

Е.Б.Замятина

**О ПРИОБРЕТЕНИИ КОМПЕТЕНЦИЙ ПО РАЗРАБОТКЕ СИСТЕМ
ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ СТУДЕНТАМИ-СИСТЕМНЫМИ
ПРОГРАММИСТАМИ НА МЕХАНИКО-МАТЕМАТИЧЕСКОМ ФАКУЛЬТЕТЕ ПГНИУ**

e_zamyatina@mail.ru

Пермский государственный национальный исследовательский университет, г.Пермь

This paper describes the content of special courses “Simulation and Simulation Languages” and “The Modern Simulation Theories“. Author considers the main issues being revealed in these courses and pays attention to the themes of practical training.

Введение

Известно, что имитационное моделирование находит свое применение в самых широких областях производства, науки, бизнеса. Действительно, очень многие исследования требуют проведения имитационных экспериментов с той целью, чтобы прогнозировать поведение исследуемого объекта или сложной динамической системы. Это дает, в свою очередь, возможность принять верное решение, например, в современных условиях ведения бизнеса[1]. Так в результате проведения реинжиниринга бизнес-процесса необходимо проверить значения таких ключевых параметров для бизнес-процессов как затраты, оперативность бизнес-операций. С этой целью целесообразно симитировать поведение модернизированного бизнес-процесса, определяя при этом результативность модернизации по ключевым показателям. Таким образом, системы имитационного моделирования являются неотъемлемой частью систем поддержки принятия решений (СППР). Важную роль имитационное моделирование играет и в системах автоматизированного проектирования (САПР). Так при проектировании компьютерных сетей важно исследовать такие проблемы, как эффективность алгоритмов маршрутизации, отказоустойчивость и безопасность компьютерных сетей и т.д. Прежде, чем применить разработанный алгоритм маршрутизации в реальной компьютерной сети, целесообразно исследовать показатели его эффективности, симитировав как компьютерную сеть, так и алгоритм. Изменяя топологию сети, показатели, характеризующие скорость линий передачи связей, скорость работы вычислительных узлов, проектировщик имеет возможность исследовать поведение разрабатываемого алгоритма в самых различных условиях.

В настоящее время существует большое количество систем имитационного моделирования, однако многие из них являются коммерческими и профессиональные версии достаточно дороги. Кроме того, они не всегда могут быть применены для конкретных исследований и для конкретных нужд пользователей. Поэтому является *актуальным* изучение основ имитационного моделирования, знания архитектуры систем имитационного моделирования, принципы построения симуляторов, механизма продвижения времени. Наряду с перечисленными компетенциями, которыми следует владеть системным программистам, необходимо иметь навыки программирования и создания имитационных моделей с применением давно и широко известных систем моделирования: GPSS[2], ANYLOGIC[3], ARENA[4] и т.д. Рассмотрим более подробно темы, которые рассматриваются на лекционных и практических занятиях специального курса «Имитационное моделирование и языки моделирования», проводимого на механико-математическом факультете по направлению «бакалавриат».

Приобретение студентами компетенций по разработке систем имитационного моделирования

На лекционных занятиях студенты подробно рассматривают основные понятия, связанные с имитационным моделированием: событие, процесс, активность, динамическая система, имитационная система, симулятор (здесь под термином «симулятор» понимают управляющую программу, которая определяет очередность наступления того или иного события»), сценарий поведения. Затем подробно рассматривается понятие времени (дискретное, непрерывное) и механизм продвижения времени. Рассказ о механизме продвижения времени предваряет теоретический материал, связанный с вопросами архитектур программных систем имитационного моделирования, которые реализуют ту или иную парадигму.

Различают событийно-ориентированные, процессо-ориентированные, объектно-ориентированные и агентно-ориентированные системы имитационного моделирования. Системы имитационного моделирования разделяются на эти классы по тому, какое понятие лежит в их основе.

Так событийно-ориентированные системы имитационного моделирования характерны тем, что поведение системы определяется последовательностью событий, которые изменяют состояние системы и обычно планируют будущие события. События представляют собой обычно отдельные процедуры. Будущие события размещаются в календаре событий. Каждое событие планируется на определенный момент будущего времени. События в календаре событий обычно отсортированы по возрастанию этих моментов времени. Глобальная системная переменная хранит текущий момент времени. Управляющая программа (симулятор) выбирает из календаря событий событие (процедуру) с наименьшим моментом времени и передает этой процедуре управление.

Реализацию событийно-ориентированную системы имитации целесообразно объяснять на примере реализации системы имитации SMPL.

Для процессо-ориентированных СИМ характерно продвижение транзактов (или заявок, этот термин принят в системах массового обслуживания (СМО)) по блокам, которые ассоциируются с процедурами. Процедуры выполняются с появлением транзакта в блоке. Типичным представителем процессо-ориентированных СИМ является СИМ GPSS, которая успешно применяется в прикладных исследованиях в самых разнообразных областях науки и производства. Подтверждением тому являются труды, представленные на регулярных конференциях ИММОД[6,7,8]. СИМ GPSS успешно продвигается на отечественном рынке программных продуктов компанией Элина-Компьютер[6].

Транзакты появляются в модели при прохождении ими блока GENERATE (или SPLIT), а удаляются в блоке TERMINATE (или ASSEMBLE). Одновременно в модели могут находиться несколько транзактов, но продвигается только один. Управляет продвижением транзактов специальная подпрограмма – интерпретатор GPSS. Он продвигает транзакт по цепям текущих событий (события, запланированные на текущий момент времени), цепям будущих событий (события, запланированные на текущий момент), цепям задержек и т.д.[2]

При изучении СИМ, которые являются процессо-ориентированными, целесообразно более подробно рассмотреть возможности программирования на языке GPSS и овладеть навыками программирования. С этой целью студентам механико-математического

факультета предлагается выполнить лабораторную работу, которая предполагает разработку программ на языке GPSS[9].

При рассмотрении объектно-ориентированной парадигмы следует сосредоточить внимание студентов на возможность ведения отдельных локальных календарей событий для каждого объекта и особенности реализации механизма продвижения времени для объектно-ориентированной системы имитации Triad, которая разрабатывалась несколько лет тому назад в ПГНИУ[10]. При объяснении материала целесообразно рассмотреть вопросы оптимизации СИМ, указывая на возможность особой организации структуры календаря событий[10].

Компонентами СИМ (чаще всего это относится к процессо-ориентированным и событийно-ориентированным моделям) являются: системные часы, списки событий, программные средства сбора статистики, подпрограмма инициализации, подпрограмма продвижения времени, подпрограммы, реализующие события, библиотечные подпрограммы, генератор статистических отчетов, управляющая программа.

Далее студентам предлагается рассмотреть вопросы организации подсистемы сбора и обработки статистических данных (для статистического моделирования), а именно, процедуры сбора и обработки данных для одноканальных устройств и многоканальных устройств (коэффициент загрузки устройства, среднее время занятости устройства и т.д.) и для очередей (максимальная длина очереди, средняя длина очереди и т.д.).

Еще одна лабораторная работа предполагает создание процедур-генераторов случайных чисел по таким законам распределения как равномерное, экспоненциальное, Вейбулла, Пуассона и т.д.[11]

Следующая лабораторная работа предполагает разработку интерфейса для системы имитационного моделирования. Разработка интерфейса предполагает возможность задания параметров для генерации заявок по некоторому закону распределения, параметров для времени обработки заявок одноканальным и многоканальным устройством (также распределенного по одному из случайных законов распределения), для задания последовательности обработки заявок (в замкнутой, приоритетной, многофазной СМО).

Результатом выполнения лабораторных работ является прототип системы имитационного моделирования, которую студенты разрабатывают самостоятельно.

При рассмотрении объектно-ориентированной парадигмы студентам предлагается более подробно познакомиться с особенностями ИМ, разработанных с помощью программных средств Simula-67 и Triad, а при изучении агентно-ориентированной парадигмы – программными средствами ANYLOGIC.

Современные программные средства имитационного моделирования

Теоретический материал, представленный выше, охватывает начальные знания о разработке систем имитационного моделирования. Однако в настоящее время в связи с развитием высокопроизводительных средств программирования распределенное имитационное моделирование также получило развитие.

Кроме того, следует обратить внимание студентов на широкое использование средств искусственного интеллекта при разработке СИМ: применение онтологий (организация адаптируемого графического интерфейса, автоматическая интеграция компонентов систем имитационного моделирования, верификация и валидация СИМ), средства Data Mining и Knowledge Discovery при обработке результатов имитационного моделирования.

С этой целью для студентов магистратуры разработан курс «Современные теории имитационного моделирования»[12,13].

Практические навыки распределенного моделирования студенты получают при изучении и реализации консервативных и оптимистических алгоритмов.

Особое место занимает вопрос построения и использования агентных моделей.

Агентное моделирование[14,15] в настоящее время завоевывает все более прочные позиции при исследовании сложных динамических систем. Это объясняется тем, что мир, в котором мы живем, все более усложняется, появляются новые взаимосвязи и взаимоотношения. Таким образом, традиционные подходы к моделированию (системная динамика, дискретное моделирование) не могут удовлетворить исследователей.

Рассмотрим пример в прикладной области – это моделирование экономических рынков, которое традиционно полагалось на представление об совершенных рынках, однородных посредниках, и долговременном равновесии, потому что эти предположения позволяли решить эти задачи аналитически. Если же отказаться от этих предположений, то используемый ранее подход к исследованию экономических рынков становится неприемлемым.

Кроме того, вычислительные мощности быстро увеличиваются. Теперь можно просчитывать крупномасштабные микромодели, что выглядело бы неправдоподобным ещё пару лет назад.

Вышесказанное позволяет сделать заключение о том, что прежние подходы к моделированию не всегда являются адекватными и следует искать новые подходы. Таким подходом является агентное моделирование.

Агентное моделирование может быть реализовано на небольших, настольных компьютерах или с использованием крупных кластеров компьютеров, или на любом другом варианте между первыми двумя.

Настольные агентные модели могут быть простыми. Обычно они разрабатываются программистом с помощью утилит, который может изучить их за несколько недель. Крупномасштабное агентное моделирование обычно выполняется с использованием специальных сред моделирования. Эти среды включают: (1)планировщика (по времени); (2) механизмы коммуникации; (3)гибкие топологии взаимодействий агентов; (4) широкий выбор устройств для хранения и отображения состояния агентов. Крупномасштабные агентные модели в основном требуют больших навыков и больших ресурсов на разработку по сравнению с настольными средами. Благодаря значительным открытым разработкам и инвестициям в разработку многие из сред для АМ доступны широкому пользователю (Repast, Swarm, NetLogo и MASON).

The REcursive Porous Agent Simulation Toolkit (Repast) – это ведущий открытый и свободный источник библиотек для крупномасштабного агентного моделирования. Repast поддерживает разработку чрезвычайно гибких моделей агентов и используется в моделировании социальных процессов. Пользователь строит свою модель, включая в свои программы компоненты из библиотеки Repast или используя визуальный Repast для среды Python Scripting.

Существует три версии Repast, названных Repast for Python (Repast Py), Repast for Java (Repast J) и Repast for the Microsoft.NET framework (Repast .NET).

Swarm (стая, рой) был первой средой разработки АМ приложений, впервые запущен в 1994г. Chris Langton at the Santa Fe Institute. Swarm – это открытый и бесплатный набор библиотек с открытым кодом, который настоящее время поддерживается Swarm Development Group (SDG NetLogo – это другая кросс платформенная мультиагентная среда для моделирования, которая широко используется и поддерживается. Первоначально основанная на системе StarLogo, NetLogo приспособливает агентные системы, состоящие из комбинации живых и программных агентов-участников.

Таким образом, возникает вопрос о разработке методических указаний и лабораторного практикума по использованию перечисленных открытых библиотек. Эти работы в настоящее время ведутся на кафедре математического обеспечения вычислительных систем.

Библиографический список

1. Емельянов А.А., Власова Е.А., Дума Р.В. Имитационное моделирование экономических процессов: Учебное пособие. М.: Финансы и статистика, 2002.
2. Шрайбер Т.Дж. Моделирование на GPSS. М.: Машиностроение, 1980.
3. Карнов Ю. Г. Имитационное моделирование систем. Введение в моделирование с AnyLogic 5. — СПб: БХВ-Петербург, 2006.
4. Маклаков С.В. Имитационное моделирование с ARENA. <http://www.compress.ru/article.aspx?id=11212>. [Электронный ресурс]. [Проверено 28.02.2013]
5. Прицкер А. Введение в имитационное моделирование и язык SLAM-II. М.: Мир, 1987.
6. www.gpss.ru. Материалы конференций ИММОД-2003, ИММОД-2005, ИММОД-2007, ИММОД-2009, ИММОД-2011. [Электронный ресурс]. [Проверено 28.02.2013]
7. www.xjtek.ru Материалы конференций ИММОД-2003, ИММОД-2005, ИММОД-2007, ИММОД-2009, ИММОД-2011. [Электронный ресурс]. [Проверено 28.02.2013]
8. www.simulation.su Материалы конференций ИММОД-2003, ИММОД-2005, ИММОД-2007, ИММОД-2009, ИММОД-2011. [Электронный ресурс]. [Проверено 28.02.2013]
9. Томашевский В., Жданова Е. Имитационное моделирование в среде GPSS. М.: Бестселлер, 2003. 416с.
10. Миков А.И. Автоматизация синтеза микропроцессорных управляющих систем. -Иркутск, Иркут.ун-т., 1987
11. Лядова Л.Н. Имитационное моделирование: Методические указания. Пермь: Пермский университет, 2003. [Электронный ресурс]
12. Замятина Е.Б. Современные теории имитационного моделирования. Специальный курс., www.edu.ru. [Электронный ресурс]. [Проверено 28.02.2013]
13. Замятина Е.Б. Современные теории и системы имитационного моделирования: учеб.-метод. пособие / Е.Б. Замятина, В.В. Ланин, Л.Н. Лядова, А.Н. Фирсов; Перм. ун-т. – Пермь, 2007.
14. Macal С.М., North M.J. Tutorial On Agent-Based Modeling And Simulation // Proceedings of the 2005 Winter Simulation Conference. 2005
15. Борщёв А. От системной динамики и традиционного ИМ к практическим агентным моделям: причины, технология, инструменты. Доступно на сайте: www.gpss.ru. [Электронный ресурс]. [Проверено 28.02.2013]

