

Е. Л. З о р и н, М. В. Ф и л и п п о в,
Н. В. Ч и ч в а р и н

РАЗРАБОТКА ЗАЩИЩЕННЫХ ПРОГРАММНЫХ СРЕДСТВ ОБУЧЕНИЯ С ПРИМЕНЕНИЕМ СЕТЕЙ ПЕТРИ

Основой публикации являются результаты проведенных исследований по разработке систем программных средств, сопровождающих учебный процесс, и выбору метода моделирования таких систем на примере электронной энциклопедии кафедры “Информационная безопасность” МГТУ им. Н.Э. Баумана. Рассмотрены основные аспекты и обоснован выбор методов модельного представления системы с использованием аппарата сетей Петри.

E-mail: genrih.gertz@gmail.com

Ключевые слова: *информационные технологии, программные средства, электронный учебник, система, обучающая система, модель, сети Петри.*

Проектирование и разработка программных средств, сопровождающих учебный процесс, в значительной степени все больше осложняются возрастанием требований, предъявляемых к ним.

Сопровождение учебного процесса техническими и вычислительными средствами обеспечивает достижение двух стратегических целей. Первая из них заключается в повышении эффективности всех видов образовательной деятельности на основе использования информационных и телекоммуникационных технологий. Вторая — в улучшении качества подготовки специалистов с новым типом мышления, соответствующим требованиям информационного общества.

Традиционно средства информационных технологий в системе профессиональной подготовки студентов должны обладать такими возможностями, как:

- совершенствование методов и технологий отбора и формирования содержания образования;
- введение и развитие новых специализированных учебных дисциплин и направлений обучения;
- внесение изменений в обучение большинству традиционных дисциплин;
- совершенствование механизмов управления системой образования.

Как показывает анализ наиболее распространенных электронных средств обучения [1 — 6], их можно условно классифицировать следующим образом:

сервисные программные средства общего назначения применяются для автоматизации рутинных вычислений, оформления учебной документации, обработки данных экспериментальных исследований;

программные средства для математического и имитационного моделирования позволяют расширить границы экспериментальных и теоретических исследований, дополнить физический эксперимент вычислительным. К моделирующим программным средствам можно также отнести предметно-ориентированные программные среды, обеспечивающие возможность оперирования моделями-объектами конкретных классов;

информационно-поисковые справочные программные системы предназначены для ввода, хранения и предоставления преподавателям и обучающимся разнообразной информации. К числу подобных систем могут быть отнесены различные гипертекстовые и гипермедиа программы с иерархической организацией материала и быстрым поиском материалов по тем или иным признакам;

автоматизированные обучающие системы (АОС) представляют собой обучающие программы сравнительно небольшого объема, с помощью которых осуществляется ознакомление учащихся с теоретическим материалом, выработка навыков и умений, а также контроль уровня знаний;

электронные учебники (ЭУ) в настоящее время являются основными электронными средствами обучения.

Анализ известных [[1 — 6] электронных средств обучения показывает также, что при их разработке и реализации не учитываются некоторые специфичные требования информационной безопасности, в последнее время все чаще предъявляемые к этим средствам:

- предотвращение неавторизованного доступа;
- ограничение несанкционированного доступа к служебным ресурсам и программному коду;
- запрет полного или частичного копирования СУБД;
- предотвращение копирования злоумышленником текста стандартными средствами;
- предотвращение сохранения злоумышленником страниц электронного учебного пособия на жесткий диск или съемный носитель информации.

Кроме того, при проектировании электронных средств обучения зачастую необходимо предусматривать разработку программных средств автоматизации процесса формирования учебных планов в сочетании с созданием подсистемы защиты от несанкционированного доступа, что является противоречивой задачей.

Следует учитывать, что физические и биометрические методы идентификации и аутентификации контроля доступа применительно

к электронным средствам обучения нерентабельны. Поэтому применение методов программной инженерии при разработке указанных средств становится актуальным.

Анализ известных электронных средств сопровождения учебного процесса. В результате проведения обзора по различным источникам для анализа выбраны вычислительные среды, наиболее распространенные среди разработчиков.

ОРОКС (прежнее название WEB-Tester) создан Московским областным центром новых информационных технологий, МИЭТ, [1]. Данный программный комплекс является многофункциональной сетевой оболочкой для создания учебно-методических модулей и организации обучения с удаленным доступом. Она реализована с использованием WWW CGI-технологии. ОРОКС позволяет строить различные учебные модули в пошаговом режиме, использовать уже готовые компоненты, а также подготавливать материалы для записи на компакт-диски. Специальные средства информационной безопасности в среде и в продуктах, создаваемых в этой среде, не предусмотрены.

Система дистанционного обучения (СДО) Прометей (разработка центра “Виртуальные технологии в образовании”) [2] — программная оболочка, обеспечивающая возможности дистанционного обучения, тестирования слушателей, управления деятельностью виртуального учебного заведения.

Система Прометей имеет модульную архитектуру, поэтому легко может быть расширена, модернизирована и масштабирована. Так же, как и в программном комплексе ОРОКС, специальные средства информационной безопасности в среде и в продуктах, создаваемых в этой среде, не предусмотрены.

Система Learning Space(Lotus)[3] представляет собой распределенную среду поддержки процесса обучения, обеспечивающую создание интерактивных курсов; публикацию этих курсов; проведение учебного процесса в синхронном и асинхронном режиме, а также в режиме самообучения.

подавляющее большинство учебных курсов, развернутых с помощью Learning Space, относится к гуманитарным дисциплинам. В публикациях отмечается, что удобство работы с учебными курсами на практике оказывается недопустимо низким: время ожидания загрузки одной страницы может составлять несколько минут даже при использовании широкополосных линий связи.

На основе Lotus реализуются такие типовые системы, как:

- система регистрации слушателей;
- система управления учебным процессом, предназначенная для учета слушателей и их успеваемости в процессе дистанционного обучения;
- система создания и публикации учебных курсов.

Система Learning Space позволяет формировать задания, состоящие из нескольких случайно выбираемых вопросов по определенной схеме. Вопросы для заданий хранятся в соответствующей базе данных. Ответы могут проверяться автоматически или преподавателем. На основании полученных ответов система генерирует подробный отчет.

Специальные средства информационной безопасности в системе не предусмотрены.

Система Learning Space 5.0 (Lotus/IBM) [5] — программная обучающая среда, объединяющая возможности классического обучения с современными информационными технологиями. Learning Space 5.0 дает возможность обучаться и преподавать как в асинхронном режиме (обращаясь к материалам курсов в удобное время), так и участвовать в on-line занятиях (в режиме реального времени). Пользователь может формировать содержание курса в любых приложениях и затем размещать созданный материал в Learning Space 5.0. Специальные средства информационной безопасности не предусмотрены.

Система Blackboard Learning группы компаний VPGROUP и Blackboard, Inc. [6] представляет собой единую интерактивную среду для обучения, взаимодействия, а также обмена информацией между студентами, преподавателями и инструкторами вуза. Система позволяет управлять виртуальной обучающей средой, создавать электронные образовательные ресурсы, обеспечивать удаленный доступ к образовательным ресурсам учебного заведения, осуществлять контроль образовательного процесса, предоставлять платформы для курсов дистанционного обучения, накапливать, структурировать, управлять доступом, пополнять образовательную базу, а также предоставлять средства коммуникации и информирования участников.

Однако данная система не отвечает растущим требованиям информационной безопасности. Межсайтовый скриптинг в BlackBoard Learning System, выполненный Security Lab, выявил уязвимые версии Black Board Learning System 6 и Blackboard Learning and Community Portal Systems 6. Установлено, что уязвимость позволяет удаленному пользователю произвести XSS-нападение [7]. В нескольких публикациях [8, 9] сообщается, что девятилетний житель штата Вирджиния (США) взломал систему дистанционного обучения, известную под названием Blackboard Learning System. Поставив свои коды доступа, хакер отредактировал учебную программу: изменил списки учащихся и перекрыл доступ в систему для учителей.

При дальнейшем развитии рассмотренных систем их следует доработать, в частности, в целях обеспечения информационной безопасности. Причем намного больше затрат потребуется на доработку, чем для

учета необходимых аспектов безопасности при систематизированном проектировании.

Цель и задачи разработки. Предметом исследований и разработки является создание электронной системы обучения (электронной энциклопедии), предназначенной для автоматизированного планирования и сопровождения учебного процесса, который организует кафедра “Информационная безопасность”. Система представляет собой систематизированную совокупность графической, текстовой, цифровой, речевой, музыкальной, видео-и фотоинформации.

При постановке задач разработки электронной энциклопедии предусмотрено предоставление доступа к справочным и иным источникам информации. Необходимо также реализовать возможность использования инструментов создания и обработки методических и справочных материалов, сопровождающих учебный процесс, в частности, формирования студентом собственного конспекта по какой-либо дисциплине по личному запросу. С помощью программного обеспечения должны осуществляться построение понятийного графа, устанавливающего семантическую взаимосвязь дисциплин, расчеты требуемого рабочего времени студентов и преподавателей, автоматизированное определение загрузки аудиторного фонда и технических средств обучения.

Подсистема защиты должна устранять противоречия между требованием предоставления доступа широкому кругу пользователей (до 500 студентов) и требованием защиты от несанкционированного копирования материалов. Таким образом, ставились задачи проведения исследований и разработки системы обучения, свободной от недостатков известных систем и учитывающей специфику кафедры. Для решения поставленных задач потребовалось использовать специальные методы программной инженерии, рассмотренные далее.

В процессе экспериментальной эксплуатации разрабатываемой электронной системы обучения сложилась и реализуется новая концепция организационного аспекта, которая заключается в следующем:

— перманентно изменять содержание и “оболочку” продукта с учетом развития потребностей учебного процесса и динамики модификации в содержательной части большинства дисциплин, преподаваемых на кафедре;

— непрерывно обеспечивать сменяемость состава редколлегии с учетом необходимости выполнения требований, указанных выше;

— вносимые изменения не должны создавать затруднения доступу в течение всего учебного года и нарушать условия информационной безопасности.

Кроме того, сложилась организационная структура, благодаря которой электронная энциклопедия существенно отличается от всех известных электронных средств сопровождения учебного процесса:

коллектив технических редакторов составляют студенты 3–6-го курсов, постоянно работают 10 человек. По мере того, как оканчивают Университет студенты-дипломники, указанный коллектив меняется. Для пополнения и перманентного обновления редакторов привлекаются студенты младших курсов – это “стажеры” числом до 20 человек;

научно-методическое сопровождение и доработки по существу выполняют 25 преподавателей кафедры;

студенты кафедры, не входящие в состав авторского коллектива, по желанию предоставляют рефераты, в которых излагают предложения по внесению поправок в материалы энциклопедии.

Существенное отличие электронной энциклопедии от большинства аналогичных средств заключается в том, что она не может быть редуцирована к бумажному варианту без потери дидактических свойств.

Анализ промежуточных результатов опытной эксплуатации позволил поставить задачу разработки модели системы, обеспечивающей сопровождение процесса развития системы и учет возникающих проблем для нахождения удовлетворительного проектного решения, принимая во внимание при этом требования информационной безопасности. Выбранный метод модельного представления системы рассматривается далее.

Выбор метода решения задач проектирования. Исследования, проведенные в процессе опытной эксплуатации энциклопедии показали, что в качестве инструмента программной инженерии целесообразно применять аппарат сетей Петри.

Формальное определение сети Петри. Существует несколько формальных определений сети Петри, отличающихся способами задания элементов и связей в сети. В данной работе приняты основные положения, изложенные в [10]. Одно из важнейших свойств сети Петри – безопасность. Сеть Петри безопасна, если безопасны все позиции сети.

Принимаем, что сеть Петри C формально описывается четверткой (тетрадно)

$$\langle C = (P, T, I, O) \rangle,$$

где P – непустое множество элементов сети, называемых местами и помечаемых фишками; T – непустое множество элементов сети, называемых переходами; I – функция инцидентности, задающая связи между элементами множеств P и T ; O – функция выходов. Также считаем, что позиция pi , принадлежащая множеству P с начальной маркировкой m является безопасной, если $m'(pi) < 1$ для любого значения m' .

Если интерпретировать сеть как условия и события, маркировка каждой позиции должна быть безопасной. В безопасной сети любой

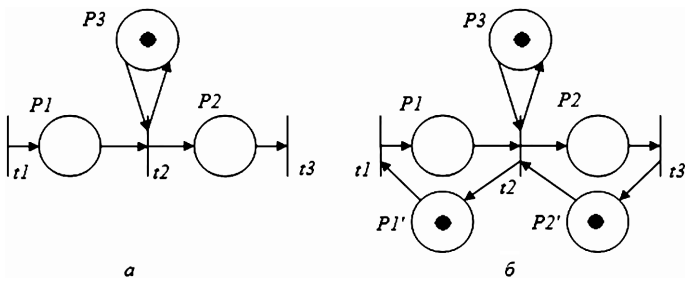


Рис. 1. Сеть Петри, не являющаяся безопасной (а), и безопасная сеть Петри (б), эквивалентная небезопасной сети (а)

переход, удаляющий фишку из pi , должен помещать фишку в позицию pi' , а всякий переход, удаляющий фишку из pi' , должен помещать фишку в pi . Начальная маркировка также должна быть модифицирована для обеспечения того, чтобы только одна фишка была либо в позиции pi , либо в pi' . Такая принудительная безопасность возможна лишь для позиций, которые в начальной маркировке являются безопасными: их входная и выходная кратность равна 0 или 1 для всех переходов. Позиция, имеющая для некоторого перехода выходную кратность 2, будет получать при его запуске две фишки и, следовательно, не может быть безопасной. На рис. 1 простая сеть Петри (а) преобразована в безопасную (б).

Установлено [11–13], что сети Петри можно использовать для моделирования систем распределения ресурсов. Для их сохранности фишки, представляющие ресурсы, не должны создаваться или уничтожаться: общее число фишек в сети должно оставаться постоянным.

Принимаем, что сеть Петри с начальной маркировкой m называется строго сохраняющей, если в ней число входов в каждый переход соответствует числу выходов $I(t_j) = O(t_j)$. Если это условие не выполняется, то запуск перехода изменяет число фишек в сети.

В сети Петри (рис. 2), которая не является строго сохраняющей, при запуске перехода $t1$ или $t2$ уменьшается число фишек на 1, а при запуске переходов $t3$ или $t4$ добавляется фишка к маркировке. Данную сеть можно преобразовать в строго сохраняющую (рис. 3).

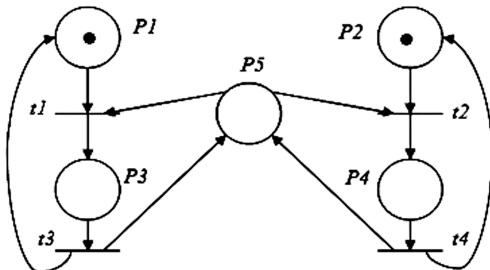


Рис. 2. Сеть Петри, не являющаяся строго сохраняющей

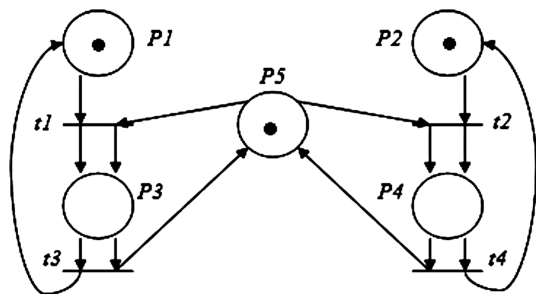


Рис. 3. Строго сохраняющая сеть Петри

Взвешенная сумма для всех маркировок должна быть постоянной. Для определения взвешенной суммы фишкам, не являющимся важными, присваивается вес 0, а другим фишкам — веса 1, 2, 3 или другое целое число. Так как каждая фишка характеризуется ее позицией в сети, веса связаны с каждой позицией сети. Вектор взвешивания $w = (w_1, w_2, \dots, w_n)$ определяется весами w_i для каждой позиции p_i , принадлежащей множеству P .

Рассмотрим систему, включающую два различных ресурса q и r и два процесса. Если оба процесса нуждаются в обоих ресурсах, ресурсы необходимо использовать совместно. Для выполнения этого требуется, чтобы каждый процесс запрашивал ресурс, а затем освобождал его. Предположим, что один из процессов (рис. 4, а) вначале запрашивает ресурс q , затем ресурс r и, наконец, освобождает q и r . Второй процесс (рис. 4, б) работает аналогично, но сначала запрашивает ресурс r , а затем q .

Начальная маркировка помечает ресурсы $q(P4)$ и $r(P5)$ как доступные и указывает на готовность процессов a и b . Одним выполнением этой сети является последовательность $t_1t_2t_3t_4t_5t_6$, а другим — $t_4t_5t_6t_1t_2t_3$. Ни одно из этих выполнений не приводит к тупику.

Рассмотрим последовательность, которая начинается переходами t_1 и t_4 : процесс a обладает ресурсом q , чтобы получить r , и в то же время процесс b обладает ресурсом r , чтобы получить q . Система

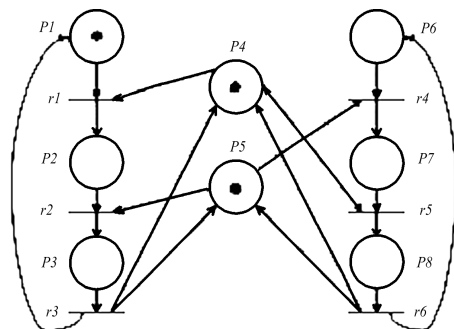


Рис. 4. Распределение ресурсов q и r для двух процессов

заблокирована, т. е. никакой процесс продолжаться не может — это тупик.

Пример использования сети Петри. Рассмотрим моделирование фрагмента ситуации, которая весьма характерна для организации доступа в многопользовательской системе, на основе задачи “о пяти философах” [5]. Фактор нечетности числа пользователей и четности числа ресурсов является здесь существенным. Поэтому в приводимом примере четное число одновременно востребованных ресурсов и пять пользователей. Далее покажем, что сетевая модель (например электронная энциклопедия) расширяется на большее число пользователей.

Пять пользователей пытаются сформировать собственный конспект из двух разделов электронной энциклопедии. Пусть для редактирования своих материалов каждому пользователю одновременно необходимы два раздела. Один из пользователей входит в энциклопедию, делает два запроса и начинает работать. При этом доступ к ресурсам энциклопедии освобождается полностью. В процессе подготовки будущего конспекта пользователю следует повторить запросы. Требуется построить такую модель, которая позволяла бы синхронизировать независимые действия пользователей и не допускала бы взаимной блокировки, когда все пользователи входят в энциклопедию, каждый из них может вызвать по два раздела, и никто не может начать работать со всеми ресурсами.

Набор действий всех пользователей можно представить матрицей M размером 5×7 , в которой i -я строка $M[i, j]$ соответствует действиям i -го пользователя. Каждый из пользователей выполняет семь действий: $M[i, 1]$ — i -й пользователь входит в энциклопедию; $M[i, 2]$ — i -й пользователь запрашивает первый раздел; $PH[i, 3]$ — i -й пользователь запрашивает второй раздел; $PH[i, 4]$ — i -й пользователь работает с разделами; $PH[i, 5]$ — i -й пользователь освобождает доступ к первому разделу; $PH[i, 6]$ — i -й пользователь освобождает доступ ко второму разделу; $PH[i, 7]$ — i -й пользователь выходит из энциклопедии.

Поведение каждого пользователя моделирует сеть Петри (рис. 5). Каждому действию i -го пользователя ставится в соответствие переход в сетевой модели, который имеет тот же номер, что и строка в матрице действий. Предполагается, что действия, связанные с разделами, пользователь может осуществлять независимо и условно одновременно. Одновременность означает, что два ресурса на время запроса доступны одному пользователю. В модели факт условной параллельности действий отображается параллельными ветвями сети: переходы с номерами $(i, 2)$ и $(i, 3)$, а также $(i, 5)$ и $(i, 6)$ могут сработать независимо и одновременно.

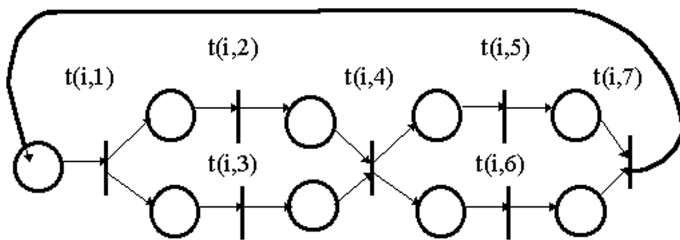


Рис. 5. Сетевая модель действий одного пользователя

Для того чтобы синхронизировать независимые действия пользователей, в модель необходимо ввести элементы, регламентирующие использование ресурсов. В качестве таких элементов могут служить дополнительные места с единичной маркировкой. Единственный маркер разрешает срабатывание только одного из разрешенных переходов, если два пользователя одновременно претендуют на один и тот же раздел. Так исключается возможность одновременного использования одного и того же ресурса.

Очевидно, что решением проблемы обеспечения отсутствия блокировок может стать запрет на одновременное присутствие более четырех пользователей. В модели это реализуется введением еще одного дополнительного места, число маркеров в котором равно четырем. Данное место связано входными дугами с первым переходом модели действий каждого из пяти пользователей. Вход пользователя в энциклопедию отображается в модели появлением маркера во входном месте перехода $t(i, 1)$. Если в дополнительном месте имеется маркер, то доступ разрешен, в противном случае пользователь будет ждать, пока один из четырех уже работающих в среде пользователей не освободит ему доступ. Результирующая сеть показана на рис. 6.

При моделировании разветвленных систем возможно использование методов расширения, которые позволяют экстраполировать результаты построения модели частной задачи. Для решения таких задач полезно применение структурированных сетей: в них некоторые из переходов являются сложными. Если срабатывает сеть данного уровня, то запускается сеть другого уровня иерархии (рис. 7). Срабатывание перехода t_2 приводит к запуску сети другого уровня. После выполнения сети в ее выходной позиции по является фишка, затем формируются фишки в выходных позициях сложного перехода. Поскольку такие сети используются для моделирования модульных вычислительных систем, они полезны для решения задач, поставленных в работе.

Учет факторов сопровождения постоянно развиваемой системы, каковой является энциклопедия, с использованием сетей Петри возможен, так как кратность ребер не является постоянной в сетях с изменяемой структурой. В самомодифицируемых сетях кратность ре-

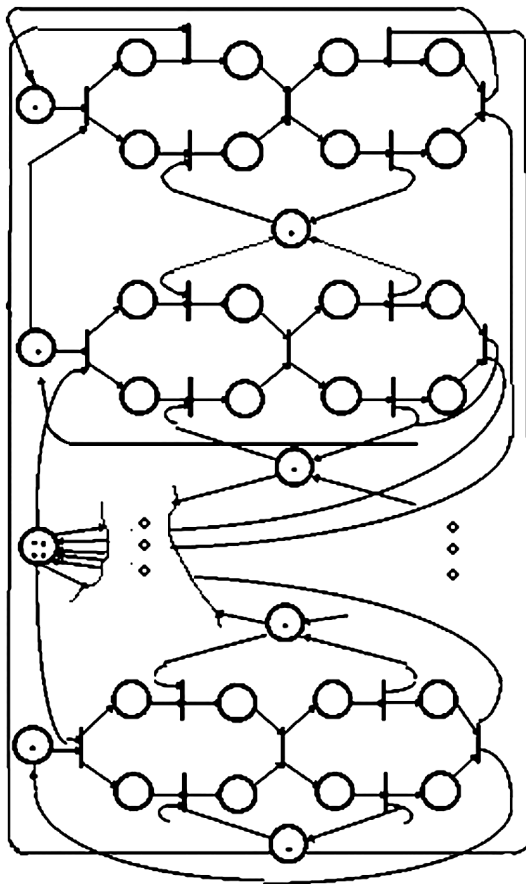


Рис. 6. Сетевая модель действий пяти пользователей

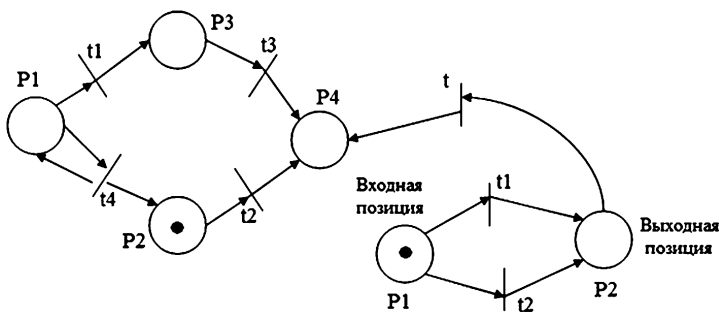


Рис. 7. Структурированная сеть Петри [11]

бер либо задается натуральным числом N , либо определяется числом фишек, находящихся во входных позициях некоторого перехода.

Качественными характеристиками могут быть отсутствие заикливания в системе, достижение некоторого состояния системы (в частности, конечного). Количественными характеристиками являются время работы некоторого маршрута в программе, время прохождения сигнала в схеме и т. д.

Для моделирования отдельных ситуаций, которые могут возникать при эксплуатации системы, применимы стохастические сети, в которых указанные характеристики являются вероятностными, т. е. вводится функция плотности вероятности срабатывания переходов или моментов нахождения фишек в позициях.

Для программной реализации модели можно представлять сети Петри в виде двудольного мультиграфа. Его структура состоит из двух заголовочных списков — списка переходов и списка позиций. Каждый элемент списка позиций — это запись, содержащая семь полей: x и y позиции (ее центра); поле с именем (номером) позиции; число фишек, принадлежащее данной позиции; число фишек до запуска перехода; поле, содержащее указатель на список переходов, в которые направлены дуги из данной позиции; поле с указателем на следующий элемент списка позиций.

Каждый элемент списка переходов включает в себя x и y перехода (его центра), а также поля: имени (номера) перехода, приоритета перехода, задержки перехода, указателя на список позиций, “исходящих” из данного перехода; указателя на список позиций, “входящих” в данный переход; указателя на следующий элемент списка переходов.

С помощью “дуговых” списков, заявленных в описании, осуществляется связь между позициями и переходами сети Петри. Там же хранится информация о числе дуг, связывающих данный переход с данной позицией или данную позицию с данным переходом. Каждый элемент “дугового” списка переходов, “подвешенного” к какому-то элементу списка позиций, содержит три поля: указатель на “исходящий” переход в “заголовочном” списке; кратность (число) дуг, поступающих в него из позиции, к которой подвешен “дуговой” список; указатель на следующий элемент списка.

К каждому элементу “заголовочного” списка переходов “подвешено” два почти одинаковых “дуговых” списка, их структура совершенно идентична. Они содержат поле указателя на “входящую”/“исходящую” позицию для перехода, к которому подвешен “дуговой” список; кратность (число) таких дуг; указатель на следующий элемент списка. Описания рассмотренных списков приведены ниже:

Type

```
Lref=^Leader; {указатель на заголовочный узел позиции}  
Mref=^Murder; {указатель на заголовочный узел перехода}  
Tref=^Trailer; {указатель на дуговой узел позиции}  
Uref=^User; {указатель на дуговой узел перехода}  
Leader=Record{описание типа заголовочного узла позиции}
```

```

Name:integer; {номер позиции}
Fish:integer; {количество фишек}
Fish_after:integer; {до запуска перехода}
x:integer;
y:integer;
Trail:Tref; {указатель на список смежности}
Next:Lref; {указатель на следующий узел в списке заголовочных
узлов}
end;
Trailer=Record{описание типа дугового узла позиции}
chi:integer; {кратность (количество дуг в переход из позиции) }
Id:Mref; {указатель на переход из позиции}
Next:Tref; {указатель на следующий узел в списке дуговых узлов}
end;
Murder=Record{тип заголовочного узла перехода}
Name: integer; {номер перехода}
x:integer;
y:integer;
Priority:integer; {приоритет перехода}
Delay:integer; {задержка перехода}
Stoppage:integer; {рабочее поле}
Permit:boolean; {разрешен ли переход к запуску}
Trail: Uref; {указатель на список последующих позиций}
Pred:Uref; {указатель на список предшествующих позиций}
Next:Mref; {указатель на следующий узел в списке заголовочных
узлов}
end;
User=Record{тип дугового узла перехода}
chi:integer; {количество дуг в позицию из перехода}
Id: Lref; {указатель на позицию, исходящую из перехода}
Next: Uref; {указатель на следующий узел в списке дуговых узлов}
end;

```

Одна часть полученной структуры подобна классической структуре Вирта, а другая — модифицированной. Таким образом, выбор метода моделирования системы можно признать обоснованным и с точки зрения возможности ее программной реализации.

СПИСОК ЛИТЕРАТУРЫ

1. Электронные средства обучения. <http://www.mocnit.zgrad.su>
2. Электронные средства обучения. www.prometeus.ru

3. Электронные средства обучения. <http://platform-tic.blogspot.com/>
4. Э л е к т р о н н ы е средства обучения. <http://www.moluch.ru/archive/27/2905/>
5. Электронные дистанционные курсы компании “WebSoft”. csbi-zirvan.ru/Нservice/acc/
6. Электронные средства обучения. ido.rudn.ru/Нnfpk/ikt/ikt5.html
7. www.velvet.by
8. www.xakep.ru/post/17451/default.asp
9. www.securitylab.ru/vulnerability/272407.php
10. www.smolensk.ru/user/sgma/MMORPH/N-4-html/6.htm
11. it.kgsu.ru/SetiPetri/sp017.html
12. П и т е р с о н. Теория сетей Петри и моделирование систем: Пер. с англ. – М.:Мир, 1984. – 264 с.
13. К о т о в В. Е. Сети Петри. – М.:Наука, 1984. – 160 с.

Статья поступила в редакцию 10.05.2012