

Модели времени в имитационном моделировании

Что же такое время? Если никто меня об этом не спрашивает, я знаю, что такое время; если бы я захотел объяснить спрашивающему – нет, не знаю.

Августин Аврелий

Рассмотрены основные подходы к решению классических задач имитационного моделирования и модели времени: дискретно-событийное и непрерывное моделирование, а также моделирование Монте-Карло. Обсуждаются их основные положения, преимущества, недостатки и конкретные реализации. На основе проведенного исследования показано место оригинального программного средства G-IPS Ultimate в ряду других программных продуктов для решения учебных и прикладных задач имитационного моделирования.

Ключевые слова: имитационное моделирование, время, модель времени, программное средство

ВВЕДЕНИЕ

Моделирование, т. е. построение и решение моделей, является одним из базовых методов исследования в естественных науках [1, 2]. Имитационным моделированием (*simulation*) неформально принято называть метод исследования, при котором изучаемая система заменяется моделью, с необходимой точностью описывающей реальную систему, после чего над моделью проводятся эксперименты с целью получения новых знаний об этой системе. Имитацией называют процесс проведения эксперимента над моделью с целью предсказать, как поведет себя реальная система, когда такой эксперимент будет поставлен на ней. Заметим, что реальная система на момент имитации может ещё не существовать. Например, в авиации конструкторы строят масштабные модели самолетов, чтобы проверить аэродинамические характеристики их конструкции. Манипулируя конструкцией крыла или формой фюзеляжа на модели, конструктор может определить, насколько эти изменения влияют на поток воздуха, попадающий на соответствующие поверхности. Если модель достаточно «правдиво» (адекватно) представляет реальный самолет, конструктор получает доказательства, что реальная система будет вести себя определенным образом, с учетом привнесенных изменений. Преимущество моделирования заключается в том, что имеется возможность провести несколько экспериментов на модели в различных условиях без необходимости нести расходы по созданию или изменению реальной системы. Гораздо проще вносить изменения в имитационную модель, чем пытаться переработать полно-размерный самолет, а в некоторых случаях эксперимент на реальной системе просто невозможен.

Никакой мыслимый объект или процесс взаимодействия объектов не существует вне времени. Любой, даже самый элементарный процесс, такой, как столкновение элементарных частиц, длится какое-то время. Однако в различных задачах время учитывается по-разному, а стало быть, по-разному отражается на проектируемых моделях и имитации.

Модель, в самом широком смысле, – это любой мысленный или знаковый образ моделируемого объекта (оригинала). К их числу относятся гносеологические образы (воспроизведение, отображение исследуемого объекта или системы объектов в виде научных описаний, теорий, формул и т. п.), схемы, чертежи, графики, планы, карты и т. д. [3]. Построение моделей основано на принципе абстракции. В имитационном моделировании существует и другое определение, согласно которому модель – это статическое представление системы. Имитация добавляет в модель временной аспект, показывая, как будет меняться система с течением времени [4]. Таким образом, имитация может быть определена как изменение состояния модели во времени. Временной фактор настолько важен, что является одним из главных критериев классификации имитационных моделей. Далее мы будем рассматривать компьютерное имитационное моделирование, т. е. исследование поведения математической модели, решаемой с помощью компьютера.

ПОНЯТИЕ ВРЕМЕНИ

На сегодняшний день не существует определения времени, охватывающего все прикладные области. С одной стороны, согласно некоторым определениям время – это одна из осей пространства-времени, согласно другим – фундаментальное понятие человеческого мышления, отображающее изменчивость мира,

процессуальный характер его существования, наличие в мире не только «вещей» (объектов, предметов), но и событий [5]. С другой стороны, время – понятие, позволяющее установить, когда произошло то или иное событие по отношению к другим событиям. Измерение времени подразумевает введение временной шкалы, пользуясь которой можно соотносить эти события¹.

Основной концепцией, используемой сегодня при формализации времени, является ось времени (временная ось или стрела времени – в зависимости от дисциплины), представляющая собой прямую (т. е. математически одномерный объект), протянутую из прошлого в будущее, со шкалой для количественного измерения промежутков времени. Из любых двух несовпадающих точек оси времени одна всегда является будущим относительно другой, что позволяет соотносить совершение событий.

В зависимости от исследуемых процессов рассматривают физическую и логическую ось времени. Физическое время привязано к некоторому устройству, генерирующему физические отметки времени, – *физическим часам*, а логическое время использует только «отношение предшествования» событий на некоторой оси времени, отметки на которой могут не совпадать с физическим временем. Введение логических осей времени связано с локальностью (демонстрируемой теорией относительности) физических часов. Логические оси позволяют рассматривать события в пространственно-протяженных системах. Любая имитация «перемещает» модель по логической оси времени, последовательно совершая действия над ней.

МОДЕЛИ ВРЕМЕНИ

Рассмотрим три типа моделирования, кардинально различными способами учитывающие время:

- дискретно-событийное моделирование,
- непрерывное моделирование,
- моделирование Монте-Карло.

Каждый тип имеет определенный способ добавления аспекта времени к модели. Дискретно-событийное моделирование зависит от наступления определенных событий для перехода модели из одного состояния в другое. Непрерывное моделирование представляет собой изменение состояния в континууме² времени независимо от событий, происходящих в системе. Термин «Монте-Карло» применяется к тем моделям, в которых течение времени не является существенным.

Проиллюстрируем различия между первым и вторым типами моделирования на примере светофора и автомобиля. Светофор может находиться в трех состояниях: красный, желтый, зеленый. Они сменяются мгновенно, когда срабатывает внутренний счетчик времени и логическая схема переключает цвет. Автомобиль не может мгновенно изменять скорость своего движения. Он приобретает ускорение, которое определяется как изменение в скорости с течением

времени. Это изменение в скорости – непрерывное во времени событие, а не мгновенное изменение [4].

Следует отметить, что в реальном мире все процессы являются непрерывными. Даже в примере со светофором изменение цвета – это также некоторый процесс, который происходит в течение некоторого времени. Однако во многих задачах можно не учитывать внутреннюю структуру процессов малой длительности и считать их мгновенными без потерь смысла и точности результатов эксперимента [6].

Дискретно-событийное моделирование

Формально дискретно-событийное моделирование можно определить как последовательность изменений в модели, вызванных хронологической последовательностью событий, происходящих в системе. Все события считаются происходящими мгновенно и приводят к изменению состояния системы. Состояние системы определяется как совокупность переменных, полностью описывающих систему на уровне абстракции соответствующей модели.

Важнейший компонент дискретно-событийного моделирования – часы. Часы отсчитывают время имитации и могут использоваться для запуска и синхронизации событий в системе (так, в примере со светофором часы могут отслеживать время переключения светофора).

В отличие от реальных систем в имитационных моделях время можно «растягивать», чтобы детально рассмотреть исследуемый процесс, или «ускорять», чтобы значительно быстрее узнать результат эксперимента. И это – одно из ключевых преимуществ имитационного моделирования.

Популярной областью применения дискретно-событийного моделирования является моделирование систем массового обслуживания (СМО), т. е. систем, которые производят обслуживание поступающих в них заявок. Это обслуживание в СМО производится определенными устройствами. Классическая СМО содержит от одного до бесконечного числа устройств. В зависимости от наличия возможности ожидания начала обслуживания СМО подразделяются на три вида:

- 1) системы с потерями, в которых заявки, не нашедшие в момент поступления ни одного свободного устройства, теряются;
- 2) системы с ожиданием, в которых имеется накопитель бесконечной ёмкости для буферизации поступивших заявок; при этом ожидающие заявки образуют очередь;
- 3) системы с накопителем конечной ёмкости (с ожиданием и ограничениями), в которых длина очереди не может превышать ёмкости накопителя; при этом заявка, поступающая в переполненную СМО (отсутствуют свободные места для ожидания), теряется.

Базовыми СМО с очередями являются следующие (рис. 1 – 4):

1. Одна очередь – одно обслуживающее устройство.
2. Одна очередь – много обслуживающих устройств.
3. Много очередей – одно обслуживающее устройство.
4. Много очередей – много обслуживающих устройств.

¹ Энциклопедия Кольера

² Континуум (от лат. continuum – непрерывное): в математике – непрерывная совокупность, например, совокупность всех точек отрезка на прямой или всех точек прямой, эквивалентная совокупности всех действительных чисел. (Большой энциклопедический словарь. 2000.)

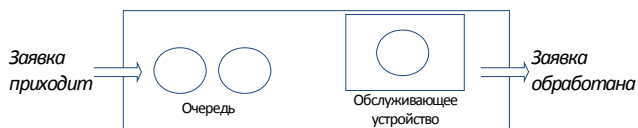


Рис. 1. СМО: одна очередь – одно обслуживающее устройство

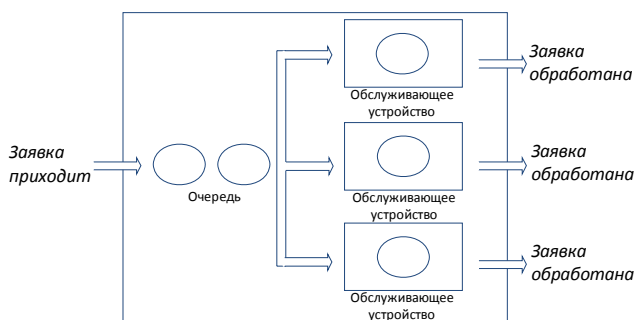


Рис. 2. СМО: одна очередь – много обслуживающих устройств

Преимущество использования моделей СМО заключается в том, что из моделей компонентов некоторой системы можно составить более сложную СМО, описывающую всю систему.

Цифровой компьютер прекрасно подходит для данного типа моделирования, ведь он, по сути, является дискретной машиной [4]. Так, программный пакет SCADA, предназначенный для разработки или обеспечения работы в реальном времени систем сбора, обработки, отображения и архивирования информации об объекте мониторинга или управления, может являться частью АСУ ТП, АСКУЭ, системы экологического мониторинга, научного эксперимента, автоматизации здания и т. д. Такого рода системы часто применяют на высокотехнологичном производстве, где присутствует набор датчиков, описывающих состояние системы, и набор устройств, позволяющих изменять это состояние. Пользовательский интерфейс дает возможность дис-

петчеру в интерактивном режиме отслеживать показатели датчиков реальной системы и получать рекомендации о дальнейших действиях. Датчики опрашиваются в дискретном времени (например, один раз в 5 секунд), и их показания обновляются в диспетчерском интерфейсе.

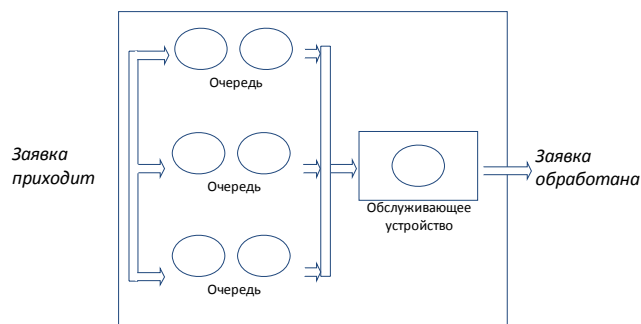


Рис. 3. СМО: много очередей – одно обслуживающее устройство

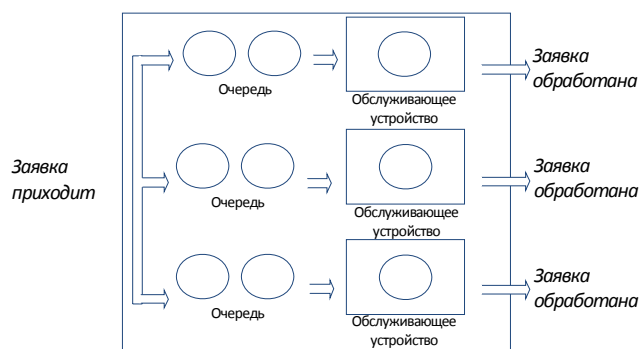


Рис. 4. СМО: много очередей – много обслуживающих устройств

В табл. 1 представлены некоторые широко известные программные продукты, применяемые для дискретно-событийного моделирования [7].

Таблица 1

**Программные продукты,
применяемые для дискретно-событийного моделирования**

Программный продукт	Фирма-разработчик	Адрес в Интернете
Arena	Systems Modeling Corporation	http://www.sm.com/
CSIM18	Mesquite Software, Inc.	http://www.mesquite.com/
Extend	Imagine That, Inc.	http://www.imaginethatinc.com/
GPSS/H	Wolverine Software Corporation	http://www.wolverinesoftware.com/
iGrafx Process	Micrografx, Inc.	http://www.micrografx.com/
Micro Saint	Micro Analysis & Design	http://www.maad.com/
ProcessModel	ProcessModel, Inc.	http://www.processmodel.com/
Promodel/MedModel	PROMODEL Corporation	http://www.promodel.com/
Silk	ThreadTec, Inc.	http://www.threadtec.com/
SIMSCRIPT II.5 and SIMPROCESS	CACI Products Company	http://www.caci.com/
Simul8	Simul8 Corporation	http://www.SIMUL8.com/
Taylor ED	F&H Simulations, Inc.	http://www.taylor-ed.com/
Witness	Lanner Group	http://www.lanner.com/corporate/

Непрерывное моделирование

Как было сказано ранее, дискретно-событийное моделирование основывается на хронологической последовательности событий, каждое из которых приводило к изменению состояния системы. Непрерывное моделирование описывается как система переменных, непрерывно изменяющих свои значения во времени. Рассмотрим такой простой пример, как падающий шарик. Единственная переменная, описывающая данную модель, – это скорость падения. Пока шарик зафиксирован, он имеет нулевую скорость падения, как только мы его выпустили, под действием силы тяжести скорость падения шарика начала возрастать. График зависимости скорости шарика от времени падения представлен на рис. 5. Эта кривая была получена нами на основе моделирования, исходя из предположения, что скорость шарика постоянно растет, поэтому она достаточно точно отражает реальное поведение физической системы.

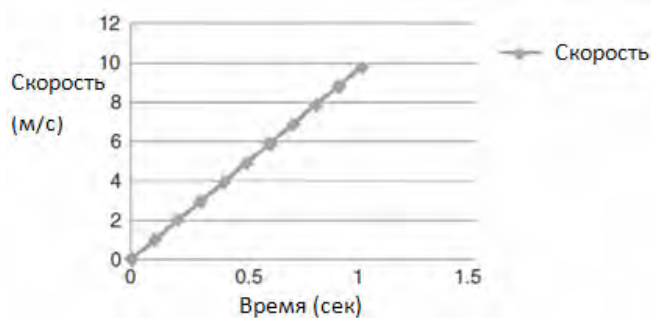


Рис. 5. Зависимость скорости падающего шарика от времени падения

К непрерывному моделированию часто прибегают в случаях исследования физических систем, включающих механические, тепловые или гидравлические компоненты. Но оно может быть использовано и для других целей, например, для исследования распространения заболеваний [4]. График, показывающий количество инфицированных людей в больших популяциях, может быть описан как непрерывная функция от времени, а следовательно, представлен в компьютере (рис. 6).

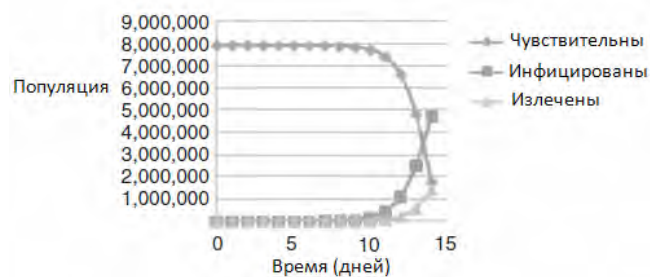


Рис. 6. Распространение заболевания в больших популяциях

Для непрерывного моделирования во многих случаях используют системы дифференциальных уравнений [8], поэтому довольно сложно назвать какие-либо программные продукты, созданные специально для непрерывного моделирования. Обычно для него используют либо универсальные математические пакеты (*Mathcad*, *Mathlab*, *Mathematica* и др.), либо уникальное программное обеспечение, разработанное под конкретную задачу.

Моделирование Монте-Карло

Как было сказано выше, термин «Монте-Карло» применяется к тем моделям, в которых течение времени не является существенным. Например, к сложным финансовым моделям, описанным с помощью электронных таблиц. Если параметрами модели являются такие количественные показатели, как, скажем, процентные ставки или денежные потоки, то задав эти параметры в виде случайных значений с некоторой функцией распределения, на выходе финансовой модели также получим набор случайных переменных. Оценки функции распределения этих переменных можно получить при многократном пересчете электронных таблиц, каждый раз принимая на вход распределения вероятностей, используемые в модели. Популярное программное обеспечение для стохастического моделирования электронных таблиц приведено в табл. 2.

Следует отметить большой потенциал данного подхода, так как аналитики в самых разных отраслях часто представляют данные в виде электронных таблиц [9]. Фактически, моделирование Монте-Карло можно считать частным случаем дискретно-событийного моделирования, в котором результат имитации становится известен после первого же наступившего события (генерации случайных величин).

Таблица 2

Программные продукты, применяемые для моделирования Монте-Карло

Программный продукт	Фирма-разработчик	Адрес в Интернете
@Risk	Palisade	http://www.palisade.com/
Crystal Ball	Oracle	http://www.decisioneering.com/
GPSS/H	Wolverine Software Corporation	http://www.wolverinesoftware.com/
Statistica	StatSoft	http://www.statsoft.ru
Excel	Microsoft	http://www.microsoft.com

Однако один результат, как правило, не пригоден для статистических оценок, в то время как выборки, полученные при многократных прогонах модели, позволяют получить статистически достоверные данные.

МОДЕЛИРОВАНИЕ В *G-IPS ULTIMATE*

Так как моделирование Монте-Карло можно считать частным случаем дискретно-событийного моделирования с дополнительной статистической обработкой, а задачи непрерывного моделирования слишком предметно-ориентированы, наибольший интерес представляют средства дискретно-событийного моделирования, которые способны охватить задачи из разных прикладных областей. Подобные системы сегодня присутствуют как в промышленности (например, *SCADA*, *GPSS* и т. п.), так и в российской высшей школе [10]. Но в процессе анализа реального применения имитационного моделирования были выявлены не только сильные стороны, но и различные недостатки, начиная с высокой стоимости и заканчивая слишком специфическим способом представления знаний, который достаточно сложно освоить. В результате было решено разработать систему, в которой эти недостатки были бы исключены или минимизированы. В настоящее время такая система разрабатывается на кафедре информационной безопасности и программной инженерии Российского государственного социального университета и называется *G-IPS Ultimate*.

Система *G-IPS Ultimate* в большой степени ориентирована на дискретно-событийное моделирование. В её основе лежит дискретное описание компонентов системы в виде графовых моделей особой топологии и переходов (связей) между этими моделями [11]. Однако она может применяться и для некоторых задач непрерывного моделирования благодаря опции, позволяющей компонентам одной системы действовать абсолютно независимо друг от друга, т.е. параллельно. Кроме того, в *G-IPS Ultimate* реализован метод построения недетерминированных моделей с помощью задания вероятностей для действий, меняющих состояние моделируемой системы. При этом невозможно будет предугадать хронологию событий, воздействующих на систему, и в системе. *G-IPS Ultimate* позволяет также исключить временной фактор в модели, что делает её вполне применимой для решения задач с помощью моделирования Монте-Карло.

Если обратить внимание на то, что большинство реально функционирующих систем можно представить в виде систем массового обслуживания, то благодаря опции параллельного опроса датчиков, параллельного выполнения действий и параллельного прогона моделей можно значительно ускорить процесс имитации и, в частности, принятия решений [12].

СПИСОК ЛИТЕРАТУРЫ

1. Рыжиков Ю. И. Имитационное моделирование. Теория и технологии. – М. : Альтекс, 2004. – 384 с.
2. Шеннон Р. Имитационное моделирование систем – искусство и наука. – М. : Мир, 1978. – 425 с.
3. Ивин А. А., Никифоров А. Л. Словарь по логике. – М. : ВЛАДОС, 1997.
4. Sokolowski J. A., Banks C. M. Principles of modeling and simulation: multidisciplinary approach. – New Jersey : A JOHN WILEY & SONS, 2009.
5. Философия : энциклопедический словарь / под ред. А. А. Ивина. – М. : Гардарики, 2004.
6. Gheorghe L. Continuous/Discrete Co-simulation interfaces from formalization to implementation. – Montreal : Polytechnique de Montreal, 2009.
7. Isken M. Computer simulation in Management Engineering // Management Engineering. – Chicago : Healthcare Information and Management Systems Society, 2001. – P. 179 – 196.
8. Kwan Hee Han. Programmable Logic Controller // Object-Oriented Modeling, Simulation and Automatic Generation of PLC Ladder Logic. – 2010.
9. Томашевский В., Жданова Е. Имитационное моделирование в среде GPSS. – М. : Бестселлер, 2003. – 416 с.
10. Карпухин И. Н., Незнанов А. А. Системы имитационного моделирования учебного назначения в российской высшей школе. – М. : Спутник, 2011. – С. 132 – 142.
11. Карпухин И. Н., Незнанов А. А. Программные средства имитационного моделирования процессов принятия решений реального времени // Интегрированные модели, мягкие вычисления, вероятностные системы и комплексы программ в искусственном интеллекте. Т. 2. – Коломна, 2009. – С. 132 – 140.
12. Рыжиков Ю. И. Имитационное моделирование систем массового обслуживания. – Л. : ВИККИ им. А. Ф. Можайского, 1991. – 111 с.
13. Емельянов А. А., Власова Е. А., Дума Р. В. Имитационное моделирование экономических процессов. – М. : Финансы и статистика, 2002. – 368 с.
14. Павловский Ю. Н., Белотелов Н. В., Бродский Ю. И. Имитационное моделирование. – М. : Академия, 2008. – 240 с.

Материал поступил в редакцию 13.03.12.