Computational Mathematics and Mathematical Physics, Vol. 40, No. 6, 2000, pp. 799–819.
Translated from Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki, Vol. 40, No. 6, 2000, pp. 838–859.

Numerical Methods for Open-Loop and Closed-Loop Optimization of Linear Control Systems

N. V. Balashevich*, R. Gabasov, and F. M. Kirillova***

** Institute of Mathematics, Belarussian Academy of Sciences, ul. Surganova 11, Minsk, 220072 Belarus*

*** Belarussian State University, pr. Skoriny 4, Minsk, 220080 Belarus*

Received June 1, 1999

Abstract—The canonical optimal control problem is considered for a linear time-dependent system in the class of discrete-time feasible controls. Based on linear programming, two primal and two dual methods for constructing optimal open-loop controls are described. A method for synthesizing optimal feedback controls is presented. The results are illustrated by analyzing a fifth-order problem. The efficiency of the methods is estimated.

1. INTRODUCTION

Linear optimal control problems have been analyzed in the most detail in the literature on optimal processes [1, 2]. At the same time, despite numerous suggestions, no sufficiently efficient numerical solution methods completely adapted to these problems is available to this day. This is particularly true with regard to closed-loop solutions. Further analysis of linear dynamic control problems should not be surprising, because serious efforts are still being made to develop new efficient methods for solving even considerably simpler static problems in linear programming (LP) [3].

The idea of using LP methods for solving linear (and nonlinear) optimal control problems is quite natural. These methods have been implemented in various forms, for example, in [4–7]. In this paper, we describe some implementations of the adaptive method presented in [8]. Our goal is to deal with the basic issues in open-loop and closed-loop optimization with due account of the dynamic facets of optimal control problems. In our study, we made use of the experience gained by implementating the original simplex method in LP. It is well known [3] that its efficiency greatly improved after it had been adapted to transportation problems and general LP problems with two-sided primal constraints by taking into account the structure of these problems in an appropriate manner.

Here, we use bounded discrete-time controls defined on a quantized time axis as feasible controls for optimizing continuous dynamical systems on a finite time interval. In our view, this is a reasonable choice for dynamic problems whose solutions cannot be constructed and implemented without using discrete-time computing and actuating devices. However, the use of discrete-time controls eliminates some analytical difficulties, but hardly simplifies the extremum problem in the constructive sense when the quantization period is short and high accuracy is desired. In particular, little effect is achieved by directly applying standard LP methods (e.g., the simplex method or interior point method), methods of linear optimal control theory, or Pontryagin's maximum principle in open-loop solutions to dynamic problems, as well as by invoking Bellman's dynamic programming approach for constructing optimal feedback controls.

The implementation of the adaptive method described here has the following advantages over their prototype: (i) the size of the support (the basic tool in the method), which largely determines the computational cost of iteration of the method, depends only on the number of terminal constraints in an optimal control problem rather than on all of its basic constraints; (ii) the implementations involve only the parameters of the original control problem, which significantly reduces the required memory resources and improves the computations, accuracy; (iii) the basic operations are performed on original (primal) and dual systems, rather than on the auxiliary objects obtained by reducing an optimal control problem to an equivalent LP problem; (iv) by storing a small amount of extra information and using parallel computations, the time required for integrating the primal and dual systems in the dual part of iteration is substantially reduced, which speeds up open-loop optimization and construction of current supports and optimal feedback realizations in finding a positional solution; (v) the efficiency of the method is virtually independent of the quantization period.

The paper is organized as follows. The canonical optimal control problem is formulated in Section 2. In Section 3, we define the support, which is the basic tool in the methods proposed here; describe the primal and dual methods for its dynamic identification; and formulate optimality and suboptimality criteria as maximum and ε-maximum principles, respectively. A procedure for replacing a feasible control in primal methods for constructing optimal open-loop controls is described in Section 4. Its combinations with two supplementary procedures for support replacement (described in Sections 5 and 6) provide two primal methods for determining optimal open-loop controls. Two dual methods for calculating optimal open-loop controls are developed in Section 7. In Section 8, we present a reducing procedure constructing (with any prescribed accuracy) piecewise continuous optimal controls with discontinuities that may be located at any point in the control time interval. Section 9 describes a new approach in constructing a closed-loop solution to an optimal control problem (synthesis of an optimal feedback control [9]), which essentially relies on the dual methods developed in the preceding sections. The results are illustrated in Section 10 by the optimal control of the oscillator consisting of two masses connected by an elastic link. The behavior of the oscillator is governed by four differential equations. An analysis of numerical results demonstrates the higher efficiency of one of the dual methods in constructing an optimal open-loop control and calculating optimal feedback control realizations, as compared to conventional LP methods, the maximum principle, and dynamic programming methods.

2. THE CANONICAL OPTIMAL CONTROL PROBLEM

Let $T = [t_*, t^*]$, $h = (t^* - t_*)/N$ (where $t_* < t^* < +\infty$ and N is a positive integer), and $T_u = \{t_*, t_* + h, \ldots, t^* - h\}$. A function $u(t), t \in T$ is called a discrete-time control (with quantization period h) if $u(t) = u(t_* + kh)$ at $t \in [t_* + kh, t_* + (k+1)/h[$, where $k = \overline{0, N-1}$.

In the class of discrete-time controls, consider the linear terminal control problem

$$c'x(t^*) \longrightarrow \max, \quad \dot{x} = A(t)x + b(t)u, \quad x(t_*) = x_0,$$
$$Hx(t^*) = g, \quad |u(t)| \le 1, \quad t \in T, \tag{2.1}$$
$$x \in \mathbb{R}^n, \quad u \in \mathbb{R}, \quad g \in \mathbb{R}^m, \quad \operatorname{rank} H = m < n,$$

where $A(t)$ and $b(t)$ ($t \in T$) are a piecewise continuous $n \times n$ matrix and an n-dimensional vector functions, respectively. The dimension n of the state $x = x(t)$ of the system to be optimized is called the order of problem (2.1).

The open-loop feasible, optimal, and ε-optimal controls, $u^\varepsilon(\cdot)$, $u^0(\cdot)$, and $u(\cdot) = (u(t), t \in T)$, as well as the corresponding trajectories $x(\cdot)$, $x^0(\cdot)$, and $x^\varepsilon(\cdot)$ of system (2.1), are defined conventionally (see [8]).

With a view to applying LP concepts and adapting LP methods, we use the Cauchy formula (see [10]) to rewrite problem (2.1) in an equivalent form as

$$\sum_{t \in T_u} c_h(t)u(t) \longrightarrow \max, \quad \sum_{t \in T_u} d_h(t)u(t) = g_0, \quad |u(t)| \le 1, \quad t \in T_u. \tag{2.2}$$

Here,

$$c_h(t) = \int_t^{t+h} \psi'_c(\vartheta)b(\vartheta)d\vartheta, \quad d_h(t) = \int_t^{t+h} G(\vartheta)b(\vartheta)d\vartheta,$$

$$g_0 = g - Hx_0(t^*),$$

$x_0(t), t \in T$ is the trajectory of system (2.1) with $u(t) = 0, t \in T$; $\psi_c(t), t \in T$ is a solution to the dual equation

$$\dot{\psi} = -A'(t)\psi \tag{2.3}$$

subject to the initial condition

$$\psi(t^*) = c;$$

and $G(t), t \in T$ is an $m \times n$ matrix function that solves the equation

$$\dot{G} = -GA(t), \quad G(t^*) = H. \tag{2.4}$$

The goal of the analysis that follows is to apply the concepts and operations employed in the adaptive method [8] to problem (2.2) with due account of the "dynamic nature" of its elements.

3. SUPPORT AND CRITERIA FOR OPTIMALITY AND SUBOPTIMALITY

Following [8], we take an arbitrary subset $T_{supp} = \{t_l,\ l = \overline{1, m}\}$ in T_u and construct the $m \times m$ matrix $D_{supp} = (d_h(t),\ t \in T_{supp})$.

The set T_{supp} is called the support of problem (2.2) if $\det D_{supp} \neq 0$.

To formulate a "dynamic" criterion for the validity of T_{supp} as a support, we define the functions

$$\chi_l(s) = \int_{t_l}^{t_l + h} F(s, \vartheta) b(\vartheta) d\vartheta, \quad s = t_l + h, \ldots, t^*, \quad l = \overline{1, m}, \tag{3.1}$$

$$\xi'_i(t) = h'_{(i)} F(t^*, t), \quad t \in T, \quad i = \overline{1, m}, \tag{3.2}$$

where $h_{(i)}$ is the ith row of H, $F(s, \vartheta) = F(s)F^{-1}(\vartheta)$, $\dot{F} = A(t)F$, and $F(0) = E$.

The function defined by (3.1) is a solution to equation (2.1) supplemented with the following initial condition and control:

$$x(t_*) = 0, \quad u(t) = \begin{cases} 1, & t \in [t_l, t_l + h[, \\ 0, & t \in T \backslash [t_l, t_l + h[. \end{cases} \tag{3.3}$$

The function defined by (3.2) solves dual equation (2.3) subject to the initial condition

$$\psi(t^*) = h_{(i)}. \tag{3.4}$$

The matrix D_{supp} can be constructed by the following two methods:

(i) (primal method) m solutions to primal system (2.1) subject to (3.3) are constructed and multiplied by the matrix H at t^* to obtain

$$D_{supp} = (H\chi_l(t^*), l = \overline{1, m}); \tag{3.5}$$

(ii) (dual method) m solutions to dual equation (2.3) subject to initial conditions (3.4) are constructed on the segments $[t, t + h]$ ($t \in T_{supp}$) to obtain the matrix

$$D_{supp} = \begin{pmatrix} \displaystyle\int_t^{t+h} \xi'_i(\vartheta) b(\vartheta) d\vartheta, & t \in T_{supp} \\ i = \overline{1, m} \end{pmatrix}. \tag{3.6}$$

The set T_{supp} is the support of problem (2.2) if and only if either (3.5) or (3.6) is a nonsingular matrix.

Following [8], we call the pair $\{u(\cdot), T_{supp}\}$ consisting of a feasible control and a support control. A support control is referred to as primally nondegenerate if $|u(t)| < 1$, where $t \in T_{supp}$.

First of all, we use the support T_{supp} to formulate a criterion for determining an optimal control. To do this, we use T_{supp} to find an m-vector ν of potentials (the vector of Lagrange multipliers associated with the support) by solving the equation $\nu' D_{supp} = c'_{supp}$, where $c_{supp} = (c_h(t), t \in T_{supp})$.

Having determined ν, we construct an analogue of the estimation vector (see [11]) as the cocontrol

$$\Delta_h(t) = c_h(t) - \nu' d_h(t), \quad t \in T_u.$$

By using (2.3), it is easy to show that

$$\Delta_h(t) = \int_t^{t+h} \psi'(\vartheta) b(\vartheta) d\vartheta, \quad t \in T_u, \tag{3.7}$$

where $\psi(t)$, $t \in T$ (cotrajectory) solves dual equation (2.3) subject to the initial condition

$$\psi(t^*) = c - H'\nu.$$

The LP optimality criterion (see [11]) can now be formulated as follows: if a support control $\{u(\cdot), T_{supp}\}$

is such that

$$\Delta_h(t) \ge 0 \text{ for } u(t) = 1, \quad \Delta_h(t) \le 0 \text{ for } u(t) = -1, \quad \Delta_h(t) = 0 \text{ for } |u(t)| < 1, \quad t \in T_u, \tag{3.8}$$

then $u(\cdot)$ is an optimal control of problem (2.1).

Vice versa, if $u(\cdot)$ is an optimal control and the support control $\{u(\cdot), T_{\text{supp}}\}$ with a certain T_{supp} is primal nondegenerate, then relations (3.8) hold for this support control.

Based on the expression for the cocontrol in terms of the cotrajectory given by (3.7), the optimality criterion can be formulated as follows.

Maximum principle. If the maximum condition

$$\int_t^{t+h} \psi'(\vartheta)b(\vartheta)d\vartheta u(t) = \max_{|u| \le 1} \int_t^{t+h} \psi'(\vartheta)b(\vartheta)d\vartheta u, \quad t \in T_{\text{n}} = T_u \backslash T_{\text{supp}}, \tag{3.9}$$

holds for an optimal control $\{u(\cdot), T_{\text{supp}}\}$, then $u(\cdot)$ is an optimal control of problem (2.1). Vice versa, if $u(\cdot)$ is an optimal control and the support control $\{u(\cdot), T_{\text{supp}}\}$ with a certain T_{supp} is primally nondegenerate, then the maximum condition (3.9) holds for this support control.

The support can also be used to determine suboptimal controls. To construct an associated pseudocontrol $\omega(t)$ ($t \in T$) based on T_{supp}, we first define the $\omega(t)$ ($t \in T_{\text{n}}$) at the nonsupport points in time:

$$\omega(t) = -1 \text{ for } \Delta_h(t) < 0, \quad \omega(t) = 1 \text{ for } \Delta_h(t) > 0, \quad \omega(t) \in [-1, 1] \text{ for } \Delta_h(t) = 0, \quad t \in T_{\text{n}}.$$

By using the representation (2.2) of problem (2.1), the values of $\omega(t)$ ($t \in T_{\text{supp}}$) at the support point can be found from the equation

$$\sum_{t \in T_{\text{supp}}} d_h(t)\omega(t) + \sum_{t \in T_{\text{n}}} d_h(t)\omega(t) = g_0.$$

The dynamic method for constructing $\omega_{\text{supp}} = (\omega(t), t \in T_{\text{supp}})$ consists of the following operations: (i) calculate the value $x_0(t^*)$ of the solution $x_0(t)$, $t \in T$ to primal system (2.1) (see Section 2); (ii) calculate the value $\kappa_0(t^*)$ of the solution $\kappa_0(t)$ ($t \in T$) to equation (2.1) with $u(t) = \omega(t)$ at $t \in T_{\text{n}}$ and $u(t) = 0$ at $t \in T_{\text{supp}}$, subject to the initial condition $\kappa(t_*) = 0$; (iii) solve the equation

$$D_{\text{supp}}\omega_{\text{supp}} = g_0 - H\kappa_0(t^*). \tag{3.10}$$

If $|\omega(t)| \le 1$ at $t \in T_{\text{supp}}$, then $u^0(t) = \omega(t)$, $t \in T_u$ is an optimal control.

The solution $\kappa(t)$, $t \in T$ to primal equation (2.1) with the discrete-time control $u(t) = \omega(t)$, $t \in T_u$, subject to the initial condition $x(t_*) = x_0$, is called a pseudotrajectory.

In accordance with [8], the number

$$\beta(u(\cdot), T_{\text{supp}}) = c'\kappa(t^*) - c'x(t^*) \tag{3.11}$$

is said to be the suboptimality estimate for the support control $\{u(\cdot), T_{\text{supp}}\}$.

The following result is based on the analysis in [8].

Suboptimality criterion. For any $\varepsilon \ge 0$, a feasible control $u(\cdot)$ is ε-optimal if and only if there exists a support T_{supp} such that

$$\beta(u(\cdot), T_{\text{supp}}) \le \varepsilon.$$

It was shown in [8] that suboptimality estimate (3.11) can be represented as

$$\beta(u(\cdot), T_{\text{supp}}) = \sum_{t \in T_{\text{n}}} \Delta_h(t)[\omega(t) - u(t)].$$

In view of (3.7), this can be used to formulate the following result.

ε-Maximum principle. For any $\varepsilon \geq 0$, a feasible control $u(\cdot)$ is ε-optimal if and only if there exists a support T_{supp} such that the ε-maximum condition

$$\int_t^{t+h} \psi'(\vartheta)b(\vartheta)d\vartheta u(t) = \max_{|u| \leq 1} \int_t^{t+h} \psi'(\vartheta)b(\vartheta)d\vartheta u - \varepsilon(t), \quad t \in T_{\text{n}}, \quad \sum_{t \in T_n} \varepsilon(t) \leq \varepsilon$$

is satisfied along the cotrajectory $\psi(t)$, $t \in T_u$.

When $\varepsilon = 0$, we have a symmetric form maximum principle: for a feasible control $u(\cdot)$ to be optimal, it is necessary and sufficient that there exists a support T_{supp} such that the maximum condition (3.9) is fulfilled for the support control $\{u(\cdot), T_{\text{supp}}\}$.

We now proceed to constructing optimal and suboptimal open-loop controls, starting with the primal method.

4. PRIMAL METHOD: REPLACEMENT OF A FEASIBLE CONTROL

The primal method is based on an iteration in which a current control is replaced by a new one at each step,

$$\{u(\cdot), T_{\text{supp}}\} \longrightarrow \{\bar{u}(\cdot), \bar{T}_{\text{supp}}\}, \tag{4.1}$$

so that

$$\beta(\bar{u}(\cdot), \bar{T}_{\text{supp}}) \leq \beta(u(\cdot), T_{\text{supp}}).$$

To simplify the presentation, we assume that the support control $\{u(\cdot), T_{\text{supp}}\}$ is both primally (see Section 3) and dually nondegenerate:

$$\Delta_h(t) \neq 0, \quad t \in T_{\text{n}},$$

$$\Delta_h(t-h)\Delta_h(t+h) < 0, \text{ if } t_* < t < t^* - h, \quad t \in T_{\text{supp}},$$

$$\Delta_h(t_* + h) \neq 0, \text{ if } t_* \in T_{\text{supp}}, \quad \Delta_h(t^* - 2h) \neq 0, \text{ if } t^* - h \in T_{\text{supp}}.$$

(The general case was analyzed in [8].) A point $t \in T_{\text{n}} \backslash t_*$ is called a nonsupport zero if $\Delta_h(t-h)\Delta_h(t) < 0$. The set of nonsupport zeros is denoted by T_{n0}. Define $T_{\text{sn}} = T_{\text{supp}} \cup T_{\text{n0}} \cup \{t_*, t^*\} = \{t_k, k \in K \cup k^* + 1\}$, where $K = \{0, 1, \ldots, k^*\}$. Denote by T_k $(k \in K)$ the intervals where the cocontrol has a definite sign:

$$T_k = \{t_{*k} = t_k, t_k + h, \ldots, t_k^* = t_{k+1} - h\}, \text{ if } t_k \notin T_{\text{supp}},$$

$$T_k = \{t_{*k} = t_k + h, t_k + 2h, \ldots, t_k^* = t_{k+1} - h\}, \text{ if } t_k \in T_{\text{supp}}.$$

If $t^* - h \in T_{\text{supp}}$, set $T_{k^*} = \varnothing$. Define the vector

$$p = \gamma \sum_{k=0}^{k^*} (-1)^k \sum_{t \in T_k} d_h(t) = \gamma \sum_{k=0}^{k^*} (-1)^k \int_{t_{*k}}^{t_k^* + h} G(\vartheta)b(\vartheta)d\vartheta, \tag{4.2}$$

where

$$\gamma = \begin{cases} \operatorname{sgn}\Delta_h(t_*), \text{ if } t_* \notin T_{\text{supp}}, \\ \operatorname{sgn}\Delta_h(t_* + h), \text{ if } t_* \in T_{\text{supp}}. \end{cases}$$

Then, (3.10) takes the form

$$D_{\text{supp}}\omega_{\text{supp}} = g_0 - p.$$

Assume that the ε-maximum principle and the inequality $|\omega(t)| \leq 1$, $t \in T_{\text{supp}}$ do not hold for the initial support control[1] $\{u(\cdot), T_{\text{supp}}\}$ for a prescribed $\varepsilon \geq 0$.

[1]The construction of an initial support control corresponds to the first phase in [8] and can be implemented by the algorithm described below.

The iteration step in (4.1) is performed by executing the following procedures: (i) $u(\cdot) \longrightarrow \bar{u}(\cdot)$; and (ii) $T_{\text{supp}} \longrightarrow \bar{T}_{\text{supp}}$.

Assume that, in addition to the problem parameters $A(t)$, $b(t)$, $t \in T$, c, H, and g_0, the following data are known at the beginning of each iteration step: (i) a feasible control $u(\cdot)$; (ii) the support T_{supp}; (iii) the set T_{n0}; (iv) the support matrix D_{supp}; (v) $G(t)$ and $\psi_c(t)$ at $t \in T_{\text{sn}} \backslash t^*$; (vi) the value of γ and the support values of the pseudocontrol $\omega(t)$ at $t \in T_{\text{supp}}$; (vii) a vector p; (viii) the vector of potentials ν; (ix) a suboptimality estimate $\beta(u(\cdot), T_{\text{supp}})$.

The data required for the first iteration step are prepared as described in Section 3. Subsequently, they are transformed in the course of iteration.

A new feasible control is constructed as

$$\bar{u}(\cdot) = u(\cdot) + \Theta^0 l(\cdot), \tag{4.3}$$

where the direction $l(\cdot)$ is set by the function

$$l(\cdot) = \omega(\cdot) - u(\cdot), \tag{4.4}$$

with $\omega(t) = (-1)^k\gamma$ at $t \in T_k$ $(k = \overline{0, k^*})$. The values of $\omega(t)$, $t \in T_{\text{supp}}$ are taken from the computer memory.

The derivative $\partial c'x(t^*)/\partial l(\cdot)$ in the direction defined by (4.4) equals $\beta(u(\cdot), T_{\text{supp}}) > \varepsilon \geq 0$. The step $\Theta^0 > 0$ in (4.3) is computed as

$$\Theta^0 = \min\{1, \Theta(t^0)\},$$

where

$$\Theta(t^0) = \min \Theta(t), \quad t \in T_{\text{supp}},$$

$$\Theta(t) = \begin{cases} (-1 - u(t))/l(t), & \text{if } l(t) < 0, \\ (1 - u(t))/l(t), & \text{if } l(t) > 0, \\ +\infty, & \text{if } l(t) = 0, \quad t \in T_{\text{supp}}. \end{cases}$$

The new feasible control $\bar{u}(\cdot)$ satisfies the equation

$$\beta(\bar{u}(\cdot), T_{\text{supp}}) = (1 - \Theta^0)\beta(u(\cdot), T_{\text{supp}}).$$

If $\beta(\bar{u}(\cdot), T_{\text{supp}}) \leq \varepsilon$, then the solution to problem (2.2) is terminated and $\bar{u}(\cdot)$ is adopted as an ε-optimal control; otherwise, the support is replaced.

5. PRIMAL METHOD: ELEMENTARY REPLACEMENT OF A SUPPORT

As a first step in constructing a new support $\bar{T}_{\text{supp}}$, we determine the variation (the direction of change) $\Delta\nu$ of the vector of potentials by solving the system of linear equations

$$-D'_{\text{supp}}\Delta\nu = (\Delta\delta_h(t), t \in T_{\text{supp}}), \tag{5.1}$$

where $\Delta\delta_h(t^0) = \operatorname{sgn}\bar{u}(t^0)$ and $\Delta\delta_h(t) = 0$, $t \in T_{\text{supp}} \backslash t^0$.

Define

$$\begin{gathered} \Delta\delta_h(t) = -\Delta\nu' d_h(t) = -\int_t^{t+h} \Delta\nu' G(\vartheta) b(\vartheta) d\vartheta, \quad t \in T_{\text{n}} \quad (\text{variation of cocontrol}), \\ \delta_h(t, \sigma) = \Delta_h(t) + \sigma\Delta\delta_h(t), \quad t \in T_u, \quad \sigma \geq 0 \quad (\text{varied cocontrol}). \end{gathered} \tag{5.2}$$

In what follows, it is assumed that

$$\Delta\delta_h(t_{*k})\Delta\delta_h(t^*_{k-1}) > 0, \text{ if } t^*_k \in T_{\text{n0}}, \quad k = \overline{1, k^*}.$$

The new support $\bar{T}_{\text{supp}}$ is constructed in several steps.

Before describing the operations required to execute the support-replacing procedure, we explain how the data stored in the computer are used to calculate the desired values of $\Delta_h(t)$, $\Delta\delta_h(t)$, and $d_h(t)$ at $t \in T_u$. Suppose that the values of $G(t)$ and $\psi_c(t)$ at some point $t \in T_u$ are stored in the computer memory.

To find $\Delta_h(t)$, $\Delta\delta_h(t)$, and $d_h(t)$, we integrate (2.3) and (2.4) over $[t, t + h]$, using $\psi_c(t)$ and $G(t)$ as initial conditions:

$$\Delta_h(t) = \int_t^{t+h} [\psi_c'(\vartheta) - \nu' G(\vartheta)] b(\vartheta) d\vartheta, \quad \Delta\delta_h(t) = -\int_t^{t+h} \Delta\nu' G(\vartheta) b(\vartheta) d\vartheta,$$
$$d_h(t) = \int_t^{t+h} G(\vartheta) b(\vartheta) d\vartheta. \tag{5.3}$$

The values of $\Delta_h(t + h)$, $\Delta\delta_h(t + h)$, and $d_h(t + h)$ are calculated by integrating (2.3) and (2.4) over $[t, t + 2h]$, with the initial conditions $\psi_c(t)$ and $G(t)$:

$$\Delta_h(t+h) = \int_{t+h}^{t+2h} [\psi_c'(\vartheta) - \nu' G(\vartheta)] b(\vartheta) d\vartheta, \quad \Delta\delta_h(t+h) = -\int_{t+h}^{t+2h} \Delta\nu' G(\vartheta) b(\vartheta) d\vartheta,$$
$$d_h(t+h) = \int_{t+h}^{t+2h} G(\vartheta) b(\vartheta) d\vartheta.$$

To calculate $\Delta_h(t - h)$, $\Delta\delta_h(t - h)$, and $d_h(t - h)$, we integrate (2.3) and (2.4) over $[t - h, t]$, with initial conditions given by $\psi_c(t)$ and $G(t)$:

$$\Delta_h(t-h) = \int_{t-h}^{t} [\psi_c'(\vartheta) - \nu' G(\vartheta)] b(\vartheta) d\vartheta, \quad \Delta\delta_h(t-h) = -\int_{t-h}^{t} \Delta\nu' G(\vartheta) b(\vartheta) d\vartheta,$$
$$d_h(t-h) = \int_{t-h}^{t} G(\vartheta) b(\vartheta) d\vartheta.$$

Let $\varphi(\sigma)$ be the value of the performance criterion in the problem dual to (2.2) at the function defined by (5.2). According to [8], in a small right neighborhood of $\sigma = 0$, the function $\varphi(\sigma)$ ($\sigma \geq 0$) is linear and decreases at the rate

$$\alpha^1 = -|\omega(t^0) - \bar{u}(t^0)| < 0.$$

This behavior of the function is retained as σ is increased from $\sigma = 0$ to $\sigma = \sigma^1 > 0$ when a new zero $t(\sigma^1)$ of varied cocontrol (5.2) is found. The calculation of σ^1 is the first step in the construction of a new support.

Step 1.

1. For each point $t \in T_{n0}$, calculate $\sigma(\tilde{t})$ at which a new zero of function (5.2) appears at either $\tilde{t} = t$ or $\tilde{t} = t - h$.

Compute $\Delta_h(t)$ and $\Delta\delta_h(t)$ in (5.3). If $\Delta_h(t)\Delta\delta_h(t) < 0$, set $\tilde{t} = t$; otherwise, compute $\Delta_h(t - h)$ and $\Delta\delta_h(t - h)$ and set $\tilde{t} = t - h$. Compute

$$\sigma(t) = -\Delta_h(\tilde{t})/\Delta\delta_h(\tilde{t}).$$

2. The variation of t^0 with increasing $\sigma > 0$ is analyzed. Let k_0 be the index of t^0 in T_{sn}.

If $t^0 < t^* - h$ and $(-1)^{k_0}\gamma\Delta\delta_h(t^0) < 0$, then $p := p - (-1)^{k_0}\gamma d_h(t^0)$, $t^0 := t^0 + h$, and $\tilde{t}^0 = t^0$. Introduce the point t^0 into the set T_{n0}. If $t^0 = t_*$, then $k^* := k^* + 1$, $\gamma := -\gamma$, the points in T_{sn} are renumbered, and $G(t^0)$ and $\psi_c(t^0)$ are stored instead of $G(\tilde{t}^0)$ and $\psi_c(\tilde{t}^0)$.

If $t^0 > t_*$ and $(-1)^{k_0}\gamma\Delta\delta_h(t^0) > 0$, then $\tilde{t}^0 = t^0 - h$ and $p := p + (-1)^{k_0}\gamma d_h(t^0)$. Introduce t^0 into the set $T_{\text{n}0}$. Compute $\Delta_h(\tilde{t}^0)$, $\Delta\delta_h(\tilde{t}^0)$, and $\sigma(t^0) = -\Delta_h(\tilde{t}^0)/\Delta\delta_h(\tilde{t}^0)$.

If $t^0 = t^* - h$ and $(-1)^{k_0}\gamma\Delta\delta_h(t^0) < 0$, then the zero of the cocontrol is said to disappear at the right boundary of the segment T. Set $p := p - (-1)^{k_0}\gamma d_h(t^0)$ and $k^* := k^* - 1$ and delete $G(t^* - h)$ and $\psi_{\text{c}}(t^* - h)$ from the computer memory.

The situation when $t^0 = t_*$ and $\gamma\Delta\delta_h(t^0) > 0$ is interpreted as the disappearance of a zero of the cocontrol at the left boundary of T. Set $p := p + d_h(t_*)\gamma$.

3. The situation at the endpoints of T is analyzed. To simplify calculations, we skip the cases when t_*, $t^* - h \in T_{\text{оп}}\backslash t^0$, which are dealt with by applying a scheme described below for a moving zero jumping over a support zero.

Compute

$$\sigma(t_*) = \begin{cases} -\Delta_h(t_*)/\Delta\delta_h(t_*), & \text{if } \Delta_h(t_*)\Delta\delta_h(t_*) < 0, \\ +\infty, & \text{if } \Delta_h(t_*)\Delta\delta_h(t_*) \ge 0, \end{cases}$$

$$\sigma(t^*) = \begin{cases} -\Delta_h(t^* - h)/\Delta\delta_h(t^* - h), & \text{if } \Delta_h(t^* - h)\Delta\delta_h(t^* - h) < 0, \\ +\infty, & \text{if } \Delta_h(t^* - h)\Delta\delta_h(t^* - h) \ge 0. \end{cases}$$

Define the set of moving zeros of the cocontrol as $T_{\text{n}}^0 = T_{\text{n}0} \cup t_* \cup t^*$.

Choose

$$\sigma^1 = \sigma(t^1) = \min_{t \in T_{\text{n}}^0} \sigma(t)$$

from the sequence $\sigma(t)$ $(t \in T_{\text{n}}^0)$, which completes Step 1.

Remark 1. If the function $\Delta(t) = [\psi_{\text{c}}'(t) - \nu' G(t)]b(t)$, $t \in T$ has more than one stationary point in the interval between two consecutive zeros of the cocontrol, then two new nonsupport zeros of cocontrol may appear in this interval during iteration. To detect them, it is sufficient to monitor the cocontrol values at the nodes in whose neighborhoods $\dot{\Delta}(t) = 0$ as σ is increased. In complex situations, new zeros of $\dot{\Delta}(t)$, $t \in T$ may also appear. They are detected by using the function $\ddot{\Delta}(t)$, $t \in T$. The number of auxiliary functions required for detecting new zeros of the cocontrol is finite and depends both on the complexity of a particular problem and on its support.

Assume that, when $\sigma = \sigma^1$, a single zero of function (5.2) appears at $\tilde{t}^1 \in T_u$.

When $\sigma = \sigma^1$, the rate of change in $\varphi(\sigma)$ $(\sigma \ge 0)$ increases by the positive increment $\Delta\alpha^1 = 2|\Delta\delta_h(\tilde{t}^1)| > 0$ and, in a small right neighborhood of σ^1, becomes equal to

$$\alpha^2 = \alpha^1 + \Delta\alpha^1 > \alpha^1.$$

If $\alpha_2 \ge 0$, we go to the final step. Otherwise, the data stored in the computer memory are changed before going to the next step.

Let k be the index of the point t^1 in T_{sn}. We distinguish the following two cases:

1. $\sigma^1 = \sigma(t^1)$.

2. $\sigma^1 = \sigma(t^1) = \sigma(t^1 + h)$.

1.1. $\tilde{t}^1 = t^1 = t - h$ or $\tilde{t}^1 = t^1 - h = t + h$ for a certain $t \in T_{\text{supp}}\backslash t^0$. This situation is interpreted as one of matching zeros of the cocontrol. By the nondegeneracy assumption, a new support control $\{\bar{u}(\cdot), \bar{T}_{\text{supp}}\}$, much the same as $\{u(\cdot), T_{\text{supp}}\}$, is nondegenerate; therefore, the moving zero jumps the support zero as $\sigma > \sigma^1$ increases.

1.1.1. $\tilde{t}^1 = t^1 = t - h$, $t \in T_{\text{supp}} \backslash t^0$. Set $p := p - 2(-1)^k \gamma d_h(\tilde{t}^1)$. Instead of $G(t^1)$ and $\psi_c(t^1)$, we calculate and store $G(t^1 + 2h)$ and $\psi_c(t^1 + 2h)$. The point t^1 in T_{n0} is replaced by $t^1 + 2h$: $t^1 := t^1 + 2h$, $\tilde{t}^1 := t^1$. The indices of the points t and t^1 in T_{sn} are changed.

1.1.2. $\tilde{t}^1 = t^1 - h = t + h$, $t \in T_{\text{supp}} \backslash t^0$. Set $p := p + 2d_h(t^1)(-1)^k \gamma$. Instead of $G(t^1)$ and $\psi_c(t^1)$, we store $G(t)$ and $\psi_c(t)$. The point t^1 in T_{n0} is replaced by t: $t^1 := t$, $\tilde{t}^1 := t^1 - h$. The indices of t and t^1 in T_{sn} are changed.

If situation 1.1 is not the care, then the location of t^1 is analyzed.

1.2. $t_* < t^1 < t^* - h$.

1.2.1. $\tilde{t}^1 = t^1$. Set $p := p - 2(-1)^k d_h(t^1)\gamma$. Instead of $G(t^1)$ and $\psi_c(t^1)$, we calculate and store $G(t^1 + h)$ and $\psi_c(t^1 + h)$. The point t^1 in T_{n0} is replaced by $t^1 + h$: $t^1 := t^1 + h$, $\tilde{t}^1 := t^1$.

1.2.2. $\tilde{t}^1 = t^1 - h$. Calculate $G(\tilde{t}^1)$, $\psi_c(\tilde{t}^1)$, and $d_h(\tilde{t}_1)$; set $p := p + 2(-1)^k d_h(\tilde{t}^1)\gamma$; and store $G(\tilde{t}^1)$ and $\psi_c(\tilde{t}^1)$ instead of $G(t^1)$ and $\psi_c(t^1)$. The point t^1 in T_{n0} is replaced by $\tilde{t}^1$: $t^1 := \tilde{t}^1$, $\tilde{t}^1 := t^1 - h$.

1.3. $t^1 = \tilde{t}^1 = t_*$. A new zero appears at the left endpoint of T. Set $p := p - 2d_h(t_*)\gamma$, $k^* := k^* + 1$, and $\gamma := -\gamma$. Calculate and store $G(t_* + h)$ and $\psi_c(t_* + h)$. Introduce the point $t_* + h$ into T_{n0}, set $t^1 = t_* + h$ and $\tilde{t}^1 = t^1$, and renumber the points in T_{sn}.

1.4. $t^1 = t^*$ and $\tilde{t}^1 = t^* - h$. A new zero appears at the right endpoint of T. Calculate $G(t^* - h)$, $\psi_c(t^* - h)$, and $d_h(t^* - h)$ and store $G(t^* - h)$ and $\psi_c(t^* - h)$. Set $p := p - 2(-1)^{k^*} d_h(t^* - h)\gamma$ and $k^* := k^* + 1$. Introduce the point $t^* - h$ into T_{n0}, set $t^1 = t^* - h$ and $\tilde{t}^1 = t^1 - h$, and renumber the points in T_{sn}.

2.1. $t_* < t^1 < t^* - h$. This situation is interpreted as a merger, with subsequent disappearance, of two zeros of the cocontrol. Remove t^1 and $t^1 + h$ from T_{n0}; set $p := p - 2(-1)^k d_h(t_1)\gamma$ and $k^* := k^* - 2$; renumber the points in T_{sn}; and delete $G(t^1)$, $\psi_c(t^1)$, $G(t^1 + h)$, and $\psi_c(t^1 + h)$ from the computer memory.

2.2. $t^1 = t_*$. A zero of the cocontrol disappears at the left boundary of T. Remove $t_* + h$ from T_{n0}; set $\gamma := -\gamma$, $p := p + 2d_h(t_*)\gamma$, and $k^* := k^* - 1$; renumber the points in T_{sn}; and delete $G(t_* + h)$ and $\psi_c(t_* + h)$ from the computer memory.

2.3. $t^1 = t^* - h$. A zero of the cocontrol disappears at the right boundary of T. Remove $t^* - h$ from T_{n0}, set $p := p - 2(-1)^k d_h(t^* - h)\gamma$ and $k^* := k^* - 1$, renumber the points in T_{sn}, and delete $G(t^* - h)$ and $\psi_c(t^* - h)$ from the computer memory.

Step 2. If situation 1 occurs after Step 1, then compute $\Delta_h(\tilde{t}^1)$ and $\Delta\delta_h(\tilde{t}^1)$ and replace $\sigma(t^1)$ in the sequence $\sigma(t)$, $t \in T_{\text{n}}^0$ by

$$\sigma(t^1) = \begin{cases} -\Delta_h(\tilde{t}^1)/\Delta\delta_h(\tilde{t}^1), & \text{if } \Delta_h(\tilde{t}^1)\Delta\delta_h(\tilde{t}^1) < 0, \\ +\infty, & \text{if } \Delta_h(\tilde{t}^1)\Delta\delta_h(\tilde{t}^1) \geq 0. \end{cases}$$

The computation of

$$\sigma^2 = \sigma(t^2) = \min_{t \in T_{\text{n}}^0} \sigma(t)$$

completes Step 2.

Calculate

$$\Delta\alpha^2 = 2\left|\Delta\delta_h(\tilde{t}^2)\right|, \quad \alpha^3 = \alpha^2 + \Delta\alpha^2.$$

When $\alpha^3 \geq 0$, go to the final step. When $\alpha^3 < 0$, update the stored data as described after Step 1, with t^1 and $\tilde{t}^1$ replaced by t^2 and $\tilde{t}^2$, respectively.

Executing K_0 steps, we obtain (see [7]) the inequalities

$$\alpha^{K_0} < 0, \quad \alpha^{K_0+1} \geq 0, \tag{5.4}$$

which imply going to the final step.

Final step. It is executed when inequalities (5.4) become valid after K_0 steps and the points t^{K_0} and $\tilde{t}^{K_0}$ are determined: $\sigma^{K_0} = \sigma(t^{K_0})$.

Construct a new support

$$\bar{T}_{\text{supp}} = (T_{\text{supp}} \backslash t^0) \cup \tilde{t}^{K_0}.$$

The suboptimality estimate for the new support control $\{\bar{u}(\cdot), \bar{T}_{\text{supp}}\}$ is

$$\beta(\bar{u}(\cdot), \bar{T}_{\text{supp}}) = (1 - \Theta^0)\beta(u(\cdot), T_{\text{supp}}) + \sum_{k=1}^{K_0} \alpha^k(\sigma^k - \sigma^{k-1}), \quad \sigma^0 = 0.$$

If $\beta(\bar{u}(\cdot), \bar{T}_{\text{supp}}) \leq \varepsilon$, then solution of problem (2.1) is terminated, because $\bar{u}(\cdot)$ is an ε-optimal control. If $\beta(\bar{u}(\cdot), \bar{T}_{\text{supp}}) > \varepsilon$, the iteration step is completed by preparing the data required to execute the next iteration step.

Let k be the index of the point t^{K_0} in T_{sn}. Since the support control $\{\bar{u}(\cdot), \bar{T}_{\text{supp}}\}$ is assumed to be nondegenerate, the final step can be executed in situations 1.2–1.4, 2.2, and 2.3. The following changes must be made in the stored data in each of these situations.

1.2.1, 2.3. Set $\bar{p} = p - (-1)^k \gamma d_h(\tilde{t}^{K_0})$ and $\bar{T}_{\text{n}0} = T_{\text{n}0} \backslash t^{K_0}$.

1.2.2. Calculate $G(\tilde{t}^{K_0})$, $\psi_{\text{c}}(\tilde{t}^{K_0})$, and $d_h(\tilde{t}^{K_0})$. Instead of $G(t^{K_0})$ and $\psi_{\text{c}}(t^{K_0})$, the values of $G(\tilde{t}^{K_0})$ and $\psi_{\text{c}}(\tilde{t}^{K_0})$ are stored. Set $\bar{p} = p + (-1)^k \gamma d_h(\tilde{t}^{K_0})$ and $\bar{T}_{\text{n}0} = T_{\text{n}0} \backslash t^{K_0}$.

1.3. Set $\bar{p} = p - \gamma d_h(t_*)$.

1.4. Calculate and store $G(t^* - h)$, $\psi_{\text{c}}(t^* - h)$, and $d_h(t^* - h)$. Set $\bar{p} = p - (-1)^{k^*} \gamma d_h(t^* - h)$ and $\bar{k}^* = k^* + 1$.

2.2. Set $\bar{p} = p - \gamma d_h(t_*)$, $\bar{T}_{\text{n}0} = T_{\text{n}0} \backslash t_* + h$, $\bar{\gamma} = -\gamma$, and $\bar{k}^* = k^* - 1$. Delete $G(t_* + h)$ and $\psi_{\text{c}}(t_* + h)$ from the computer memory.

A new support matrix $\bar{D}_{\text{supp}}$ is constructed by replacing the column $d_h(t^0)$ in D_{supp} with $d_h(\tilde{t}^{K_0})$. The support values of the pseudocontrol $\bar{\omega}(t)$ $(t \in \bar{T}_{\text{supp}})$ are found by solving equation (3.10), with the use of the known value of $\sum_{t \in \bar{T}_{\text{n}}} d_h(t)\bar{\omega}(t) = \bar{p}$. If $|\bar{\omega}(t)| \leq 1$ $(t \in T_{\text{supp}})$, the computations are terminated, because $\bar{\omega}(t)$, $t \in T_u$ is an optimal control.

The method is finite under certain nondegeneracy conditions (see [11]). There exists a modified adaptive method, described in [11], that is finite for any problem (2.2).

6. BATCH REPLACEMENT OF A SUPPORT: AN ALTERNATIVE PRIMAL METHOD

Let us describe another method for problem (2.2), based on a different support-replacing procedure.

The point $t^0 \in T_{\text{supp}}$ determined by executing the first procedure in Section 4 is such that $|\omega(t^0)| > 1$. In the general case, several support points may have this property. We use some of them to construct a new variation $\Delta\delta_h(t)$ $(t \in T_u)$, which leads to a new support-replacing procedure.

Define the function

$$\rho(s) = \rho(\omega(s), [-1, 1]), \quad s \in T_{\text{supp}},$$

where $\rho(\omega, [-1, 1])$ is the distance from the point ω to the segment $[-1, 1]$. Choose a number $\mu > 0$ (the parameter of the method) such that the set

$$T^{+}_{\text{supp}} = \{s \in T_{\text{supp}}\colon \rho(s) \ge \mu\} = \{t^{01}, \dots, t^{0l}\}$$

is nonempty.

Use $\rho(s)$ ($s \in T^{+}_{\text{supp}}$) to construct a variation $\Delta\delta_h(t)$, $t \in T_u$ of the cocontrol as follows. First, set

$$\Delta\delta_h(t) = \rho(t)\operatorname{sgn}\omega(t), \quad t \in T^{+}_{\text{supp}}, \quad \Delta\delta_h(t) = 0, \quad t \in T_{\text{supp}} \backslash T^{+}_{\text{supp}}.$$

Then, find the variation $\Delta\nu$ of the vector of potentials in (5.1). The functions $\Delta\delta_h(t)$ ($t \in T_{\text{n}}$) and $\delta_h(t, \sigma)$ ($t \in T_u$, $\sigma \ge 0$) are constructed as described above (see Section 5). The value of α^1 is now

$$\alpha^1 = -\sum_{t \in T_{\text{supp}}} \Delta\delta_h(t)[\omega(t) - \bar{u}(t)].$$

Instead of $t^0(\sigma)$ ($\sigma \ge 0$), we introduce l functions $t^{01}(\sigma), \dots, t^{0l}(\sigma)$, $\sigma \ge 0$. Increasing σ, we again find a minimum point $\sigma^* = \sigma^{K_0}$ of the dual performance criterion. The corresponding point $\tilde{t}^{K_0} \in T_{\text{n}}$ is included in the new support in addition to the previous support points $t \in T_{\text{supp}} \backslash T^{+}_{\text{supp}}$. The other $l - 1$ new support points in $\bar{T}_{\text{supp}}$ are the elements of $T_{\text{n}0}$ at which the absolute value of function (5.2) at $\sigma = \sigma^*$ is minimal.

An iteration step completed by verifying that the resulting set $\bar{T}_{\text{supp}}$ is a support. The value of μ can be refined in the course of iteration by calculating suboptimality estimates for the resulting support controls. At the final iteration steps, when $\max_{s \in T_{\text{supp}}} \rho(s)$ becomes insignificant, it is reasonable to employ the first method with an elementary support replacement (see Sections 4 and 5).

7. DUAL METHODS

The support-replacing procedures employed in the two primal methods for problem (2.2) described above are now used to construct dual methods, which do not require any information on the initial feasible control.

Suppose that only the model in problem (2.2) and a support T_{supp} are known[2]. The support T_{supp} is used to find the associated pseudocontrol $\omega(t)$, $t \in T_u$ (see Section 3).

The quasi-control associated with T_{supp} is defined as

$$\tilde{\omega}(t) = \begin{cases} \omega(t), & \text{if } |\omega(t)| \le 1, \\ \operatorname{sgn}\omega(t), & \text{if } |\omega(t)| > 1, \quad t \in T_u. \end{cases}$$

The quasi-control differs from the pseudocontrol in that the primal constraints $|\tilde{\omega}(t)| \le 1$, $t \in T_u$ hold for the former, but the terminal constraint $H\tilde{x}(t^*) = g$ may be violated on the corresponding trajectory $\tilde{x}(t)$, $t \in T$.

The residual $\tilde{f} = g - H\tilde{x}(t^*) = g_0 - p - D_{\text{supp}}\tilde{\omega}_{\text{supp}}$ (where $\tilde{\omega}_{\text{supp}} = (\tilde{\omega}(t), t \in T_{\text{supp}})$) of the terminal constraints for quasi-controls is stored.

Let $t^0 \in T_{\text{supp}}$ be a point in time such that

$$\rho(t^0) = \max \rho(t), \quad t \in T_{\text{supp}}.$$

Using this t^0 in the support-replacing procedure of the first primal method (described in Sections 4 and 5), we obtain an iteration step of the first dual method for problem (2.2).

An iteration step of the second dual method is executed as the support-replacing procedure in the second primal method (see Sections 4 and 6) without using any information on the first procedure.

The residual $\tilde{f}$ of the terminal constraints for quasi-controls varies in the course of iteration in the dual methods. The residual value is monitored so that the computation is terminated when the required accuracy of the terminal constraints is achieved. At the same time, the degree of violation of primal constraints for

[2]An efficient method for constructing an initial support is described in [7].

pseudocontrol may still be significant at the support point. The quasi-control can also be used in the primal methods described above. This makes it possible to construct optimal open-loop controls for problems with interval terminal constraints $g_* \le Hx(t^*) \le g^*$.

8. REDUCTION OF DISCRETE-TIME OPTIMAL CONTROLS TO PIECEWISE CONSTANT OPTIMAL CONTROLS

The methods described above can be employed to construct discrete-time optimal controls for problem (2.1). Such controls can change their values only at the quantization points $t \in T_u$.

Let us describe a reducing procedure [8] for constructing optimal controls of problem (2.1) in the class of piecewise constant functions that may be discontinuous at arbitrary points in T. In nonsingular problems (2.1) with measurable feasible controls, optimal controls are piecewise constant functions.

We now define the concept of support for problem (2.1) with piecewise continuous feasible controls. Following [8], we say that $T_{\text{supp}} = \{\tau_j \in T, j = \overline{1, m}\}$ is a support if the support matrix $D_{\text{supp}} = (HF(t^*, t)b(t), t \in T_{\text{supp}})$ is nonsingular.

The reducing procedure is executed by solving the set of algebraic equations

$$\sum_{i=0}^{k^*} (-1)^i \gamma \int_{t_i}^{t_{i+1}} G(t)b(t)dt = g_0,$$
$$(c' - \nu' H)F(t^*, t_j)b(t_j) = 0, \quad j = \overline{1, k^*}, \tag{8.1}$$

for the unknowns

$$t_i, \quad i = \overline{1, k^*}, \quad \nu. \tag{8.2}$$

Here, t_i $(i = \overline{1, k^*})$ are the zeros of the cocontrol $\Delta(t) = [\psi_c'(t) - \nu' G(t)]b(t)$ $(t \in T)$, ν is the vector of potentials, $t^0 = t_*$, $t_{k^*+1} = t^*$, and $\gamma = \operatorname{sgn}\Delta(t_* + 0)$.

When $\dot{\Delta}(t_i) \ne 0$ for $i = \overline{1, k^*}$, the Jacobi matrix of system (8.1) is nonsingular. Therefore, it can be solved by Newton's method. A sufficiently good initial approximation of elements (8.2) required to ensure its convergence can be constructed by the methods described in Sections 4–7 with a small h.

9. SYNTHESIS OF A CLOSED-LOOP SOLUTION TO THE CANONICAL OPTIMAL CONTROL PROBLEM

Let us describe the dual methods presented in Section 7 as applied in the synthesis of optimal feedback controls. The foregoing analysis is focused on open-loop solutions to problem (2.1). Optimal open-loop controls play an important role in applications. They have been the subject of most published studies concerning optimal processes. Optimal feedback controls are even more important for applications, but their analysis and construction are associated with great difficulties, which explains why relevant publications have been rather scarce.

To introduce the concept of an optimal feedback control (closed-loop solution) of problem (2.1), we embed it in the family of problems

$$c'x(t^*) \longrightarrow \max, \quad \dot{x} = A(t)x + b(t)u, \quad x(\tau) = z, \quad Hx(t^*) = g, \quad |u(t)| \le 1, \quad t \in T(\tau) = [\tau, t^*], \tag{9.1}$$

parameterized by a scalar $\tau \in T_u$ and an n-vector z.

Let $u^0(t|\tau, z)$, $t \in T(\tau)$ be an optimal open-loop control in problem (9.1) at the position (τ, z), and let $X(\tau)$ be the set of states z for which problem (9.1) has a solution for a fixed τ.

In the optimal control theory, the function

$$u^0(\tau, z) = u^0(\tau|\tau, z), \quad z \in X(\tau), \quad \tau \in T_u, \tag{9.2}$$

is said to be an optimal feedback control, or a closed-loop solution, of problem (2.1). Its construction is conventionally referred to as the classical optimal feedback control problem or the synthesis of an optimal system.

The classical optimal feedback control problem has not been solved analytically even for linear problem (9.1) with $A(t) \equiv A$ and $b(t) \equiv b$, $t \ge t_*$, which is the simplest optimal control problem. A new approach to the synthesis of optimal systems was proposed in [13, 14], where it was implemented for time-independent systems by using Newton's and LP methods. Below, we describe a new implementation of the approach described in [14] that is essentially based on the dual methods for optimal control problems presented in Section 7.

The idea of the new approach is suggested by an analysis of the classical optimal feedback control problem. Assume that function (9.2) has been constructed. We apply feedback control (9.2) to the control system of problem (2.1) and analyze the behavior of the resulting closed-loop system under constantly acting piecewise continuous disturbances $w(t)$, $t \in T$:

$$\dot{x} = A(t)x + b(t)u^0(t, x) + w(t), \quad x(t_*) = x_0. \tag{9.3}$$

By the trajectory of equation (9.3), we mean a solution to the equation

$$\dot{x} = A(t)x + b(t)u(t) + w(t), \quad x(t_*) = x_0,$$

$$u(t) = u^0(\tau, x(\tau)), \quad t \in [\tau, \tau + h[, \quad \tau = t_* + kh, \quad k = 0, 1, \dots, N-1.$$

The introduction of disturbances into system (9.3) is of fundamental importance in the constructive theory of closed-loop solutions, because the idea (necessity) of feedback control is that the resulting closed-loop system is capable of rejecting unknown disturbances that cannot be taken into account in (2.1) and handled by any optimal open-loop control.

Suppose that an initial state $x(t_*) = x_0^*$ and a disturbance $w^*(t)$, $t \in T$ occur during a particular control process. They correspond to a trajectory $x^*(t)$, $t \in T$ of the closed-loop system that satisfies the identity

$$\dot{x}^*(t) \equiv A(t)x^*(t) + b(t)u^0(t, x^*(t)) + w^*(t), \quad t \in T, \quad x(t_*) = x_0^*. \tag{9.4}$$

It is evident from (9.4) that feedback control (9.2) is utilized incompletely in a control (not for all $z \in X(\tau)$, $\tau \in T_u$). Only its values $u^*(t) = u^0(t, x^*(t))$, $t \in T_u$ along an isolated trajectory $x^*(t)$, $t \in T_u$ are required. Moreover, the function $u^*(t)$, $t \in T_u$ may not be known in advance, and it suffices that the values of $u^*(\tau)$ can be calculated at points $\tau \in T_u$ corresponding to current states $x^*(\tau)$. The function $u^*(t)$, $t \in T_u$ is called a realization of an optimal feedback control in a particular control process.

An optimal feedback control $u^*(t)$, $t \in T_u$ is said to be constructed (realized) in real time if the time $s(\tau)$ required to compute $u^*(\tau) = u^0(\tau, x^*(\tau))$ in each current position $(\tau, x^*(\tau))$ does not exceed $h > 0$. This is done by a device called an optimal controller.

The signal $u^*(\tau)$ is fed as input to system (9.3) starting from $\tau + s(\tau)$ (preceded by $u^*(t) = u^*(\tau - h)$, $t \in [\tau - h + s(\tau - h), \tau + s(\tau)[$). This delay may result in a deviation of the transient process from the calculated trajectory, which is treated as an additional disturbance. It will be taken into account when calculating $u^*(\tau + h)$ at $\tau + h$. In regular and sliding modes, the difference between $u^*(\tau - h)$ and $u^*(\tau)$ is small over a greater portion of T after regularization; therefore, the optimal controller generates a transient process that differs little from that taking place in the closed-loop system with optimal feedback.

Let us show that the dual methods described in Section 7 can be used as a natural basis for designing an operation algorithm for an optimal controller. Assume that such an algorithm has been constructed and the optimal controller has operated at $t_*, t_* + h, \dots, \tau$ and produced control signals $u^*(t_*), u^*(t_* + h), \dots, u^*(\tau)$, respectively. Denote by $x^*(\tau + h)$ the state of the control system at $\tau + h$ resulting from the action of these signals and a disturbance $w^*(t)$, $t \in [t_*, \tau + h[$.

By definition (9.2), the open-loop solution $u^0(t|\tau + h, x^*(\tau + h))$, $t \in T(\tau + h)$ to problem (9.1) at $(\tau + h, x^*(\tau + h))$ is required for the controller to calculate the realization $u^*(\tau + h)$ of feedback control (9.2). By assumption, the optimal controller has constructed an open-loop solution $u^0(t|\tau, x^*(\tau))$, $t \in T(\tau)$ to problem (9.1) with $z = x^*(\tau)$ at the preceding point τ. According to the Bellman optimality principle [12], a part of this solution $u^0(t|\tau, x^*(\tau))$, $t \in T(\tau + h)$ is the optimal open-loop control for the position $(\tau + h, x^0(\tau + h))$

resulting from the action of $u^*(\tau) = u^0(\tau|\tau, x^*(\tau))$ at $(\tau, x^*(\tau))$ in the absence of any disturbance ($w(t) = 0$, $t \in [\tau, \tau + h[$). The actual state $x^*(\tau + h)$ is related to the ideal state $x^0(\tau + h)$ as follows:

$$x^*(\tau+h) = x^0(\tau+h) + \int_\tau^{\tau+h} F(\tau+h)F^{-1}(s)w^*(s)ds.$$

For small $h > 0$ and bounded $w^*(t)$ ($t \in [\tau, \tau + h)[$, the vectors $x^*(\tau + h)$ and $x^0(\tau + h)$ are nearly identical. Therefore, an optimal control $u^0(t|\tau + h, x^*(\tau + h))$, $t \in T(\tau + h)$, can be generated by the controller by correcting the already known control $u^0(t|\tau, x^*(\tau))$, $t \in T(\tau + h)$. Mathematically, this problem can be formulated as follows. Given the optimal support $T^0_{\text{supp}}(\tau)$ of problem (9.1) at position $(\tau, x^*(\tau))$, find an optimal support $T^0_{\text{supp}}(\tau + h)$ and an optimal control $u^*(\tau + h) = u^0(\tau + h|\tau + h, x^*(\tau + h))$ in the problem

$$\begin{gathered} c'x(t^*) \longrightarrow \max, \quad \dot{x} = A(t)x + b(t)u, \quad x(\tau+h) = x^*(\tau+h), \\ Hx(t^*) = g, \quad |u(t)| \le 1, \quad t \in T(\tau+h). \end{gathered} \tag{9.5}$$

Recall the notation introduced above for $(\tau, x^*(\tau))$: $T_u(\tau) = \{\tau, \tau + h, \ldots, t^* - h\}$; $T_*(\tau) = \{t_*, t_* + h, \ldots, \tau - h\}$; $T_{n0}(\tau)$ is the set of nonsupport zeros at $(\tau, x^*(\tau))$; $T_{sn}(\tau) = T_{\text{supp}}(\tau) \cup T_{n0}(\tau) \cup \{\tau, t^*\}$, $|T_{sn}(\tau)| = k^*(\tau) + 1$, and $T_k(\tau)$ ($k = \overline{0, k^*(\tau)}$) are the intervals where the cocontrol $\Delta^\tau_h(t)$ ($t \in T_u(\tau)$) has a definite sign.

Assume that the following data have been obtained by solving problem (9.1) at $(\tau, x^*(\tau))$ and stored: (i) an optimal support $T^0_{\text{supp}}(\tau)$; (ii) the set $T_{n0}(\tau)$; (iii) the support matrix $D_{\text{supp}}(\tau)$ and the vector $d_h(\tau)$; (iv) $G(t)$ and $\psi_c(t)$ at $t \in T_{sn}(\tau)\backslash t^*$; (v) $u^*(\tau)$ and the number

$$\gamma(\tau) = \begin{cases} u^*(\tau), & \tau \notin T^0_{\text{supp}}(\tau), \\ \operatorname{sgn}\Delta^\tau_h(\tau+h), & \tau \in T^0_{\text{supp}}(\tau); \end{cases}$$

(vi) the vector

$$p(\tau) = \gamma(\tau)\sum_{k=0}^{k^*(\tau)}(-1)^k \sum_{t \in T_k(\tau)} d_h(t);$$

(vii) the vector of potentials $\nu(\tau)$; (viii) the vector $y^*(\tau) = G(\tau)x^*(\tau)$.

At τ, the optimal controller solved a problem that had the following form as represented in LP terms:

$$\sum_{t \in T_u(\tau)} c_h(t)u(t) \longrightarrow \max, \quad y^*(\tau) + \sum_{t \in T_u(\tau)} d_h(t)u(t) = g,$$

$$|u(t)| \le 1, \quad t \in T_u(\tau).$$

The problem to be solved at $\tau + h$ is formulated as

$$\begin{gathered} \sum_{t \in T_u(\tau)} c_h(t)u(t) \longrightarrow \max, \\ y^*(\tau) + \int_\tau^{\tau+h} G(t)w^*(t)dt + \sum_{t \in T_u(\tau)} d_h(t)u(t) = g, \\ u^*(\tau) \le u(\tau) \le u^*(\tau+h), \quad |u(t)| \le 1, \quad t \in T_u(\tau+h). \end{gathered} \tag{9.6}$$

Setting

$$g(\tau) = g - y^*(\tau),$$

we rewrite (9.6) that

$$\sum_{t \in T_u(\tau)} c_h(t)u(t) \longrightarrow \max, \quad \sum_{t \in T_u(\tau)} d_h(t)u(t) = \bar{g}(\tau+h),$$
$$u^*(\tau) \le u(\tau) \le u^*(\tau+h), \quad |u(t)| \le 1, \quad t \in T_u(\tau+h), \tag{9.7}$$

where

$$\bar{g}(\tau+h) = g(\tau) + \Delta g(\tau), \quad \Delta g(\tau) = -\int_{\tau}^{\tau+h} G(t)w^*(t)dt. \tag{9.8}$$

Equation

$$x^*(\tau+h) = F(\tau+h)F^{-1}(\tau)x^*(\tau) + \int_{\tau}^{\tau+h} F(\tau+h)F^{-1}(t)b(t)dtu^*(\tau) + \int_{\tau}^{\tau+h} F(\tau+h)F^{-1}(t)w^*(t)dt$$

entails

$$\int_{\tau}^{\tau+h} F^{-1}(t)w^*(t)dt = F^{-1}(\tau+h)x^*(\tau+h) - F^{-1}(\tau)x^*(\tau) - \int_{\tau}^{\tau+h} F^{-1}(t)b(t)dt\, u^*(\tau).$$

Combining this with (9.8), we obtain

$$\Delta g(\tau) = y^*(\tau) + d_h(\tau)u^*(\tau) - y^*(\tau+h). \tag{9.9}$$

To determine $\bar{g}(\tau+h)$, we integrate (2.4) over $[\tau, \tau+h]$ with $G(\tau)$ as an initial condition and calculate and store $G(\tau+h)$ and $y^*(\tau+h)$. The resulting (9.8) and (9.9) can be used to correct $T^0_{\text{supp}}(\tau)$, construct $T^0_{\text{supp}}(\tau + h)$, and calculate $u^*(\tau+h)$ at $\tau+h$.

The following two situations may arise at $\tau + h$ (after the current state $x^*(\tau + h)$) is measured): (1) $\Delta g(\tau) = 0$; (2) $\Delta g(\tau) \ne 0$.

Let us analyze case (1). It includes the following subcases: (1a) $\tau \notin T^0_{\text{supp}}(\tau)$; (1b) $\tau \in T^0_{\text{supp}}(\tau)$.

In case (1a), we set $T^0_{\text{supp}}(\tau+h) = T^0_{\text{supp}}(\tau)$ and

$$u^*(\tau+h) = \begin{cases} \gamma(\tau), & \text{if } \tau+h \notin T_{\text{sn}}(\tau), \\ \omega(\tau+h), & \text{if } \tau+h \in T^0_{\text{supp}}(\tau), \\ -\gamma(\tau), & \text{if } \tau+h \in T_{\text{n0}}(\tau). \end{cases}$$

The pseudocontrol $\omega(\tau+h)$ can be obtained by solving the equation

$$D_{\text{supp}}(\tau)\omega_{\text{supp}} = \bar{g}(\tau+h) - p(\tau) \tag{9.10}$$

or reading from the computer memory as $\omega(\tau+h) = u^0_\tau(\tau+h)$ if the support values $u^0_\tau(t)$, $t \in T^0_{\text{supp}}(\tau)$ are stored.

The stored data corresponding to the time τ are transformed into data for $\tau+h$ by setting

$$\gamma(\tau+h) = \begin{cases} \gamma(\tau), & \text{if } \tau+h \notin T_{\text{sn}}(\tau), \\ -\gamma(\tau), & \text{if } \tau+h \in T_{\text{sn}}(\tau), \end{cases}$$

$p(\tau+h) = p(\tau) - \gamma d_h(\tau)$, $T_{\text{n0}}(\tau+h) = T_{\text{n0}}(\tau)$, $D_{\text{supp}}(\tau+h) = D_{\text{supp}}(\tau)$, $\nu(\tau+h) = \nu(\tau)$, $T_{\text{sn}}(\tau+h) = T_{\text{sn}}(\tau)\backslash\tau \cup \tau+h$ and $k^*(\tau+h) = k^*(\tau)$ if $\tau+h \notin T_{\text{sn}}(\tau)$, and $T_{\text{sn}}(\tau+h) = T_{\text{sn}}(\tau)\backslash\tau$ and $k^*(\tau+h) = k^*(\tau) - 1$ if $\tau+h \in T_{\text{sn}}(\tau)$. The values of $G(\tau+h)$, $\psi_c(\tau+h)$, and $d_h(\tau+h)$ are calculated and stored instead of $G(\tau)$, $\psi_c(\tau)$, and $d_h(\tau)$.

In case (1b), $T^0_{\text{supp}}(\tau + h)$ is constructed by analyzing problem (9.6). Choosing $T^0_{\text{supp}}(\tau)$ as an initial support $T^0_{\text{supp}}(\tau + h)$, we invoke a dual method from Section 7 to remove τ from the support. This can be done by using $\Delta\delta_h(\tau) = \pm 1$. The support obtained by iterating the dual method is the required $T^0_{\text{supp}}(\tau + h)$. Using the results of iteration, we rewrite the data corresponding to $\tau + h$ and set

$$u^*(\tau+h) = \begin{cases} \gamma(\tau+h), & \text{if } \tau+h \notin T^0_{\text{supp}}(\tau+h), \\ \omega(\tau+h), & \text{if } \tau+h \in T^0_{\text{supp}}(\tau+h). \end{cases}$$

Consider case (2). Solving equation (9.10), we find the vector $\omega_{\text{supp}} = (\omega(t), t \in T^0_{\text{supp}}(\tau))$. The following situations may arise: (2a) it holds that

$$|\omega(t)| \le 1, \quad t \in T^0_{\text{supp}}(\tau); \tag{9.11}$$

(2b) inequalities (9.11) are violated.

In subcase (2a), we proceed as in case (1).

In subcase (2b), the optimal support $T^0_{\text{supp}}(\tau + h)$ of problem (9.5) is constructed by using any dual method described above. If $\tau \in T^0_{\text{supp}}(\tau)$, then τ is removed from the initial support $T_{\text{supp}}(\tau + h)$ "by brute force" at the first iteration step. In solving problem (9.7) with $\tau \notin T_{\text{supp}}(\tau + h)$, the support is transformed on the set $T_u(\tau + h)$, which prevents τ from entering $T^0_{\text{supp}}(\tau + h)$. When iterating the dual method, we may find that there does not exist any finite step σ resulting in a new zero of $\delta_h(t, \sigma)$, $t \in T_u(\tau + h)$. This means that problem (9.7) has no feasible controls; i.e., the current state $x^*(\tau + h)$ has left the set $X(\tau + h)$.

Frequent repetition of the situation when $\tau \in T^0_{\text{supp}}(\tau)$ means that a sliding mode has developed in system (9.3). Techniques designed for regularizing the sliding mode in the framework of our approach were described in [16].

An analysis of the operations involved in the dual methods described in Section 7 shows that modern microprocessors make it possible to implement optimal feedback controls of fairly high order to problem (2.1). Several estimates were presented in [15], and further information is contained in the results of optimization of a fourth-order dynamic system in the example considered in Section 10.

Finally, we describe a starting procedure for the optimal-controller operation algorithm. The following two situations are possible at the initial moment $\tau = t_*$: (i) an initial state $x(t_*) = x_0^*$ is known in advance (before the start of the control process); (ii) the initial state is not known exactly, but it is known to belong to a bounded set X_0.

In the former situation, the methods described in Sections 4–7 are used in advance (before the start of an actual control process) to construct an optimal support control $\{u^0(t|t_*, x_0^*), t \in T_u;\ T^0_{\text{supp}}(t_*)\}$ for problem (2.1). At this stage, the time required to solve problem (9.1) is not critically important. After such a preparation, the optimal controller sets $u^*(t_*) = u^0(t_*|t_*, x_0^*)$ at the initial moment $\tau = t_*$.

Suppose that the latter situation is the case. Then, the preparatory work include the following. The set X_0 is covered with a finite net with nodes at $x_0^1, \dots, x_0^q$. An optimal support control $\{u^0(t|t_*, x_0^p), t \in T_u;\ T^{0p}_{\text{supp}}\}$ is constructed for problem (9.1) at each x_0^p $(p = \overline{1, q})$. At the initial moment $\tau = t_*$, the optimal controller finds the node x_0^* nearest to the actual state $x_0^{p_*}$, and one of the dual methods described above is utilized to correct the optimal support control $\{u^0(t|t_*, x_0^{p_*}), t \in T_u;\ T^{0p_*}_{\text{supp}}\}$. The number q of nodes and their location are chosen so that the time required to correct $\{u^0(t|t_*, x_0^{p_*}), t \in T_u;\ T^{0p_*}_{\text{supp}}\}$ to the optimal support control $\{u^0(t|t_*, x_0^*), t \in T_u;\ T^0_{\text{supp}}(t_*)\}$ does not exceed $h > 0$.

The use of optimal support controls prepared in advance for the nodes of subsets of $X(\tau)$, $\tau \in T_u$ also makes it possible to construct realizations of a positional solution to problem (2.1) in cases when the disturbance is the sum of a bounded piecewise continuous component and an impulsive component (as the δ-function and its derivatives).

Remark 2. It is assumed here that $A(t)$ and $b(t)$, $t \in T$ are known in advance and do not change in the course of the control process. This assumption can be weakened by assuming that reliablly predicted values of $A(t|\tau)$ and $b(t|\tau)$ ($t \in T(\tau)$), rather than the functions themselves, are available at each current point in time $\tau \in T$. Then, it suffices to use $A(t|\tau)$ and $b(t|\tau)$, $t \in T(\tau)$ instead of $A(t)$ and $b(t)$, $t \in T(\tau)$. Analogously, given a reliable prediction $w(t|\tau)$, $t \in T(\tau)$ for the disturbance at τ, one can use it instead of $w(t) \equiv 0$, $t \in T(\tau)$ in calculating $u^*(\tau)$. Such modifications are quite consistent with the classical approach to the optimal feedback control problem. Modern formulations are characterized by a more accurate description of uncertainties, but these interesting issues are beyond the scope of this paper.

10. EXAMPLE

To illustrate some of the results obtained here, we consider the problem of damping a two-mass oscillator in a given time with minimum fuel consumption (Fig. 1). The mathematical model of the problem has the form

$$\int_0^{25} u(t)dt \longrightarrow \min, \quad \dot{x}_1 = x_3, \quad \dot{x}_2 = x_4,$$

$$\dot{x}_3 = -x_1 + x_2 + u, \quad \dot{x}_4 = 0.1x_1 - 1.02x_2, \quad x_1(0) = x_2(0) = 0, \quad x_3(0) = 2, \quad x_4(0) = 1, \qquad (10.1)$$

$$x_1(25) = x_2(25) = x_3(25) = x_4(25) = 0, \quad 0 \le u(t) \le 1, \quad t \in [0, 25[,$$

where $x_1 = x_1(t)$ and $x_2 = x_2(t)$ are the respective deviations of the first and second masses from their equilibrium positions; $x_3 = dx_1/dt$, $x_4 = dx_2/dt$; and $u = u(t)$ is the fuel consumption per second at time t.

Reducing problem (10.1) to canonical form (2.1), we obtain a fifth-order optimal control problem. In the class of discrete-time controls with the quantization period $h = 25/1000 = 0.025$, problem (10.1) is equivalent to an LP problem of size 4×1000.

We used the method with elementary support replacement described in Section 7 to construct an optimal open-loop control of problem (10.1). The initial support was the set $T_{\text{supp}} = \{5, 10, 15, 20\}$ of times spaced uniformly along the control segment $T = [0, 25]$. This support is associated with a four-element set of nonsupport zeros of cocontrol: $T_{\text{n0}} = \{3.725, 9.725, 15.3, 21.3\}$. The problem was solved in 12 iteration steps; i.e., the 4×4 support matrix was changed 12 times during the construction of an optimal open-loop control. The optimal value of the performance criterion was found to be 6.331252. The movements of $t \in T_{\text{sn}}$ during the iteration process are shown in Fig. 2. The times t_5 and t_6 joined together and disappeared during the first iteration step. The number near each node of a curve is the displacement of the corresponding time (in units

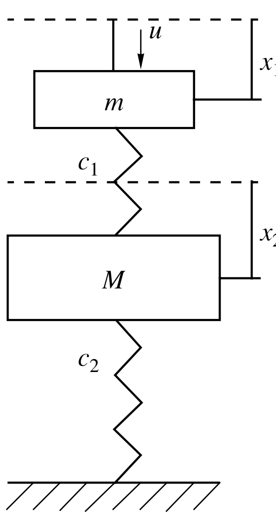

Fig. 1.

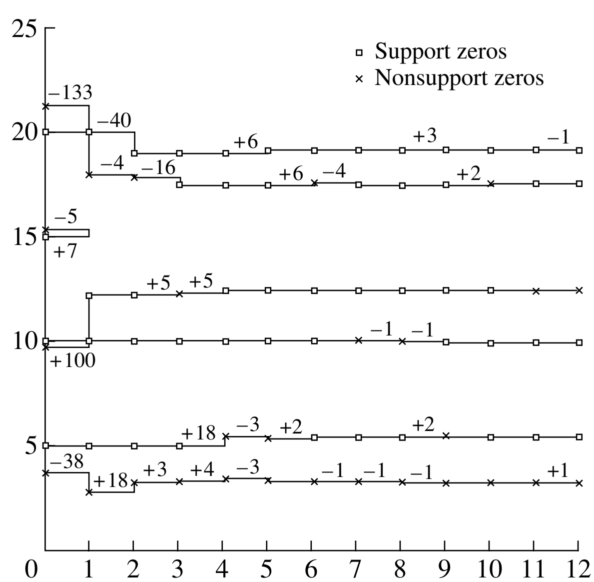

Fig. 2.

Table

h	Number of iteration steps	Performance criterion	Complexity	
			efficient method	fast method
0.25	9	6.353339	2.64	2.41
0.0025	15	6.330941	2.4149	2.2018
0.001	20	6.330938	2.3912	2.23564

of h) during the iteration process. The figure shows that the displacements of the times in T_{n0} become negligible after the fourth iteration step. This means that even the quasi-controls constructed after the fourth iteration step generate trajectories that satisfy the terminal constraints of problem (10.1) with a high accuracy. For example, the norm of the residual of the terminal constraints for the quasi-control associated with the support is 0.205, 0.080, 0.028, and 0.01 after the fourth, sixth, eighth, and tenth iteration steps, respectively. The norm of the residual of the terminal constraints for the quasi-control associated with the initial support equals 1.077. For applied problems with approximate fulfillment of terminal constraints, this property of the method makes it possible to terminate the process of open-loop optimization or the process of correcting supports in an optimal synthesis after a few iteration steps.

These data give a glimpse of the efficiency of the method. In our opinion, the time required for constructing optimal open-loop controls is not important at present, when very powerful computers are available. It is important that the method employed produce a reliable solution in a reasonable time. However, as is customary, we present relevant calculations. First, we choose a characteristic to be compared. It is not always reasonable to compare the numbers of iteration steps required in different methods, because the iteration steps of various methods widely differ in their content. A more natural characteristic for determining the efficiency of a method is the number of integrations of its primal or dual system in the case of limited main memory resources (see [5]). Accordingly, we choose the unit of complexity to be the time required for integration of the primal or dual system over the entire control segment T. When the method can be parallelized, its complexity is defined as the time required to solve a problem on a set of processors. This characterization of methods for problem (2.1) is not absolute (precise), because it ignores the extra time required to execute other necessary operations (in addition to integration). However, it can be used to assess methods as a first approximation.

The method admits efficient and fast implementations. In either case, $m + 1$ processors were used to prepare the data for the first iteration step. Each processor performed two integrations of the dual system over the entire segment T (one integration for computing the support matrix D_{supp} and the vector of potentials ν and the other for computing a cocontrol). In the efficient implementation, $m + 1$ processors were used to solve the problem by iteration, and the time (in units of h) required to do this was calculated as the sum of displacements of all moving zeros at all iteration steps. In our example, this sum was found to be $433h$. Consequently, the complexity of the efficient implementation is 2.433. When the problem was solved by the fast procedure, $r(m + 1)$ processors (with r being the number of moving zeros) and an extra main memory space were used. In parallel with the motion of a zero, the dual system was integrated in the direction of possible motion for the other moving zeros, and $\psi_c(t)$, $G(t)$, and $\sigma(t)$ were stored for a set of consecutive nodes. To calculate the time required for iteration, we determined the moment corresponding to the maximum displacement at each iteration step. The sum of these numbers was found to be $231h$. Thus, the complexity of the fast implementation in our example is 2.231.

The table lists some characteristics concerning the solution of problem (10.1) for other quantization periods.

Remark 3. The minimum complexity associated with an open-loop solution to problem (2.1) is equal to unity. This is the complexity of checking a feasible control $u(t)$, $t \in T_u$ for optimality: if ν is an optimal Lagrange vector, then, by the maximum principle, it is sufficient to integrate dual system (2.3) once from right to left starting from $\psi(t^*) = c - H'\nu$ and to check inequalities (3.9).

Minimum complexity "can" also be achieved in constructing an optimal open-loop control for problem (2.1) when it is reduced to (2.2) and is solved by LP methods. In this case, it is sufficient to integrate the system once in order to construct an input matrix for an LP problem. However, such a static approach requires large main memory resources and differs fundamentally from the conventional approaches based on dynamic models of (2.1) requiring small amounts of stored data. When the required amount of data is

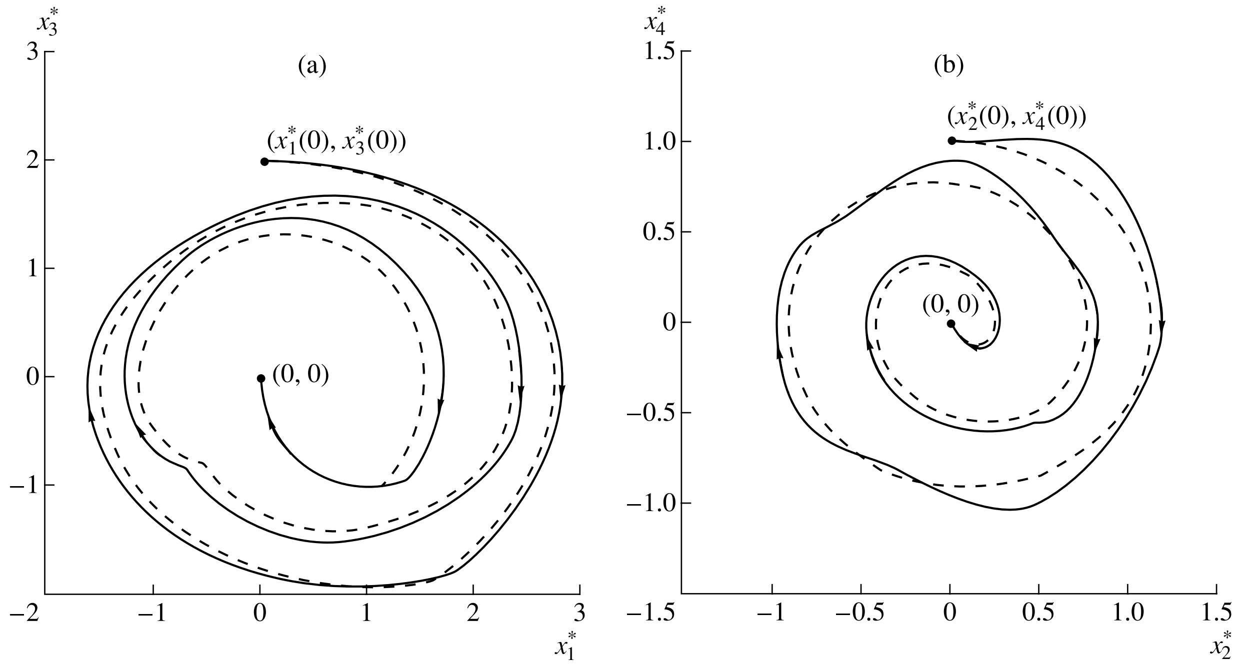

Fig. 3.

large, the search for data is comparable to the integration of the primal or dual system in terms of time, which substantially distorts the concept of complexity introduced above.

Elements of the dynamic approach to LP problems similar to (2.2) can be traced in the well-known column generation method, in which the data required for the simplex method are obtained by manipulating the original information (in small amount).

As far as we know, all numerical optimal control methods based on Pontryagin's maximum principle have a considerably higher complexity.

Let us construct a closed-loop solution to problem (10.1). Assume that the initial state is known in advance (as mentioned above, the general case does not introduce radical changes), and a bounded piecewise continuous disturbance acts on the oscillating system during the control process. Then, the behavior of the system is governed by the equations

$$\dot{x}_1 = x_3, \quad \dot{x}_2 = x_4, \quad \dot{x}_3 = -x_1 + x_2 + u, \quad \dot{x}_4 = 0.1x_1 - 1.02x_2 + w.$$

Assume that the disturbance has the form

$$w^*(t) = 0.3\sin 4t, \quad t \in [0, 9.75[, \quad w^*(t) \equiv 0, \quad t \geq 9.75,$$

which is not known to the optimal controller, and the current state $x^*(\tau)$ is known at each moment $\tau \in T_u$.

Figure 3 shows the projections of transition processes in system (10.1) with an optimal feedback control onto the planes x_1x_3 and x_2x_4. Dashed curves represent optimal open-loop trajectories, and solid curves are trajectories obtained under the action of $u^*(\tau)$ ($\tau \in T_u$) and $w^*(\tau)$ ($\tau \in T$). A realization of $u^*(t)$, $t \in T$ is shown in Fig. 4. The actual values of the support control are shown by crosses. The behavior of zeros $t_i(\tau) \in T_{\text{sn}}(\tau)$ ($\tau \in T_u$) is shown in Fig. 5. The graph demonstrates that new elements of $T_{\text{n0}}(\tau)$ repeatedly appeared for a short time and then disappeared during the control process. Figure 6 shows the complexity of correction of current supports by the fast dual method at each $\tau \in T_u$. Clearly, this task can be accomplished in the time $h = 0.025$ by many state-

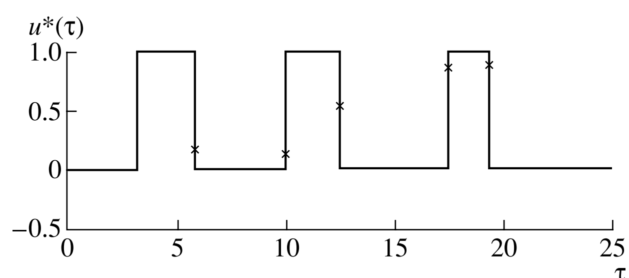

Fig. 4.

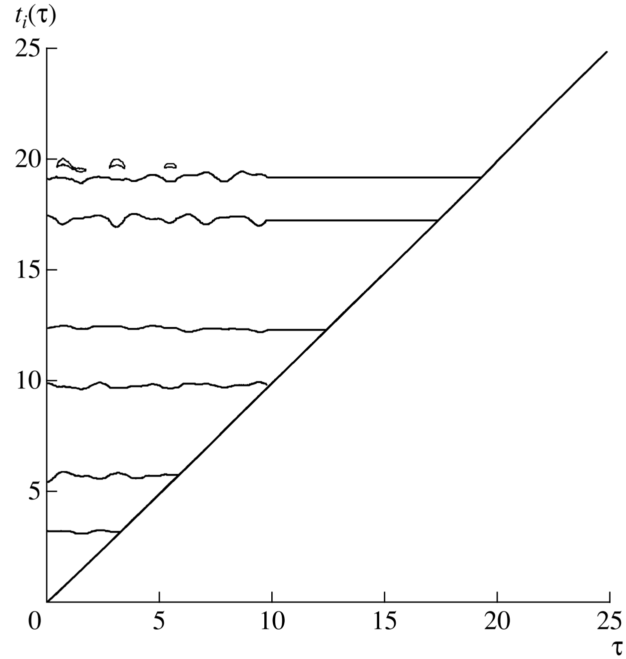

Fig. 5.

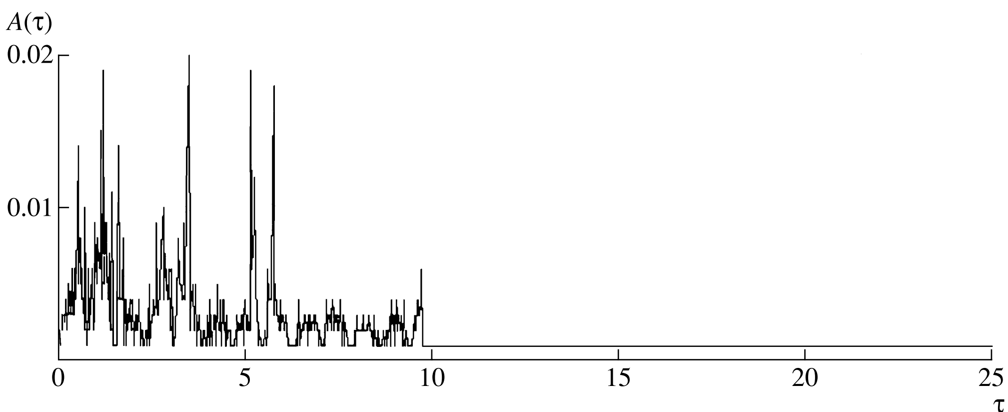

Fig. 6.

of-the-art microprocessors if real time in problem (10.1) is measured, for example, in seconds.

Remark 4. When classical dynamic programming methods (see [12]) are employed to synthesize optimal feedback controls in problem (10.1), the complexity of the correction becomes zero; i.e., no microprocessors are needed. Simultaneously, we have no analytical difficulty associated with justifying the Bellman equation, because it is easy to show that this equation is a recurrence in the class of discrete-time controls and can easily be solved. Let χ be the required accuracy of a solution to problem (10.1). A function $\tilde{u}(t, x)$ $(x \in X(t), t \in T_u)$ is said to be a χ-approximate positional solution to the problem if the following conditions are fulfilled: (i) $0 \le \tilde{u}(t, x) \le 1$, $x \in X(t)$, $t \in T_u$; (ii) $\|H(\tilde{x}(t^*) - x^0(t^*))\| \le \chi$; and (iii) $\int_0^{t^*} |u^0(t, x^0(t)) - \tilde{u}(t, \tilde{x}(t))| dt \le \chi$. The set $X(t)$, $t \in T_u$ is covered with finite χ_1-nets $G_\chi(t)$ $(t \in T_u)$ such that the union of the χ_1-neighborhoods of all nodes in $G_\chi(t)$ includes $X(t)$, and the optimal controls $u^0(t, x)$ $(x \in G_\chi(t), t \in T_u)$ constructed for the net nodes are χ-approximate solutions to problem (10.1) in their χ_1-neighborhoods of nodes. If $u^0(t, x)$ $(x \in G_\chi(t), t \in T_u)$ are preliminarily computed and tabulated, then, in the course of an actual control process, it is sufficient to find the node $\check{x}(\tau) \in G_\chi(\tau)$ closest to the current state $x^*(\tau)$ at $\tau \in T_u$ and to use the control $\tilde{u}(\tau, x^*(\tau)) = u^0(\tau, \check{x}(\tau))$. One disadvantage of this method for synthesis of optimal controls is well known. Bellman called it the curse of dimensionality: when $\chi > 0$ is small, the main memory required for $u^0(t, x)$ ($x \in G_\chi(t)$, $t \in T_u$) becomes huge even for small $n \geq 3$. The use of other types of memory leads to an abrupt increase in the time required to read relevant information.

Finite nets have also been used in the methods suggested here for general optimal synthesis. However, first, they were constructed only for a priori unknown initial states, and, second, they had a considerably smaller number of nodes, because the required solution accuracy was rapidly achieved after rough initial approximations had been corrected.

ACKNOWLEDGMENTS

This work was conducted under the State program for fundamental research "Algorithm 20" and was also supported by the Belarussian Republican Foundation for Basic Research, project no. F98-014.

REFERENCES

1. Krasovskii, N.N., *Teoriya upravleniya dvizheniem* (Theory of Motion Control), Moscow: Nauka, 1968.
2. Gamkrelidze, R.V., On the Theory of Optimal Processes in Linear Systems, *Dokl. Akad. Nauk SSSR*, 1957, vol. 116, no. 1, pp. 9–11.
3. Dantzig, G.B., *Linear Programming and Extensions*, Princeton (New Jersey): Princeton Univ. Press, 1966. Translated under the title *Lineinoe programmirovanie, ego primeneniya i obobshcheniya*, Moscow: Progress, 1966.
4. Dantzig, G.B., Linear Optimal Control Processes and Mathematical Programming, *SIAM J. Control*, 1966, vol. 4, no. 1, pp. 56–60.
5. Fedorenko, R.P., *Priblizhennoe reshenie zadach optimal'nogo upravleniya* (Approximate Solution of Optimal Control Problems), Moscow: Nauka, 1978.
6. Propoi, A.I., *Elementy teorii optimal'nykh diskretnykh protsessov* (Elements of the Theory of Optimal Discrete Processes), Moscow: Nauka, 1973.
7. Gabasov, R. and Kirillova, F.M., *Metody lineinogo programmirovaniya. Ch. 3. Spetsial'nye zadachi* (Linear Programming Methods, Part 3: Special Problems), Minsk: Bel. Gos. Univ., 1980.
8. Gabasov, R. and Kirillova, F.M., *Konstruktivnye metody optimizatsii. Ch. 2. Zadachi upravleniya* (Constructive Optimization Methods, Part 2: Control Problems), Minsk: Universitetskoe, 1984.
9. Gabasov, R., Kirillova, F.M., Kostyukova, O.I., and Pokataev A.V. *Konstruktivnye metody optimizatsii. Ch. 5. Nelineinye zadachi* (Constructive Optimization Methods, Part 5: Nonlinear Problems), Minsk: Universitetskoe, 1998.
10. Gabasov, R. and Kirillova, F.M., *Optimizatsiya lineinykh sistem* (Optimization of Linear Systems), Minsk: Belarus. Gos. Univ., 1973.
11. Gabasov, R., Kirillova, F.M., and Tyatyushkin, A.I., *Konstruktivnye metody optimizatsii. Ch. 1. Lineinye zadachi* (Constructive Optimization Methods, Part 1: Linear Problems), Minsk: Universitetskoe, 1984.
12. Bellman, R., *Dynamic Programming*, Princeton (New Jersey): Princeton Univ. Press, 1957. Translated under the title *Dinamicheskoe programmirovanie,* Moscow: Inostr. Literatura, 1960.
13. Gabasov, R., Kirillova, F.M., and Kostyukova, O.I., Construction of Feedback Optimal Controls in a Linear Problem, *Dokl. Akad. Nauk SSSR*, 1991, vol. 320, no. 6, pp. 1294–1299.
14. Gabasov, R., Kirillova, F.M., and Prischepova, S.V., *Optimal Feedback Control*, London: Springer, 1995.
15. Balashevich, N.V., Gabasov, R., and Kirillova, F.M., Optimal Closed-Loop Moving Control of Linear Dynamic Objects, *Izv. Ross. Akad. Nauk, Teoriya Sistemy Upravleniya*, 1998, no. 3, pp. 40–47.
16. Balashevich, N.V., Gabasov, R., and Kirillova, F.M., Suboptimal Controller Regulator for Smoothing Controls and Filtering High-Frequency Disturbances on Sliding-Mode Intervals, *Izv. Ross. Akad. Nauk, Tekhn. Kibern.,* 1993, no. 6, pp. 25–32.