УДК 378.147+004.021

Д. Х. Имаев, Е. Е. Котова

Система управления процессом обучения с логическими алгоритмами принятия решений

Рассматриваются этапы разработки математических моделей систем управления учебным процессом, логические алгоритмы принятия решений о распределении дидактических ресурсов, компьютерные модели и пример имитации системы обучения.

Математические модели, процесс обучения, уровень знаний, логические алгоритмы управления, консервативная тактика обучения

Концептуальная модель системы управления процессом обучения с разделением дидактических ресурсов [1] включает модель обучаемого/студента как объекта управления, подсистемы оценки знаний, принятия решений о дополнительных ресурсах и их реализации. Ресурс обучения представляется двумя составляющими в соответствии со стратегией разделения дидактических ресурсов. Базовая часть является общей для всех студентов и подразумевает групповую форму организации занятий. Дополнительная часть ресурса подразумевает индивидуальную форму организации занятий и рассчитывается по фактическому уровню освоенных знаний. По терминологии теории управления базовая часть курса образует разомкнутую систему управления. Индивидуальные дополнительные занятия, рассчитанные на «отстающих» студентов, организованы по принципу обратной связи.

Разработки математических моделей систем управления учебным процессом сводятся к поэтапному уточнению принятых допущений с целью «приближения» моделей к реальному процессу, что приводит к различным типам и классам моделей (табл. 1).

На первом этапе [1] принимаются допущения об идеализации подсистем оценки результатов и средств реализации дидактических ресурсов.

Таблица 1

Этап	Тип, класс моделей	Принятые допущения (условия применимости)	Формы представления моделей
1	Непрерывная, стационарная линейная	Допущения об «идеальности» оценки уровня знаний и реализации дидактических ресурсов. Оценка уровня знаний и коррекция дидактических усилий по непрерывной шкале	Математические модели подсистем в форме дифференциальных уравнений и передаточных функций
2	Непрерывная, нелинейная	Допущения об «идеальности» оценки уровня знаний и времени реализации дидактических ресурсов. Учет мотивации обучаемого в виде нелинейной обратной связи	Математические модели подсистем в форме нелинейных дифференциальных уравнений
3	Непрерывно-дискретная (гибридная)	Объект непрерывный. Оценка знаний и реализация дидактических ресурсов производится периодически во времени	Математические модели в форме дифференциально-разностных уравнений
4	Гибридная модель с непрерывным временем и квантованием уровней переменных	Вводятся конечные множества оценок уровня знаний и дидактических усилий с четкими границами	Математическая модель подсистемы «студент» в виде дифференциального уравнения, подсистемы принятия решений – в виде асинхронного конечного автомата
5	Нелинейная непрерывная модель	Оценки уровня знаний с помощью функций принадлежности к нечетким множествам	Нелинейные дифференциальные уравнения. Аппроксимация нелинейной зависимости в форме механизма вывода нечеткой логики
6	Гибридная модель с дискретизацией времени и квантованием уровней переменных	Учитываются периодичность контроля успеваемости и конечность множества оценок уровня знаний и дидактических усилий	Математическая модель подсистемы «студент» в виде разностного уравнения. Модель логического управляющего устройства в виде синхронного конечного автомата

Предполагается, что уровень знаний оценивается в любое время с любой точностью, а информация от преподавателя обучаемому передается с любой степенью «дозировки».

Оценка успеваемости студента по четкой балльно-рейтинговой системе, а также конечное число уровней дидактических усилий приводит к принятию логических алгоритмов принятия решений об интенсивности дидактических усилий. На рис. 1 изображена принципиальная схема обратной связи формирования интенсивности дополнительных занятий с отстающими студентами. Переменная $y^*(t)$ задает график индивидуального обучения, рассчитанного на «среднего» студента, переменная $y(t)$ – уровень знаний отстающего студента, $\delta y = y^* - y$ – текущее отклонение уровня знаний от графика.

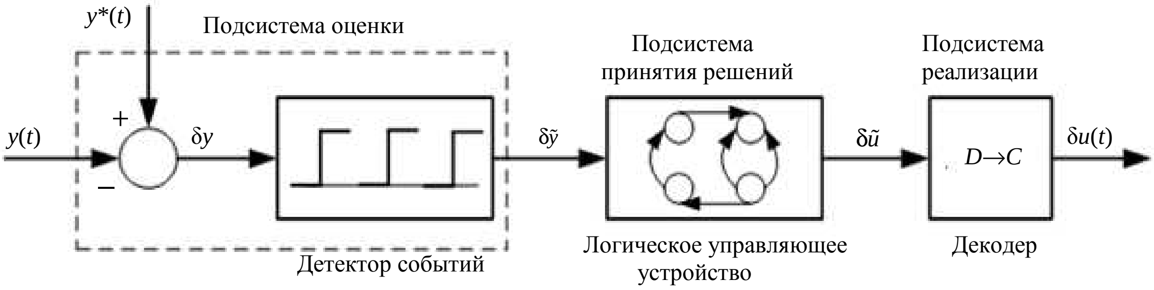

Рис. 1

Подсистема оценки знаний сопоставляет непрерывные сигналы в виде фактического и желаемого уровней знаний $y(t)$ и $y^*(t)$, дает результат в виде множества символов $\delta\tilde{y}$, соответствующих одной из ситуаций. Отображение непрерывных величин на множество значений лингвистической переменной реализуется на пороговых элементах, разбивающих множества действительных значений отклонения уровня знаний от графика на конечное число подмножеств. Пороги четкие: каждое значение выхода принадлежит строго одному подмножеству.

Подсистема принятия решений представляет логическое устройство с памятью, выход которого $\delta\tilde{u}$ является лингвистической переменной, задающей интенсивность занятий. Значения $\delta\tilde{u}$ принадлежат конечному множеству – числу градаций уровней дидактических усилий.

Подсистема реализации дидактических усилий – преобразователь $D \to C$ – сопоставляет с выходными символами логического устройства $\delta\tilde{u}$ действительные значения $\delta u(t)$. Сигнал выхода $\delta u(t)$ представляет собой кусочно-постоянную функцию времени, уровни которой принадлежат конечному множеству значений дидактических усилий. Уровни изменяются с появлением нового символа на входе логического устройства в зависимости от состояния логического устройства, которое может помнить историю.

Логические алгоритмы принятия решений. Моделью логических устройств, входы и выходы которых принимают значения на конечных множествах, является автомат [2]. Конечный автомат задается пятеркой $< S,\ Y,\ U,\ \sigma,\ \lambda >$, где S, Y, U – множества состояний, входов и выходов автомата; σ – функция переходов; λ – функция выходов. Функционирование автомата можно описать так:

$$\tilde{s}' = \sigma(\tilde{s}, \delta\tilde{y}); \quad \tilde{s}_0,$$
$$\delta\tilde{u} = \lambda(\tilde{s}, \delta\tilde{y}).$$

Здесь $\tilde{s}'$ – последующее состояние, зависящее от предыдущего состояния $\tilde{s}$ и от входа $\delta\tilde{y}$; $\tilde{s}_0$ – начальное состояние автомата.

Механизм вывода (рис. 1) является моделью в терминах «вход-состояние-выход». Исключение внутренних переменных системы вывода дает описание в терминах «вход-выход». Такая операция имеет смысл, так как статическая характеристика (СХ) преобразования является привычной для инженеров графической формой описания нелинейностей. Статические характеристики являются моделями в терминах «вход-выход», т. е. непосредственно связывают вход δy и выход δu преобразования. При переходе от механизма четкого вывода к СХ исключаются внутренние переменные $\delta\tilde{y}, \delta\tilde{u}, \tilde{s}, \tilde{s}'$. Функциональная зависимость $\delta u = F(\delta y)$ реализуется последовательным соединением аналого-символьного преобразователя (детектора событий), логического устройства и символьно-аналогового преобразователя (декодера $D \rightarrow C$), выполняющих информационную, алгоритмическую и исполнительную функции. Таким образом, подсистемы оценки, принятия решений и реализации дидактических усилий представляют механизм вывода в четкой (классической, булевской) логике (рис. 1) [3], [4].

Формирование тактики принятия решений сводится к заданию детектора событий, автомата и декодера (см. рис.1). Детектор событий и декодер определяют входной и выходной алфавиты автомата, т. е. множества *Y* и *U*. Для описания автомата необходимо выбрать множество его состояний *S*, а также задать функции перехода σ и выхода λ.

Прежде всего следует выбрать число ситуаций/событий, характеризующих успеваемость студента, т. е. перечислить элементы множества значений лингвистической переменной $\delta\tilde{y}$, например, Y = { a = «опережает», $b =$«успевает», c = «отстает», $d =$«сильно отстает»}. Далее назначаются уровни дидактических воздействий $u(t)$ множеством значений лингвистической переменной $\delta\tilde{u}$, например: U = {0 = «не требуется дополнительных занятий», 1 = «небольшая интенсивность занятий», 2 = «средняя интенсивность», 3 = «усиленные занятия»}. После этого устанавливаются пороги детектора событий, например, в процентах отклонения от графика, и реальные уровни занятий.

Тактика выбора интенсивности дополнительных занятий завершается формированием логики принятия решений. Можно предложить два подхода к формированию логики.

При первом подходе задача сводится к заполнению базы знаний в виде продукционных правил.

1. Если $\delta\tilde{y}=a$, то $\delta u=0$

(если студент опережает график, то не требуется дополнительных занятий).

2. Если $\delta\tilde{y}=b$, то $\delta u=1$

(если студент успевает, то интенсивность занятий небольшая).

3. Если $\delta\tilde{y}=c$, то $\delta u=2$

(если студент отстает, то интенсивность занятий средняя)

4. Если $\delta\tilde{y}=d$, то $\delta u=3$

(если студент сильно отстает, то занятия усиленные).

Приведенный пример простейшей логики определяет уровень интенсивности занятий в зависимости только от текущей успеваемости. При необходимости учета состояния студента и предыстории процесса обучения число правил увеличивается, а их структура усложняется. Например, одно из правил может иметь вид:

если $\delta\tilde{y}=b$ и $\tilde{s}=c$, то $\delta\tilde{u}=0$ и $\tilde{s}'=\tilde{s}=b$, т. е. если студент успевает, а перед этим отставал от графика, то не требуется дополнительных занятий и следует запомнить новое состояние. Здесь условие $\tilde{s}=c$ обозначает предыдущее состояние обучаемого, а решение $\delta\tilde{u}=0$ зависит как от текущего состояния $\delta\tilde{y}=b$, так и от предыдущего состояния $\tilde{s}=\delta\tilde{y}$. Кроме того, устройство переходит в новое состояние $\tilde{s}'=\tilde{s}=b$. Такая логика предусматривает наличие памяти в подсистеме принятия решений.

По базе правил находится необходимое число состояний автомата, после чего формируются функции перехода σ и выхода λ автомата.

При втором подходе для придания наглядности алгоритмам принятия решений предлагается использование СХ – графиков функции $\delta u = = F(\delta y)$. Построение СХ следует начинать с построения «сетки» в системе координат $(\delta y, \delta u)$, узлам которой отвечают значения порогов детектора событий/ситуаций и уровни интенсивности дидактических усилий.

Очевидно, для однозначных СХ соответствующий тривиальный автомат имеет единственное состояние, так как логическое устройство не имеет памяти, а простейшая логика принятия решений определяет уровень интенсивности занятий в зависимости только от текущей успеваемости. Например, для приведенной системы из четырех правил СХ представляет собой однозначную кусочно-постоянную (релейную) характеристику (рис. 2).

В том случае, когда логическое устройство принятия решений имеет память, кусочно-постоянные СХ становятся многозначными и изображают релейные характеристики с точностью до логики переходов между ветвями. На рис. 3 приведен пример многозначной СХ гистерезисного типа, отражающей логику «консервативной» тактики обучения. Как видно из СХ, переходы на последующие уровни интенсивности занятий происходят с «пространственным» запаздыванием, т. е. назначение или отмена дополнительных ресурсов происходит не сразу после выявления текущей ситуации, как это имеет место в случае простейшей тактики, а с учетом последующей ситуации.

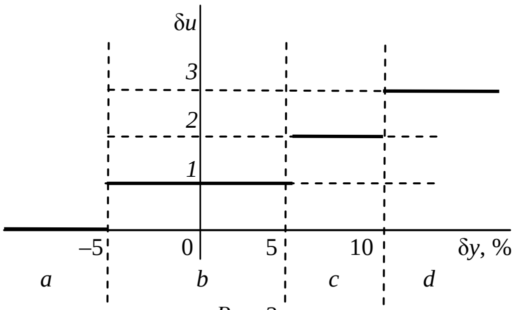

Рис. 2

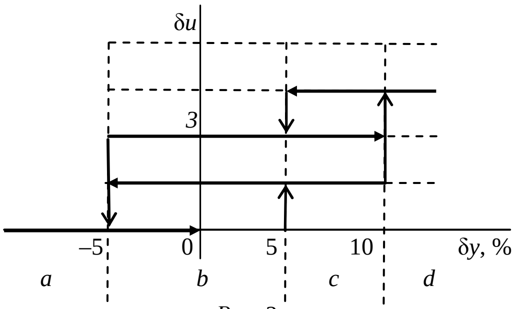

Рис. 3

Могут быть предложены примеры СХ, описывающие «прогрессивную» тактику, когда переходы на последующие уровни происходят с опережением. Очевидно, что различных тактик «консервативного» и «прогрессивного» типа может быть несколько. Выбор из них относится к задаче формирования адекватной тактики индивидуальных занятий и осуществляется путем многократных компьютерных имитационных экспериментов.

Реализация тактики обучения. Если исходная информация о тактике обучения представлена в форме СХ, то возникает задача реализации – построение механизма четкого вывода. Необходимость представления релейных СХ на языке механизма четкого вывода объясняется тем, что это упрощает реализацию тактики обучения в виде программ, а также компьютерную имитацию процесса обучения.

Переход от СХ – описания «вход-выход» – к форме «вход-состояние-выход» неоднозначен и зависит от выбора переменных состояния. Естественно распорядиться выбором так, чтобы минимизировать число состояний автомата или/и упростить характеристические функции σ и λ [2]. Реализация сводится к описанию детектора событий, автомата и декодера (см. рис.1). Детектор событий и декодер определяют входной и выходной алфавиты автомата, т. е. множества X и Y.

Для описания автомата необходимо выбрать множество его состояний S, а также задать функции перехода σ и выхода λ. Очевидно, для однозначных СХ задача тривиальна, так как автомат имеет единственное состояние.

Рассмотрим методику построения механизма четкого вывода по СХ на примере многозначной СХ, отвечающей «консервативной» тактике обучения (рис. 3). Удобной формой представления информации о логике переключений между ветвями многозначной СХ является граф переходов [3]. Графы переходов становятся графами автомата Мура, если в вершинах графа указать символы состояния/выхода, а на дугах проставить входные символы, приводящие к смене состояний. Полученный таким путем автомат, как правило, содержит избыточные состояния, что ставит задачу минимальной реализации.

Следующим действием является минимизация числа состояний автомата. В результате появляется автомат Мили, граф которого изображен на рис. 4. Вершины графа отвечают состояниям автомата, а на дугах указываются входные символы и значения выхода.

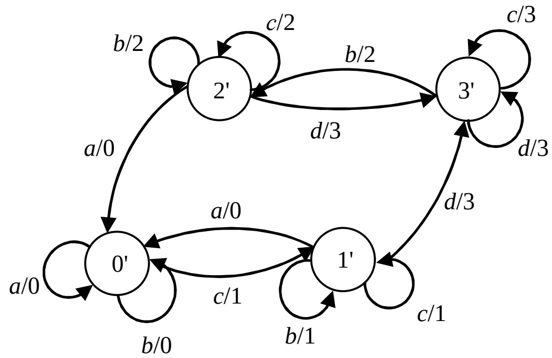

Рис. 4

Компьютерные модели системы управления процессом обучения с алгоритмом принятия решений в четкой логике. Выберем язык графического редактора Simulink среды научно-технических расчетов MATLAB фирмы "The MathWorks, Inc." Компьютерная модель системы управления учебным процессом с разделением ресурсов приведена на рис. 5.

Модели студентов описаны передаточной функцией вида $\kappa/[s(\tau s+1)]$, учитывающей два фактора: κ – темп восприятия информации и усвоения знаний; τ — вовлеченность в учебный процесс (инерция восприятия знаний) [1]. Для «среднего» судента κ/τ = 3/3, а «слабого» – 1/5, В компьютерной модели эти студенты представлены в форме подсистем Average student и Challenged student.

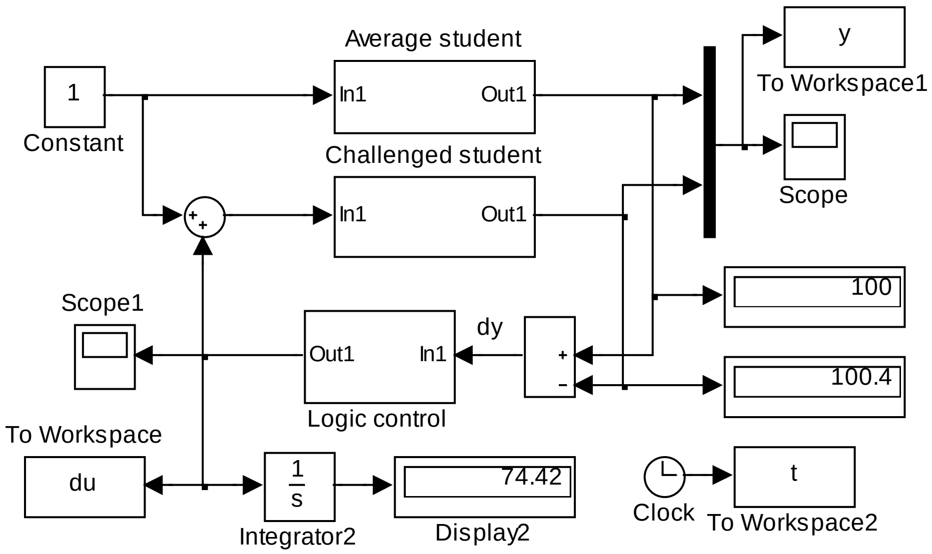

Рис. 5

Компьютерная модель подсистемы четкого вывода Logic control реализуется как последовательное соединение аналого-символьного преобразователя (детектора событий), автомата и декодера (рис. 6) [3]. Детектор событий представляет собой совокупность пороговых элементов, реализованных на блоках Compare To Constant. Булевы переменные на их выходах вместе с состоянием автомата на выходе блока Memory объединены в вектор с помощью мультиплексора Mux. Блок Combinatorial Logic реализует функцию переходов-выходов автомата и выдает вектор, объединяющий новое состояние и код символа выхода. Блок Demux разделяет составляющие вектора выхода блока Combinatorial Logic. Декодер реализован с помощью блока Combinatorial Logic1, который ставит в соответствие булевой переменной действительные значения.

Программная реализация «консервативной» тактики начинается с кодировки символов в виде бинарных (булевых) векторов:

– входного алфавита $Y = \{a, b, c, d\}$

$$a \Rightarrow \begin{bmatrix}0\\0\\0\end{bmatrix}; b \Rightarrow \begin{bmatrix}1\\0\\0\end{bmatrix}; c \Rightarrow \begin{bmatrix}1\\1\\0\end{bmatrix}; d \Rightarrow \begin{bmatrix}1\\1\\1\end{bmatrix};$$

– выходного алфавита $U = \{0, 1, 2, 3\}$

$$0 \Rightarrow \begin{bmatrix}0\\0\end{bmatrix}; 1 \Rightarrow \begin{bmatrix}0\\1\end{bmatrix}; 2 \Rightarrow \begin{bmatrix}1\\0\end{bmatrix}; 3 \Rightarrow \begin{bmatrix}1\\1\end{bmatrix};$$

– состояния автомата $S = \{0', 1', 2', 3'\}$

$$0' \Rightarrow \begin{bmatrix}0\\0\end{bmatrix}; 1' \Rightarrow \begin{bmatrix}0\\1\end{bmatrix}; 2' \Rightarrow \begin{bmatrix}1\\0\end{bmatrix}; 3' \Rightarrow \begin{bmatrix}1\\1\end{bmatrix}.$$

Функция переходов-выходов автомата Мили в форме таблицы приведена в табл. 2.

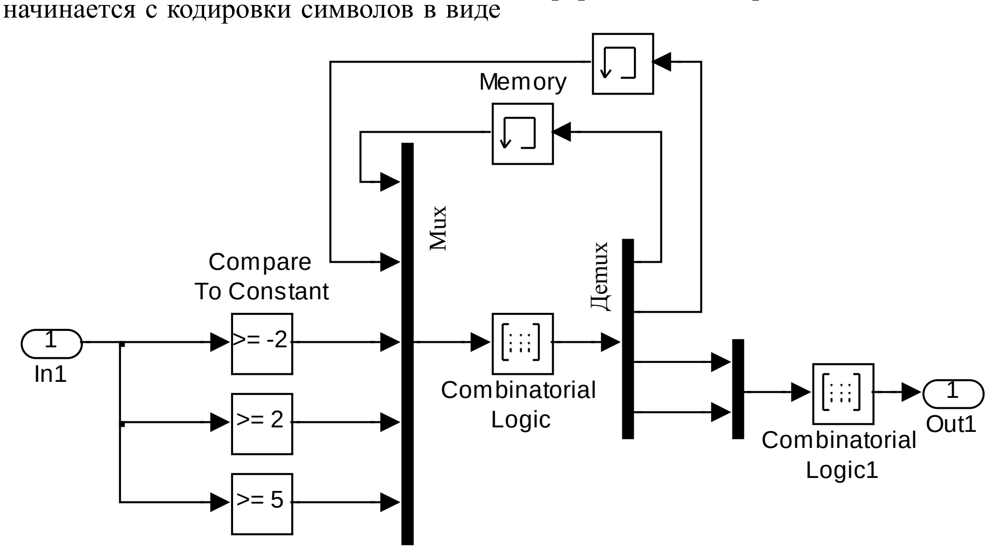

Рис. 6

Таблица 2

Row Index	Состояние старое		Вход			Состояние новое		Выход	
	$\tilde{s}_1$	$\tilde{s}_2$	δy_1	δy_2	δy_3	$\tilde{s}_1^{new}$	$\tilde{s}_2^{new}$	δu_1	δu_2
1	0	0	0	0	0	0	0	0	0
5	0	0	1	0	0	0	0	0	0
7	0	0	1	1	0	0	1	0	1
9	0	1	0	0	0	0	0	0	0
13	0	1	1	0	0	0	1	0	1
15	0	1	1	1	0	0	1	0	1
16	0	1	1	1	1	1	1	1	1
17	1	0	0	0	0	0	0	0	0
21	1	0	1	0	0	1	0	1	0
23	1	0	1	1	0	1	0	1	0
24	1	0	1	1	1	1	1	1	1
29	1	1	1	0	0	1	0	1	0
31	1	1	1	1	1	1	1	1	1
32	1	1	1	1	1	1	1	1	1

Индекс строк Row Index кодирует строки первых 5 столбцов табл. 2, рассматриваемых как двоичное представление десятичных чисел плюс единица. Поле редактирования блока Combinatorial Logic заполняется матрицей с 32 строками и 4 столбцами, соответствующими последним четырем столбцам табл.2: [0000; zeros(5, 4); 0101; zeros(5, 4); 0101; 0000; 0101; 1111; zeros(4, 4); 1010; 0000; 1010; 1111; zeros(4, 4); 1010; 0000; 1111; 1111]. Отсутствующие в табл. 2 строки, отвечающие невозможным ситуациям, заполнены нулями.

Пример имитации системы с консервативной тактикой. Рассмотрим зависимость процессов от пороговых значений δy – отклонения успеваемости студента от графика (см. рис. 3). Примем значения уровней дидактических усилий δu = = 0, 1, 2, 3.

На рис. 7 изображены траектории процессов накопления знаний: 1 – среднего студента и базового ресурса; 2 – слабого студента с пороговыми значениями (–2, 2, 5); 3 – (–5, 5, 10); 4 – (–10, 10, 20)

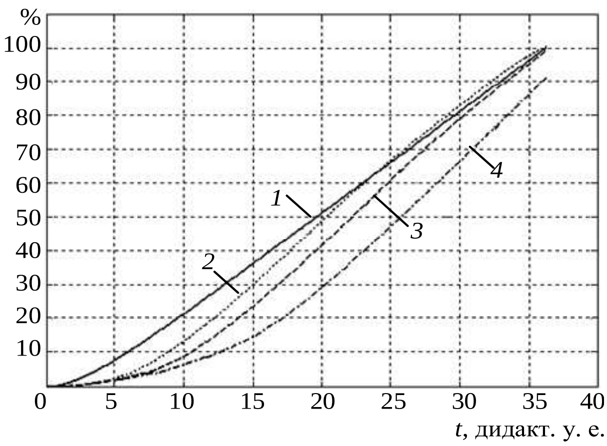

Рис. 7

для суммарного ресурса (базового + дополнительного). Кривые *2* и *3* удовлетворяют условию достижения 100 % , кривая 4 – 91,2 %.

На рис. 8 приведены графики изменений дополнительных ресурсов в зависимости от времени для пороговых значений степени отставания (–2, 2, 5) и (–5, 5, 10).

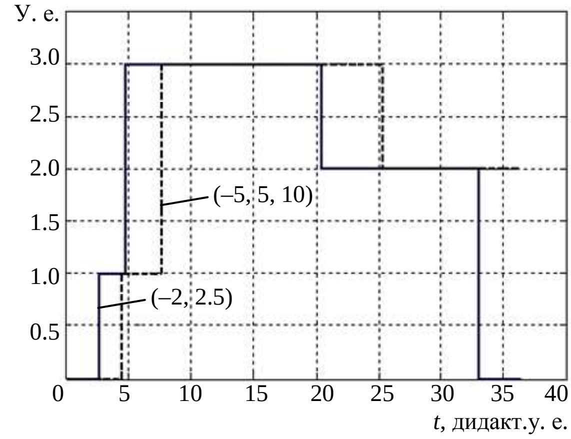

Рис. 8

Формы кривых демонстрируют различия в распределении применения дополнительных ресурсов по времени и количеству условных дидактических единиц, но процессы являются удовлетворительными: благодаря дополнительным занятиям слабые студенты достигают 100 % усвоения знаний.

Таким образом, балльная система оценок успеваемости, квантование объема дидактических ресурсов и логические алгоритмы принятия решений об интенсивности дополнительных занятий отражают реальные условия организации и проведения занятий. Дальнейшее повышение адекватности моделей может быть достигнуто при учете периодичности контроля уровня знаний и коррекции дидактических усилий.

СПИСОК ЛИТЕРАТУРЫ

1. Имаев Д. Х., Котова Е. Е. Модели и алгоритмы принятия решений о распределении дидактических ресурсов в среде обучения // Изв. СПбГЭТУ «ЛЭТИ». 2013. Вып. 8. С. 79–85.

2. Карпов Ю. Г. Теория автоматов. СПб.: Питер, 2003.

3. Имаев Д. Х. Дискретные системы управления: учеб. пособие. СПб: Изд-во СПбГЭТУ «ЛЭТИ», 2005.

4. Имаев Д. Х. Механизмы вывода в четкой логике и задачи теории управления // Управление и информационные технологи / Н. Н. Кузьмин, А. Ю. Дорогов, С. Е. Душин и др. СПб.: Технолит, 2011.

D. H. Imaev, E. E. Kotova

LEARNING MANAGEMENT SYSTEM WITH A LOGICAL DECISION-MAKING ALGORITHMS

The article considers stages of design of mathematical models for control systems of a learning process, as well as logical algorithms (those expressed as logical inference rules) of didactic resources management and distribution. We also discuss computer models of a learning system and an example of its simulation.

Mathematical models, learning process, level of knowledge, logical algorithms, conservative learning tactics