

Рис.1 Фазовый портрет системы (11)–(13)

Литература

1. *L. de Cesare, A. di Liddo, S. Ragni* On the shape of optimal price and advertizing strategies arising in innovation diffusion // *Dynamic systems and applications*, 2004. Vol. 13, no. 1. P. 49–60.
2. *Габасова О.Р., Габасов Р., Дмитрук Н. М.* Синтез оптимальной политики для производственно-финансовой модели фирмы // *Автоматика и телемеханика*, 1998. № 9. С. 100–117.

О ПРОЕКТИРОВАНИИ И РЕАЛИЗАЦИИ ГЕНЕРАТОРА МНОГОВАРИАНТНЫХ ТЕСТОВЫХ ЗАДАНИЙ ДЛЯ СИСТЕМЫ КОМПЬЮТЕРНОГО ТЕСТИРОВАНИЯ

О. В. Горовцова

Многочисленные исследования по компьютерному тестированию специалистами из различных областей знаний обусловлены его практической значимостью в сфере образования

Актуальными являются вопросы использования систем компьютерного тестирования для контроля знаний. Один из аспектов, требующих детального рассмотрения, связан с процессом создания тестовых заданий, а в частности, с их генерацией.

Отметим, что не существует универсального генератора тестовых заданий. Это связано с отличиями и особенностями в формулировке вопросов и заданий для разных дисциплин и областей знаний. Поэтому нужно создавать отдельные алгоритмы генерации условий для каждого типа заданий. Задача генерации тестовых заданий очень трудоемкая и

требует наличия специалистов в предметной области, по которой проводится тестирование, в программировании, в методике преподавания.

Использование разработанных для отдельных видов заданий генераторов освобождает от рутинного процесса составления и формулирования каждого задания отдельно. И число сгенерированных тестовых заданий может быть очень велико. Это зависит от конкретного алгоритма генерации и области возможных значений параметров. Помимо этого, генерация тестовых заданий позволяет решить некоторые практические проблемы, возникающие в процессе проведения тестирования. Если существует база тестовых заданий, материалы из которой используются повторно (например, при проведении промежуточного и итогового контроля), то существует риск заготовки шпаргалок и механического запоминания ответов учащимися. При одновременной сдаче теста у нескольких испытуемых могут быть одинаковые задания, что также повышает вероятность списывания. А сгенерированные задания отличаются уникальностью.

Формально принцип генерации заданий может быть представлен в виде шаблона задания, представляющего собой заготовку текста, в котором могут меняться фрагменты. То есть вместо определенных частей текста в шаблоне используются параметры, значения которых генерируются алгоритмами.

Помимо формулировки тестового задания необходимо наличие правильного решения или ответа. Поэтому вместе с генерацией условия должен существовать алгоритм, решающий задачу и не допускающий формирование условий, не имеющих решения. Блок-схема такого алгоритма показана на рис. 1. В случае, если задача с какими-то сгенерированными параметрами не имеет корректного решения, процесс генерации задания возобновляется, то есть имеет циклический характер [1, с. 41].

Циклический характер имеет также сам алгоритм генерации заданий, в результате чего получается набор однотипных неповторяющихся заданий.

Рассмотрим частный случай генерации тестовых заданий, имеющих точный однозначный ответ, для дисциплин с математической нотацией. Средством генерации тестовых заданий выбрана система компьютерной алгебры Wolfram Mathematica. Функциональные возможности системы позволяют использовать ее для этих целей благодаря возможности аналитических преобразований и численных расчетов, работы со строками, наличию средств программирования. Язык программирования Wolfram поддерживает различные парадигмы программирования, поэтому есть

возможность программирования каждой задачи в естественной для нее парадигме [2, с. 6].

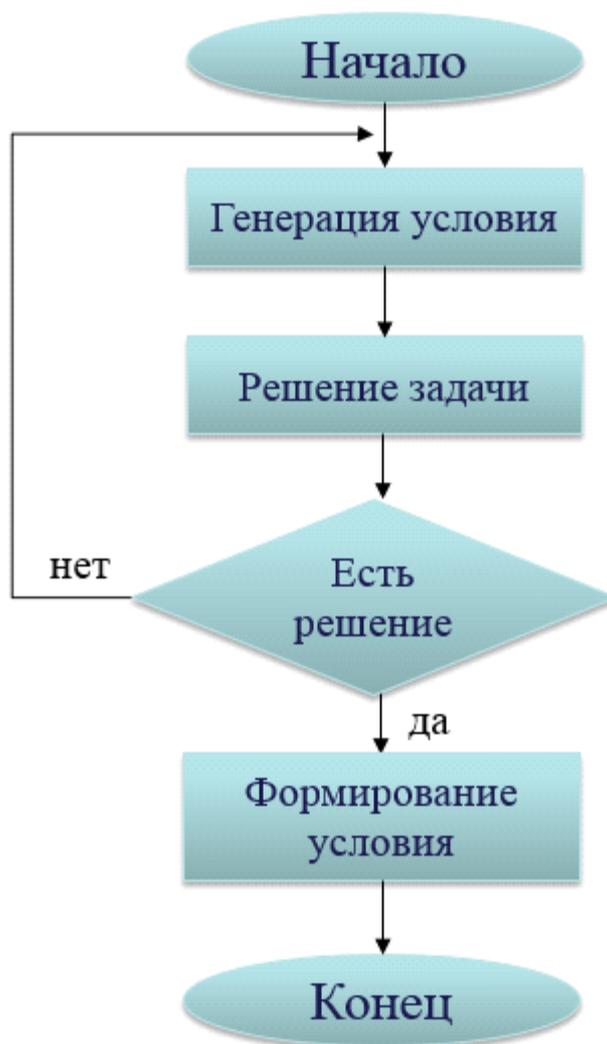


Рис. 1. Алгоритм генерации задания

Шаблон задания генерируется с помощью функции StringJoin. Строка формулировки задания состоит из неизменяемых участков текста и сгенерированных по определенным правилам и специальными алгоритмами изменяемых частей.

Система Wolfram Mathematica поддерживает широко распространенные форматы данных и обеспечивает средства для экспорта и импорта данных. С помощью функции Export можно экспортировать данные в требуемый формат [3, с. 44].

Задания и ответы, сгенерированные в системе, сохраняются в виде файлов в различных форматах, а затем используются в тестовой оболочке. Выбор формата, в который экспортируются сгенерированные зада-

ния, зависит от потребностей составителя тестовых заданий и возможностей системы компьютерного тестирования.

На рис. 2 приведен пример использования сгенерированного условия в системе компьютерного тестирования e_Exam. Здесь изображен слайд системы тестирования, на котором используется экспортированное в формат PNG задание.

The screenshot shows a user interface for the e_Exam system. At the top, there are fields for 'Пользователь' (User) with the value '104', 'Набранных баллов' (Score) with '0', 'Потерянных баллов' (Lost score) with '0', and 'Время' (Time) with '00:21:15'. The main heading is 'Сходимость в метрических пространствах' (Convergence in metric spaces) with a sub-heading 'Задание 1' (Task 1). The task text asks to find the limit of a sequence x_n in the metric space $C[-5, 4]$ if it exists. The sequence is defined as $x_n = \frac{8n^3 t^7 - 4}{10n^3 t^{21} + 7}$. Below the text, there are instructions: 'Введите ответ в поле ниже в соответствии с указаниями по вводу ответов (из файла instructions.pdf). Если предела нет, то в поле ответа необходимо написать нет. После ввода ответа нажмите кнопку подтверждения.' There is an 'Ответ:' label, an input field, a 'Подтверждаю ответ' button, and a '>>' button. A link 'Переход на следующий слайд' is also visible.

Рис. 2. Использование сгенерированного задания

В системе компьютерного тестирования e_Exam есть возможность размещения файлов для скачивания. Поэтому сгенерированные задания можно предоставлять не только в виде текста или изображения, но и использовать для них выбранный автором задания формат файлов.

Отдельного внимания заслуживает формат сохранения ответов сгенерированных заданий в текстовом файле с расширением .txt. Это обусловлено тем, что ответ на задание вводится в поле ответа в клиентской части системы компьютерного тестирования и в таком виде может быть передан на сервер и обработан. А значит и правильный ответ, используемый в качестве эталона для проверки, должен быть представлен также в строковом виде. Но здесь возникает задача формализации правил ввода ответов, связанная с преобразованием выражений-ответов к унифицированному виду, однозначно интерпретируемому как правильный ответ.

Таким образом, генерация тестовых заданий является трудоемким процессом, но созданные генератором задания удобно использовать в системах компьютерного тестирования и тем самым повышать эффективность работы с системой компьютерного тестирования и совершенствовать процесс обучения.

Литература

1. Кручинин В. В. Генераторы в компьютерных учебных программах. – Томск: Изд-во Том. ун-та, 2003. – 200 с.
2. Таранчук В. Б. Введение в язык Wolfram: учебные материалы для студентов факультета прикладной математики и информатики специальности 1-31 03 04 «Информатика». – Минск: БГУ, 2015. – 51 с.
3. Таранчук В. Б. Основы программирования на языке Wolfram: учебные материалы для студентов факультета прикладной математики и информатики специальности 1-31 03 04 «Информатика». – Минск: БГУ, 2015. – 49 с.

РАСПОЗНАВАНИЕ МАТЕМАТИЧЕСКИХ ВЫРАЖЕНИЙ НА ИЗОБРАЖЕНИИ

В. М. Гошко

ВВЕДЕНИЕ

С теоретической и практической точек зрения, распознавание образов – сложная задача. Для распознавания образов человек использует весь накопленный опыт и знания. Кроме того, человек не перестает обучаться и, обладая абстрактным мышлением, может очень точно выделять характерные признаки образов, что при высокой "вычислительной мощности" головного мозга значительно повышает шансы на успешное распознавание.

Часто для улучшения качества распознавания текста каждый символ слова рассматривают в его контексте и используют словари для уточнения результата, что не применимо при распознавании математических формул. Кроме того, структура математической формулы сложнее и более непредсказуема, чем структура текста [1]. По этой причине важность высокой точности распознавания отдельных символов возрастает.

Описание этапов подготовки изображения к распознаванию опущено. Дано описание реализованных алгоритмов. Предложен алгоритм распознавания символов с использованием персептрона и векторного скелета символа, а также комбинированный подход на основе персептрона с последующим уточнением результата, используя метод сравнения с шаблоном. Были проведены замеры производительности некоторых классических алгоритмов и предложенных модификаций на одинаковых входных изображениях.