

V. R. Pen, I. V. Kovalev, S. I. Levchenko

KINETICS OF DESTRUCTION OF POLYMERS WITH DISTRIBUTED IN KINETIC PARAMETERS AS STOCHASTIC PROCESS

It is shown possibility to present the kinetic model polychronic process polymer of the irregular construction, at row of the suggestions, as stochastic process that enables to solve the getting equation of Kolmogoroff and find the evident type to density of the distribution for kinetic ensembles. Find solution allows to simulate the destruction kinetics of polymers of the irregular construction having several types of the chemical relationships.

УДК 004.588

А. В. Редькина

ОБУЧЕНИЕ СИНТЕЗУ АЛГОРИТМОВ¹

Рассмотрены вопросы разработки обучающих программ, позволяющих самостоятельно вести поиск решения задач и контролирующих на каждом этапе уровень усвоения знаний. Предложен алгоритм обучения процессу алгоритмизации в курсе программирования с использованием автоматизированной обучающей системы.

Рост интеллектуализации отраслей и технологий порождает все больший спрос на профессиональные и высококвалифицированные кадры. Поэтому работы по интеграции новых информационных технологий в образовательный процесс как для повышения индивидуализации обучения и развития адаптивных методов обучения, так и для дистанционного образования представляются весьма перспективными.

Одной из основных целей внедрения современных автоматизированных систем учебного назначения является повышение доступности и качества профессиональной подготовки обучаемых. Однако, как свидетельствуют факты, существенных успехов в этом направлении удается достичь далеко не всегда. В первую очередь это относится к сфере инженерного и естественно-научного образования, где важную роль в процессе обучения играют практические занятия по решению задач и выполнению лабораторных работ. Основным направлением решения проблемы информатизации инженерно-технического образования является разработка обучающих программ, охватывающих широкий спектр образовательных задач и ориентированных в первую очередь на автоматизацию практикумов, позволяющих приобрести профессиональные навыки [1]. Здесь необходимо отметить работы А. В. Соловова, И. П. Норенкова, А. М. Зимина, С. И. Маслова, А. А. Полякова и др. Для повышения дидактической эффективности обучающих программ все чаще используются технологии экспертных систем.

В настоящее время разработаны мощные информационно-образовательные среды (Learning Space, WebCT, BlackBoard, MOODLE), обеспечивающие поддержку методического комплекса дисциплин и представляющие ту часть знаний, которая называется декларативными зна-

ниями и может быть преобразована в тексты, рисунки, видеоролики и т. д., но не позволяющие вырабатывать процедурную часть знаний – навыки и умения. Существуют свои программы и для обучения процедурной части знания, но они имеют разрозненный характер и не интегрированы в единый учебно-методический комплекс. Чаще всего эти программы решают частные задачи и не обеспечивают поддержку полной траектории обучения, включающей представление знаний, выработку навыков и умений, а также контроль полученных декларативных и процедурных знаний.

Одна из дисциплин, которую нельзя освоить без решения задач, – это программирование. Существуют различные методики обучения программированию. В 1960–1970 гг. основой курса программирования была алгоритмизация, хотя с появлением персональных компьютеров акценты в обучении были смещены в сторону кодирования. Но, тем не менее, именно раздел «Алгоритмизация» определяет всю дальнейшую квалификацию программиста. Этот раздел является сложным для начинающих программистов и трудоемким для преподавателей.

Проблема усугубляется разным стартовым уровнем обучаемых. При обучении программированию необходимо выработать алгоритмическое мышление, а сделать это без индивидуального подхода к каждому студенту невозможно. Необходимость разработки обучающей системы, адаптивной к уровню знаний обучаемых, также диктует и развитие дистанционного образования.

Для исследования возможностей, представляемых для обучения синтезу алгоритмов, был проведен анализ программ, обучающих языкам программирования.

В настоящее время среди программ, содержащих элементы обучения алгоритмизации, можно выделить олим-

¹ Работа выполнена при финансовой поддержке научно-методического проекта Сибирского федерального университета «Решение некоторых задач прикладной математики и информатики для повышения потенциала образовательного процесса».

пиадные сайты по программированию, где разбираются и тестируются задачи в режиме online, а также программы «Конструктор блок-схем (Flowchart builder)» Нижегородского государственного университета и «AlgoMaster» Национального технического университета Украины (Киевского политехнического института). Они предназначены для исследования алгоритмов и содержат подсистемы интерактивного ввода блок-схем, интерпретации и анимации. Эти программы являются хорошим дидактическим материалом, но они не поддерживают элементов контроля знаний, не анализируют блок-схем, не проверяют правильность построения и не предлагают задачи к решению, т. е. не имеют четкой ориентации на полный цикл обучения алгоритмизации. Таким образом, средств обучения, выполняющих полный цикл обучения алгоритмизации и являющихся сетевыми приложениями, для начинающих программистов нет.

С другой стороны, имеются мощные информационно-образовательные среды, обеспечивающие общение преподавателя и обучаемого через Интернет, позволяющие обучать декларативным знаниями и контролировать эти знания. Поэтому расширение функциональности информационно-образовательных сред через предоставление возможности получать процедурные знания, разрешающее эвристические действия обучаемого, является актуальной задачей.

Для решения поставленной задачи информационные обучающие среды были проанализированы по следующим критериям:

- функциональной полноте;
- переносимости;
- наличию стандартизированных инструментальных средств адаптации и сопровождения;
- открытости кода;
- ценовой политике;
- соответствию международным стандартам поддержки дистанционного обучения.

В результате исследования на функциональную полноту выяснилось, что информационно-образовательные среды Learning Space, WebCT, BlackBoard, MOODLE обладают примерно равными функциональными возможностями. Однако среда Learning Space достаточно сложна для установки, эксплуатации и дополнения собственными программными наработками, а среды WebCT, BlackBoard очень дороги. Среда MOODLE (Modular Object-Oriented Dynamic Learning Environment, модульная объектно-ориентированная динамическая обучающая среда) не требует коммерческого программного обеспечения для организации полноценного образовательного сервера. Кроме того, открытый код и изначальная ориентация этой системы на развитие, дополнение и изменение позволяет легко воплотить замысел преподавателя или студента. Поэтому при разработке автоматизированной системы обучения, расширяющей функциональность информационно-образовательной среды и позволяющей обучаемым действовать эвристически, была выбрана среда для обучения MOODLE.

Традиционно обучение ведется по следующему алгоритму (рис. 1). Преподаватель дает теоретический материал (основные понятия, свойства, способы описания),

технологии и примеры решения типовых задач, затем обучаемым предлагается самостоятельно решить задачу. Если задача не решается, то преподаватель поясняет процесс решения, а если и после этого задача студентами не решена, то предлагается готовое решение. Когда индивидуальное задание выдается каждому обучаемому, то преподавателю необходимо понять логику десятков алгоритмов, найти и объяснить ошибки либо предложить готовый алгоритм. Процесс получается трудоемким. К тому же часть студентов требует дополнительного внимания из-за низкого стартового уровня или психологических особенностей. Таким образом, главным недостатком традиционного обучения является отсутствие индивидуализации и адаптивности процесса обучения.

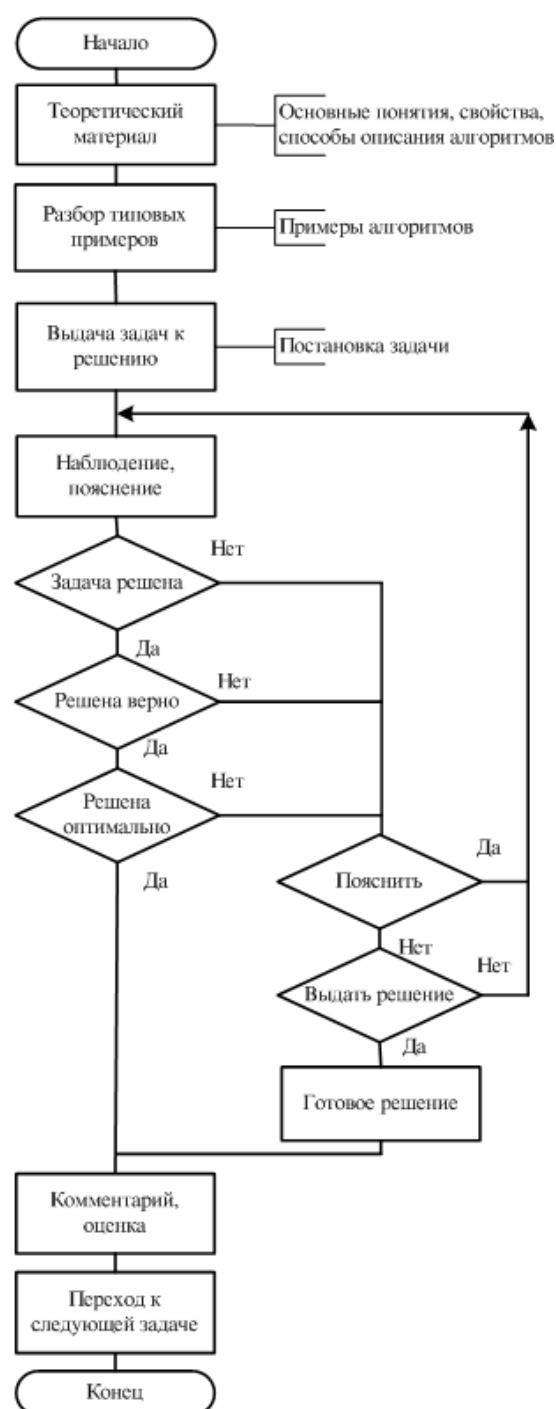


Рис. 1. Алгоритм традиционного процесса обучения

Подходы к автоматизации процессов обучения основаны на исследовании процессов представления знаний и контроля их усвоения. Поскольку автоматизируется процесс обучения решению задач, то выполнить контроль усвоения знаний с помощью традиционного тестирования невозможно. И самым трудоемким здесь является процесс проверки правильности алгоритма. Каким же образом можно провести эту проверку? Известно, что для одной и той же задачи можно составить несколько верных алгоритмов, поэтому проверка на их идентичность не позволит вынести окончательный вердикт. Правильность построенного алгоритма предложено проверять в два этапа: на первом этапе проводить семантический анализ алгоритма, а на втором – сравнивать результаты работы двух блок-схем, одна из которых составлена экспертом, а вторая – обучаемым (рис. 2). Преподаватель освобождается от части рутинной работы, а студент имеет возможность самостоятельно решать задачи, получая информацию о своих достижениях. Но процесс обучения алгоритмизации не может быть полностью автоматизирован, так как разрабатываемое программное обеспечение есть средство обучения, способствующее усвоению материала и преподаватель по-прежнему остается центральной фигурой, определяющей ход учебного процесса.

Формализация траектории обучения синтезу алгоритмов проведена согласно этапам познавательной деятельности теории познания и представлена следующей многоэтапной моделью:

- этап А – демонстрационный (восприятие). На этом этапе обучаемый получает возможность познакомиться и детально проанализировать типовые алгоритмы с помощью визуализатора. Исполнитель запускает алгоритм на выполнение по шагам, позволяет вводить исходные данные, выделяет графически активный блок, а также позволяет контролировать значения переменных.

- этап В – обучение по аналогии (осмысление и фиксация знаний). Здесь появляется возможность выбрать условие задачи, найти аналог среди типовых алгоритмов, модифицировать его и протестировать визуализатором. На этом этапе обучающая система подключает семантический анализатор, который контролирует, соответствует ли структура алгоритма правилам построения, и только потом сравнивает результаты работы алгоритмов, полученных от эксперта и обучаемого, и на основании этого выдает сообщения;

- этап С – обучение по конечному результату (фиксирование личного опыта). На этом этапе студент полностью самостоятельно решает выбранную задачу, пользуясь редактором и визуализатором, а обучающая система контролирует правильность построения и верность результатов;

- этап D – обучение в эксперименте (поисковая деятельность). На последнем этапе обучаемый строит блок-схему алгоритма произвольной задачи, обучающая система проверяет соответствие структуры блок-схемы алгоритма правилам построения графов-схем и позволяет провести ее трассировку, а обучаемый несет ответственность за верность результата.

Для каждого этапа данной траектории разработаны свои алгоритмы процесса обучения. Благодаря много-

этапной конструкции эту классическую траекторию можно легко модифицировать. Если по классической траектории обучение идет от простого к сложному, то на произвольной траектории считается, что студент лучше знает, что ему изучать. Но в любом случае усвоение знаний контролируется, и если возникает необходимость, то произвольная траектория превращается в управляемую. Кроме того, если автоматизированную обучающую систему использовать только при обучении на этапе С, то она превращается в тренажер, а если на этапе D, то в редактор блок-схем.

На основе предложенной многоэтапной модели управления процессом обучения и алгоритмов обучения разработана структурная схема автоматизированной обучающей системы (АОС) на основе экспертной системы, содержащей обучающее ядро и обладающей языком общения, который совпадает с языком проблемной области (языком блок-схем), что позволяет обучающимся самостоятельно решать задачи, а также имеющей подсистему подсказок. База знаний хранит формулировки задач, их решения в виде блок-схем, тесты для проверки, комментарии. Кроме того, в отличие от традиционного процесса обучения, для повышения наглядности обучаемому предоставляется возможность самостоятельно провести исследование блок-схемы алгоритма путем трассировки и визуализации.

В ходе разработки автоматизированной системы обучения решались проблемы построения базы знаний, редактора блок-схем, преобразователя блок-схемы в промежуточное представление, семантического и синтаксического анализаторов, интерпретатора, визуализатора блок-схем, блока оптимальности.

Моделирование АОС выполнено на языке Visual C++. Сформирована инструментальная среда «Algorithm», которая включает анализатор, редактор и визуализатор графических схем алгоритмов. Апробация этой среды проводилась на примере усвоения курса «Программирование на языке высокого уровня» в Политехническом институте Сибирского федерального университета. Цель эксперимента – определить качественные изменения уровня усвоения учебного материала, возникающие при применении АОС.

В эксперименте принимало участие 4 группы студентов (105 человек), разделенных на два потока. При этом основной поток состоял из двух групп (первая группа – 27, вторая группа – 25 студентов) и обучался по обычной форме подготовки (без применения АОС). Студенты двух других групп (третья группа – 27, четвертая группа – 26 студентов), входили в контрольный поток и обучались с использованием АОС. Для повышения эффективности обучения, исходя из возможностей технического оснащения учебных аудиторий, группы разбивались на подгруппы в количестве от 12 до 14 студентов. По окончании курса обучения алгоритмизации студентам всех групп было выдано итоговое задание.

Результаты экспериментов приведены в таблице.

Анализ этих данных показывает, что контрольный поток, который обучался с использованием АОС, справился с итоговым заданием с меньшим количеством попыток. Таким образом, повысилась интенсивность обуче-

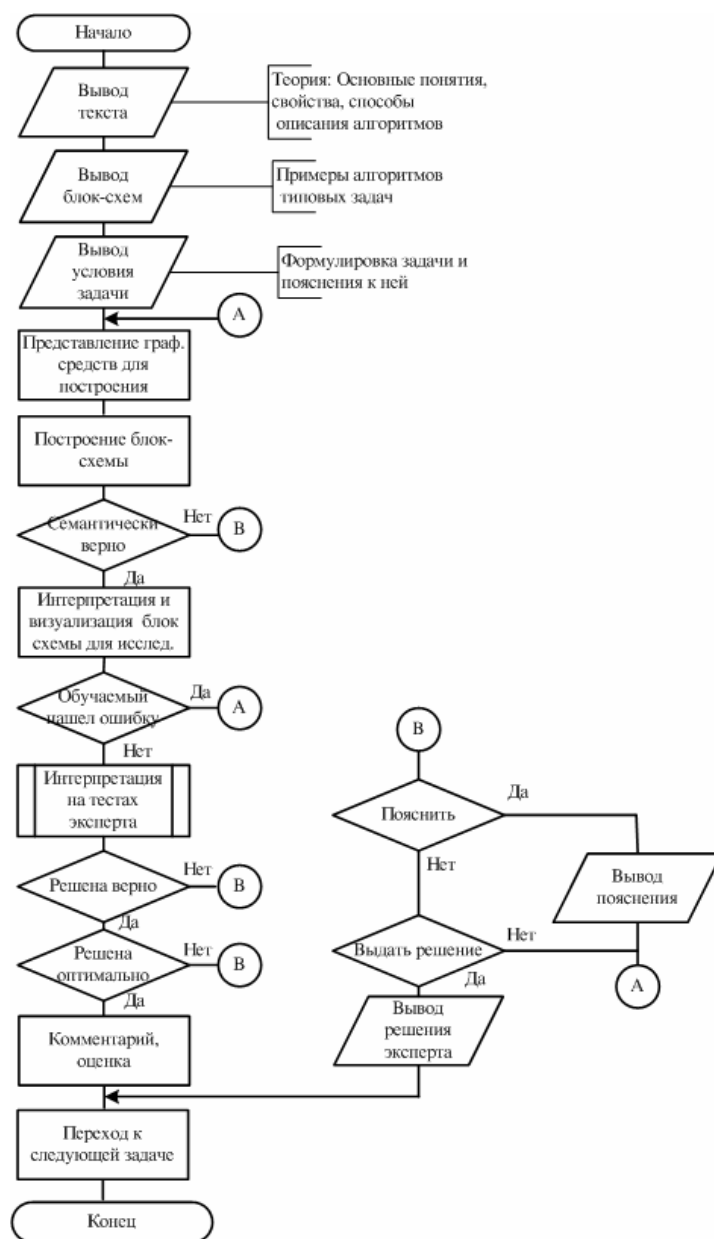


Рис. 2. Алгоритм автоматизированного процесса обучения

Сравнение результатов выполнения итогового задания

Показатель	Основной поток				Контрольный поток			
	Группа 1		Группа 2		Группа 3		Группа 4	
	Подгруппа 1	Подгруппа 2	Подгруппа 1	Подгруппа 2	Подгруппа 1	Подгруппа 2	Подгруппа 1	Подгруппа 2
Число слушателей	13	14	13	12	13	14	13	13
1 попытка	3	5	2	4	4	5	6	7
2 попытки	5	5	6	4	5	6	4	4
3 попытки	5	4	5	4	4	3	3	2
Среднее количество попыток по подгруппе	2,14	2,00	2,23	2,00	2,00	1,70	1,77	1,53
Среднее количество попыток по группе	2,07		2,12		1,85		1,65	
Среднее количество попыток по потоку	2,08				1,75			

ния, а время, затрачиваемое на обучение, уменьшилось. Результаты экспериментов подтверждают эффективность использования АОС: повышается качество подготовки студентов, возрастает мотивация в обучении, сокращается время, затрачиваемое преподавателем на рутинную работу по контролю правильности решения задач, на подборку тестов.

Таким образом, автоматизированная обучающая система «Algorithm» способствует формированию практических знаний (умений, навыков), адаптируется к индивидуальным способностям обучаемых, позволяет вести эвристический поиск решения задач, контролирует степень обученности. Она может применяться при изучении курсов «Программирование на языке высокого уровня», «Информатика», «Информатика и программирование», «Математическая логика и теория алгоритмов» и т. д.

В ходе работы были решены следующие задачи:

- предложен подход к автоматизации труда преподавателя при получении и закреплении приемов и навыков синтеза алгоритмов;

- разработана многоэтапная модель процесса обучения синтезу алгоритмов, отличительной чертой которой является свойство динамической адаптации и индивидуализации процесса обучения применительно к особенностям восприятия и усвоения материала каждого обучаемого;

- на основе предложенной модели разработан обобщенный алгоритм процесса обучения синтезу алгоритмов;

- разработано программное и алгоритмическое обеспечение, реализующее принципы работы обучающей системы на основе экспертной системы, содержащей базу знаний, редактор, семантический анализатор, исполнитель и визуализатор блок-схем;

- проведены исследования эффективности использования АОС, в ходе которых было доказано, что интенсивность и качество обучения возрастают, также растет творческая активность обучаемых, усиливается образность мышления, что способствует хорошему закреплению полученных знаний, умений и навыков.

Библиографический список

1. Федеральная целевая программа развития образования на 2006–2010 годы [Электронный ресурс]. Электрон. дан. Режим доступа: <http://www.ed.gov.ru>. Загл. с экрана.
2. Норенков, И. П. Управление знаниями в информационно-образовательной среде / И. П. Норенков // Системы открытого образования [Электронный ресурс]. Электрон. журн. Режим доступа: <http://www.engineer.bmstu.ru>. Загл. с экрана.
3. Соловов, А. В. Проектирование компьютерных систем учебного назначения: учеб. пособие / А. В. Соловов; Самар. гос. авиац. ун-т. Самара, 1995.
4. Норенков, И. П. Информационные технологии в образовании / И. П. Норенков, А. М. Зимин; Моск. гос. техн. ун-т. им. Н. Э. Баумана. М., 2004.

A. V. Redkina

TEACHING TO SYNTHESIS OF ALGORITHMS

The problems of development of the teaching programs allowing independently to conduct search of the decision of problems and supervising at each stage a level of mastering of knowledge are considered. The algorithm of teaching to algorithmization process with the use of automated courseware in the course of programming is proposed.

УДК 62–83:621.63

А. В. Зуев, В. И. Иванчура

ЧАСТОТНО-УПРАВЛЯЕМЫЙ АСИНХРОННЫЙ ЭЛЕКТРОПРИВОД ВЕНТИЛЯТОРА ДЕКАРБОНИЗАТОРА

Рассмотрена замкнутая система автоматического регулирования расхода воздуха в процессе декарбонизации воды на основе частотно-управляемого асинхронного электропривода. Определена и смоделирована экспериментальная зависимость требуемого расхода воздуха в функции основных измеряемых параметров.

Надежность и экономичность теплоэнергетических установок и систем теплоснабжения в значительной мере зависят от интенсивности внутренней коррозии оборудования и трубопроводов под действием циркулирующей в них воды. Коррозионная агрессивность воды обусловлена рядом физико-химических факторов, среди которых одним из главных является присутствие в воде растворенных газов. К наиболее распространенным и опасным коррозионно-агрессивным газам относятся кислород и диоксид углерода CO_2 (другие назва-

ния CO_2 – свободная углекислота, углекислый газ, угольный ангидрид).

Удаление из воды растворенных газов является важной составной частью технологических процессов подготовки добавочной питательной воды паровых котлов и подпиточной воды систем теплоснабжения. В качестве основного метода удаления растворенных газов используется физическая десорбция (декарбонизация).

Отсутствие строгих требований к массообменной эффективности декарбонизаторов длительное время было