

УДК 623 ГРЕЧУШКИН Игорь Васильевич¹,
кандидат технических наук,
старший научный сотрудник
igrechuhkin1949@yandex.ru
ФЕДОТОВ Алексей Михайлович¹,
phedotow18@yandex.ru

¹НИИ (ВСИ МТО ВС РФ) ВА МТО
191123, г. Санкт-Петербург, Воскресенская наб., д. 10а,

ПРИМЕНЕНИЕ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ ДЛЯ АНАЛИЗА И СИНТЕЗА СИСТЕМЫ МТО

В статье рассмотрены основные подходы имитационного моделирования анализа и синтеза систем материально-технического обеспечения.

Ключевые слова: материально-техническое обеспечение; имитационное моделирование; дискретно-событийное моделирование; агентное моделирование; система AnyLogic.

Grechushkin I.V., Fedotov A.M.

APPLICATION OF IMITATING MODELLING FOR THE ANALYSIS AND SYNTHESIS OF THE MTO SYSTEM

In article the main approaches of imitating modeling of the analysis and synthesis of systems of material support are considered.

Keywords: material support; imitating modeling; discrete and event modeling; agentny modeling; AnyLogic system.

В связи с тем, что применение аналитического моделирования для анализа и синтеза процессов функционирования системы МТО ограничено сложностью разработки математических моделей перспективным является использование имитационного моделирования [1].

Необходимо отметить, что в настоящее время принципиальные возможности имитационного моделирования весьма велики, так как оно позволяет при надобности исследовать системы любой сложности, назначения и степенью детализации. Ограничениями являются только мощность

используемой компьютерной техники и трудоемкость подготовки сложного комплекса программ. Методы имитационного моделирования развиваются, как правило, в направлении исследования степени подобия имитационных моделей реальным системам и разработки типовых методов и приемов для создания имитационных моделей.

Имитационное моделирование представляет собой процесс конструирования модели реальной системы и постановки экспериментов на данной модели с целью понять поведение системы или оценить (в рамках ограничений, накладываемых некоторым или совокупностью критериев) разные стратегии, которые обеспечивают функционирование этой системы.

Искусство моделирования состоит в способности анализировать проблему, выделять существенные черты и модифицировать ее основные предположения, характеризующие систему и совершенствовать модель до тех пор, пока она не выдаст полезные для практики результаты.

Для разработки модели необходимо выполнить следующие действия:

- 1) разложить общую задачу исследования на ряд более простых задач (детализация);
- 2) сформулировать цели моделирования;
- 3) найти аналоги;
- 4) рассмотреть численный пример, соответствующий данной задаче;
- 5) принять определенные обозначения;
- 6) записать очевидные соотношения, законы, связи;
- 7) если модель поддается математическому описанию, то использовать его в виде формул, уравнений, неравенств.

Моделирование начинается с изучения объекта. На первом этапе формируются законы, управляющие исследованием, происходит отделение информации от реального объекта, формируется существенная информация, происходит первый шаг абстракции. Преобразование информации зависит от решаемой задачи. Информация, существенная для одной задачи, может оказаться несущественной для другой. Потеря существенной информации приводит к неверному решению или не позволяет получить решение. Учет несущественной информации вызывает

лишние сложности, а также создает непреодолимые препятствия на пути к решению. Переход от реального объекта к информации о нем возможен только тогда, когда поставлена задача. Постановка задачи уточняется только по мере изучения объекта.

Таким образом, на первом этапе параллельно идут процессы целенаправленного изучения объекта и уточнения задачи. Также на этом этапе информация об объекте подготавливается к обработке на компьютере. Строится так называемая формальная модель явления, которая содержит:

- набор постоянных величин, констант, которые характеризуют моделируемый объект в целом и его составные части, называемых статистическим или постоянными параметрами модели;

- набор переменных величин, изменяя значение которых можно управлять поведением модели, называемых динамическими или управляющими параметрами;

- формулы и алгоритмы, связывающие величины в каждом из состояний моделируемого объекта;

- формулы и алгоритмы, описывающие процесс смены состояний моделируемого объекта.

На втором этапе формальная модель реализуется на компьютере, выбираются подходящие программные средства для этого, строится алгоритм решения проблемы, создаётся программа, которая реализует этот алгоритм, далее написанная программа отлаживается и тестируется на специально подготовленных тестовых моделях.

Тестирование представляет собой процесс выполнения программы с целью выявления ошибок. Проверить компьютерную модель на соответствие её основным свойствам объекта наиболее эффективно путём использования простых модельных примеров, когда результат моделирования известен заранее.

На третьем этапе осуществляется непосредственно вычислительный эксперимент с использованием компьютерной модели. При этом проводятся исследования поведения модели в различных ситуациях, при тех или иных наборах динамических параметров, осуществляется прогнозирование или оптимизация

характеристик, параметров модели в зависимости от поставленной задачи.

Результатом компьютерного эксперимента будет информационная модель явления, в виде графиков, зависимостей одних параметров от других, диаграмм, таблиц, демонстрации явления в реальном или виртуальном времени и др.

Имитационное (динамическое) моделирование рассматривает модель как совокупность правил, которые определяют, в какое состояние перейдет моделируемый объект из некоторого предшествующего состояния.

При имитационном моделировании сложных систем используются алгоритмы, которые воспроизводят на компьютере процессы их функционирования, при этом имитируются элементарные явления и процессы с сохранением их логической структуры и последовательности протекания во времени.

Различие между аналитическими и имитационными моделями достаточно условное и определяется возможностями математического аппарата.

Следует особо отметить метод кибернетического моделирования, который предложил Н. Винер в виде схемы так называемого «черного ящика». В нем отсутствует непосредственное подобие процессов, происходящих в моделях, реальным физическим процессам. В данном случае рассматривают объект-модель как «черный ящик», имеющий ряд входов и выходов, и моделируют некоторые связи между ними.

Чаще всего при использовании кибернетического метода проводят анализ поведенческой стороны объекта при различных воздействиях внешней среды.

В современном имитационном моделировании сформировались и наиболее широко применяются три основных подхода [2,3,4,5,6,]: дискретно-событийное моделирование, системная динамика и агентное моделирование.

Дискретно-событийное моделирование - подход к моделированию, предлагающий абстрагироваться от непрерывной природы событий и рассматривать только основные события моделируемой системы, такие, как: «ожидание», «обработка заказа», «движение с грузом», «разгрузка» и другие.

Основным объектом в этой системе является 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. Дискретно-событийную модель можно рассматривать как глобальную схему обслуживания заявок. Аналитические результаты для большого количества частных случаев таких моделей рассматриваются в теории массового обслуживания.

Дискретно-событийное моделирование наиболее развито и имеет огромную сферу приложений — от логистики и систем массового обслуживания до транспортных и производственных систем. Основателем данного подхода является Джеффри Гордон, который в 1960-х спроектировал и реализовал на мэйнфреймах IBM систему GPSS.

Сегодня существует целый ряд инструментов, поддерживающих такой подход в моделировании: GPSS/PC, GPSS/H, GPSS World, Object GPSS, Arena, SimProcess, Enterprise Dynamics, Auto-Mod и др.

GPSS World—типичный современный представитель GPSS-семейства, реализованный для работы в среде MS Windows. Наличие встроенных инструментов статистической обработки результатов моделирования, встроенного языка программирования расчетов PLUS и др. позволяет создавать средствами GPSS World различные модели, в том числе систем массового обслуживания.

Системная динамика была предложена как методология в 1961 году Дж. Форрестером как инструмент исследования информационных обратных связей в производственно-хозяйственной деятельности, для того чтобы выяснить, каким образом взаимодействуют организационная структура, усиления и задержки (в принятии решений и действиях), оказывая влияние на эффективность предприятия. Процессы, происходящие в реальном мире, в системной динамике представляются в терминах накопителей (фондов) и потоков между ними. Системно-динамическая модель описывает поведение системы и

ее структуру как множество взаимодействующих обратных положительных и отрицательных задержек и связей.

Системная динамика – парадигма моделирования, где для исследуемой системы строятся графические диаграммы причинных связей и глобальных влияний одних параметров на другие во времени, а затем созданная на основе этих диаграмм модель имитируется на компьютере. Данный вид моделирования более всех других парадигм помогает понять суть происходящего выявления причинно-следственных связей между явлениями и объектами. С помощью системной динамики строятся модели бизнес-процессов, развития города, модели производства, динамики популяции, экологии и развития эпидемии. Такая модель математически представляется в виде системы дифференциальных уравнений.

Методы системной динамики поддерживаются следующими инструментами: DYNAMO, Stella, Vensim, PowerSim, iThink, ModelMaker и др.

Пакет Vensim представляет собой инструмент для визуального моделирования, поддерживающий разработку концептуальной модели, документирование, моделирование, анализ результатов и оптимизацию моделей динамических систем. Пакет имеет графический редактор для построения с помощью мыши классических моделей, Equation Editor для завершения формирования модели, а также развитые средства визуализации поведения модели.

Агентное моделирование - относительно новое (1990-2000-ые гг.) направление в имитационном моделировании, которое используется для исследования децентрализованных систем, динамика функционирования которых, определяется не глобальными правилами и законами (как в других парадигмах моделирования), а наоборот, когда эти глобальные правила и законы являются результатом индивидуальной активности участников процесса. В этой модели нет единой точки, определяющей поведение системы в целом. Цель агентных моделей — получить представление об этих глобальных правилах, общем поведении системы, на основе предположений об индивидуальном, частном поведении её отдельных активных объектов и взаимодействии этих объектов в системе. Агент —

некая сущность, обладающая активностью, автономным поведением, может принимать решения в соответствии с некоторым набором правил, взаимодействовать с окружением, а также самостоятельно изменяться. Для описания поведения агентов используются карты состояний, являющиеся стандартным инструментом UML.

Для систем, содержащих большое количество активных объектов с отчетливо выраженным индивидуальным поведением, агентное моделирование является более универсальным подходом, т. к. оно позволяет учитывать структуру и поведение системы любой сложности.

Другим важным достоинством агентного моделирования является возможность разработки модели даже в отсутствие априорной информации о глобальных зависимостях. Зная индивидуальную логику поведения участников процесса, можно построить агентную модель и спрогнозировать ее глобальное поведение. Также агентная модель проще в сопровождении, поскольку уточнения вносятся на локальном уровне по мере накопления данных.

Концепция агентного моделирования позволяет осуществить переход от моделей системной динамики и дискретно-событийных моделей к агентным моделям с помощью процедуры конвертации. Для системно-динамических моделей может потребоваться деагрегация накопителей на множества агентов (при условии активности и различимости этих агентов). В дискретно-событийных моделях индивидуальные объекты - транзакты также могут стать агентами (при условии децентрализации логики прохождения транзактов).

Проведённый анализ рассмотренных подходов в имитационном моделировании показывает, что аппарат системной динамики обычно оперирует непрерывными во времени процессами, а дискретно-событийное и агентное моделирование чаще всего используются для дискретных во времени процессов. С другой стороны, системная динамика предполагает максимальный уровень абстракции модели, дискретно-событийное моделирование отражает абстракции низкого и среднего уровня. Агентное моделирование может применяться на любом уровне модели любого масштаба (рис.1).

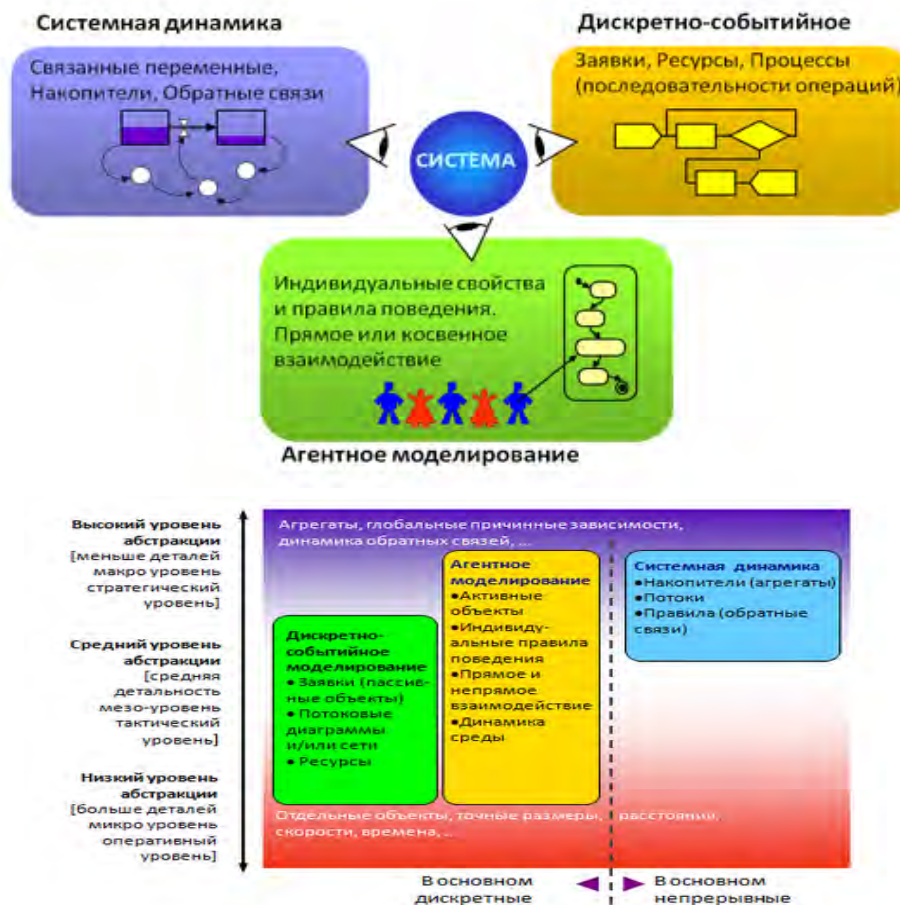


Рисунок 1- Принципы и уровень абстракции системной динамики, дискретно-событийного и агентного моделирования

Анализ рассмотренных методов имитационного моделирования показывает, что для имитационного моделирования систем МТО наиболее эффективными являются дискретно-событийное и агентное моделирование.

В работе [7] проведён анализ преимуществ и недостатков дискретно-событийного (процессного) и агентного подходов к имитационному моделированию. Результаты анализа представлены в таблице 1.

Для решения задачи выбора подхода к моделированию в данной работе был выполнен их сравнительный анализ по наиболее важным критериям: сложность проектирования модели и вычислительная сложность реализации процессов функционирования модели, обуславливающие время моделирования.

Таблица 1- Сравнительная оценка процессного и агентного моделирования

Критерий	Моделирование	
	процессное	агентное
Методология проектирования модели	От процесса к объектам	От объектов к процессам
Децентрализация процессов и событий	По процессам (объектам обслуживания)	По объектам-агентам (заявкам, устройствам)
Уровень децентрализации	Высокий	Низкий
Сложность проектирования и сопровождения (статич./динамич.)	Низкая Высокая	Средняя
Вычислительная сложность	Высокая	Средняя

Сложность проектирования моделей традиционно определяется необходимостью использования дополнительных объемов программного кода и, как следствие, увеличением времени проектирования (время прямо пропорционально сложности).

Для процессных моделей дополнительный код применяется в редких случаях, поскольку большинство возможных событий и их реализаций уже предусмотрено системными разработчиками в программных блоках моделей, а проектировщику модели требуется выбрать дисциплины обслуживания, указать параметры процесса обслуживания и события системы. Для более «сложных» процессных моделей проектировщику требуется работать с классами заявок и методами работы со списками заявок.

В отличие от процессной модели агентная модель характеризуется практической сложностью реализации. Обусловлено это необходимостью реализации с помощью диаграмм состояний алгоритмов постановки, выборки, сортировки агентов в очереди-состоянии, программной реализации дисциплин функционирования и выбора каналов обслуживания. Агентные модели требуют программной реализации элементов сбора статистики, в то время как для

объектов диаграммы процессов разработаны «системные» методы сбора вероятностно-временных характеристик. Программная реализация агентных моделей требует от разработчика углубленных знаний объектно-ориентированного программирования, методов работы с диаграммами состояний, и поэтому сложность проектирования агентных систем выше, чем процессных. Предлагаемая авторами оценка ограничена сверху высоким уровнем моделирования систем при реализации не в системах моделирования, а с помощью языков программирования, например, C++. Обратной стороной относительной сложности проектирования агентных моделей является простота ее сопровождения, поскольку уточнения вносятся на локальном уровне по мере накопления данных об объекте.

Распространенная оценка вычислительной сложности является функцией объема работы, выполняемой алгоритмом моделирования в зависимости от исходных данных. Объем работы определяется временем и вычислительными ресурсами (объемом затраченной в расчетах оперативной памяти). Анализ структуры реализации обслуживающих элементов процессной модели на примере AnyLogic показывает для каждого процессного блока большое число (более 20) переменных (параметров), что свидетельствует о высоких затратах вычислительных ресурсов на организацию процесса в системе. Такой же объем процессов в агентном исполнении занимает гораздо меньший объем памяти. Объемы памяти, которые требуется для реализации заявок в процессной и агентной средах, одинаковы, так как практически отсутствуют отличия в структурах заявок.

Ряд экспериментов, проведенный с простыми и сложными моделями, показывает практическое равенство времени моделирования процессного и агентного подходов.

Таким образом, используя предложенные критерии и результаты анализа основных этапов имитационного моделирования, разработчик может определить метод моделирования. В более общем смысле, полученные результаты объясняют рациональность известных высказываний [3], [5]: «При выборе между процессным (дискретно-событийным) и

агентным моделированием надо иметь в виду, что агентные модели обычно более трудоемки в построении и калибровке, поэтому, если система хорошо описывается в виде последовательности операций над (пассивными) объектами, разделяющими ресурсы, нет смысла строить агентную модель»; «Агентное моделирование является эффективным при моделировании систем, содержащих большие количества активных объектов и динамические взаимосвязи между ними».

Кроме того, отметим, что наиболее эффективными являются инструменты, объединяющие в себе лучшие черты исследуемых подходов к моделированию [4]. Уже сегодня известен пример такой мощной системы имитационного моделирования – AnyLogic, предоставляющей разработчику

возможность взаимоувязанного использования в рамках одной модели дискретно-событийного («процессного») и агентного подходов.

В настоящее время наиболее перспективным программным продуктом, который используется для имитационного моделирования, является AnyLogic. Продукт получил название AnyLogic, потому что он поддерживал все три известных метода моделирования (системную динамику; дискретно-событийное (процессное) моделирование; агентное моделирование), а также любую комбинацию этих подходов в пределах одной модели.

С помощью AnyLogic стало возможным разрабатывать модели во многих областях, в том числе и обороне. Инструмент обладает современным графическим интерфейсом и позволяет использовать язык Java для разработки моделей.

Последней версией программы является AnyLogic 7, которая написана на языке программирования Java в популярной среде разработки Eclipse.

AnyLogic включает в себя графический язык моделирования, а также позволяет пользователю расширять созданные модели с помощью языка Java. Интеграция компилятора Java в AnyLogic предоставляет более широкие возможности для создания моделей, а также создание Java апплетов, которые возможно открыть в любом браузере. Эти апплеты позволяют размещать модели AnyLogic на веб-сайтах. Также помимо Java-апплетов, AnyLogic Professional поддерживает создание Java-приложений, в

этом случае пользователь сможет запускать модель без инсталляции AnyLogic.

Среда моделирования AnyLogic поддерживает проектирование, разработку, документирование модели, выполнение компьютерных экспериментов с моделью, включая различные виды анализа — от анализа чувствительности до оптимизации параметров модели относительно некоторого критерия.

Таким образом, проведенный анализ существующих подходов к имитационному моделированию показывает, что для моделирования процессов анализа и синтеза систем МТО целесообразно использовать дискретно-событийное и агентное моделирование в среде AnyLogic.

Библиографические ссылки

1. Р. Шеннон Имитационное моделирование систем - искусство и наука.- М.: Мир, 1978- 420 с.
- 2 Румянцев М. Средства имитационного моделирования бизнес процессов // Корпоративные системы. – 2007.- №2 .- С.6- 10.
- 3 Борщев А. В. Практическое агентное моделирование и его место в арсенале аналитика // Exponenta PRO.- 2004. - № 3–4 (7–8). - С. 38–47.
- 4 Девятков В. В. Методология и технология имитационных исследований сложных систем: современное состояние и перспективы развития. М.: Вузовский учебник; ИНФРА-М, 2014. 448 с.
- 5 Борщев А. В. Применение имитационного моделирования в России – состояние на 2007 г. // сб. докл. III Всерос. науч.-практ. конф. Имитационное моделирование. Теория и практика» (ИММОД-2007),СПб., 17–19 окт. 2007. Т. I. С. 11–16.
- 6 Wai Kin Victor Chan, Young-Jun Son, Charles M. Macal, Agent-based simulation tutorial – simulation of emergent behavior and differences between agentbased simulation and discrete-event simulation // Proc. of the Winter Simulation Conf., Baltimore, Maryland, Dec. 05-08. 2010.
- 7 Даденков С.А., Кон Е.Л. Анализ моделей и методов агентного и дискретно-событийного имитационного моделирования // Известия СПб ГЭТУ 2 «ЛЭТИ» - 2015.- №5