

РАЗРАБОТКА КОМПЬЮТЕРНОЙ СИСТЕМЫ ОБУЧЕНИЯ РЕШЕНИЮ МНОГОХОДОВЫХ ЗАДАЧ

Рассматривается модель адаптивной компьютерной системы для обучения решению сложных многоходовых задач. Описывается ход решения задачи в виде ориентированного графа, построенного на множестве элементарных операций, а также аспекты индивидуализации и стратегии обучения. Представлен формат для описания многоходовой задачи и приведен пример работы системы с использованием задачи, описанной в данном формате.

Ключевые слова: электронные средства обучения, информационные системы, обучение решению многоходовых задач.

Введение

Компьютерное обучение является активно развивающейся областью в современном образовании. По сравнению с традиционным образованием оно имеет ряд преимуществ: автоматизированный сбор аттестационных данных, минимальное привлечение преподавательского состава, доступность за пределами учебного заведения и др. [1; 2]. В первую очередь, компьютерное обучение обеспечивает студента доступным учебным материалом и способствует развитию способностей к самостоятельному обучению [3]. Преподавателю системы компьютерного обучения предоставляются большие возможности по контролю за успеваемостью студентов и планированию учебного плана [4].

Основой для систем компьютерного обучения послужили работы Э. Торндайка [5], Б. Скиннера [6], Г. Паска [7] и др. в области так называемого программированного обучения. В основе программированного обучения лежат представления об обучении как процессе, основанном на целенаправленном управлении усвоением знаний учащимися, информационном обеспечении и индивидуализации. Основная концепция программированного обучения: «уйти от контроля и перейти к самоконтролю» [8].

Программированное обучение тесно связано с бихевиоризмом – направлением в психологии, которое изучает поведения человека, т. е. всякую его реакцию в ответ на внешний стимул, посредством которой он приспосабливается к окружающей среде [9]. Торндайк утверждал, что обучение человека должно строиться на базе поведенческих, а не сознательных принципов. Поэтому он пытался описать обучение человека с помощью простых правил. Таким образом, он вывел два закона: закон тренировки и закон эффекта. Первый из них гласит: чем чаще повторяется определенная реакция на ситуацию, тем прочнее связь между ними, а прекращение тренировки (повторения) приводит к ослаблению этой связи. Второй закон гласит: если связь между ситуацией и реакцией сопровождается состоянием удовлетворенно-

сти (удовольствия) индивида, то прочность этой связи возрастает. И наоборот: прочность связи уменьшается, если результат действия приводит к состоянию неудовлетворенности [5].

Термин «программированное обучение» впервые был использован Б. Скиннером. В его концепции говорилось о том, что учебный материал должен подаваться мелкими порциями, ответы должны приниматься в открытой форме, и при этом все обучаемые должны проходить один и тот же учебный курс вне зависимости от индивидуальных способностей [6].

Иной подход предложил Н. Кроудер. По его концепции учебный материал дается в виде больших сложных заданий, которые разбиваются на более мелкие, если обучаемый не в силах справиться с исходным. Кроме того, в отличие от концепции Скиннера Кроудер для приема ответов от обучаемого предлагал закрытую форму, другими словами, обучаемый должен выбрать верный вариант ответа из предложенных. Но самым значимым отличием является индивидуализация учебного курса в зависимости от ответов обучаемого, а также подробные пояснения после каждого задания [10].

Самое широкое распространение получила концепция адаптивного обучения, заложенная Г. Паском. По этой концепции обучающая программа должна постоянно адаптироваться под человека, который с ней взаимодействует, корректировать учебный курс, поддерживая оптимальный уровень сложности [7]. Адаптивность может выражаться по-разному. Например, если обучаемый успешно справляется с предлагаемыми ему стандартными тестовыми заданиями, то на следующем шаге ему необходимо предложить более сложные задания. И наоборот: если обучаемый не справляется со стандартными тестами – нужно дать ему задания полегче. Следовательно, система компьютерного обучения должна реагировать на поведение обучаемого таким образом, чтобы он максимально усвоил учебный материал и при этом не потерял интерес к обучению.

В настоящее время существует множество систем компьютерного обучения, как свободных, так и коммерческих, которые реализуют различные концепции программированного обучения. Широкое распространение получили системы адаптивного тестирования (системы тестирования, основанные на концепции Г. Паска). В данной области можно выделить работы Е. Р. Пантелеева [11], И. Г. Жуковой [12], Т. Н. Тягуновой [13] и др.

Преимущество таких тестов заключается в том, что:

- в режиме обучения система подстраивает задания под уровень знаний студента и корректирует учебный курс таким образом, чтобы более «слабый» студент смог решить определенный минимум заданий, а более «сильный» – задания сложнее, чем стандартные;
- в режиме проверки знаний система может определить уровень знаний студента за меньшее число заданий, чем это может быть при обычном тестировании.

Однако при обучении решению задач, решение которых достигается за несколько шагов (например, задачи на доказательство), большинство таких систем имеют существенный недостаток, а именно: в расчет берется только конечный результат, а ход решения не рассматривается. Следовательно, невозможно определить, действительно ли обучаемый понял решение задачи.

Рассмотрим некоторые из существующих систем компьютерного обучения, в которых присутствует частичная реализация наблюдения за ходом решения задачи. Например, система ДИСФОР [14] для решения задач по органической химии, которая используется уже более 20 лет. В ней присутствуют элементы адаптивности, выраженные в диалоге с обучаемым, вывод дополнительных вопросов при неправильных ответах, комментировании ответов. Задания в этой системе задаются с помощью так называемого «графа диалога», что дает возможность задавать и многоходовые задачи. К недостаткам данной системы можно отнести ее узкую специализированность и тот факт, что данная система устарела как вследствие появления новых технологий в программировании, так и из-за изменений в предметной области.

Система Web-обучения ГИПЕРТЕСТ 2.0 [15] предоставляет гораздо больше возможностей для решений многоходовых задач. Здесь задачи описаны в виде раскрашенной сети Петри, что помимо самого описания дает еще и понятную визуализацию хода решения задачи. Однако промежуточные вопросы при каждом повторном решении не меняются в зависимости от успешности ответов обучаемого, другими словами, на уровне решения конкретной задачи отсутствует индивидуализация обучения.

Таким образом, стоит актуальная задача разработки компьютерной системы тестирования и обучения процессу решения сложных многоходовых задач, которая позволила бы полностью индивидуализировать обучение (не только на уровне учебного курса, но и на уровне решения отдельной задачи) и, таким образом, значительно повысить эффективность обучения. Данная задача наиболее актуальна для обучения таким дисциплинам, как математика, физика, экономика, поскольку большинство задач в этих дисциплинах решаются за определенное число шагов.

Индивидуализация и стратегии обучения

Образование – это метод приобретения стандартизированного знания. Поэтому большинство учебных программ в традиционном обучении создаются без учета индивидуальных особенностей обучаемых. И действительно, если учебную программу подстраивать под каждого обучаемого, то пришлось бы затратить огромное количество ресурсов. Однако в компьютерной обучающей системе можно автоматизировать данный процесс. Для того чтобы система могла учитывать индивидуальные особенности обучаемого, между системой и студентом должна присутствовать обратная связь [16]. На основании результатов, полученных при решении задач студентом, система должна принять решение, как продолжать учебный процесс.

Корректировка учебного курса может происходить различными способами, которые определяются стратегией обучения. Данные способы можно классифицировать следующим образом [16; 17]:

- 1) по целям работы с электронным учебным курсом – учитывая пожелания студента (или организаторов проведения обучения) по отношению к изучаемой дисциплине;
- 2) по составу учебного материала в электронном учебном курсе – изменять набор дидактического материала, включаемого в текущую реализацию курса;
- 3) по составу тестовой выборки (другими словами, набора заданий, включенных в тест) при компьютерном тестировании – изменять состав тестовой выборки в зависимости от уровня обученности студента;
- 4) по последовательности предъявления учебного материала – формировать начальную траекторию обучения (с последующей корректировкой, вырабатывать рекомендации для повторения и проч.).

Реакция системы после получения ответа (верного или неверного) может существенно зависеть от цели проведения теста. Возможны, в частности, следующие предельные случаи.

- Тестирование с целью контроля знаний (отсутствие обучения). Система задает серию вопросов, не давая никаких комментариев и подсказок.
- Тестирование с целью обучения (сочетание тестирования и обучения). На каждый неверный или неточный ответ система выдает развернутое объяснение либо пытается подвести обучаемого к верному решению через ряд дополнительных вопросов.
- Демонстрация (отсутствие тестирования). Система проводит обучаемого по всем шагам решения.

Алгоритм поведения системы (или стратегия тестирования) не зависит от конкретного теста и, будучи формализован набором параметров, выбирается лицом, проводящим тест, исходя из текущих учебных задач. Также стратегия тестирования может включать алгоритм выбора степени «дотошности» или «поверхностности» опроса и критерии его завершения (которое, как правило, ускоряется при особенно высоких или особенно низких текущих показателях).

Такая схема взаимодействия обучаемого и системы похожа на диалог преподавателя и студента на экзамене. Если студент уверенно отвечает на все вопросы преподавателя, то преподаватель не задает ему дополнительных вопросов и ставит оценку. Он видит, что студент знает материал. В обратном же случае количество дополнительных вопросов увеличивается.

По завершении теста (успешном или принудительном) система проводит анализ ответов и оценку затраченного времени и на основании полученных результатов выводит на экран балл за решение задачи и персональные рекомендации о том, какой материал следует изучить обучаемому дополнительно для достижения лучшего результата.

Опираясь на данные, полученные в результате решения студентом задач, система сможет не только выдавать индивидуальные задачи, но задавать дополнительные вопросы при решении задачи там, где это будет действительно необходимо. Таким образом, за счет адаптации общего учебного курса под каждого студента повысится результативность в обучении студентов и, следовательно, возрастет эффективность всей системы.

Представление хода решения задачи

Задачи, решение которых достигается за несколько шагов, наиболее характерны для математики, физики, экономики и технических наук. Кроме того, среди них есть такие задачи, которые могут быть решены несколькими способами. Таким образом, все множество шагов решения задачи можно представить в виде множества ее элементарных действий, а решением задачи будет ориентированный граф, проходящий по вершинам множества решений (рис. 1).

Элементарное действие (операция) может быть представлено как функциональное преобразование $F_k : I_k \rightarrow O_k$ вектора входных параметров в вектор-результат. Такое преобразование либо может считаться неразложимым (что необходимо для таких простейших операций, как, например, сравнение чисел), либо выразаться через некую иерархию более простых действий (рис. 2). Например, для того чтобы найти косинус угла между двумя векторами, необходимо скалярное произведение этих векторов разделить на произведение абсолютных величин (модулей) этих векторов. В свою очередь, операция «косинус угла между двумя векторами» зависит от операций «абсолютная величина вектора», «скалярное произведение векторов», а также от операций «произведение» и «деление». Операция «скалярное произведение векторов» также имеет зависимости.

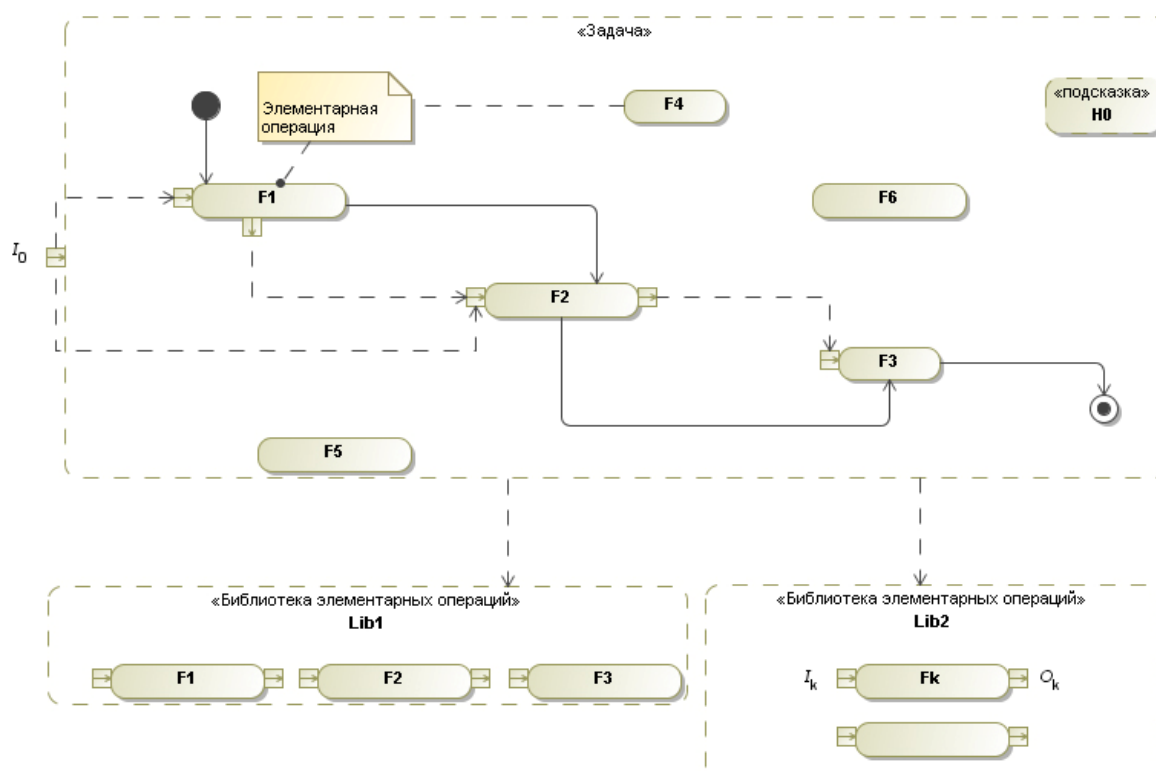


Рис. 1. Представление множества шагов решения задачи в виде ориентированного графа

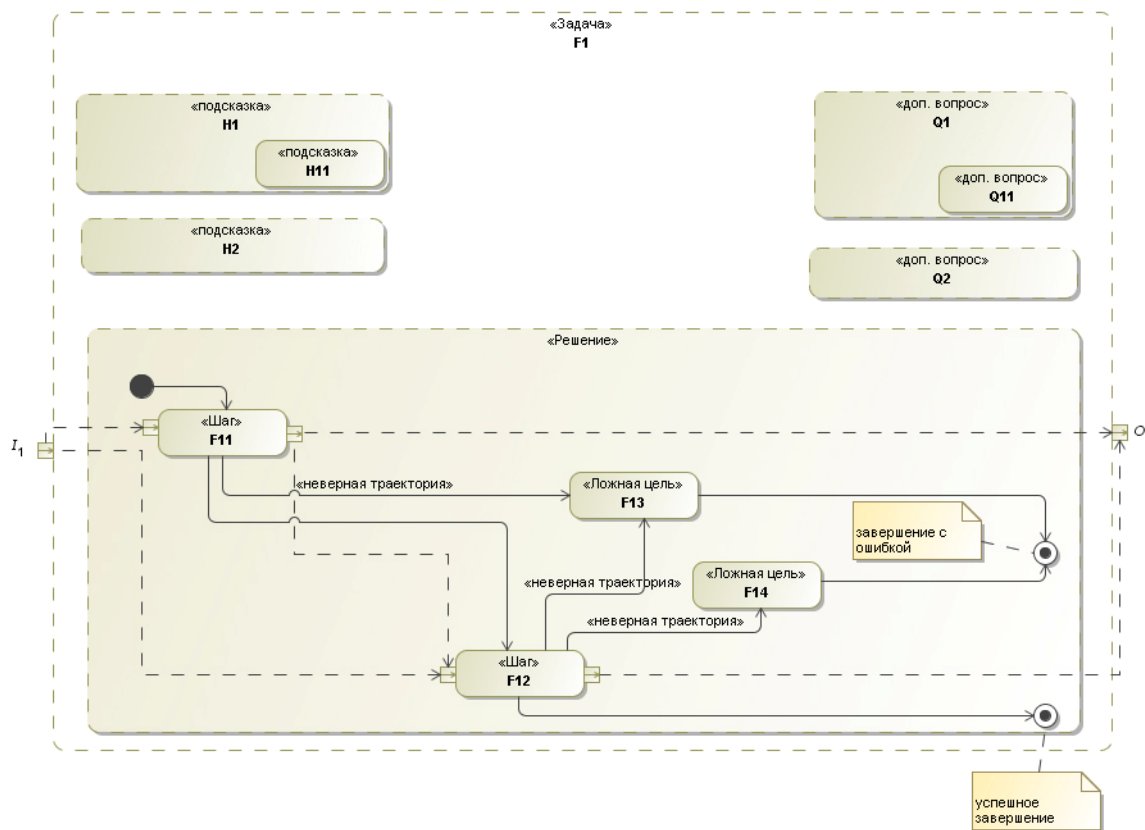


Рис. 2. Представление элементарной операции

Элементарная задача («элементарность» понимается здесь в относительном смысле) может иметь процедуру решения, а также множество дополнительных вопросов и подсказок, в свою очередь устроенных иерархически (см. рис. 2).

Таким образом, множество действий задачи составляют только те операции, которые входят в ориентированный граф решений. Этого достаточно для демонстрации решения задачи, но для проверки знаний и обучения необходимо ввести «ложные цели» – тупиковые вершины в ориентированном графе решений (см. рис. 2). Если студент указал такой путь, который заканчивается ложной целью, то это означает, что задача решена неверно. Практически это может выглядеть следующим образом: система предлагает студенту на текущем шаге выбрать правильный ответ из нескольких предложенных вариантов. Если он выбирает неверный вариант, то это означает, что он попал в тупиковую вершину.

Следовательно, решение задачи сводится к нахождению правильного маршрута на множестве действий, который соответствовал бы маршруту в ориентированном графе решений, проходящем от исходной точки (начальные условия задачи) до конечной (результат решения задачи). При этом во множестве действий присутствуют не только операции, используемые для решения задачи, но и ложные цели.

Описание тестовых задач

В настоящее время авторами предпринимаются попытки создания языка описания обучающих тестов (т. е. задач, на которых система может тестировать студентов, обучать их или делать то и другое в разумном сочетании), достаточно богатого полезными для реальных задач конструкциями.

Рассмотрим сначала простой пример (рис. 3).

```

<task description="Определение квадрата на плоскости">
  <question>
    Даны четыре точки на плоскости: A, B, C, D. Требуется доказать, что ABCD - квадрат.
    Укажите необходимые действия.
  </question>
  <solution>
    <unordered>
      <suggestions>
        <variant id="mod">Определение модуля вектора</item>
        <variant id="scalar">Определение скалярного произведения векторов</item>
        <variant id="angle">Вычисление угла между векторами</item>
        ...
      </suggestions>
      <variant ref="mod">
        <question>
          Модули каких векторов нужно найти?
        </question>
        <solution>
          <unordered csv="AB,BC,CD,AD" />
        </solution>
        <addQuestion>
          <question>
            Определите модуль вектора AB, где  $A(\{0\}; \{1\})$  и  $B(\{2\}; \{3\})$ 
          <answer>{4}</answer>
          <params>
            <variant csv="5,6,2,2,5" />
            <variant csv="-1,7,5,-1,10" />
            ...
          </params>
          </question>
        </addQuestion>
      </variant >
      <variant ref="angle">
        ...
      </variant >
    </unordered>
  </solution>
</task>

```

Рис. 3. Описание задачи об определении квадрата на плоскости

Данный xml-документ описывает задачу как совокупность вопроса (question) и решения (solution), причем решение представляет собой набор шагов без определенной последовательности (unordered). В качестве шага выбирается операция из числа предложенных (suggestions). Для правильного решения должны быть выбраны операции «Определение модуля вектора» и «Вычисление угла между векторами». Сами эти операции могут рассматриваться как подзадачи (в зависимости от выбранной стратегии тестирования), включая вопрос и решение. Кроме того, задачи могут включать дополнительные вопросы (addQuestion), также по сути своей являющиеся подзадачами. В данном примере дополнительный вопрос носит вычислительный характер, с несколькими вариантами условия.

В диалоговом режиме решение может выглядеть следующим образом (рис. 4).

Следует отметить, что данные, введенные студентом, проверяются и фильтруются надлежащим образом прежде, чем начнется их обработка. Например, на шаге «Модули каких векторов нужно найти?» (см. рис. 4), не нужно соблюдать определенную последовательность векторов, чтобы ответ был засчитан системой как верный. Кроме того, поскольку в данной задаче на данном шаге не важно, какая точка будет началом вектора и какая точка будет концом, то одинаково верными будут записи векторов AD и DA , BC и CB и т. д.

Даны четыре точки на плоскости: A, B, C, D . Требуется доказать, что $ABCD$ - квадрат. Укажите необходимые действия.

- Определение модуля вектора
- Определение скалярного произведения векторов
- Вычисление угла между векторами
- Вычисление проекции вектора на ось
- Вычисление определителя
- Решение уравнения
- Преобразование координат

Верно.

Модули каких векторов нужно найти?

AB, BC, CD, AD

Верно. Дополнительный вопрос:

Определите модуль вектора AB , где $A(-1; 7)$ и $B(5; -1)$

10

Верно.

Укажите дальнейшие действия.

- Определение модуля вектора
- Определение скалярного произведения векторов
- Вычисление угла между векторами
- Вычисление проекции вектора на ось
- Вычисление определителя
- Решение уравнения
- Преобразование координат

OK

Рис. 4. Решение задачи об определении квадрата на плоскости

Для более сложных задач, очевидно, желательна повторная используемость и самодостаточность описаний составляющих их операций, которые следует сгруппировать в модули (такие как «Аналитическая геометрия на плоскости») и в категории с подкатегориями (например, «Операции с векторами», «Решение уравнений» и т. д.), а также четко определить требования к контексту выполнения (в частности, необходимые входные аргументы операций). Тогда становится возможным, задавшись набором базовых задач-операций в конкретной предметной области, конструировать определения задач более высокого уровня из базовых элементов, связывая их посредством таких объектных отношений, как обобщение (специализация), агрегация и композиция.

Для иллюстрации рассмотрим более сложный пример (рис. 5).

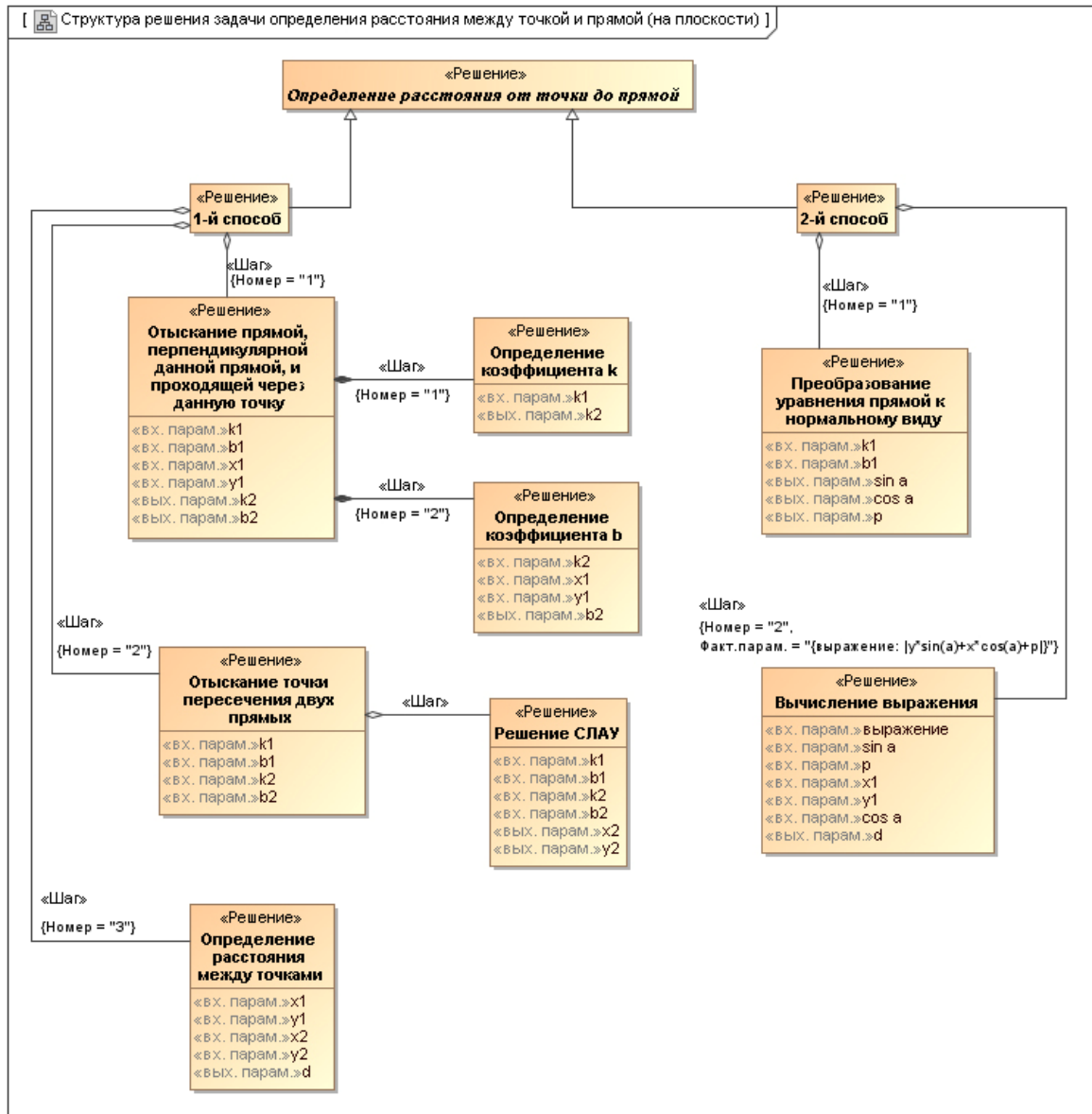


Рис. 5. Структура решения задачи определения расстояния между точкой и прямой (на плоскости)

Здесь структура решения задачи показана в нотации UML как совокупность стереотипизированных классификаторов (решений подзадач) со стереотипизированными связями (элементами или упорядоченными шагами). Тот факт, что основная задача имеет несколько решений, на языке объектного подхода можно выразить как решение-абстракцию, имеющую несколько конкретных разновидностей-реализаций (следует иметь в виду, что разные решения, как правило, не равноценны, и тогда «стоимость» решения также должна быть отражена на диаграмме и в коде описания задачи). Далее, решение задачи некоторого уровня состоит из решения набора подзадач (с учетом порядка или без), что, как правило, имеет смысл агрегации (т. е. решение-агрегат содержит решения-части, которые, в принципе, могут принадлежать и другим агрегатам). На первом шаге первого способа решения, однако, обе подзадачи имеют смысл лишь в контексте объемлющей задачи, что можно трактовать как композицию (т. е. безраздельную принадлежность задаче подзадач).

Возможность визуального представления структуры задачи особенно полезна для создания пользовательского интерфейса программы-оболочки, генерирующей xml-описания задач.

На рис. 5, однако, показаны только сами решения и во избежание усложнения опущены такие элементы задачи, как ее условие, стоимость в баллах, предельное время решения, предлагаемые варианты, дополнительные вопросы, подсказки, ссылки на источники информации и т. д.

Важно, чтобы с помощью языка описания тестовых заданий можно было описать не только отдельную задачу, но и целые классы заданий по различным дисциплинам. Таким образом, при разработке этого языка авторы руководствовались тем, что он должен быть одновременно простым, понятным и емким.

Заключение

В результате проведенных исследований была разработана модель компьютерной обучающей системы для обучения решению многоходовых задач, формат описания тестовых заданий на языке разметки XML, а также прототип компьютерной обучающей системы, который частично реализует данную модель и использует задачи аналитической геометрии, описанные в разработанном формате. Прототип системы показал работоспособность модели на разнообразных задачах и возможность расширения круга задач за счет повторного использования существующих. В настоящее время авторы работают над полноценной реализацией этой модели. Выходные данные, полученные от работы такой системы, будут представлять немалую ценность для дальнейших исследований в этой области.

Список литературы

1. Добронец Б. С., Шмагрис Ю. В. Компьютерное обучение процессу решения сложных задач // Повышение качества высшего профессионального образования: Материалы Всерос. науч.-метод. конф. Красноярск, 2007. Ч. 1. С. 297–301.
2. Добронец Б. С., Углев В. А., Устинов В. А. Системный подход к процессу обучающего компьютерного тестирования // Информационные технологии. 2008. № 4. С. 81–87.
3. Добронец Б. С. Информационная система обучения решению сложных задач // Теоретические и прикладные вопросы современных информационных технологий: Материалы Всерос. науч.-техн. конф.: В 2 ч. Улан-Удэ, 2009. Ч. 1. С. 270–273.
4. Ващенко Г. В., Добронец Б. С. АРМ преподавателя в рамках информационной образовательной среды вуза // Повышение качества высшего профессионального образования: Материалы Всерос. науч.-метод. конф.: В 3 ч. / Под ред. С. А. Подлесного. Красноярск, 2009. Ч. 3. С. 36–39.
5. Торндайк Э. Процесс учения у человека: Пер. с англ. / Под ред. С. Е. Гайсиновича. М., 1935.
6. Skinner B. F. Science and Human Behavior. The B. F. Skinner Foundation. 1953. URL: http://www.bfskinner.org/bfskinner/Society_files/Science_and_Human_Behavior.pdf
7. Pask G. The Cybernetics of Human Learning and Performance. L.: Hutchison. 1975. 347 p.
8. Скибицкий Э. Г., Толстова И. Э., Шефель В. Г. Методика профессионального обучения: Учеб. пособие. Новосибирск, 2008. 166 с. URL: <http://txtb.ru/88/>
9. Ждан А. Н. История психологии: Учебник. М.: Изд-во МГУ, 1990. 367 с. URL: http://www.pedlib.ru/Books/1/0288/1_0288-1.shtml
10. Краудер Н. О различиях между линейным и разветвленным программированием // Программированное обучение за рубежом: Сб. ст. / Под ред. И. И. Тихонова. М.: Высш. шк., 1968. С. 58–67.
11. Пантелеев Е. Р. Среда разработки программ дистанционного обучения и профильного тестирования ГИПЕРТЕСТ: инструментальные средства // Информационные технологии. 2001. № 8. С. 34–40.
12. Жукова И. Г., Сипливая М. Б., Шабалина О. А. Концепция открытой адаптивной контрольно-обучающей системы на основе персонализации процесса обучения // Информационные технологии. 2003. № 3. С. 20–23.
13. Тягунова Т. Н. Философия компьютерного тестирования Текст. М., 2003. 246 с.

14. Мануйлов А. В., Родионов В. И. Система ДИСФОР: опыт компьютеризации обучения органической химии в НГУ // Эффективность информационных технологий в высшей школе: Материалы Межвуз. науч.-практ. конф. Новороссийск; М.: Изд-во МГУ, 1994. С. 42–44.
15. Пантелеев Е. Р., Карпов Я. Э. Модели и методы интерпретации решения расчетных задач в среде web-обучения // Вестн. ИГЭУ. 2009. Вып. 3. С. 66–70
16. Углев В. А. О специфике индивидуализации обучения в автоматизированных обучающих системах // Философия образования. 2010. № 2. С. 68–74
17. Углев В. А., Устинов В. А., Добронетц Б. С. Модель структурной адаптации электронных учебных курсов с помощью обучающего компьютерного тестирования // Вестн. Новосиб. гос. ун-та. Серия: Информационные технологии. 2009. Т. 7, вып. 2. С. 74–87.

Материал поступил в редколлегию 17.09.2012

Yu. A. Krapivko, P. A. Svetashkov, B. S. Dobronets

**DEVELOPMENT OF THE COMPUTER-AIDED TRAINING SYSTEM
OF MULTI-STEP PROBLEMS SOLVING**

The article describes the way of computer-aided training system of multi-step problems solving development. It provides information about existing e-learning technologies, notion of problem solving way by the elementary operations oriented graph, individualization and teaching strategy. It suggests a multi-step problem description format, based on XML, and provides with system implementation example, which can read such description.

Keywords: e-learning, information system, teaching of multi-step problems solving.