

ТЕОРИИ, МОДЕЛИ И ТЕХНОЛОГИИ ОБУЧЕНИЯ МАТЕМАТИКЕ И ИНФОРМАТИКЕ В СИСТЕМЕ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

DOI: 10.24888/2500-1957-2023-4-29-45

УДК
378.14**ИНСТРУМЕНТЫ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ
В УЧЕБНОМ ПРОЦЕССЕ ТЕХНИЧЕСКОГО УНИВЕРСИТЕТА****Бескин Андрей Леонидович**к.т.н., доцент
beskin@mirea.ru
г. МоскваМИРЭА – Российский технологический
университет**Музылев Николай Викторович**к.ф.-м.н., доцент
muzylev@mirea.ru
г. МоскваМИРЭА – Российский технологический
университет

Аннотация. Подготовка инженеров в России в настоящее время – одна из главных задач высшей школы. Их острая нехватка ощущается во всех сферах экономики, также изменились требования к компетенциям современного инженера. Происходящая технологическая революция – переход к Индустрии 4.0 – требует новых цифровых компетенций инженера, в том числе, свободного владения инструментами компьютерного моделирования. Этим определяется актуальность данного исследования. В статье излагается методика применения таких инструментов в учебном процессе подготовки студентов и аспирантов инженерных направлений в техническом университете, в том числе, в преподавании дисциплин из учебного плана, в проведении научно-исследовательской работы и в подготовке выпускных работ. В 2020-2023 годах был проведен эксперимент по применению шести инструментов компьютерного моделирования в рамках освоения учебных дисциплин: «Моделирование систем» и «Моделирование программных систем» на втором курсе направления подготовки бакалавров «Программная инженерия», на третьем курсе направления «Математическое моделирование и информатика» в дисциплине «Алгоритмы и теория сложности» и на третьем курсе подготовки аспирантов разных направлений. Эксперимент завершился успешно: процент отличных оценок в разных группах достиг 70-85%, многие проекты моделей были использованы в выпускных работах, аспиранты запланировали включить свои исследования в диссертации. В ходе эксперимента были использованы облачные и дистанционные технологии. Все результаты эксперимента сохранены в облачном архиве и доступны преподавателям и студентам с любого устройства, подключенного к интернету. В перспективе данная методика планируется к использованию в преподавании разных учебных дисциплин, но в большей степени – в курсовых, выпускных работах и диссертациях.

Ключевые слова: профессиональная подготовка, программная инженерия, дискретно-событийная модель, агентное моделирование, модельный эксперимент.

Введение

Опыт преподавания учебных дисциплин по инженерным направлениям позволил авторам прийти к выводу о полезности и даже необходимости использования имитационных моделей для выявления сущности объектов и процессов. Пример изучения операционных систем, параллельных приложений, отказоустойчивых и высокопроизводительных вычислительных систем доказывает этот тезис: проведение реальных экспериментов либо недоступно студентам в принципе, либо затрудняет выявление причинно-следственных связей между входными параметрами и результатами.

Модели дают выявить наиболее существенные свойства и зависимости в системах быстро и точно. Для создания таких моделей требуются подходящие инструменты. Их выбор для применения в конкретной учебной дисциплине важен для достижения целей обучения. Зачастую содержание классической учебной дисциплины по моделированию систем сводится к освоению сложного пакета моделирования и выполнению типовых практических заданий. Предлагается расширить выбор инструментов моделирования быстрыми в освоении и бесплатными программами, специализированными на изучении важных компонентов сложных систем. К примеру, алгоритмов работы планировщиков операционных систем или схем отказоустойчивых программно-аппаратных комплексов.

Легкие в освоении и бесплатные инструменты моделирования могут быть полезны при изучении различных учебных дисциплин в инженерных и ИТ направлениях подготовки, а также при выполнении курсовых работ, НИР, на практиках, при подготовке выпускных квалификационных работ.

Выбор средств моделирования

В настоящее время доступны десятки инструментов компьютерного моделирования, как универсальные (Matlab-Simulink, Anylogic, GPSS World, Arena, CPNTools/CPN IDE), так и специализированные (Anylogic, Flexsim, iThink, Boson NetSim, AGNES). Основной проблемой их применения в учебном процессе является высокая стоимость лицензий и достаточно долгий период их освоения студентами. Поэтому в учебных дисциплинах продолжительностью один семестр трудно применить более одного из перечисленных пакетов.

Одним из решений данной проблемы стало создание собственных специализированных программ моделирования, не требующих инсталляции и больших ресурсов компьютера, полностью бесплатных. Учитывая направленность подготовки студентов на кафедре – программная инженерия, студенты, участвующие в таких проектах, получают неоценимый практический опыт разработки и внедрения полезных приложений.

За последние 25 лет работы в МИРЭА было инициировано несколько десятков студенческих проектов по созданию подобных инструментов. Часть из них оказалась успешной, и до сих пор применяется в преподавании ИТ-дисциплин (Бескин, 2023).

Отметим плюсы и минусы применения самописных инструментов в учебном процессе, их конкурентные преимущества:

1. Простота освоения: от 10 минут до 1 часа.
 2. Компактность кода, низкие требования к мощности компьютера, портативность (не требуют инсталляции).
 3. Быстродействие: задания выполняются от нескольких секунд, до нескольких минут (при статистических исследованиях).
 4. Открытый код (некоторые хранятся в GitHub), возможность доработки.
- Недостатки большинства таких проектов (из-за нехватки времени у разработчиков):
1. Нет надежного авторского сопровождения, из-за чего затруднительно устранять остаточные ошибки в коде.
 2. Ограниченный функционал (не введены очевидные полезные функции).
 3. Некачественная программная документация (или ее отсутствие).
 4. Привязка к конкретному программному окружению: обычно требуется Windows,

библиотеки (QT, .Net).

Несмотря на определенные недостатки, следующие самописные программы моделирования (Бескин, 2023) до сих пор успешно применяются в учебном процессе:

1. Симуляторы сетей Петри: QPNet версий 1.2 и 1.3.2.
2. Имитатор планировщиков ОС: VMOS версии 1.8.
3. Эмулятор канальных процессоров мейнфреймов IBM версии NewLoch.
4. Построитель PGVMT временных диаграмм многопоточных планировщиков ОС версии ПГБ4.
5. Генератор хеш-функций и графический анализатор их качества версии OSPGen 2.2.

В рабочие программы таких дисциплин, как «Операционные системы», «Параллельное программирование», «Организация вычислительных процессов», «Высокопроизводительные вычисления» введены задания с использованием нескольких перечисленных программ. Для дисциплин, где главное содержание – собственно моделирование, например, «Моделирование программных систем», применяются все перечисленные самописные программы плюс более мощные пакеты, такие как (Anylogic PLE).

Применение инструментов моделирования

Выбор инструментов компьютерного моделирования зависит от направления подготовки: для будущих программистов и тестировщиков требуются специальные модели, основанные на коде программ (Бескин, 2015), средств их статического анализа (Бескин, 2016). Для проектирования больших и сложных дискретных систем последние года всё чаще используется парадигма мультиагентного моделирования (Weins, 2019). Для моделирования дискретно-непрерывных и крупномасштабных систем применяются три парадигмы имитационного моделирования — дискретно-событийная, системная динамика и агентный подход. В пакете Anylogic реализованы все три подхода, что дает возможность создавать гибридные модели. В учебном процессе в МИРЭА используется студенческая PLE версия пакета (Anylogic PLE).

Универсальные пакеты. На кафедре применялись лицензионные программы Anylogic, GPSS World, True и другие. Из самописных – эмуляторы сетей Петри, позволяющие создавать и исследовать детальные модели систем массового обслуживания ограниченной сложности.

Специализированные пакеты. Разработаны на кафедре МОСИТ (ранее МОВС) Института ИТ РТУ МИРЭА в рамках курсового проектирования и выпускных работ.

Высокоскоростные эмуляторы сетей Петри QPNet. Используются в учебном процессе как простые в освоении инструменты дискретно-событийного моделирования. Не требуют инсталляции, объем 14 Мбайт.

В мировом сообществе наиболее популярна программа моделирования CPNTools, которая в настоящее время трансформировалась в CPN IDE (CPNTools/CPN IDE). Несмотря на большую популярность программы, содержательные примеры и полное руководство труднодоступны. Так же, как пакет Anylogic, CPNTools использует встроенный в ОС Java движок. Поэтому быстроедействие этих систем недостаточно для проведения длительных испытаний с миллиардами событий. Для QPNet, написанном на профессиональном языке программирования с оптимизацией, нет подобных ограничений, возможны испытания с миллиардами событий на обычном офисном компьютере.

В МИРЭА тема сетей Петри и их эмуляторов актуальна с начала 90-х годов. Регулярно появляются научные статьи на эту тему (Кулагин, 2021; Кудж, 2015). Но многолетнее внедрение в учебный процесс получил пакет программ QPNet. Доступны в облаке как программы, так и подробное руководство по их применению с примерами (Бескин, 2023).

Решаемые задачи. Построение моделей, их статистическое исследование, оптимизация ресурсов и алгоритмов обслуживания заявок, определение показателей эффективности. За счет высокого быстрогодействия возможно испытание устойчивости систем к редким и крайне редким нештатным ситуациям.

Построитель временных диаграмм параллельных и многопоточных процессов ПГБ4. Применяется в учебном процессе при изучении управления процессами в операционных

ТЕОРИИ, МОДЕЛИ И ТЕХНОЛОГИИ ОБУЧЕНИЯ МАТЕМАТИКЕ И ИНФОРМАТИКЕ В СИСТЕМЕ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

системах, в том числе синхронизация процессов, управление ресурсами, взаимные блокировки. Сразу выдает готовую временную диаграмму или определяет тупиковую ситуацию.

Эмулятор планировщиков операционных систем VMOS. Визуализирует выполнение вычислений и операций обмена данными для нескольких параллельных процессов и каналов ввода-вывода. Имеет встроенный абстрактный язык описания процессов и систем процессов. Позволяет визуализировать тупики в ОС, имеет несколько вариантов настройки систем процессов для устранения тупика.

Эмулятор канальной системы ввода-вывода мейнфреймов NewLoch. Визуализирует процессы обмена данными в каналах и устройствах, потери данных при нехватке времени на обмен. Имеет систему настройки параметров каналов и устройств. Ведет протокол всех действий пользователя и журнал операций ввода-вывода.

Анализатор качества и скорости хеш-функций OSPGEN. Генерирует пять типов стандартных хеш-функций по заданным параметрам, определяет процент коллизий при обработке словарей в диапазоне 10-50000 слов, визуализирует результат в форме графиков или двумерных цветных диаграмм. Имеет встроенный генератор словарей из случайных слов с различными наборами допустимых символов и заданного диапазона длины слов.

Приведенные выше программные средства созданы студентами в ходе выполнения курсовых и выпускных работ. Все эти проекты прошли многолетнюю стадию разработки и модернизации, многие имели до 10 релизов. Апробация выполнялась в рамках различных учебных дисциплин. Необходимые изменения вносились авторами или создавался новый релиз другими разработчиками.

Плюсы и минусы собственных инструментов моделирования приведены выше. Для повышения качества собственных разработок необходима организация творческих коллективов, которые могут вести проект не один семестр в рамках конкретной учебной дисциплины, а несколько лет под руководством опытных преподавателей.

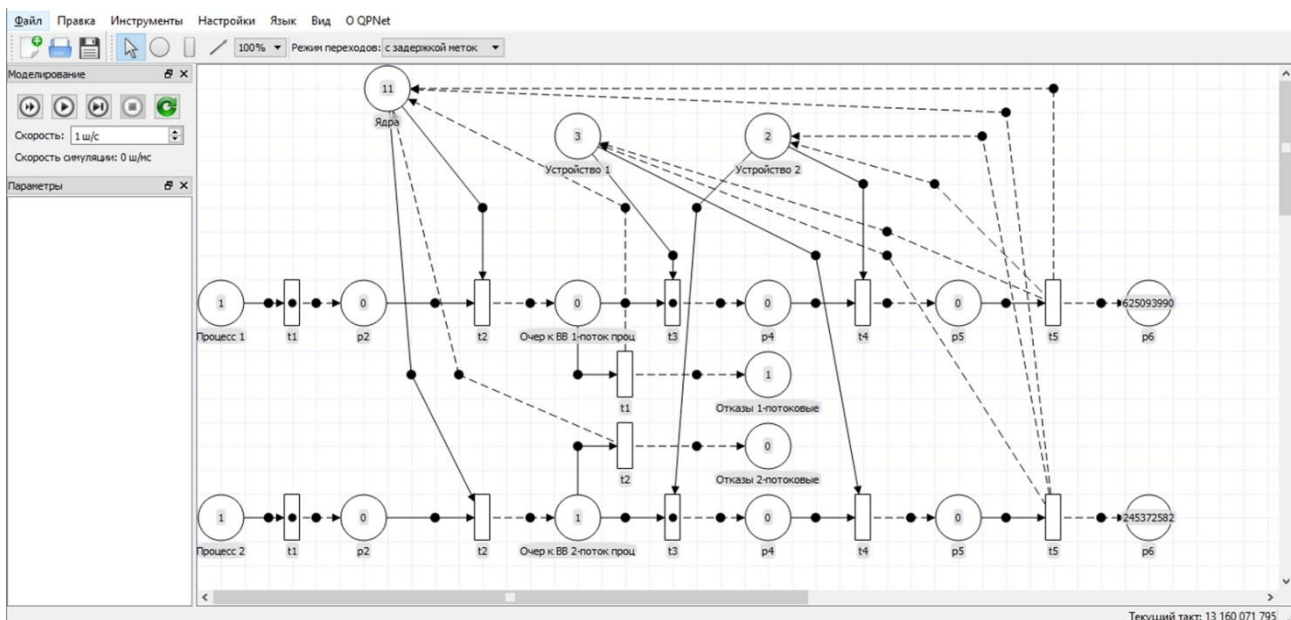


Рис. 1. QPNet 1.3.2: один отказ планировщика ОС на 870 млн процессов

Наиболее интересный проект QPNet прошел стадию развития с 90-х годов до 2014 года. Его эволюция отражена в сетевом ресурсе <https://sourceforge.net/projects/qpnet/>. Некоторые версии не нашли применения в учебном процессе из-за их излишней сложности, другие не были реализованы из-за организационных проблем (проект модульных сетей Петри с параллельными вычислениями на удаленном сервере, языком управления экспериментами и мощной встроенной статистикой).

Примеры.

П1. Планировщик реального времени (рис. 1). Две очереди процессов, 1-поточные и 2-поточные, частота второй заявки ниже в 2,5 раза первой. Ресурсы системы – вычислительные ядра (24), два типа устройств ввода-вывода (по 4 экз.), порядок захвата устройств процессами взаимно-обратный (из-за чего возможны тупики). 2-поточные процессы имеют приоритет при выделении ресурсов. Время моделирования 13 млрд системных тактов. Один отказ (т.е. длина очереди 10 или более) в очереди однопоточных процессов (прошло 625 млн). Нет отказов (длина очереди 6 или более) для 2-поточных (245 млн).

П2. Трехканальная память с мажоритарным клапаном (рис. 2). Основной параметр – частота сбоев в каналах (в примере установлено 50 тыс. тактов или секунд, что достаточно часто для высоконадежных компонентов). 3 варианта общего сбоя памяти: сбой сразу в двух каналах, 1-2, 1-3, и 2-3. Критерий работоспособности памяти – ноль общих сбоев за N лет. Скорость работы модели: около 10 секунд счета на 1 год работы памяти (для офисного компьютера типа Core i3, 4 млн событий в секунду).

Увеличение частоты сбоев в каналах в 5 раз приводит к отказу системы уже на 2-м году работы. Для исходной частоты сбоев в каналах время безотказной работы измеряется многими десятками лет, что не соответствует реальности, т.к. не учтена вероятность отказа каналов.

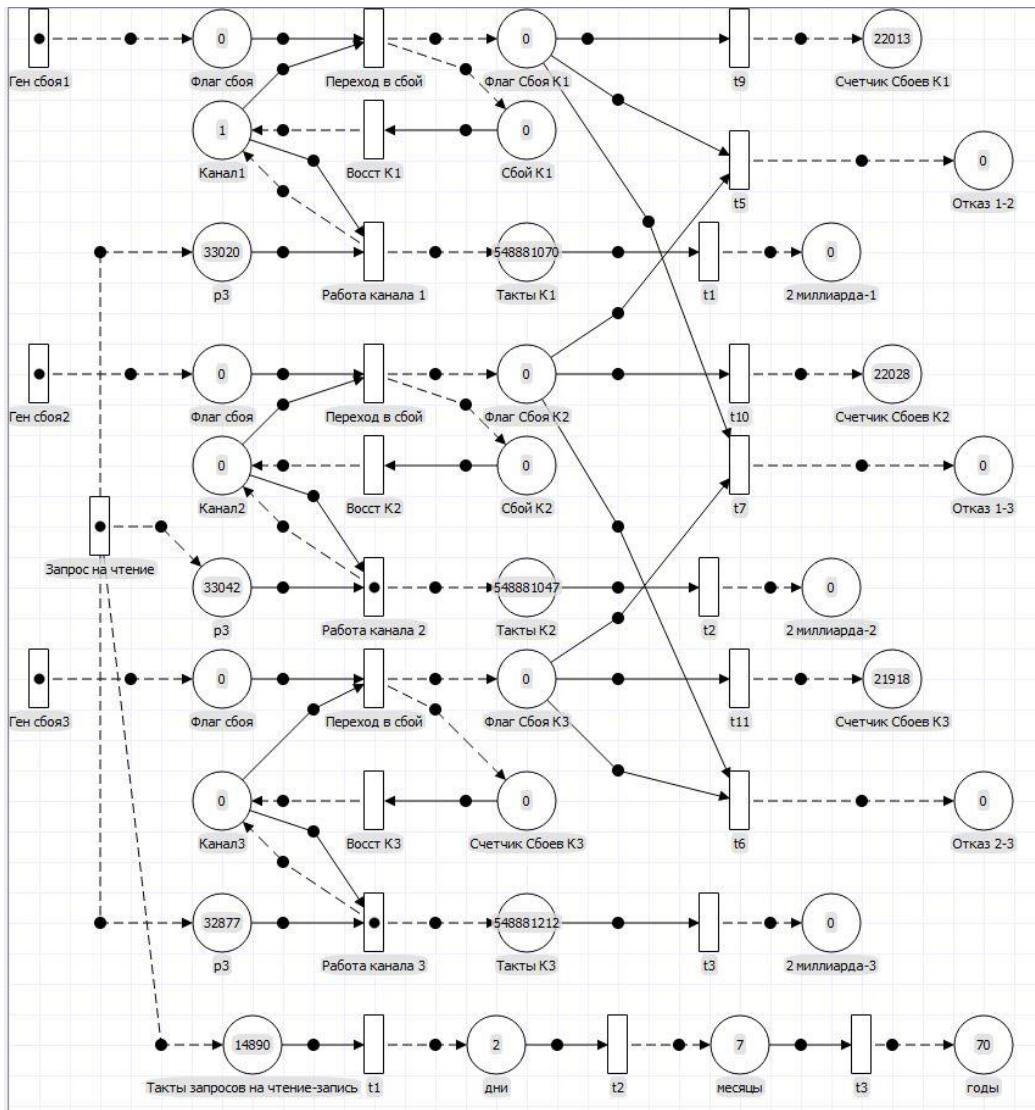


Рис. 2. QPNet 1.3.2: 70 лет безотказной работы 3-канальной памяти

ТЕОРИИ, МОДЕЛИ И ТЕХНОЛОГИИ ОБУЧЕНИЯ МАТЕМАТИКЕ
И ИНФОРМАТИКЕ В СИСТЕМЕ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

При отказе одного из каналов система становится неработоспособной. Решение этой проблемы будет рассмотрено ниже (5-канальная схема памяти).

Для моделирования отказоустойчивых вычислительных систем обычно разрабатываются проблемно-ориентированные программные средства, опирающиеся на платформу Java, которые неприменимы в других приложениях (Винс, 2016). В примерах П.2 и П.3 показано, что универсальный инструмент QPNet способен решать подобные задачи.

П3. Определение показателей эффективности по заданным критериям. Используется вторая составная часть пакета QPNet, автоматически выполняющая множество прогонов модели с их рестартом по достижению одного из заданных параметров. В качестве примера рассмотрим решение проблемы 3-канальной памяти, связанной с вероятным отказом каналов при длительной работе (рис. 2). Предлагается 5-канальная модель памяти с голосованием 4:1 и 3:2 при сбое в одном и двух каналах. Также вводится вероятность отказа каналов с заданной частотой.

Предложенная структура 5-канальной памяти устойчива к одномоментному сбою в двух разных каналах, а также к отказу 2 каналов из 5 (при этом сохраняется устойчивость к сбою в одном канале). Модель получилась значительно сложнее 3-канальной памяти, но скорость ее работы осталась приемлемой: около 1 млн тактов в секунду или 1 год за 40 секунд.

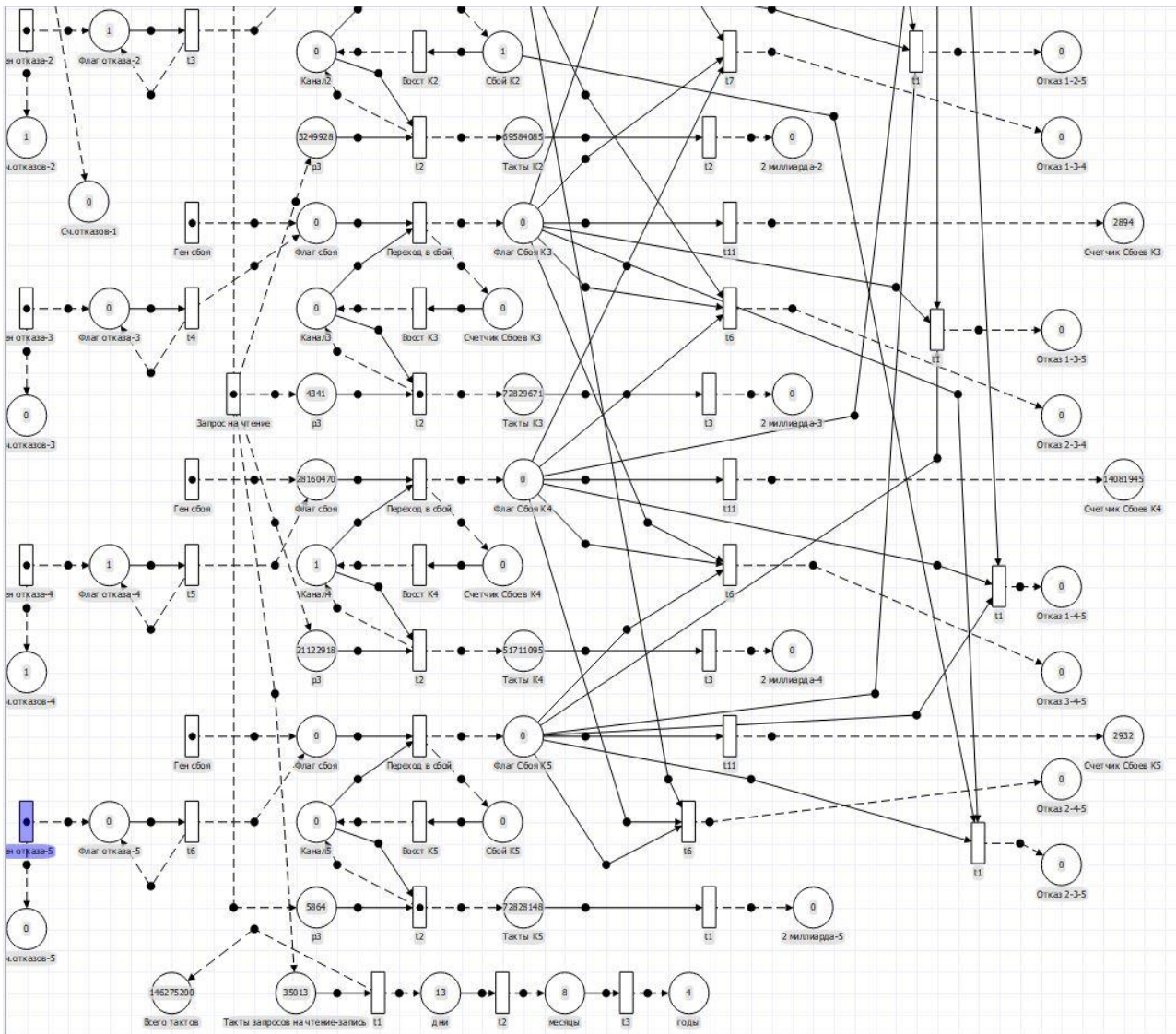


Рис. 3. Модель 5-канальной памяти. Ноль отказов за 4 года 8 месяцев

На рисунке 3 представлена основная часть модели 5-канальной памяти. Отказом считается одновременная ошибка в 3-х каналах (при отказе канала считается, что на его выходе – постоянная ошибка). Всего 10 вариантов отказов (тройные сбои в каналах): 1-2-3, 1-2-4, ..., 3-4-5. В данном примере среднее время наработки канала на отказ составляет 150 млн тактов или 4 года 8 месяцев. Частота сбоев в каналах 50 тыс. тактов, это чаще чем один раз в сутки. Увеличение этой частоты в 5 раз практически не влияет на работоспособность системы.

Полученный результат не дает необходимой оценки надежности системы, т.к. не определена вероятность того, что система проработает безотказно не менее N лет с заданной достоверностью. Для получения такой оценки необходимо провести серию экспериментов (от 5 до 10 достаточно), в каждом из которых определяется доля успешных прогонов модели (например, ноль отказов за 5 лет). Вторая часть пакета (QPNet, версия 1.2), выполняет проведение одного эксперимента, если задать критерии успеха (5 лет работы системы) и провала (число отказов системы больше нуля).

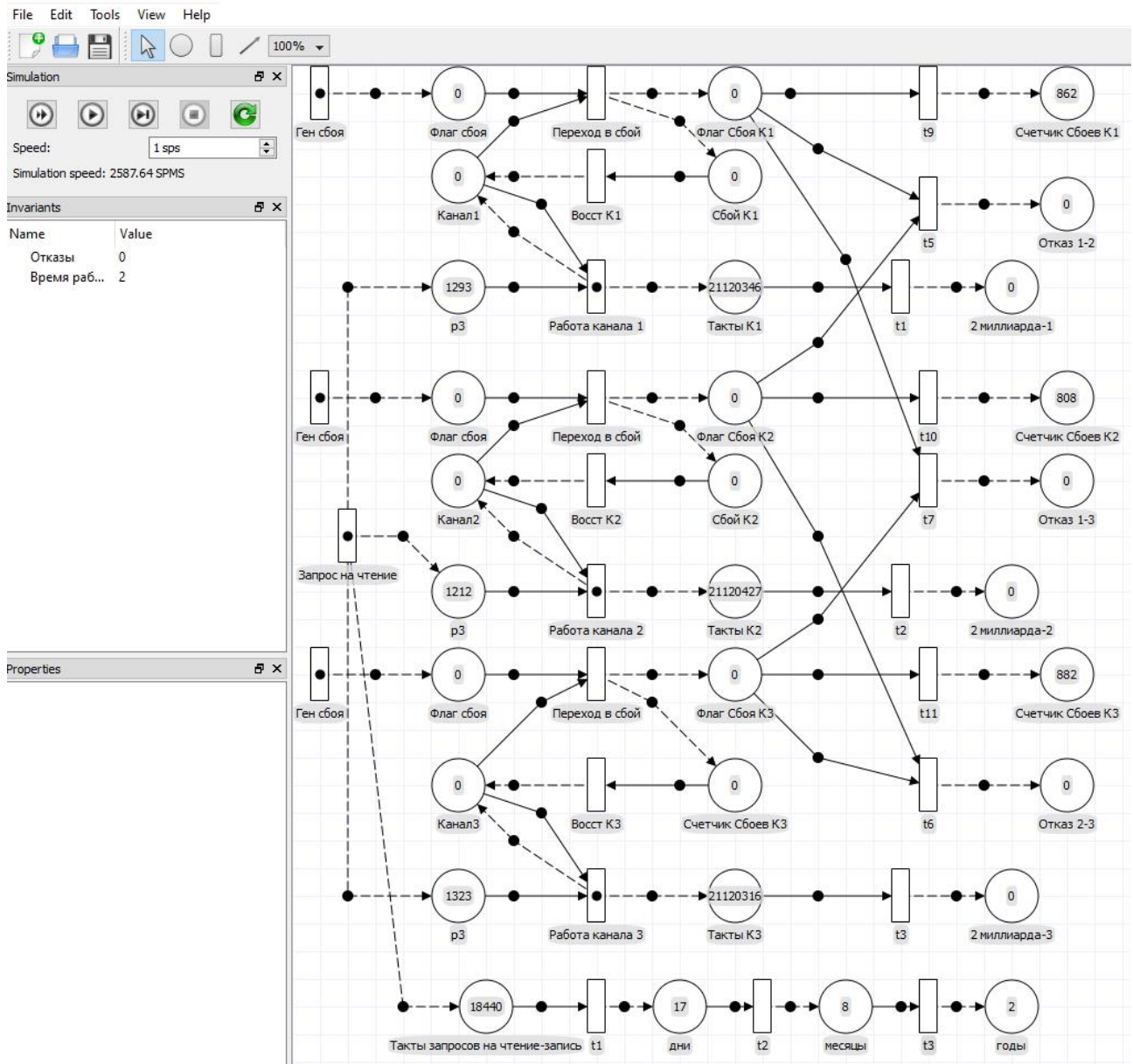


Рис.4. Эксперимент по определению успешности модели по заданным критериям

ТЕОРИИ, МОДЕЛИ И ТЕХНОЛОГИИ ОБУЧЕНИЯ МАТЕМАТИКЕ И ИНФОРМАТИКЕ В СИСТЕМЕ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

На рисунке 4 приведен пример такого эксперимента для 3-канальной памяти. Вероятность успеха составила более 89%. Для повышения точности оценки (что потребуется для моделирования сверхнадежных систем) необходимо проводить эксперимент намного дольше (ограничение на число прогонов составляет 2^{31}). Количество экспериментов и обработка их результатов будут рассмотрены в разделе «Технология моделирования».

П4. Моделирование параллельных процессов и многопоточных процессов в построителе временных диаграмм типичной операционной системе реального времени ПГБ4. Программа позволяет исследовать такие временные диаграммы и возможные тупиковые ситуации при распределении ресурсов ОС (S_1, S_2, \dots). Моделируется также процесс управления каналным вводом-выводом (рис. 5). Процессы описаны на абстрактном языке, в котором различаются только операторы счета, захвата и освобождения ресурсов, запроса канала и ожидания завершения обмена. При фиксации тупика предлагается метод выхода.

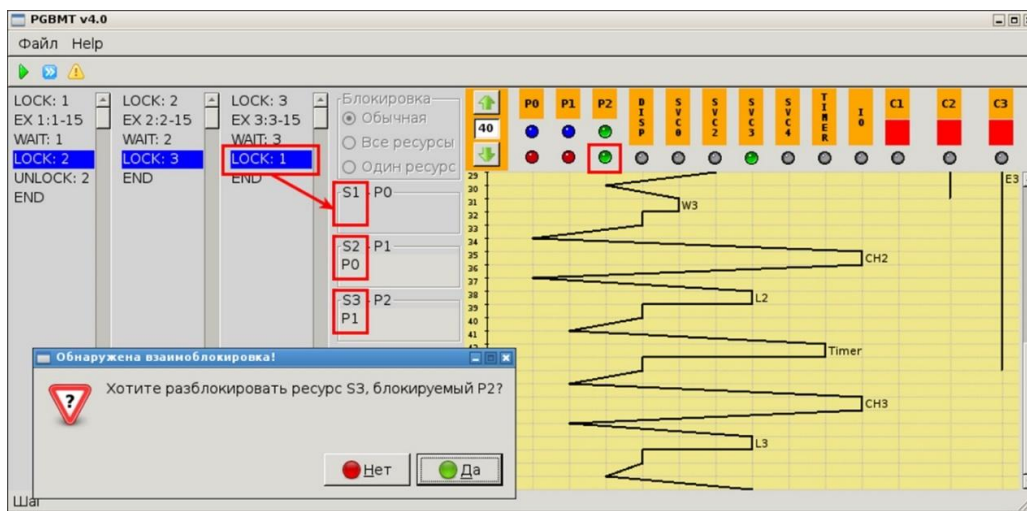


Рис. 5 Временная диаграмма ПГБ4. Ситуация взаимной блокировки процессов

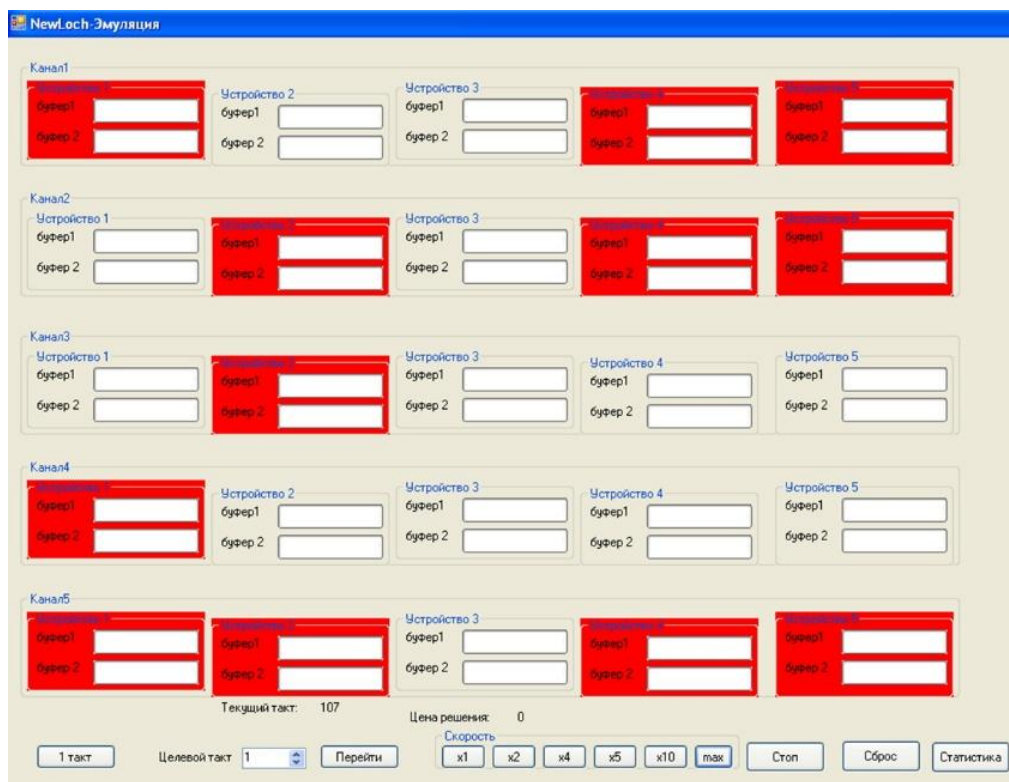


Рис. 6. Симуляция работы каналов и устройств ввода-вывода мейнфрейма

Также можно исследовать влияния частоты работы интервального таймера на временные диаграммы, а также различные режимы управления ресурсами (захват процессом не более одного ресурса, захват процессом сразу всех необходимых ресурсов и динамическое распределение ресурсов по запросам).

П5. Симулятор каналов ввода-вывода мейнфреймов NewLoch (рис. 6). Учитывая труднодоступность реальных компьютеров с канальной системой ввода-вывода, симулятор может дать представление об алгоритмах работы системы, методах настройки параметров на заданную нагрузку. Пользователь программы имеет возможность настройки параметров каналов и устройств ввода-вывода (пропускная способность, задержка, размер буфера).

Также есть возможность выбора типа канала: последовательный (селекторный) или параллельный (мультиплексный).

Условная стоимость изменения параметров системы исчисляется в баллах. При снижении требований сумма баллов уменьшается, иначе – увеличивается. Ставится задача при заданной нагрузке на систему ввода-вывода в реальном времени настроить ее так, чтобы не было потерь данных (красный цвет на рисунке 6) с минимальной суммой баллов.

П6. Симулятор многозадачных планировщиков операционных систем VMOS (рис. 7). Показывает в динамике выполнение параллельных процессов вычисления и ввода-вывода. Имеет панель настроек, позволяющую выбрать скорость симуляции, один из вариантов планировщика, загрузить отдельные процессы или целую систему процессов, изменить частоту интервального таймера. Задания описаны на абстрактном языке, включающем операторы счета, захвата и освобождения канала, обмена данными по каналу, изменения приоритета процесса

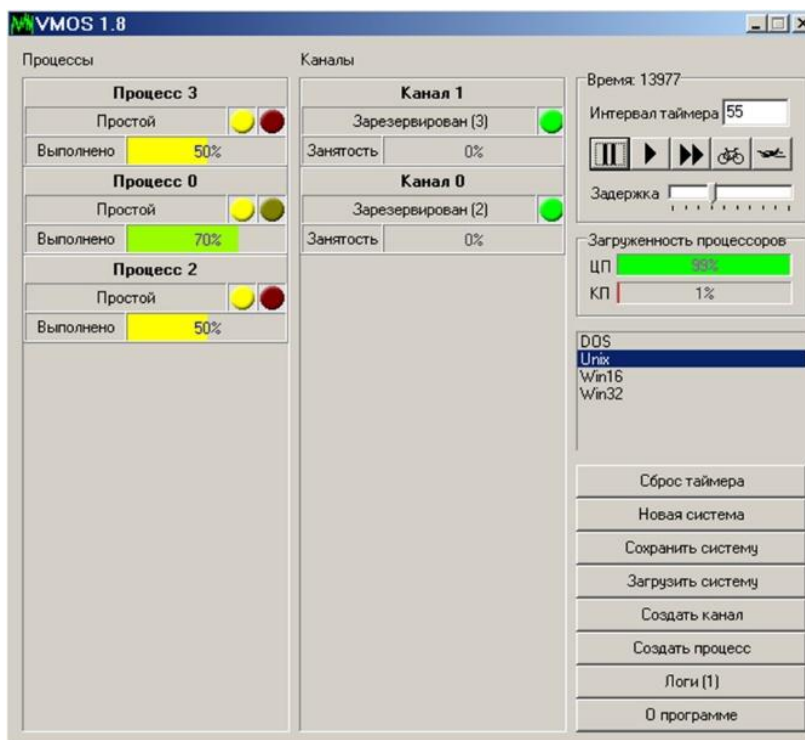


Рис. 7. Симулятор планировщиков операционных систем

Симулятор позволяет исследовать системы процессов в разных режимах, в том числе создавать тупиковые ситуации при управлении каналами, искать методы их разрешения. Незавершенная версия симулятора позволяет исследовать до 16 вариантов многозадачных и многопоточных планировщиков ОС.

П7. Анализатор качества и скорости хеш-функций OSPGEN 2. Помимо числовой оценки работы хеш-функций выводит графический образ, который может идентифицировать скрытые дефекты вычислителя и функции (рис. 8). В данном примере показано следствие ошибки целочисленного деления некоторых степпингов процессоров Интел выпуска 2003-

ТЕОРИИ, МОДЕЛИ И ТЕХНОЛОГИИ ОБУЧЕНИЯ МАТЕМАТИКЕ И ИНФОРМАТИКЕ В СИСТЕМЕ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

2004 гг. Большинство процессоров выполняли эту операцию без ошибки (вертикальная линия ровная), некоторые давали ошибку в последнем бите машинного слова (излом линии). Обнаружить данную ошибку другими средствами было крайне сложно, и подавляющее большинство владельцев дефектных процессоров не замечали никаких проблем с вычислениями.

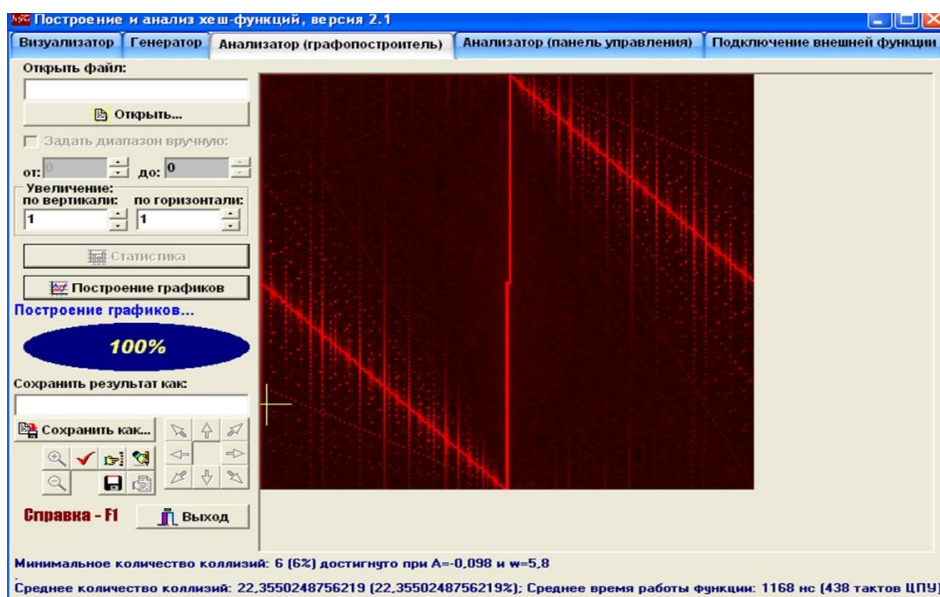


Рис.8. Образ мультипликативной хеш-функции с дефектом в центре

Программа рассчитана на генерацию и анализ стандартных простейших хеш-функций, главное достоинство которых – короткий ключ (8 или 16 бит) и высокое быстродействие. Область применения таких функций (помимо чисто учебных целей) – организация прямого доступа к небольшим массивам данных (несколько десятков записей для однобайтовой функции, сотни и тысячи записей – для двухбайтовой).

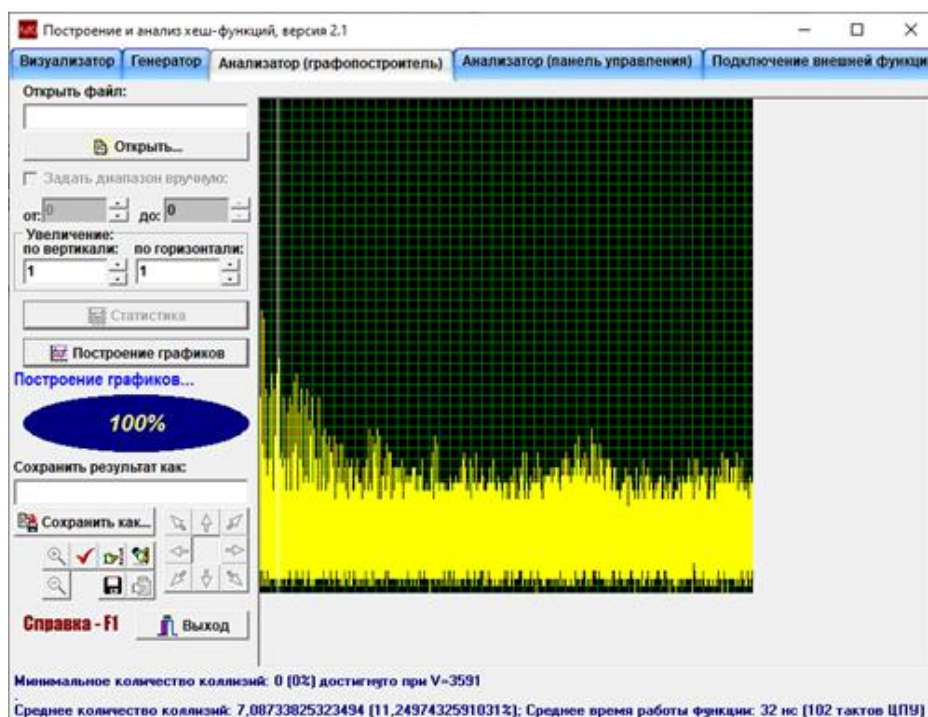


Рис. 9. Хеш-функция для языка ассемблера машины Бека SIQ

Пример простейшей хеш-функции (деления) полезной для ассемблирования исходного кода учебной машины Л. Бека УУМ/ДС (SIQ) (Бек, 1988) приведен на рисунке 9. Идеальный вариант (ноль коллизий) найден быстро, время выполнения функции 32 наносекунды. Словарь языка макроассемблера учебной машины включает 73 ключевых слова, включая директивы.

Графики на рисунке 8 и рисунке 9 отличаются по характеру: первый отображает количество коллизий в зависимости от двух параметров функции (чем ярче цвет, тем больше коллизий). На втором графике показана функция с одним параметром V (ось абсцисс), количество коллизий показывает ось ординат, вертикальная черта – первое значение V , при котором нет коллизий.

Данный пример иллюстрирует визуальный способ моделирования сложных объектов, который может дать гораздо больше полезной информации, чем изучение огромных массивов цифровых данных.

П8. Модель фрагмента российской сети суперкомпьютеров «Синтерра», созданная в профессиональном пакете моделирования Anylogic (студенческая версия PLE). Модель включает две части: вычислительную (рис. 10) и сетевую (рис. 11).

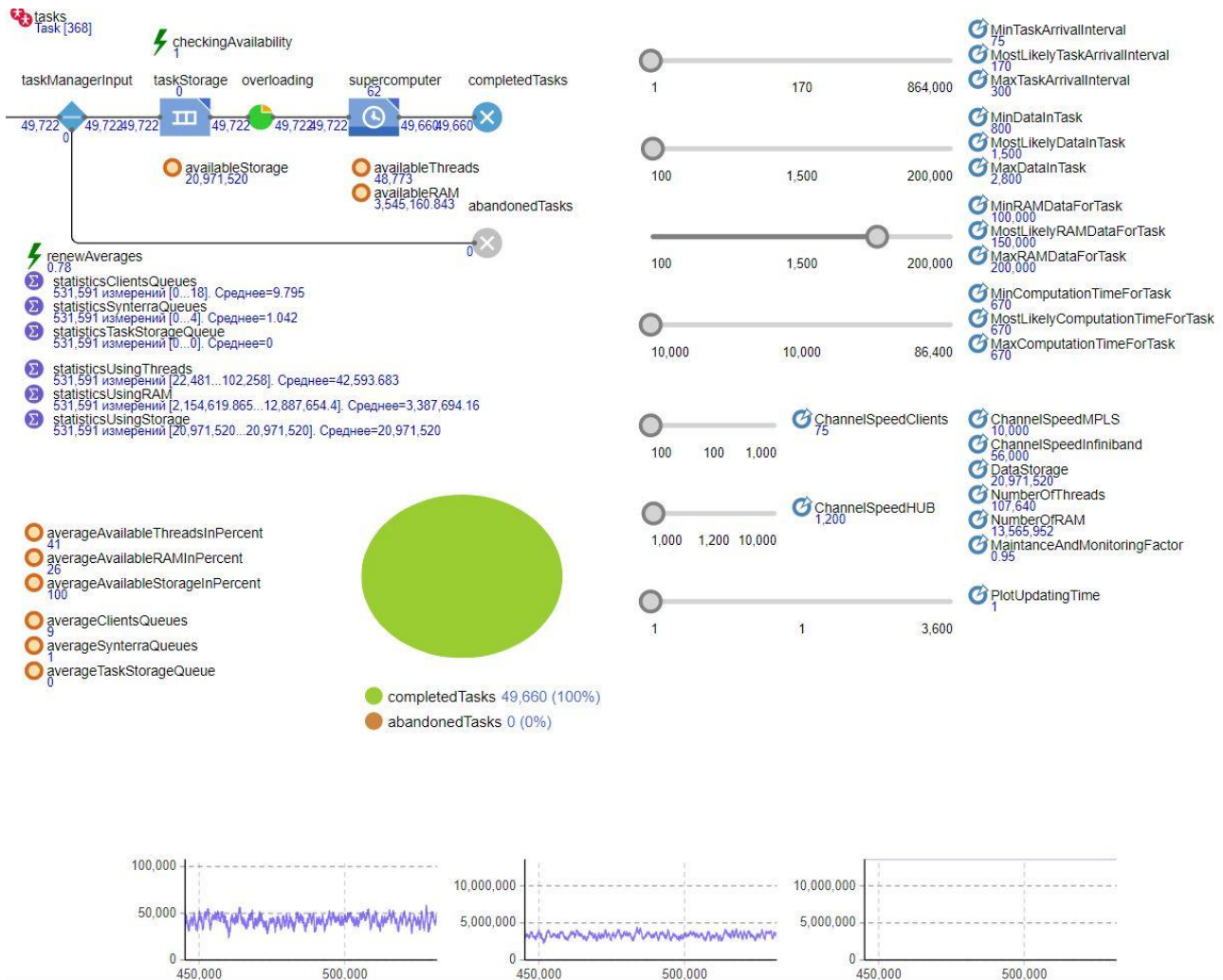


Рис.10. Вычислительная часть фрагмента «Синтерра» и ее параметры

Модель позволяет определять пропускную способность сети и вычислителя для потока заданий с разными параметрами (число ядер для задачи – до 65 тыс., объем оперативной памяти – до сотен гигабайт, объем передаваемых данных – до сотен гигабайт, время вычисления от десятков до тысяч секунд). Если образуется очередь заданий в сети или очередь к вычислителю растет, необходимо увеличить мощность системы (пропускная

ТЕОРИИ, МОДЕЛИ И ТЕХНОЛОГИИ ОБУЧЕНИЯ МАТЕМАТИКЕ И ИНФОРМАТИКЕ В СИСТЕМЕ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

способность каналов, количество ядер вычислителя, объем оперативной памяти суперкомпьютера).

На рисунке 11 представлен результат прохождения по сети 50 тыс. заданий от региональных конечных пользователей к московскому вычислительному центру. Из-за большого объема передаваемых на обработку данных в сети образуются очереди: от 766 до 796 в региональных центрах приема заявок и 6253 – в московском хабе. Таким образом, до вычислительного центра дошло 31303 заявки из 50000.

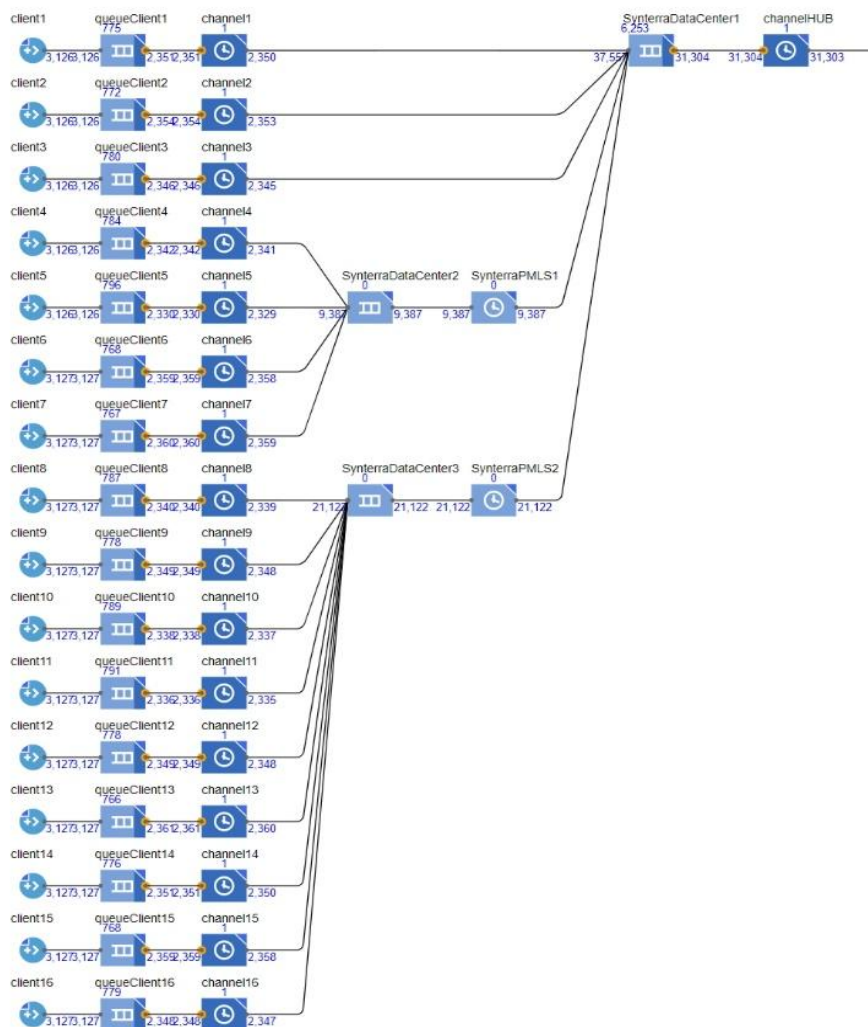


Рис. 11. Сетевая часть модели «Синтерра»

Технология моделирования

Приведенные примеры демонстрируют конечную стадию процесса моделирования – испытание отлаженных моделей. Процесс создания моделей весьма трудоемок и требует применения определенной технологии, зависящей в деталях от средств моделирования, но имеющей общую каскадную схему от постановки задачи до статистической обработки полученных результатов и формулированию правильных выводов о системе.

При использовании эмуляторов сетей Петри такой цикл включает этапы:

- постановка задачи (согласование ТЗ);
- создание конвейера обработки одного типа заявок, его быстрая визуальная отладка;
- контроль очередей и оптимизация ресурсов с применением встроенной статистики

(рис.12);

- клонирование до 2-4 конвейеров асинхронной обработки разных типов заявок с общими ресурсами и балансирование нагрузки;
- введение 4 типов нештатных ситуаций, исследование устойчивости к ним, настройка ресурсов системы и алгоритма в новых условиях;
- определение показателей эффективности (инварианты), внесение в код модели, привязка событий к ним (рис. 4). Здесь использовано 2 инварианта – количество отказов (прибавка +1 в случае любого из трех отказов) и число лет работы. В данном примере критерий успеха есть 5 лет работы и ноль отказов, критерий провала – первый отказ до 5 лет работы;
- проведение серии из 5-10 экспериментов с количеством прогонов модели по 10-100 тысяч в каждом. Число прогонов зависит от требуемой точности оценки. Если ставится задача синтеза высоконадежной модели с вероятностью безотказной работы 99.9% и достоверностью оценки 99%, может потребоваться порядка 100 тысяч прогонов и более.

Обработка полученных результатов: определение показателя эффективности как доверительного интервала с заданной достоверностью (рис. 13). Если доверительный интервал оказывается слишком широким, увеличивают число прогонов в испытаниях на предыдущем этапе. Если среднее значение процента успехов не соответствует заданию, возвращаются к начальным этапам, вплоть до корректировки ТЗ.

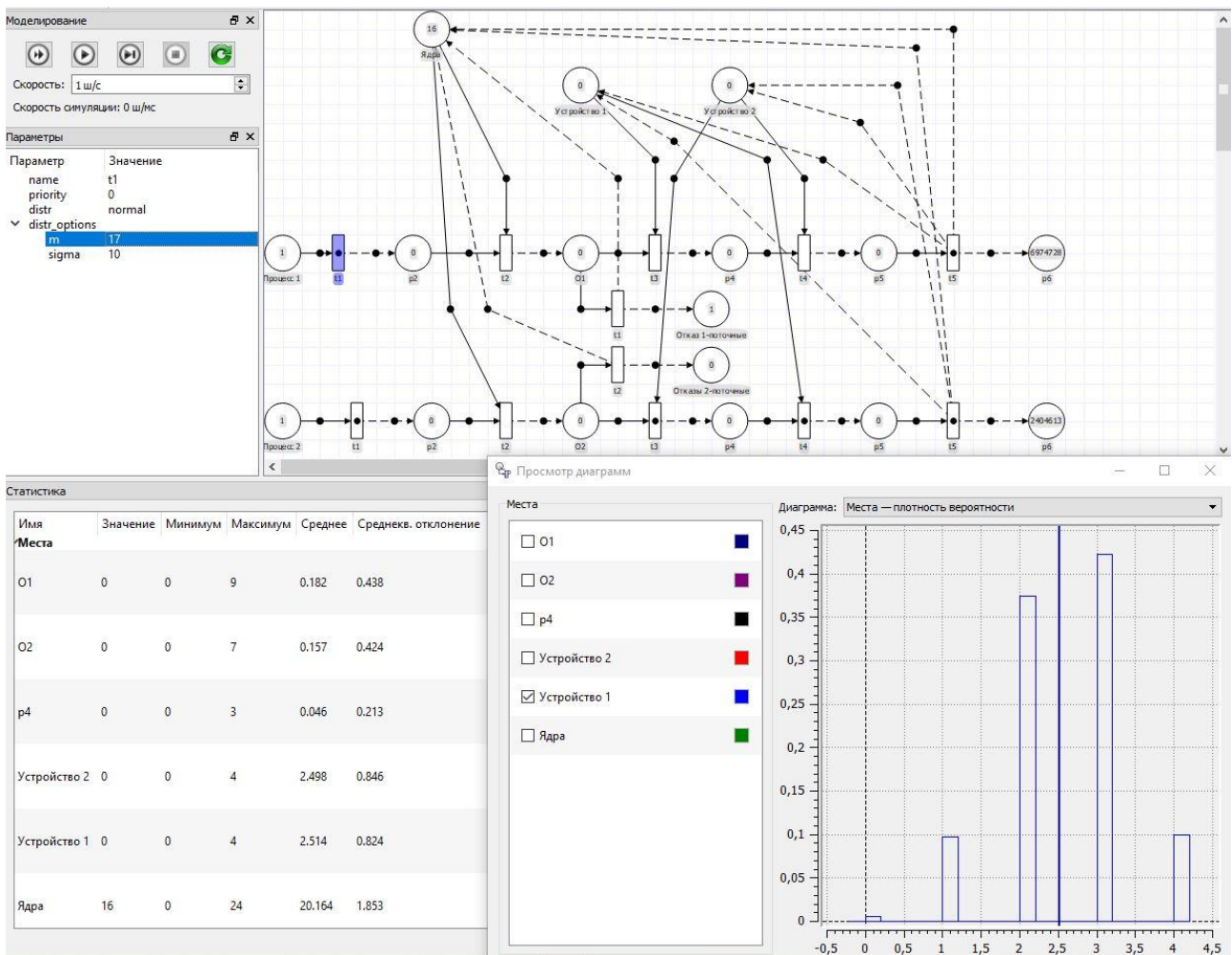


Рис. 12. Встроенная статистика для примера III

С использованием распределения Стьюдента получаем следующие значения доверительных интервалов:

- $13,2456 \pm 0,0250$ % вероятность провала с достоверностью 95%;
- $13,2456 \pm 0,0415$ % вероятность провала с достоверностью 99%.

Точность определения среднего значения и доверительных интервалов явно избыточ-

ТЕОРИИ, МОДЕЛИ И ТЕХНОЛОГИИ ОБУЧЕНИЯ МАТЕМАТИКЕ И ИНФОРМАТИКЕ В СИСТЕМЕ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

на, как минимум, можно сократить точность на 2-3 знака без влияния на результаты испытаний. Соответственно, длительность каждого испытания можно было сократить в десятки раз.

Но в случае высоких требований к надежности (например, вероятность отказа не более 0,1%) может потребоваться увеличение длительности испытаний. В пакете QPNet пока не предусмотрена автоматическая остановка испытания по достижению заданного числа прогонов модели. Эта остановка выполняется вручную, как и сохранения скриншота с результатом.

- Оформление отчета и его публичная защита.
- Сохранение файлов модели и других материалов в архиве полезных моделей (если модель представляет достаточную ценность и новизну).

Полезные модели

В ходе выполнения практических заданий, курсовых и дипломных проектов было создано несколько десятков моделей, представляющих ценность для учебного процесса и инженерных разработок:



Рис. 13. Результаты серии испытаний с избыточным числом прогонов модели

- Сегмент университетской сети суперкомпьютеров РФ «Синтерра».
- Модели 3- и 5-канальной мажоритарной памяти.
- Модель функционирования ОС Android для смартфонов.
- Модель приложения интернета вещей для смартфона.
- Модель отказоустойчивого кластера ЦП для долгоживущего спутника.
- Модель «Антикафе» со случайными циклами обслуживания заявок (часть заявок остается в системе, проходя повторно по разным конвейерам обслуживания).
- Модель производства вычислительных комплексов на базе процессоров «Эльбрус».
- Модель планировщика процессов в ОС с приоритетами.
- Модели конвейеризации операций в процессоре.
- Логистика аэропортового хозяйства и ВПП.
- Безопасность системы жизнеобеспечения на космической станции.
- Модель управления трафиком в районе станции метро Юго-Западная.
- Модель конвейера в цехе контроля герметичности электронных изделий.
- Модель системы обратного водоснабжения.
- Модель подсистемы кэш памяти микропроцессора.

Полезные модели сохраняются в облачном архиве и доступны студентам и аспирантам следующих лет обучения.

Выводы

1. Оптимальный набор инструментов компьютерного моделирования повышает качество подготовки инженеров в рамках учебных планов бакалавров, специалистов и магистров. Не менее полезен он и при обучении в аспирантуре.
2. В преподавании дисциплин инженерного профиля целесообразно применение легких в освоении и бесплатных средств моделирования.
3. Учебная дисциплина типа «Моделирование систем» не должна быть основана на

изучении единственного пакета компьютерного моделирования, особенно, если он требует дорогостоящей лицензии и мощных компьютеров.

4. В учебном процессе подготовки инженеров полезно сочетание универсальных пакетов компьютерного моделирования и специализированных программ, легких в освоении и не требующих лицензирования.

5. Применение в качестве основного программного средства в односеместровом курсе моделирования мощных и сложных пакетов требует значительного времени на освоение интерфейса, функций, языка моделирования. На выполнение практических заданий с помощью таких средств остается мало времени, построенные модели получаются достаточно упрощенными. Используя простые в освоении быстродействующие инструменты моделирования, такие как QPNet, слушатели имеют достаточно времени для создания и глубокого исследования содержательных моделей, представляющих практический интерес.

6. Разработка в технических вузах собственных программ компьютерного моделирования решает две задачи:

- создание легких в освоении и полезных в преподавании различных дисциплин специализированных или универсальных программных средств;
- повышение общих компетенций в IT-сфере участников данных проектов.

7. Применение в учебном процессе различных инструментов компьютерного моделирования дает более глубокое представление о парадигмах и методах моделирования:

- дискретно-событийный подход;
- системная динамика;
- агентное моделирование;
- визуальное моделирование;
- специальные алгоритмы и интерфейсы проблемно-ориентированных средств компьютерного моделирования.

8. В ходе многолетнего применения различных средств компьютерного моделирования целесообразно поддерживать облачный архив полезных моделей, включающий полный набор файлов по каждому проекту (задание, проект модели, коды, программа испытаний и оптимизации, отчет по проведению статистических испытаний, математическая обработка полученных результатов, общий отчет по проекту, включая скриншоты и выводы по результатам моделирования). Подобный архив является ценным учебно-методическим пособием для будущих инженеров.

9. Описанные в статье инструменты, созданные в МИРЭА, методические материалы, полезные модели доступны бесплатно в облачном хранилище <https://disk.yandex.ru/d/TfUQMNhDMJ2DhA>. Также эти материалы выложены в системе дистанционного обучения РТУ МИРЭА <https://online-edu.mirea.ru/course/view.php?id=8774>

Список литературы

- Бек Л. Введение в системное программирование. Пер. с англ. Н.А. Богомолова, В.М. Вязовского, С.Е. Морковина; под ред. Л.Н. Королева. М.: Мир, 1988.
- Бескин А.Л., Кублик Е.И. Моделирование программных систем [Электронный ресурс]: Учебное пособие. М.: Российский технологический университет, 2023.
- Бескин А.Л. Задания к практическим занятиям по дисциплине «Моделирование программных систем»: учебно-методическое пособие. М.: Российский технологический университет, 2023.
- Бескин А.Л. Графовая модель как основа верификации бортовых программ // АНО ИТ-Стандарт: сборник трудов VI международной конференции «ИТ-Стандарт 2015». М., 2015.
- Бескин А.Л. О тестировании и доказательстве правильности программ // АНО ИТ-Стандарт: сборник трудов VII международной конференции «ИТ-Стандарт 2016». М., 2016.
- Бескин А.Л. Моделирование программных систем [Электронный ресурс]: Методические

указания по использованию эмуляторов сетей Петри. М.: Российский технологический университет, 2023

Винс Д.В. Имитационное моделирование управляющих и вычислительных процессов в высокопроизводительных вычислительных системах: дис. ... канд. техн. наук. Новосибирск, 2016.

Кудж С.А., Логинова А.С. Моделирование с использованием сетей Петри // Вестник МГТУ МИРЭА. 2015. № 1 (6). С. 10–22.

Кулагин В.П., Логинов А.А. Анализ программных средств для работы с сетями Петри // Информационные технологии. 2021. №2. С. 89–96. DOI: 10.17587/it.27.89-96.

Anylogic PLE – загрузка <https://www.anylogic.ru/downloads/personal-learning-edition-download/>
CPNTools/CPN IDE – загрузка приложений. <https://cpntools.org/>

QPNet – эмуляторы модифицированных асинхронных сетей Петри QPNet 1.3.2 дизайн и отладка моделей со встроенной статистикой.

QPNet 1.2 – автоматическое выполнение моделей по достижению заданных показателей эффективности (инварианты) с рестартом

Weins D. Chernykh I. Vorobyev V. Logashenko I. Development of simulation model of HPC system for super charm-tau factory. // Journal of physics: conference series 2. Ser. 2nd Workshop on Numerical Modeling in MHD and Plasma Physics: Methods, Tools, and Outcomes. Том 1336. 2019. С 12-25.

COMPUTER MODELING TOOLS IN THE EDUCATIONAL PROCESS OF A TECHNICAL UNIVERSITY

Beskin A. L.
associate professor
beskin@mirea.ru
Moscow

MIREA – Russian State University

Muzylev N. V.
associate professor
muzylev@mirea.ru
Moscow

MIREA – Russian State University

Abstract. The training of engineers in Russia is currently one of the main tasks of higher education. Their acute shortage is felt in all spheres of the economy, and the requirements for the competencies of a modern engineer have also changed. The ongoing technological revolution – the transition to Industry 4.0 – requires new digital competencies of an engineer, including fluency in computer modeling tools. The article describes the methodology of using such tools in the educational process of preparing students and postgraduates of engineering fields at a technical university. Including teaching subjects from the curriculum, conducting research and preparing graduation papers. In 2020-2023, an experiment was conducted on the use of six computer modeling tools as part of the development of the academic disciplines "System Modeling" and "Software Systems Modeling" in the second year of the bachelor's degree program "Software Engineering", in the third year of the direction "Mathematical Modeling and Computer Science" in the discipline "Algorithms and complexity Theory" and in the third year training of graduate students in various fields. The experiment was completed successfully: the percentage of excellent grades in different groups reached 70-85%, many models were used in graduation papers, graduate students planned to include their research in their dissertations. During the experiment, cloud and remote technologies were used. All the results of the experiment are stored in a cloud archive and are available to teachers and students

from any device connected to the Internet. In the future, this technique is planned to be used in teaching various academic disciplines, but to a greater extent – in term papers, graduation papers and dissertations.

Keywords: professional training, software engineering, discrete event model, agent-based modeling, model experiment.

References

- Anylogic PLE. <https://www.anylogic.ru/downloads/personal-learning-edition-download/>
- Beck, L. L. (1997). An introduction to system programming. Addison-Wesley.
- Beskin, A. L., Kublik, E. I. (2023). Modeling of software systems [Electronic resource]: Textbook. Moscow: Russian Technological University. (In Russ.)
- Beskin, A. L. (2023). Assignments for practical classes in the discipline “Modeling of software systems”: an educational and methodological guide. Moscow: Russian Technological University. (In Russ.)
- Beskin, A. L. (2015). Grafovaya model' kak osnova verifikacii bortovyh programm [Graph model as the basis for verification of on-board programs]. *ANO IT-Standart: sbornik trudov VI mezhdunarodnoj konferencii «IT-Standart 2015»*. Moscow. (In Russ.)
- Beskin, A. L. (2016). O testirovanii i dokazatel'stve pravil'nosti program [About testing and proving the correctness of programs]. *ANO IT-Standart: sbornik trudov VII mezhdunarodnoj konferencii «IT-Standart 2016»*. Moscow. (In Russ.)
- Beskin, A. L. (2023). Modeling of software systems [Electronic resource]: Methodological guidelines for the use of Petri net emulators. Moscow: Russian Technological University. (In Russ.)
- CPNTools/CPN IDE – downloads: <https://cpntools.org/>
- Kudzh, S. A., Loginova, A. S. (2015). Modeling using Petri nets. *Herald of MSTU MIREA*, 1 (6). 10-22. (In Russ., abstract in Eng.)
- Kulagin, V. P., Loginov, A. A. (2021). Analysis of software tools for working with Petri nets. *Information technology*, 2, 89-96. DOI: 10.17587/it.27.89-96. (In Russ., abstract in Eng.)
- QPNet – emulators of modified asynchronous Petri nets QPNet 1.3.2: design and debugging of models with built-in statistics.
- QPNet 1.2 Automatic execution of Petri nets models to achieve specified quality indicators (invariants) with restart.
- Weins, D. (2016). *Imitacionnoe modelirovanie upravlyayushchih i vychislitel'nyh processov v vysokoproizvoditel'nyh vychislitel'nyh sistemah*. [Ph.D. Dissertation]. Novosibirsk. (In Russ.)
- Weins, D. Chernykh, I. Vorobyev, V. Logashenko, I. (2019). Development of simulation model of HPC system for super charm-tau factory. *Journal of physics: conference series 2. Ser. "2nd Workshop on Numerical Modeling in MHD and Plasma Physics: Methods, Tools, and Outcomes"*, 1336.

Статья поступила в редакцию 11.12.2023

Принята к публикации 14.12.2023