

УДК 62.51+681.5.011
ББК Ж 30

СТАБИЛИЗАЦИЯ 2D-СИСТЕМЫ ИТЕРАТИВНОГО ОБУЧАЮЩЕГО УПРАВЛЕНИЯ ПРИ ПЕРЕДАЧЕ ДАННЫХ ПО КАНАЛУ СЕТИ СО СЛУЧАЙНЫМ ЗАПАЗДЫВАНИЕМ И ПОТЕРЕЙ ПАКЕТОВ ДАННЫХ¹

Валов С. А.²

(Арзамасский политехнический институт НГТУ
им. Р.Е. Алексеева, Арзамас)

В статье рассматривается дискретная 2D-система итерационного обучающего управления, в которой датчик и регулятор обмениваются данными через сетевой канал связи. В ней решается задача стабилизации системы в условиях возможного появления запаздывания передачи данных и потери пакетов данных в сетевом канале. Рассматривается управление на основе вектора состояния и вектора выходного сигнала. Запаздывание и потеря пакета данных моделируются с помощью марковских цепей. Результирующая замкнутая система преобразуется к дискретной 2D-системе Роессера с марковскими переключениями. Для однопакетной передачи данных даются достаточные условия стабилизации системы в терминах линейных матричных неравенств. Приводится численный пример для иллюстрации полученных результатов.

Ключевые слова: итерационное обучающее управление, сетевой канал связи, запаздывание, марковская модель, стохастическая устойчивость, линейные матричные неравенства.

¹ Работа выполнена при финансовой поддержке РФФИ, гранты №10-08-00843, №11-01-97025.

² Сергей Александрович Валов, аспирант (serval_nn@mail.ru).

1. Введение

Большинство современных систем автоматического управления представляют собой технически сложные устройства (например, самолет или ядерный реактор). Они включают в свой состав большое количество подсистем и должны обеспечивать решение комплексных задач. Одной из востребованных в настоящее время задач является построение и реализация алгоритмов управления с итерационным обучением, обеспечивающих достижение системой заданной траектории за конечное время и устойчивость во время повторений в процессе обучения.

Итерационное обучающее управление – это метод управления для систем, работающих в повторяющемся режиме, с требованием, чтобы желаемая траектория прослеживалась с заданной точностью на определенном интервале времени [26]. Итерационное обучающее управление использует информацию с предыдущих итераций для формирования сигнала управления в попытке уменьшить ошибку слежения на следующей итерации. Примерами таких систем являются робот-манипулятор, который должен многократно повторять определенные однотипные действия, процессы серийного производства, химические серийные процессы, системы слежения и т.п. [3, 5, 7, 10, 13].

Итерационное обучающее управление впервые было математически определено Arimoto, Kawamura, Miyazaki в 1984 [5]. С этого момента область итерационного обучающего управления стала предметом интенсивных теоретических и практических научных исследований.

Основы систем итерационного обучающего управления были заложены в цикле статей [5, 6]. Результаты дальнейших исследований представлены в работах [10, 17, 18, 26]. За последние несколько лет была разработана теория линейного и нелинейного, робастного, адаптивного и оптимального итерационного обучающего управления как для непрерывных по времени, так и дискретных систем. Основные результаты по итерационному обучающему управлению представлены в обзорных статьях [3, 9, 19].

В [4, 14] 2D-теория была применена к итерационному обучающему управлению. Итерационное обучающее управление, рассмотренное в контексте теории повторяющихся процессов, представлено в [22].

В современных промышленных системах такие компоненты, как объект управления, регулятор, датчики и исполнительные элементы сложно расположить в одном месте, поэтому их разделяют и размещают на определенном расстоянии друг от друга. Для связи компонентов такой распределенной системы используют сетевые каналы связи (например, беспроводная сеть). Такие системы называются сетевыми системами управления [23, 32]. В отличие от традиционных систем управления с прямой обратной связью, сетевые системы управления обладают такими преимуществами, как низкая стоимость, простота установки и диагностики, возможность изменения конфигурации, высокая надежность, быстродействие системы; они требуют меньше энергии и обладают меньшим весом [30].

Однако включение сетей связи в контур обратной связи вносит определенные трудности, такие как запаздывание передачи данных, потеря пакетов данных, что приводит к неустойчивости и низкой эффективности [1, 30, 32]. В свою очередь, инженер обладает незначительными ресурсами для использования данной архитектуры и должен учитывать отрицательные моменты, вызванные особенностями сетей связи. Оценка и компенсация запаздывания и потерей пакетов данных – одна из центральных областей исследований сетевых систем управления [24, 28, 29, 31].

Большинство работ по сетевому управлению охватывают только одномерные системы [23, 24, 25, 28]. Новой задачей является синтез управления для двумерной системы итерационного обучения, компоненты которой обмениваются данными через сетевой канал связи, в котором возможны запаздывание передачи пакета данных и его потеря. Данная задача не рассматривалась ранее в литературе и представляет несомненный интерес к изучению, с учетом того факта, что результаты для

одномерных сетевых систем управления трудно применить для анализа двумерных систем, имеющих более сложную динамику.

В [2] были предложены первые варианты моделей систем управления с итерационным обучением и повторяющихся процессов в условиях, когда управление осуществляется через цифровой канал связи с информационными ограничениями.

В данной работе развиваются идеи [2] для двумерной дискретной системы итерационного обучающего управления в условиях, когда данные передаются через сетевой канал связи. Предполагается, что запаздывание и потеря пакета данных возникают только на стороне датчик-регулятор. Такой случай характерен для многих реализаций сетевых систем управления [29].

Процесс запаздывания передачи данных и потери пакета данных описываются при помощи однородных марковских цепей. При помощи расширения векторов состояния сетевая система итерационного обучающего управления преобразуется к дискретной $2D$ -системе Роессера с марковскими переключениями. Ставится задача синтеза управления с обратной связью по выходу и вектору состояния совместно, обеспечивающего стохастическую устойчивость результирующей системы.

В данной статье для однопакетной передачи информации даются достаточные условия устойчивости системы в терминах линейных матричных неравенств, а также условия нахождения матриц управления. Приводится численный пример решения задачи стабилизации модели двигателя постоянного тока в условиях управления и передачи данных по сети.

2. Сетевая реализация системы итерационного обучающего управления

Итерационное обучение можно представить как двумерный динамический процесс [14]. Первый процесс отражает динамику системы во времени. Второй процесс отражает динамику итерационного обучения:

$$(1) \begin{cases} x(t+1, k) = Ax(t, k) + Bu(t, k), \\ y(t, k) = Cx(t, k); \end{cases}$$

где t – дискретное время; k – номер шага обучения; $x \in R^n$ – вектор состояния; $y \in R^p$ – выходной вектор; $u \in R^m$ – входной вектор управления и A, B, C – действительные матрицы соответствующей размерности.

В течение обучающего процесса каждая переменная динамической системы может быть выражена как функция двух независимых переменных: времени t и итерации k .

Граничные условия для системы (1) имеют следующий вид:

$$(2) \begin{cases} x(0, k) = x_0, & k = 0, 1, \dots, \\ u(t, 0) = 0, & t = 0, 1, \dots, N. \end{cases}$$

Алгоритм итерационного обучения состоит в следующем. На каждом шаге обучения все доступные измерению переменные запоминаются и затем используются на следующем шаге для модификации управления и достижения заданной траектории системы. Закон итерационного обучающего управления представляется в виде

$$(3) \quad u(t, k) = u(t, k-1) + \Delta u(t, k-1),$$

$$(4) \quad e(t, k) = y_r(t) - y(t, k),$$

где Δu обозначает модификацию сигнала управления; $e(t, k)$ – ошибка на k -й итерации; $y_r(t)$ – заданная траектория системы.

Цель итерационного обучающегося управления – уменьшение ошибки от одной итерации к другой, что для двумерной структуры системы формально выражается как

$$(5) \quad \lim_{k \rightarrow \infty} \|e(t, k)\| = 0.$$

1.1. СЕТЕВАЯ АРХИТЕКТУРА

Рассматривается сетевая система управления с однопакетной передачей информации, которая подразумевает, что данные объединяются в один сетевой пакет и передаются одновременно через дискретные промежутки времени [30]. Такому типу передачи соответствуют сети с пакетом данных большого размера.

Например, возможно использование сети *Ethernet*, один пакет данных которой может содержать до 1500 В [32].

Сетевая архитектура включает датчик, регулятор и исполнительный элемент, которые все подключены через сеть связи и синхронизированы по времени (рис. 1).

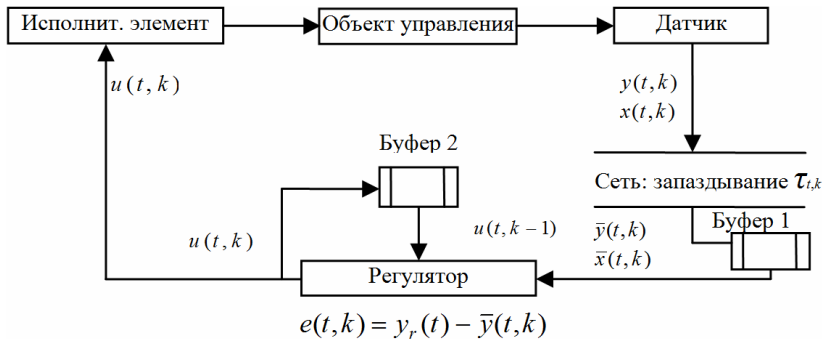


Рис. 1. Сетевая реализация системы итерационного обучения

Влияние квантования сигнала, пакетных очередей и неправильной кодировки в сети связи не рассматривается. Предполагается, что сеть связи находится как на стороне датчик-регулятор, так и на стороне регулятор-исполнительный элемент, однако потеря пакетов данных и появление запаздывания в передачи данных возникает только на стороне датчик-регулятор.

Датчик формирует пакет данных и передает его на регулятор через равные отрезки времени. Регулятор имеет принимающий буфер обмена 1 (рис. 1), который содержит самую последнюю информацию, полученную от датчика. Регулятор считывает данные из буфера периодически через малые периоды времени. Данные буфера обмена 1 обновляются по мере прихода нового пакета данных. Буфер обмена 2 (рис. 1) используется для запоминания управляющего сигнала на текущей итерации.

Считаем, что вектор состояния системы $x(t, k)$ доступен измерению. Выходной вектор системы $y(t, k)$ определяется из второго уравнения (1). Пакет данных, передаваемый на регуля-

тор для формирования управления, включает вектора $x(t, k)$, $y(t, k)$.

1.2. МОДЕЛИРОВАНИЕ ЗАПАЗДЫВАНИЯ И ПОТЕРИ ПАКЕТА ДАННЫХ

Обозначим $\tau_{t,k}$ – запаздывание передачи пакета данных текущей итерации k на стороне датчик-регулятор в дискретный момент времени t :

$$(6) \quad 0 \leq \tau_{t,k} \leq d, \quad d > 0, \quad d \in Z.$$

Запаздывание $\tau_{t,k}$ одинаково действует на величины $x(t, k)$, $y(t, k)$, поступающие на регулятор в одном пакете данных. Модель запаздывания и потери пакета данных строится следующим образом. В случае запаздывания передачи пакета данных (далее по тексту просто запаздывание) информация о пакете с данными $y(t, k - \tau_{t,k})$, $x(t, k - \tau_{t,k})$ будет использована регулятором. Поэтому $\bar{y}(t, k) = y(t, k - \tau_{t,k})$, $\bar{x}(t, k) = x(t, k - \tau_{t,k})$. В случае потери пакета данных регулятор использует информацию с предыдущего шага: $\bar{y}(t, k) = \bar{y}(t, k - 1)$, $\bar{x}(t, k) = \bar{x}(t, k - 1)$. Модель передачи пакета данных на стороне датчик-регулятор описывается следующим образом:

$$(7) \quad [\bar{x}(t, k), \bar{y}(t, k)] = \begin{cases} x(t, k - \tau_{t,k}), y(t, k - \tau_{t,k}) - \text{пакет успешно передан} \\ \bar{x}(t, k - 1), \bar{y}(t, k - 1) - \text{пакет потерян.} \end{cases}$$

Данная модель передачи пакета данных позволяет включить потери пакетов данных в модель запаздывания и рассматривать систему с запаздыванием [25].

Регулятор всегда использует самую последнюю информацию из буфера обмена 1. Поэтому, если на шаге k в буфере обмена 1 хранятся данные с предыдущей успешной передачи $k - \tau_{t,k}$ и не поступает новых данных с шага $k + 1$ в дискретный момент времени t (данные потеряны или происходит запаздывание передачи данных по времени), тогда на регулятор поступают данные с шага $k - \tau_{t,k}$, хранимые в буфере обмена 1.

Пример работы сетевой системы с запаздыванием приведен на рис. 2.

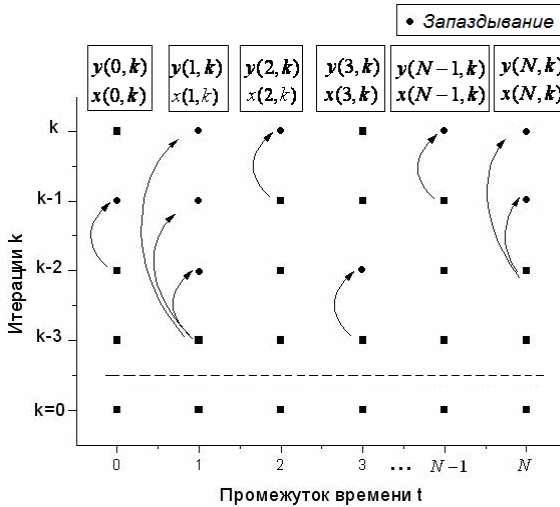


Рис. 2. Принцип работы сетевой системы с запаздыванием

Действие запаздывания $\tau_{t,k}$ на сигнал, поступающий на регулятор, описывается как

$$(8) \quad \bar{x}(t, k) = x(t, k - \tau_{t,k}),$$

$$(9) \quad \bar{y}(t, k) = y(t, k - \tau_{t,k}).$$

Как видно из (8)–(9), запаздывание в уравнениях возникает только в области итераций. Такая особенность связана с организацией сетевой архитектуры для рассматриваемой системы. Предполагается, что в буфере обмена 1 регулятора отводится память под данные на каждом дискретном отрезке времени $t = 0, 1, \dots, N$. Датчик в дискретный момент времени t посылает на регулятор пакет данных, которые прописываются в буфер обмена с отметкой времени по t . Таким образом, буфер обмена всегда содержит информацию о траектории системы на всех дискретных отрезках времени, но на разных итерациях, в зависимости от успешности передачи данных. Если в момент времени t произошла потеря пакета данных (абсолютная потеря пакета) или пакет данных поступает в буфер обмена регулятора с превышением лимита времени, то данные в памяти буфера с

отметкой времени по t не обновляются. Поэтому из буфера обмена на регулятор поступают необновленные данные с предыдущей итерации (см. рис. 2).

Запаздывание $\tau_{t,k}$ моделируем при помощи однородных марковских цепей. Марковская цепь позволяет учитывать зависимости между запаздываниями текущей и предыдущими итерациями [31].

Пусть $\tau_{t,k}$ – однородные марковские цепи со значениями $H = \{0, 1, 2, \dots, d\}$ и равными матрицами вероятности $P = (p_{ij})$. Вероятность перехода $\tau_{t,k}$ (переход из состояния i в j) в дискретный момент времени t определяются как

$$(10) \quad p_{ij} = P(\tau_{t,k+1} = j | \tau_{t,k} = i),$$

где $p_{ij} \geq 0$, $i, j \in H$, и

$$\sum_{j=0}^d p_{ij} = 1 \quad \text{для } t = 0, 1, \dots, N.$$

Структура матрицы вероятности выглядит следующим образом [25]:

$$(11) \quad P = \begin{bmatrix} p_{00} & p_{01} & 0 & 0 & \dots & 0 \\ p_{10} & p_{11} & p_{12} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{d0} & p_{d1} & p_{d2} & p_{d3} & \dots & p_{dd} \end{bmatrix}.$$

Каждый столбец представляет вероятность перехода из текущего состояния во все остальные. Диагональные элементы представляют вероятность прихода данных с одинаковым запаздыванием. Элементы, расположенные выше диагонали, представляют вероятности больших по времени запаздываний; элементы, расположенные ниже диагонали, указывают на потерю пакета данных или обновление данных.

1.3. РЕЗУЛЬТИРУЮЩАЯ СИСТЕМА С ОБРАТНОЙ СВЯЗЬЮ

Ошибка обучения на k -й итерации определяется формулой (4). При наличии запаздывания $\tau_{t,k}$ на стороне датчик-регу-

лятор регулятор использует данные $\bar{y}(t, k) = y(t, k - \tau_{t, k})$. Поэтому ошибка обучения, определяемая на регуляторе, есть

$$(12) \quad e(t, k) = y_r(t) - \bar{y}(t, k) = y_r(t) - y(t, k - \tau_{t, k}).$$

Регулятор периодически вычисляет ошибку обучения на основе данных, которые прописываются в буфере обмена 1. Исходя из того, что в работе регулятора не возникает запаздываний по времени (регулятор не простаивает), то, используя (1) и (12), можно получить:

$$(13) \quad e(t, k+1) - e(t, k) = y(t, k - \tau_{t, k}) - y(t, k+1 - \tau_{t, k+1}) = \\ -CA\eta(t, k - \tau_{t, k}) - CB\Delta u(t-1, k - \tau_{t-1, k}),$$

$$(14) \quad \eta(t, k) = x(t-1, k+1) - x(t-1, k),$$

$$(15) \quad \eta(t, k - \tau_{t, k}) = x(t-1, k+1 - \tau_{t-1, k+1}) - x(t-1, k - \tau_{t-1, k}),$$

$$(16) \quad \Delta u(t-1, k - \tau_{t-1, k}) = u(t-1, k+1 - \tau_{t-1, k+1}) - u(t-1, k - \tau_{t-1, k}),$$

где сигнал $u(t-1, k - \tau_{t-1, k})$ хранится в буфере 1 как результат предыдущей итерации; $u(t-1, k+1 - \tau_{t-1, k+1})$ – управляющей сигнал на следующей итерации.

Из (1) и (3) находим

$$(17) \quad \eta(t+1, k) = A\eta(t, k) + B\Delta u(t-1, k).$$

Приращение управления, связанное с процессом итеративного обучения, формируем на основе векторов $\eta(t, k)$, $e(t, k)$, с учетом запаздывания:

$$(18) \quad \Delta u(t-1, k - \tau_{t-1, k}) = K_1(\tau_{t, k})\eta(t, k - \tau_{t, k}) + K_2(\tau_{t, k})e(t, k - \tau_{t, k}),$$

где $K_1(\tau_{t, k})$, $K_2(\tau_{t, k})$ – совокупность управлений на основе $\tau_{t, k}$ в момент времени t .

Приращение управления без учета $\tau_{t, k}$ выглядит следующим образом:

$$(19) \quad \Delta u(t-1, k) = K_1(0)\eta(t, k) + K_2(0)e(t, k).$$

Уравнения (13) и (17) с учетом (18) и (19) можно переписать в компактную форму:

$$(20) \quad \begin{bmatrix} \eta(t+1, k) \\ e(t, k+1) \end{bmatrix} = \begin{bmatrix} A+BK_1(0) & BK_2(0) \\ 0 & I \end{bmatrix} \cdot \begin{bmatrix} \eta(t, k) \\ e(t, k) \end{bmatrix} +$$

$$+ \begin{bmatrix} 0 & 0 \\ -CA - CBK_1(\tau_{t,k}) & -CBK_2(\tau_{t,k}) \end{bmatrix} \cdot \begin{bmatrix} \eta(t, k - \tau_{t,k}) \\ e(t, k - \tau_{t,k}) \end{bmatrix}.$$

Система (20) представляет 2D-систему Роессера [20, 21] со случайным запаздыванием $\tau_{t,k}$. Функции $\tau_{t,k}$ – это однородные марковские процессы на вероятностном пространстве с целочисленными значениями $H = \{0, 1, 2, \dots, d\}$.

Граничные условия для системы (20) следуют из (1)–(4) и (14):

$$(21) \quad \begin{aligned} \eta(1, k) &= 0, \quad k = 0, 1, \dots, \\ e(t, 0) &= y_r(t) - CA^t x_0, \quad t = 1, 2, \dots \end{aligned}$$

Определим следующие расширенные вектора состояний (20):

$$(22) \quad \bar{\eta}(t, k) = \begin{bmatrix} \eta(t, k) \\ \eta(t, k-1) \\ \vdots \\ \eta(t, k-d) \end{bmatrix}, \quad \bar{e}(t, k) = \begin{bmatrix} e(t, k) \\ e(t, k-1) \\ \vdots \\ e(t, k-d) \end{bmatrix}.$$

Величины $\eta(t, k - \tau_{t,k})$ и $e(t, k - \tau_{t,k})$ определим следующим образом:

$$(23) \quad \eta(t, k - \tau_{t,k}) = V_1(\tau_{t,k}) \bar{\eta}(t, k),$$

$$(24) \quad e(t, k - \tau_{t,k}) = V_2(\tau_{t,k}) \bar{e}(t, k),$$

где

$$(25) \quad V_1(\tau_{t,k}) = \begin{cases} [I_{n \times n} & 0 & \dots & 0] & \tau_{t,k} = 0, \\ [0 & \dots & I_{n \times n} & \dots & 0] & \tau_{t,k} > 0, \end{cases}$$

$$(26) \quad V_2(\tau_{t,k}) = \begin{cases} (1+d) \text{ блок} \\ [I_{p \times p} & 0 & \dots & 0] & \tau_{t,k} = 0, \\ [0 & \dots & I_{p \times p} & \dots & 0] & \tau_{t,k} > 0. \end{cases}$$

(1+d) блок

Все элементы матриц $V_1(\tau_{t,k})$ и $V_2(\tau_{t,k})$ нулевые, за исключением $\tau_{t,k}$ блока, где он представлен единичной матрицей. Будем рассматривать систему (20) относительно $\tau_{t,k} = 0$ и

$\tau_{t,k} = 1, 2, \dots, d$. С учетом (22)–(26) уравнение (20) можно записать в следующем виде:

– для $\tau_{t,k} = 0$:

$$(27) \begin{bmatrix} \eta(t+1, k) \\ e(t, k+1) \end{bmatrix} = \begin{bmatrix} A+BK_1(0) & BK_2(0) \\ -CA-CBK_1(0) & I-CBK_2(0) \end{bmatrix} \cdot \begin{bmatrix} \eta(t, k) \\ e(t, k) \end{bmatrix}.$$

– для $\tau_{t,k} = 1, 2, \dots, d$:

$$(28) \begin{bmatrix} \bar{\eta}(t+1, k) \\ \bar{e}(t, k+1) \end{bmatrix} = \begin{bmatrix} \bar{A}+\bar{B}\bar{K}_1(0) & \bar{B}\bar{K}_2(0) \\ -\bar{D}\bar{V}_1(\tau_{t,k})-\bar{N}\bar{K}_1(\tau_{t,k})\bar{S}_1(\tau_{t,k}) & \bar{F}-\bar{N}\bar{K}_2(\tau_{t,k})\bar{S}_2(\tau_{t,k}) \end{bmatrix} \begin{bmatrix} \bar{\eta}(t, k) \\ \bar{e}(t, k) \end{bmatrix},$$

где $\bar{A} = \text{diag}(A, \dots, A)$, $\bar{B} = \text{diag}(B, \dots, B)$, $\bar{K}_1(0) = \text{diag}(K_1(0), \dots, K_1(0))$,

$\bar{K}_2(0) = \text{diag}(K_2(0), \dots, K_2(0))$, $\bar{N} = ES_1$, $\bar{S}_1(\tau_{t,k}) = S_2\bar{V}_1(\tau_{t,k})$,

$\bar{S}_2(\tau_{t,k}) = S_3\bar{V}_2(\tau_{t,k})$, $S_2 = [I_{n \times n} \ 0_{n \times n} \ \dots \ 0_{n \times n}]$, $S_3 = [I_{p \times p} \ 0_{p \times p} \ \dots \ 0_{p \times p}]$,

$$S_1 = \begin{bmatrix} I_{m \times m} \\ 0_{m \times m} \\ \vdots \\ 0_{m \times m} \end{bmatrix}, \bar{D} = \begin{bmatrix} CA & \dots & 0_{p \times n} \\ \vdots & \ddots & \vdots \\ 0_{p \times n} & \dots & 0_{p \times n} \end{bmatrix}, \bar{E} = \begin{bmatrix} CB & \dots & 0_{p \times m} \\ \vdots & \ddots & \vdots \\ 0_{p \times m} & \dots & 0_{p \times m} \end{bmatrix},$$

$$\bar{F} = \begin{bmatrix} I_{p \times p} & 0_{p \times p} & \dots & 0_{p \times p} & 0_{p \times p} \\ I_{p \times p} & 0_{p \times p} & \dots & 0_{p \times p} & 0_{p \times p} \\ 0_{p \times p} & I_{p \times p} & \dots & 0_{p \times p} & 0_{p \times p} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0_{p \times p} & 0_{p \times p} & \dots & I_{p \times p} & 0_{p \times p} \end{bmatrix},$$

$$\bar{V}_1(\tau_{t,k}) = \begin{bmatrix} V_1(\tau_{t,k}) \\ 0_{n \times n(d+1)} \\ \vdots \\ 0_{n \times n(d+1)} \end{bmatrix}, \bar{V}_2(\tau_{t,k}) = \begin{bmatrix} V_2(\tau_{t,k}) \\ 0_{p \times p(d+1)} \\ \vdots \\ 0_{p \times p(d+1)} \end{bmatrix}.$$

Система (28) представляет собой дискретную $2D$ -систему Роессера марковской структуры (марковская система с переключениями) фиксированного размера и свободную от запаздыва-

ния. Для (28) подразумевается, что матрицы управления $K_1(0)$ и $K_2(0)$ известны.

Обозначим матрицы $\Phi(0)$ и $\Phi(i)$, вектора состояния системы $x(t, k)$ и $x'(t, k)$ как

$$(29) \quad \Phi(0) = \begin{bmatrix} A + BK_1(0) & BK_2(0) \\ -CA - CBK_1(0) & I - CBK_2(0) \end{bmatrix},$$

$$(30) \quad \bar{\Phi}(i) = \begin{bmatrix} \bar{A} + \bar{B}\bar{K}_1(0) & \bar{B}\bar{K}_2(0) \\ -\bar{D}\bar{V}_1(\tau_{t,k}) - \bar{N}K_1(\tau_{t,k})\bar{S}_1(\tau_{t,k}) & \bar{F} - \bar{N}K_2(\tau_{t,k})\bar{S}_2(\tau_{t,k}) \end{bmatrix},$$

$i = 1, 2, \dots, d$.

$$(31) \quad x(t, k) = \begin{bmatrix} \bar{\eta}(t, k) \\ \bar{e}(t, k) \end{bmatrix} = \begin{bmatrix} x_1(t, k) \\ x_2(t, k) \end{bmatrix}, \quad x'(t, k) = \begin{bmatrix} \bar{\eta}(t+1, k) \\ \bar{e}(t, k+1) \end{bmatrix} = \begin{bmatrix} x_1(t+1, k) \\ x_2(t, k+1) \end{bmatrix}.$$

Система (28) является стохастической, поэтому для нее необходимо соответствующим образом ввести понятие стохастической устойчивости.

Определение 1. Рассмотрим 2D-систему (28) с граничными условиями (21), которая удовлетворяет

$$(32) \quad M \sum_{i=0}^{\infty} \|x(i+1, 0)\|^2 < \infty.$$

Тогда будем говорить, что система управления (28) стохастически устойчива, если

$$\lim_{t+k \rightarrow \infty} M \|x(t, k)\|^2 = 0.$$

Из определения следует, что:

$$(33) \quad \lim_{k \rightarrow \infty} M \{ \|x(t, k)\| \} = \lim_{k \rightarrow \infty} x(t, k) = 0,$$

$$(34) \quad \lim_{k \rightarrow \infty} \begin{bmatrix} \bar{\eta}(t, k) \\ \bar{e}(t, k) \end{bmatrix} = \lim_{k \rightarrow \infty} \begin{bmatrix} \eta(t, k) \\ e(t, k) \end{bmatrix} = 0,$$

$$(35) \quad \lim_{k \rightarrow \infty} e(t, k) = 0,$$

что является целью итеративного обучающего управления.

3. Основные результаты

Следующая теорема дает достаточные условия стохастической устойчивости для системы (28) в терминах линейных матричных неравенств.

Теорема 1 (достаточные условия устойчивости). *Стохастическая система Роессера (28) марковской структуры устойчива, если существуют матрицы $P(i) = \text{diag}\{P_1, P_2(i)\} > 0$, $i = \tau_{t,k} = \{1, 2, \dots, d\}$, и выполняются следующие матричные неравенства:*

$$(36) L(i) = \bar{\Phi}(i)^T \bar{P}(j) \bar{\Phi}(i) - P(i) < 0,$$

$$\text{где } \bar{P}(j) = \text{diag}\left\{P_1, \sum_{j=0}^d p_{ij} P_2(j)\right\}, P_1 = P_1^T > 0, P_2(i) = P_2(i)^T > 0.$$

Доказательство.

Пусть функция Ляпунова для системы (28) представлена в виде суммы функций Ляпунова по векторам состояний $x_1(t, k)$ и $x_2(t, k)$:

$$(37) V(x(t, k)) = V_1(x_1(t, k)) + V_2(x_2(t, k)),$$

где

$$(38) V_1(x_1(t, k), k) = x_1^T(t, k) P_1(\tau_{t,k}) x_1(t, k),$$

$$V_2(x_2(t, k), k) = x_2^T(t, k) P_2(\tau_{t,k}) x_2(t, k).$$

$$(39) \Delta V(x(t, k)) = M\{V_1(x_1(t+1, k)) + V_2(x_2(t, k+1)) - V_1(x_1(t, k)) - V_2(x_2(t, k)) |_{\tau_{t,k}=i}\} =$$

$$= M\{V_1(x_1(t+1, k)) + V_2(x_2(t, k+1)) |_{\tau_{t,k}=i}\} - [V_1(x_1(t, k), \tau_{t,k} = i) + V_2(x_2(t, k), \tau_{t,k} = i)] =$$

$$= x_1^T(t+1, k) M\{P_1(\tau_{t+1,k}) |_{\tau_{t,k}=i}\} x_1(t+1, k) +$$

$$+ x_2^T(t, k+1) M\{P_2(\tau_{t,k+1}) |_{\tau_{t,k}=i}\} x_2(t, k+1) -$$

$$- x_1^T(t, k) P_1(\tau_{t,k} = i) x_1(t, k) - x_2^T(t, k) P_2(\tau_{t,k} = i) x_2(t, k).$$

$$(40) M\{P_2(\tau_{t,k+1}) |_{\tau_{t,k}=i}\} = \sum_{j=0}^d p_{ij} P_2(j) = \bar{P}_2(j) \text{ для } t = 0, 1, \dots, N.$$

$$\begin{aligned}
 (41) \quad \Delta V(x(t, k)) &= \\
 &= \begin{bmatrix} x_1(t+1, k) \\ x_2(t, k+1) \end{bmatrix}^T \cdot \begin{bmatrix} P_1(i) & 0 \\ 0 & \bar{P}_2(j) \end{bmatrix} \cdot \begin{bmatrix} x_1(t+1, k) \\ x_2(t, k+1) \end{bmatrix} - \\
 &- \begin{bmatrix} x_1(t, k) \\ x_2(t, k) \end{bmatrix}^T \cdot \begin{bmatrix} P_1(i) & 0 \\ 0 & P_2(i) \end{bmatrix} \cdot \begin{bmatrix} x_1(t, k) \\ x_2(t, k) \end{bmatrix} = \\
 &= x^T(t, k) \text{diag}\{P_1(i) \quad \bar{P}_2(i)\}x(t, k) - x^T(t, k) \text{diag}\{P_1(i) \quad P_2(i)\}x(t, k) = \\
 &= x^T(t, k) [\bar{\Phi}(i)^T \bar{P}(j) \bar{\Phi}(i) - P(i)]x(t, k) = x^T(t, k) L(i)x(t, k).
 \end{aligned}$$

Для $L(i) < 0$ из (41) с учетом $x(t, k) \neq 0$ имеем:

$$\begin{aligned}
 (43) \quad \frac{M\{V(x'(t, k))|_{\tau_{t,k=i}}\} - V(x(t, k), \tau_{t,k} = i)}{V(x(t, k), \tau_{t,k} = i)} &< -\frac{x^T(t, k)L(i)x(t, k)}{x^T(t, k)P(i)x(t, k)} \leq \\
 &\leq -\min_{i \in H} \left\{ \frac{\lambda(L(i))}{\lambda(P(i))} \right\} = \beta - 1.
 \end{aligned}$$

$$(44) \quad \beta = 1 - \min_{i \in H} \left\{ \frac{\lambda_m(L(i))}{\lambda_M(P(i))} \right\}.$$

Из (43) и (44) получаем:

$$(45) \quad \beta > \frac{M\{V(x'(t, k))|_{\tau_{t,k=i}}\}}{V(x(t, k), \tau_{t,k} = i)} > 0.$$

С учетом (43), $0 < \beta < 1$. Отсюда

$$\begin{aligned}
 (46) \quad M\{V(x'(t, k))|_{\tau_{t,k=i}}\} &< \beta \cdot M\{V(x(t, k), \tau_{t,k} = i)\}. \\
 M\{V_1(x_1(t+1, k)) + V_2(x_2(t, k+1))|_{\tau_{t,k=i}}\} &< \\
 &< \beta \cdot M\{V_1(x_1(t, k), \tau_{t,k} = i) + V_2(x_2(t, k), \tau_{t,k} = i)\}.
 \end{aligned}$$

Как и в [27], рассмотрим сумму всевозможных функций $V(x(t, k))$ на множестве $D(r)$:

$$(47) \quad D(r) = \{(t, k) : t + k = r, t \geq 1, k \geq 0\}.$$

$$\begin{aligned}
 (48) \quad \sum_{t+k \in D(r+1)} M\{V(x(t, k))\} &= M\{V_1(x_1(r+1, 0))\} + M\{V_1(x_1(r, 1))\} + \dots \\
 &\dots + M\{V_1(x_1(2, r-1))\} + M\{V_1(x_1(1, r))\} + M\{V_2(x_2(r+1, 0))\} + \\
 &+ M\{V_2(x_2(r, 1))\} + M\{V_2(x_2(r-1, 2))\} + \dots + M\{V_2(x_2(1, r))\} =
 \end{aligned}$$

$$\begin{aligned}
 &= M \sum_{j=0}^r \left[V_1(x_1(r+1-j, j)) + V_2(x_2(r+1-j, j)) \Big|_{\tau_{r+1-j, j=m}} \right] \\
 (49) \quad &\sum_{t+k \in D(r)} M \{V(x(t, k))\} = M \{V_1(x_1(r, 0))\} + M \{V_1(x_1(r-1, 1))\} + \dots \\
 &+ M \{V_1(x_1(1, r-1))\} + M \{V_2(x_2(r, 0))\} + M \{V_2(x_2(r-1, 1))\} + \dots \\
 &\dots + M \{V_2(x_2(1, r-1))\} = \\
 &= M \sum_{j=0}^r \left[V_1(x_1(r-j, j)) + V_2(x_2(r-j, j)) \Big|_{\tau_{r-j, j=l}} \right] \\
 &M \sum_{j=0}^r \left[V_1(x_1(r+1-j, j)) + V_2(x_2(r+1-j, j)) \Big|_{\tau_{r+1-j, j=m}} \right] < \\
 &< \beta \cdot M \sum_{j=0}^r \left[V_1(x_1(r-j, j)) + V_2(x_2(r-j, j)) \Big|_{\tau_{r-j, j=l}} \right]
 \end{aligned}$$

Отсюда

$$\begin{aligned}
 (50) \quad &M \sum_{j=0}^r \left[V_1(x_1(r+1-j, j)) + V_2(x_2(r+1-j, j)) \Big|_{\tau_{r+1-j, j=m}} \right] < \\
 &< M \sum_{j=0}^r \beta^j \left[V_1(x_1(1, r), \tau_{1,r}) + V_2(x_2(r+1, 0), \tau_{r+1,0}) \right]
 \end{aligned}$$

Для $r \geq 0$ и последнего неравенства имеем:

$$\begin{aligned}
 (51) \quad &\sum_{r=0}^N M \sum_{j=0}^r \left[V_1(x_1(r+1-j, j)) + V_2(x_2(r+1-j, j)) \Big|_{\tau_{r+1-j, j=m}} \right] < \\
 &< (1 + \beta + \dots + \beta^N) M \{V_1(x_1(1, 0), \tau_{1,0}) + V_2(x_2(1, 0), \tau_{1,0})\} + \\
 &< (1 + \beta + \dots + \beta^{N-1}) M \{V_1(x_1(1, 1), \tau_{1,1}) + V_2(x_2(2, 0), \tau_{2,0})\} + \\
 &+ \dots + M \{V_1(x_1(1, N), \tau_{1,N}) + V_2(x_2(N+1, 0), \tau_{N+1,0})\} < \\
 &< (1 + \beta + \dots + \beta^N) M \sum_{r=0}^N \left[V_1(x_1(1, N), \tau_{1,N}) + V_2(x_2(N+1, 0), \tau_{N+1,0}) \right]
 \end{aligned}$$

В нашем случае, согласно (21), $V_1(x_1(1, N)) = 0$, $N = 0, 1, \dots$

$$\begin{aligned}
 (52) \quad &\sum_{r=0}^N M \sum_{j=0}^r \left[V_1(x_1(r+1-j, j)) + V_2(x_2(r+1-j, j)) \Big|_{\tau_{r+1-j, j=m}} \right] < \\
 &< (1 + \beta + \dots + \beta^N) M \sum_{r=0}^N \left[V_2(x_2(N+1, 0), \tau_{N+1,0}) \right] <
 \end{aligned}$$

$$< (1 + \beta + \dots + \beta^N) M \sum_{r=0}^N \left[V_1(x_1(N+1,0)) + V_2(x_2(N+1,0)) \Big|_{\tau_{N+1,0}=h} \right].$$

Из (52) следует, что

$$(53) \sum_{r=0}^N M \sum_{j=0}^r \left[x^T(r+1-j, j) P(m) x(r+1-j, j) \right] < < (1 + \beta + \dots + \beta^N) M \sum_{r=0}^N \left[x^T(r+1,0) P(h) x(r+1,0) \right].$$

Отсюда

$$(54) \sum_{r=0}^N M \sum_{j=0}^r x^T(r+1-j, j) x(r+1-j, j) < < \alpha (1 + \beta + \dots + \beta^N) M \sum_{r=0}^N \left[x^T(r+1,0) x(r+1,0) \right]$$

где $\alpha = \frac{\max_{m \in H} \lambda(P(m))}{\min_{m \in H} \lambda(P(h))} > 0$.

$$(55) \sum_{r=0}^N M \sum_{j=0}^r \|x(r+1-j, j)\|^2 < \alpha (1 + \beta + \dots + \beta^N) M \sum_{r=0}^N \|x(r+1,0)\|^2.$$

Из (32) и $0 < \beta < 1$ следует, что правая часть неравенства (55) ограничена при $N \rightarrow \infty$. Поэтому ряд левой части неравенства сходится

$$(56) M \sum_{j=0}^r \|x(r+1-j, j)\|^2 \rightarrow 0,$$

при $r \rightarrow \infty$. Переходя к пределу для (56), получаем стохастическую устойчивость системы (28).

Аналогичным образом можно установить достаточные условия устойчивости для системы без учета запаздывания (27).

Теорема 2 (2D-система (27) относительно $\tau_{i,k} = 0$). Дискретная 2D-система Роессера (27) устойчива, если существуют матрицы $P(0) = \text{diag}\{P_1, P_2(0)\} > 0$ и выполняются следующие матричные неравенства:

$$(57) L(0) = \Phi(0)^T \bar{P}(0) \Phi(0) - P(0) < 0,$$

где $\bar{P}(0) = \text{diag} \left\{ P_1, \sum_{j=0}^d p_{0j} P_2(j) \right\}$, $P_1 = P_1^T > 0$, $P_2(0) = P_2(0)^T > 0$.

Для вычисления матриц усиления $K_1(\tau_{t,k})$, $K_2(\tau_{t,k})$, $\tau_{t,k} = \{1, 2, \dots, d\}$, установлен следующий результат на основании теоремы 1 и преобразовании линейных матричных неравенств (48).

Теорема 3 (нахождение матриц управления $K_1(i)$ и $K_2(i)$). Стохастическая система Роессера (28) стохастически устойчива, если существуют симметричные матрицы $X_1 > 0$, $X_2(i) > 0$, $i = \tau_{t,k} = \{1, 2, \dots, d\}$, $\bar{Z} = \text{diag} \{X_2(0), \dots, X_2(d)\} > 0$ и $T_1(i)$, $T_2(i)$, такие что выполняются следующие линейные матричные неравенства:

(58) $M(i) =$

$$\begin{bmatrix} -X_1 & 0 & [\bar{A}X_1 + \bar{B}_1X_1]^T & [-\bar{D}\bar{V}_1(i)X_1 - \bar{N}T_1(i)\bar{S}_1(i)]^T W(i) \\ 0 & -X_2(i) & [\bar{B}_2X_2(i)]^T & [\bar{F}X_2(i) - \bar{N}T_2(i)\bar{S}_2(i)]^T W(i) \\ \bar{A}X_1 + \bar{B}_1X_1 & \bar{B}_2X_2(i) & -X_1 & 0 \\ W^T(i) [-\bar{D}\bar{V}_1(i)X_1 - \bar{N}T_1(i)\bar{S}_1(i)] & W^T(i) [\bar{F}X_2(i) - \bar{N}T_2(i)\bar{S}_2(i)] & 0 & -\bar{Z} \end{bmatrix} < 0,$$

$$\bar{S}_1(i)X_1 = Y_1(i)\bar{S}_1(i), \bar{S}_2(i)X_2(i) = Y_2(i)\bar{S}_2(i), \bar{B}_1 = \bar{B}K_1(0),$$

$$\bar{B}_2 = \bar{B}K_2(0),$$

$$W(i) = \left(\sqrt{p_{i0}} I, \dots, \sqrt{p_{id}} I \right), i = 1, \dots, d.$$

Если выполняется (58), тогда матрицы управления $K_1(i)$ и $K_2(i)$ могут быть вычислены как

$$(59) K_1(i) = T_1(i)Y_1(i)^{-1}, K_2(i) = T_2(i)Y_2(i)^{-1}.$$

Доказательство.

Определим следующие матрицы:

$$(60) W(i) = \left(\sqrt{p_{i0}} I_{p \times p}, \dots, \sqrt{p_{id}} I_{p \times p} \right),$$

$$(61) Zn = \text{diag} \{P_2(0), \dots, P_2(d)\}.$$

Выражение для $\bar{P}(j)$ теоремы 1 может быть переписано следующим образом:

$$(62) \bar{P}(j) = \text{diag} \left\{ P_1, \sum_{j=0}^d p_{ij} P_2(j) \right\} = \bar{W}(i) \cdot \bar{Z} \cdot \bar{W}^T(i),$$

где $\bar{W}(i) = \begin{bmatrix} I & 0 \\ 0 & W(i) \end{bmatrix}$, $\bar{Z}_n = \begin{bmatrix} P_1 & 0 \\ 0 & Z_n \end{bmatrix}$.

В соответствии с (48) теоремы 1, получаем:

(63) $\bar{\Phi}(i)^T \bar{W}(i) \cdot \bar{Z}_n \cdot \bar{W}^T(i) \bar{\Phi}(i) - P(i) < 0$,

где $\bar{\Phi}(i) = \begin{bmatrix} \bar{A} + \bar{B}\bar{K}_1(0) & \bar{B}\bar{K}_2(0) \\ -\bar{D}\bar{V}_1(\tau_{t,k}) - \bar{N}\bar{K}_1(\tau_{t,k})\bar{S}_1(\tau_{t,k}) & \bar{F} - \bar{N}\bar{K}_2(\tau_{t,k})\bar{S}_2(\tau_{t,k}) \end{bmatrix}$,

$i = 1, 2, \dots, d$.

Дополнения Шура [8] для формулы (59) дают следующие линейные матричные неравенства:

(64) $M(i) =$

$$\begin{bmatrix} -P_1 & 0 & [\bar{A} + \bar{B}_1 X_1]^T & [-\bar{D}\bar{V}_1(i) - \bar{N}\bar{K}_1(i)\bar{S}_1(i)]^T W(i) \\ 0 & -P_2(i) & [\bar{B}_2]^T & [\bar{F} - \bar{N}\bar{K}_2(i)\bar{S}_2(i)]^T W(i) \\ \bar{A} + \bar{B}_1 & \bar{B}_2 & -P_1 & 0 \\ W^T(i) [-\bar{D}\bar{V}_1(i) - \bar{N}\bar{K}_1(i)\bar{S}_1(i)] & W^T(i) [\bar{F} - \bar{N}\bar{K}_2(i)\bar{S}_2(i)] & 0 & -\bar{Z}_n^{-1} \end{bmatrix} < 0$$

Введем матрицы $X_1 = P_1^{-1}$, $X_2(i) = P_2^{-1}(i)$, и домножим слева и справа неравенство (63) на матрицу $diag\{X_1, X_2(i), I, I\}$, получим:

(65) $M(i) =$

$$\begin{bmatrix} -X_1 & 0 & [\bar{A}X_1 + \bar{B}_1 X_1]^T & [-\bar{D}\bar{V}_1(i)X_1 - \bar{N}\bar{K}_1(i)\bar{S}_1(i)X_1]^T W(i) \\ 0 & -X_2(i) & [\bar{B}_2 X_2(i)]^T & [\bar{F}X_2(i) - \bar{N}\bar{K}_2(i)\bar{S}_2(i)X_2(i)]^T W(i) \\ \bar{A}X_1 + \bar{B}_1 X_1 & \bar{B}_2 X_2(i) & -X_1 & 0 \\ W^T(i) [-\bar{D}\bar{V}_1(i)X_1 - \bar{N}\bar{K}_1(i)\bar{S}_1(i)X_1] & W^T(i) [\bar{F}X_2(i) - \bar{N}\bar{K}_2(i)\bar{S}_2(i)X_2(i)] & 0 & -\bar{Z} \end{bmatrix} < 0$$

где $\bar{Z} = diag\{X_2(0), \dots, X_2(d)\}$.

Для линейных матричных неравенств (65) применим метод, представленный Cesar A.R. Crusius и Alexandre Trofino [11].

Пусть

$\bar{S}_1(i)X_1 = Y_1(i)\bar{S}_1(i)$, $T_1(i) = K_1(i)Y_1(i)$ и

$\bar{S}_2(i)X_2(i) = Y_2(i)\bar{S}_2(i)$, $T_2(i) = K_2(i)Y_2(i)$,

тогда после несложных преобразований получим окончательные линейные матричные неравенства вида:

(66) $M(i) =$

$$\begin{bmatrix} -X_1 & 0 & [\bar{A}X_1 + \bar{B}_1 X_1]^T & [-\bar{D}\bar{V}_1(i)X_1 - \bar{N}T_1(i)\bar{S}_1(i)]^T W(i) \\ 0 & -X_2(i) & [\bar{B}_2 X_2(i)]^T & [\bar{F}X_2(i) - \bar{N}T_2(i)\bar{S}_2(i)]^T W(i) \\ \bar{A}X_1 + \bar{B}_1 X_1 & \bar{B}_2 X_2(i) & -X_1 & 0 \\ W^T(i) [-\bar{D}\bar{V}_1(i)X_1 - \bar{N}T_1(i)\bar{S}_1(i)] & W^T(i) [\bar{F}X_2(i) - \bar{N}T_2(i)\bar{S}_2(i)] & 0 & -\bar{Z} \end{bmatrix} < 0$$

$$\bar{S}_1(i)X_1 = Y_1(i)\bar{S}_1(i), \quad \bar{S}_2(i)X_2(i) = Y_2(i)\bar{S}_2(i).$$

Тогда матрицы управления $K_1(i)$ и $K_2(i)$ определяются как

$$(67) \quad K_1(i) = T_1(i)Y_1(i)^{-1}, \quad K_2(i) = T_2(i)Y_2(i)^{-1}.$$

Аналогичным образом можно найти матрицы управления $K_1(0)$ и $K_2(0)$ для системы (27).

Теорема 4 (нахождение матриц управления $K_1(0)$ и $K_2(0)$).

Дискретная 2D-система Роессера (27) устойчива, если существуют симметричные матрицы $X_1 > 0$, $X_2(0) > 0$, $\bar{Z} = \text{diag}\{X_2(0), \dots, X_2(d)\} > 0$ и $T_1(0)$, $T_2(0)$, такие что выполняются следующие линейные матричные неравенства:

$$(68) \quad M(0) =$$

$$\begin{bmatrix} -X_1 & 0 & [AX_1 + BT_1(0)]^T & [-CA X_1^T - CB T_1(0)]^T W(0) \\ 0 & -X_2(0) & [BT_2(0)]^T & [X_2(0) - CB T_2(0)]^T W(0) \\ AX_1 + BT_1(0) & BT_2(0) & -X_1 & 0 \\ W^T(0)[-CA X_1^T - CB T_1(0)] & W^T(0)[X_2(0) - CB T_2(0)] & 0 & -\bar{Z} \end{bmatrix} < 0$$

Если выполняется (68), тогда матрицы управления $K_1(0)$ и $K_2(0)$ могут быть вычислены как:

$$(69) \quad K_1(0) = T_1(0)X_1^{-1}, \quad K_2(0) = T_2(0)X_2(0)^{-1}.$$

4. Численный пример

Для иллюстрации работы предлагаемого метода полученные результаты были применены к модели двигателя постоянного тока. Рассматривается двигатель, где ротор работает под действием источника постоянного тока, а величина напряженности магнитного поля на обмотке полюсов переменная [16]. Управление углом вращения двигателя осуществляется при помощи изменения напряжения источника тока, подсоединенного к обмотке полюсов. Двигатель вращает механическую нагрузку (рис. 3).

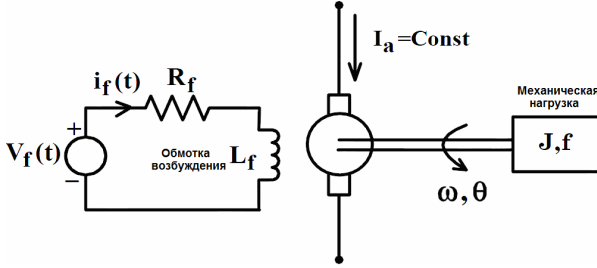


Рис. 3. Модель электродвигателя постоянного тока

Для данной модели приводятся уравнения пространства состояний [12]:

$$(70) \quad \begin{aligned} \dot{x}(t) &= Ax(t) + Bv_f(t), \\ y(t) &= Cx(t), \quad t \geq 0; \end{aligned}$$

где $x(t) = [i_f(t) \quad \omega(t) \quad \theta(t)]$, $y(t) = \theta(t)$,

$$A = \begin{bmatrix} -\frac{R_f}{L_f} & 0 & 0 \\ \frac{k_m}{J} & -\frac{f}{J} & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} \frac{1}{L_f} \\ 0 \\ 0 \end{bmatrix}, \quad C = [0 \quad 0 \quad 1],$$

где R_f и L_f – сопротивление обмотки полюса и индуктивность; k_m – крутящий момент двигателя; J и f – момент инерции механической нагрузки и коэффициент трения соответственно. Также $v_f(t)$, $i_f(t)$ – напряжение источника тока для обмотки и текущее напряжение в контуре; $\omega(t)$ и $\theta(t)$ – частота вращения электродвигателя и угол вращения соответственно.

Для принятой модели необходимо определить приводящее напряжение $v_f(t)$, чтобы выходной сигнал электродвигателя периодически соответствовал заданному сигналу $y_d(t)$ на интервале времени и ошибка обучения с увеличением количества итерации сводилась к нулю.

Уравнения состояния электродвигателя можно привести к дискретной форме, введя период дискретизации $T = 0,01$ с и устанавливая следующие значения параметров модели (70) [12]:

$$(71) R_f = 20 \text{ Ом}; L_f = 1 \text{ Гн}; k_m = 100 \frac{\text{Н} \cdot \text{м}}{\text{А}}; f = 0,5 \frac{\text{Н} \cdot \text{мс}}{\text{рад}};$$

$$J = 2 \frac{\text{Н} \cdot \text{мс}^2}{\text{рад}}; t_f = 12 \text{ с}.$$

Уравнения состояния в дискретной форме записываются как

$$(72) \begin{cases} x(t+1, k) = A_D x(t, k) + B_D u(t, k), \\ y(t, k) = C_D x(t, k), \\ t = 0, 1, \dots, 1200, k = 0, 1, \dots; \end{cases}$$

где

$$A_D = \begin{bmatrix} 0,8187 & 0 & 0 \\ 0,4526 & 0,9975 & 0 \\ 0,0023 & 0,01 & 1 \end{bmatrix}, B_D = \begin{bmatrix} 0 \\ 0,0197 \\ 0,0211 \end{bmatrix}, C_D = [0 \ 0 \ 1].$$

Точная траектория системы (72) показана на рис. 4 и имеет вид:

$$(73) y_{ref}(i) = 10 \left(1 + \sin \left(\frac{2\pi}{M} i - \frac{\pi}{2} \right) \right), \quad 1 \leq i \leq M, \quad M = \frac{t_f}{T} = 1200.$$

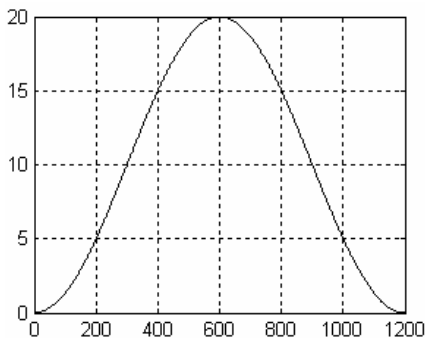


Рис. 4. Точная траектория системы

Исходная система в разомкнутом состоянии неустойчива, вектор ее собственных значений равен $[1,000; 0,9976; 0,8187]$.

Управление и передача данных осуществляется через сеть связи предложенной сетевой архитектуры (рис. 2). Предполагается, что случайное запаздывание $\tau_{t,k}$, возникающее в сетевой системе на стороне датчик-регулятор, принимает значения из диапазона $\tau_{t,k} = \{0, 1, 2, 3\}$, а матрица вероятностей переходов (переход из состояния i в j) в дискретный момент времени t задается следующим образом:

$$(74) \text{Pr} = \begin{bmatrix} 0,8 & 0,2 & 0,0 & 0,0 \\ 0,6 & 0,3 & 0,1 & 0,0 \\ 0,5 & 0,3 & 0,1 & 0,1 \\ 0,5 & 0,3 & 0,1 & 0,1 \end{bmatrix}.$$

Тогда

$$W(0) = [0,8 \ 0,2 \ 0,0 \ 0,0], \quad W(1) = [0,6 \ 0,3 \ 0,1 \ 0,0], \\ W(2) = [0,5 \ 0,3 \ 0,1 \ 0,1], \quad W(3) = [0,5 \ 0,3 \ 0,1 \ 0,1].$$

Запаздывание $\tau_{t,k}$ может увеличиваться не более чем на один шаг и до максимального значения $\tau_{t,k} = 3$, а уменьшатся на сколько угодно возможное количество шагов.

Наличие у регулятора принимающего буфера, который содержит последнюю информацию, полученную от датчика, упрощает закон управления, связанный с процессом итерационного обучения для системы с запаздыванием $\tau_{t,k} = \{1, 2, 3\}$.

Приращение управления (17), связанное с процессом итерационного обучения, представляется в виде:

для $\tau_{t,k} = 0$:

$$(75) u(t, k+1) = u(t, k) + K_1(0)[x(t, k+1) - x(t, k)] + \\ + K_2(0)[y_r(t, k+1) - y(t, k+1)].$$

для $\tau_{t,k} = 1, 2, \dots, d$:

$$(76) u(t, k+1) = u(t, k) + K_2(\tau_{t,k})[y_r(t, k+1) - y(t, k+1 - \tau_{t,k+1})].$$

Закон управления (75) не учитывает матрицу $K_1(\tau_{t,k})$.

Значение сигнала управления на первой итерации принималось $u(t, 0) = 10$, точность сходимости итерационного обу-

чающего процесса eps на каждом дискретном промежутке времени достигает величины $eps = 10^{-4}$.

Линейные матричные неравенства теоремы 3 дают множество решений. Значения матрицы $K_2(\tau_{i,k})$ могут оказаться очень малыми, поэтому сходимость обучающей схемы будет очень медленной. Одним из способов максимизации значений $K_2(\tau_{i,k})$ является минимизация целевой функции для линейных матричных неравенств (58)–(59) [13]:

$$(77) f(T_2(i)Y_2(i)) = -T_2(i) + Y_2(i)h, \quad i = 1, 2, 3,$$

где $T_2(i)$ и $Y_2(i)$ – матрицы теоремы 3; h – задаваемый положительный скаляр.

При решении линейных матричных неравенств (58)–(59), (68) для рассматриваемой модели двигателя постоянного тока с принятыми характеристиками сетевой системы, с учетом (77) и $h = 0,01$, были получены соответствующие матрицы управления (75), (76). Вычисления проводились в среде *MATLAB* с применением пакетов *YALMIP* и *SEDUMI* при относительной погрешности $\varepsilon = 1e - 7$.

$$K_1(0) = [14,4420 \quad -3,0985 \quad -43,3439]; \quad K_2(0) = 15,0146;$$

$$K_2(1) = 3,9687; \quad K_2(2) = 4,0165; \quad K_2(3) = 2,1462.$$

На рис. 5 и рис. 6 представлены графики моделирования запаздывания $\tau_{i,k}$, описываемого с помощью марковских цепей, на промежутке времени $t = 100$ и $t = 1000$, что соответствует 0,1 с и 1 с соответственно.

На рис. 7 и рис. 8 показаны графики суммарной ошибки обучения по итерациям и ошибки обучения на каждой итерации по всем промежуткам времени.

На основании графика, представленного на рис. 7, можно отметить, что алгоритм обучения обеспечивает монотонную сходимость суммарной ошибки обучения.

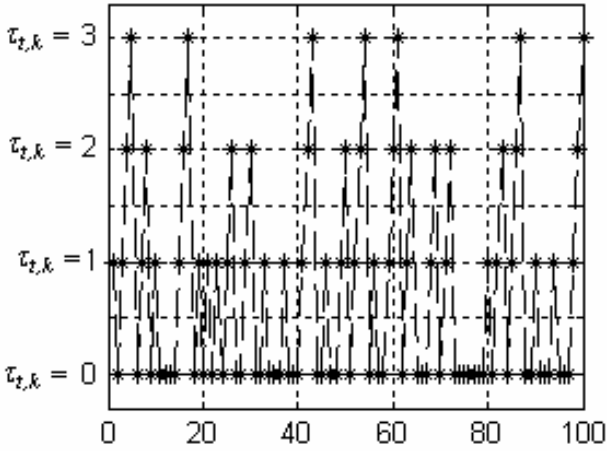


Рис. 5. Запозывание $\tau_{t,k}$ на промежутке времени $t = 100$

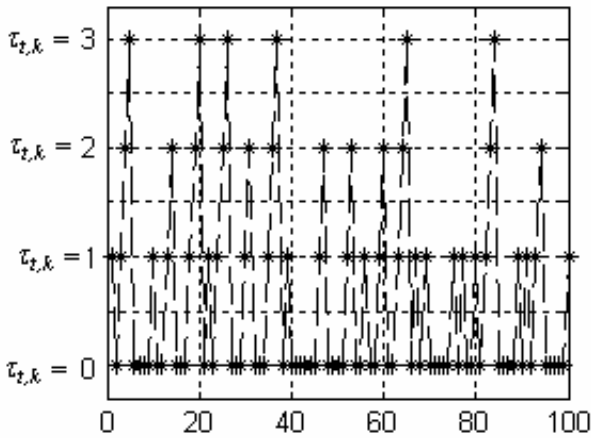


Рис. 6. Запозывание $\tau_{t,k}$ на промежутке времени $t = 1000$

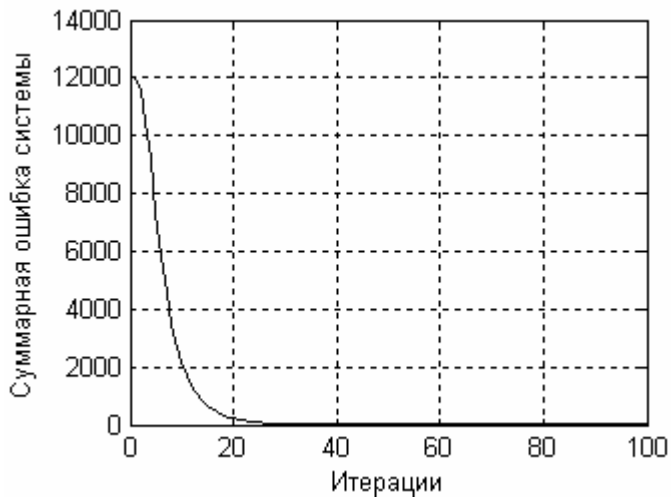


Рис. 7. Суммарная ошибка обучения по итерациям

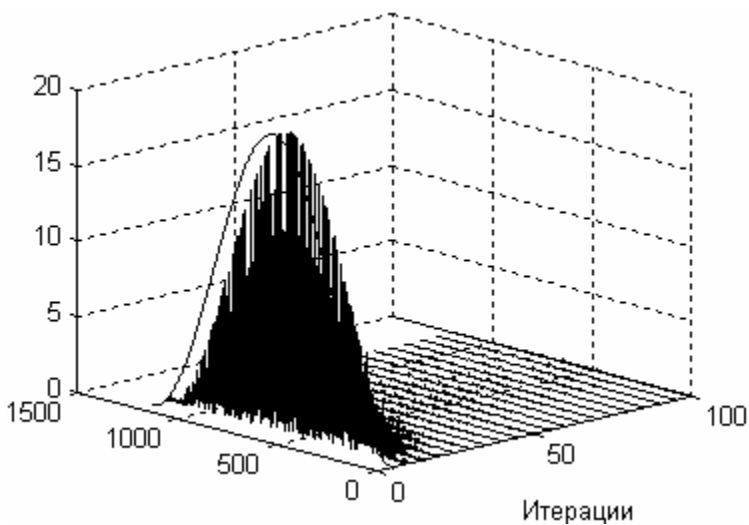


Рис. 8. Ошибка обучения на каждой итерации по всем промежуткам времени

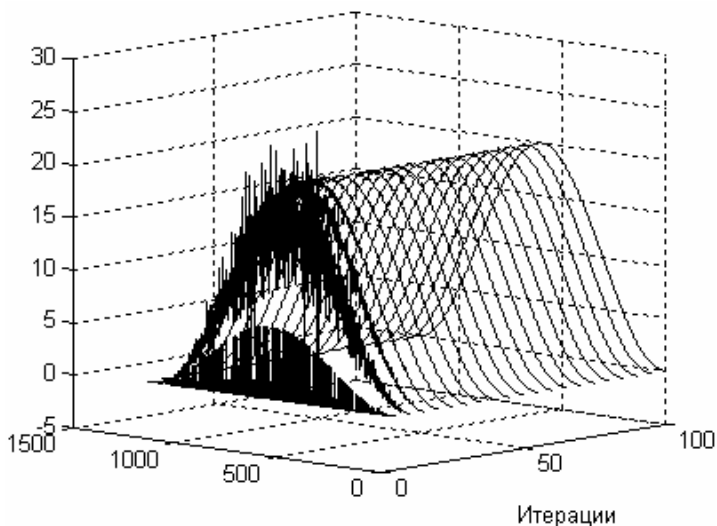


Рис. 9. Полученные траектории системы

На рис. 9 представлены траектории системы, полученные в результате реализации итерационного процесса обучения при передаче данных через сетевой канал связи. По данным графика можно увидеть, что в начальный период обучения системы траектории имеют скачкообразный характер. Это связано с наличием запаздывания и потерей пакетов данных в сети связи. При увеличении итераций ошибка обучения уменьшается, прослеживается точная траектория системы. Необходимо также отметить тот факт, что в течение итерационного обучения точная траектория системы прописывается и в буфер обмена регулятора.

На рис. 10 и рис. 11 показаны графики суммарной ошибки обучения на всем отрезке времени на каждой итерации и ошибки обучения на каждой итерации по всем промежуткам времени для системы без учета запаздывания и потерь пакетов данных в сетевом канале связи.

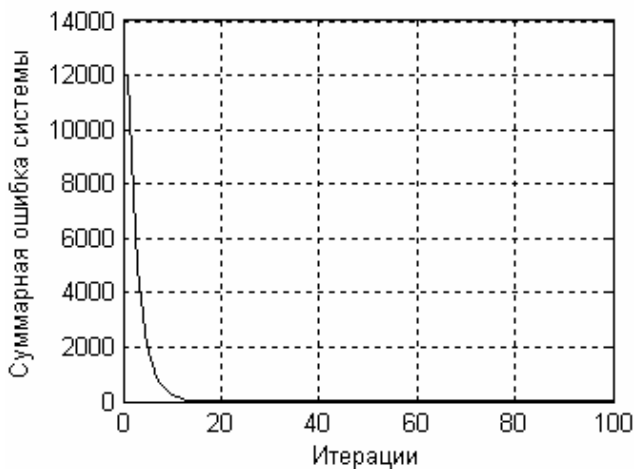


Рис. 10. Суммарная ошибка обучения по итерациям для системы без учета запаздывания и потери пакетов данных

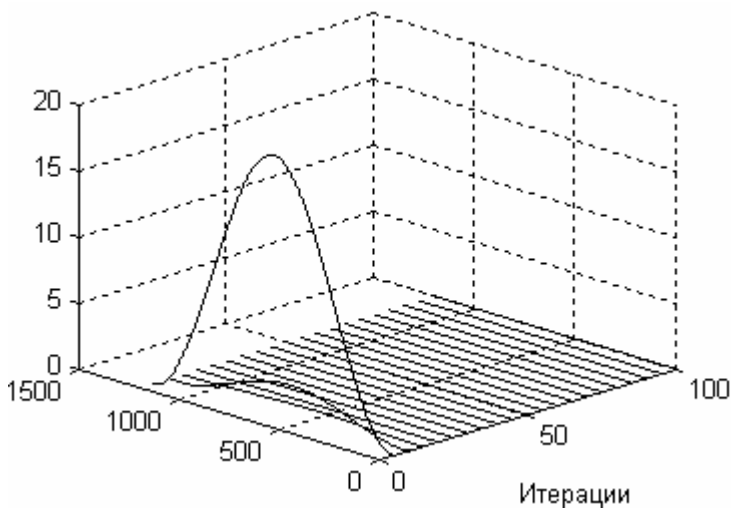


Рис. 11. Ошибка обучения на каждой итерации по всем промежуткам времени для системы без учета запаздывания и потери пакетов данных

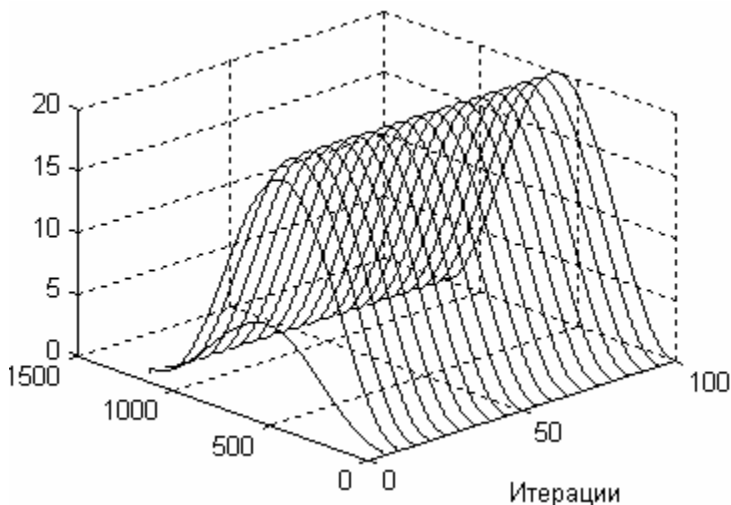


Рис. 12. Полученные траектории системы для системы без учета запаздывания и потери пакетов данных.

На рис. 12 представлены траектории системы, полученные в результате реализации итерационного процесса обучения при отсутствии запаздывания и потери пакета данных в сетевом канале связи. По графику видно, что не наблюдается скачков траекторий системы. На основании графика, представленного на рис. 10, можно отметить, что алгоритм обучения, при отсутствии запаздывания, также обеспечивает монотонную сходимость суммарной ошибки обучения.

Результаты численного моделирования двигателя постоянного тока показывают, что предлагаемый метод имеет неплохую сходимость и обеспечивает устойчивость системы во время итераций.

Литература

1. АНДРИЕВСКИЙ Б.Р., МАТВЕЕВ А.С., ФРАДКОВ А.Л. *Управление и оценивание при информационных ограничениях: к единой теории управления, вычислений и связи* // Автоматика и телемеханика. – 2010. – №4. – С. 34–99.

2. ВАЛЮБ С.А. *Построение моделей сетевого управления для систем с итерационным обучением и повторяющихся процессов* // Управление большими системами. Материалы VII Всероссийской школы-конференции молодых ученых. – ПГТУ, 2010. – Т. 1. – С. 15–22.
3. AHN H.S, CHEN Y., MOORE K.L. *Iterative learning control: brief survey and categorization* // IEEE Transactions on Systems, Man and Cybernetics. – 2007. – Part C, Vol. 37, №2. – P. 1109–1121.
4. AMANN N., OWENS D.H., ROGERS E. *2D systems theory applied to learning control systems* // Proceedings of the 33rd IEEE Conference on Decision and Control, Lake Buena Vista, Florida, USA, 1994. – P. 985–986.
5. ARIMOTO S., KAWAMURA S., MIYAZAKI F. *Bettering operations of robots by learning* // Journal of Robotic Systems. – 1984. – Vol. 1, №2. – P. 123–140.
6. ARIMOTO S., KAWAMURA S., MIYAZAKI F., TAMAKIE S. *Learning control theory for dynamic systems* // Proceedings of the 24th IEEE Conference on Decision and Control, Ft. Lauderdale, Florida, USA, 1985. – Vol. 24 – P. 1375–1380.
7. BIEN Z., XU J.X. *Iterative Learning Control- Analysis, Design, Integration and Applications*. – Kluwer Academic Publishers. – 1998. – 372 p.
8. BOYD S., GHAOUL L.E., FERON E., BALAKRISHNAN V. *Linear Matrix Inequalities in System and Control Theory*. – SIAM, 1994. – 203 p.
9. BRISTOW D.A., THARAYIL M., ALLEYNE A.G. *A survey of iterative learning control* // IEEE Control Systems Magazine. – 2006. – Vol. 26, №3. – P. 96–114.
10. CHEN Y.Q., WEN C. *Learning Control: Convergence, Robustness and Applications* // Lecture Notes series on Control and Information Science. V. LNCIS-248., London: Springer Verlag, 1999. – 248 p.
11. CRUSIUS C.A.R., TROFINO A. *Sufficient LMI Conditions for Output Feedback Control Problems* // IEEE Transactions On Automatic Control. – 1999. – Vol. 44, №5. – P. 1053–1057.

12. DORF R.C., BISHOP R.H. *Modern Control Systems*. – 10th edition. – Prentice Hall, 2004. – 1018 p.
13. HLADOWSKI L., GALKOWSKI K., CAI Z., ROGERS E., FREEMAN C.T., LEWIN P.L. *Using 2D systems theory to design output signal based iterative learning control laws with experimental verification* // Proceedings of the 47th IEEE Conference on Decision and Control, December 2008. – P. 3026–3031.
14. KUREK J.E., ZAREMBA M.B. *Iterative learning control synthesis based on 2-D system theory* // IEEE Transactions on Automatic Control. – 1993. – №38(1). – P. 121–125.
15. LONGMAN R.W. *Iterative learning control and repetitive control for engineering practice* // International Journal of Control. – 2000. – Vol. 73, №10. – P. 930–954.
16. MADADY ALI. *PID Type Iterative Learning Control with Optimal Gains* // International Journal of Control, Automation, and Systems. – 2008. – Vol. 6, №2. – P. 194–203.
17. MOORE K.L. *Iterative Learning Control for Deterministic Systems* // Advances in Industrial Control – London: Springer Verlag, 1993. – 152 p.
18. MOORE K.L. *Iterative learning control – an expository overview* // Applied and Computational Controls, Signal Processing and Circuits. – 1999. – № – P. 151–241.
19. OWENS D.H., AMANN N., ROGERS E. *Iterative learning control: an overview of recent algorithms* // Applied Mathematics and Computer Science. – 1995. – №5. – P. 425–438.
20. PASZKE W., LAM J., GALKOWSKI K., XU S., LIN Z. *Robust stability and stabilization of 2D discrete state-delayed systems* // System Control. – 2004. – №51. – P. 277–291.
21. ROESSER R. *A discrete state-space model for linear image processing* // IEEE Transactions on Automatic Control. – 1975. – №AC 20 (1). – P. 1–10.
22. ROGERS E., GALKOWSKI K., OWENS D.H. *Control Systems Theory and Applications for Linear Repetitive Processes* // Lecture Notes in Control and Information Sciences. Springer Verlag, 2007. – Vol. 349. – 466 p.

23. WANG FEI-YUE, LIU D. *Networked Control Systems Theory and Applications*. – London: Springer-Verlag, 2008. – 362 p.
24. WU J., CHEN T. *Design of Networked Control Systems With Packet Dropouts* // IEEE Transactions On Automatic Control. – 2007. – Vol. 52, №7. – P. 1314–1319.
25. XIAO L., HASSIBI A., HOW J.P. *Control with random communication delays via a discrete-time jump system approach* // Proceedings of American Control Conference, Chicago, USA, 2000. – Vol. 3. – P. 2199–2204.
26. XU J.X., TAN Y. *Linear and Nonlinear Iterative Learning Control*. – Springer, 2003. – 196 p.
27. XU J., YU Li. *H_∞ Control of 2-D Discrete State Delay Systems* // International Journal of Control, Automation, and Systems. –2006. – Vol. 4, №4. – P. 516–523.
28. YU M., WANG L., CHU W., HAO F. *Stabilization of Networked Control Systems with Data Packet Dropout and Transmission Delays: Continuous-Time Case* // European Journal of Control. – 2005. – №11. – P. 40–49.
29. YU M., WANG L., CHU W. *Stability analysis of networked systems with packet dropout and transmission delays: discrete-time case* // Asian Journal of Control. – 2005. – Vol. 7, №4. – P. 433–439.
30. ZAMPIERI S. *Trends in Networked Control Systems* // 17th IFAC World Congress, 2008. – P. 2886–2894.
31. ZHANG L., SHI Y., CHEN T., HUANG B. *A new method for stabilization of networked control systems with random delays* // IEEE Transaction On Automatic Control. – 2005. –Vol. 50, № 8. – P. 1177–1181.
32. ZHANG W., BRANICKY M., PHILIPS S. *Stability of Networked Control Systems* // IEEE Control Systems Magazine. – 2001. – №21(2). – P. 84–99.

STABILIZATION OF 2D BASED ITERATIVE LEARNING CONTROL SYSTEM OVER NETWORK WITH RANDOM COMMUNICATION DELAYS AND PACKET DROPOUTS

Sergey Valov, Arzamas Polytechnic Institute of R. E. Alekseev, Nizhny Novgorod State Technical University, Arzamas, post-graduate student (serval_nn@mail.ru)

Abstract: A two-dimensional discrete-time iterative learning control system where a sensor and a controller exchange information over network communication channel, is considered. The problem of system stabilization along the trial is being solved in the presence of possible time-varying delays and packet dropouts. A control law is designed for both system state and output vector. Time-varying delays and packet dropouts are modeled as independent Markovian chains. The resulting closed-loop system is transformed into a 2D discrete-time Roesser model with Markovian jumps characterization. Then, sufficient conditions for system stabilization of single-packet transmission are given in terms of linear matrix inequalities. A numerical example illustrates the effectiveness of the results.

Keywords: iterative learning control, network communication channel, delay, Markovian model, stochastic stability, linear matrix inequalities.

Статья представлена к публикации членом редакционной коллегии М. В. Губко