dots, \lambda_n^{(0)})$ ;  $\lambda^{(1)} = (\lambda_1^{(1)}, \dots, \lambda_n^{(1)})$ . Если приближение  $\lambda^{(k)} = (\lambda_1^{(k)}, \dots, \lambda_n^{(k)})$  построено, то вектор поправки  $\Delta\lambda^{(k)} = \lambda^{(k+1)} - \lambda^{(k)} = (\Delta\lambda_1^{(k)}, \dots, \Delta\lambda_n^{(k)})$  определяется из следующей системы линейных уравнений:

$$\Gamma^{(k)} \Delta\lambda^{(k)} = -\Psi^{(k)} \quad (k = 1, 2, \dots), \quad (20)$$

где

$$\Gamma^{(k)} = (a_{ij}^{(k)})_{\substack{i=1, \dots, n \\ j=1, \dots, n}} \quad (21)$$

или

$$\Gamma^{(k)} = (b_{ij}^{(k)})_{\substack{i=1, \dots, n \\ j=1, \dots, n}}; \quad (22)$$

$$a_{ij}^{(k)} = [\lambda_j^{(k)} - \lambda_j^{(k-1)}]^{-1} [\psi_i(\lambda_1^{(k-1)}, \dots, \lambda_{j-1}^{(k-1)}, \lambda_j^{(k)}, \dots, \lambda_n^{(k)}) - \psi_i(\lambda_1^{(k-1)}, \dots, \lambda_j^{(k-1)}, \lambda_{j+1}^{(k)}, \dots, \lambda_n^{(k)})];$$

$$b_{ij}^{(k)} = [\lambda_j^{(k-1)} - \lambda_j^{(k)}]^{-1} [\psi_i(\lambda_1^{(k)}, \dots, \lambda_{j-1}^{(k)}, \lambda_j^{(k-1)}, \dots, \lambda_n^{(k-1)}) - \psi_i(\lambda_1^{(k)}, \dots, \lambda_j^{(k)}, \lambda_{j+1}^{(k-1)}, \dots, \lambda_n^{(k-1)})];$$

$$\Psi^{(k)} = (\psi_1(\lambda_1^{(k)}, \dots, \lambda_n^{(k)}), \dots, \psi_n(\lambda_1^{(k)}, \dots, \lambda_n^{(k)})).$$

Нсвое приближение  $\lambda^{(k+1)} = (\lambda_1^{(k+1)}, \dots, \lambda_n^{(k+1)})$  найдется из соотношений

$$\lambda_i^{(k+1)} = \lambda_i^{(k)} + \Delta\lambda_i^{(k)} \quad (i = 1, 2, \dots, n).$$

Как видно из формул (20), (21) и (22), для нахождения значений функций  $\psi_i$  на каждом шаге итерации требуется  $n$  раз интегрировать задачу Коши для уравнения (5). Исключением является первый шаг, при котором требуется  $n+1$  интегрирований. Порядком сходимости метода при обоих вариантах является  $\frac{1}{2}(1 + \sqrt{5}) \approx 1,618$ .

Метод Стеффенсена (ср. [4]). Выберем начальное приближение  $\lambda^{(0)}$ . Если приближение  $\lambda^{(k)}$  построено, то вектор поправки  $\Delta\lambda^{(k)} = \lambda^{(k+1)} - \lambda^{(k)}$  определяется из следующей системы линейных уравнений:

$$\Lambda^{(k)} \Delta\lambda^{(k)} = -\Psi^{(k)} \quad (k = 0, 1, \dots), \quad (23)$$

где для выбора  $\Lambda^{(k)}$  имеются опять две возможности:

$$\Lambda^{(k)} = (c_{ij}^{(k)})_{\substack{i=1, \dots, n \\ j=1, \dots, n}} \quad (24)$$

или

$$\Lambda^{(k)} = (d_{ij}^{(k)})_{\substack{i=1, \dots, n \\ j=1, \dots, n}}; \quad (25)$$

$$c_{ij}^{(k)} = (\gamma_j^{(k)} \psi_j^{(k)})^{-1} [\psi_i(\lambda_1^{(k)} - \gamma_1^{(k)} \psi_1^{(k)}, \dots, \lambda_{j-1}^{(k)} - \gamma_{j-1}^{(k)} \psi_{j-1}^{(k)}, \lambda_j^{(k)}, \dots, \lambda_n^{(k)}) - \psi_i(\lambda_1^{(k)} - \gamma_1^{(k)} \psi_1^{(k)}, \dots, \lambda_j^{(k)} - \gamma_j^{(k)} \psi_j^{(k)}, \lambda_{j+1}^{(k)}, \dots, \lambda_n^{(k)})];$$

$$d_{ij}^{(k)} = (-\gamma_j^{(k)} \psi_j^{(k)})^{-1} [\psi_i(\lambda_1^{(k)}, \dots, \lambda_{j-1}^{(k)}, \lambda_j^{(k)} - \gamma_j^{(k)} \psi_j^{(k)}, \dots, \lambda_n^{(k)} - \gamma_n^{(k)} \psi_n^{(k)}) - \psi_i(\lambda_1^{(k)}, \dots, \lambda_j^{(k)}, \lambda_{j+1}^{(k)} - \gamma_{j+1}^{(k)} \psi_{j+1}^{(k)}, \dots, \lambda_n^{(k)} - \gamma_n^{(k)} \psi_n^{(k)})],$$

$\gamma_j^{(k)} \neq 0$  ( $j = 1, \dots, n$ ) — довольно малые по абсолютной величине числа.

Как видно из формул (23), (24) и (25), для нахождения значений функций  $\psi_i$  на каждом шаге итерации требуется  $n+1$  раз интегрировать задачу Коши для уравнения (5). Порядком сходимости метода при обоих вариантах является 2, т. е. как и у метода Ньютона (см. приложение).

Разностный аналог метода градиентов [5]. Пусть выбраны два начальных приближения:  $\lambda^{(0)}$  и  $\lambda^{(1)}$ . Если приближение  $\lambda^{(k)}$  построено, то новое приближение  $\lambda^{(k+1)}$  найдется из соотношения

$$\lambda^{(k+1)} = \lambda^{(k)} - \varepsilon_k \Gamma^{(k)*} \psi^{(k)} \quad (k = 1, 2, \dots), \quad (26)$$

где  $\Gamma^{(k)*}$  — матрица, сопряженная к матрице  $\Gamma^{(k)}$  (см. формулу (20));  $\varepsilon_k$  — положительные числа;  $\inf_k \varepsilon_k > 0$ . Используя (21) и (22), получим два варианта метода (26):

$$\lambda_i^{(k+1)} = \lambda_i^{(k)} - \varepsilon_k (\lambda_i^{(k)} - \lambda_i^{(k-1)})^{-1} \sum_{j=1}^n \psi_j(\lambda_1^{(k)}, \dots, \lambda_n^{(k)}) \times \quad (27)$$

$$\times [\psi_j(\lambda_1^{(k-1)}, \dots, \lambda_{i-1}^{(k-1)}, \lambda_i^{(k)}, \dots, \lambda_n^{(k)}) - \psi_j(\lambda_1^{(k-1)}, \dots, \lambda_i^{(k-1)}, \lambda_{i+1}^{(k)}, \dots, \lambda_n^{(k)})]$$

или

$$\lambda_i^{(k+1)} = \lambda_i^{(k)} - \varepsilon_k (\lambda_i^{(k-1)} - \lambda_i^{(k)})^{-1} \sum_{j=1}^n \psi_j(\lambda_1^{(k)}, \dots, \lambda_n^{(k)}) \times \quad (28)$$

$$\times [\psi_j(\lambda_1^{(k)}, \dots, \lambda_{i-1}^{(k)}, \lambda_i^{(k-1)}, \dots, \lambda_n^{(k-1)}) - \psi_j(\lambda_1^{(k)}, \dots, \lambda_i^{(k)}, \lambda_{i+1}^{(k-1)}, \dots, \lambda_n^{(k-1)})]$$

$(i = 1, \dots, n; k = 1, 2, \dots).$

Как и у метода хорд, для нахождения значений  $\psi_i$  требуется на каждом шаге  $n$  интегрирований уравнения (5) (при первом шаге  $n+1$ ). Метод сходится со скоростью геометрической прогрессии и его можно рекомендовать только для использования в начале вычислительного процесса для уточнения начальных приближений. Для выбора  $\varepsilon_k$  имеются некоторые соображения в [5].

4. Рассмотренные выше методы можно использовать и для решения более сложных задач, когда применим принцип максимума (учитывая условия трансверсальности и т. д.). Мы не будем останавливаться на таких применениях. Рассмотрим только случай, когда в задаче (1) — (2) конечное время  $T$  не фиксировано. Тогда по принципу максимума [6] надо добавить еще условие

$$\sum_{i=1}^n p_i(T) \dot{f}_i(x(T), u(T), T) = 0. \quad (29)$$

После исключения из условия максимума гамильтониана  $u = u(x, p, t)$  можно (29) представить в виде

$$\chi(y_1(T), \dots, y_{2n}(T)) = 0. \quad (30)$$

На  $(k+1)$ -м шаге итерации  $T = T^{(k)}$  определяется как минимальное время, при котором

$$\chi(y_1^{(k)}(T), \dots, y_n^{(k)}(T), \lambda_1^{(k)}, \dots, \lambda_n^{(k)}) = 0. \quad (31)$$

Отметим еще одну возможность для решения задачи (1)–(2) ( $T$  фиксировано). Так как различным выборам значений  $y_i(T)$  ( $i = 1, \dots, n$ ) соответствуют различные невязки в выполнении начальных условий, можно написать

$$\varphi_i(y_1(T), \dots, y_n(T)) = y_i(0) - x_i^0. \quad (32)$$

Обозначив  $y_i(T) = \mu_i$ , приходится решать систему уравнений

$$\varphi_i(\mu_1, \mu_2, \dots, \mu_n) = 0 \quad (i = 1, 2, \dots, n). \quad (33)$$

Для вычисления значений функций  $\varphi_i = y_i(0) - x_i^0$  в точке  $(\mu_1, \dots, \mu_n)$  приходится интегрировать систему

$$\begin{cases} \dot{y}_i = g_i(y_1, \dots, y_{2n}, t) & (i = 1, \dots, 2n) \\ y_i(T) = \mu_i & (i = 1, \dots, n) \\ y_{n+i}(T) = -c_i & (i = 1, \dots, n) \end{cases} \quad (34)$$

справа налево. Для решения системы (33) можно применить методы, рассмотренные выше. Отметим, что этот путь часто оказывается более удобным, так как для выбора начальных значений  $\mu_i^{(0)}$  ( $i = 1, \dots, n$ ) имеются физические соображения (см. [7]).

5. Рассмотрим наконец более подробно линейную задачу по быстродействию. Используя вышеизложенные идеи, приведем эту задачу к решению нелинейной системы, где неизвестными являются время процесса и начальные значения импульсов.

Пусть даны уравнения движения динамической системы

$$\dot{x}(t) = A(t)x(t) + B(t)u(t), \quad (35)$$

где

$$x(t) = (x_1(t), \dots, x_n(t))$$

$$u(t) = (u_1(t), \dots, u_m(t))$$

$$A(t) = (a_{ij}(t))_{\substack{i=1, \dots, n \\ j=1, \dots, n}}; \quad B(t) = (b_{ik}(t))_{\substack{i=1, \dots, n \\ k=1, \dots, m}}.$$

Требуется с минимальным временем  $T$  ввести систему из положения  $x(0) = x^0$  в положение  $x(T) = x^1$  ( $x^0, x^1$  — фиксированные точки в фазовом пространстве). Предполагается, что

$$\max_{t \in [0, T]} (|u_1(t)|, \dots, |u_m(t)|) \leq L. \quad (36)$$

Обозначим через  $Y(t)$  фундаментальную матрицу системы  $\dot{x}(t) = A(t)x(t)$ , т. е.  $Y(t)$  удовлетворяет следующему матричному дифференциальному уравнению:

$$\begin{cases} \dot{Y}(t) = A(t)Y(t) \\ Y(0) = E, \end{cases} \quad (37)$$

где  $E$  является  $n$ -мерной единичной матрицей.

Система для определения импульсов выражается в виде

$$\dot{p}(t) = -A^*(t)p(t). \quad (38)$$

Обозначим  $p(0) = \lambda$ . Как известно, тогда

$$p(t) = (Y^{-1}(t))^* \lambda = Y^*(-t)\lambda \quad (39)$$

и гамильтониан

$$H = (p(t), A(t)x(t) + B(t)u(t)). \quad (40)$$

Из условия максимума гамильтониана

$$\begin{aligned} u_i &= l \operatorname{sign}(B^* p)_i = l \operatorname{sign}(B^* Y^*(-t)\lambda)_i = \\ &= l \operatorname{sign}((Y(-t)B)^* \lambda)_i = l \operatorname{sign}(G^* \lambda)_i, \end{aligned} \quad (41)$$

где

$$G(t) = Y(-t)B(t).$$

Вектор фазовых координат выражается в виде

$$\begin{aligned} x(t) &= Y(t)x^0 + Y(t) \int_0^t Y(-\tau)B(\tau)u(\tau) d\tau = \\ &= Y(t)x^0 + Y(t) \int_0^t G(\tau) \operatorname{sign}(G^*(\tau)\lambda) d\tau. \end{aligned} \quad (42)$$

Нам нужно найти такие значения  $\lambda$ , чтобы  $x(T) - x^1 = 0$ , т. е. для определения  $\lambda$  получим систему уравнений:

$$Y(T)x^0 + Y(T) \int_0^T G(t) \operatorname{sign}(G^*(t)\lambda) dt - x^1 = 0. \quad (43)$$

Для определения времени  $T$  получим уравнение (см. [6]):

$$\max_u H(T) = 1$$

или

$$\begin{aligned} (Y^*(-T)\lambda, A(T)[Y(T)x^0 + Y(T) \int_0^T G(t) \operatorname{sign}(G^*(t)\lambda) dt] + \\ + l B(T) \operatorname{sign}(G^*(T)\lambda) = 1. \end{aligned} \quad (44)$$

Учитывая (43), можно (44) выписать в виде

$$(Y^*(-T)\lambda, A(T)x^1 + lB(T)\text{sign}(G^*(T)\lambda)) = 1$$

или

$$(\lambda, Y(T)A(T)x^1) + l(G^*(T)\lambda, \text{sign}(G^*(T)\lambda)) = 1.$$

Справедливо равенство \*

$$(G^*(T)\lambda, \text{sign}(G^*(T)\lambda)) = \|G^*(T)\lambda\|.$$

Итак, для определения неизвестных  $\lambda_1, \dots, \lambda_n, T$  получим следующую систему уравнений:

$$\begin{cases} Y(T)x^0 + Y(T) \int_0^T G(t)\text{sign}(G^*(t)\lambda)dt - x^1 = 0, \\ (\lambda, Y(T)A(T)x^1) + l\|G^*(T)\lambda\| - 1 = 0. \end{cases} \quad (45)$$

Систему (13) можно приближенно решить, используя, например, метод хорд или метод Стеффенсена.

**Приложение.** Отметим, что предложенные в пункте 3 алгоритмы Стеффенсена являются в некотором смысле более общими, чем алгоритмы статьи [4]. Именно для нахождения решения операторного уравнения  $P(x) = 0$  рассматривается алгоритм

$$x^{(k+1)} = x^{(k)} - [P(x^{(k)}, \tilde{x}^{(k)})]^{-1} P(x^{(k)}), \quad (46)$$

где

$$\tilde{x}^{(k)} = x^{(k)} - \gamma^{(k)} P(x^{(k)}); \gamma^{(k)} \neq 0; \quad (47)$$

$P(x', x'')$  — разделенная разность для оператора  $P(x)$ .

Покажем, что при естественных условиях метод (46) имеет квадратную скорость сходимости. Пусть  $x^*$  является решением уравнения  $P(x) = 0$ . Тогда

$$\begin{aligned} x^* - x^{(k+1)} &= x^* - x^{(k)} + [P(x^{(k)}, \tilde{x}^{(k)})]^{-1} P(x^{(k)}) = \\ &= [P(x^{(k)}, \tilde{x}^{(k)})]^{-1} [P(x^{(k)}, \tilde{x}^{(k)}) (x^* - x^{(k)}) + P(x^{(k)}) - P(x^*)] = \\ &= - [P(x^{(k)}, \tilde{x}^{(k)})]^{-1} [P(x^*, x^{(k)}) - P(x^{(k)}, \tilde{x}^{(k)})] (x^* - x^{(k)}). \end{aligned}$$

Пусть в некоторой окрестности элемента  $x^*$  выполнены оценки

$$\| [P(x', x'')]^{-1} \| \leq B; \quad \| P(x', x'') \| \leq M$$

$$\| P(x', x'') - P(x'', x''') \| \leq K_1 \| x' - x'' \| + K_2 \| x'' - x''' \|.$$

Тогда получим

$$\| x^* - x^{(k+1)} \| \leq B (K_1 \| x^* - x^{(k)} \| + K_2 \| x^{(k)} - \tilde{x}^{(k)} \|) \| x^* - x^{(k)} \|.$$

\* При этом норма вектора  $v = (v_1, \dots, v_m)$  определена равенством

$$\| v \| = |v_1| + \dots + |v_m|.$$

Так как

$$\begin{aligned} x^{(k)} - \tilde{x}^{(k)} &= \gamma^{(k)} P(x^{(k)}) = \gamma^{(k)} (P(x^{(k)}) - P(x^*)) = \\ &= -\gamma^{(k)} P(x^*, x^{(k)}) (x^* - x^{(k)}), \end{aligned}$$

то

$$\|x^{(k)} - \tilde{x}^{(k)}\| \leq |\gamma^{(k)}| M \|x^* - x^{(k)}\|$$

и

$$\|x^* - x^{(k+1)}\| \leq B(K_1 + K_2 |\gamma^{(k)}| M) \|x^* - x^{(k)}\|^2,$$

что и требовалось доказать.

Отметим, что из соображений общности  $\gamma^{(k)}$  рассматриваются в (24) и (25) как векторы.

#### ЛИТЕРАТУРА

1. Гродзовский Г. Л., Иванов Ю. Н., Токарев В. В., Инж. ж., 4, вып. 2 (1964).
2. Levine M. D., Automatica, 3, 203—217 (1966).
3. Schmidt J. W., Z. angew. Math. u Mech., 43, № 1—2, 1—8 (1963); 43, № 3, 97—110 (1963).
4. Ульм С. Ю., Ж. вычисл. матем. и матем. физ., 4, № 6, 1093—1097 (1964).
5. Ульм С., Изв. АН Эст. ССР. Сер. физ.-матем. и техн. наук, 12, № 2, 132—140 (1963).
6. Розоноэр Л. И., Автоматика и телемеханика, 20, № 10, 1320—1334 (1959); 20, № 11, 1441—1458 (1959).
7. Кларк С. Н., Frost P. A., IEEE Trans. Automatic Control, AC10, No. 2, 189—193 (1965).

Институт кибернетики  
Академии наук Эстонской ССР

Поступила в редакцию  
23/V 1966

S. ULM

#### MAKSIMUMPRINTSIIBIST TULENEVATE RAJAÜLESANNETE LAHENDAMISEST

Vaadeldakse Newtoni meetodi ja nn. interpolatsioonimeetodite rakendamist Pontrjagini maksimumprintsibiist tulenevate rajaülesannete lahendamiseks (puuduvate algtingimuste leidmise meetod). Lineaarne aeg-optimaalne ülesanne taandatakse mittelineaarse võrrandisüsteemi lahendamisele, kus tundmatuteks on protsessi lõppaeg ja impulsside algväärtused.

S. ULM

#### ON SOLVING BOUNDARY VALUE PROBLEMS OF THE MAXIMUM PRINCIPLE

In this article some algorithms (Newton's method, the secant method and Steffensen's method) for the solving of boundary value problems of the maximum principle are considered. For finding a time-optimal program control, a system of nonlinear equations is derived.