ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2023, Vol. 84, No. 9, pp. 1065–1074.

Static Feedback Design in Linear Discrete-Time Control Systems Based on Training Examples

V. A. Mozzhechkov

Tula State University, Tula, Russia
e-mail: v.a.moz@yandex.ru

Received November 14, 2022
Revised June 21, 2023
Accepted July 20, 2023

Abstract—The problem of static feedback design in linear discrete time-invariant control systems is considered. The desired behavior of the system is defined by a set of its output variation laws (training examples) and by a requirement to the degree of its stability. Controller's structural constraints are taken into account. Explicit relations are obtained and an iterative method based on these relations is proposed to find a good initial approximation of the desired gain matrix and to refine it sequentially. In the general case, simple-structure gain matrices are found: in such matrices, only those components are nonzero that are necessary and sufficient to give the system the desired properties. Some examples are provided to illustrate the method.

Keywords: linear discrete-time control systems, feedback, design

DOI: 10.25728/arcRAS.2023.79.83.001

1. INTRODUCTION

A considerable number of works are devoted to the design of static feedback in linear control systems. As a rule, the desired behavior of the system is defined by requiring that the roots of its characteristic polynomial belong to some value set or by minimizing an integral quadratic functional that assesses the quality of transients. Accordingly, the problems under consideration are placing the poles of the transfer function of a closed loop system (modal control) and designing a linear quadratic controller (LQR). There exist [1] effective methods for solving them exactly provided that all components of the state vector can be used in the controller and no explicit constraints are imposed on the choice of gain coefficients. However, these problems turn out to be intractable in the case of controller's structural constraints [2, 3], particularly under the unavailability of some state variables (e.g., when designing output feedback). In such a case, pole placement is an NP-hard problem [2, 4] that often reduces to a nonsmooth and nonconvex optimization problem in the space of controller's parameters [2, 5]. Necessary and sufficient conditions for the existence of a solution were established for this problem [6–9], but it was not possible to develop methods for obtaining an exact solution [2, 3]. At the same time, algorithms were proposed to calculate an approximate solution. A significant part of them involve Lyapunov functions for the design of stabilizing controllers and the reduction of the original problem to nonlinear matrix inequalities by repeatedly solving linear matrix inequalities (LMIs) during iterative refinement of the desired solution [9–13]. The papers [14–16] investigated the possibility of using the LMI technique to consider the sparse feedback design requirements that limit freedom in choosing the controller structure. Along with the ones mentioned above, algorithms were proposed to design stabilizing output-feedback controllers by minimizing the spectral abscissa of a closed loop system by its direct calculation and solving the corresponding nonlinear programming problem based on methods that take into account the peculiarities of the design problem [2, 3]. The algorithms presented in [8, 17, 18] involve external algebra methods to find an initial approximation of the desired output-feedback gain matrix for the modal control problem; this approximation is then refined iteratively. For the LQR problem with output feedback, necessary conditions for the existence of a solution in the form of a system of nonlinear matrix equations were obtained [19, 20] and corresponding iterative algorithms for the approximate solution of this problem were proposed [20–24]. Numerical methods for solving the LQR problem with a sparse feedback matrix based on the LMI technique were considered in [14–16, 25]; for the first time, such a problem was solved in [26] by reducing to a nonlinear discrete programming problem. However, the algorithms mentioned do not ensure an exact solution and are heuristic: their convergence was not proved rigorously.

The problem considered below essentially differs from classical static feedback design problems as follows. The desired behavior of the system is defined by a set of its output variation laws, acting as training examples. They can be trajectories corresponding, e.g., to a feedback control law that should be simplified using a simpler controller in the designed system (in particular, a system with state feedback can be a source of training examples for output feedback design) or to a program control law or human control that should be implemented in the designed system based on a feedback control law. Together with the closeness of the system trajectories to the trajectories given as training examples, the requirement to ensure a given degree of its stability is considered. In addition, the constraints imposed on the feedback structure are taken into account. They can be expressed as the requirement to use output feedback, the requirement that some elements of the gain matrix be zero, and the requirement to eliminate its structural redundancy. The latter is equivalent to obtaining a simple-structure gain matrix [27–31]: in such matrices, only those components are nonzero that are necessary and sufficient to give the system the desired properties. The goal of design is to approximate the system behavior to the desired one by choosing the elements and structure of the gain matrix. This problem statement is novel and has not been considered in the works devoted to controller design, including those involving machine learning methods [32–35].

In this paper, we derive explicit relations and propose a corresponding iterative method to find a good initial approximation of the desired gain matrix and to refine it sequentially. The novel method allows designing all possible simple-structure gain matrices.

2. PROBLEM STATEMENT

Consider a control system described by the equations

$$x_{k+1} = Ax_k + Bu_k, \tag{1}$$

$$y_k = Cx_k, \tag{2}$$

$$u_k = Ky_k, \tag{3}$$

where k denotes discrete time from the set of natural numbers; x_k, y_k, and u_k are the state, output, and control vectors, respectively; the components of the vectors x_k, y_k, and u_k as well as the elements of constant matrices A, B, C, and K are real numbers; the controller's gain matrix K has to be determined, the other matrices are supposed given.

Consider structural constraints imposed on the controller (3). They are usually reduced [2, 14, 15, 26] to requiring zero value for some elements of the gain matrix $K = (k_{i,j})$. Therefore, we introduce the condition

$$k_{i,j} = 0,\ \forall (i,j) \notin \check{S}, \tag{4}$$

where $\check{S}$ is the set of number pairs (i, j) for the elements of the gain matrix K that are not required to be zero.

We define the desired behavior of system (1)–(4) by specifying the corresponding desired trajectories $Y_\gamma = (y_k^\gamma)$, $k \in \{\overline{1,N}\}$, of the output (2) of system (1)–(4) for some set of initial conditions x_0^γ, $\gamma \in \{\overline{1,q}\}$.

In other words, we define a set

$$Q = \{(x_0^\gamma, Y_\gamma)\}, \quad \gamma \in \{\overline{1,q}\}, \tag{5}$$

in which the pairs (x_0^γ, Y_γ) are training examples.

In system (1)–(4), perfectly matching the desired behavior, the equality $y(x_0^\gamma, K)_k = y_k^\gamma$ holds for the initial conditions $x(0) = x_0^\gamma$ at each time instant $k \in \{\overline{1,N}\}$. Let us require this condition for each pair $(x_0^\gamma, Y_\gamma) \in Q$, i.e.,

$$y(x_0^\gamma, K)_k = y_k^\gamma, \ \forall k \in \{\overline{1,N}\}, \ \forall \gamma \in \{\overline{1,q}\}. \tag{6}$$

The possibility that (6) is satisfied approximately will be described as follows:

$$\varepsilon_k^{\gamma-} \leq y(x_0^\gamma, K)_k - y_k^\gamma \leq \varepsilon_k^{\gamma+}, \ \forall k \in \{\overline{1,N}\}, \ \forall \gamma \in \{\overline{1,q}\}, \tag{7}$$

where $\varepsilon_k^{\gamma-}$ and $\varepsilon_k^{\gamma+}$ are given constant vectors.

Generally, conditions (7) do not ensure the stability of system (1)–(4). Therefore, together with (7), we require the necessary degree of Schur stability for the matrix $A_c = A + BKC$ of the closed loop system (1)–(4), i.e.,

$$\rho(A_c(K)) \leq 1 - \sigma, \tag{8}$$

where $\rho(A_c(K))$ denotes the spectral radius of the matrix $A_c(K)$ and σ is a given degree of stability.

Let the matrix K be chosen through the best approximation of the behavior of system (1)–(4) to the desired one by minimizing the Euclidean norm of the vector $\Delta y(K)$ composed of the residuals $y(x_0^\gamma, K)_k - y_k^\gamma$ of all equations (6):

$$|\Delta y(K)| \to \min_K . \tag{9}$$

In the case of a given structure of the controller (a fixed set $\check{S}$ defining its structure), the problem under consideration is to find the matrix K in system (1)–(4) that satisfies the requirements (7)–(9).

In general, we will solve the structural design problem: determine all sets $\check{S}$ and the corresponding matrices K for which conditions (7)–(9) hold and the structure of the controller (3), (4) is simple. This means [27–31] that only those components of the matrix K are nonzero that are necessary and sufficient to give system (1)–(4) the desired properties. Formally, the problem of determining a set Ω of simple structures of the controller (3), (4) consists in the following: find admissible structures $\check{S} \in \zeta$ for which a less complex admissible structure cannot be specified. (A structure $\check{S}'$ is considered simpler than $\check{S}$ if $\check{S}' \subset \check{S}$.) In other words, it is required to find

$$\Omega = \left\{ \check{S} \in \zeta \mid \{\check{S}' \in \zeta \mid \check{S}' \subset \check{S}\} = \varnothing \right\}, \tag{10}$$

where ζ denotes the set of admissible structures, i.e., those for which there exists a matrix K satisfying conditions (1)–(4) and (7)–(9). The formula $\{\check{S}' \in \zeta \mid \check{S}' \subset \check{S}\} = \varnothing$ indicates the absence of an admissible structure $\check{S}'$ simpler than a structure $\check{S} \in \Omega$.

3. ANALYSIS OF THE PROBLEM

Given x_0^γ and K, the solution of system (1)–(3) can be written [1, p. 20] as follows:

$$y(x_0^\gamma, K)_k = CA^k x_0^\gamma + C\sum_{i=0}^{k-1} A^{k-i-1}BKy(x_0^\gamma, K)_i,\ \forall k \in \{\overline{1, N}\}. \tag{11}$$

In view of (11), condition (6) is equivalent to the system of equations

$$CA^k x_0^\gamma + C\sum_{i=0}^{k-1} A^{k-i-1}BKy(x_0^\gamma, K)_i = y_k^\gamma,\ \forall k \in \{\overline{1, N}\},\ \forall \gamma \in \{\overline{1, q}\}. \tag{12}$$

Applying identity transformations yields the system

$$CA^k x_0^\gamma + C\sum_{i=0}^{k-1} \Big(y(x_0^\gamma, K)_i^{\mathrm{T}} \otimes A^{k-i-1}B\Big)\operatorname{vec}(K) = y_k^\gamma,\ \ \forall k \in \{\overline{1, N}\},\ \forall \gamma \in \{\overline{1, q}\}, \tag{13}$$

where $\otimes$ denotes the Kronecker product [36, p. 83] and $\operatorname{vec}(\cdot)$ is the vectorization function [36]. (It produces a column vector by the successive connection of all columns of the argument matrix.) We write system (13) as

$$Y_{0\gamma} + G_\gamma(K)\operatorname{vec}(K) = Y_\gamma,\ \forall \gamma \in \{\overline{1, q}\}, \tag{14}$$

where $Y_{0\gamma}$, Y_γ, and $G_\gamma(K)$ are the column vectors composed of the blocks $CA^k x_0^\gamma$, y_k^γ, and $G_{k\gamma}(K) = C\sum_{i=0}^{k-1}\Big(y(x_0^\gamma, K)_i^{\mathrm{T}} \otimes A^{k-i-1}B\Big)$, respectively, $k \in \{\overline{1, N}\}$.

From (14) and (4) it follows that

$$G_\gamma(K)_S\operatorname{vec}(K)_S = \hat{Y}_\gamma,\ \forall \gamma \in \{\overline{1, q}\}, \tag{15}$$

where the matrix $G_\gamma(K)_S$ and the vector $\operatorname{vec}(K)_S$ contain the columns of the matrix $G_\gamma(K)$ and the coordinates of the vector $\operatorname{vec}(K)$, respectively, whose numbers are specified in the set S. (In accordance with the set $\check{S}$, the former set determines the numbers of the coordinates of the vector $\operatorname{vec}(K)$ that are not required to be zero.) In addition, $\hat{Y}_\gamma = Y_\gamma - Y_{0\gamma}$.

Let all the desired trajectories $Y_\gamma = (y_k^\gamma)$, $k \in \{\overline{1, N}\}$, $\gamma \in \{\overline{1, q}\}$, belong to the set of solutions of system (1)–(4). Then $y(x_0^\gamma, K)_i$ in the expressions (12), (13) can be replaced by y_i^γ; as a result, the matrix $G_\gamma(K)$ in (15) becomes constant and independent of the desired unknown matrix K. In this case, system (15) can be represented as

$$\bar{G}_{\gamma S}\operatorname{vec}(K)_S = \hat{Y}_\gamma,\ \forall \gamma \in \{\overline{1, q}\}, \tag{16}$$

where $\bar{G}_\gamma$ is the column vector of the blocks $\bar{G}_{k\gamma} = C\sum_{i=0}^{k-1}\Big({y^\gamma}_i^{\mathrm{T}} \otimes A^{k-i-1}B\Big)$, $k \in \{\overline{1, N}\}$.

Proposition 1. *System* (1)–(4) *perfectly matches the desired behavior given by the set of training examples* (5)*, i.e., the requirement* (6) *holds, if and only if all the desired trajectories* Y_γ, $\gamma \in \{\overline{1, q}\}$, *belong to the set of solutions of system* (1)–(4) *and the matrix* K *given* (4) *is the solution of the system of linear equations* (16)*.*

The proof of Proposition 1 is postponed to the Appendix.

According to Proposition 1, the feasibility of system (16) is a necessary and sufficient condition for equalities (6), i.e., a condition for the exact reproduction of all training examples by the designed system.

Due to the equivalence of equations (6) and (15), conditions (9) and (7) are equivalent to the requirements

$$\sum_{\gamma=1}^{q} \mid G_\gamma(K)_S \mathrm{vec}(K)_S - \hat{Y}_\gamma \mid^2 \to \min_K, \tag{17}$$

$$\hat{Y}_\gamma + \varepsilon_\gamma^- \le G_\gamma(K)_S \mathrm{vec}(K)_S \le \hat{Y}_\gamma + \varepsilon_\gamma^+, \quad \gamma \in \{\overline{1,q}\}. \tag{18}$$

Proposition 2. *The behavior of system* (1)–(4) *best approximates the desired one specified by the set of training examples* (5)*, i.e., the requirements* (7)–(9) *hold, if and only if the matrix K given* (4) *is the solution of the nonlinear least-squares problem* (17) *with the constraints* (18) *and* (8)*.*

The proof of Proposition 2 is given in the Appendix.

4. THE SOLUTION METHOD

4.1. Solution of the Problem with a Given Controller Structure

Assume that the controller has a given structure, i.e., the set $\check{S}$ is specified. The desired matrix K corresponding to conditions (4) and (7)–(9) can be determined by solving problem (4), (17), (18), (8) (see Proposition 2). It will be called the statical controller training (SCT) problem. The success in solving this problem will significantly depend on the choice of the initial approximation (on how close the initial values of the desired unknowns are to the solution).

The solution of system (16) is a good initial approximation in the SCT problem. In general, we can take its approximate solution, i.e., the matrix $\underline{K}$ for which the vector $\mathrm{vec}(\underline{K})_S$ minimizes the Euclidean norm of the difference between the left- and right-hand sides of system (16) (the normal pseudosolution)

$$\mathrm{vec}(\underline{K})_S = \bar{G}_S^+ \hat{Y}, \tag{19}$$

where $\bar{G}_S^+$ is the Moore–Penrose pseudoinverse of the matrix of system (16) and $\hat{Y}$ is the right-hand side of system (16).

The closeness of the matrix $\underline{K}$ to the desired solution can be argued as follows. Let conditions (7)–(8) be feasible and $\check{K}$ be the solution of the SCT problem. If the desired trajectories belong to the set of trajectories possible in system (1)–(4), by Proposition 1 the matrix $\check{K}$ will coincide with the solution of system (16), i.e., $\check{K} = \underline{K}$. A small discrepancy between the desired and possible trajectories leads to a small discrepancy between the matrices $\check{K}$ and $\underline{K}$ since small changes in the parameters of system (1)–(4) correspond to small changes in its solutions and vice versa. The feasibility of conditions (7)–(8) means the closeness of the desired and possible trajectories in system (1)–(4); hence, if the desired solution of the SCT problem exists, it will be close to $\underline{K}$. (Hereinafter, we estimate the closeness of matrices by the Frobenius norm.)

Generally speaking, the matrix $\underline{K}$ differs from the desired solution because its definition does not fully considers conditions (7)–(9). Therefore, using it as a starting point, we will find a solution corresponding to the entire set of requirements.

The efficiency of solving the SCT problem can be improved by taking into account its peculiarities. Note that this problem turns into a linear least-squares problem with linear constraints [37, p. 225] (hereinafter, the LSL problem) when replacing, first, $G_\gamma(K)_S$ in (17), (18) with a fixed matrix $G_\gamma(K^*)_S$ corresponding to the fixed matrix K^* and, second, the function $\rho(A_c(K))$ in (8) with its linear approximation near of K^*. Such a linearization procedure is acceptable when seeking a solution in a small neighborhood of the matrix K^*. Therefore, it is possible to approach the solution of the SCT problem sequentially at each search step by solving the LSL problem with the matrix K^* found at the previous step.

The algorithm for solving the SCT problem proposed in this paper includes the following stages.

1. Choose the normal pseudosolution of system (16) as an initial approximation of the desired vector $\mathrm{vec}(K)_S$.

2. Perform an iterative search for the solution. At the 0th iteration, take $\mathrm{vec}(K^{(0)})_S = \mathrm{vec}(\underline{K})_S$. (The iteration number is specified as the superscript in brackets.)

At each jth iteration, solve the LSL problem

$$\sum_{\gamma=1}^{q} \mid G_\gamma^{(j-1)}\alpha^{(j)} - \hat{Y}_\gamma \mid^2 \to \min_K, \tag{20}$$

$$\hat{Y}_\gamma + \varepsilon_\gamma^- \le G_\gamma^{(j-1)}\alpha^{(j)} \le \hat{Y}_\gamma + \varepsilon_\gamma^+, \quad \gamma \in \{\overline{1,q}\}, \tag{21}$$

$$r_0^{(j-1)} + r_1^{(j-1)}\alpha^{(j)} \le 1 - \sigma, \tag{22}$$

where $\alpha^{(j)} \equiv \mathrm{vec}(K^{(j)})_S$ is the vector of unknowns, $G_\gamma^{(j-1)}$ is the column composed of the blocks $G_{k\gamma}(K^{(j-1)}) = C\sum_{i=0}^{k-1}\left(y\left(x_0^\gamma, K^{(j-1)}\right)_i^{\mathrm{T}} \otimes A^{k-i-1}B\right)$, $k \in \{\overline{1,N}\}$, and $r_0^{(j-1)} + r_1^{(j-1)}\alpha^{(j)}$ is the linear approximation of the function $\rho(A_c(K))$ near $K^{(j-1)}$. Conditions (21), (22) may fail when solving the LSL problem (20)–(22). In this case, the search procedure is stopped with stating that the solution of the SCT problem could not be found (because it does not exist or the algorithm is not efficient enough).

3. The search procedure is successfully completed when the vector of unknowns $\alpha^* = \alpha^{(j)}$ satisfying conditions (21) and (22) is obtained and either the difference $|\,\alpha^{(j)} - \alpha^{(j-1)}\,|$ or the objective function (20) becomes small enough, or a given number of iterations is exhausted. Take the matrix $K = \mathrm{vec}_S^{-1}(\alpha^*)$ as the solution, where $\mathrm{vec}_S^{-1}(\cdot)$ is the inverse of the vectorization function. (Given (4), it reconstructs the matrix K from the argument vector.)

The method presented above is substantially similar to the Gauss–Newton iterative algorithm for solving the unconstrained nonlinear least-squares problem. At each iteration of this algorithm, Taylor's theorem is applied to linearize the objective function and solve the resulting linear least-squares problem. In contrast, the novel method essentially exploits the peculiarities of problem (17), (18), (8) and, consequently, requires no differentiation to linearize the objective function. For this purpose, as stated above, it suffices to fix the matrix K within the next iteration. In addition, the novel method is a constrained optimization method: it considers conditions (18) and (8)) when solving the nonlinear least-squares problem. At each iteration of the novel method, the LSL problem is solved, which belongs to the class of convex programming problems [38, 39]. For such problems, the existing effective optimization procedures yield the solution or state its absence. (For example, we mention the lsqlin function in Matlab.)

4.2. Solution of the Structural Design Problem

Assume that the controller structure is not given: the set $\check{S}$ is not specified in the initial problem data and must be determined. In this case, we have the structural design problem. Within the adopted formalization (10), it consists in finding sets $\check{S}$ and corresponding matrices K for which conditions (7)–(9) hold and the structure of the controller (3), (4) is simple [27–31]. It can be solved using the algorithm for designing general-form simple structures [31]. The procedure proposed in subsection 4.1 may serve to assess the acceptability of the controller structure and calculate the corresponding parameters.

5. EXAMPLES

Example 1. Consider the model of a two-mass system [1, p. 52, p. 125]. Assume that the output is composed of all components of the state vector except the second one. Given a time discretization step of 0.01, unit masses, and a stiff spring linking them, we obtain the following matrices of system (1), (2):

$$A = \begin{pmatrix} 1 & 0 & 0.01 & 0 \\ 0 & 1 & 0 & 0.01 \\ -0.01 & 0.01 & 1 & 0 \\ 0.01 & -0.01 & 0 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 0 \\ 0 \\ 0.01 \\ 0 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

In (3), the desired matrix K has dimensions 1×3. All its components are allowed to be nonzero; therefore, $\check{S} = \{(1,1)\ (1,2)\ (1,3)\}$ in (4).

We define the desired behavior of system (1)–(4) as follows. Let the desired trajectories correspond to the optimal control by the minimum energy criterion that transfers the system from the initial states $x_0^1 = (-1; 1; 1; -1)$ and $x_0^2 = (-10; 10; -1; -1)$ to the origin in time $k = 250$. Together with the initial conditions, these trajectories $Y_1 = (y_k^1)$, $Y_2 = (y_k^2)$, $k \in \{\overline{1{,}500}\}$, form the set of training examples (5) $Q = \{(x_0^1, Y_1), (x_0^2, Y_2)\}$. They can be calculated using the known dependencies [1, p. 128].

First, we solve the design problem without the constraints (7), (8) (i.e., the unconstrained optimization problem of the objective function (9)). After three iterations, the novel method described in Section 4.1 yields the gain matrix $K = (-10.671\ \ -4.124\ \ -13.745)$. The corresponding degree of stability is $\sigma = 0.964 \times 10^{-2}$, and the objective function takes a value of 37.25.

Example 2. To improve stability, we increase the value σ to 1.2×10^{-2} and reduce the amplitude of oscillations on the final interval (for $k \in \{300, \ldots, 500\}$), restricting the admissible deviation of the output coordinates from the desired trajectories to the values ± 0.5 and ± 1.5 for x_0^1 and x_0^2, respectively. (In Example 1, these deviations are 0.68 and 2.23.) Given the above requirements, it is therefore necessary to solve the constrained optimization problem (9), (7), (8). Three iterations of the novel method result in $K = (-13.012\ \ -5.310\ \ -16.821)$; in addition, $\sigma = 1.2 \times 10^{-2}$, conditions (7) and (8) hold, and the value of the objective function is 120.32.

Example 3. Consider the lateral motion model of an aircraft presented in [26, p. 182]. For a time discretization step of 0.001, we obtain the following matrices of system (1):

$$A = \begin{pmatrix} 1000 & 0 & 1 & 0.044 & 0 \\ -1.215 & 999 & 0.131 & 0 & 0 \\ 0.430 & 0.021 & 1000 & 0 & 0 \\ 0 & 1 & 0 & 1000 & 0 \\ 0 & 0 & 1 & 0 & 1000 \end{pmatrix} \times 10^{-3}, \quad B = \begin{pmatrix} 0 & 0 \\ -0.040 & 1.587 \\ 0.381 & -0.067 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \times 10^{-3}.$$

In equation (2), C is an identity matrix of dimensions 5×5.

Solving for system (1)–(3) the LQR problem with the minimization criterion $\sum_{k=1}^{\infty} x_k^{\mathrm{T}} x_k$, we find the gain matrix of the controller (3)

$$K_{\mathrm{LQR}} = -\begin{pmatrix} 2.049 & 0.098 & 3.937 & 0.096 & 0.766 \\ -0.110 & 1.100 & -0.168 & 1.031 & -0.642 \end{pmatrix}.$$

Let the first component of the state vector be excluded from the controller's variables for its structural simplification. This can be done by writing condition (4) of the design problem as

$k_{1,1} = 0$, $k_{2,1} = 0$ (equivalently, the design problem of an output feedback containing all components of the state vector except the first one). Accordingly, the set $\check{S}$ in (4) is given and includes all number pairs of the elements of the matrix K except (1,1) and (2,1).

The set of training examples Q consists of the trajectories $Y_\gamma = (y_k^\gamma)$, $\gamma \in \{\overline{1,5}\}$, $k \in \{\overline{1,10^4}\}$, of system (1)–(3) with the LQR controller (with the gain matrix $K = K_{\mathrm{LQR}}$) corresponding to initial conditions x_0^γ where the component with number γ is 1 and the others are zero. Let the components of the vectors $\varepsilon_k^{\gamma-}$ and $\varepsilon_k^{\gamma+}$ in (7) be assigned by requiring that the admissible deviation of the trajectories y_k^γ of system (1)–(3) from the desired ones lies within $\pm 1\%$ of their maximum absolute values at each time instant k. In addition, the degree of stability of the designed system must be not smaller than that of system (1)–(3) with the LQR controller. For this purpose, $\sigma = 4 \times 10^{-5}$ is chosen in (8).

Using the novel method, we find the gain matrix

$$K = \begin{pmatrix} 0 & 6.622 & -13.519 & 8.180 & -8.181 \\ 0 & -1.420 & 0.621 & -1.414 & 1.001 \end{pmatrix}.$$

The solution is obtained after four iterations upon satisfying the assigned constraints without progress in decreasing the objective function.

Example 4. We modify the problem of Example 3 as follows. Let the first component of the state vector be excluded from the output by redefining the matrix C in equation (2) as a matrix of dimensions 4×5 obtained by eliminating the first row from the matrix C of Example 3. In this case, the desired matrix K has dimensions 2×4. We solve the structural design problem of the system in the statement presented in subsection 4.2. The novel method yields the sets $\check{S}$ and the corresponding matrices K (see the table) for which conditions (21) and (22) are satisfied and the controller (3), (4) has a simple structure [27–31].

Table

No.	Gain matrix	No.	Gain matrix
1	$\begin{pmatrix} 4.819 & -10.920 & 5.898 & -6.696 \\ 0 & -1.444 & 0.403 & -0.184 \end{pmatrix}$	3	$\begin{pmatrix} 5.230 & -11.510 & 6.413 & -7.030 \\ -0.318 & -0.984 & 0 & 0.0786 \end{pmatrix}$
2	$\begin{pmatrix} 6.084 & -12.742 & 7.500 & -7.736 \\ -0.994 & 0 & -0.867 & 0.644 \end{pmatrix}$	4	$\begin{pmatrix} 5.112 & -11.344 & 6.268 & -6.936 \\ -0.225 & -1.120 & 0.120 & 0 \end{pmatrix}$

6. CONCLUSIONS

This paper has proposed a novel approach to designing static feedback in linear discrete time-invariant control systems. Within this approach, the desired behavior of the system is defined by a set of its output variation laws (training examples). The problem statement and solution method can be generalized to the case dynamic controllers based on the known procedure [3] for reducing dynamic feedback design to an equivalent static feedback design.

The algorithm for solving the static controller learning problem (see subsection 4.1) is heuristic: its convergence has been confirmed by computational experiments without rigorous proof.

APPENDIX

Proof of Statement 1. Let the matrix K be the solution of system (16). Equations (16) and (6) are equivalent if all the desired trajectories Y_γ, $\gamma \in \{\overline{1,q}\}$, belong to the set of solutions of system (1)–(4); see the considerations above. Hence, under all other hypotheses of the proposition, choosing the matrix K based on equalities (16) ensures the requirements (6). This proves the sufficiency part of Proposition 1. If the matrix K is not the solution of system (16), violating equations (16) will also violate conditions (6). If some of the desired trajectories Y_γ, $\gamma \in \{\overline{1,q}\}$, do not belong to the set of solutions of system (1)–(4), the equality $y_k = y_k^\gamma$ will not hold for them at each time instant $k \in \{\overline{1,N}\}$. Therefore, conditions (6) will fail as well. This proves the necessity part of Proposition 1.

Proof of Statement 2. This result follows from the equivalence of conditions (1)–(4) and (7)–(9) (on the one hand) and conditions (4), (8), (17), and (18) (on the other hand).

REFERENCES

1. Polyak, B.T., Khlebnikov, M.V., and Rapoport, L.B., *Matematicheskaya teoriya avtomaticheskogo upravleniya* (Mathematical Theory of Automatic Control), Moscow: Lenand, 2019.
2. Sadabadi, M.S. and Peaucelle, D., From Static Output Feedback to Structured Robust Static Output Feedback: A Survey, *Ann. Rev. Control*, 2016, vol. 42, pp. 11–26.
3. Syrmos, V.L., Abdallah, C.T., Dorato, P., and Grigoriadis, K., Static Output Feedback—a Survey, *Automatica*, 1997, vol. 33, no. 2, pp. 125–137.
4. Toker, O. and Ozbay, H., On the Np-Hardness of Solving Bilinear Matrix Inequalities and Simultaneous Stabilization with Static Output Feedback, *IEEE American Control Conference*, Seattle, 1995, pp. 2525–2526.
5. Toscano, R., *Structured Controllers for Uncertain Systems: A Stochastic Optimization Approach*, New York: Springer-Verlag, 2013.
6. Rosinova, D., Vesely, V., and Kucera, V., A Necessary and Sufficient Condition for Static Output Feedback Stabilizability of Linear Discrete-Time Systems, *Kybernetika*, 2003, vol. 39, pp. 447–459.
7. Cao, Y.Y., Lam, J., and Sun, Y.X., Static Output Stabilization: An ILMI Approach, *Automatica*, 1998, vol. 34, no. 12, pp. 1641–1645.
8. Wang, X., Pole Placement by Static Output Feedback, *J. Math. Syst. Estim. Control*, 1992, vol. 2, no. 2, pp. 205–218.
9. Pakshin, P.V. and Ryabov, A.V., A Static Output Feedback Control for Linear Systems, *Autom. Remote Control*, 2004, vol. 65, no. 4, pp. 559–566.
10. Agulhari, C.M., Oliveira, R.C., and Peres, P.L., LMI Relaxations for Reduced-Order Robust H^∞-control of Continuous-Time Uncertain Linear Systems, *IEEE Trans. Autom. Control*, 2012, vol. 57, no. 6, pp. 1532–1537.
11. Ebihara, Y., Tokuyama, K., and Hagiwara, T., Structured Controller Synthesis Using LMI and Alternating Projection Method, *Int. J. Control*, 2004, vol. 77, no. 12, pp. 1137–1147.
12. Grigoriadis, K.M. and Beran, E.B., Alternating Projection Algorithms for Linear Matrix Inequalities Problems with Rank Constraints, in *Advances in Linear Matrix Inequality Methods in Control*, Philadelphia: SIAM, 2000, pp. 251–267.
13. Leibfritz, F., An LMI-Based Algorithm for Designing Suboptimal Static H^2/H^∞ Output Feedback Controllers, *SIAM J. Control Optim.*, 2001, vol. 39, no. 6, pp. 1711–1735.
14. Polyak, B.T., Khlebnikov, M.V., and Shcherbakov, P.S., Sparse Feedback in Linear Control Systems, *Autom. Remote Control*, 2014, vol. 75, no. 12, pp. 2099–2111.
15. Bykov, A.V. and Shcherbakov, P.S., Sparse Feedback Design in Discrete-Time Linear Systems, *Autom. Remote Control*, 2018, vol. 79, no. 7, pp. 1175–1190.
16. Lin, F., Fardad, M., and Jovanović, M.R., Design of Optimal Sparse Feedback Gains via the Alternating Direction Method, *IEEE Trans. Autom. Control*, 2013, vol. 58, no. 9, pp. 2426–2431.
17. Belozyorov, V.Y., New Solution Method of Linear Static Output Feedback Design Problem for Linear Control Systems, *Linear Algebra Appl.*, 2016, vol. 504, pp. 204–227.

18. Blumthaler, I. and Oberst, U., Design, Parametrization, and Pole Placement of Stabilizing Output Feedback Compensators via Injective Cogenerator Quotient Signal Modules, *Linear Algebra Appl.*, 2012, vol. 436, pp. 963–1000.

19. Johnson, T. and Athans, M., On the Design of Optimal Constrained Dynamic Compensators for Linear Constant Systems, *IEEE Trans. Autom. Control*, 1970, vol. 15, pp. 658–660.

20. Moerder, D. and Calise, A., Convergence of a Numerical Algorithm for Calculating Optimal Output Feedback Gains, *IEEE Trans. Autom. Control*, 1985, vol. 30, pp. 900–903.

21. Choi, S. and Sirisena, H., Computation of Optimal Output Feedback Gains for Linear Multivariable Systems, *IEEE Trans. Autom. Control*, 1974, vol. 19, pp. 254–258.

22. Kreisselmeier, G., Stabilization of Linear Systems by Constant Output Feedback Using the Riccati Equation, *IEEE Trans. Autom. Control*, 1975, vol. 20, pp. 556–557.

23. Toivonen, H.T., A Globally Convergent Algorithm for the Optimal Constant Output Feedback Problem, *Int. J. Control*, 1985, vol. 41, no. 6, pp. 1589–1599.

24. Geromel, J., Peres, P., and Souza, S., Convex Analysis of Output Feedback Structural Constraints, *Proc. IEEE Conf. on Decision and Control*, San Antonio, 1993, pp. 1363–1364.

25. Iwasaki, T. and Skelton, R., All Controllers for the General H^{∞} Control Problem: LMI Existence Conditions and State Space Formulas, *Automatica*, 1994, vol. 30, pp. 1307–1317.

26. Paraev, Yu.I. and Smagina, V.I., Problems of Simplifying the Structure of Optimal Controllers, *Avtomat. i Telemekh.*, 1975, no. 6, pp. 180–183.

27. Mozzhechkov, V.A., *Prostye struktury v teorii upravleniya* (Simple Structures in Control Theory), Tula: Tula State University, 2000.

28. Mozzhechkov, V.A., Design of Simple-Structure Linear Controllers, *Autom. Remote Control*, 2003, vol. 64, no. 1, pp. 23–36.

29. Mozzhechkov, V.A., Design of Simple Robust Controllers for Time-Invariant Dynamic Systems, *J. Comput. Syst. Sci. Int.*, 2021, vol. 60, pp. 353–363.

30. Mozzhechkov, V.A., Synthesis of Simple Relay Controllers in Self-oscillating Control Systems, *Autom. Remote Control*, 2022, vol. 83, no. 9, pp. 1393–1403.

31. Mozzhechkov, V.A., Simple Structures in Problems of Control Theory: Formalization and Synthesis, *J. Comput. Syst. Sci. Int.*, 2022, vol. 61, pp. 295–312.

32. Vapnik, V.N., An Overview of Statistical Learning Theory, *Transactions on Neural Networks*, 1999, vol. 10, no. 5, pp. 988–999.

33. Vorontsov, K.V., Combinatorial Bounds for the Quality of Learning by Precedents, *Dokl. Akad. Nauk*, 2004, vol. 394, no. 2, pp. 175–178.

34. Mohri, M., Rostamizadeh, A., and Talwalkar, A., *Foundations of Machine Learning*, Massachusetts: MIT Press, 2012.

35. Schmidhuber, J., Deep Learning in Neural Networks, *Neural Networks*, 2015, vol. 61, pp. 85–117.

36. Ikramov, Kh.D., *Chislennoe reshenie matrichnykh uravnenii* (Numerical Solution of Matrix Equations), Moscow: Nauka, 1984.

37. Gill, Ph.E., Murray, W., Wright, M.H., *Practical Optimization*, London: Academic Press, 1981.

38. Bertsekas, D.P., *Convex Optimization Algorithms*, Belmont: Athena Scientific, 2015.

39. Polyak, B., *Introduction to Optimization*, Optimization Software, 1987.

This paper was recommended for publication by P.V. Pakshin, a member of the Editorial Board