

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ГРАЖДАНСКОЙ АВИАЦИИ



Н.И. Черкасова

МОДЕЛИРОВАНИЕ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ

Учебное пособие

Москва
2019

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ВОЗДУШНОГО ТРАНСПОРТА

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ГРАЖДАНСКОЙ АВИАЦИИ (МГТУ ГА)»**

Кафедра вычислительных машин, комплексов, систем и сетей

Н.И. Черкасова

МОДЕЛИРОВАНИЕ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

Учебное пособие

Утверждено Редакционно-
издательским советом МГТУ ГА
в качестве учебного пособия

Москва
2019

УДК

ББК 6Ф7.3

Ч-48

Печатается по решению редакционно-издательского совета
Московского государственного технического университета ГА

Рецензенты:

Надейкина Л.А. (МГТУ ГА) – канд. физ.-мат. наук, доцент;

Соловьев В.Р. (МФТИ) – канд. физ.-мат. наук, доцент

Черкасова Н.И.

Ч-48 Моделирование вычислительных систем: Учебное пособие. /Н.И. Черкасова. —
Воронеж: ООО «МИР», 2019. — 80 с.

ISBN

Учебное пособие издается в соответствии с рабочей программой учебной дисциплины «Моделирование вычислительных систем» по учебному плану для студентов III курса направления 09.03.01 всех форм обучения.

В учебном пособии представлены материалы курса, описывающие Сети Петри и системы имитационного моделирования, содержит материалы, предназначенные для освоения профессиональных компетенций.

Рассмотрено и одобрено на заседании кафедры 18.06.2019 г. и методического совета 18.06.2019 г.

В авторской редакции.

ББК 6Ф7.3

Св. тем. план 2019 г.

поз. 26

ЧЕРКАСОВА Наталья Ивановна

МОДЕЛИРОВАНИЕ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ

Учебное пособие

Подписано в печать 08.07.2019 г.

Формат 60x80/16 Печ. л. 3 Усл. печ. л. 2,79

Заказ 513/ Тираж 30 экз.

Московский государственный технический университет ГА

125993 Москва, Кронштадтский бульвар, д.20

Отпечатано ООО «МИР»

394033, г. Воронеж, Ленинский пр-т 119А, лит. Я, оф. 215

ISBN

© Московский государственный
технический университет ГА, 2019

Содержание

	Стр.
1. Аналитическое моделирование.	4
1.1. Понятие «Модель»	4
1.2. Натурное моделирование	8
1.3. Классификация моделей	10
2. Аналитическое моделирование на основе систем массового обслуживания (СМО).	34
3. Моделирование компьютерных систем.	37
3.1. Аналитическое моделирование сетей	38
3.2. Аналитическое моделирование сетей на основе систем массового обслуживания	39
4. Сети Петри.	45
4.1. Свойства сетей Петри.	49
4.2. Анализ сетей Петри.	49
4.3. Подклассы и расширения сетей Петри.	51
4.4. Раскрашенные сети Петри.	53
5. Имитационное моделирование.	58
5.1. Понятие имитационного моделирования	58
5.2. Преимущества и недостатки имитационного моделирования.	61
6. Системы имитационного моделирования	63
Литература	80

1. Аналитическое моделирование.

В наше время, в эпоху научно-технического прогресса, в процессе которой человечество идет по пути использования все более усложняющихся технических средств, а также использования сложных природных объектов или систем, все в той или иной степени сталкиваются с понятием системы, автономные объекты и объекты, связанные в системы, независимо от характера своего образования - гуманитарного или технического.

Следует отметить важность умения правильно распознавать системы и правильно использовать это понятие. Очевидно, что всякая система - это сложный объект, который чаще всего нельзя свести к простой совокупности частей благодаря наличию комплекса взаимодействий, объединяющих разные части системы в одно целое. В некоторых случаях отдельные части оказываются связаны друг с другом не только непосредственно, но и с помощью опосредованных связей. Более того, некоторые части системы могут вообще не иметь друг с другом непосредственных связей, но подвергаться воздействию друг на друга через посредников. Тогда, как нетрудно сделать вывод, поведение отдельных частей системы может приобретать новое качество, отсутствовавшее у таких частей по отдельности. В целом же система может проявить такое поведение, предсказать которое обычно невозможно путем простой логической экстраполяции известных свойств отдельных частей, составляющих эту систему. Такое свойство систем именуется специфическим термином «эмерджентность» или более широко известным словосочетанием «системный эффект». Представим более общее определение - Системный эффект (эмерджентность) - возникновение у сложной системы новых свойств за счет взаимодействия составляющих систему узлов [1,2].

Таким образом, умение предсказать и правильно использовать системный эффект становится в нашу эпоху наиболее важным для специалиста практическим навыком, а по мере усложнения систем, задача определения системного эффекта, становится все более актуальной.

Рассмотрим основные понятия и определения «Моделирования систем»

1.1. Понятие «Модель»

Безусловно существуют различные формулировки и определения этих основных понятий, однако, все они базируются на общих понятиях, рассмотренных ниже.

Моделирование - исследование объектов познания на их моделях; построение и изучение моделей реально существующих объектов, процессов или явлений с целью получения объяснений этих явлений, а также для предсказания явлений, интересующих исследователя.

Моделирование определяется следующими тремя основными понятиями:

- *объект* –реальный предмет, процесс, который требуется изучить или описать;
- *система* – объект, процесс в котором участвующие элементы связаны определенными связями и отношениями;
- *модель* – результат отображения исследуемого объекта, системы в известных компонентах и правилах.

Итак, было введено общее понятие модели, как представление (отображение) исходного объекта в рамках заданных форм и правил. Рассмотрим более подробно некоторые из развернутых определений данного понятия.

Модель – представление объекта, системы или понятия в некоторой форме, отличной от реального существования, для его (ее) замещения другой системой с целью изучения оригинала или воспроизведения его каких - либо свойств.

Модель может быть точной копией какого-либо объекта (хотя и в другом масштабе и из другого материала) или сохранять лишь часть значимых при постановке задачи его свойств. Во втором случае модель является абстракцией объекта.

Когда модель полностью дублирует объект, то она ему изоморфна. Абстрактные модели являются гомоморфными по отношению к объекту.

Большинство моделей лишь гомоморфны, т.е. являются упрощенным представлением реального мира или абстракцией. Тем не менее, они отвечают на поставленные в рамках той или иной задачи вопросы.

Модель обладает определенными свойствами. Модель должна быть:

- адекватной, т.е. в полной мере в соответствии с постановкой задачи воспроизводить оригинал;
- простой и понятной для пользователя;
- целенаправленной, но надежной в смысле отсутствия неправомерных ответов;
- адаптивной, т.е. иметь возможность переходить к другим модификациям или обновлять данные, или совершенствоваться во взаимодействии с пользователем;
- удобной в управлении или иметь «дружественный» интерфейс;

- полной, с точки зрения решения выделенных или главных задач.

Модель должна реализовывать определенный набор функций. Однако, четкого определения функций модели нет, так как представления системы с помощью модели носит общий характер. Тем не менее, можно определить пять основных направлений функций.

- Основной функцией будет, безусловно, - прогнозирование поведения моделируемых объектов.
- Модели могут упорядочить нечеткие или противоречивые понятия (например, при сетевом планировании некоторого производственного процесса).
- Все языки, в основе которых лежит слово, оказываются неточными, при описании сложных понятий и определений. Речь идет о том, что рассматриваемая система более компактно представляется в форме принятых обозначений, блоков, уравнений и т.д., нежели в словесном описании (например, - уравнение теплопроводности).
- Модели часто применяются как средство обучения управления случайностями до возникновения критической ситуации (модели космических кораблей, тренажеры для обучения водителей и др.).
- И наконец, - модели позволяют производить контролируемые эксперименты в ситуациях, где экспериментирование на реальных объектах экономически нецелесообразно или практически невозможно. Здесь обычно варьируют несколько параметров системы, поддерживая остальные неизменными, и наблюдают результаты эксперимента. Часто, моделируя систему, можно узнать значительно больше о ее внутренних взаимосвязях, чем оперируя с реальной системой.

Основой моделирования является эксперимент. Эксперименты необходимы в любой науке для установления закономерностей. Здесь еще нет моделирования, но если в процессе проведения эксперимента удалось установить, что повторение условий его проведения приводит к повторению результата, то сам этот факт дает основу для прогнозирования результата в будущем.

Моделирование на стадии эксперимента происходит при попытке выявления зависимостей результата от наиболее существенных факторов. Это позволяет обобщить результат экспериментирования в виде некоторой математической формулы. Например, для свободно падающих тел Галилеем был установлен факт независимости динамики свободного падения тел в вакууме вблизи поверхности земли от их массы, плотности, размеров. Это обстоятельство может быть проверено путем измерения зависимости скорости свободного падения от времени для контрольной группы тел. Если нанести результаты измерений на график $V(t)$, то через совокупность экспериментальных точек с учетом погрешностей измерений для каждого

тела легко может быть проведена прямая (рис.1.), что позволяет выполнить аппроксимацию по формуле (уравнение прямой):

$$V(t) = V_0 + g \cdot t \quad (1)$$

С помощью статистической обработки с некоторой погрешностью может быть найден тангенс угла наклона данной прямой, то есть величина g , являющаяся по смыслу ускорением свободного падения. Сопоставление величины ускорения для разных тел позволяет повторить вывод Галилея. Таким образом, устанавливается некоторое соотношение, выражаемое формулой (1).

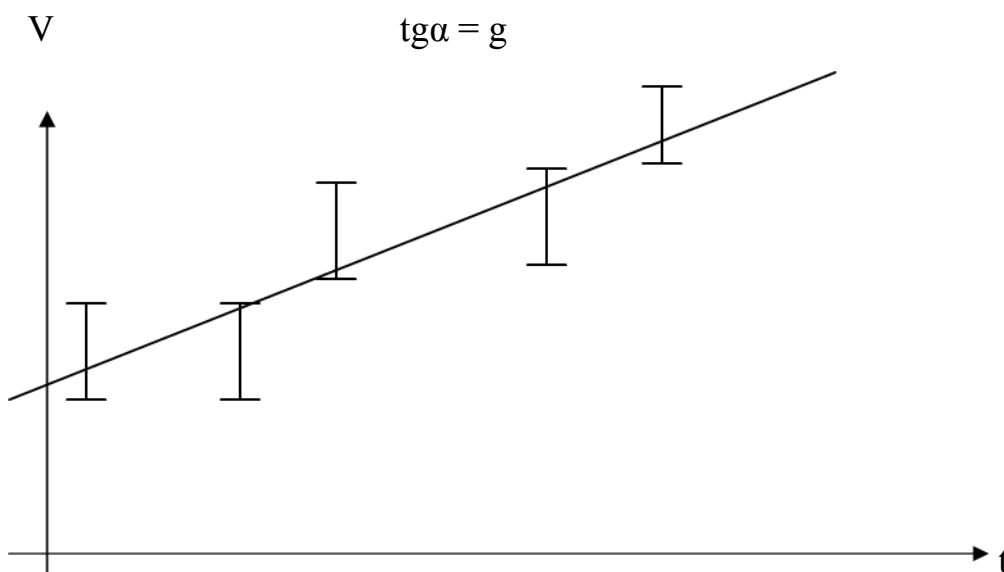


Рис. 1. Зависимость V от t

И хотя природа этого явления так и осталась непонятна, но, имея формулу (1), мы обладаем инструментом прогнозирования.

Рассмотрим известную информационную технологию «черного ящика», получившей широкое распространение в кибернетике. «Черный ящик», соответствующий формуле (1), схематично можно изобразить так, как это показано на рис. 2.

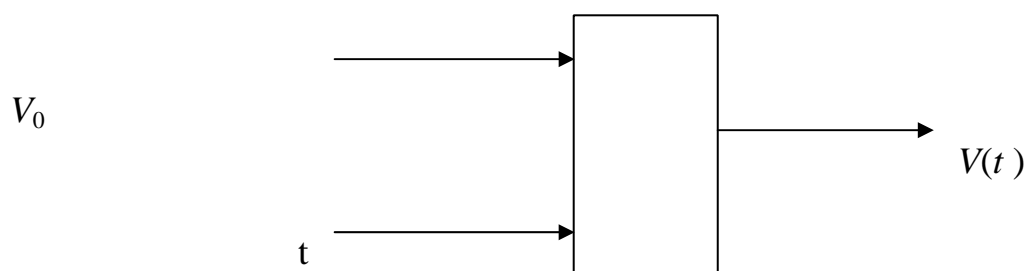


Рис. 2. Модель «черный ящик»

Рассмотрим, как исторически возникало большинство физических «законов». Закон – это на самом деле результат абсолютизации математической зависимости, найдя которую редко удается понять природу явления. Абсолютизация, тем не менее, никогда не является достаточно достоверной. Например, достоверно известно, что закон Кулона справедлив вплоть до атомных размеров, но нарушается на расстояниях, сопоставимых с межнуклонными размерами внутри атомного ядра (т.е. $\sim 10^{-14}$ м).

Таким образом, все известные естественнонаучные законы на самом деле являются, в лучшем случае, всего лишь моделями реальных явлений. Так происходит потому, что реальные объекты всегда сложнее наших представлений о них. С одной стороны, это ограничивает, но с другой стороны это освобождает от необходимости придерживаться строго выбранной модели. На самом деле практически всегда одна модель может быть заменена другой достаточно эффективной моделью.

1.2. Натурное моделирование

Распространённым видом моделирования является физическое моделирование (натурное моделирование, макетирование). Оно состоит в замене изучения некоторого объекта или явления его эквивалентным аналогом, имеющим схожую физическую природу.

Например, рассмотрим движение математического маятника, описываемое вторым законом Ньютона, и колебания в линейном LC-контуре (рис. 3).

Движение математического маятника в проекции на направление скорости груза V описывается уравнением:

$$m \cdot \frac{dV}{dt} = m \cdot g \cdot \sin\phi, \quad (2)$$

где m – масса маятника; g – ускорение свободного падения; ϕ – угол отклонения маятника. Добавим к нему почти очевидное кинематическое соотношение, с учетом знаков выбранных переменных состояния:

$$V = -l \cdot \omega = -l \cdot \frac{d\phi}{dt}, \quad (3)$$

где l – длина подвеса; ω – угловая скорость груза относительно точки подвеса.

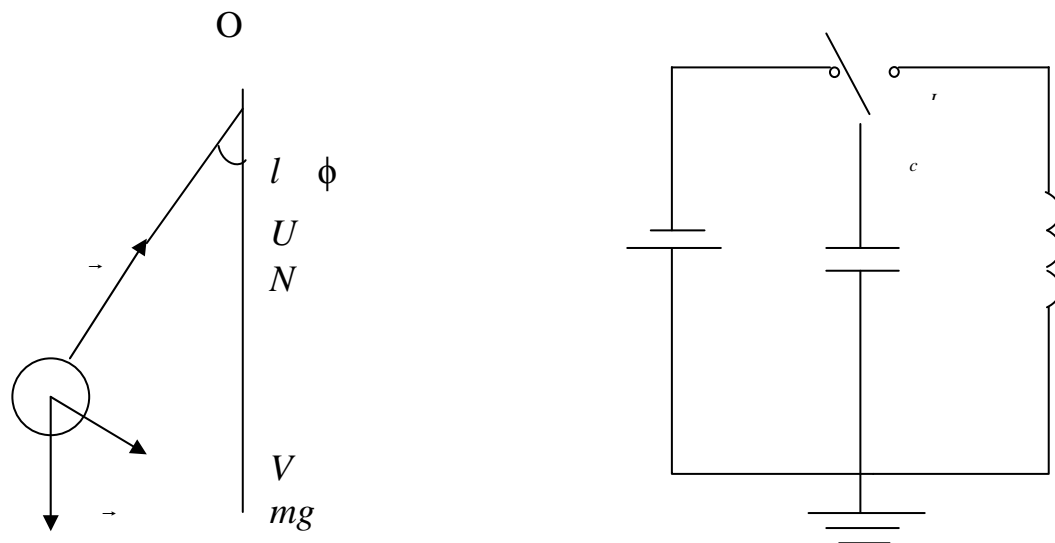


Рис. 3. Пример физического моделирования

С другой стороны колебания в LC-контуре после переключения ключа и разрядки конденсатора на индуктивность можно описать, учитывая, что сумма токов через конденсатор и индуктивность равна нулю. Это позволяет воспользоваться уравнением, где производная от тока через индуктивность взята с целью использовать в дальнейшем закон Фарадея:

Таким образом, с математической (описательной) точки зрения, колебания в линейном LC контуре полностью эквивалентны колебаниям математического маятника при малых отклонениях от положения равновесия. Вместо изучения колебаний в одной системе можно исследовать колебания на примере (макете) другой системы.

Аналогичные примеры моделирования широко известны в областях авиастроения, кораблестроения, при проектировании плотин и архитектурных сооружений, ракетостроении, при испытании вооружений и т.п.

Главная причина применения такого моделирования – в сокращении затрат по изготовлению и отладке производства изделий. Сокращение достигается в трудозатратах, затратах ресурсов, затратах времени. По существу, по этому пути идут также все технологии, развивающие модельные ряды и ряды прототипов. Это автомобилестроение, химия, металлургия, производство вычислительной техники и т.п. Здесь каждый прототип может рассматриваться как модель будущего изделия.

Обобщая информацию о видах моделирования, нетрудно заметить, что все они в той или иной мере связаны с математикой, то есть имеют непосредственное отношение к математическому моделированию. При этом наряду с умением правильно применять математические модели не менее актуально умение правильно ставить задачи математического моделирования.

Далее рассмотрим и другие типы моделирования.

1.3. Классификация моделей

Описанные выше определения можно представить в другом контексте.

Так модель описывается, как любой образ, аналог, мысленный или установленный, изображение, описание, схема, чертеж, и т. п. какого-либо объекта, процесса или явления, который в процессе познания (изучения) замещает оригинал, сохраняя некоторые важные для данного исследования типичные свойства.

В этом же ключе, представим моделирование, как исследование какого-либо объекта или системы объектов путем построения и изучения их моделей. А также - это использование моделей для определения или уточнения характеристик и рационализации способов построения вновь конструируемых объектов.

Рассмотрим и другое определение. Модель (фр. *modèle*, от лат. *modulus* — «мера, аналог, образец») - некоторый материальный или мысленно представляемый объект или явление, являющийся упрощённой версией моделируемого объекта или явления (прототипа) и в достаточной степени повторяющий свойства, существенные для целей конкретного моделирования (опуская несущественные свойства, в которых он может отличаться от прототипа).

Модели обычно применяются для нужд познания (созерцания, анализа и синтеза) и конструирования. В качестве модели может выступать отображение, схема, копия, макет, изображение.

Моделью может быть серийный повторяемый проект, имеющий набор определённых, свойственных только данной модели параметров и характеристик. Это делается даже в одном ряду изделий (проектов). Модель решений может иметь несколько версий или вариантов, что является моделированием деятельности, проектирования, управления большими проектами и т. п.

Очевидно, что модель является средством для изучения прежде всего сложных систем.

В общем случае сложная система представляется как многоуровневая конструкция из взаимодействующих элементов, объединяемых в подсистемы различных уровней. К сложным системам, в т.ч., относятся информационные системы. Проектирование таких сложных систем осуществляется в два этапа.

1 *Внешнее проектирование:*

На этом этапе проводят выбор структуры системы, основных ее элементов, организация взаимодействия между элементами, учет воздействия внешней среды, оценка показателей эффективности системы.

2 Внутреннее проектирование - проектирование отдельных элементов системы:

Типичным методом исследования сложных систем на первом этапе является моделирование их на ЭВМ.

В результате моделирования получаются зависимости, характеризующие влияние структуры и параметров системы на ее эффективность, надежность и другие свойства. Эти зависимости используются для получения оптимальной структуры и параметров системы.

Модель, сформулированная на языке математики с использованием математических методов, называется математической моделью.

Для имитационного моделирования характерно воспроизведение явлений, описываемых математической моделью, с сохранением их логической структуры, последовательности чередования во времени. Для оценки искомых величин может быть использована любая подходящая информация, циркулирующая в модели, если только она доступна регистрации и последующей обработке.

Искомые величины при исследовании процессов методом имитационного моделирования обычно определяют, как средние значения по данным большого числа реализаций процесса. Если число реализаций N , используемых для оценки искомых величин, достаточно велико, то в силу закона больших чисел получаемые оценки приобретают статистическую устойчивость и с достаточной для практики точностью могут быть приняты в качестве приближенных значений искомых величин.

Сущность метода имитационного моделирования применительно к задачам массового обслуживания состоит в следующем. Строятся алгоритмы, при помощи которых можно вырабатывать случайные реализации заданных потоков однородных событий, а также моделировать процессы функционирования обслуживающих систем. Эти алгоритмы используются для многократного воспроизведения реализации случайного процесса обслуживания при фиксированных условиях задачи. Получаемая при этом информация о состоянии процесса подвергается статистической обработке для оценки величин, являющихся показателями качества обслуживания

Процесс моделирования догматически не формализован, поэтому наибольшая роль в этом процессе принадлежит опыту разработчика. Но процесс создания модели может стать основой для начинающих разработчиков и по мере накопления опыта может быть индивидуализирован.

Математическая модель, являясь абстрактным образом моделируемого объекта или процесса, не может быть его полным аналогом. Достаточно сходства в тех элементах, которые определяют цель исследования. Для качественной оценки сходства вводится понятие адекватности модели объекту и в связи с этим раскрываются понятия изоморфизма и изофункционализма. Рекомендаций, позволяющих автоматически, создавать адекватные математические модели, нет, хотя некоторыми авторами предпринимались попытки по разработке различных методик. При этом, очевидно, что окончательное суждение об адекватности модели дает практика, т.е. сопоставление модели с действующим объектом.

Наличие большого количества признаков и зависимостей для моделей, предполагает, что можно составить различные виды классификаций в зависимости от одного или нескольких признаков, общих для той или иной группы моделей.

Наиболее распространенные виды классификации моделей определяются различными признаками. Рассмотрим некоторые из них.

Признаки классификации моделей:

1. · по области использования;
2. · по фактору времени;
3. · по отрасли знаний;
4. · по форме представления;
5. · по признаку реализации и т. д.

Классификация моделей по области использования:

- *Учебные модели* - используются при обучении.
- *Опытные* - используют для исследования и прогнозирования его будущих характеристик.
- *Научно - технические* - создаются для исследования процессов и явлений.
- *Игровые* – репетиция поведения объекта в различных условиях.
- *Имитационные* – отражение реальности в той или иной степени.

Учебные модели используются при обучении. Это могут быть наглядные пособия, различные тренажеры, обучающие программы.

Опытные модели- это уменьшенные или увеличенные копии проектируемого объекта. Их называют также *натурными* и используют для исследования объекта и прогнозирования его будущих характеристик. Например, модель крыла самолета «продувается» в аэродинамической трубе с целью исследования его обтекаемости; модель гидростанции еще при

разработке проекта помогает решить гидротехнические, экологические и многие другие проблемы.

Научно-технические модели создаются для исследования процессов и явлений. К таким моделям можно отнести прибор для получения грозового электрического разряда, модель движения планет Солнечной системы, модель работы двигателя внутреннего сгорания.

Игровые модели — это различного рода игры: деловые, экономические, военные. С помощью таких моделей можно разрешать конфликтные ситуации, оказывать психологическую помощь, проигрывать поведение объекта в различных ситуациях.

Имитационные модели не просто отражают реальность с той или иной степенью точности, а имитируют ее. Эксперимент с моделью либо многократно повторяется при разных исходных данных, чтобы изучить и оценить последствия каких-либо действий на реальную обстановку, либо проводится одновременно со многими другими похожими объектами, но поставленными в разные условия. По результатам исследования делаются выводы. Подобный метод выбора правильного решения называется *методом проб и ошибок*. К примеру, в ряде опытов на мышах испытывается новое лекарственное средство, чтобы выявить побочные действия и уточнить дозировки.

Отметим, что проводить *реальные* эксперименты в различных областях чаще всего практически невозможно, поэтому прибегают к имитационному моделированию.

Классификация с учетом временного фактора (динамики)

Статические модели отражают объект в определенный момент времени, без учета происходящих с ним изменений. В этих моделях отсутствует временной фактор

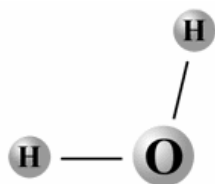


Рис.4 Модель молекулы воды.

Примером статической модели может служить макет или рисунок молекулы воды, состоящей из атомов водорода и кислорода (Рис).

Динамические модели отражают процесс изменения объекта во времени.

Химический опыт, проводимый в лаборатории, является примером динамической модели. Известно, что кислород реагирует с водородом с

выделением энергии. Поэтому даже при малых количествах исходных веществ реакция сопровождается громким хлопком. Таким образом, модель, позволяет предсказать возможные последствия соединения этих распространенных в природе и жизненно необходимых человеку веществ.

Один и тот же объект возможно изучать, применяя и статическую и динамическую модели. Например, при строительстве дома рассчитывают прочность и устойчивость к постоянной нагрузке его фундамента, стен, балок - это статическая модель здания. Но еще надо обеспечить противодействие ветрам, движению грунтовых вод, сейсмическим колебаниям и другим изменяющимся во времени факторам. Это можно решить с помощью динамических моделей.

Классификация по отрасли знаний

Здесь можно выделить следующие виды моделей: **физические** (например, законы Ньютона); **химические** (например, перегонка нефти); **географические** (интерактивные географические карты); **исторические** (генеалогическое дерево); **социологические**; **экономические** (рис.5), **математические** (например, математическая модель полета снаряда) и т.п.

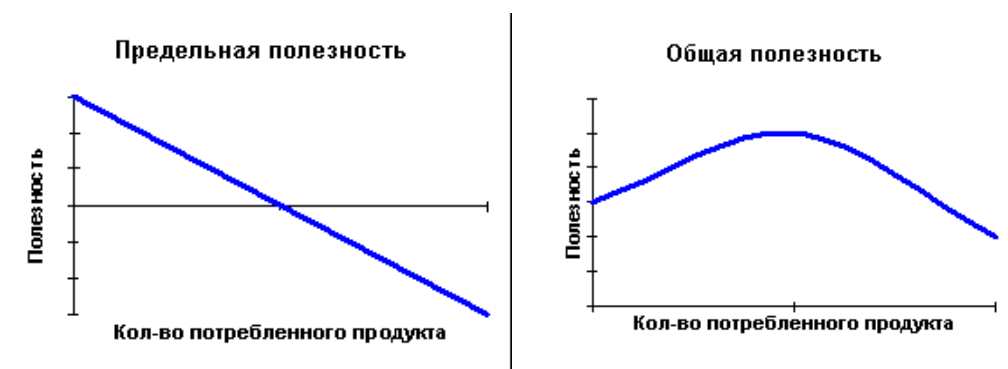


Рис. 5. Экономическая модель

Классификация по способу представления

Классификация моделей по способу представления представлена далее.

В соответствии с этой классификацией модели делятся на две большие группы: **материальные** (или предметные) и **информационные** (абстрактные).

Материальные модели можно назвать **предметными**, **физическими**. Они воспроизводят геометрические и физические свойства оригинала и всегда имеют реальное воплощение.

Материальные модели - это, к примеру, карты при изучении истории и географии, схемы солнечной системы и звездного неба, макет многоступенчатой ракеты и др.

Материальные модели - это и различные физические и химические опыты. В них моделируются процессы, например, реакция между водородом и кислородом.

Информационная модель- совокупность информации, характеризующая свойства и состояния объекта, процесса, явления, а также взаимосвязь с внешним миром. **Информационная модель** - это описание объекта.

Информация, характеризующая объект или процесс, может иметь разный объем и форму представления, выражаться различными средствами.

Знаковые и вербальные информационные модели

К информационным моделям можно отнести *вербальные* (от лат. «verbalis» — устный) модели, полученные в результате умозаключений. Они могут остаться мысленными или быть выражены словесно. К таким моделям можно отнести идею, возникшую у изобретателя, и музыкальную тему, промелькнувшую в голове композитора, и рифму, прозвучавшую пока еще в сознании поэта.

Вербальная модель- информационная модель в мысленной или разговорной форме.

Знаковая модель- информационная модель, выраженная специальными знаками, т. е. средствами любого формального языка.

Знаковые модели окружают нас повсюду. Это рисунки, тексты, графики и схемы. Вербальные и знаковые модели, как правило, взаимосвязаны. Мысленный образ, родившийся в мозгу человека, может быть облечен в знаковую форму. И наоборот, знаковая модель помогает сформировать в сознании верный мысленный образ.

К знаковым моделям относятся:

- *математические модели*- это модели, построенные с использованием математических понятий и формул. Например, модель равноускоренного прямолинейного движения:

$$S = v_0 t + \frac{at^2}{2}$$

- *специальные*– представлены на специальных языках (ноты, химические формулы);
- *алгоритмические*– программы.

Существуют и другие подходы к классификации информационных моделей.

В зависимости от структуры информационные модели делятся на:

- табличные;

- иерархические;
- сетевые.

В **табличной** информационной модели объекты или их свойства представлены в виде списка, а их значения размещаются в ячейках таблицы. Примером химической табличной модели служит Периодическая система элементов Менделеева.

В **иерархической** информационной модели объекты распределены по уровням, причем, элементы нижнего уровня входят в состав одного из элементов более высокого уровня. Такие модели строятся в процессе классификации объектов. Примером может служить модель классификации современных компьютеров.

Сетевые информационные модели применяются для описания таких систем, в которых связь между элементами имеет сложную структуру (например, структура сети Интернет).

Компьютерные и некомпьютерные модели

В информатике рассматриваются модели, которые можно создавать и исследовать с помощью компьютера. В этом случае модели делят на **компьютерные и некомпьютерные**.

Компьютерная модель- это модель, реализованная средствами программной среды.

В настоящее время выделяют два вида **компьютерных** моделей:

1. *структурно-функциональные*, которые представляют собой условный образ объекта, описанный с помощью компьютерных технологий;
2. *имитационные*, представляющие собой программу или комплекс программ, позволяющий воспроизводить процессы функционирования объекта в разных условиях.

Значение компьютерного моделирования сложно переоценить. К нему прибегают при исследовании сложных систем в различных областях науки, при создании образов исчезнувших животных, растений, зданий и т. п. Редкий кинорежиссер сегодня обходится без компьютерных эффектов. Кроме того, современное компьютерное моделирование является мощным инструментом развития науки.

Классификация моделей по фактору времени:

- **Статические** – модели, описывающие состояние системы в определенный момент времени (единовременный срез информации по данному объекту).

Примеры моделей: классификация животных...., строение молекул, список посаженных деревьев, отчет об обследовании состояния зубов в школе и тд.

· *Динамические* – модели, описывающие процессы изменения и развития системы (изменения объекта во времени). *Примеры:* описание движения тел, развития организмов, процесс химических реакций.

Классификация моделей по отрасли знаний (отрасли деятельности человека):

- математические;
- биологические;
- химические;
- социальные;
- экономические;
- исторические и т. д.

Классификация моделей по форме представления:

· *Материальные* – это предметные (физические) модели. Они всегда имеют реальное воплощение. Отражают внешнее свойство и внутреннее устройство исходных объектов, суть процессов и явлений объекта-оригинала. Это экспериментальный метод познания окружающей среды. *Примеры:* детские игрушки, скелет человека, чучело, макет солнечной системы, школьные пособия, физические и химические опыты.

· *Абстрактные (нематериальные)* – не имеют реального воплощения. Их основу составляет информация. Это теоретический метод познания окружающей среды.

По признаку реализации:

· *Мысленные* – формируются в воображении человека в результате раздумий, умозаключений, иногда в виде некоторого образа. Это модель сопутствует сознательной деятельности человека.

· *Вербальные* – мысленные модели выраженные в разговорной форме. Используется для передачи мыслей.

· *Информационные* – целенаправленно отобранная информация об объекте, которая отражает наиболее существенные для исследователя свойств этого объекта.

Нас, как правило, интересуют информационные модели, поэтому рассмотрим их классификацию подробнее.

Типы информационных моделей:

· **Табличные**– объекты и их свойства представлены в виде списка, а их значения размещаются в ячейках прямоугольной формы. Перечень однотипных объектов размещен в первом столбце (или строке), а значения их свойств размещаются в следующих столбцах (или строках).

· **Иерархические** – объекты распределены по уровням. Каждый элемент высокого уровня состоит из элементов нижнего уровня, а элемент нижнего уровня может входить в состав только одного элемента более высокого уровня.

· **Сетевые**– применяют для отражения систем, в которых связи между элементами имеют сложную структуру.

Информационные модели по степени формализации:

· Образно-знаковые модели:

1. **Геометрические** (рисунок, пиктограмма, чертеж, карта, план, объемное изображение).

2. **Структурные** (таблица, граф, схема, диаграмма).

3. **Словесные** (описание естественными языками).

4. **Алгоритмические** (нумерованный список, пошаговое перечисление, блок-схема).

· Знаковые модели:

1. **Математические** – представлены матем.формулами, отображающими связь параметров.

2. **Специальные** – представлены на спец. языках (ноты, хим.формулы).

3. **Алгоритмические** – программы.

Существуют и другие классификации моделей, приведу для примера еще одну.

Классификация моделей по различным классификационным признакам

1. Принадлежность к иерархическому уровню:

· Модели микроуровня.

· Модели макроуровня.

· Модели метауровня.

2. Характер взаимоотношения с внешней средой:

· Открытые (непрерывный обмен).

· Закрытые (слабая связь с внешней средой).

3. Характер отображаемых свойств объекта:

- Структурные.
- Функциональные.

4. Способ представления свойств объекта:

- Аналитические.
- Алгоритмические.
- Имитационные.

5. Способ получения модели:

- Теоретические.
- Эмперические.

6. Причинная обусловленность:

- Детерминированные.
- Вероятностные.

7. По отношению ко времени:

- Статические.
- Динамические.

8. По типу уравнений:

- Линейные.
- Нелинейные.

9. По множеству значений переменных:

- Непрерывные.
- Дискретные.
- Дискретно-непрерывные.

10. По назначению:

- Технические.
- Экономические.
- Социальные и т. д.

Наиболее полный обзор основных классификаций и типов моделей представлен в [3]. Приведем его описание

В общем случае все модели, независимо от областей и сфер их применения, бывают трех типов: познавательные, прагматические и инструментальные.

Познавательная модель – форма организации и представления знаний, средство соединения новых и старых знаний. Познавательная модель обычно подгоняется под реальность и является теоретической моделью.

Прагматическая модель – средство организации практических действий, рабочего представления целей системы для ее управления. Реальность в них подгоняется под некоторую прагматическую модель. Это, как правило, прикладные модели.

Инструментальная модель – средство построения, исследования и использования прагматических или познавательных моделей.

Познавательные отражают существующие, а прагматические – хоть и не существующие, но желаемые и, возможно, исполнимые отношения и связи.

Вся остальная классификация моделей выстраивается по отношению к объекту-оригиналу, методам изучения и т.п.

Классификация моделей по степени абстрагирования

модели от оригинала

По степени абстрагирования от оригинала (рис. 6) модели могут быть разделены на материальные (физические) и идеальные. К **материальным** относятся такие способы, при которых исследование ведется на основе модели, воспроизводящей основные геометрические, физические, динамические и функциональные характеристики изучаемого объекта. Основными разновидностями физических моделей являются [4]:

- натурные;
- квазинатурные; масштабные

Натурные – это реальные исследуемые системы, которые являются макетами и опытными образцами. Натурные модели имеют полную адекватность с системой-оригиналом, что обеспечивает высокую точность и достоверность результатов моделирования; другими словами, модель натурная, если она есть материальная копия объекта моделирования. Например, глобус – натурная географическая модель земного шара.

Квазинатурные (от лат. «квази» – почти) – это совокупность натуральных и математических моделей. Этот вид моделей используется в случаях, когда математическая модель части системы не является удовлетворительной или

когда часть системы должна быть исследована во взаимодействии с остальными частями, но их еще не существует либо их включение в модель затруднено или дорого.



Рис.6. Классификации моделей по степени абстрагирования от объекта-оригинала

Масштабные модели – это системы той же физической природы, что и оригинал, но отличающиеся от него размерами. В основе масштабных моделей лежит математический аппарат теории подобия, который предусматривает соблюдение геометрического подобия оригинала и модели и соответствующих масштабов для их параметров. Примером масштабного моделирования являются любые разработки макетов домов, а порой и целых районов при проведении проектных работ при строительстве. Также масштабное моделирование используется при проектировании крупных объектов в самолетостроении и кораблестроении.

Аналоговое моделирование основано на аналогии процессов и явлений, имеющих различную физическую природу, но одинаково описываемых формально (одними и теми же математическими уравнениями, логическими схемами и т.п.). В качестве аналоговых моделей используются механические, гидравлические, пневматические системы, но наиболее широкое применение получили электрические и электронные аналоговые модели, в которых сила тока или напряжение является аналогами физических величин другой природы. Например, является общеизвестным, что математическое уравнение колебания маятника имеет эквивалент при записи уравнения колебаний тока.

Идеальное моделирование носит теоретический характер. Различают два типа идеального моделирования: интуитивное и знаковое.

Под **интуитивным** будем понимать моделирование, основанное на интуитивном представлении об объекте исследования, не поддающемся формализации либо не нуждающимся в ней. В этом смысле, например, жизненный опыт каждого человека может считаться его интуитивной моделью окружающего мира.

Знаковым называется моделирование, использующее в качестве моделей знаковые преобразования различного вида: схемы, графики, чертежи, формулы, наборы символов и т.д., включающие совокупность законов, по которым можно оперировать с выбранными знаковыми элементами. Знаковая модель может делиться на лингвистическую, визуальную, графическую и математическую модели.

Модель **лингвистическая**, если она представлена некоторым лингвистическим объектом, формализованной языковой системой или структурой. Иногда такие модели называют вербальными, например, правила дорожного движения – языковая, структурная модель движения транспорта и пешеходов на дорогах.

Модель **визуальная**, если она позволяет визуализировать отношения и связи моделируемой системы, особенно в динамике. Например, на экране компьютера часто пользуются визуальной моделью объектов, клавиатуры в программе-тренажере по обучению работе на клавиатуре.

Модель **графическая**, если она представлена геометрическими образами и объектами, например, макет дома является натурной геометрической моделью строящегося дома.

Важнейшим видом знакового моделирования является **математическое** моделирование, классическим примером математического моделирования является описание и исследование основных законов механики И. Ньютона средствами математики.

Классификация математических моделей Математические модели классифицируются:

- по принадлежности к иерархическому уровню;
- характеру отображаемых свойств объекта;
- способу представления свойств объекта; – способу получения модели;
- форме представления свойств объекта;
- по содержанию вероятностных компонентов.

- **По принадлежности к иерархическому уровню** математические модели делятся на модели микроуровня, макроуровня, метауровня (рис. 7).

Математические модели на **микроуровне** процесса отражают физические процессы, протекающие, например, при резании металлов. Они описывают процессы на уровне перехода (прохода).

Математические модели на **макроуровне** процесса описывают технологические процессы.

Математические модели на **метауровне** процесса описывают технологические системы (участки, цехи, предприятие в целом).

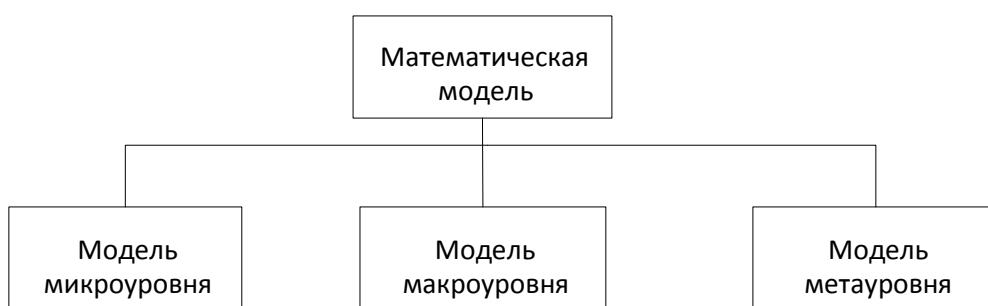


Рис. 7. Классификации математических моделей по принадлежности к иерархическому уровню

- **По характеру отображаемых свойств объекта** модели можно классифицировать на структурные и функциональные (рис. 8).

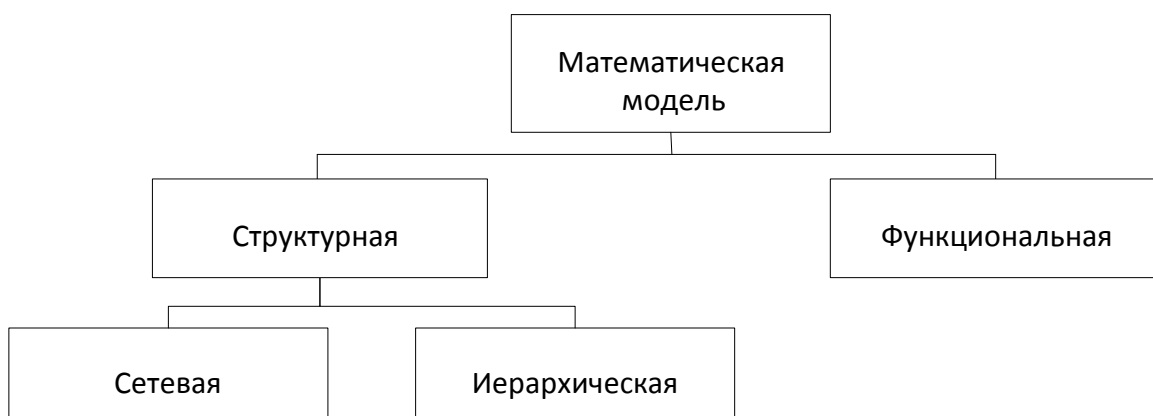


Рис. 8 Классификации математических моделей по характеру отображаемых свойств объекта

Модель **структурная**, если она представима структурой данных или структурами данных и отношениями между ними; например, структурной моделью может служить описание (табличное, графовое, функциональное или другое) трофической структуры экосистемы. В свою очередь, структурная модель может быть иерархической или сетевой.

Модель **иерархическая** (древовидная), если представима некоторой иерархической структурой (деревом); например, для решения задачи нахождения маршрута в дереве поиска можно построить древовидную модель, приведенную на рис. 9.

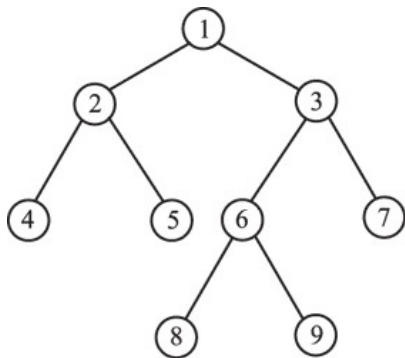


Рис. 9 Модель иерархической структуры

Модель **сетевая**, если она представима некоторой сетевой структурой.

□ По способу представления свойств объекта (рис. 1.0) модели делятся на аналитические, численные, алгоритмические и имитационные [2, 4, 6, 8, 16].



Рис. 10. Классификации математических моделей по способу представления свойств объекта

Аналитические математические модели представляют собой явные математические выражения выходных параметров как функций от параметров входных и внутренних и имеют единственные решения при любых начальных условиях. Например, процесс резания (точения) с точки зрения действующих сил представляет собой аналитическую модель. Также квадратное уравнение, имеющее одно или несколько решений, будет аналитической моделью.

Модель будет **численной**, если она имеет решения при конкретных начальных условиях (дифференциальные, интегральные уравнения).

Модель **алгоритмическая**, если она описана некоторым алгоритмом или комплексом алгоритмов, определяющим ее функционирование и развитие. Введение данного типа моделей (действительно, кажется, что любая модель может быть представлена алгоритмом ее исследования) вполне обосновано, т.к. не все модели могут быть исследованы или реализованы алгоритмически. Например, моделью вычисления суммы бесконечного убывающего ряда чисел может служить алгоритм вычисления конечной суммы ряда до некоторой заданной степени точности. Алгоритмической моделью корня квадратного из числа X может служить алгоритм вычисления его приближенного, сколь угодно точного значения по известной рекуррентной формуле.

Модель будет **имитационная**, если она предназначена для испытания или изучения возможных путей развития и поведения объекта путем варьирования некоторых или всех параметров модели, например модель экономической системы производства товаров двух видов. Такую модель можно использовать в качестве имитационной с целью определения и варьирования общей стоимости в зависимости от тех или иных значений объемов производимых товаров.

По способу получения модели делятся на теоретические и эмпирические (рис. 11).

Теоретические математические модели создаются в результате исследования объектов (процессов) на теоретическом уровне. Например, существуют выражения для сил резания, полученные на основе обобщения физических законов. Но они неприемлемы для практического использования, т.к. очень громоздки и не совсем адаптированы к реальным процессам обработки материалов.



Рис. 11. Классификации математических моделей по способу получения модели

Эмпирические математические модели создаются в результате проведения экспериментов (изучения внешних проявлений свойств объекта с помощью измерения его параметров на входе и выходе) и обработки их результатов методами математической статистики.

По форме представления свойств объекта модели делятся на логические, теоретико-множественные и графовые (рис. 12).

Модель *логическая*, если она представлена предикатами, логическими функциями, например совокупность двух логических функций может служить математической моделью одноразрядного сумматора.

Модель *теоретико-множественная*, если она представлена с помощью некоторых множеств и отношений принадлежности к ним и между ними.

Модель *графовая*, если она представлена графом или графами и отношениями между ними.



Рис. 12. Классификации математических моделей по форме представления свойств объекта

По содержанию вероятностных компонентов модели делятся на детерминированные и стохастические (рис. 13).

Если модель не содержит вероятностных (стохастических) компонентов, она называется **детерминированной**. Примером такой модели является система дифференциально-разностных уравнений, описывающих химическую реакцию либо класс систем, описываемых уравнениями в частных производных параболического типа. В детерминированной модели результат можно получить, когда для нее заданы все входные величины.

Однако множество систем моделируются с несколькими случайными входными величинами, в результате чего создается **стохастическая (вероятностная)** модель. Примерами таких моделей являются системы массового обслуживания и управления запасами. Стохастические модели выдают результат, который является случайным сам по себе, и поэтому может он рассматриваться как оценка истинных характеристик модели [3].



Рис. 13. Классификации математических моделей по содержанию вероятностных компонентов

Классификация моделей по степени устойчивости

Все модели могут быть разделены на устойчивые и неустойчивые.

Устойчивой является такая система, которая, будучи выведена из своего исходного состояния, стремится к нему. Она может колебаться некоторое время около исходной точки подобно обычному маятнику, приведенному в движение, но возмущения в ней со временем затухают и исчезают.

В **неустойчивой** системе, находящейся первоначально в состоянии покоя, возникшее возмущение усиливается, вызывая увеличение значений соответствующих переменных или их колебания с возрастающей амплитудой.

Классификация моделей по отношению к внешним факторам

По отношению к внешним факторам модели могут быть разделены на открытые и замкнутые.

Замкнутой моделью является модель, которая функционирует вне связи с внешними (экзогенными) переменными. В замкнутой модели изменения значений переменных во времени определяются внутренним

взаимодействием самих переменных. Закрытая модель может выявить поведение системы без ввода внешней переменной. Пример: информационные системы с обратной связью являются закрытыми системами. Это самонастраивающиеся системы, и их характеристики вытекают из внутренней структуры и взаимодействий, которые отражают ввод внешней информации.

Модель, связанная с внешними (экзогенными) переменными, называется *открытой*.

Классификация моделей по отношению ко времени

Существуют две классификации моделей по отношению к временному фактору. Модели могут быть:

- непрерывными или дискретными;
- статическими или динамическими.

Непрерывная модель описывает систему во времени с помощью представления, в котором переменные состояния меняются непрерывно по отношению ко времени. Примером непрерывной модели является сложная система дифференциальных уравнений, которые устанавливают отношения для скоростей изменения переменных состояния во времени. В *дискретной* модели значения переменных можно определить только в конкретные моменты времени [4].

По отношению к временному фактору модели делятся на динамические и статические (рис. 14).

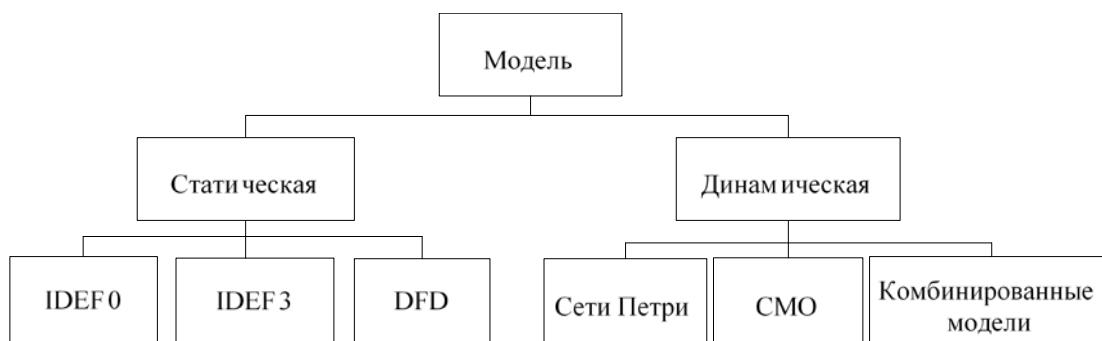


Рис. 14. Схема классификации математических моделей по отношению ко времени

Модель называется *статической*, если среди параметров, участвующих в ее описании, нет временного параметра. Статическая модель в каждый момент времени дает лишь «фотографию» системы, ее срез.

Одним из видов статических моделей являются структурные модели.

Динамической моделью называется модель, если среди ее параметров есть временной параметр, т.е. она отображает систему (процессы в системе) во времени.

Этапы разработки моделей

Процесс моделирования имеет итерационный характер и проводится в рамках ранее сформулированных целей и с соблюдением границ моделирования. Построение начинается с изучения (обследования) реальной системы, ее внутренней структуры и содержания взаимосвязей между ее элементами, а также внешних воздействий, и завершается разработкой модели.

Моделирование - от постановки задачи до получения результатов - проходит следующие этапы:

1. Анализ требований и проектирование:

- Постановка и анализ задачи и цели моделирования.
- Сбор и анализ исходной информации об объекте моделирования.
- Построение концептуальной модели. 4. Проверка достоверности концептуальной модели.

2. Разработка модели:

- Выбор среды моделирования.
- Составление логической модели.
- Назначение свойств модулям модели.
- Задание модельного времени.
- Верификация модели.

3. Проведение эксперимента:

- Запуск модели, прогон модели.
- Варьирование параметров модели и сбор статистики.
- Анализ результатов моделирования.

4. Подведение итогов моделирования согласно поставленной цели и задачи моделирования

На рис.15. представлена схема этапов моделирования. Отметим, что при разработке конкретных моделей с определенными целями и границами моделирования нет необходимости выполнять все подэтапы. Например, при разработке статических моделей IDEF0, DFD 3 и 4 подэтапы «Разработки модели» не выполняются, т.к. эти методологии не предусматривают задание временных параметров модели.

На первом этапе моделирования – «Анализ требований и проектирование» – формулируется концептуальная модель, строится ее формальная схема и решается вопрос об эффективности и целесообразности моделирования системы.

Концептуальная модель (КМ) – это абстрактная модель, определяющая состав и структуру системы, свойства элементов и причинноследственные связи, присущие анализируемой системе и существенные для достижения целей моделирования. В таких моделях обычно в словесной форме приводятся сведения о природе и параметрах (характеристиках) элементарных явлений исследуемой системы, о виде и степени взаимодействия между ними, о месте и значении каждого элементарного явления в общем процессе функционирования системы.



Рис. 15. Схема этапов моделирования.

При создании КМ практически параллельно формируется область исходных данных (информационное пространство системы) – этап подготовки исходных данных. На данном этапе выявляются количественные характеристики (параметры) функционирования системы и ее элементов, численные значения которых составят исходные данные для моделирования. Очевидно, что значительная часть параметров системы – это случайные величины. Поэтому особое значение при формировании исходных данных имеют выбор законов распределения случайных величин, аппроксимация

функций и т.д. В результате выявления свойств модели и построения концептуальной модели необходимо проверить адекватность модели.

Второй этап моделирования - «*Разработка модели*». Здесь происходит уточнение или выбор программного пакета моделирования. Выбор средств моделирования: программные и технические средства выбираются с учетом ряда критериев. Непременное условие - необходимость и достаточность средств для реализации концептуальной модели. Среди других критериев можно назвать доступность, простоту и легкость освоения, скорость и корректность создания программной модели.

После выбора среды проектирования концептуальная модель, сформулированная на предыдущем этапе, воплощается в компьютерную модель, т.е. решается проблема алгоритмизации и детализации модели.

Модель системы представляется в виде совокупности частей (элементов, подсистем). В эту совокупность включаются все части, которые обеспечивают сохранение целостности системы, с одной стороны, а с другой – достижение поставленных целей моделирования (получения необходимой точности и достоверности результатов при проведении компьютерных экспериментов над моделью). В дальнейшем производится окончательная детализация, локализация (выделение системы из окружающей среды), структуризация (указание и общее описание связей между выделенными элементами системы), укрупненное описание динамики функционирования системы и ее возможных состояний.

Для выполнения подэтапа «*Задание модельного времени*», нужно ввести понятие модельного времени. В компьютерной модели переменная, обеспечивающая текущее значение модельного времени, называется *часами модельного времени*.

Существует два основных подхода к продвижению модельного времени: *продвижение времени от события к событию* и *продвижение времени с постоянным шагом* [4].

Подход, использующий продвижение времени в модели от события к событию, применяется всеми основными компьютерными программами и большинством разработчиков, создающих свои модели на универсальных языках (рис. 16) [4].

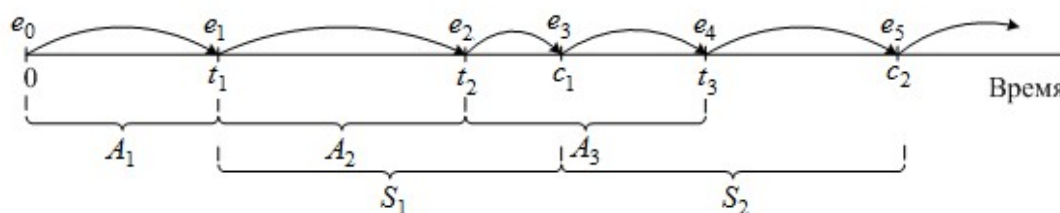


Рис. 16. Механизм продвижения модельного времени от события к событию

При использовании продвижения времени от события к событию часы модельного времени в исходном состоянии устанавливаются в 0, и определяется время возникновения будущих событий. После этого часы модельного времени переходят на время возникновения ближайшего события, и в этот момент обновляются состояние системы, с учетом произошедшего события, а также сведения о времени возникновения будущих событий. Затем часы модельного времени продвигаются ко времени возникновения следующего нового ближайшего события, обновляется состояние системы и определяется время будущих событий и т.д. Процесс продвижения модельного времени от времени возникновения одного события ко времени возникновения другого продолжается до тех пор, пока не будет выполнено какое-либо условие останова, указанное заранее. Поскольку в дискретно-событийной имитационной модели все изменения происходят только во время возникновения событий, периоды бездействия системы просто пропускаются, и часы переводятся со времени возникновения одного события на время возникновения другого. При продвижении времени с постоянным шагом такие периоды бездействия не пропускаются, что приводит к большим затратам компьютерного времени. Следует отметить, что длительность интервала продвижения модельного времени от одного события к другому может быть различной [4].

При продвижении времени с постоянным шагом dt часы модельного времени продвигаются точно на dt единиц времени для какого-либо соответствующего выбора значения dt . После каждого обновления часов выполняется проверка, чтобы определить, произошли какие-либо события в течение предыдущего интервала времени dt или нет. Если на этот интервал запланированы одно или несколько событий, считается, что данные события происходят в конце интервала, после чего состояние системы и статистические счетчики соответствующим образом обновляются. Продвижение времени посредством постоянного шага показано на рис. 17, где изогнутые стрелки показывают продвижение часов модельного времени, а e_i ($i = 1, 2, \dots$) – это действительное время возникновения события i любого типа, а не значение часов модельного времени. На интервале $[0, dt]$ событие происходит в момент времени e_1 , но оно рассматривается как произошедшее в момент времени dt . На интервале $[dt, 2 dt]$ события не происходят, но все же модель выполняет проверку, чтобы убедиться в этом. На интервале $[2 dt, 3 dt]$ события происходят в моменты времени e_2 и e_3 , однако считается, что они произошли в момент времени $3 dt$ и т.д. В ситуациях, когда принято считать, что два или несколько событий происходят в одно и то же время, необходимо применение ряда правил, позволяющих определять, в каком порядке обрабатывать события.

Таким образом, продвижение времени посредством постоянного шага имеет два недостатка:

1. возникновение ошибок, связанных с обработкой событий в конце интервала, в течение которого они происходят,
2. необходимость решать, какое событие обрабатывать первым, если события, происходящие в разное время, рассматриваются как одновременные.

Подобного рода проблемы можно частично решить, сделав интервалы dt менее продолжительными, но тогда возрастает число проверок возникновения событий, что приводит к увеличению времени выполнения задачи. Принимая во внимание это обстоятельство, продвижение времени с помощью постоянного шага не используют в дискретно-событийных имитационных моделях, когда интервалы времени между последовательными событиями могут значительно отличаться по своей продолжительности [4].

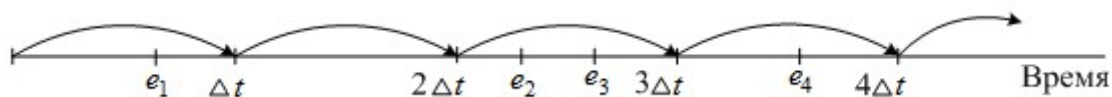


Рис. 17. Пример продвижения модельного времени посредством постоянного шага

В основном этот подход предназначен для систем, в которых можно допустить, что все события в действительности происходят в один из моментов n времени dt ($n = 0, 1, 2, \dots$) для соответственно выбранного dt . Так, в экономических системах данные часто предоставляются за годовые промежутки времени, поэтому естественно в имитационной модели установить продвижение времени с шагом, равным одному году. Следует заметить, что продвижение времени посредством постоянного шага может быть выполнено с помощью механизма продвижения времени от события к событию, если планировать время возникновения событий через dt единиц времени, т.е. данный подход является разновидностью механизма продвижения времени от события к событию.

Третий этап – «Проведение эксперимента» – является решающим, на котором, благодаря процессу имитации моделируемой системы, происходит сбор необходимой информации, ее статической обработки в интерпретации результатов моделирования, в результате чего принимается решение: либо исследование будет продолжено, либо закончено. Если известен результат, то можно сравнить его с полученным результатом моделирования. Полученные выводы часто способствуют проведению дополнительной серии экспериментов, а иногда и изменению модели. Основой для выработки

решения служат результаты тестирования и экспериментов. Если результаты не соответствуют целям моделирования (реальному объекту или процессу), значит, допущены ошибки на предыдущих этапах или входные данные не являются лучшими параметрами в изучаемой области, поэтому разработчик возвращается к одному из предыдущих этапов.

Подэтап «Анализ результатов моделирования» представляет собой всесторонний анализ полученных результатов с целью получения рекомендаций по проектированию системы или ее модификации.

На этапе «*Подведение итогов моделирования согласно поставленной цели и задачи моделирования*» проводят оценку проделанной работы, сопоставляют поставленные цели с полученными результатами и создают окончательный отчет по выполненной работе.

2 Аналитическое моделирование на основе систем массового обслуживания

Системы массового обслуживания - это системы, в которые в случайные моменты времени поступают заявки на обслуживание, при этом поступившие заявки обслуживаются с помощью имеющихся в распоряжении системы каналов обслуживания [5.6].

С позиции моделирования процесса массового обслуживания ситуации, когда образуются очереди заявок (требований) на обслуживание, возникают следующим образом. Поступив в обслуживающую систему, требование присоединяется к очереди других (ранее поступивших) требований. Канал обслуживания выбирает требование из находящихся в очереди, с тем чтобы приступить к его обслуживанию. После завершения процедуры обслуживания очередного требования канал обслуживания приступает к обслуживанию следующего требования, если таковое имеется в блоке ожидания.

Цикл функционирования системы массового обслуживания подобного рода повторяется многократно в течение всего периода работы обслуживающей системы. При этом предполагается, что переход системы на обслуживание очередного требования после завершения обслуживания предыдущего требования происходит мгновенно, в случайные моменты времени.

Примерами систем массового обслуживания могут служить:

1. посты технического обслуживания автомобилей;
2. посты ремонта автомобилей;
3. персональные компьютеры, обслуживающие поступающие заявки или требования на решение тех или иных задач;

4. станции технического обслуживания автомобилей;
5. аудиторские фирмы;
6. отделы налоговых инспекций, занимающиеся приемкой и проверкой текущей отчетности предприятий;
7. телефонные станции и т. д.

Основными компонентами системы массового обслуживания любого вида являются [5,6]:

1. входной поток поступающих требований или заявок на обслуживание;
2. дисциплина очереди;
3. механизм обслуживания.

Входной поток требований. Для описания входного потока требуется задать вероятностный закон, определяющий последовательность моментов поступления требований на обслуживание и указать количество таких требований в каждом очередном поступлении. При этом, как правило, оперируют понятием «вероятностное распределение моментов поступления требований». Здесь могут поступать как единичные, так и групповые требования (требования поступают группами в систему). В последнем случае обычно речь идет о системе обслуживания с параллельно-групповым обслуживанием.

Дисциплина очереди - это важный компонент системы массового обслуживания, он определяет принцип, в соответствии с которым поступающие на вход обслуживающей системы требования подключаются из очереди к процедуре обслуживания. Чаще всего используются дисциплины очереди, определяемые следующими правилами:

- первым пришел — первый обслуживается;
- пришел последним — обслуживается первым;
- случайный отбор заявок;
- отбор заявок по критерию приоритетности;
- ограничение времени ожидания момента наступления обслуживания (имеет место очередь с ограниченным временем ожидания обслуживания, что ассоциируется с понятием «допустимая длина очереди»).

Механизм обслуживания определяется характеристиками самой процедуры обслуживания и структурой обслуживающей системы. К характеристикам процедуры обслуживания относятся: продолжительность процедуры обслуживания и количество требований, удовлетворяемых в результате выполнения каждой такой процедуры. Для аналитического описания характеристик процедуры обслуживания оперируют понятием «вероятностное распределение времени обслуживания требований».

Следует отметить, что время обслуживания заявки зависит от характера самой заявки или требований клиента и от состояния и возможностей обслуживающей системы. В ряде случаев приходится также учитывать вероятность выхода обслуживающего прибора по истечении некоторого ограниченного интервала времени.

Структура обслуживающей системы определяется количеством и взаимным расположением каналов обслуживания (механизмов, приборов и т. п.). Прежде всего следует подчеркнуть, что система обслуживания может иметь не один канал обслуживания, а несколько; система такого рода способна обслуживать одновременно несколько требований. В этом случае все каналы обслуживания предлагают одни и те же услуги, и, следовательно, можно утверждать, что имеет место параллельное обслуживание.

Система обслуживания может состоять из нескольких разнотипных каналов обслуживания, через которые должно пройти каждое обслуживаемое требование, т. е. в обслуживающей системе процедуры обслуживания требований реализуются последовательно. Механизм обслуживания определяет характеристики выходящего (обслуженного) потока требований.

Предметом теории массового обслуживания является установление зависимости между факторами, определяющими функциональные возможности системы массового обслуживания, и эффективностью ее функционирования. В большинстве случаев все параметры, описывающие системы массового обслуживания, являются случайными величинами или функциями, поэтому эти системы относятся к стохастическим системам.

Случайный характер потока заявок (требований), а также, в общем случае, и длительности обслуживания приводит к тому, что в системе массового обслуживания происходит случайный процесс. По характеру случайного процесса, происходящего в системе массового обслуживания (СМО), различают системы марковские и немарковские. В марковских системах входящий поток требований и выходящий поток обслуженных требований (заявок) являются пуассоновскими. Пуассоновские потоки позволяют легко описать и построить математическую модель системы массового обслуживания. Данные модели имеют достаточно простые решения, поэтому большинство известных приложений теории массового обслуживания используют марковскую схему. В случае немарковских процессов задачи исследования систем массового обслуживания значительно усложняются и требуют применения статистического моделирования, численных методов с использованием ЭВМ.

Независимо от характера процесса, протекающего в системе массового обслуживания, различают два основных вида СМО:

1. системы с отказами, в которых заявка, поступившая в систему в момент, когда все каналы заняты, получает отказ и сразу же покидает очередь;
2. системы с ожиданием (очередью), в которых заявка, поступившая в момент, когда все каналы обслуживания заняты, становится в очередь и ждет, пока не освободится один из каналов. Системы массового обслуживания с ожиданием делятся на системы с ограниченным ожиданием и системы с неограниченным ожиданием.

В системах с ограниченным ожиданием может ограничиваться:

1. длина очереди;
2. время пребывания в очереди.

В системах с неограниченным ожиданием заявка, стоящая в очереди, ждет обслуживание неограниченно долго, т.е. пока не подойдет очередь.

Все системы массового обслуживания различают по числу каналов обслуживания:

1. одноканальные системы;
2. многоканальные системы.

Приведенная классификация СМО является условной. На практике чаще всего системы массового обслуживания выступают в качестве смешанных систем. Например, заявки ожидают начала обслуживания до определенного момента, после чего система начинает работать как система с отказами.

Сущность метода имитационного моделирования применительно к задачам массового обслуживания состоит в следующем. Строятся алгоритмы, при помощи которых можно вырабатывать случайные реализации заданных потоков однородных событий, а также моделировать процессы функционирования обслуживаемых систем. Эти алгоритмы используются для многократного воспроизведения реализации случайного процесса обслуживания при фиксированных условиях задачи. Получаемая при этом информация о состоянии процесса подвергается статистической обработке для оценки величин, являющихся показателями качества обслуживания

3. Моделирование компьютерных систем.

Ни один проект крупной сети со сложной топологией в настоящее время не обходится без исчерпывающего моделирования будущей сети. Программы, выполняющие эту задачу, достаточно сложны и дороги. Целью моделирования является определение оптимальной топологии, адекватный выбор сетевого оборудования, определение рабочих характеристик сети и возможных этапов будущего развития [7]. Необходимо помнить, что сеть,

слишком точно оптимизированная для решений задач текущего момента, может потребовать серьезных переделок в будущем. При этом на модели можно опробовать влияние всплесков широковещательных запросов или реализовать режим коллапса (для Ethernet), что вряд ли можно применить в работающей сети.

В процессе моделирования выясняются следующие параметры:

1. предельные пропускные способности различных фрагментов сети и зависимости потерь пакетов от загрузки отдельных станций и\или внешних каналов;
2. время отклика основных серверов в разных режимах, в том числе таких, которые в реальной сети крайне нежелательны;
3. влияние установки новых серверов на перераспределение информационных потоков;
4. решение оптимизации топологии при возникновении узких мест в сети (размещение серверов, внешних шлюзов, организация опорных каналов и пр.);
5. выбор того или иного типа сетевого оборудования (например, 10BaseTX или 100BaseFX) или режима его работы (например, cut-through, store-and-forward для мостов и переключателей и т.д.);
6. выбор внутреннего протокола маршрутизации и его параметров (например, метрики);
7. определение предельно допустимого числа пользователей того или иного сервера;
8. оценка влияния мультимедийного трафика на работу локальной сети, например, при подготовке видеоконференций.

Результаты моделирования должны иметь точность 10–15 % [3]

, т.к. этого достаточно для большинства целей и не требует слишком большого количества машинного времени. Для моделирования поведения реальной сети необходимо знать все ее рабочие параметры. Чем точнее будет воспроизведено поведение сети, тем больше машинного времени это потребует. Кроме того, необходимо сделать некоторые предположения относительно распределения загрузки для конкретных ЭВМ и других сетевых элементов, задержек в переключателях, мостах, времени обработки запросов в серверах. Здесь нужно учитывать и характер решаемых на ЭВМ задач.

В зависимости от целей и задач моделирования сетей выделяют моделирование аналитическое и имитационное.

3.1. Аналитическое моделирование сетей

Использование аналитических методов связано с необходимостью построения математических моделей локальных вычислительных сетей (ЛВС) в строгих математических терминах. Аналитические модели вычислительных сетей носят обычно вероятностный характер и строятся на основе понятий аппарата теорий массового обслуживания, вероятностей и марковских процессов, а также методов диффузной аппроксимации. Могут также применяться дифференциальные и алгебраические уравнения [7].

При использовании этого математического аппарата часто удается быстро получить аналитические модели для решения достаточно широкого круга задач исследования ЛВС. В то же время аналитические модели имеют ряд существенных недостатков, к числу которых следует отнести:

- значительные упрощения, свойственные большинству аналитических моделей (представление потоков заявок как простейших; предположение об экспоненциальном распределении длительностей обслуживания заявок; невозможность обслуживания заявок одновременно несколькими приборами, например, процессором и оперативной памятью, и др.). Подобные упрощения, а зачастую искусственное приспособление аналитических моделей с целью использования хорошо разработанного математического аппарата для исследования реальных ЛВС, ставят иногда под сомнение результаты аналитического моделирования [9];
- громоздкость вычислений для сложных моделей, например, использование для представления в модели процесса функционирования современной ЛВС по методу дифференциальных уравнений Колмогорова требует (для установившегося режима) решения сложной системы алгебраических уравнений [9];
- сложность аналитического описания вычислительных процессов ЛВС. Большинство известных аналитических моделей можно рассматривать лишь как попытку подхода к описанию процессов функционирования ЛВС;
- недостаточная развитость аналитического аппарата в ряде случаев не позволяет в аналитических моделях выбирать для исследования наиболее важные характеристики (показатели эффективности) ЛВС. Особенно большие затруднения при аналитическом моделировании связаны с учетом в процессах функционирования ЛВС программных средств операционных систем и другого общего ПО [9].

Указанные особенности позволяют заключить, что аналитические методы имеют самостоятельное значение лишь при исследовании процессов функционирования ЛВС в первом приближении и в частных, достаточно специфичных задачах. В этих случаях возможности исследования аналитических моделей ЛВС существенно расширяют приближенные

методы, например, - методы диффузионной аппроксимации, методы операционного анализа и аналитические сетевые модели.

3.2. Аналитическое моделирование сетей на основе систем массового обслуживания

При аналитическом моделировании исследование процессов или объектов заменяется построением их математических моделей и исследованием этих моделей. В основу метода положены идентичность формы уравнений и однозначность соотношений между переменными в уравнениях, описывающих оригинал и модель. Поскольку события, происходящие в локальных вычислительных сетях, носят случайный характер, то для их изучения наиболее подходящими являются вероятностные математические модели теории массового обслуживания [4].

Аналитическая модель сети представляет собой совокупность математических соотношений, связывающих между собой входные и выходные характеристики сети. При выводе таких соотношений приходится пренебрегать какими-то малосущественными деталями или обстоятельствами [3].

Телекоммуникационная сеть при некотором упрощении может быть представлена в виде совокупности процессоров (узлов), соединенных каналами связи. Сообщение, пришедшее в узел, ждет некоторое время до того, как оно будет обработано. При этом может образоваться очередь таких сообщений, ожидающих обработки. Время передачи или полное время задержки сообщения D равно:

$$D = T_p + S + W, \quad ()$$

где T_p , S и W , соответственно, время распространения, время обслуживания и время ожидания. Одной из задач аналитического моделирования является определение среднего значения D . При больших нагрузках основной вклад дает ожидание обслуживания W . Для описания очередей в дальнейшем будет использована нотация Д. Дж. Кенделла:

$A/B/C/K/m/z$,

где A – процесс прибытия;

B – процесс обслуживания;

C – число серверов (узлов);

K – максимальный размер очереди (по умолчанию – ∞); m – число клиентов (по умолчанию – ∞); z – схема работы буфера (по умолчанию FIFO).

Буквы А и В представляют процессы прихода и обслуживания и обычно заменяются следующими буквами, характеризующими закон, соответствующий распределению событий:

D – постоянная вероятность;

M – марковское экспоненциальное распределение;

G – обобщенный закон распределения;

E_k – распределение Эрланга порядка k;

H_k – гиперэкспоненциальное распределение порядка k [4].

Наиболее распространенными схемами работы буферов являются

FIFO (First-In-First-Out), LIFO (Last-In-First-Out) и FIRO (First-InRandom-Out). Например, запись M/M/2 означает очередь, для которой времена прихода и обслуживания имеют экспоненциальное распределение, имеется два сервера, длина очереди и число клиентов могут быть сколь угодно большими, а буфер работает по схеме FIFO [9].

Среднее время пребывания заявки в системе определяется **теоремой Литтла**:

Для любой системы массового обслуживания, при любом характере потока заявок, при любом распределении времени обслуживания и при любой дисциплине обслуживания среднее время пребывания заявки в системе ($T_{\text{сист}}$) в **стационарном режиме** равно среднему числу заявок в системе ($K_{\text{сист}}$), деленному на **интенсивность** потока заявок I [6].

$$T_{\text{сист}} = K_{\text{сист}} / I.$$

Для моделирования ЛВС наиболее часто используются следующие типы СМО:

1. *Одноканальные СМО с ожиданием.*

Представляют собой один обслуживающий прибор с бесконечной очередью. Данная СМО является наиболее распространенной при моделировании. С той или иной долей приближения с ее помощью можно моделировать практически любой узел ЛВС.

2. *Одноканальные СМО с потерями.*

Представляют собой один обслуживающий прибор с конечным числом мест в очереди. Если число заявок превышает число мест в очереди, то лишние заявки теряются. Этот тип СМО может быть использован при моделировании каналов передачи в ЛВС.

3. *Многочанальные СМО с ожиданием.*

Представляют собой несколько параллельно работающих обслуживающих приборов с общей бесконечной очередью. Данный тип СМО часто используется при моделировании групп абонентских терминалов ЛВС, работающих в диалоговом режиме.

4. *Многоканальные СМО с потерями.*

Представляют собой несколько параллельно работающих обслуживающих приборов с общей очередью, число мест в которой ограничено. Эти СМО, как и одноканальные с потерями, часто используются для моделирования каналов связи в ЛВС.

5. *Одноканальные СМО с групповым поступлением заявок.*

Представляют собой один обслуживающий прибор с бесконечной очередью. Перед обслуживанием заявки группируются в пакеты по определенному правилу.

6. *Одноканальные СМО с групповым обслуживанием заявок.*

Представляют собой один обслуживающий прибор с бесконечной очередью.

Заявки обслуживаются пакетами, составляемыми по определенному правилу. Последние два типа СМО могут использоваться для моделирования таких узлов ЛВС, как центры (узлы) коммутации.

Локальная вычислительная сеть в целом может быть представлена в виде сети массового обслуживания. Различают открытые, замкнутые и смешанные сети.

Открытой называется сеть массового обслуживания, состоящая из M узлов, причем хотя бы в один из узлов сети поступает извне входящий поток заявок, и имеется сток заявок из сети. Для открытых сетей характерно то, что интенсивность поступления заявок в сеть не зависит от состояния сети, т.е. от числа заявок, уже поступивших в сеть. Открытые сети используются для моделирования ЛВС, работающих в неоперативном режиме. Каждая заявка поступает на вход соответствующего узла коммутации, где определяется место ее обработки. Затем заявка передается на «свой» сервер или по каналу связи – на «соседний» сервер, где обрабатывается, после чего возвращается к источнику и покидает сеть [9].

Замкнутой называется сеть массового обслуживания с множеством узлов M без источника и стока, в которой циркулирует постоянное число заявок. Замкнутые СМО используются для моделирования таких ЛВС, источниками информации для которых служат абонентские терминалы, работающие в диалоговом режиме. В этом случае каждая группа абонентских терминалов представляется в виде многоканальной системы массового обслуживания с ожиданием и включается в состав устройств сети [9].

Различают простой и сложный режимы работы диалоговых абонентов. В простом режиме абоненты не производят никаких действий, кроме отправки заданий в ЛВС и обдумывания полученного ответа [9].

Абоненты с терминалов посылают запросы, которые по каналам связи поступают на узлы коммутации, а оттуда – на обработку на «свой» или «соседний» сервер. Дальнейшая обработка осуществляется так же, как в открытой сети [14].

При сложном режиме диалога работа абонентов представляется в виде совокупности операций некоего процесса, называемого технологическим процессом. Каждая операция технологического процесса моделируется соответствующей СМО. Часть операций предусматривает обращение к ЛВС, а часть операций может такого обращения не предусматривать [19].

Алгоритм работы самой ЛВС такой же, как для замкнутой сети.

Смешанной называется сеть массового обслуживания, в которой циркулирует несколько различных типов заявок (трафика), причем относительно одних типов заявок сеть замкнута, а относительно других типов заявок сеть открыта. С помощью смешанных СМО моделируются такие ЛВС, часть абонентов, которых работает в диалоговом, а часть – в неоперативном режиме. Для диалоговых абонентов также различают простой и сложный режим работы. Часто смешанные СМО моделируют ЛВС, в которых сервер дополнительно загружается задачами, решаемыми на фоне работы самой сети [19].

Алгоритм работы сети для диалоговых абонентов аналогичен алгоритму работы замкнутой сети, а алгоритм работы сети для неоперативных абонентов – алгоритму работы открытой сети.

Различают экспоненциальные и не экспоненциальные модели ЛВС.

Экспоненциальные модели основаны на предположении о том, что потоки заявок, поступающие в ЛВС, являются пуассоновскими, а время обслуживания в узлах ЛВС имеет экспоненциальное распределение.

Для таких сетей получены точные методы для определения их характеристик; трудоемкость получения решения зависит в основном от размерности сети [19].

Однако в большинстве сетей (и локальных сетей в частности) потоки не являются пуассоновскими. Модели таких сетей называются неэкспоненциальными. При анализе неэкспоненциальных сетей в общем случае отсутствуют точные решения, поэтому наибольшее применение здесь находят приближенные методы.

Одним из таких методов является метод диффузионной аппроксимации. Использование диффузионной аппроксимации позволило, к настоящему времени получить приближенные аналитические зависимости для определения характеристик всех типов СМО, рассмотренных выше.

При этом не требуется точного знания функций распределения случайных величин, связанных с данной СМО (интервалов между поступлениями заявок временем обслуживания в приборах), а достаточно только знание первого (математического ожидания) и второго (дисперсии или квадрата коэффициента вариации – ККВ) моментов этих величин [19].

Применение диффузионной аппроксимации при анализе ЛВС основано на следующем:

- по каждому типу заявок вычисляется интенсивность поступления заявок данного типа в узлы сети так, как если бы данный поток заявок циркулировал в сети только один;
- по определенному правилу, зависящему от типа СМО и дисциплины обслуживания, складываются потоки заявок от всех источников;
- по определенному правилу определяется среднее время обслуживания в каждом узле ЛВС;
- полученные значения подставляются в соответствующую диффузионную формулу и определяются характеристики узлов ЛВС;
- определяются характеристики ЛВС в целом.

Постановка задачи анализа ЛВС при этом примет следующий вид.

Дано:

- число узлов ЛВС;
- тип каждого узла ЛВС (тип СМО, моделирующей данный узел);
- дисциплина обслуживания в каждом узле ЛВС;
- общее число типов источников заявок, работающих в диалоговом режиме;
- общее число типов источников заявок, работающих в неоперативном режиме;
- для диалоговых источников в случае сложного режима работы число технологических процессов каждого типа, число операций в каждом технологическом процессе, среднее и ККВ времени выполнения каждой операции, матрица вероятностей передач между операциями, а также наличие или отсутствие на каждой операции обращения к ЛВС;

- для диалоговых источников в случае простого режима работы число источников (терминалов) каждого типа, среднее и ККВ времени реакции абонента на ответ сети;
- для неоперативных абонентов – средняя интенсивность поступления заявок и ККВ времени между поступлениями заявок; по каждому типу заявок (диалоговому и неоперативному) средняя интенсивность обслуживания в каждом узле ЛВС, ККВ времени обслуживания в узлах ЛВС и матрица вероятностей передач между узлами.

Требуется найти:

- среднее значение и дисперсию (или стандартное отклонение) времени задержки заявки каждого типа в ЛВС в целом;
- среднее значение и дисперсию (или стандартное отклонение) времени задержки в узлах ЛВС;
- загрузку узлов ЛВС;
- вероятность потери заявки в узле ЛВС (для узлов, моделируемых СМО с потерями).

Ограничения могут быть следующими:

- загрузка узлов не должна превышать 1;
- вероятность потери заявки не должна превышать 1;
- все характеристики должны быть положительны.

Иногда представляет интерес определение такого показателя, как максимальное время задержки заявки каждого типа в ЛВС. Максимальное время это такое время, превышение которого допустимо лишь для некоторого, наперед заданного процента заявок каждого типа. Для определения максимального времени используется методика, основанная на аппроксимации функции распределения времени задержки в сети эрланговским или гиперэкспоненциальным распределением, при этом необходимо задавать долю (процент) заявок, для которых рассчитывается максимальное время.

4. Сети Петри

Часто аналитики в задачах моделирования и анализа сложных параллельных и асинхронных систем обращаются к формальным системам, основанным на использовании математического аппарата сетей Петри. Формальная часть теории сетей Петри, основанная в начале 60-х годов немецким математиком Карлом Петри, в настоящее время содержит большое количество моделей, методов и средств анализа, имеющих обширное

количество приложений практически во всех отраслях вычислительной техники [10,11].

Прикладная теория сетей Петри связана главным образом с применением сетей Петри к моделированию систем, их анализу и получающимся в результате этого глубоким проникновением в моделируемые системы [8].

Моделирование в сетях Петри осуществляется на событийном уровне. Определяются, какие действия происходят в системе, какие состояние предшествовали этим действиям и какие состояния примет система после выполнения действия. Выполнение событийной модели в сетях Петри описывает поведение системы. Анализ результатов выполнения может сказать о том, в каких состояниях пребывала или не пребывала система, какие состояния в принципе не достижимы. Однако такой анализ не дает числовых характеристик, определяющих состояние системы.

Формальный аппарат сетей Петри предназначен для моделирования систем различного рода и отражает состояния исследуемой системы состоянием сети. Состояние сети Петри определяется ее маркировкой. Количество и распределение фишек сети определяют динамику исследуемой системы. Сеть Петри выполняется посредством запусков переходов в результате удаления фишек из его входных позиций и добавления их в выходные позиции перехода. Последовательность срабатываний переходов полностью определяет поведение сети. Таким образом, сеть Петри описывает структуру системы, ее состояние и поведение [8].

При введении ряда дополнительных правил и условий в алгоритмы моделирования получают различные разновидности сетей Петри. Это необходимо для определения модельного времени, которое позволит моделировать не только последовательность событий, но и их привязку ко времени. В настоящее время выделяют следующие разновидности сетей Петри:

1. временная сеть Петри (переходы обладают весом, определяющим продолжительность срабатывания - задержку);
2. стохастическая сеть Петри (задержки являются случайными величинами);
3. функциональная сеть Петри (задержки определяются как функции некоторых аргументов, например: количества меток в каких-либо позициях, состояния некоторых переходов);
4. цветная (раскрашенная) сеть Петри (метки могут быть различных типов, обозначаемых цветами, тип метки может быть использован как аргумент в функциональных сетях).

Более подробно рассмотрим их далее.

Основными свойствами сети Петри являются:

- ограниченность (число меток в любой позиции сети не может превысить некоторого значения K);
- безопасность (частный случай ограниченности);
- сохраняемость (постоянство загрузки ресурсов);
- достижимость (возможность перехода сети из одного заданного состояния, характеризуемого распределением меток, в другое);
- живость (возможностью срабатывания любого перехода при функционировании моделируемого объекта).

Однако формальная модель сетей Петри, в силу своей универсальности, имеет ряд недостатков, затрудняющих практическое применение для моделирования сложных систем. К основным таким недостаткам можно отнести следующие:

- высокая трудоемкость анализа сетей большой размерности, а реальные бизнес-процессы предприятия моделируются именно сетями большой размерности;

описательная мощность сетей Петри недостаточна для содержательного моделирования систем;

- обычные сети Петри не отражают требуемые временные характеристики моделируемой системы;
- фишка сети Петри не представляет собой никакой информации, кроме самого факта ее наличия, поэтому чрезвычайно сложно отразить преобразование информации при срабатывании переходов сети Петри;
- невозможность проведения логических преобразований и, как следствие, невозможность управления продвижением фишек по сети.

Недостатки сетей Петри не позволяют описывать сложные системы и в настоящее время используются для описания простейших операций. Также эти факторы явились причиной разработки подклассов и расширений сетей Петри, в которых вводятся определенные ограничения на структуру сети, что позволяет использовать более простые алгоритмы для ее анализа либо дополнительные элементы формальной системы, призванные увеличить ее описательную мощность.

В моделировании ВС сети Петри обеспечивают описание как алгоритмов и программ, так и собственно вычислительных систем и их устройств, а также порождаемых вычислительных процессов. Сети Петри используются для решения разнообразных задач анализа, синтеза и

оптимизации. В том числе для решения прикладных задач и в основном задач, связанных с моделями и средствами параллельной обработки информации.

В сетях Петри вводятся объекты двух типов: динамические – изображаются метками (маркерами) внутри позиций и статические – им соответствуют вершины сети Петри.

Распределение маркеров по позициям называют маркировкой. Маркеры могут перемещаться в сети. Каждое изменение маркировки называют событием, причем каждое событие связано с определенным переходом. Считается, что события происходят мгновенно и одновременно при выполнении некоторых условий.

Каждому условию в сети Петри соответствует определенная позиция. Совершению события соответствует срабатывание (возбуждение или запуск) перехода, при котором маркеры из входных позиций этого перехода перемещаются в выходные позиции. Последовательность событий образует моделируемый процесс. Перемещаемые по сети маркеры часто называют фишками

Состояние сети Петри определяется ее маркировкой. Количество и распределение фишек сети определяют динамику исследуемой системы. Сеть Петри выполняется посредством запусков переходов в результате удаления фишек из его входных позиций и добавления их в выходные позиции перехода. Последовательность срабатываний переходов полностью определяет поведение сети. Таким образом, сеть Петри описывает структуру системы, ее состояние и поведение

Достоинства аппарата сетей Петри:

- 1) Такие сети позволяют моделировать асинхронность и недетерминизм параллельных, независимых событий, параллелизм конвейерного типа, конфликтные ситуации между процессами (в сети Петри могут одновременно и независимо друг от друга сработать несколько переходов).
- 2) Сети Петри позволяют описывать как типовые ситуации в дискретных подсистемах, так и общую динамику работы сложной асинхронной системы.
- 3) Сети Петри позволяют производить иерархическую детализацию программных и аппаратных подсистем модели, производить совместное отображение структуры управления и потоков данных.
- 4) В последние годы появился ряд классов сетей, ориентированных на моделирование сложных систем с учетом таких фактов, как приоритетность процессов (сети с проверкой на нуль), временные

параметры событий, совместного отображения структуры управления и потоков данных.

- 5) Позволяет использовать единые методологические позиции для описания программного обеспечения, аппаратных средств и информационного обмена между системами;
- 6) Имеет большую анализирующую мощьность, которая позволяет формальными средствами доказывать существование или отсутствие определенных состояний сети Петри.

По определению сеть Петри является совокупностью объектов:

$$П = \langle P, T, I, O, \mu \rangle,$$

где конечное множество позиций $P = \{p_1, p_2, \dots, p_n\}$, обозначаются кружочками O ; конечное множество переходов $T = \{t_1, t_2, \dots, t_m\}$, обозначаются планками $-$, I – входная функция переходов; O – выходная функция переходов; μ - вектор маркировки сетей Петри.

Моделирование в СП осуществляется на событийном уровне. Переход отображает действия, происходящие в системе, а позиции состояния, предшествующие этим действиям. Модель Петри служит для анализа причинно-следственных связей в системе. Анализ результатов моделирования позволяет определить, в каких состояниях пребывала система.

Сеть Петри есть мультиграф, так как он допускает существование кратных дуг от одной вершины графа к другой. Так как дуги являются направленными, то это ориентированный мультиграф. Вершины графа можно разделить на два множества (позиции и переходы) таким образом, что каждая дуга будет направлена от элемента одного множества (позиций или переходов) к элементу другого множества (переходов или позиций); следовательно, такой граф является двудольным ориентированным мультиграфом (рис.18).

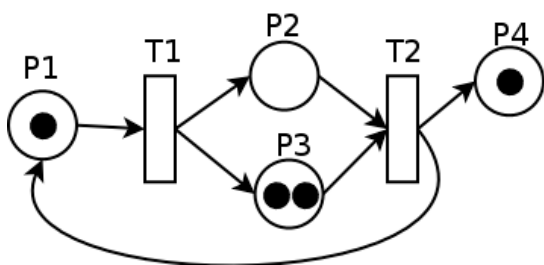


Рис. 18 - Пример сети Петри.

4.1. Свойства сетей Петри

Модель на основе сети Петри позволяет анализировать систему на наличие желательных или нежелательных свойств.

- 1) **Ограниченность.** Это свойство связано с введением ограничений на число меток в позициях.
- 2) **Безопасность.** Позиция сети Петри называется безопасной, если число фишек в ней никогда не превышает единицы.
- 3) **Сохраняемость.** Сеть Петри $S = (P, T, F, H, \mu_0)$ называется строго сохраняющей, если сумма фишек по всем позициям остается строго постоянной в процессе выполнения сети.
- 4) **Живость.** Под живостью перехода t_i понимают принципиальную возможность его срабатывания при функционировании сети Петри.
- 5) **Достижимость.** Свойство достижимости используется при установлении возможности возникновения некоторой ситуации в системе.

4.2. Анализ сетей Петри

Основная задача анализа сетей Петри – **задача достижимости**: достижима ли маркировка μ' из начальной маркировки μ_0 данной сети Петри.

Для ее решения имеется два основных подхода. Первый основан на построении дерева достижимости. Дерево достижимости - это ориентированное корневое дерево, вершинам которого, соответствуют возможные маркировки, а дугам – переходы. Начальная маркировка соответствует корню дерева. Из него исходят дуги, соответствующие разрешенным переходам. На каждом шаге строится очередной ярус дерева.

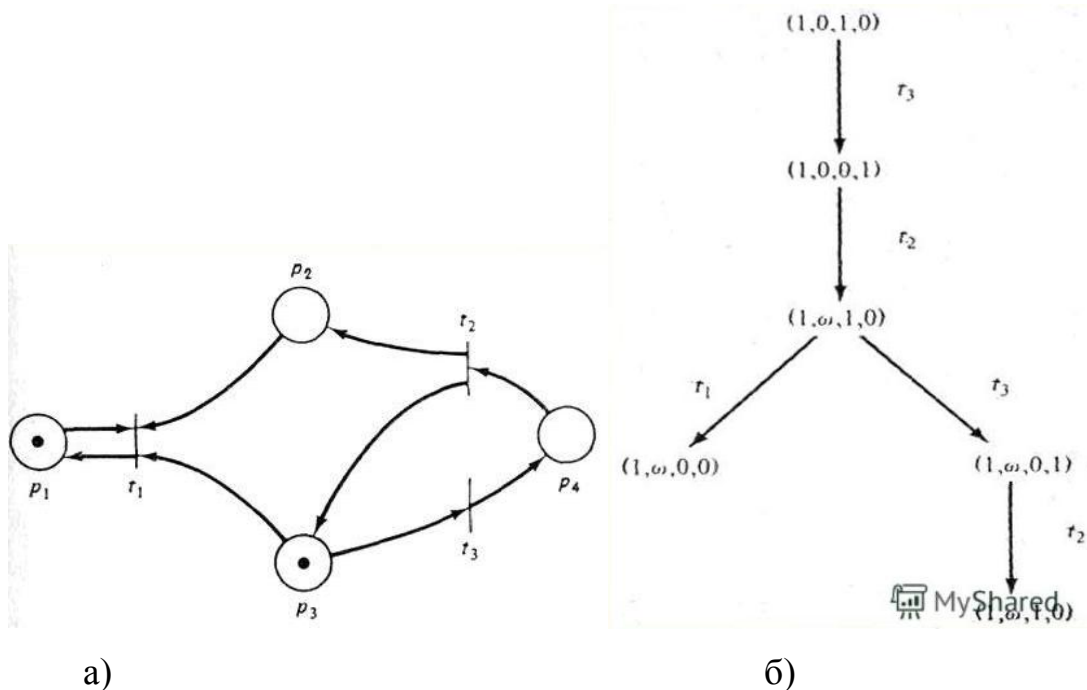


Рис. 19. - Дерево достижимости для сети Петри а), после четырех шагов имеет вид б).

Дерево представляет все возможные последовательности запусков переходов. Всякий путь, начинающийся в корне, соответствует допустимой последовательности переходов.

Другой подход к анализу сетей Петри называется матричным и основан на их матричном представлении. Пусть осуществляется последовательность срабатываний переходов:

$$\sigma' = t_{i_1}, t_{i_2}, \dots, t_{i_k}$$

В результате получится маркировка

$$\mu'^T = \mu_0^T + e^T(i_1) * D + e^T(i_{12}) * D + \dots + e^T(i_k) * D = \mu_0^T + f^T(\sigma) \cdot D$$

Для того чтобы существовала последовательность срабатываний σ , которая приводит из μ_0 в μ' , необходимо, чтобы вектор $f(\sigma)$ являлся неотрицательным целым решением матричного уравнения

$$\mu'^T = \mu_0^T + x^T D$$

Если это уравнение не имеет решений, разметка μ' является недостижимой на данной сети из μ_0 . Пусть $x = f(\sigma) = (2, 3, 1)$, это означает, что переход t_1 сработал два раза, переход t_2 сработал три раза, а переход t_3 – один раз.

Недостатки матричного анализа в том, что вектор срабатывания $f(\sigma)$ не дает информации о порядке срабатывания переходов, кроме того, возможны недействительные решения уравнения, т. е. получаем некоторую последовательность переходов, которая не возможна.

4.3. Подклассы и расширения сетей Петри

Анализ сетей большой размерности является достаточно трудоемким. В связи с этим разработаны подклассы сетей Петри, в которых вводятся определенные ограничения на структуру сети, что позволяет использовать более простые алгоритмы для ее анализа.

К подклассу *автоматных графов* относят сети Петри, в которых каждый переход имеет одну входную и одну выходную позиции. Такие сети описывают последовательные процессы и как математическая модель эквивалентны конечным автоматам. В автоматных графах легко представить конфликтные ситуации, но нельзя моделировать создание и уничтожение фишек, необходимых для моделирования параллельных процессов или ожидания.

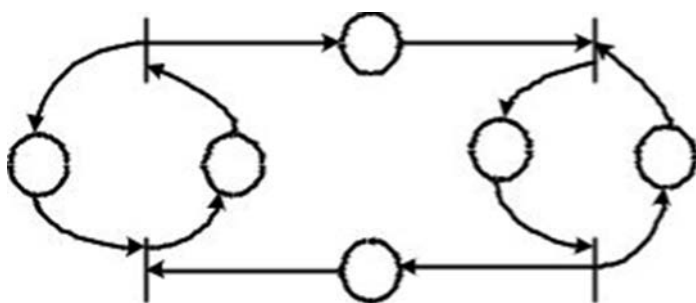


Рис.20. Маркированный граф

К подклассу *маркированных графов* относятся сети Петри, в которых каждая позиция имеет только один вход и один выход. Маркированные графы являются двойственными по отношению к автоматным графам. Они позволяют моделировать параллельность и синхронизацию, но не могут моделировать конфликты или принятие решений, зависящих от данных. Наиболее интересными структурными компонентами маркированных графов являются циклы. Пример маркированного графа приведен на Рис.20.

К подклассу *устойчивых сетей Петри* относятся сети, которые обладают следующим свойством: если при любой маркировке μ два любых перехода t_i и t_j оказываются разрешенными, то срабатывание одного из них не исключает возможности срабатывания другого перехода.

В теории сетей Петри предложены также несколько расширений, ориентированных на увеличение моделирующих возможностей сетей.

Приоритетные сети Петри и сети с проверкой на нуль позволяют учитывать приоритетность событий в модели. В сетях с проверкой на нуль

вводится дополнительное множество дуг запрета (сдерживающих дуг). На графе сети Петри такие дуги на конце имеют не стрелку, а маленький кружок (Рис.21.).

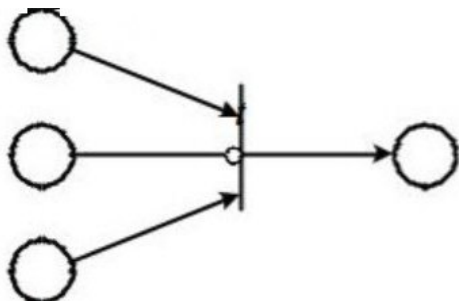


Рис.21. Сеть Петри

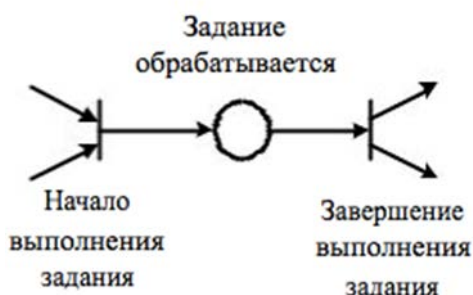


Рис. 22. Временные сети Петри

Временные сети Петри позволяют отразить в модели временные параметры системы. Если моделируемое событие имеет отличную от нуля длительность, как например, событие «задание обрабатывается», то оно представляется в виде двух мгновенных событий типа «начало события», «конец события» и условия «событие происходит» (Рис.22.). Считается, что события происходят неодновременно. Позиции во временных сетях взвешиваются временем выполнения.

E-сети, или оценочные сети – наиболее мощное расширение сетей Петри, являющееся средством описания моделей функционирования вычислительных систем. В *E-сетях* учитывается фактор времени, усложнена логика работы переходов, введены различные операции над метками.

4.4. Раскрашенные сети.

Большого внимания заслуживают сети высокого уровня, такие как раскрашенные сети Петри (Color Petri Net), являющиеся модификацией сетей Петри и отличающиеся хорошо разработанным математическим аппаратом,

широко применяемые для самых разнообразных практических целей. Основной причиной высокой эффективности этих формальных моделей является то, что они без потери возможностей формального анализа позволяют исследователю получить значительно более краткие и удобные описания, чем те, которые могут быть сделаны с помощью сетей низкого уровня. В сетях высокого уровня сложность моделей может быть разделена между структурой сети, надписями и описаниями. Это позволяет осуществлять описание значительно более сложных систем и анализировать процессы преобразования данных с помощью общепринятых математических выражений вместо сложного набора позиций, переходов и дуг. Раскрашенные сети Петри, в отличие от обычных сетей Петри, позволяют описывать структуру системы в виде иерархии диаграмм.

Раскрашенная сеть Петри - это графоориентированный язык для проектирования, описания, имитации и контроля распределенных и параллельных систем. Графическими примитивами показывается течение процесса, а конструкциями специального языка имитируется необходимая обработка данных. Сеть представляет собой направленный граф с двумя типами вершин - *позициями* и *переходами* при этом дуги не могут соединять вершины одного типа, т.е. граф является двудольным. Множество позиций описывают состояния системы. Переходы описывают условия изменения состояний. Позиции называются *входными* для конкретного перехода, если направление дуги, указывает на переход. Позиции называются *выходными* для перехода, если дуга ведет от перехода к позиции.



Рис.23. Элементы сети Петри

Сеть представляет собой асинхронную систему, в которой фишки перемещаются по позициям через переходы. Переход может, если во всех входных позициях для данного перехода присутствует хотя бы одна фишка и выполнено логическое выражение, ограничивающее переход (*спусковая функция*).



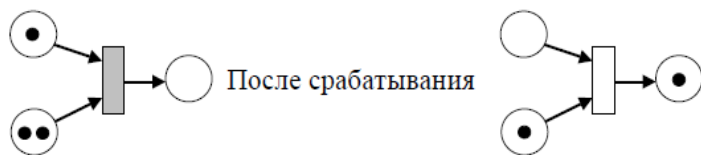


Рис.24. Срабатывание перехода сети Петри

В отличие от "классических" сетей Петри, в раскрашенных немаловажную роль играет типизация данных, основанная на понятии *множества цветов*, которое аналогично типу в декларативных языках программирования. Соответственно, для манипуляции цветом применяют переменные, функции и другие элементы, известные из языков программирования. Ключевой элемент раскрашенной сети Петри - позиция - имеет определенное значение из множества цветов.

Цвет позиции определяет тип фишек, которые могут там находиться. Конкретизация фишки, находящейся в данной позиции определяется инициализирующим выражением начальной разметки или формируется в результате правильного выполнения шага итерации сети Петри.

Дуги могут иметь пометки в виде выражений (переменных, констант или функций), определенных для множества цветов, и использоваться, либо для "вычленения" компонент сложного цвета фишек при определении условия срабатывания перехода, либо для изменения цвета фишки следующей позиции после срабатывания перехода.

Для анализа систем реального времени введен временной механизм, реализованный с помощью глобальных часов и, так называемых, *штампов*, которые несут фишки. Временной штамп фишки назначается при ее инициализации в начальной разметке или при создании фишки переходом и наращивается выражениями на переходах или дугах. В результате, фишка становится доступной для перехода, если ее штамп оказался меньше значения счетчика глобальных часов. Часы наращивают свое значение в случае, если на данный момент времени ни один переход сети не разрешен.

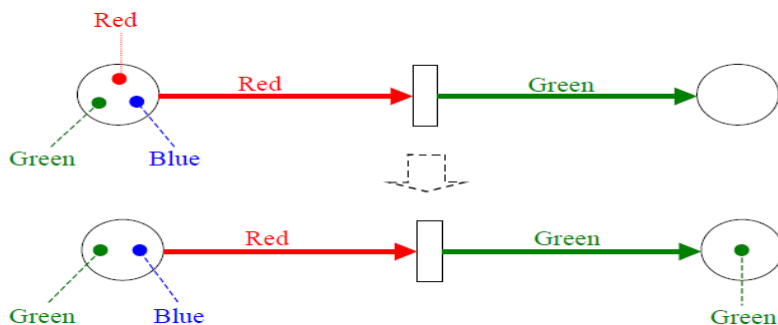


Рис.25. Простейшая раскрашенная сеть Петри

Раскрашенные сети Петри являются модификацией сетей Петри и отличаются хорошо разработанным математическим аппаратом, широко применяются для самых разнообразных практических целей (рис.25).

Основной причиной высокой эффективности этих формальных моделей является то, что они без потери возможностей формального анализа позволяют исследователю получить значительно более краткие и удобные описания, чем те, которые могут быть сделаны с помощью сетей низкого уровня.

В сетях высокого уровня сложность моделей может быть разделена между структурой сети, надписями и описаниями. Это позволяет осуществлять описание значительно более сложных систем и анализировать процессы преобразования данных с помощью общепринятых математических выражений вместо сложного набора позиций, переходов и дуг.

Раскрашенные сети Петри, в отличие от обычных сетей Петри, позволяют описывать структуру системы в виде иерархии диаграмм.

Но у данного аппарата моделирования также не устранен ряд недостатков, которые присущи сетям Петри. К таким недостаткам можно отнести:

- необходимость знания разработчиком специфического языка описания моделей;
- отсутствие использования принципов объектно-ориентированного подхода;
- низкую гибкость и трудоемкость описания систем в случае их декомпозиции до уровня некоторых элементарных операций.

Раскрашенные сети Петри до сих пор часто применяются для моделирования сетей, применение данной методологии довольно ограничено, однако не исключает целесообразность использования моделей сетей Петри для решения некоторых специфических задач.

Пример применения метода сетей Петри при моделировании процессов в сложных организационно-технических системах. Рассмотрим более подробно некоторые варианты использования, описанного выше, метода для моделирования в задачах гражданской авиации.

Сети Петри и их модификации находят все большее применение при моделировании технологических процессов в сложных организационно-технических системах. Применение сетей Петри для моделирования предполетного досмотра пассажиров, ручной клади и багажа, являющегося одним из важнейших направлений обеспечения транспортной безопасности и верификации ее протокола [12].

Аналитически маркированная сеть Петри может быть представлена в следующем виде : $C=(P, T, I, O, M_0)$.

$$P=\{p_1,p_2,p_3,\dots,p_{28}\}$$

$$T=\{t_1,t_2,t_3,\dots,t_{23}\}$$

Начальная маркировка сети:

$$M_0=\{n,0,0,1,0\}$$

Сеть работает следующим образом. В соответствии с требованиями документов, все пассажиры допускаются на рейс после прохождения предполагаемого досмотра. В ходе досмотра в целях обеспечения транспортной безопасности осуществляются мероприятия по обследованию физических лиц, а также транспортных средств, багажа, ручной клади и другое, в целях обнаружения оружия, взрывчатых веществ или других устройств, в отношении которых установлен запрет на перемещение в зону транспортной безопасности.

Пассажир допускается на рейс, если по всем проверяемым атрибутам получает допуск. Пассажир не допускается на рейс, если хотя бы по одному из атрибутов он не соответствует требованиям.

При заданной маркировке сети переход t_2 становится разрешенным. При его срабатывании, что соответствует объявлению регистрации на рейс, фишка для позиции p_4 убирается, а одна фишка появляется в позиции p_2 . Это соответствует приглашению пассажиров к стойке регистрации. Становится разрешенным переход t_1 . При его срабатывании, что соответствует началу регистрации i -го пассажира, по одной фишке удаляются из позиций p_1, p_2 , а одна фишка помещается в позицию p_3 . Это соответствует процессу проверки документов у очередного пассажира. Становится разрешенным переход t_4 , соответствующий окончанию проверки документов у пассажира и принятию решения о соответствии или несоответствии документов требованиям для перевозки. При срабатывании перехода t_4 фишка из позиции p_3 убирается и либо одна фишка с вероятностью W_1 появляется в позиции p_5 (документы не соответствуют), либо по одной фишке с вероятностью W_2 (документы соответствуют) помещается в позиции p_6, p_7, p_8, p_9 . Если фишка появилась в позиции p_5 (документы не соответствуют), то разрешенным становится переход t_6 . При его срабатывании, что соответствует отказу в допуске пассажира на рейс по причине несоответствия документов, фишка убирается из позиции p_5 и по одной фишке появляется в позициях p_{10} (собирается информация о не допущенных на рейс пассажирах из-за несоответствия документов) и p_2 (следующий пассажир приглашается на регистрацию). При появлении по одной фишке в позициях p_6, p_7, p_8 , что соответствует необходимости продолжения проверки по атрибутам « p_i – пассажир», « r_i – ручная кладь» и « b_i – багаж», разрешенными становятся переходы t_6, t_7, t_8

соответственно. Появление фишки в позиции p_9 означает, что очередной пассажир по атрибуту « di документы» может быть допущен на рейс (отображается информация о ходе досмотра i -го пассажира – документы соответствуют требованиям).

Срабатывание перехода t_7 соответствует проверки наличия у пассажира ручной клади. При этом одна фишка убирается из позиции p_7 и, если у пассажира с вероятностью W_4 отсутствует ручная кладь, то одна фишка появляется в позиции p_{18} (отображается информация о ходе досмотра i -го пассажира, к которому нет претензий по атрибуту « ri – ручная кладь»). Наличие фишки в позиции p_{11} делает разрешенным переход t_{11} , его срабатывание соответствует началу проверки ручной клади. При этом фишка убирается из позиции p_{11} , и одна фишка появляется в позиции p_{16} , что соответствует процессу проверки ручной клади и делает разрешенным переход t_{16} , срабатывание которого соответствует окончанию проверки и принятию решения о соответствии или несоответствии ручной клади требованиям для перевозки. При срабатывании перехода t_{16} фишка убирается из позиции p_{16} . Если фишка появилась в позиции p_{22} (ручная кладь не соответствует), то разрешенным становится переход p_{20} . При его срабатывании, что соответствует отказу в допуске пассажира на рейс по причине несоответствия ручной клади, фишка убирается из позиции p_{22} и по одной фишке появляется в позициях p_{25} (собирается информация о недопущенных на рейс пассажирах из-за несоответствия ручной клади) и p_2 (следующий пассажир приглашается на регистрацию). Если фишка появилась в позиции p_{23} (ручная кладь соответствуют), то становятся разрешенным переход t_{21} . При срабатывании перехода t_{21} , одна фишка из позиции p_{23} убирается и одна фишка появляется в позиции p_{18} .

Таким образом, чтобы пассажир был допущен на рейс, необходимо, чтобы в позициях p_9 , p_{13} , p_{18} , p_{24} было по одной фишке. При этом становится разрешенным переход t_{18} , соответствующий допуску на рейс проверенного (i -го) пассажира. При его срабатывании по одной фишке убираются из позиций p_9 , p_{13} , p_{18} , p_{24} и одна фишка появляется в позиции p_{19} , где количество фишек будет соответствовать количеству пассажиров, прошедших регистрацию и допущенных на данный рейс. Переходы t_3 , t_5 , t_9 , t_{13} соответствуют сбросу информации об атрибутах проверяемого пассажира, если по одному из них пассажир не будет допущен на рейс. Какие из этих переходов будут срабатывать, зависит от того, какие из атрибутов были проверены и по какому из атрибутов пассажир получил отказ в допуске. При этом для срабатывания перехода t_3 необходимо наличия фишки в позициях p_9 и p_2 , перехода t_5 – в позициях p_{13} и p_2 , перехода t_9 – в позициях p_{18} и p_2 , перехода t_{13} – в позициях p_{24} и p_2 . При срабатывании переходов t_3 , t_5 , t_9 , t_{13} фишки убираются из позиций p_9 , p_{13} , p_{18} , p_{24} соответственно. При

срабатывании этих переходов в позиции p_2 фишка не убывает, так как эта позиция соединена с переходами t_3, t_5, t_9, t_{13} двунаправленными дугами.

Предложенная модель сетей Петри для процесса обработки заявок на транспортном-экспедиционном обслуживании для предполетного досмотра пассажиров, ручной клади и багажа имеет достоинства в том, что она позволяет исследовать работоспособность моделируемой системы, оптимальность ее структуры, эффективность процесса ее функционирования, оценить временные показатели, а также возможность достижения в процессе функционирования определенных состояний.

Сети Петри, благодаря своей удобной форме представления, развитой семантике, а также средств моделирования и анализа, идеально подходят для моделирования систем транспортного комплекса. Модели сетей Петри позволяют исследовать работоспособность моделируемых систем, оптимальность их структуры, эффективность процесса их функционирования, а также возможность достижения в процессе функционирования определенных состояний. Сети Петри и их обобщения являются удобным и мощным средством моделирования процессов, позволяют наглядно представить динамику функционирования систем и составляющих их элементов.

5. Имитационное моделирование

5.1. Понятие имитационного моделирования

Несмотря на большие достижения математического моделирования, многие реальные ситуации невозможно адекватно представить с помощью соответствующих математических моделей. В одних случаях этому мешает определенная «жесткость» математики как языка описания и представления событий и явлений. Кроме того, даже если есть возможность формализовать рассматриваемую жизненную ситуацию посредством построения математической модели, полученная на ее основе задача оптимизации может быть слишком сложной для современных алгоритмов решения задач этого класса.

Альтернативой математическому моделированию сложных систем может служить имитационное моделирование (ИМ). Этот вид моделирования часто является наилучшим (если не единственным) способом исследования реальных систем. Термин «имитационное моделирование» означает, что мы имеем дело с такими моделями, с помощью которых нельзя заранее вычислить или предсказать поведение системы, а для предсказания поведения системы необходим вычислительный эксперимент (имитация) на математической модели при заданных исходных данных [3,4].

Различие между математической и имитационной моделями заключается в том, что в последней вместо явного математического описания взаимоотношения между входными и выходными переменными реальная система разбивается на ряд достаточно малых (в функциональном отношении) элементов или модулей. Затем поведение исходной системы имитируется как поведение совокупности этих элементов, определенным образом связанных (путем установления соответствующих взаимосвязей между ними) в единое целое. Вычислительная реализация такой модели начинается с входного элемента, далее проходит по всем элементам, пока не будет достигнут выходной элемент модели.

Имитационные модели принято классифицировать по следующим наиболее распространенным признакам и уже описанным признакам:

- по способу взаимодействия с пользователем;
- способу изменения модельного времени;
- цели эксперимента.

Классификация имитационных моделей схематично показана на рис. 26.



Рис. 26. Классификация имитационных моделей

По способу взаимодействия с пользователем имитационные модели могут быть автономными и интерактивными. Автономные модели не требуют вмешательства исследователя после определения режима

моделирования и задания исходных данных, взаимодействие пользователя с такими моделями сводится только к вводу исходной информации и управлению началом и окончанием работы моделей. Интерактивные модели предусматривают диалог с пользователем в том или ином режиме в соответствии со сценарием моделирования, позволяющие пользователю приостанавливать сеанс моделирования/изменять значения параметров модели, корректировать перечень регистрируемых данных и т.д.

Как было сказано ранее существует два механизма изменения модельного времени: продвижение времени от события к событию и продвижение времени с постоянным шагом.

Процесс построения имитационных моделей представляет собой последовательное выполнение этапов моделирования.

Названные выше этапы имитационного исследования редко выполняются в строго заданной последовательности, начиная с определения проблемы и заканчивая документированием. В ходе имитационного исследования могут быть сбои в прогонах модели, ошибочные допущения, от которых впоследствии приходится отказываться, т.е. на каждом этапе возможно возвращение назад, к предыдущим этапам. Именно такой итеративный процесс даёт возможность получить модель, которая позволяет принимать решения.

Вычислительные аспекты имитационных моделей обычно сравнительно несложные, но, как правило, очень трудоемкие. Поэтому реализация таких моделей подразумевает использование вычислительной техники.

Имитационные модели значительно гибче в представлении реальных систем, чем их математические «конкуренты». Причина такой гибкости заключается в том, что при имитационном моделировании исходная система рассматривается на элементарном уровне, в то время как математические модели стремятся описать системы на глобальном, как можно более общем уровне.

Но за гибкость имитационных моделей приходится платить высокими требованиями к потребляемым временным и вычислительным ресурсам. Поэтому реализация некоторых имитационных моделей даже на современных быстрых и высокопроизводительных компьютерах может быть очень медленной.

5.2. Преимущества и недостатки имитационного моделирования.

Таким образом, имитационное моделирование является мощным инструментом исследования поведения реальных систем. Методы имитационного моделирования позволяют собрать необходимую

информацию о поведении системы путем создания ее компьютеризированной модели. Эта информация используется затем для проектирования системы.

Основное достоинство ИМ:

- возможность описания поведения компонент (элементов) процессов или систем на высоком уровне детализации;
- отсутствие ограничений между параметрами ИМ и состоянием внешней среды;
- возможность исследования динамики взаимодействия компонент во времени и пространстве параметров системы [6].

Эти достоинства обеспечивают имитационному методу широкое распространение.

Рекомендуется использовать имитационное моделирование в следующих случаях:

1. Если не существует законченной постановки задачи исследования и идет процесс познания объекта моделирования. Имитационная модель служит средством изучения явления.
2. Если аналитические методы имеются, но математические процессы сложны и трудоемки, имитационное моделирование дает более простой способ решения задачи.
3. Когда, кроме оценки влияния параметров (переменных) процесса или системы, желательно осуществить наблюдение за поведением компонент (элементов) процесса или системы в течение определенного периода.
4. Когда имитационное моделирование оказывается единственным способом исследования сложной системы из-за невозможности наблюдения явлений в реальных условиях (реакции термоядерного синтеза, исследования космического пространства).
5. Когда необходимо контролировать протекание процессов или поведение систем путем замедления или ускорения явлений в ходе имитации.
6. При подготовке специалистов для новой техники, когда на имитационных моделях обеспечивается возможность приобретения навыков в эксплуатации новой техники.
7. Когда изучаются новые ситуации в поведении реальных процессов и систем. В этом случае имитация служит для проверки новых стратегий и правил проведения натуральных экспериментов.

8. Когда особое значение имеет последовательность событий в проектируемых процессах и системах, и модель используется для предсказания узких мест в их функционировании [4].

Однако ИМ наряду с достоинствами имеет и недостатки:

- разработка хорошей ИМ часто обходится дороже создания аналитической модели и требует больших временных затрат;
- может оказаться, что ИМ неточна (что бывает часто), и исследователь не в состоянии измерить степень этой неточности;
- зачастую исследователи обращаются к ИМ, не представляя тех трудностей, с которыми они встретятся и совершают при этом ряд ошибок методологического характера [].

И тем не менее, ИМ является одним из наиболее широко используемых методов при решении задач синтеза и анализа сложных процессов и систем.

Применительно к вычислительным сетям их имитационные модели воспроизводят: процессы генерации сообщений приложениями; разбиение сообщений на пакеты и кадры определенных протоколов; задержки, связанные с обработкой сообщений, пакетов и кадров внутри операционной системы; процесс получения доступа компьютером к разделяемой сетевой среде; процесс обработки поступающих пакетов маршрутизатором и т.д. При имитационном моделировании сети не требуется приобретать дорогостоящее оборудование – его работы имитируется программами, достаточно точно воспроизводящими все основные особенности и параметры такого оборудования.

Преимуществом имитационных моделей является возможность подмены процесса смены событий в исследуемой системе в реальном масштабе времени на ускоренный процесс смены событий в темпе работы программы. В результате за несколько минут можно воспроизвести работу сети в течение нескольких дней, что дает возможность оценить работу сети в широком диапазоне варьируемых параметров.

Результатом работы имитационной модели являются собранные в ходе наблюдения за протекающими событиями статистические данные о наиболее важных характеристиках сети: временах реакции, коэффициентах использования каналов и узлов, вероятности потерь пакетов и т.п.

6. Системы имитационного моделирования

Существуют специальные, ориентированные на моделирование вычислительных сетей программные системы, в которых процесс создания модели упрощен. Такие программные системы сами генерируют модель сети

на основе исходных данных о ее топологии и используемых протоколах, об интенсивностях потоков запросов между компьютерами сети, протяженности линий связи, о типах используемого оборудования и приложений.

Программные системы моделирования могут быть узко специализированными и достаточно универсальными, позволяющие имитировать сети самых различных типов. Качество результатов моделирования в значительной степени зависит от точности исходных данных о сети, переданных в систему имитационного моделирования. Используя моделирование при проектировании или реинжиниринге вычислительной системы, теперь можно сделать следующее: оценить пропускную способность сети и ее компонентов, определить узкие места в структуре вычислительной системы; сравнить различные варианты организации вычислительной системы; осуществить перспективный прогноз развития вычислительной системы; предсказать будущие требования по пропускной способности сети, используя данные прогноза; оценить требуемое количество и производительность серверов в сети; сравнить различные варианты модернизации вычислительной системы; оценить влияние на вычислительную систему модернизации ПО, мощности рабочих станций или серверов, изменения сетевых протоколов.

Исследование параметров вычислительной системы при различных характеристиках отдельных компонентов позволяет выбрать сетевое и вычислительное оборудование с учетом производительности, качества обслуживания, надежности и стоимости. Моделирование позволяет минимизировать стоимость оборудования, предназначенного для использования в вычислительной системе. Моделирование становится эффективным при числе рабочих станций 50-100, а когда их более 300, общая экономия средств может составить 30 - 40% от стоимости проекта.

Программные системы моделирования сетей - инструмент, который может пригодиться любому администратору корпоративной сети, особенно при проектировании новой сети или внесении кардинальных изменений в уже существующую. Продукты данной категории позволяют проверить последствия внедрения тех или иных решений еще до оплаты приобретаемого оборудования. Конечно, большинство из этих программных пакетов стоят достаточно дорого, но и возможная экономия может быть тоже весьма ощутимой.

Программы имитационного моделирования сети используют в своей работе информацию о пространственном расположении сети, числе узлов, конфигурации связей, скоростях передачи данных, используемых протоколах и типе оборудования, а также о выполняемых в сети приложениях.

Обычно имитационная модель строится не с нуля. Существуют готовые имитационные модели основных элементов сетей: наиболее

распространенных типов маршрутизаторов, каналов связи, методов доступа, протоколов и т.п. Эти модели отдельных элементов сети создаются на основании различных данных: результатов тестовых испытаний реальных устройств, анализа принципов их работы, аналитических соотношений. В результате создается библиотека типовых элементов сети, которые можно настраивать с помощью заранее предусмотренных в моделях параметров.

Системы имитационного моделирования обычно включают также набор средств для подготовки исходных данных об исследуемой сети - предварительной обработки данных о топологии сети и измеренном трафике. Эти средства могут быть полезны, если моделируемая сеть представляет собой вариант существующей сети и имеется возможность провести в ней измерения трафика и других параметров, нужных для моделирования. Кроме того, система снабжается средствами для статистической обработки полученных результатов моделирования.

Систем динамического моделирования вычислительной системы достаточно много, они разрабатываются в разных странах. Кроме того, зачастую развитые системы диагностирования установленной вычислительной системы (интеллектуальные кабельные тестеры, сканеры, анализаторы протоколов) также причисляют к системам моделирования, что не соответствует действительности.

Можно провести классификацию систем по двум связанным критериям: цена и функциональные возможности. Нужно отметить, что функциональные возможности систем моделирования жестко связаны с их ценой. Анализ предлагаемых на рынке систем показывает, что динамическое моделирование вычислительных систем - дело весьма дорогостоящее. Все системы динамического моделирования могут быть разбиты на две ценовые категории:

- Недорогие - (сотни и тысячи долларов).
- High-end - (десятки тысяч долларов, в полном варианте - сто и более тысяч долларов).

Недорогие системы отличаются от дорогих тем, насколько подробно удастся в них описать характеристики отдельных частей моделируемой системы. Они позволяют получить лишь приблизительные или оценочные результаты, не дают статистических характеристик и не предоставляют возможности проведения подробного анализа системы.

Системы класса high-end позволяют собирать исчерпывающую статистику по каждому из компонентов сети при передаче данных по каналам связи и проводить статистическую оценку полученных результатов. По функциональности системы моделирования, используемые при исследовании вычислительных систем, могут быть разбиты на два основных класса:

1. Системы, моделирующие отдельные элементы (компоненты) системы.
2. Системы, моделирующие вычислительную систему целиком.

В Таблице 1 приведены характеристики нескольких популярных систем имитационного моделирования различного класса - от простых программ, предназначенных для установки на персональном компьютере, до мощных систем, включающих библиотеки большинства имеющихся на рынке коммуникационных устройств и позволяющих в значительной степени автоматизировать исследование сети.

Таблица 1. Характеристики систем имитационного моделирования различного класса

Компания и продукт	Стоимость (долл)	Тип сети	Требуемые ресурсы	Примечания
American HYTech, Prophesy	1495	ЛС	8МбОП, 6 Мбдиск, DOS, Windows, OS/2	Оценивание производительности при работе с текстовыми и графическими данными по отдельным сегментам и сети в целом
CACI Product, COMNET III	34500-39500	ЛС, ГС	32 Мб ОП, 100 Мбдиск, Windows, Windows NT, OS/2, Unix	Моделирует сети X.25, ATM, Frame Relay, связи LAN-WAN, SNA, DECnet, протоколы OSPF, RIP. Доступ CSMA/CD и токентный доступ, FDDI и др. Встроенная библиотека маршрутизаторов 3COM, Cisco, DEC, HP, Wellfleet, ...
Make System, NetMaker XA	6995-14995	ЛС, ГС	128 МбОП, 2000Мбдиск, AIX, Sun OS, Sun Solaris	Проверка данных о топологии сети; импорт информации о трафике, получаемой в

				реальном времени
NetMagic System, StressMagik		ЛС	2 МБ ОП, 8 МБ диск, Windows	Поддержка стандартных тестов измерения производительности; имитация пиковой нагрузки на файл-сервер
Network Analysis Center, MIND	9400-70000	ГС	8 МБ ОП, 65 МБ диск, DOS, Windows	Средство проектирования, оптимизации сети, содержит данные о стоимости типичных конфигураций с возможностью точного оценивания производительности
Network Design and Analysis Group, AutoNet/Designer		ГС	8 МБ ОП, 40 МБ диск, Windows, OS/2	Определение оптимального расположения концентратора в ГС, возможность оценки экономии средств за счет снижения тарифной платы, смены поставщика услуг и обновления оборудования; сравнение вариантов связи через ближайшую и оптимальную точку доступа, а также через мост и местную телефонную сеть
Network Design and Analysis Group, AutoNet/MeshNET		ГС	8 МБ ОП, 40 МБ диск, Windows, OS/2	Моделирование полосы пропускания и оптимизация расходов на организацию ГС путем имитации поврежденных

				линий, поддержка тарифной сетки компаний AT & T, Sprint, WiTel, Bell
Network Design and Analysis Group, AutoNet/Performance-1		ГС	8 МБ ОП, 1 Мб диск, Windows, OS/2	Моделирование производительности иерархических сетей путем анализа чувствительности к длительности задержки, времени ответа, а также узких мест в структуре сети
Network Design and Analysis Group, AutoNet/Performance-3		ГС	8 МБ ОП, 3 Мб диск, Windows, OS/2	Моделирование производительности многопротокольных объединений локальных и глобальных сетей; оценивание задержек в очередях, прогнозирование времени ответа, а также узких мест в структуре сети; учет реальных данных о трафике, поступающих от сетевых анализаторов
System& Networks, BONES	20000-40000	ЛС, ГС	32 МБ ОП, 80 Мб диск, Sun OS, Sun Solaris, HP-UX	Анализ воздействия приложений клиент-сервер и новых технологий на работу сети
MIL3, Opnet	16000-40000		16 МБ ОП, 100 Мб диск, DEC AXP, Sun OS, Sun Solaris, HP-UX	Имеет библиотеку различных сетевых устройств, поддерживает анимацию, генерирует карту сети, моделирует

				полосу пропускания.
--	--	--	--	---------------------

На сайте Национального общества имитационного моделирования [13] представлен расширенный список программных продуктов и систем имитационного моделирования. Ниже представлены некоторые из них.

Программные продукты:

AGNES, система

AGNES (AGent NEtwork Simulator) – система имитационного моделирования больших систем с дискретными событиями. Система AGNES является кроссплатформенной (JAVA приложение), возможен распределенный запуск (на локальной сети или вычислительном кластере) и полунатурное моделирование (при наличии реальных объектов, поддерживающих спецификацию FIPA). Область применения: локальные компьютерные сети, беспроводные сенсорные сети, вычисления на суперЭВМ. При разработке использован Java Agent Development Framework (JAVA не ниже версии 1.6, JADE с 4-ой версии). AGNES распространяется бесплатно по лицензии LGPL, Разработчик системы: Подкорытов Дмитрий Игоревич (d.podkorytov@mail.ru), Институт вычислительной математики и математической геофизики СО РАН. Подробное описание AGNES в статье Подкорытова Д.И. «Агентно-ориентированная среда моделирования сетевых систем AGNES» // Ползуновский вестник, 2012. № 2/1, С. 94-99. (электронный доступ).

AnyLogic, система моделирования

Система AnyLogic поддерживает три технологии создания имитационных моделей: процессно-ориентированный (дискретно-событийный), системно динамический и агентный, а также любую их комбинацию. Графический интерфейс AnyLogic, инструменты и библиотеки позволяют быстро создавать модели для широко спектра задач от моделирования производства, логистики, бизнес-процессов до стратегических моделей развития компании и рынков. AnyLogic стал корпоративным стандартом на бизнес-моделирование во многих транснациональных компаниях, широко используется в образовании.

Сайт: <http://www.anylogic.ru/>

Arena, система

Arena - система дискретного моделирования. Сфера основных приложений системы - имитационное моделирование производственных технологических процессов и операций, складской учет, банковская деятельность, оптимизация обслуживания клиентов в сфере услуг, транспортные задачи.

Разработчик: Rockwell Automation Inc., Wexford, PA, США. Сайт: <http://www.arenasimulation.com/>

Boson NetSim, система

Система Boson NetSim - это коммерческий симулятор, по сути своей эмулятор, сетевых устройств компании Cisco на основе Cisco IOS. Данная система позволяет получить практические знания по работе с сетевыми устройствами, начиная от обычных управляемых свичей, и заканчивая роутерами 7-го поколения. В поставку включается утилита для моделирования сети. В ней можно смоделировать любой тип сети или взять готовую сеть из примеров. Сертификация специалистов Cisco CCNP (Cisco Certified Network Professional) проходит именно в этой программе. Система по причине низкой стоимости при достаточном функционале используется в более чем 250 университетах мира.

Сайт: <http://www.boson.com/netsim-cisco-network-simulator>

Сайт: <http://www.boson.com/network-simulator/ccnp-cisco-network-simulator>

Galatea, платформа

Galatea - система мульти-агентного моделирования, язык мульти программирования, симуляционная платформа. Galatea распространяется, как Open Source программное обеспечение под GNU General Public License. Galatea поддерживается Центром моделирования и имитационного моделирования (Centre for Simulation and Modelling), университет Анд, Венесуэла.

Сайт: <http://galatea.sourceforge.net/Home.htm>

GloMoSim, система

GloMoSim (Global Mobile Information System Simulator) - система имитационного моделирования, рассчитанная на масштабируемость. GloMoSim - бесплатная академическая версия пакета QalNET, разрабатываемого компанией Scalable Network Technologies. Версия GloMoSim не включает в себя большинство современных протоколов и предназначена исключительно для обучения. GloMoSim написана на языке PARSEC (версия языка Си, рассчитанная на параллельные вычисления). PARSEC разработан в Parallel Computing Laboratory at UCLA. Крупноформатность этого языка обеспечивает работу GloMoSim в разных операционных системах: Solaris, GNU/Linux, FreeBSD, MS Windows.

Сайт: <http://www.pcl.cs.ucla.edu>.

GPSS - Future, система

Система GPSS - Future развивает Object GPSS в плане более гибкой работы со списком будущих событий. Такой список рассматривается, как наследник списка пользователя. Когда не удастся продвинуть ни одну заявку из списка текущих событий, то в любой версии GPSS должна вызываться процедура продвижения к новому моменту модельного времени. В GPSS - Future, как правило, вначале из списка будущих событий извлекаются заявки, чье время ожидания уже истекло, а, если их нет, то извлекаются заявки с минимальным временем завершения ожидания. В последнем случае, продвигается вперед текущее время моделирования. Эта процедура используется по умолчанию. При желании можно переписать процедуру продвижения к новому моменту модельного времени. Имеется набор таких возможных процедур. Для генерации заявок используются процедура Future.NewWaitProc (инициализация) и блок Future.NewWait (генерация). Задержку выполняет блок Future.Wait (Future – это список будущих событий, который предлагается системой по умолчанию прямо в шаблоне).

Разработчик: Королёв Анатолий Георгиевич, Северодонецкий Технологический институт, Северодонецк, Украина.

Сайты: <http://objectgpss.narod.ru/>;

<http://objectgpss.ucoz.ru/>

GPSS/H, система

Система GPSS/H - это среда моделирования общего назначения, охватывает области как дискретного, так и непрерывного моделирования.

Разработчик: компания Wolverine Software Corp., США.

Сайт: <http://www.wolverinesoftware.com>

GPSS World, система

Система GPSS World - это среда моделирования общего назначения, охватывает области как дискретного, так и непрерывного моделирования. GPSS World включает PLUS-язык программирования нижнего уровня моделирования. Моделирование с использованием PLUS выражений может быть включено почти везде в GPSS программы, в любом блоке или процедуре вызова. Язык PLUS позволяет программно управлять размещением результатов. Система GPSS World разрешает многозадачность, позволяя нескольким имитационным процессам выполняться одновременно. Разработчик: компания Minuteman Software Corp., США.

Сайт: <http://www.minutemansoftware.com/>

GPSS World - Расширенный редактор, система

Расширенный редактор GPSS World - это универсальная система имитационного моделирования, охватывающая весь цикл имитационных исследований, от постановки задачи до документирования результатов. Основные особенности системы: высокий уровень интерактивности при проведении исследования, упрощение разработки моделей и проведения исследований, большой объем текстовой документации и оперативных подсказок. Введено понятие имитационного проекта и имитационного приложения (независимого от редактора исполняемого кода модели). Проекты и приложения имеют интегрированные базы данных моделей и результатов исследования. Имитационное приложение может быть отделено от среды разработки и представлено в виде исполняемого модуля. Возможно использование клиентской части и системы GPSS World как на одном компьютере, так удаленное в ЛВС. Возможна организация облачного моделирования в сети Интернет. Имеется бесплатная студенческая версия системы (работает только при наличии студенческой лицензии GPSS World, отключена опция создания автономных исполняемых EXE модулей, нет удаленного и облачного использования GPSS World). Для академического и коммерческого использования имеется несколько видов лицензий. Более подробная информация о продукте и порядке лицензирования на сайте разработчика.

Разработчик: компания ООО «Элина-компьютер», Казань, Россия.

Сайт: <http://www.elina-computer.ru/>

iThink и Stella, системы

Программное обеспечение iThink и Stella предназначено для моделирования непрерывно-дискретных процессов. По сравнению с iThink в Stella имеются возможности по построению моделей большой размерности и их свертки. Операционные среды - Windows and Macintosh.

Разработчик: Isee systems Inc., Lebanon, NH, США.

Сайт: <http://www.iseesystems.com/>

iWebsim, программа

iWebsim – веб-приложение для имитационного моделирования динамических систем. Реализованы парадигмы системной динамики, дискретно-событийного и агентного моделирования. Для работы с iWebsim достаточно иметь веб-браузер, поддерживающий HTML5, и выход в Интернет или локальную сеть (если доступ к серверу iWebsim осуществляется только в рамках локальной сети). Зарегистрированный пользователь iWebsim имеет возможность создавать свои проекты, сохранять их в БД и осуществлять на их основе имитационные эксперименты (включая простые имитационные эксперименты, оптимизационные и статистические эксперименты, а также эксперименты по анализу чувствительности моделируемой системы). Для

физических лиц доступ к веб-приложению iWebsim осуществляется бесплатно. Для корпоративных клиентов доступ к веб-приложению осуществляется по предварительному согласованию с разработчиком. Разработчик: Балухто Алексей Николаевич, д.т.н., ООО «НПЦ «Интелком», г. Королев, Московская обл.
Сайт: <http://www.iwebsim.ru/>

MaDKit, мультиагентная платформа разработки

MaDKit – мультиагентная платформа разработки, написана на Java. MaDKit позволяет быстро создавать пользовательские приложения для моделирования мультиагентных систем. MaDKit является свободно распространяемым программным обеспечением под GNU GPL.

Разработчики: Gutknecht Olivier, Ferber Jacques, Michel Fabien; Франция.
Сайт: <http://www.madkit.org/>

NetLogo, среда мультиагентного моделирования

NetLogo предназначена для моделирования ситуаций и феноменов, происходящих в природе и обществе. Удобное средство для моделирования сложных, развивающихся во времени систем. Позволяет давать указания сотням и тысячам независимых «агентов», действующим параллельно. В российском образовании NetLogo используется для демонстрации сетевых феноменов и для моделирования социальных феноменов в учебных курсах по менеджменту. NetLogo поддерживается Национальным научным фондом США (National Science Foundation). NetLogo является свободно распространяемым программным обеспечением и действует на различных платформах. Реализация – на Scala и Java. Автор NetLogo – Ури Виленский (Uri Wilensky), директор Northwestern University's Center for Connected Learning and Computer-Based Modeling.

Сайт: <http://ccl.northwestern.edu/netlogo/>

NetSim, система

NetSim – система для дискретно-событийного моделирования сетей. Разработчик: компания Tetcos, Бангалор, Индия.

Сайт: <http://www.tetcos.com>.

NS-3

NS-3 - система дискретно-событийного моделирования сетевых структур. NS-3 является свободно распространяемым программным обеспечением. Поддерживаемые платформы – GNU/Linux, FreeBSD, Mac OS X. Система NS-3 разработана на C++ и Python. Первая версия системы вышла в 2008 году. NS-3 является свободно распространяемым программным обеспечением. Разработчики NS-3 – Том Хендерсон (Tom Henderson),

Джордж Райли (George Riley), Салли Флойд (Sally Floyd), Сумит Рой (Sumit Roy). Финансирование разработки осуществлял американский Национальный научный фонд (U.S. National Science Foundation – NSF).

Site: <https://www.nsnam.org/>

Object GPSS, система

Object GPSS – инструментальное средство для написания моделей в стиле GPSS непосредственно на языке Delphi (Object Pascal). Каждая модель на Object GPSS представляет собой Include-файл (Model.pas), содержащий описание всех объектов модели и набор из 6 процедур: Initial, CloseAllObj, ResetAll, ModelTxt, Report, Modeling. Практически все части модели, кроме «начинки» процедуры ModelTxt, создаются программой – конвертером. Для создания исполняемой модели следует скомпилировать модель вместе с остальными стандартными частями проекта. Полученный exe - файл является моделью конкретной системы и с ней можно проводить эксперименты. В системе легко расширять набор команд и блоков для моделирования. Модели на Object GPSS выглядят более естественно, чем у традиционных версий GPSS. Логика построения моделей более прозрачна и более соответствует логики обычных программ.

Разработчик: Королёв Анатолий Георгиевич, Северодонецкий Технологический институт, Северодонецк, Украина.

Сайты: <http://objectgpss.ucoz.ru/>

OMNeT++ (OMNeT++ INET Framework), система

Система OMNeT++ - это среда имитационного моделирования дискретных событий и состояний с открытым исходным кодом. Изменение состояния моделируемой системы происходит в дискретные моменты времени по списку будущих событий (future event list), отсортированных по времени. Событием может быть: начало передачи пакета, тайм-аут и т. п. События происходят на основе выполнения простых модулей (simple module). У такого модуля есть функции инициализации, обработки сообщения, действия и завершения работы. Обмен сообщениями между модулями осуществляется по каналам (channel), с которыми модули соединены своими шлюзами (gate), или непосредственно через шлюзы. Шлюз может быть входящим или исходящим, соответственно, для приема и отправки сообщений. Основная область применения OMNeT++ – моделирование сетей передачи данных, ИТ систем и бизнес процессов. Компоненты OMNeT++ написаны на C++. На базе среды моделирования OMNeT++ построен симулятор различных протоколов беспроводных сенсорных сетей Castalia. В нём также реализована модель, соответствующая стандарту IEEE 802.15.4. На базе рассматриваемой среды моделирования существуют библиотеки INETMANET и MiXiM, которые позволяют создавать модели беспроводных сенсорных сетей, но на

текущий момент готовые модели отсутствуют. Достоинства системы OmNET++:

- а) свободное распространение;
- б) реализация моделей на C++;
- в) наличие графического режима;
- г) широкий диапазон применений;
- д) подробная документация.

Сайт: <http://www.omnetpp.org>

OpenGPSS

OpenGPSS - on-line discrete-event simulation system, working with the language GPSS, designed for automatic distribution of tasks between the nodes of the cluster for parallel computing and the subsequent assembly of the results. Developer: Kiev centre of simulation, Kiev, Ukraine. Site: <http://simulation.kiev.ua/>

OpenMVLShell, среда моделирования

OpenMVLShell - открытая среда для моделирования сложных динамических систем (аналогичная OpenModelica). Среда представляет собой набор модулей, связанных с решением математических задач, возникающих при моделировании многокомпонентных сложных динамических систем. Пользователь может заменить существующий компонент, не меняя остальных, и проверить правильность и эффективность предлагаемых собственных решений. Авторский коллектив разработчиков – Исаков А.А., Сениченков Ю.Б., Санкт-Петербург, Россия. Сайт: <https://dcn.ftk.spbstu.ru/index.php?id=275>

OPNET Modeler (с 2012 года позиционируется как SteelCentral), система

Система Opnet Modeler предлагает пользователям графическую среду для создания, выполнения и анализа событийного моделирования сетей связи. Система может быть использована для проверки протоколов связи, анализа взаимодействий протоколов, оптимизации и планирования сети. Возможно выполнять проверку правильности аналитических моделей и описание протоколов. После моделирования пользователь получает: прогнозируемые задержки между конечными и промежуточными узлами сети, пропускные способности каналов, коэффициенты использования сегментов, буферов и процессоров источники задержек и узких мест сети. Система оперирует с узлами трех типов - процессорными, узлами-маршрутизаторами и коммутаторами. Узлы могут присоединяться с помощью портов к коммуникационным каналам любого типа, от каналов локальных сетей до спутниковых линий связи. Узлы и каналы могут характеризоваться средним временем наработки на отказ и средним временем восстановления для

моделирования надежности сети. Моделируется не только взаимодействие компьютеров в сети, но и процесс разделения процессора каждого компьютера между его приложениями. Ключевые возможности:

1. быстрый дискретный движок моделирования событий; библиотека моделей протоколов и устройств с исходными кодами (OPNET Mode);
2. объектно-ориентированное и иерархическое моделирование;
3. событийное, гибридное и опциональное аналитическое моделирование;
4. 32- и 64-битное ядро параллельного моделирования;
5. поддержка параллельных вычислений для распределенного моделирования; опция System-in-the-Loop для сопряжения моделей с «живыми системами»; реалистичное моделирование и анализ приложений;
6. открытый интерфейс для интеграции с внешними файлами объектов; встроенный графический интерфейс отладки и анализа.

Разработчик: Riverbed Technology, San Francisco, CA 94107, USA.

UML2 SP, язык имитационного моделирования

UML2 SP – язык объектно-ориентированного имитационного моделирования. Поддерживает первые три этапа имитационного исследования:

- определение целей моделирования, определение требований к программе-симулятору;
- концептуальное моделирование;
- формальное описание.

Частично поддерживается четвертый этап – программирование. Основной акцент в языке делается на концептуальном моделировании. Позволяет применять методологию Unified Process для разработки программ-симуляторов. Позиционируется как язык научного имитационного моделирования. Может быть реализован на UML-редакторах, поддерживающих профили UML. Проект доступен в открытом коде на сервисе GitHub.

Разработчик: Гурьянов Василий Иванович, Филиал Санкт-Петербургского государственного экономического университета, Чебоксары, Чувашская Республика.

Сайт: <https://vgurianov.github.io/uml-sp/>

WebGPSS, система

Разработка WebGPSS осуществлена Стокгольмской школой высшей экономики. Руководитель проекта профессор Ингольф Сталл. Эта система предназначена для изучения языка GPSS и разработки простейших имитационных моделей при работе в сети интернет. Сейчас WebGPSS - профессиональная система удаленного обучения GPSS. Языком имитационного моделирования в WebGPSS служит модернизированная версия Micro-GPSS. Основными отличиями являются графический интерфейс для работы в сети и некоторые педагогические упрощения. Суть обучения состоит в последовательном выполнении 26 онлайн - уроков по GPSS.

Сайт: <http://www.webgpss.com/>

МВТУ, среда моделирования

Программный комплекс «МВТУ» (SimInTech) предназначен для исследования динамики и проектирования разнообразных систем и устройств. По своим возможностям является альтернативой зарубежным программным продуктам Simulink, VisSim.

Программный комплекс «МВТУ» применяется для проектирования систем автоматического управления, следящих приводов и роботов-манипуляторов, ядерных и тепловых энергетических установок, а также для решения нестационарных краевых задач (теплопроводность, гидродинамика и др.). Используется в учебном процессе, позволяя моделировать различные явления в физике, электротехнике, в динамике машин и механизмов, в астрономии.

Может функционировать в многокомпьютерных моделирующих комплексах, в том числе и в режиме удаленного доступа к технологическим и информационным ресурсам.

Авторский коллектив разработчиков - Козлов О.С., Кондаков Д.Е., Скворцов Л.М., Тимофеев К.А., Ходяковский В.В., Щекатуров А.М., Москва, Россия.

Сайт: <http://3v-services.com>

Рассмотрим более подробно одну из часто встречающейся в учебном процессе систему моделирования.

Система Arena

Система имитационного моделирования *Arena* - основной программный продукт Systems Modeling. Корпорация Systems Modeling была основана в 1982 г. Деннисом Педгеном, автором SIMAN – первого промышленно-ориентированного общецелевого языка имитационного моделирования. В настоящее время область деятельности Systems Modeling включает в себя

имитационное моделирование и разработку технологического программного обеспечения [3,13].

Система Arena позволяет моделировать виды деятельности, представленные на рис. 26. В том числе, Arena позволяет моделировать сети и системы телекоммуникаций.

Arena снабжена удобным объектно-ориентированным интерфейсом и обладает возможностями по адаптации к различным предметным областям. Система не требует написания программного кода и исключительно проста в использовании, но требует значительного времени для освоения и достаточно глубоких знаний теории вероятностей, математической статистики, теории систем массового обслуживания и сетей Петри.

Для отображения результатов моделирования используется анимационная система Cinema animation. Интерфейс Arena включает в себя средства для работы с данными, в том числе электронные таблицы, базы данных, ODBC, OLE, поддержку формата DXF.

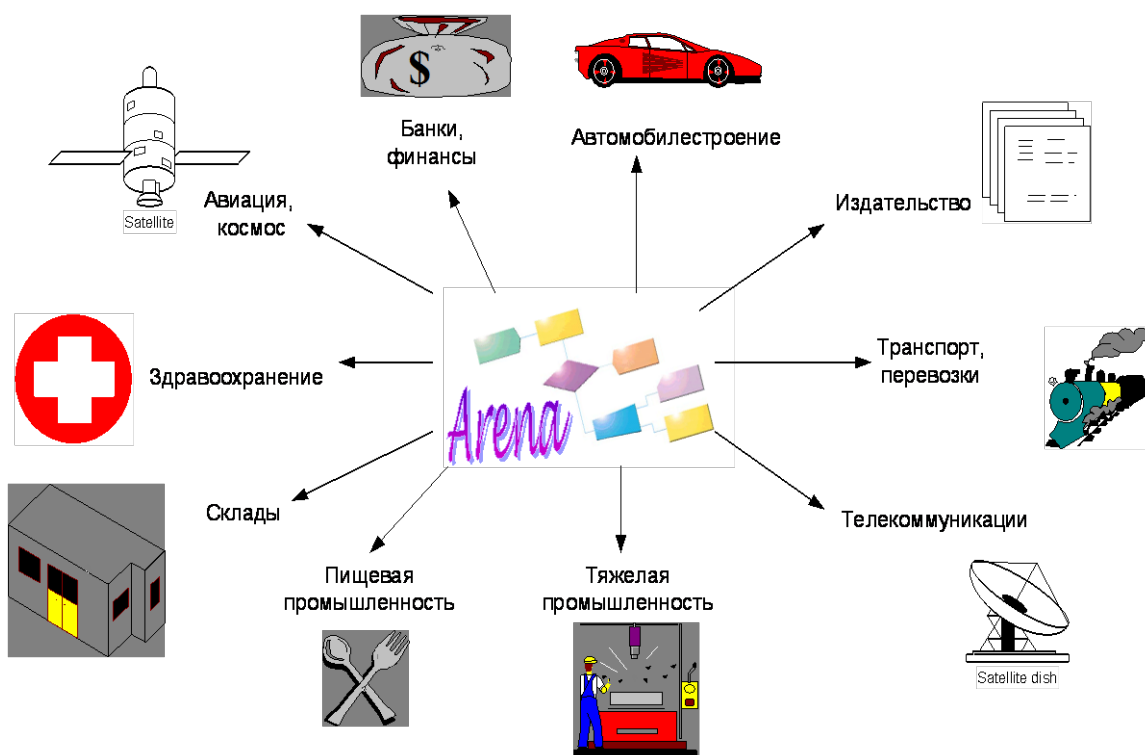


Рис. 26. Области применения Arena

Система имитационного моделирования Arena включает:

- двухмерный графический редактор;
- трехмерный графический редактор (пакет 3D player);

- редакторы временных шаблонов и расписаний;
- редактор символов и библиотека графических заготовок;
- связь с библиотекой графических заготовок и буфером обмена Microsoft.

Arena позволяет использовать дискретное, непрерывное, а также совмещенное дискретно-непрерывное моделирование.

Данный программный продукт поддерживает возможность взаимодействия с пакетом VBA Visual Basic for Applications корпорации Microsoft; объектной моделью ActiveX для внешнего управления, доступом через ADO/ODBC к базам данных (Oracle, Access, Excel, SQL); поддерживает импорт файлов из пакета AutoCad (в формате dxf); импорт данных из пакета Visio; импорт данных из пакета Blue Pumpkin Workforce, коммуникация между отдельными процессами.

При моделировании процессов и систем в Arena, используются три строительные панели:

1. *Basic Process Panel* – модули панели являются фундаментом для создаваемых моделей. Объекты данной панели инструментов состоят из графических модулей: Create, Dispose, Process, Deeside, Separate, Batch, Assign, Record и модулей данных: Entity, Resource, Queue, Variable, Schedule, и Set.
2. *Advanced Process Panel* – панель усовершенствованных процессов; позволяет моделировать более сложные процессы. Если в Basic Process Panel один модуль может быть наделен несколькими внутренними свойствами, то в Advanced Process эти свойства вынесены как самостоятельные графические модули. Панель состоит из 13 графических модулей (Flowchart Modules): Delay, Dropoff, Hold, Match, Pickup, ReadWrite, Release, Remove, Seize, Search, Signal, Store, Unstore и семи модулей данных (Data Modules): Advanced Set Module, Expression, Failure, File, StateSet, Statistic и Storage.
3. *Advanced Transfer Panel* – панель процессов передачи, перемещения; содержит 17 графических модулей: Enter Module, Leave Module, Pick Station, Route, Station; Access, Convey, Exit, Start, Stop; Activate, Allocate, Free, Halt, Move, Request, Transport и пять модулей данных: Sequence, Conveyor, Segment, Transporter, Distance.

Использование в учебном процессе определяется следующими свойствами данного продукта:

1. В основе программного пакета (ПП) Arena заложены математические аппараты СМО и раскрашенных сетей Петри, а комбинация, берет достоинства обоих математических аппаратов, и возникает синергетический эффект от такого объединения.

2. В ПП Arena первоначально разработчиками предлагается использовать этот пакет для моделирования, анализа сетей и СТК.
3. Компания-разработчик предлагает бесплатную академическую лицензию ПП для учебных заведений.

Литература

1. Шэннон Р. Имитационное моделирование систем – искусство и наука.– М.: Мир, 1978.
2. Бусленко Н.П. Моделирование сложных систем.– М.: Наука, 197
3. Замятина О.М. Моделирование сетей: учебное пособие / О.М. Замятина: Томский политехнический университет. - Томск: Изд-во Томского политехнического университета, 2011
4. Лоу А.М., Кельтон В.Д. Имитационное моделирование. Классика CS. – 3-е изд. – СПб.: Питер; Киев: Издательская группа ВНУ, 2004.
5. Матвеев В.Ф., Ушаков В.Г. Системы массового обслуживания - М.: Изд-во МГУ, 1984.
6. Клейнрок Л. Теория массового обслуживания. Перевод с англ. /Пер. И. И. Грушко; ред. В. И. Нейман – М.: Машиностроение, 1979.
7. Локальные вычислительные сети. Кн. 3. Организация функционирования, эффективность, оптимизация: справочник / под ред. С.В. Назарова. – М.: Финансы и статистика, 1995
8. Питерсон Дж. Теория сетей Петри и моделирование систем: пер. с англ. – М.: Мир, 1984.
9. Хемди А. Таха Введение в исследование операций = Operations Research: An Introduction. – 7-е изд. – М.: [Вильямс](#), 2007.
10. Пащенко Ф.Ф. Введение в моделирование систем. - М.: Финансы и статистика, 2006.
11. Котов В.Е. Сети Петри. – М.: Главная редакция физико-математической литературы, 1984. – 160 с.
12. Юдаев В.В., Зубков Б.В. Применение сетей Петри для моделирования и верификации протоколов обеспечения транспортной безопасности [Электронный ресурс] <https://elibrary.ru/item.asp?id=27656734>
13. <http://simulation.su/static/ru-information.html>