Автоматика и телемеханика, № 1, 2024

Стохастические системы

© 2024 г. **А.В. НАУМОВ, д-р физ.-мат. наук (naumovav@mail.ru), А.Е. СТЕПАНОВ (Rus.fta@yandex.ru), А.Э. УСТИНОВ (entro1122@gmail.com) (Московский авиационный институт (национальный исследовательский университет))**

О ЗАДАЧЕ МАКСИМИЗАЦИИ ВЕРОЯТНОСТИ УСПЕШНОГО ПРОХОЖДЕНИЯ ОГРАНИЧЕННОГО ПО ВРЕМЕНИ ТЕСТА[1]

Рассмотрена задача поиска оптимальной последовательности выполнения набора заданий в ограниченном по времени тесте, т.е. определяется группа заданий для обязательного первоначального выполнения в тесте, задания в другой группе выполняются в оставшееся время до окончания теста. За каждое правильно выполненное задание теста тестируемый получает определенное количество баллов. В качестве критерия предлагается вероятность превышения суммарным числом набранных за тест баллов определенного уровня, являющегося фиксированным параметром задачи, с одновременным выполнением ограничения на время проведения теста. Случайными параметрами задачи являются время ответа пользователя на каждое задание теста. Правильность ответа пользователя на задание моделируется случайной величиной с распределением Бернулли. Полученная задача стохастического билинейного программирования сводится к детерминированной целочисленной задаче математического программирования.

Ключевые слова: ограниченный по времени тест, задача максимизации вероятности, целочисленное математическое программирование.

DOI: 10.31857/S0005231024010056

1. Введение

В парадигме адаптивного компьютерного тестирования [1–6] задача построения оптимальной стратегии прохождения теста связана с формированием индивидуальной траектории обучения. Данная задача представляется актуальной в различных областях образовательного процесса: подготовка к сдаче абитуриентами единого государственного экзамена (ЕГЭ), прохождение регулярных тестирований в системе дистанционного обучения (СДО), проверка остаточных знаний обучающихся и т.д. Как правило, подобные формы тестирования ограниченны по времени, а структура теста заранее известна с точностью до типов заданий, или разделов курса, на проверку знания которых направлены эти задания. При этом имеется значительная статистика

[1] Работа выполнена при поддержке Российского научного фонда (проект № 23-21-00293).

выполнения тестируемыми подобных заданий в процессе обучения как по типу заданий, так и по индивидуальному пользователю. Часто, например в условиях использования в образовательном процессе СДО, сбор и хранение этой информации автоматизированы, как это сделано в СДО CLASS.NET Московского авиационного института [7, 8]. Это позволяет обоснованно использовать в рассматриваемой задаче математические модели учитываемых случайных параметров, например времени, затрачиваемого тестируемым на решение задания определенного типа. Модели времени ответа пользователя на задание теста широко представлены в литературе. Ван дер Линденом была предложена логнормальная модель времени [1], а в [9, 10] в качестве моделей использовались гамма-распределение и дискретное распределение. Частота правильного решения тестируемым задач определенного типа (определенной тематики, соответствующей номеру задания в тесте), полученная на основе данных о работе тестируемого в процессе обучения, может служить хорошей оценкой параметра распределения Бернулли, моделирующего верное решение тестируемым соответствующего задания в тесте. Все задания теста, как правило, характеризуются определенным количеством баллов, которые набирает тестируемый, верно решивший их. Суммарное количество набранных в ходе тестирования баллов характеризует качество подготовки тестируемого и является основой для его оценивания. Достижение тестируемым определенного суммарного балла за тест может служить для него определенным целевым показателем. В качестве стратегии тестируемого в условиях наличия в задаче упомянутых выше случайных параметров может служить набор заданий теста, которые следует выполнять в первую очередь.

В литературе достаточно широко представлены задачи построения различных тестов с целью проверки уровня знаний тестируемых, в том числе в вероятностной или квантильной постановке [3, 4, 6, 9, 10]. Однако публикации, в которых рассматривались бы задачи построения оптимальной для тестируемого стратегии прохождения теста, авторам не известны.

В работе формулируется задача поиска оптимальной стратегии тестируемого (в упомянутом выше смысле) по критерию максимизации вероятности набора суммарного балла за тест выше определенного уровня, выбираемого тестируемым. При этом учитывается вероятностное ограничение, связанное с тем, что время, затраченное тестируемым на выполнение теста, не должно превышать общее фиксированное время тестирования.

Данная задача стохастического билинейного программирования на основе доверительного метода [11] согласно методике, предложенной в [12], сводится к целочисленной задаче математического программирования. Исходные данные для оценки параметров используемых в задаче вероятностных моделей взяты из статистики работы пользователей СДО МАИ CLASS.NET. В работе обсуждаются также результаты численного эксперимента и зависимость оптимального значения критерия от суммарного балла, превысить который стремится тестируемый.

2. Распределения используемых в работе случайных параметров

В рассматриваемой задаче используются два векторных случайных параметра. Одним из них является вектор $X = \mathrm{col}\,(X_1, \ldots, X_n)$, координаты которого моделируют правильность решения i-го задания теста, состоящего из n заданий. Предполагается, что X_i являются независимыми случайными величинами, имеющими распределение Бернулли, с параметрами p_i, $i = 1, \ldots, n$, оцениваемыми частотой правильных ответов тестируемого на аналогичные задания i-го типа в ходе подготовки к тестированию или в процессе обучения. Равенство единице случайной величины X_i моделирует правильность решения тестируемым i-го задания, а равенство нулю – противоположное событие. Другим случайным параметром является вектор $T = \mathrm{col}(T_1, \ldots, T_n)$, координаты которого характеризуют время, затрачиваемое тестируемым на решение задания i-го типа. Случайные величины T_i, $i = 1, \ldots, n$ также предполагаются независимыми. Однако предполагать независимость между величинами X и T было бы опрометчиво, поэтому для каждого значения X_i (0 или 1) оценивается свое распределение случайной величины T_i также на базе статистики решения тестируемым заданий аналогичного типа. Непрерывные распределения времени ответа пользователя на задание (Ван дер Линдена [1], Гамма-распределения [9]) не позволяют найти точное решение задачи в вероятностной постановке, поэтому в работе используется дискретизированная модель времени ответа с тремя значениями, моделирующими ситуации быстрого решения, стандартного решения и решения с затруднениями. Методика построения дискретного закона распределения времени ответа тестируемого на задания теста может быть различной: от дискретизации различными способами непрерывных моделей распределения (например, модели Ван дер Линдена [1]), параметры которых определяются на основе статистических данных о времени решения тестируемым задач соответствующего класса, до использования исходной гистограммы распределения, построенной по тем же статистическим данным. В этой работе для каждого задания теста по имеющимся статистическим данным, полученным из системы CLASS.NET [8], строится вариационный ряд времени ответа тестируемого на аналогичные задания, который разбивается на три равные части по расстоянию между максимальным и минимальным элементами. Для каждой части вычисляется выборочное среднее, которое используется в качестве соответствующего возможного значения интересующей нас случайной величины. Вероятности полученных трех возможных значений полагаются равными частотам попадания элементов используемой выборки в соответствующие выбранные диапазоны. Таким образом, общий вектор случайных величин имеет дискретное распределение с числом реализаций $D = 2^n 3^n$. Вероятности каждой реализации могут быть найдены с помощью формулы умножения вероятностей и использования условного распределение времени ответа тестируемого на задания теста при условиях правильного или неправильного его решения.

3. Постановка задачи и метод ее решения

Требуется определить стратегию тестируемого при выполнении им ограниченного по времени теста из n заданий. Стратегия определяется вектором булевых переменных $u \in \{0,1\}^n$, где

$$u_i \triangleq \begin{cases} 1, & \text{если тестируемый пытается решать } i\text{-е задание теста,} \\ 0 & \text{в противном случае,} \end{cases} \qquad i = \overline{1,n}.$$

За каждое i-е задание теста начисляется b_i баллов. Для успешного преодоления теста требуется набрать не менее φ баллов. Время выполнения теста ограничено величиной $\overline{T}$. В качестве критерия оптимизации выбрана вероятность успешного преодоления теста с одновременным выполнением ограничения по времени его прохождения.

Рассмотрим функцию вероятности

$$P_{\varphi,\overline{T}}(u) \triangleq P\left\{\sum_{i=1}^{n} u_i X_i b_i \geqslant \varphi,\ \sum_{i=1}^{n} u_i T_i \leqslant \overline{T}\ \right\}.$$

В ней величины φ и $\overline{T}$ играют роль параметров. Для обеспечения разумности постановки задачи наложим ограничения на указанные параметры: $0 < \varphi \leqslant \sum_{i=1}^{n} b_i$ и $\sum_{i=1}^{n} T_i^{\min} \leqslant \overline{T}$, где $T_i^{\min}$ – минимальное время решения тестируемым i-й задачи. Тогда задача поиска оптимальной стратегии тестируемым может быть сформулирована следующим образом:

$$P_{\varphi,\overline{T}}(u) \to \max_{u \in \{0,1\}^n}. \tag{1}$$

Данная задача является задачей стохастического программирования с булевыми переменными.

Как уже было сказано выше, число всех возможных реализаций вектора случайных параметров $\mathrm{col}(X^\top, T^\top)$ равно $D = 2^n 3^n$. Рассмотрим вектор $\delta \in \{0,1\}^D$, каждая ν-я координата которого соответствует одной из реализаций $\mathrm{col}((x^\nu)^\top, (t^\nu)^\top)$ вектора $(X^\top, T^\top)^\top$ и может принимать значения 0 или 1. Пусть $\Upsilon \triangleq e^T b$, где $e = \mathrm{col}(1,\dots,1) \in R^n$, т.е. $\Upsilon = \sum_{i=1}^{n} b_i$ – максимальное количество баллов, которое можно набрать за тест. Пусть $p_\nu = P(\mathrm{col}(X^\top, T^\top) = \mathrm{col}((x^\nu)^\top, (t^\nu)^\top))$, $\nu = \overline{1,D}$. Тогда аналогично методике, предложенной в [12] для решения задачи минимизации функции квантили на основании доверительного метода [11], задача стохастического программирования (1) может быть сведена к детерминированной задаче оптимизации с булевыми переменными:

$$\sum_{\nu=1}^{D} p_\nu \delta_\nu \to \max_{\substack{u \in \{0,1\}^n \\ \delta \in \{0,1\}^D}} \tag{2}$$

при ограничениях

$$\varphi - \Upsilon - \delta_\nu \left(\sum_{i=1}^{n} u_i x_i^\nu \; b_i - \Upsilon \right) \leqslant 0, \quad \nu = \overline{1, D}, \tag{3}$$

$$\delta_\nu u^T t^\nu - \overline{T} \leqslant 0, \quad \nu = \overline{1, D}. \tag{4}$$

В рассмотренной выше задаче оптимальное значение вектора δ определяет вид оптимального доверительного множества (в терминах доверительного метода [11]) в виде тех реализаций вектора случайных параметров, которым соответствуют единицы в оптимальном векторе δ. Суммарная вероятностная мера таких реализаций максимизируется в рассматриваемой задаче, а ограничения на время выполнения теста и набранного за тест количества баллов выполняются, в то время как для оставшихся реализаций, которым соответствуют нулевые значения координат оптимального вектора δ, эти ограничения в исходной задаче могут не выполняться, а ограничения (3) и (4) выполняются по построению.

Задача (2)–(4), построенная строго по методике [12], является задачей билинейного программирования (с билинейной системой ограничений), что в совокупности с булевостью переменных и большой размерностью делает ее трудно разрешимой. Однако структура рассматриваемой задачи позволяет переписать ее в виде задачи линейного программирования (ЗЛП), что возможно позволит использовать специальные методы решения ЗЛП с булевыми переменными, реализованные в современных пакетах прикладных оптимизационных программ. Вид этой ЗЛП следующий:

$$\sum_{\nu=1}^{D} p_\nu \delta_\nu \to \max_{\substack{u \in \{0,1\}^n \\ \delta \in \{0,1\}^D}} \tag{5}$$

при ограничениях

$$\delta_\nu \varphi \leqslant \sum_{i=1}^{n} u_i x_i^\nu \; b_i, \quad \nu = \overline{1, D}, \tag{6}$$

$$u^T t^\nu \leqslant \delta_\nu \overline{T} + (1 - \delta_\nu) T^{MAX}, \quad \nu = \overline{1, D}, \tag{7}$$

где

$$T^{MAX} = \sum_{i=1}^{n} T_i^{\max},$$

а $T_i^{\max}$ – максимальная из всех возможных реализаций случайной величины T_i, $i = \overline{1, n}$.

Если размерности задач (2)–(4), (5)–(7) допускают их решение стандартными процедурами из известных библиотек оптимизационных программ, то решение может быть найдено с их помощью. Однако эти задачи содержат дополнительный вектор переменных оптимизации $\delta \in \{0,1\}^D$ большой размерности, что с учетом большого числа ограничений делает их трудноразрешимыми полным перебором возможных значений булевых переменных оптимизации и требует разработки специальных методов решения, учитывающих структуру задачи. Далее рассмотрим алгоритм решения исходной задачи. Эффективность его применения по сравнению со стандартными библиотечными процедурами решения задач (2)–(4), (5)–(7) обсудим в разделе, касающимся результатов численного эксперимента.

4. Алгоритм решения исходной задачи

Шаг 0. Из всех 2^n стратегий $u \in \{0,1\}^n$ выбираем N, образующих множество $\underline{U}$, для элементов которого выполнены условия

$$\sum_{i=1}^{n} u_i b_i \geqslant \varphi, \quad \sum_{i=1}^{n} u_i T_i^{\min} \leqslant \overline{T}.$$

Смысл в том, что так отсеиваем стратегии, заведомо не подходящие по суммарному времени или количеству баллов даже в самом оптимистичном случае, когда все выбранные для решения задачи решены верно и за минимально возможное время.

Перенумеруем все элементы множества $\underline{U}$. Таким образом, число от 1 до N однозначно определяет элемент множества. Под u^m будем понимать m-й элемент множества $\underline{U}$. Положим $m := 1$, $P^* := 0$, а $u^* := (0,\ldots,0)^\top$.

На этом шаге инициируется внешний цикл перебора всех N выбранных стратегий оптимизации.

Шаг 1. Если $m > N$, то переходим к шагу 5. В противном случае $P_m := 0$.

Вспомогательный параметр P_m используется далее для расчета вероятности выполнения ограничений при $u = u^m$.

Шаг 2. Предположим, что вектор u^m содержит ровно K единиц. Предположим, что ненулевыми компонентами вектора u^m являются компоненты с номерами $i_1,\ldots,i_K$. Рассмотрим подвектор $\mathrm{col}(X_{i_1},\ldots,X_{i_K})$ случайного вектора X. Положим $J := 2^K$, а $j := 1$.

На этом шаге инициализируется цикл перебора всех возможных реализаций $\mathrm{col}(x_{i_1}^j,\ldots,x_{i_K}^j)$, $j = \overline{1,2^K}$.

Шаг 3. Если $j > J$ и $P_m > P^*$, то полагаем $P^* := P_m$, $u^* := u^m$, $m := m+1$ и переходим к шагу 1.

Если $j > J$ и $P_m \leqslant P^*$, то полагаем $m := m+1$ и переходим к шагу 1.

В противном случае, если для реализации $\mathrm{col}(x_{i_1}^j,\ldots,x_{i_K}^j)$ выполняется условие

$$\sum_{i \in \{i_1,i_2,\ldots,i_K\}} u_i^m b_i x_i^j \geqslant \varphi,$$

то полагаем $L := 3^K$, $l := 1$ и переходим к шагу 4. Если указанное условие не выполняется, то полагаем $j := j + 1$ и переходим к началу шага 3.

На этом шаге инициализируется цикл перебора всех возможных реализаций $\mathrm{col}(t^l_{i_1}, \ldots, t^l_{i_K})$, $l = \overline{1, L}$ подвектора $\mathrm{col}(T_{i_1}, \ldots, T_{i_K})$ случайного вектора T.

Шаг 4. Если $l > L$, то полагаем $j := j + 1$ и переходим к шагу 3. В противном случае, если для реализации $\mathrm{col}(t^l_{i_1}, \ldots, t^l_{i_K})$ выполняется условие

$$\sum_{i \in \{i_1, i_2, \ldots, i_K\}} u^m_i t^l_i \leqslant \overline{T},$$

то полагаем

$$P_m := P_m + \prod_{i \in \{i_1, i_2, \ldots, i_K\}} P(T_i = t^l_i | X_i = x^j_i) P(X_i = x^j_i).$$

Полагаем $l := l + 1$ и переходим к началу шага 4.

Шаг 5. Полагаем оптимальное значение критерия равным P^*, а оптимальное значение стратегии равным u^*.

Заметим, что во всех вложенных циклах, рассмотренных в алгоритме, происходит существенное сокращение объема требуемого перебора возможных значений переменных оптимизации. Объем полного перебора может сократиться на порядок в зависимости от выбранных значений параметров задачи φ и $\overline{T}$.

5. Результаты численного эксперимента

Исходные распределения для решения задачи получены на основе анализа функционирования системы дистанционного обучения МАИ CLASS.NET [8]. Будем предполагать число заданий в тесте $n = 10$. Оценки параметров исходных распределений заданы в табл. 1–3.

Рассмотрим величину $b^{\max} = \sum_{i=1}^n b_i$. В результате работы предложенного алгоритма были получены зависимости оптимальных решений от значений параметров задачи φ и $\overline{T}$, см. табл. 4, 5.

Расчеты проводились на компьютере ASUS X550LC (Intel Core i5 2.3 GHz, 8Gb RAM). Задача линейного программирования решалась пакетом IBM Cplex из библиотеки программ Python. Как видно, значения параметров задачи существенно влияют на скорость работы авторского алгоритма. Так, увеличение желаемого количества баллов за тест приводит к уменьшению вероятности достижения этого результата и уменьшению времени расчета с использованием предложенного алгоритма за счет снижения объема перебора допустимых стратегий оптимизации u. Сравнительный анализ времени работы алгоритмов показывает существенный его рост при увеличении числа задач в тесте, см. табл. 6. Все алгоритмы при одинаковых параметрах задачи

Таблица 1. Вероятность правильного решения заданий теста

Номер задачи в тесте	Вероятность правильного решения	Число баллов за задачу
1	0,90	1
2	0,91	1
3	0,95	1
4	0,97	1
5	0,90	1
6	0,80	2
7	0,65	3
8	0,75	2
9	0,50	4
10	0,55	3

Таблица 2. Условное распределение времени ответа тестируемого на задание теста в случае его неправильного решения

Номер задачи в тесте	Условное распределение времени ответа			
1	t_1^j	60	100	160
	$P(T_1 = t_1^j \mid X_1 = 0)$	0,3	0,55	0,15
2	t_2^j	70	130	250
	$P(T_2 = t_2^j \mid X_2 = 0)$	0,25	0,6	0,15
3	t_3^j	60	150	270
	$P(T_3 = t_3^j \mid X_3 = 0)$	0,2	0,45	0,35
4	t_4^j	100	200	350
	$P(T_4 = t_4^j \mid X_4 = 0)$	0,15	0,6	0,25
5	t_5^j	75	140	210
	$P(T_5 = t_5^j \mid X_5 = 0)$	0,2	0,45	0,35
6	t_6^j	190	290	400
	$P(T_6 = t_6^j \mid X_6 = 0)$	0,1	0,65	0,25
7	t_7^j	310	380	450
	$P(T_7 = t_7^j \mid X_7 = 0)$	0,2	0,4	0,4
8	t_8^j	180	250	320
	$P(T_8 = t_8^j \mid X_8 = 0)$	0,1	0,3	0,6
9	t_9^j	360	480	600
	$P(T_9 = t_9^j \mid X_9 = 0)$	0,1	0,3	0,6
10	t_{10}^j	320	400	470
	$P(T_{10} = t_{10}^j \mid X_{10} = 0)$	0,3	0,55	0,15

Таблица 3. Условное распределение времени ответа тестируемого на задание теста в случае его правильного решения

Номер задачи в тесте	Условное распределение времени ответа			
1	t_i^j	60	180	300
	$P(T_1 = t_1^j \mid X_1 = 1)$	0,3	0,5	0,2
2	t_i^j	75	190	330
	$P(T_2 = t_2^j \mid X_2 = 1)$	0,15	0,6	0,25
3	t_i^j	60	120	250
	$P(T_3 = t_3^j \mid X_3 = 1)$	0,15	0,35	0,5
4	t_i^j	130	200	350
	$P(T_4 = t_4^j \mid X_4 = 1)$	0,1	0,3	0,6
5	t_i^j	75	140	210
	$P(T_5 = t_5^j \mid X_5 = 1)$	0,2	0,45	0,35
6	t_i^j	200	275	380
	$P(T_6 = t_6^j \mid X_6 = 1)$	0,2	0,35	0,45
7	t_i^j	310	380	450
	$P(T_7 = t_7^j \mid X_7 = 1)$	0,2	0,4	0,4
8	t_i^j	200	290	370
	$P(T_8 = t_8^j \mid X_8 = 1)$	0,25	0,4	0,35
9	t_i^j	380	470	650
	$P(T_9 = t_9^j \mid X_9 = 1)$	0,1	0,25	0,65
10	t_i^j	150	275	500
	$P(T_{10} = t_{10}^j \mid X_{10} = 1)$	0,3	0,55	0,15

Таблица 4. Зависимость оптимального решения от параметра φ при $\overline{T} = 0{,}8\,T^{\max}$

φ	Оптимальная стратегия	Опт. значение критерия	Время расчета (сек)	Число исслед. стратегий
$0{,}4b^{\max}$	1. 1. 1. 1. 1. 1. 1. 1. 0. 1.	0,9241	29,9	727
$0{,}5b^{\max}$	1. 1. 1. 0. 1. 1. 1. 1. 1. 1.	0,7874	18,2	511
$0{,}6b^{\max}$	1. 1. 1. 0. 1. 1. 1. 1. 1. 1.	0,5777	8,9	295
$0{,}7b^{\max}$	1. 1. 1. 0. 1. 1. 1. 1. 1. 1.	0,3830	3,2	131
$0{,}8b^{\max}$	1. 1. 1. 1. 1. 1. 1. 1. 1. 1.	0,2028	0,9	39
$0{,}9b^{\max}$	1. 1. 1. 1. 1. 1. 1. 1. 1. 1.	0,0816	0,1	5

получили совпадающие решения для всех значений n. При $n \geqslant 7$ не удалось решить ЗЛП из-за проблем с недостатком памяти для хранения матрицы системы ограничений. Наибольшую эффективность показал алгоритм авторов, превосходя для больших n на порядок время работы других рассматриваемых алгоритмов.

Таблица 5. Зависимость оптимального решения от параметра $\overline{T}$ при $\varphi = 0{,}6\ b^{\max}$

$\overline{T}$	Оптимальная стратегия	Оптим. значение критерия	Время расчета (с)	Число исслед. стратегий
$0{,}4T^{\max}$	0. 0. 0. 0. 0. 1. 1. 0. 1. 1.	0,0633	1,8	273
$0{,}5T^{\max}$	0. 0. 1. 0. 1. 1. 1. 1. 0. 1.	0,1733	9,5	296
$0{,}6T^{\max}$	1. 1. 1. 0. 1. 1. 1. 1. 0. 1.	0,2786	8,7	296
$0{,}7T^{\max}$	1. 1. 1. 1. 1. 1. 1. 1. 0. 1.	0,5049	9,0	296
$0{,}8T^{\max}$	1. 1. 1. 0. 1. 1. 1. 1. 1. 1.	0,5777	8,9	296
$0{,}9T^{\max}$	1. 1. 1. 1. 1. 1. 1. 1. 1. 1.	0,7213	8,3	296

Таблица 6. Время работы алгоритмов (с) для разных значений n

n	ЗЛП	Полный перебор	Алгоритм авторов
1	0,0010	0,0340	0,0010
2	0,0010	0,0400	0,0016
3	0,0102	0,0580	0,0020
4	0,0202	0,0800	0,0029
5	0,0628	0,2000	0,0077
6	0,0991	0,6100	0,0117
7	–	2,1900	0,0636
8	–	9,3800	0,1600
9	–	48,1000	2,1500
10	–	3010,0000	9,1100

6. Заключение

В данной статье рассматривается задача стохастического программирования для поиска оптимальной стратегии прохождения ограниченного по времени теста по критерию максимума вероятности преодоления тестируемым определенного числа набранных за тест баллов с учетом ограничения на время выполнения теста. Для тестируемого считается известной вероятность правильного решения каждого задания теста. Случайным является также время, затрачиваемое тестируемым на решение каждого задания.

Рассмотренная задача стохастического программирования с вероятностным критерием качества сводится к детерминированной задаче большой размерности, для которой можно использовать стандартные оптимизационные процедуры из известных библиотек программ. Кроме этого, предлагается алгоритм направленного перебора возможных значений дискретной стратегии оптимизации, сокращающий временные затраты на решение задачи. Проведенный численный эксперимент подтвердил эффективность применения разработанного алгоритма решения исходной задачи по сравнению с применением стандартных оптимизационных процедур для ее детерминированных эквивалентов. Эта эффективность, определяемая разностью времени вычисления оптимальной стратегии различными способами, растет с увеличением числа заданий в тесте. Численный эксперимент показал также существенную зависимость оптимального решения и времени его вычисления от параметров задачи, что обосновывает актуальность дальнейшего совершенствования алгоритма ее решения.

Полученные в работе результаты могут быть распространены на квантильную постановку рассматриваемой задачи, когда тестируемый стремится максимизировать число набранных за тест баллов с сохранением выбранного уровня вероятности выполнения всех ограничений задачи. Это является отдельным исследованием, результаты которого планируется авторами к публикации.

СПИСОК ЛИТЕРАТУРЫ

1. *Van der Linden W.J., Scrams D.J., Schnipke D.L., et al.* Using Response-Time Constraints to Control for Differential Speededness in Computerized Adaptive Testing // Appl. Psychol. Measur. 1999. V. 23. No. 3. P. 195–210.
2. *Rasch G.* Probabilistic models for some intelligence and attainment tests. Chicago: The University of Chicago Press, 1980.
3. *Куравский Л.С., Мармалюк П.А., Алхимов В.И., Юрьев Г.А.* Новый подход к построению интеллектуальных и компетентностных тестов // Моделирование и анализ данных. 2013. № 1. С. 4–28.
4. *Kuravsky L.S., Margolis A.A., Marmalyuk P.A., Panfilova A.S., Yuryev G.A., Dumin P.N.* A Probabilistic Model of Adaptive Training // Appl. Math. Sci. 2016. V. 10. No. 48. P. 2369–2380.
5. *Наумов А.В., Мартюшова Я.Г.* Адаптация системы дистанционного обучения на основе статистической обработки результатов работы пользователей // Электр. журн. “Труды МАИ”. № 109. декабрь. 2019 г.
6. *Босов А.В., Мартюшова Я.Г., Наумов А.В., Сапунова А.П.* Байесовский подход к построению индивидуальной траектории пользователя в системе дистанционного обучения // Информатика и ее применения. 2020. Т. 14. № 3. С. 86–93.
7. *Наумов А.В., Джумурат А.С., Иноземцев А.О.* Система дистанционного обучения математическим дисциплинам CLASS.NET // Вест. компьют. и информ. технологий. 2014. № 10. С. 36–40.
8. СДО МАИ CLASS.NET [Электронный ресурс] // URL: `https://distance.kaf804.ru/` (дата обращения: 27.02.2023)
9. *Босов А.В., Мхитарян Г.А., Наумов А.В., Сапунова А.П.* Использование гамма-распределения в задаче формирования ограниченного по времени теста // Информатика и ее применения. 2019. Т. 13. № 4. С. 12–18.
10. *Наумов А.В., Мхитарян Г.А., Черыгова Е.Е.* Стохастическая постановка задачи формирования теста заданного уровня сложности с минимизацией квантили времени выполнения // Вест. компьют. и информ. технологий. 2019. № 2. С. 37–46.
11. *Кан Ю.С., Кибзун А.И.* Задачи стохастического программирования с вероятностными критериями. М.: Физматлит, 2009.

12. *Кибзун А.И., Наумов А.В., Норкин В.И.* Сведение задачи квантильной оптимизации с дискретным распределением к задаче смешанного целочисленного прораммирования // АиТ. 2013. № 6. С. 66–86.

Kibzun A.I., Naumov A.V., Norkin V.I. On reducing a quantile optimization problem with discrete distribution to a mixed integer programming problem // Autom. Remote Control. 2013. V. 74. No. 6. P. 951–967.

Статья представлена к публикации членом редколлегии А.И. Кибзуном.

Поступила в редакцию 18.10.2023

После доработки 30.11.2023

Принята к публикации 21.12.2023