

Имитационная модель поддержки планирования учебного процесса в высшем учебном заведении

А.А. Горбунов

*преподаватель кафедры управления информационными системами и цифровой инфраструктурой
Национальный исследовательский университет «Высшая школа экономики»
Адрес: 101000, г. Москва, ул. Мясницкая, д. 20
E-mail: agorbunov@hse.ru*

Е.А. Исаев

*кандидат технических наук
профессор, заведующий кафедрой управления информационными системами и цифровой инфраструктурой
Национальный исследовательский университет «Высшая школа экономики»
Адрес: 101000, г. Москва, ул. Мясницкая, д. 20
E-mail: eisaev@hse.ru*

А.Ф. Моргунов

*кандидат технических наук
доцент кафедры управления информационными системами и цифровой инфраструктурой
Национальный исследовательский университет «Высшая школа экономики»
Адрес: 101000, г. Москва, ул. Мясницкая, д. 20
E-mail: amorgunov@hse.ru*

Аннотация

В вузах и втузах с профильными ИТ-специальностями в одном семестре могут обучаться работе с программными продуктами сразу несколько потоков, курсов, специальностей. Поэтому перед ИТ-службами учебных заведений возникает задача создания инфраструктуры учебных приложений, которая сможет обеспечить поддержку учебного процесса. Следует учитывать, что число специальностей, на которых изучаются информационные технологии, с каждым годом растет (например, в НИУ ВШЭ преподаются дисциплины-майоры, куда может записаться студент с любой специальности). Также в последнее время популярностью стали пользоваться дистанционные курсы. Если не планировать нагрузку, то с учетом будущих трендов, мощности даже самой высокотехнологичной инфраструктуры будут недостаточны. Расчет соответствующей нагрузки на инфраструктуру необходимо производить в процессе планирования учебных дисциплин, что позволит выполнять резервирование соответствующих мощностей и тем самым организовать эффективный учебный процесс.

Разработчики программного обеспечения используют различные бенчмаркинг-инструменты, которые сложны и не предоставляют необходимой информации для участников планирования учебного процесса.

В статье рассматривается построение имитационной модели поддержки планирования учебного процесса. Моделирование осуществляется с использованием возможностей инструмента AnyLogic 7. Целью данной работы является разработка имитационной модели, предназначенной для оценки нагрузки на информационные системы, используемые в ходе учебного процесса. Помимо описания модели, в статье приведены результаты расчетов с ее использованием для различных вариантов размещения информационной системы (в частном облаке или на сервере в университете). Результаты моделирования подтверждены данными, полученными в ходе проведения практических занятий в вузе. Данная модель дает возможность планировать учебный процесс с целью добиться равномерности нагрузки на сервисы. В случае необходимости модель позволяет принять решение о месте размещения учебной информационной системы: на серверах университета или в частном облаке.

Ключевые слова: учебный процесс, информационная система, ERP-система, имитационная модель, частное облако, ИТ-инфраструктура.

Цитирование: Gorbunov A.A., Isaev E.A., Morgunov A.F. A simulation model for educational process planning in an institution of higher education // Business Informatics. 2017. No. 2 (40). P. 57–67. DOI: 10.17323/1998-0663.2017.2.57.67.

Введение

Имитационное моделирование [1–3] находит широкое применение в различных областях науки и производства. В научных изданиях приводятся результаты моделирования сложных экономических [4], социально-экономических [5] и технических систем [6].

Модели технических систем, в частности, позволяют оценивать нагрузку на серверное оборудование. Подобные задачи возникают, например, при внедрении и эксплуатации корпоративных информационных систем. В высших учебных заведениях для обучения студентов работе с корпоративными информационными системами могут использоваться ресурсы собственной или облачной серверной инфраструктуры. Для организации эффективных практических занятий сотрудники, участвующие в планировании учебного процесса, должны обладать информацией о нагрузке на ИТ-инфраструктуру. Потребность в такой оценке особенно велика на этапе разработки учебных программ и учебных планов. Равномерное распределе-

ние нагрузки на оборудование в течение учебного года позволит избежать сбоев при работе с ним и в результате повлиять на эффективность использования учебного времени.

1. Постановка задачи

Процесс планирования учебных дисциплин, для преподавания которых требуется наличие оборудованных компьютерных классов и доступ к специализированному программному обеспечению классов ERP, CPM и CRM, приведен на *рисунке 1*.

Для образовательных учреждений ведущие российские и зарубежные вендоры программного обеспечения предоставляют мощности для использования программных продуктов из собственного облака, что освобождает от необходимости установки, приобретения лицензий, создания ИТ-инфраструктуры, обеспечения доступа и т.д. Предлагаемые мощности ограничены и, как показывает практика (*таблица 1*), не всегда обеспечивают стабильную, производительную работу при обучении большого количества студентов в одном периоде.

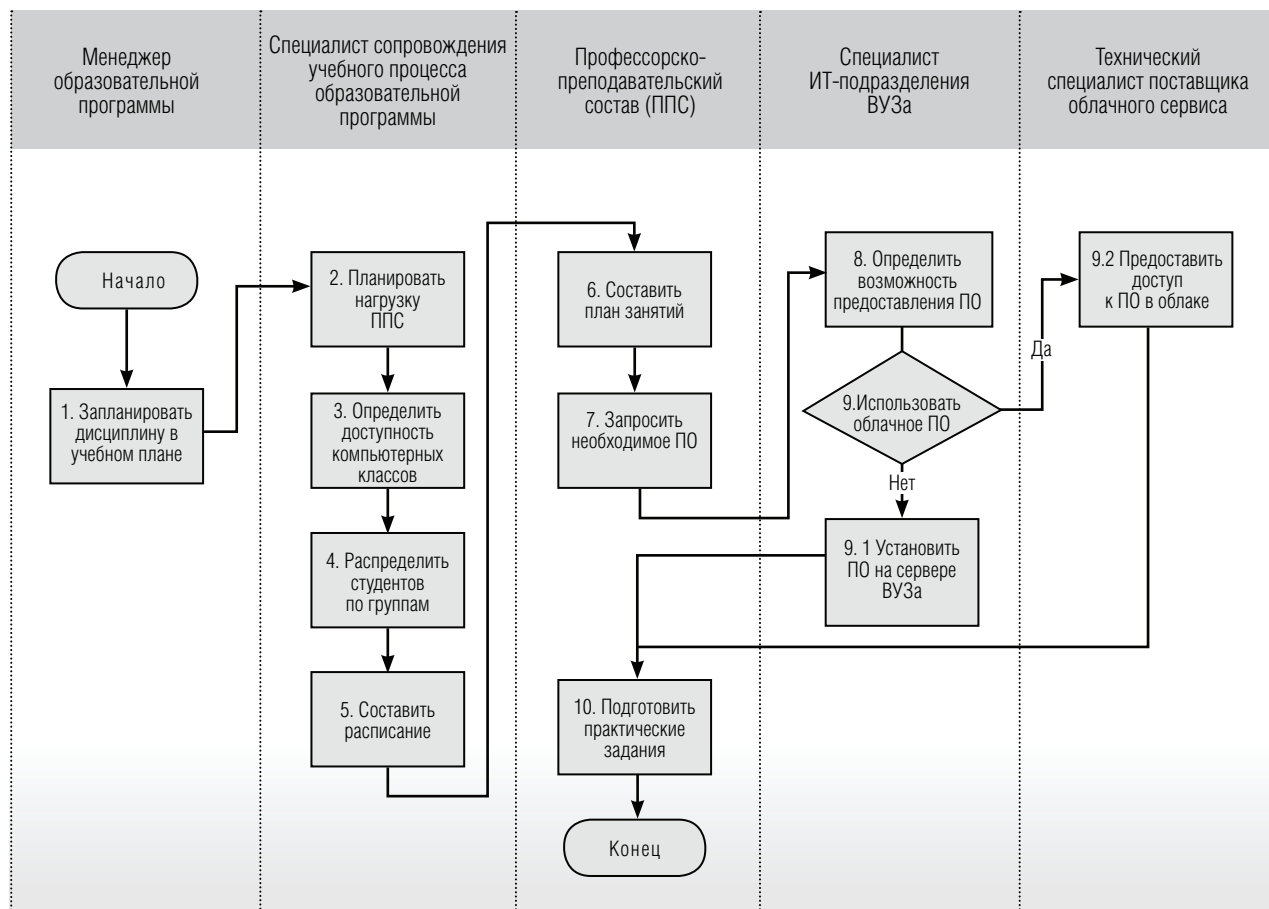


Рис. 1. Процесс планирования учебных дисциплин

Таблица 1.

Наблюдаемые показатели работы облачных сервисов

Провайдер сервиса (вендор)	Программный Продукт	Количество обучаемых в семестре	Время недоступности сервиса (дн.)	Время медленной работы сервиса (дн.)
SAP	SAP ERP	400	5	10
1C	1C: Управление предприятием ERP 2		15	36

В российских вузах и втузах с профильными ИТ-специальностями в одном семестре могут обучаться работе с программными продуктами сразу несколько потоков, курсов, специальностей. Поэтому перед ИТ-службами учебных заведений возникает задача создания инфраструктуры учебных приложений, которая сможет обеспечить поддержку учебного процесса. Следует учитывать, что число специальностей, на которых изучаются информационные технологии, с каждым годом растет (например, в НИУ ВШЭ преподаются дисциплины-майноры, куда может записаться студент с любой специальности). Также в последнее время популярностью стали пользоваться дистанционные курсы. Если не планировать нагрузку, то с учетом будущих трендов, мощности даже самой высокотехнологичной инфраструктуры будут недостаточны. Расчет соответствующей нагрузки на инфраструктуру необходимо производить в процессе планировании учебных дисциплин, что позволит выполнять резервирование соответствующих мощностей и тем самым организовать эффективный учебный процесс.

Разработчики программного обеспечения используют различные бенчмаркинг-инструменты, которые сложны и не предоставляют необходимую информацию для участников планирования учебного процесса.

Целью данной работы является разработка имитационной модели, предназначенной для оценки нагрузки на информационные системы, используемые в ходе учебного процесса.

Для достижения поставленной цели определены следующие задачи:

- ♦ определение параметров моделирования, имитация которых предоставит необходимую информацию для участников процесса планирования учебного процесса;
- ♦ разработка математической модели и ее реализация в AnyLogic 7;
- ♦ проведение экспериментов и сравнение результатов с реальными наблюдениями.

2. Основные предположения модели

Имитационная модель опирается на ряд предположений, которые подтверждаются на практике в течение периода использования ERP-системы.

Предположение 1. Практические занятия посещают студенты, которые сдают выполненные задания в разное время. На основании длительности выполнения заданий студентов можно разделить на группы следующим образом:

- ♦ выполняющие задание вовремя:
 - ♦ выполняющие задания за время занятий (раньше срока);
 - ♦ выполняющие задания в срок (если студент не выполнил задание за время занятий, задание выполняется во внеаудиторное время в срок до следующего занятия);
- ♦ выполняющие задание с опозданием:
 - ♦ выполняющие задания на одну неделю позже срока;
 - ♦ выполняющие задания на две недели позже срока;
 - ♦ выполняющие задания перед зачетным занятием в конце семестра.

Предположение 2. Студенты готовы к выполнению задания и на время его выполнения влияют только производительность и доступность сервиса. Выделены следующие параметры сервиса:

- ♦ медленная работа приложения (при этом большинство пользователей системы выполняют операции с ожиданием):
 - ♦ во время занятий;
 - ♦ во внеаудиторное время;
- ♦ недоступность приложения:
 - ♦ во время занятий;
 - ♦ во внеаудиторное время.

Предположение 3. Время выполнения практического задания при различных состояниях сервиса различается. Например, расчетное время выполне-

ния задания неподготовленным пользователем составляет 2 часа, если сервис доступен, но работает медленно, и 1,5 часа, если сервис работает быстро.

Предположение 4. Производительность и доступность сервиса обеспечивается ресурсами сервера.

Практические занятия по дисциплинам «Корпоративные информационные системы» и «Информационные системы управления» преподавались в НИУ ВШЭ период с сентября по декабрь 2016 г. В данный период практические занятия с использованием облачного сервиса выполняли 400 студентов бакалавриата старших курсов.

Студенты были разделены следующим образом:

✧ 240 студентов 4 курса направления «Менеджмент» выполняли практические занятия группами максимальной численностью 15 человек (16 подгрупп);

✧ 160 студентов 3 курса различных направлений в рамках общеуниверситетской дисциплины (майнора) выполняли работы в группах максимальной численностью 32 человека (5 подгрупп).

Следует также отметить, что студенты ранее не изучали корпоративные информационные системы и не имели практических навыков, позволяющих выполнить практические задания, опираясь на предыдущий опыт работы. Однако они получили необходимую информацию о порядке работы в ин-

формационной системе от преподавателя в начале занятия. Данное уточнение позволяет исключить связь наличия навыков работы и выполненных в срок практических заданий.

Предположение 5. Если сервис доступен и приложение работает быстро, то студент способен выполнить задание за отведенное на занятии время, а если приложение работает медленно — то во внеаудиторное время, в срок.

Предположение 6. При организации работы ERP-системы в собственной инфраструктуре ресурсы единственного сервера будут использоваться для поддержки производительности и доступности двух приложений.

Для целей моделирования выделяются несколько практических сценариев, которые будут выполняться группами студентов на ERP-системах разных вендоров (или одного вендора), при этом системы используют ресурсы единственного сервера. Таким образом, модель можно представить как систему массового обслуживания [7, 8], в которой присутствуют три источника (студенты, выполняющие задания), очереди запросов от двух информационных систем и единственный сервер, обрабатывающий запросы.

В качестве инструментария моделирования использована функциональность информационной системы AnyLogic 7 [4].

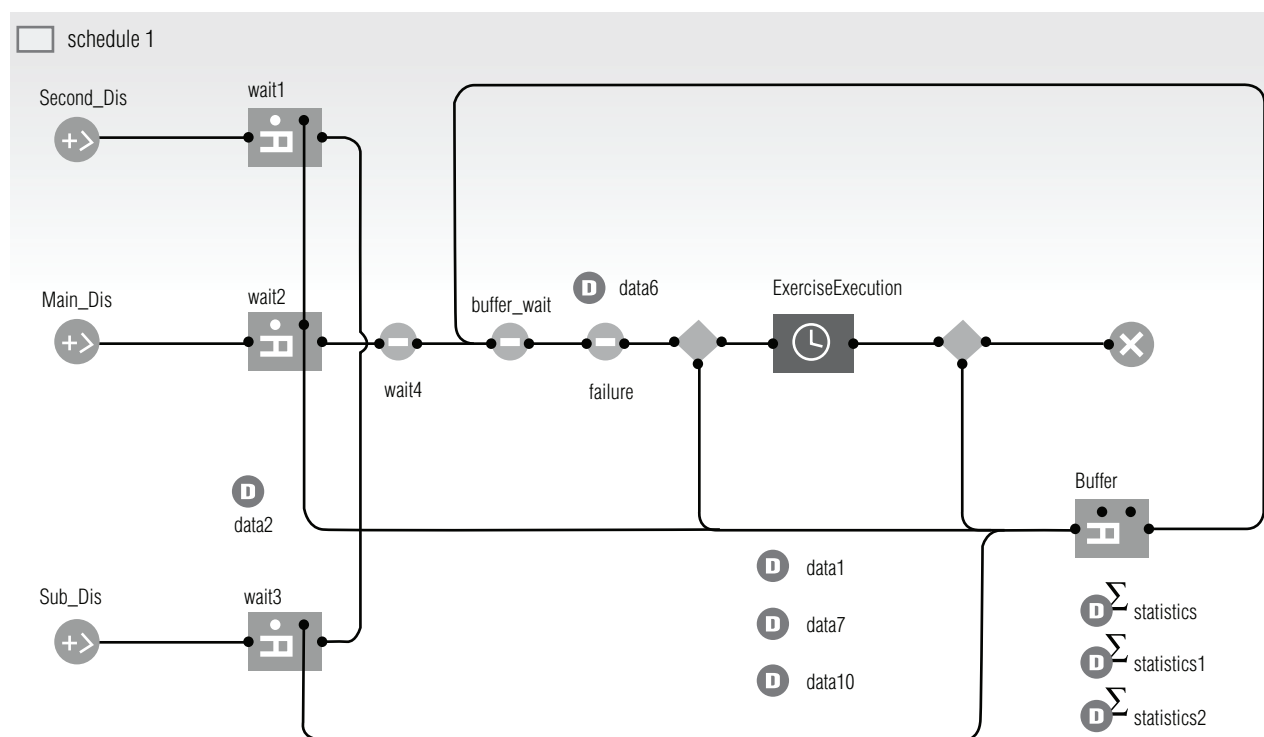


Рис. 2. Логическая схема имитационной модели

3. Возможности и основные расчетные параметры модели

3.1. Расчет значений различных временных параметров на основе индивидуальных характеристик студентов

Студенты приходят на занятия по расписанию, количество пришедших на занятие отображается в блоках *Main_Dis*, *Sub_Dis*, *Second_Dis*. Пришедшие на занятие студенты попадают в очередь (блоки *wait1*, *wait2* или *wait3*) (рисунки 2), которая имитирует ожидание студентов во время объяснения преподавателем порядка выполнения задания, теоретических основ и т.д.

Продолжительность времени объяснения материала преподавателем τ_0 является случайной величиной, распределенной в интервале 10–30 минут, и зависит от ранга $r > 0$ (сложности задачи):

$$\tau_0 = U(10, 30) \cdot \frac{r}{2}, \quad (1)$$

где $U(10, 30)$ – функция равномерного распределения с параметрами 10, 30.

После завершения объяснения преподавателем студенты приступают к практическим занятиям с некоторой временной задержкой продолжительностью τ_i'' , $i \in \{1, \dots, n\}$, которая зависит от скорости выполнения задач, индивидуальной для каждого студента (формула 3).

Скорость выполнения задания (k_i) показывает, какой процент времени занятия i -й студент занят его выполнением. Обозначим переменной d время занятия, тогда справедливы следующие формулы:

♦ для вычисления времени, в течение которого i -й студент занят выполнением задания на занятии:

$$\tau_i' = k_i d - \tau_0 \quad (2)$$

♦ для вычисления временной задержки перед началом выполнения задания i -м студентом:

$$\tau_i'' = (1 - k_i) \cdot d. \quad (3)$$

Время занятия может быть вычислено как сумма временных параметров:

$$d = \tau_0 + \tau_i' + \tau_i''. \quad (4)$$

При поступлении студента в один из блоков (*Main_Dis*, *Sub_Dis*, *Second_Dis*) ему присваивается значение скорости выполнения, которое вычисляется следующим образом: выбирается случайное равномерно распределенное число из интервала $[0, 1)$, которое умножается на 100.

Чем меньше значение k_i , тем меньшее время студент занят выполнением практического задания τ_i' и тем больше время τ_i'' задержки перед началом выполнения задания.

Пример 1. Пусть студенты решают задание, сложность (ранг) которого $r = 2$. По формуле 1 вычислим время объяснения задания преподавателем $\tau_0 = 30$ мин. Произведем расчет параметров τ_i' , τ_i'' модели для двух студентов ($i = 1, 2$). Предположим, что скорости выполнения задания первым и вторым студентами $k_1 = 0,85$, $k_2 = 0,4$. Тогда за время занятия d (2 часа 40 минут) первый студент может быть занят выполнением задания в течение времени

$$\tau_1' = 0,85 \cdot (2 \text{ ч. } 40 \text{ мин.}) - 30 \text{ мин.} = 1 \text{ ч. } 46 \text{ мин.},$$

а второй студент – в течение времени

$$\tau_2' = 0,4 \cdot (2 \text{ ч. } 40 \text{ мин.}) - 30 \text{ мин.} = 34 \text{ мин. (формула 2).}$$

При этом временная задержка перед началом выполнения задания для первого студента составляет $\tau_1'' = 24$ минуты, а для второго –

$$\tau_2'' = 1 \text{ час } 36 \text{ минут (формула 3).}$$

Для того чтобы в результате имитационных экспериментов установить количество студентов, способных выполнить практическое задание ранга r на занятии, а также тех, кому требуется внеаудиторное время для самостоятельной работы введем следующие обозначения:

t_{j1} , t_{j2} – время начала и окончания j -го занятия, $j \in \{1, \dots, m\}$, где m – число занятий;

t_{ij} – момент времени завершения выполнения j -го задания i -м студентом ($t_{ij} \geq 0$);

z – расчетное (эталонное) время выполнения задания (константа, определяется преподавателем);

$g_i(t_{ij}, \tau_i')$ – вероятность того, что i -й студент потенциально способен выполнить j -е задание:

$$g_i(t_{ij}, \tau_i') = \begin{cases} 1, & \text{если } z \leq \tau_i' \text{ и } t_{ij} \leq t_{j2} \\ 0, & \text{если } z > \tau_i' \text{ и } (z \leq \tau_i' \text{ и } t_{ij} > t_{j2}). \end{cases} \quad (5)$$

Пример 2. Опираясь на расчеты, приведенные в примере 1, определим кто из двух студентов способен выполнить первое ($j = 1$) задание, ранг которого $r = 2$. Пусть занятие начинается в момент $t_{j1} = 12$ часов 10 минут, заканчивается в момент $t_{j2} = 15$ часов 00 минут, при этом расчетное время выполнения задания, определенное преподавателем, составляет $z = 1$ час 40 минут. Первый студент завершает задание в момент $t_{11} = 14$ часов 50 минут. Применяя формулу 5, получаем: $g_1(t_{11}, \tau_1') = 1$ (1 ч. 40 мин. $\leq$ 1 ч. 46 мин.; 14 ч. 50 мин. $\leq$ 15 ч. 00 мин.). Для второго студента первый параметр не имеет значения, поскольку $z = 1$ ч. 40 мин. $>$ $\tau_2' = 34$ мин., $g_2(t_{21}, \tau_2') = 0$.

Таким образом, из двух студентов только первый сможет выполнить задание на занятии, а второму потребуется для этого внеаудиторное время.

Студенты, которым требуется время для внеаудиторной работы ($g_i(t_i, \tau_i') = 0$, блок *Buffer*) могут заходить в систему для того, чтобы доделать упражнения, не выполненные за основное время занятия d . Вход в систему (блок *wait_buffer*) во внеаудиторное время происходит через определенные интервалы времени, которые распределены по экспоненциальному закону. Число входов в систему (число попаданий в буфер) i -го студента для завершения практического задания определяется формулой:

$$c = \left\lceil \frac{z}{\tau_i'} - 1 \right\rceil \cdot 100. \quad (6)$$

Студент i завершает практическое задание и удаляется из буфера после того, как зайдет в систему с раз.

Пример 3. В примере 2 показано, что одному из студентов потребуется внеаудиторное время для завершения задания 1. Для того, чтобы определить, сколько раз студент будет приступать к заданию во внеаудиторное время, рассчитаем параметр c . В качестве значения параметра τ_2' используется значение, рассчитанное в примере 1. Выполним расчет:

$$c = \left\lceil \frac{1 \text{ час } 40 \text{ мин.}}{34 \text{ мин.}} - 1 \right\rceil \cdot 100 = 100.$$

3.2. Расчет влияния показателей надежности вычислительной инфраструктуры облачного сервиса учебной ERP-системы на различные временные параметры

Реальная вычислительная инфраструктура, поддерживающая облачный сервис учебной ERP-системы, является непрозрачной, что представляет большую сложность при разработке имитационной модели. Среди научных работ, посвященных тематике архитектур вычислительных систем, выделяются публикации В.Г. Хорошевского [6], в которых приведены методики расчета показателей надежности, а также осуществимости решения задач средствами вычислительных систем. Предположим, что вычислительная инфраструктура облачного сервиса соответствует архитектуре вычислительной системы со структурной избыточностью [6].

Данная вычислительная система выглядит для пользователя как виртуальная система, имеющая число элементарных машин [6], мощность которых позволяет осуществить реализацию задач соответствующих рангов (задач соответствующей сложности). Воспользуемся методиками [6], предположив,

что ранг задачи — это число элементарных машин, которые балансируют нагрузку при выполнении задачи максимальным количеством пользователей.

Суть данного предположения состоит в следующем: если существует вероятность, что практические задания (задачи) будут выполняться одновременно множеством пользователей (студентов), то для решения задачи требуется число элементарных машин, равное рангу задачи. Как следует из наблюдений за сдачей практических заданий в рамках дисциплин, предусматривающих выполнение работ с использованием программного обеспечения, теоретически всегда следует допускать ситуацию значительной нагрузки (в течение нескольких часов, дней и т.д.) на вычислительную инфраструктуру.

Пусть $R(t)$, $U(t)$ — функции соответственно надежности и восстановимости системы со структурной избыточностью. Тогда математическое ожидание времени безотказной работы и среднее время восстановления вычислительной системы (ВС) можно вычислить по формулам [6]:

$$\theta = \int_0^{\infty} R(t) dt \quad (7)$$

$$T = \int_0^{\infty} t dU(t). \quad (8)$$

Если среднее время безотказной работы уменьшается, то τ_i'' (средняя задержка перед началом выполнения задания) увеличивается, а τ_i' (время занятости студента на занятии) уменьшается. Также для приведенной зависимости является верным, что среднее число студентов, которые будут выполнять задания за время занятия, будет уменьшаться, в то время как среднее количество студентов, которым требуется внеаудиторное время, будет увеличиваться. Таким образом, при уменьшении среднего времени безотказной работы буфер студентов, не выполнивших задания за основное время занятия, будет в среднем выше, чем при работе в отказоустойчивых режимах.

3.3. Определение влияния интенсивности потока задач на различные временные параметры

Предположим, что вычислительная система облачного сервиса работает в режиме обслуживания потока задач. Тогда воспользуемся упрощением общего случая, рассматриваемого в работе [6]. В общем случае, присутствуют задачи различных рангов $1 \leq r \leq N$, где N — количество элементарных машин

(ЭМ) в ВС. Для каждого ранга выделяются одна или несколько подсистем, балансирующих нагрузку на ЭМ, количество которых равно соответствующему рангу. Для определения осуществимости решения задач требуется рассчитать математическое ожидание количества задач, находящихся в системе, и количество ЭМ, занятых решением задач [6, 9].

Рассмотрим два случая:

♦ случай 1: Поток задач имеет слабую интенсивность. Случай соответствует ситуации, при которой группы студентов немногочисленны, поток параллельных занятий не присутствует. Данный случай прост, поэтому не требует тщательного рассмотрения и проведения имитационных экспериментов;

♦ случай 2. Поток задач, поступающих на ВС, очень интенсивный (например, если группы студентов имеют большую численность, а занятия ведутся в несколько параллельных потоков). В этом случае существует вероятность максимальной нагрузки на вычислительную систему.

Случай 2 является предметом проведения экспериментов в имитационной модели.

4. Проведение экспериментов

Для решения практической задачи по оценке нагрузки на оборудование и программное обеспечение при выполнении практических заданий требуется рассчитать и сравнить показатели организации проведения занятий по дисциплине при различных условиях.

Разработан ряд экспериментов, проведение которых позволяет получить выходные показатели при следующих вариантах организации дисциплины:

♦ эксперимент 1: облачный сервис для единственных дисциплин с одним потоком студентов;

♦ эксперимент 2: собственная инфраструктура для единственной дисциплины и двух потоков студентов;

♦ эксперимент 3: собственная инфраструктура для двух дисциплин (расширенный эксперимент 2).

4.1. Эксперимент 1.

**Выполнение практических заданий
производится с помощью ERP-системы
в облачном сервисе**

Условия:

- первая неделя: в сервисе работают 2 ЭМ, что позволяет балансировать нагрузку при одновременной работе 30 пользователей, на занятии студенты выполняют задание с рангом 2;

- вторая неделя: в сервисе отмечается медленная работа, число ЭМ снижается до 1, на занятии студенты выполняют задания с рангом 1;

- третья и четвертая недели: первую половину всего времени сервис недоступен, вторую половину — работает единственная ЭМ, студенты выполняют задание с рангом 1.

4.2. Эксперимент 2.

**Выполнение практических заданий
производится с помощью ERP-системы,
поддерживаемой серверной
инфраструктурой вуза**

Условия:

- серверная инфраструктура поддерживает две ERP-системы разных производителей, которые используются для выполнения практических заданий двумя параллельными группами студентов (дисциплина №1) в одно и то же время, число ЭМ в инфраструктуре равно 2, т.е. 30 пользователей;

- максимальный ранг задач, которые могут выполняться студентами, — 1, т.е. максимальное число студентов в группе — не более 15.

4.3. Эксперимент 3.

Расширенный вариант эксперимента 2

Условия:

- условия эксперимента 2;

- в вузе преподается курс 2, в котором студенты также выполняют практические задания с использованием ERP-системы, поддерживаемой серверной инфраструктурой вуза;

- дни занятий по курсу 2 не совпадают с днями занятий по курсу 1;

- студенты курса 2, не выполнившие задания за время основного занятия, могут заходить в систему, чтобы доделать задания в внеаудиторное время;

- время входа студентов в систему во внеаудиторное время является случайным.

В процессе проведения каждого эксперимента выполнялось 10 прогонов, усредненные показатели представлены в *таблице 2*.

4.4. Анализ результатов экспериментов

Показатели «средняя задержка перед выполнением», «остаток в буфере» и «число невыполненных заданий» в экспериментах 2 и 3 (*таблица 2*) значительно ниже аналогичных для эксперимента 1. В

Таблица 2.

Результаты экспериментов

Показатель	Номер эксперимента		
	1	2	3
1. Средние показатели за период моделирования с 22.09.16 12:10 по 15.12.16 15:00			
Скорость выполнения заданий (общая)	0,605	0,597	0,592
Расчетное время занятости студента выполнением задания на занятии (ч.)	1,36	0,96	0,95
Задержка перед выполнением задания (ч.)	3,21	0,48	0,52
Время объяснения задания преподавателем (ч.)	0,5	0,5	0,5
Число задач, ожидающих решения во внеаудиторное время (буфер)	32	12	30
Продолжительность времени между повторными входами в систему (мин.)	49	19	18
Число заходов в систему во внеаудиторное время	3	2	2
Время безотказной работы системы (средняя наработка до отказа) (дн.)	12	30	30
Время, за которое задание выполнено на занятии (ч.)	1,4	1,4	1,4
Продолжительность выполнения задания во внеаудиторное время в срок до дедлайна (дн.)	0,98	1,37	1,47
Продолжительность времени от окончания занятия до дедлайна (дн.)	2	2	2
Продолжительность выполнения задания во внеаудиторное время с опозданием в 1 неделю (дн.)	4,25	3,02	3,79
Продолжительность выполнения задания во внеаудиторное время с опозданием в 2 недели (дн.)	11,5	0	0
2. Накопленные показатели за период моделирования с 22.09.16 12:10 по 15.12.16 15:00			
Число заданий, выполненных на занятии	133	321 (160 заданий на поток)	467 (155 заданий на поток)
Число заданий, выполненных досрочно	48	291 (145 заданий на поток)	210 (70 заданий на поток)
Число заданий, выполненных с опозданием в 1 неделю	57	172 (86 заданий на поток)	480 (160 заданий на поток)
Число заданий, выполненных с опозданием в 2 недели	63	0	0
Число невыполненных заданий	23	35 (17 заданий на поток)	68 (22 заданий на поток)
3. Остаточные показатели на конец периода моделирования 15.12.16 15:00			
Задания, ожидающие решения во внеаудиторное время (остаток в буфере)	141	111 (55 заданий на поток)	170 (56 заданий на поток)
Выполненные задания	301	784 (392 заданий на поток)	1157 (385 заданий на поток)

дополнение следует также учесть, что эксперименты 2 и 3 проведены для двух и трех одновременных потоков, а число выполненных заданий на конец периода в расчете на один поток в среднем больше почти на 1/3, чем значение аналогичного показателя для эксперимента 1 (таблица 2).

Таким образом, вариант дисциплины, проводимой с использованием облачного сервиса ERP-системы, уступает вариантам, предусматривающих использование собственной инфраструктуры для поддержки учебного приложения по большинству показателей.

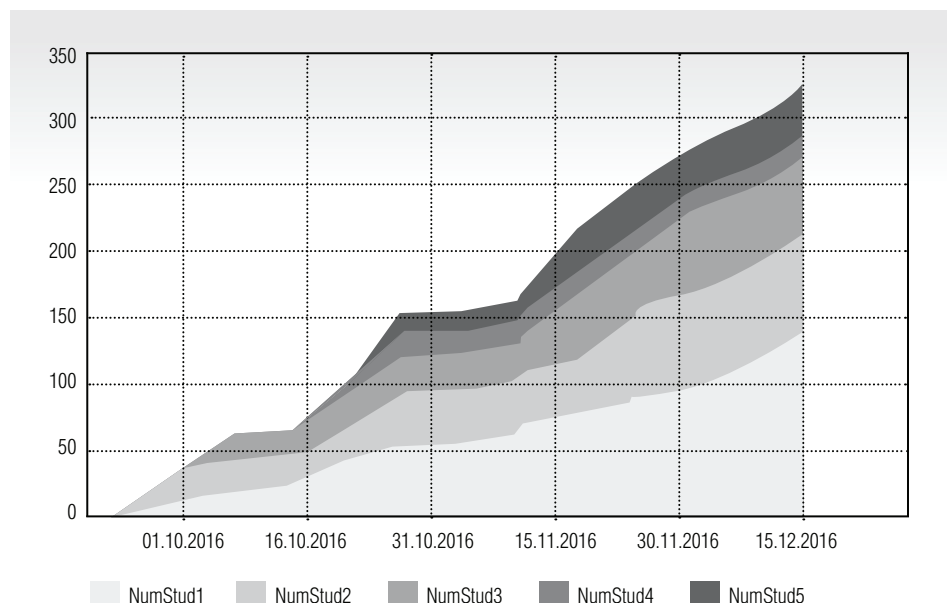


Рис. 3. Результат имитационного эксперимента 1

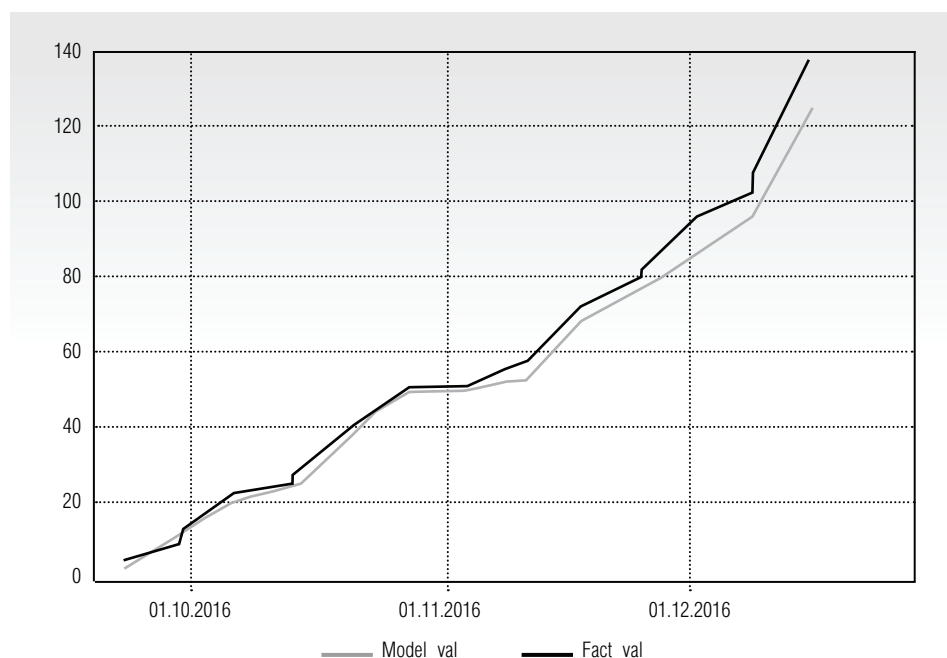


Рис. 4. Сравнение результатов эксперимента и фактических наблюдений

5. Сравнение результатов моделирования с фактическими наблюдениями

В результате имитационного эксперимента 1 были получены значения накопленных показателей *таблицы 2*. На *рисунке 3* представлены результаты одного из прогонов, где NumStud1 – число заданий, выполненных на занятии; NumStud2 – число заданий, выполненных до дедлайна; NumStud3 –

число заданий, выполненных с опозданием в 1 неделю; NumStud4 – число заданий, выполненных с опозданием в 2 недели; NumStud5 – число невыполненных заданий.

Условия эксперимента 1 соответствуют условиям, в которых проводились практические занятия по дисциплине «Корпоративные информационные системы». Для определения достоверности данных, полученных в результате моделирова-

ния, были проведены эксперименты по сверке с использованием фактических наблюдений. Выполнено 10 прогонов в рамках эксперимента 1, полученные средние значения по совокупности прогонов для одного из смоделированных показателей (NumStud1), а также фактические наблюдения отражены на *рисунке 4*. Сравнение показывает, что расхождение между результатами численных экспериментов и фактическими данными не превышает 6,5%.

Заключение

В рамках данной работы продемонстрирован подход, который основывается на использовании имитационной модели для оценки нагрузки на программное обеспечение, используемое при проведении практических занятий в вузе. Задача оценки нагрузки на программное обеспечение актуальна для преподавателей, менеджеров учебных программ, специалистов по учебно-методической работе, занимающихся планированием различных аспектов учебных дисциплин, связанных с исполь-

зованием «тяжелого» программного обеспечения. Существующие подходы предназначены для целей оценки нагрузки при начале внедрения или оптимизации уже находящихся в эксплуатации информационных систем, являются сложными и не предоставляют необходимой информации для вышеприведенных групп пользователей.

Модель разработана с учетом анализа результатов использования в учебном процессе облачного сервиса ERP-приложения и опирается на различные предположения, связанные с влиянием параметров инфраструктуры (сервера) на длительность выполнения практических сценариев в ERP-приложении. Она может быть использована при планировании – подготовке плана занятий (в т.ч. при расчете объема практических заданий); определении семестра, в котором возможно запланировать преподавание дисциплины (в т.ч. ее продолжительности), расчете загрузки преподавателя, расчете максимального количества студентов в группе, а также при определении типа инфраструктуры размещения программного обеспечения (собственный сервер, облачный сервис и т.д.). ■

Литература

1. Акопов А.С. Имитационное моделирование: Учебник и практикум для академического бакалавриата. М.: Юрайт, 2014.
2. Лычкина Н.Н. Имитационное моделирование экономических процессов: Учебное пособие. М.: ИНФРА-М, 2014.
3. Хорошевский В.Г. Архитектура вычислительных систем: Учебное пособие. М.: МГТУ им. Н.Э. Баумана, 2008.
4. Алиев Т.И. Основы моделирования дискретных систем. СПб: СПбГУ ИТМО, 2009.
5. Павский В.А., Павский К.В. Оценка показателей осуществимости решения задач на распределенных вычислительных системах // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2008. № 4 (5). С. 61–68.
6. Саати Т.Л. Элементы теории массового обслуживания и ее приложения. М.: Либроком, 2010.
7. Борщев А.В. Практическое агентное моделирование и его место в арсенале аналитика. [Электронный ресурс]: <http://www.gpss.ru/immod05/p/borshev/print.html> (дата обращения 18.03.2017).
8. Павловский Ю.Н., Белотелов Н.В., Бродский Ю.И. Имитационное моделирование. Учебное пособие для студентов высших учебных заведений. М.: Академия, 2008.
9. Сидоренко В.Н., Красносельский А.В. Имитационное моделирование в науке и бизнесе: подходы, инструменты, применение // Бизнес-информатика. 2009. № 2 (08). С. 52–57.