



Дискретно-событийное моделирование ст. преподаватель кафедры Вычислительной техники Тихвинский В.И.

Основные понятия

Дискретный (от лат. *discretus* — разделенный, прерывистый) — прерывистый; состоящий из отдельных частей; дробный; изменяющийся между несколькими различными стабильными состояниями; существующий лишь в отдельных точках.

Событие — то, что имеет место, происходит, наступает в произвольной точке пространства-времени; значительное происшествие, явление или иная деятельность как факт общественной или личной жизни; подмножество исходов эксперимента.

Дискретно-событийное моделирование(а)

Дискретно-событийное моделирование (англ. discrete-event simulation, DES) — это разновидность имитационного моделирования в котором функционирование системы представляется как хронологическая последовательность событий. Событие происходит в определенный момент времени и знаменует собой изменение состояния системы.

Дискретно-событийное моделирование наиболее развито и имеет огромную сферу приложений — от логистики и систем массового обслуживания до транспортных и производственных систем. Этот вид моделирования наиболее подходит для моделирования производственных процессов.



Дискретно-событийное моделирование(б)

Дискретно-событийное моделирование — подход, соответствующий низкому и среднему уровням абстракции. Термин дискретно-событийное моделирование исторически закрепился за моделированием систем обслуживания потоков объектов некоторой природы: клиентов банка, автомобилей на заправочной станции, телефонных вызовов, пациентов в поликлиниках и т.п. Обслуживание при этом может быть достаточно сложным. Например, рассматривая конвейерные системы для поточной сборки изделий как системы массового обслуживания, разработчик модели должен учитывать характеристики самих конвейеров, алгоритмы сборки изделий и разного рода дополнительные условия (например, наличие ресурсов конкретного типа). Дискретно-событийный подход широко используется в моделировании бизнес-процессов, производства, логистики, здравоохранения и т.д.



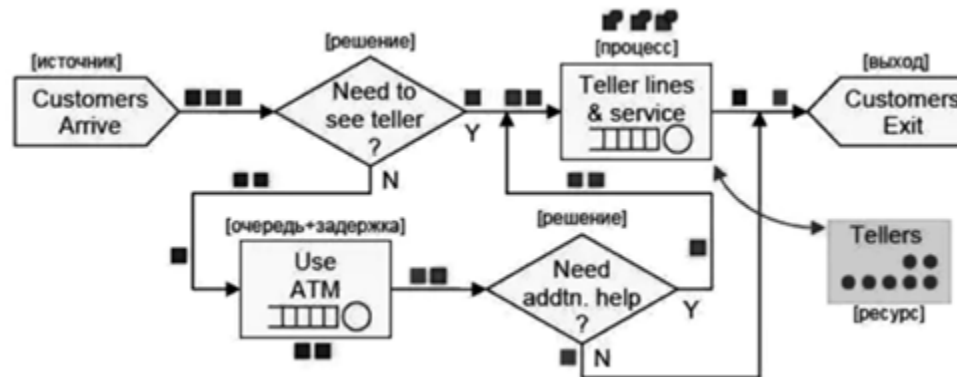
Дискретно-событийное моделирование(в)

В основе дискретно-событийной модели лежит схема обработки заявок, содержащая, как правило, стохастические элементы. Значения переменной время генерируются системным таймером по механизму At , т.е. по существенным моментам времени. Назначение такой модели — сбор количественных характеристик, позволяющих анализировать эффективность функционирования схемы.

Довольно часто термин дискретно-событийное моделирование используется в более узком смысле — процессное моделирование, для которого характерно представление динамики моделируемой системы как последовательности операций над некими пассивными заявками (транзактами, требованиями). Заявками могут быть документы, клиенты, транспортные средства, пакеты данных и т.п., а операциями над заявками — задержка, захват ресурса, разделение и т.п. Заявки в дискретно-событийном моделировании не контролируют свою динамику, но обладают определенными атрибутами, которые могут влиять на процесс их обработки (например, тип звонка, сложность работы) или накапливать статистику. Процессное моделирование используется на среднем или низком уровне абстракции — каждый объект моделируется индивидуально, но обезличенно, т.е. многие характеристики объекта не учитываются.

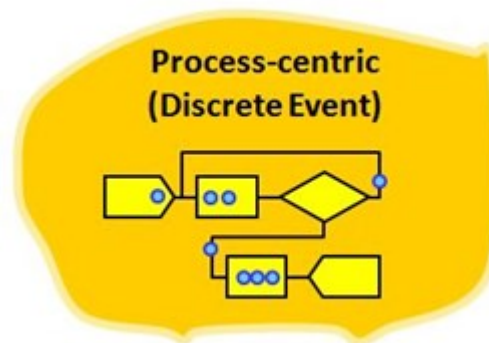
Дискретно-событийное моделирование(г)

Большинство систем имитационного моделирования, поддерживающих дискретно-событийный подход, предлагают пользователю формализацию модели в терминах заявок, ресурсов и потоковых диаграмм. Типичная потоковая диаграмма для дискретно-событийной модели работы отделения банка в терминах системы имитационного моделирования Arena.



Назначение дискретно-событийное моделирования

Чтобы анализировать процессы, протекающие в мире, иногда удобно рассматривать их как последовательность отдельных важных моментов – событий. Дискретное моделирование как раз и позволяет построить имитационную модель, представляя реальные действия событиями.



Компоненты системы дискретно-событийного моделирования(а)

Часы - Основной компонент системы, синхронизирующий изменения системы, т.е. возникновение событий.

Список событий - Система моделирования поддерживает по крайней мере один список событий моделирования. (Однопоточные системы моделирования, основанные на мгновенных событиях, имеют только одно текущее событие. В то время как многопоточные системы моделирования и системы моделирования, поддерживающие интервальные события, могут иметь несколько текущих событий. В обоих случаях имеются серьёзные проблемы с синхронизацией между текущими событиями.)



Компоненты системы дискретно-событийного моделирования(б)

Генератор случайных чисел - Дискретно-событийные модели делятся на детерминированные и стохастические, в зависимости от того, каким образом генерируются события и основные характеристики очередей: время наступления событий, длительность обслуживания, количество клиентов, поступающих в очередь в единицу времени.

Компоненты системы дискретно-событийного моделирования(в)

Статистика - Основные данные, которые собираются в системах дискретно-событийного моделирования: Средняя занятость (доступность) ресурсов, среднее количество клиентов в очереди, среднее время ожидания в очереди.

Условие завершения - Условием завершения могут выступать: Возникновение заданного события (например, достижение 10-минутного времени ожидания в очереди), Прохождение заданного числа циклов по часам системы моделирования.

Некоторые примеры ДСМ

Вот некоторые примеры событий: покупатель вошел в магазин, на складе закончили разгружать грузовик, конвейер остановился, в производство запущен новый продукт, уровень запасов достиг некоего порога и т.д. В дискретно-событийном моделировании движение поезда из точки А в точку Б будет представлено двумя событиями: отправление и прибытие, а само движение становится "задержкой" (интервалом времени) между ними.



Ограничение термина ДСМ

Термин "дискретно-событийное моделирование", однако, обычно используется в более узком смысле для обозначения "процессного" моделирования, где динамика системы представляется как последовательность операций (прибытие, задержка, захват ресурса, разделение и др.) над некими сущностями (entities, по-русски - транзакты, заявки), представляющими клиентов, документы, звонки, пакеты данных, транспортные средства и т.п. Эти сущности пассивны, они сами не контролируют свою динамику, но могут обладать определёнными атрибутами, влияющими на процесс их обработки (например, тип звонка, сложность работы) или накапливающими статистику (общее время ожидания, стоимость). Процессное моделирование используется на среднем или низком уровне абстракции: каждый объект моделируется индивидуально, как отдельная сущность, но множество деталей "физического уровня" (геометрия, ускорения/замедления) опускается. Такой подход широко используется в моделировании бизнес-процессов, производства, логистики, здравоохранения и т.д.



История ДСМ (а)

Дискретно-событийное моделирование обязано своим рождением Джеффри Гордону, который в начале 1960-х спроектировал и реализовал на мэйнфреймах 1 IBM систему GPSS.

Основной объект в этой системе — пассивный транзакт (заявка на обслуживание), который может определенным образом представлять собой работников, детали, сырье, документы, сигналы и т. п. «Перемещаясь» по модели, транзакты становятся в очереди к одноканальным и многоканальным устройствам, захватывают и освобождают эти устройства, расщепляются, уничтожаются и т. д. Таким образом, дискретно-событийную модель можно рассматривать как глобальную схему обслуживания заявок. Аналитические результаты для большого количества частных случаев таких моделей рассматриваются в теории массового обслуживания.



История ДСМ (б)

GPSS (англ. General Purpose Simulation System — система моделирования общего назначения) — язык моделирования, используемый для имитационного моделирования различных систем, в основном систем массового обслуживания.

Основные версии GPSS

GPSS I (1961)

GPSS II (1962)

GPSS III (1965)

GPSS/360 (1967)

GPSS V (1975)

GPSS/H (1978)

GPSS/PC (1984)

Micro-GPSS (1) (1990)

GRAMOS-GPSS (1993)

GPSS World (1993)

Micro-GPSS (2) (1996)

История ДСМ (в)

Недостатки: В программе на языке GPSS достаточно сложно представить непосредственно процессы обработки данных на уровне алгоритмов. Кроме того, модель представляет собой программу, а значит не имеет графической интерпретации, что затрудняет процесс разработки модели и снижает наглядность модели в целом.

AnyLogic, напротив, позволяет легко создавать визуально непрерывные анимации для логически дискретных процессов.



Библиотека PROCESS MODELING LIBRARY (a)

Основное средство процессного моделирования в AnyLogic - это библиотека Process Modeling Library. В эту библиотеку вошли объекты для определения "потока" процесса (process workflow): Source (источник), Sink (выход из системы), Delay (задержка), Queue (очередь), Service (обслуживание), SelectOutput (выбор пути), и т.д., а также задействованных в процессе ресурсов. Все объекты гибкие и настраиваемые: параметры могут изменяться динамически, действия могут зависеть от атрибутов заявок, и т.д.

Библиотека PROCESS MODELING LIBRARY (b)

Объекты имеют "точки расширения" типа onEnter/onExit - это места, где можно определить действия, производимые над заявками при их прохождении через объект.

Компоненты модели, построенные из блоков библиотеки Process Modeling Library могут естественным образом взаимодействовать с компонентами системной динамики, с агентами или с низкоуровневыми примитивами AnyLogic - картами состояний и событиями.



Сетевое моделирование(а)

В Process Modeling Library также входят объекты, разработанные для моделирования процессов, происходящих в пространстве: таких, где объекты-заявки и ресурсы перемещаются в некой сети. Это подмножество объектов значительно упрощает моделирование некоторых типов систем, например, производства, внутризаводской логистики, супермаркета, склада, госпиталя.

Сетевое моделирование(б)

Для использования этого подхода, называемого сетевым моделированием (Network Based Modeling), нужно определить топологию сети, множества ресурсов, и собственно процесс. Процесс в данном случае - это комбинация объектов типа "переместиться" или "присоединить ресурс" и обычных объектов, задающих бизнес-процесс. Заявки и ресурсы автоматически анимируются движущимися по сегментам сети или находящимися в её узлах; эта анимация может также комбинироваться с обычной.

О зачете

На зачет допускается студент сдавший не менее 5 практических работ. И составивший электронные отчеты по практическим, подписавший отчеты у преподавателя и сам, и разместивший их в облаке группы, которое делает староста группы. В отчетах должна быть постановка задачи, скриншоты выполнения задачи и краткое их описание. Проект практической работы защищается либо непосредственно по запуску оногo, либо, по видео запуска проекта. Последний срок сдачи отчетов на этой неделе по расписанию. Студент сдавший 6 практических работ, составивший, подписавший и разместивший отчеты в облаке группы получает зачет автоматом. Иные студенты отвечают на вопросы к зачету.

Вопросы к зачету(а)

Общие вопросы:

1. Основные понятия теории агентов.
2. Общая классификация агентов.
3. Основные отличия одноагентных и многоагентных систем.

Вопросы по выполненным практическим заданиям:

4. Описания различных методов для моделирования агентов в среде AnyLogic.
5. Привести примеры агентов из практических работ.
6. Привести пример трех видов многоагентных систем используемых в практических работах.
7. Привести пример построения многоагентной системы из практической работы.

Вопросы к зачету(б)

Вопросы по выполненным практическим заданиям:

8. Создание популяции агентов на примере практической работы.
9. Модели коллективного поведения агентов на примере практической работы.
10. Изучение коллективного поведения агентов с использованием графиков на примере практической работы.
11. Методы изменения состояния агента на примере практической работы.
12. Методы изменения поведения агента на примере практической работы.
13. Примеры задач, решаемых посредством агентов, на примере практической не типовой работы.

Вопросы к зачету(в)

Вопросы по выполненным практическим заданиям:

14. Показать многоагентную модель, в которой используются агенты различного типа.
15. Может ли агентом являться не человек, доказать на примере.
16. Может ли в многоагентной системе, в множестве агентов использоваться только один агент одного типа, доказать на примере.
17. Как можно показать изменение состояния агента в среде AnyLogic?
18. Какие методы применяются для изменения состояния агента в среде AnyLogic?
19. Какие методы применяется для задания перемещения агента в среде AnyLogic?

Вопросы к зачету(г)

Вопросы по выполненным практическим заданиям:

20. Какие методы применяются для создания 3-D модели в среде AnyLogic?
21. Что такое накопитель и поток при многоагентном моделировании, показать на примере.
22. Что такое формула потока при многоагентном моделировании, показать на примере.
23. Какой метод в среде AnyLogic наиболее подходит для моделирования поведения людей в магазине, объяснить на примере типовой или не типовой практической работы.
24. Как в среде AnyLogic моделируется распространение вируса, объяснить на примере типовой практической работы.



Вопросы к зачету(д)

Вопросы по выполненным практическим заданиям:

25. Как в среде AnyLogic моделируется работа электрокаров, объяснить на примере типовой практической работы.
26. Как в среде AnyLogic моделируется работа фуры, объяснить на примере типовой практической работы.
27. Привести пример многоагентной системы, в которой используются агенты различного типа на примере типовой практической работы.
28. Привести пример многоагентной системы, в которой используются агенты одного типа на примере типовой практической работы.
29. Как может отражаться изменение состояния агента в среде AnyLogic, показать на примерах практических работ.
30. Как в среде AnyLogic моделируется изменения состояния покупателя товара, объяснить на примере типовой практической работы.



Спасибо за внимание!