

# Разработка имитационной модели информационной системы на основе виртуальных машин

М. В. Лившиц

Санкт-Петербургский государственный электротехнический университет  
«ЛЭТИ» им. В.И. Ульянова (Ленина)  
marina.livshits24@gmail.com

**Аннотация.** В статье рассматривается разработка имитационной модели информационной системы (ИС) с целью определения параметров проектируемой системы. При моделировании ИС на базе ЭВМ, проектируемая ИС рассматривается как виртуальная машина с предустановленным необходимым программным обеспечением, выполняющим обработку мультимедийных данных. В качестве исходных данных для построения имитационной модели предлагается брать минимальные, либо заданные параметры аппаратной платформы, такие как количество ядер, объем оперативной памяти и другие.

**Ключевые слова:** информационная система, имитационное моделирование, виртуальная машина, терминальное устройство

## I. ВВЕДЕНИЕ

Для создания моделей информационных систем в настоящее время часто применяется имитационное моделирование. Имитационно-моделирующий комплекс представляет собой инструмент для поддержки принятия решений, позволяющий на основе некоторых знаний о элементах системы уменьшить неопределенность знаний о характеристиках и поведении системы в целом [1]. Выбор будущей архитектуры аппаратно-программного комплекса моделирования зависит от сложности исследуемого, создаваемого в модели, объекта или системы, перечня анализируемых показателей модели, в том числе статистических. В настоящее время известно большое количество подходов к моделированию тех или иных объектов

## II. ПОСТАНОВКА ЦЕЛИ МОДЕЛИРОВАНИЯ

Создаваемая имитационная компьютерная модель должна описывать проектируемую информационную систему (ИС) на базе ЭВМ, взаимодействующие с ней другие ИС и каналы связи между ними с целью определения необходимых минимальных параметров ИС. Модель должна удовлетворять следующим требованиям:

- Вычислительные задачи проектируемой ИС и взаимодействующих с ней ИС являются входными параметрами и устанавливаются в модель в виде скомпилированных исполняемых файлов с необходимыми ресурсными файлами.

- Параметры аппаратно-программной платформы проектируемой ИС должны быть изменяемыми.
- Обновление вычислительных задач и их конфигураций должно быть автоматизировано.
- Модель должна выдавать параметры производительности проектируемой ИС, получаемые при выполнении заданных вычислительных задач.
- ИС, взаимодействующие с проектируемой ИС, должны быть представлены набором готовых компонентов.
- Каналы связи между ИС должны быть представлены набором готовых имитаторов.

## III. НАЗНАЧЕНИЕ МОДЕЛИ ИНФОРМАЦИОННОЙ СИСТЕМЫ

На этапе планирования ИС, включающей в себя аппаратные средства и многопользовательское программное обеспечение, существует известная проблема, когда, невозможно дать обоснованную оценку производительности данной ИС в конкретных условиях проведения испытаний этой системы на конечном оборудовании в реальных условиях. Имитационное моделирование позволяет выработать выгодное целесообразное решение, выбор которого подкреплен результатами объективного анализа. Рациональное решение в отличие от основанного на суждении не зависит от опыта, накопленного в прошлом [2].

Цикл разработки аппаратно-программной ИС часто рассматривается как последовательность трех фаз [3]:

- разработка прикладного программного обеспечения;
- анализ задачи и выбора аппаратных средств;
- согласование аппаратурных средств и программного обеспечения (далее – ПО), а также его отладка.

Разрабатываемая имитационная модель предназначена для применения на двух последних фазах. Прикладное программное обеспечение, для удобства работы поставляемое в виде скомпилированного файла или

скриптового сценария, является входным параметром. Аппаратные средства моделируются посредством виртуальных машин. Работая с созданной имитационной моделью можно осуществлять отладку и контроль параметров проектируемой системы. На основании полученной в ходе моделирования информации о проектируемой ИС формируется решение.

#### IV. ОПИСАНИЕ ИМИТАЦИОННОЙ МОДЕЛИ ИС НА ОСНОВЕ ВИРТУАЛЬНЫХ МАШИН

Создание имитационной модели ИС, предназначенной для обработки мультимедийной информации, было проведено с помощью средства создания виртуальных машин VMware ESXi. Для возможности создания сложных сетевых топологий была создана виртуальная машина для GNS3.

В подготовку виртуальной машины входит: создание виртуальной машины в среде виртуализации, выделение ресурсов (числа процессоров, объема дискового пространства, объема оперативной и видео памяти), подключение внешних устройств, подключение необходимого числа сетевых адаптеров и их настройку, установка операционной системы, установка необходимых приложений.

Виртуальная модель строится из подготовленных типовых виртуальных машин. Однако типовые решения не всегда могут обеспечить необходимый набор элементов виртуальной модели. Следовательно, элементы виртуальной модели образуют объединение множеств:

$$A_{vm} \cup B_{vmodel} = \{vm: vm \in A_{vm} \vee vm \in B_{vmodel}\}.$$

Такой подход обеспечит быстрое создание виртуальной модели, и дальнейшее внесение изменений, связанных с операционными системами и элементами виртуальной модели.

Операции включения виртуальных машин VMware ESXi, создание, удаление, включение, изменение выделенных ресурсов и настройки, осуществляются через клиент vSphere.

Созданная коллекция виртуальных машин  $A_{vm}$  вместе с входящими в неё виртуальными машинами GNS3  $vm_{GNS3}$  и эмулятора канала  $vm_{ch}$  были использованы для создания виртуальной модели мультимедийной связи. Компоненты виртуальной модели могут изменяться независимо от элементов созданной коллекции.

Виртуальные машины, составляющие виртуальную модель, являются объединением множества коллекции виртуальных машин и множества дополнительных виртуальных машин созданных при разработке виртуальной модели  $A_{vm} \cup B_{vmodel}$ . Непосредственно виртуальная модель является объединением множества  $B_{vmodel}$  и пересечения двух рассмотренных множеств или операцией отрицания на разность указанных множеств:

$$B_{vmodel} \cap (A_{vm} \cup B_{vmodel}) = \overline{A_{vm} \setminus B_{vmodel}}$$

Каждое абонентское терминальное устройство (регулятор), создает воздействие  $x$  на мультимедийный

терминал (объект). Воздействие  $x$  можно разделить на три группы потоков: видео, аудио и информационный.

Поток видео представляет собой последовательность закодированных кадров, поступающих с видеокamеры или видеофайла. Поток видео характеризуется несколькими параметрами, часто имеющими стандартизованные значения:

- размер, определяющий высоту, ширину и соотношение сторон;
- частота кадров в секунду;
- битовая скорость;
- графический формат.

Поток аудио представляет собой последовательность оцифрованных звуковых фрагментов, поступающих с микрофона, аудио файла или звуковой дорожки видео файла. Поток данного типа характеризуется несколькими параметрами, часто имеющими стандартизованные значения:

- частота дискретизации;
- битовая скорость;
- аудиоформат.

В информационный поток входят передаваемые файлы, мгновенные сообщения, сообщения служебных протоколов SIP, SNMP и других. Источниками информационных потоков часто являются операционные системы, запущенные сервисы или запущенные приложения.

В созданной виртуальной модели, рассматриваемые потоки генерируются виртуальными машинами-источниками и назначаются виртуальным машинам-получателям. Для передачи по сети потоки разбиваются на части, передача происходит с постоянной скоростью. К каждой части добавляются заголовки протоколов, после чего поток в виде кадров передается по сети. Пакеты любого потока можно описать вектором  $M$ , элементами которого будут: скорость передачи  $v$ , передаваемые данные  $D$ , значение приоритета  $p$ , задержка передачи пакета  $\tau_z$ .

$$M = \{v, D, B_{QoS}, \tau_z\}$$

Потоком  $\Phi$  является последовательность пакетов одного из трех перечисленных типов, имеющие одинаковые адреса и порты источника и получателя пакетов. Все кадры одного потока имеют одинаковые значение полей приоритета  $B_{QoS}$ . Поток  $\Phi$ , состоящий из  $n$  кадров, определяется выражением:

$$\Phi = \sum_{i=1}^n M_i$$

Передаваемые данные  $D$  определяются матрицей фиксированного размера  $i \cdot j$ , с байтами данных  $b$ . Элементы матрицы содержат информационные байты

приложений и байты служебных заголовков. Стоит заметить, что значением одного из элементов матрицы так же является поле приоритета  $P$  находящимся в заголовках, но для удобства это значение было продублировано в отдельном элементе вектора  $\Phi$ .  $i \cdot j$  фактически определяет длину передаваемого пакета в байтах.

$$D = \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1i} \\ b_{21} & b_{22} & & b_{2i} \\ \vdots & \vdots & \ddots & \vdots \\ b_{j1} & b_{j2} & \dots & b_{ji} \end{pmatrix}$$

Значение приоритета описывается вектором  $B_{QoS}$ , элементами которого являются значения двух полей заголовков  $b$ , определяющих приоритет пакета. Значения полей задаются в битах.

Виртуальные машины созданной виртуальной модели передают пакеты по технологии Ethernet/IP. В заголовке Ethernet присутствует поле  $b_{Cos}$  длиной 3 бита, называемое CoS. В заголовке IP присутствует поле  $b_{DSCP}$  длиной 6 бита, называемое DSCP.

$$B_{QoS} = (b_{Cos}, b_{DSCP})$$

Технология CoS производит разделение трафика на 8 потоков, каждый из которых передается по сети со своим уровнем качества обслуживания. В качестве параметров выступают: принадлежность к потоку передаваемому со своим уровнем качества обслуживания; способ пересылки через узел; вероятность отброса пакетов; уведомление о перегрузке сети. Обработка значений полей приоритета задается конфигурацией сетевого оборудования.

За единицу потока был принят пакет  $M$ , имеющий число байт  $D$ , которые были сгенерированы мультимедийным приложением на абонентском терминале и переданы по сетевому интерфейсу виртуальной машины со скоростью  $v$  Мб/с и значениями полей приоритета  $B_{QoS}$ . Время  $\tau_z$  определяет время необходимое для достижения пакетом конечной виртуальной машины.

Виртуальная сеть создает воздействия, влияющие на параметры передаваемого потока  $\Phi_{tx}$ . Поток, принимаемый виртуальной машиной, которой он предназначен  $\Phi$ , можно представить как:

$$\Phi_{rx} = f_{vmodel}(\Phi_{tx})$$

где функция  $f_{vmodel}(\Phi)$  определяет воздействие виртуальной модели на передаваемый трафик. При этом, воздействия на трафик передаваемый в одну сторону, могут отличаться от воздействий на трафик передаваемый в обратную сторону. Так как элементы виртуальной модели соединены последовательно, и каждый из элементов воздействует на поток  $\Phi_{tx}$ , то выходной поток является произведением всех воздействий:

$$\Phi_{rx} = f_{vmodel}(\Phi_{tx}) = f_{\Phi_1}(\Phi_{tx}) \cdot f_{\Phi_2}(\Phi_{tx}) \dots f_{\Phi_n}(\Phi_{tx}) = \prod_{i=1}^n f_{\Phi_i}(\Phi_{tx})$$

где функция  $f_{\Phi_i}$  определяет воздействие на поток элемента модели  $i$ . Поток  $\Phi$  состоит из нескольких пакетов, и,

учитывая соотношение выше, принимаемый поток  $\Phi_{rx}$  определяется соотношением:

$$\Phi_{rx} = \prod_{i=1}^n f_{\Phi_i}(\Phi_{tx}) = \prod_{i=1}^n f_{\Phi_i} \left( \sum_{i=1}^n M_i \right) = \prod_{i=1}^n f_{\Phi_i} \left( \sum_{i=1}^n [v \ D \ B_{QoS} \ \tau_z]_i \right)$$

Виртуальная машина отправляет пакеты с некоторым интервалом между пакетами  $\tau_{tx}$ , называемым в соответствии с ГОСТ 29099-91, задержка передачи. Интенсивность отправки пакетов  $\lambda$  является величиной обратной задержке передачи:

$$\lambda = \frac{1}{\tau_{tx}}$$

На объект приходит несколько потоков  $\Phi_{rx}$  от разных источников. Потоки, принятые объектом за время  $t$ , будут определяться соотношением:

$$\sum_{i=1}^n \Phi_{rx_i} = \sum_{i=1}^n \int_0^t \prod_{i=1}^n \lambda_i \cdot f_{\Phi_i} \left( \sum_{i=1}^n M_i \right) \cdot dt$$

Так как объект может выполнять ответную передачу  $\Phi'$ , которую так же можно формализовать с помощью потоков перечисленных типов, справедливо записать систему уравнений:

$$\begin{cases} \sum_{i=1}^n \Phi_{rx_i} = \sum_{i=1}^n \int_0^t \prod_{i=1}^n \lambda_i \cdot f_{\Phi_i} \left( \sum_{i=1}^n M_i \right) \cdot dt \\ \sum_{i=1}^n \Phi'_{rx_k} = \sum_{i=1}^n \int_0^t \prod_{i=1}^n \lambda'_i \cdot f_{\Phi'_i} \left( \sum_{i=1}^n M'_i \right) \cdot dt \end{cases}$$

Функция  $f_{vmodel}$  является воздействием виртуальной модели  $A_{vm} \setminus B_{vmodel}$  на поток  $\Phi$ . Определение результата функции  $f_{vmodel}$  полностью производится экспериментом с виртуальной моделью.

В разработанной виртуальной модели использована многомашинная конфигурация. То есть, создана одна виртуальная инфраструктура с заданным типом взаимодействия элементов. Общая схема виртуальной модели мультимедийной связи  $A_{vm} \setminus B_{vmodel}$  приведена на рис. 1. Разработанная виртуальная модель состоит из терминалов, эмулятора канала и симулятора сети.

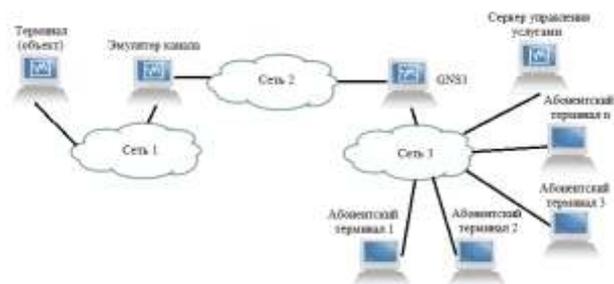


Рис. 1. Схема виртуальной модели мультимедийной связи

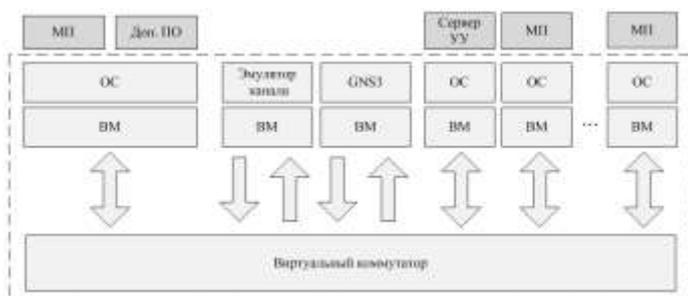


Рис. 2. Архитектура схема виртуальной модели (МП – мультимедийное приложение; Доп.ПО – дополнительное программное обеспечение; ОС – операционная система; ВМ – виртуальная машина; Сервер УУ – сервер управления услугами)

Абонентские терминалы принадлежат множеству виртуальных машин  $A_{vm}$ . Каждый абонентский терминал имеет приложение, предназначенное для взаимодействия с абонентами и выполнения организации мультимедийных сеансов связи между такими же терминалами объединенными вычислительной сетью. Мультимедийное приложение работает с данными  $D$  всех или части перечисленных выше типов потоков  $\Phi$ . Абонентский терминал может выполнять функции источника и приемника потоков  $\Phi$ . Соединение и взаимодействие абонентского терминала с другими элементами виртуальной модели определено конфигурацией средства виртуализации VMware ESXi согласно схеме на рис. 2.

#### V. ИСХОДНЫЕ ДАННЫЕ ДЛЯ ИМИТАЦИОННОЙ МОДЕЛИ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Задача имитационной компьютерной модели состоит в описывании проектируемой ИС на базе ЭВМ, взаимодействующих с ней других ИС и каналов связи между ними, с целью определения параметров проектируемой системы. При моделировании ИС на базе ЭВМ, проектируемая ИС рассматривается как виртуальная машина с предустановленным необходимым программным обеспечением, выполняющим обработку мультимедийных данных. Параметры аппаратной платформы задаются в виде ограничений ресурсов сервера, где развернута модель, выделяемых виртуальной машине. ИС, взаимодействующие с проектируемой ИС, представляют собой похожие виртуальные машины, но без явного ограничения выделенных ресурсов сервера. Каналы связи представляются заранее подготовленными виртуальными машинами со специально подобранным программным обеспечением, позволяющим вносить задержки и изменения в транзитный трафик. В качестве исходных данных для построения имитационной модели проектируемой ИС следует брать минимальные, либо

заданные параметры аппаратной платформы, такие как количество ядер, объем оперативной памяти и другие. Соединения виртуальных машин между собой может состоять из трех сетей, либо гибко изменено при необходимости построения более сложной модели. Входными данными модели являются:

- Набор вычислительных задач в виде скомпилированных исполняемых файлов, с необходимыми ресурсными файлами и динамически подгружаемыми библиотеками.
- Набор ограничений аппаратных ресурсов, выделяемых проектируемой ИС.
- Набор связей между проектируемой ИС и взаимодействующих с ней ИС.

#### VI. ЗАКЛЮЧЕНИЕ

На сегодняшний день существуют различные подходы к разработке программного обеспечения – модели жизненного цикла разработки программного обеспечения. Применение разработанной имитационной компьютерной модели при проектировании по методике ЦБР предполагается на этапе – проектирование ИС. При моделировании могут выполняться следующие действия:

- Создается архитектура ИС (в т. ч. параметры аппаратно-программной части);
- Проводится квалификационное тестирование ПО в составе ИС (в т. ч. функциональное тестирование, тестирование на проникновение, фазинг тестирование).

#### БЛАГОДАРНОСТИ

Выражаю особую благодарность заведующему кафедрой Автоматики и процессов управления СПбГЭТУ «ЛЭТИ» им. В.И. Ульянова (Ленина) д.т.н. Шестопалову Михаилу Юрьевичу за ценные советы в направлении исследования, советы и рекомендации при подготовке данной статьи.

#### СПИСОК ЛИТЕРАТУРЫ

- [1] Кондрашин М.А., Арсенов О.Ю., Козлов И.В. Применение технологии виртуализации и облачных вычислений при построении сложных распределенных моделирующих систем // Труды МАИ, 2016, вып. 89. С. 34.
- [2] Хемди А. Таха. Гл.18 Имитационное моделирование // в книге «Введение в исследование операций = Operations Research: An Introduction». 7-е изд. М.: Вильямс, 2007. С. 667-705.
- [3] Зиглер К. Методы проектирования программных систем. М.: Мир, 1985. С. 15-23.