

**Хасан Али Аль-Абабнех**  
*Кандидат технических наук*  
**Hassan Ali al-Ababneh**  
*Ph.D.*

**Файсел М. Е. Сардейх**  
*Кандидат технических наук*  
**Faisal M. E. Sardieh**  
*Ph.D.*

## МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ СИСТЕМЫ РАСЧЕТА ОСНОВНЫХ ПАРАМЕТРОВ СЕРВЕРНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ И СЕТЕЙ

## MATHEMATICAL MODELING OF THE SYSTEM OF CALCULATING THE MAIN PARAMETERS OF SERVER COMPUTER SYSTEMS AND NETWORKS

**Аннотация.** Разработана математическая модель для расчета основных параметров серверных вычислительных систем и сетей, позволяющая повысить эффективность использования серверных компьютерных ресурсов. Представлены результаты работы модели при расчете параметров серверной системы для электронной коммерции, ориентированной на торговлю автомобилями.

**Ключевые слова:** Математическая модель, сетевая инфраструктура, сервер, нагрузка.

**Summary.** A mathematical model is developed for calculating the main parameters of server computing systems and networks, which makes it possible to increase the efficiency of server computer resources. The results of the model work are presented in the calculation of the parameters of the server system for e-commerce oriented to trade in cars.

**Key words:** Mathematical model, network infrastructure, server, load.

Реально существующую деятельность по математическому моделированию невозможно в полном объеме охватить никаким определением. Однако, учитывая направление исследований для данной работы, можно прибегнуть к концепции, высказанной А. А. Самарским и А. П. Михайловым [1, с. 7], определившим математическую модель как «эквивалент» объекта, отражающий в математической форме важнейшие его свойства — законы, которым он подчиняется, связи, присущие составляющим его частям, и т.д.». Авторы предлагают рассматривать математическую модель в виде триады «модель-алгоритм-программа»: «Создав триаду «модель-алгоритм-программа», исследователь получает в руки универсальный, гибкий и недорогой инструмент, который вначале отлаживается, тестируется в пробных вычислительных экспериментах. После того как адекватность (достаточное соответствие) триады исходному объекту установлена, с моделью

проводятся разнообразные и подробные «опыты», дающие все требуемые качественные и количественные свойства и характеристики объекта» [1, с. 7–8].

Таким образом, математическое моделирование системы расчета основных параметров серверных вычислительных систем и сетей можно построить по описанному выше принципу, причем совокупность «модель-алгоритм-программа» позволяет в достаточной мере «просчитать» все качественные и количественные свойства исследуемой системы, на основании которых можно сделать достоверный прогноз об эффективности использования серверных компьютерных ресурсов.

Поскольку рассматриваемая математическая модель системы расчета основных параметров серверных вычислительных систем и сетей является не самоцелью, а только средством для решения определенных задач, в частности, повышения эффективности

использования серверных компьютерных ресурсов, то необходимость создания математической модели вытекает из выбираемой методологии решения поставленных задач. Однако для решения сложных задач целесообразно использовать так называемый **системный подход**, в котором моделирование является также одним из основных методов исследования [2]. В целом системный подход предполагает обширные этапы решения поставленной задачи, однако, в рамках расчета основных параметров серверной инфраструктуры, можно выделить следующие:

- 1) изучение предметной области;
- 2) формальная постановка задачи моделирования;
- 3) математическое моделирование исследуемых объектов и процессов;
- 4) обработка результатов моделирования;
- 5) формулирование и оценка альтернативных решений;
- 6) формулирование выводов и предложений по повышению эффективности использования исследуемых объектов.

Первые два пункта можно соотнести к составляющей триады «модель»; пункт 3 — к составляющей «алгоритм»; пункт 4 — к составляющей «программа». Пункты 5 и 6 являются результатом моделирования и фактически представляют собой обратную связь, корректирующую входные параметры с учетом полученных результатов.

Таким образом, процесс исследования можно представить в виде формальной системы, представляющей собой совокупность функций выходов (1), переходов (2) и управления процессом (3):

$$Y_{(t)} = f(X_{(t)}, Q_{(t)}), \tag{1}$$

$$Q_{(t)} = g(X_{(t)}, Q_{(t-1)}), \tag{2}$$

$$X_{(t)} = h(Y_{(t-1)}). \tag{3}$$

Здесь  $X_{(t)}$  — множество значений входных факторов в обозначенный момент времени  $t$ ,  $Q_{(t)}$  — множество значений параметров, характеризующих различные внутренние состояния сложной системы в этот же момент времени,  $Y_{(t)}$  и  $Y_{(t-1)}$  — множества значений измеряемых показателей изучаемых свойств системы в соответствующие моменты времени. Уравне-

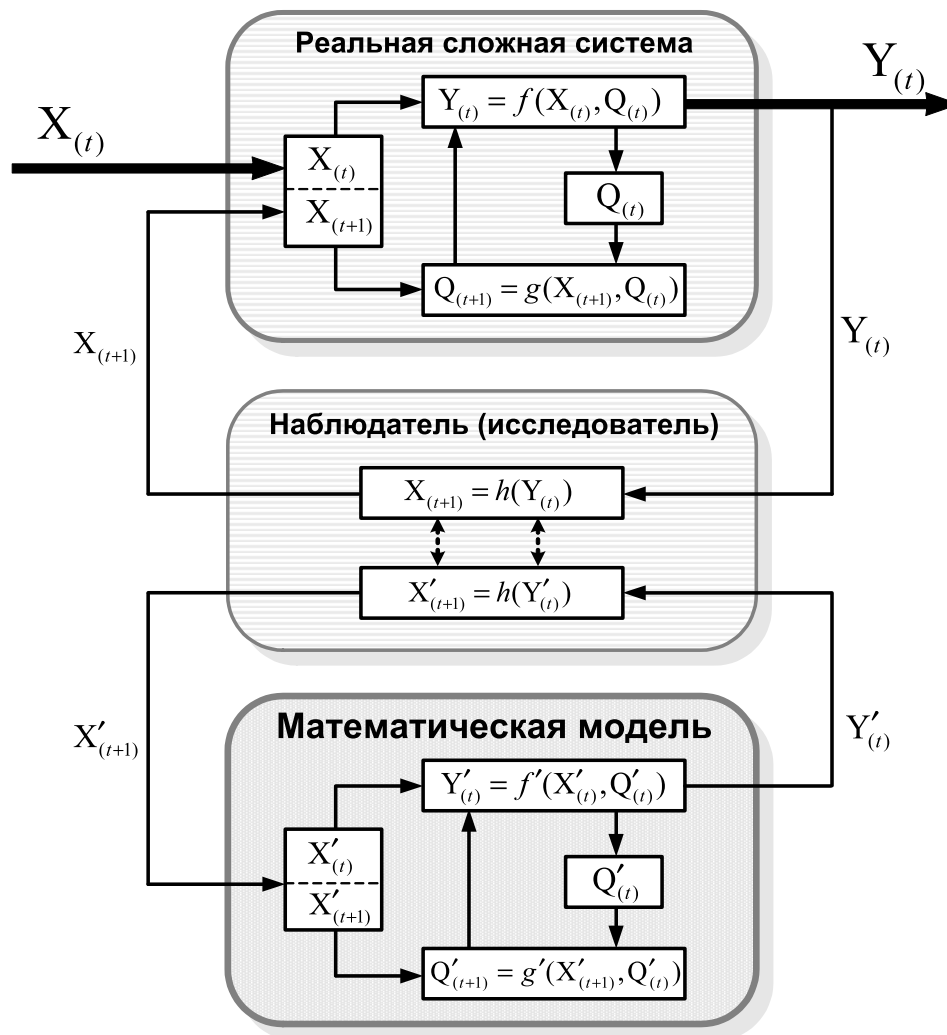


Рис. 1. Формализованная схема процесса исследования сложной системы [4]

ния (1) и (2) моделируют суть изучаемого процесса. Уравнение (3) является математическим описанием (т.е. непосредственно математической моделью) процесса воздействий наблюдателя (исследователя) на изучаемую систему. Наблюдателю доступно только определенное подмножество  $Y'_{(t)}$  наблюдаемых параметров и весьма ограниченное подмножество управляемых факторов  $X'_{(t)}$ . Представление наблюдателя о внутренних состояниях исследуемой системы также ограничено некоторым подмножеством  $Q'_{(t)}$ . Поэтому в представлении исследователя математическая модель исследуемой им системы имеет вид:

$$Y'_{(t)} = f'(X'_{(t)}, Q'_{(t)}), \quad (4)$$

$$Q'_{(t+1)} = g'(X'_{(t+1)}, Q'_{(t)}). \quad (5)$$

На рисунке 1 приведена формализованная схема процесса исследования сложной системы.

Приложения, основанные на Web-технологиях, строятся в соответствии с комплексной (многоуровневой) архитектурой [3, с. 480]. Уровень 1 (уровень представления) реализует интерфейс пользователя с сетью. Уровень 2 (прикладной уровень) включает в себя набор правил для реализации логики приложений. Отделение прикладного уровня от уровня представления и обработки данных (уровень 3) дает большую автономию для приложений и делает их более надежными [4].

Комплексные архитектуры (рис. 2) включают в себя балансировщик нагрузки, Web-серверы на

уровне 1, серверы приложений на уровне 2 и серверы баз данных на уровне 3. Эти различные элементы взаимосвязаны через разнообразные локальные сети, маршрутизаторы и брандмауэры.

Путь запроса для трехуровневой архитектуры может быть следующим:

1. Запросы поступают на балансировщик нагрузки через маршрутизатор, который соединяет сайт с Интернетом.

2. Балансировщик нагрузки [5] решает, какой из Web-серверов должен получить запрос.

3. Брандмауэр 1 изолирует сеть от внутренней части сайта (зона, доступная для внешнего мира, называется «демилитаризованной», DMZ – demilitarized zone) и гарантирует, что любой Web-сервер на уровне 1 получает запросы только от балансировщика нагрузки, а не из внешнего мира.

4. Брандмауэр 2 гарантирует, что серверы приложений соединяются только через Web-сервер.

5. Если серверу приложений необходимы данные из сервера баз данных, запрос должен пройти через брандмауэр 3.

При анализе свойств структурных элементов серверов одного типа в комплексной архитектуре его целесообразно рассматривать как типовой элемент сложной информационно-вычислительной системы. При этом можно более детально представить указанную на рис. 2 комплексную архитектуру, которая

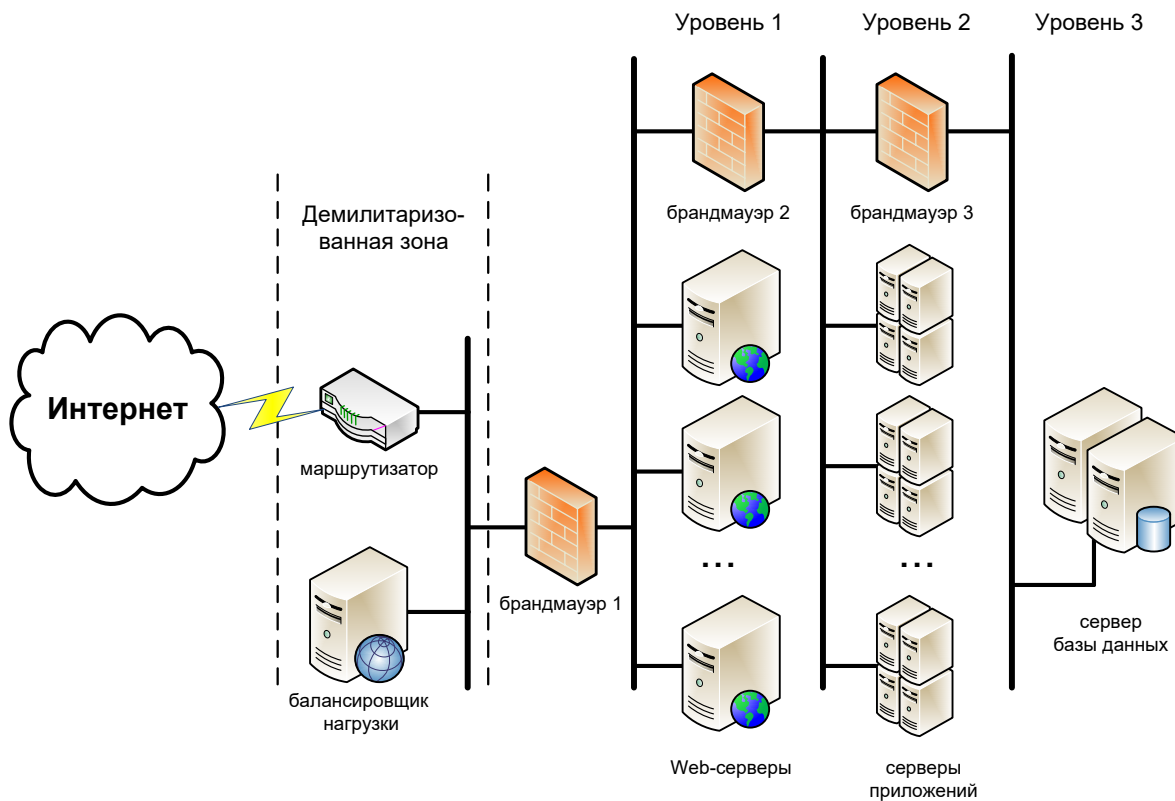


Рис. 2. Типовая трехуровневая архитектура [6, с. 348]

в общем виде выглядит так, как это представлено на рисунке 3.

Обозначения на рисунке приняты следующие:

ISP – Internet Service Provider, обеспечивающий подключение серверного пула к глобальной инфраструктуре Интернет;

LAN1 и LAN2 – локальные сети, обеспечивающие взаимодействие серверов;

WS – Web-server, обслуживающий запросы по HTTP-протоколу;

AS – Application-server, обеспечивающий доступ к прикладным программам;

DB – DataBase-server, обеспечивающий работу с базами данных.

При этом CPU означает процессорную часть соответствующих серверов, а IO – подсистему ввода-вывода.

Каждая задача характеризуется значениями определенных свойств, которыми являются:

- $P_{запр}$  – общий поток запросов в единицу времени, застр/ч;
- $V$  – пропускная способность сети, Мб/с;
- $K_{пр}^{CPU}$  – коэффициент пропускной способности процессорного блока (server CPU) серверов системы;
- $K_{пр}^{IO}$  – коэффициент пропускной способности блока ввода-вывода (server IO) серверов системы;
- $k_{запр}$  – количество передаваемых системе запросов;
- $t_{запр}$  – время обслуживания одного запроса, мс.

Структурные элементы каждого сервера обладают следующими свойствами:

- устройства ввода-вывода – коэффициентами пропускной способности ( $r_{in/out}$ );
- устройства памяти – объемом памяти ( $C_{RAM}, C_{HDD}$ );
- процессоры – коэффициентом быстродействия ( $r_{CPU}$ ).

Определение принципов построения математической модели данного процесса следует начать с определения характера потока задач, поступающих на вход системы и принципов его математического описания. При этом все решаемые задачи целесообразно разде-

лять на две группы: к первой группе можно отнести случайно возникающие задачи, требующие одноразового решения, а ко второй группе – задачи, требующие периодического решения с учетом новых исходных данных. Данные положения принимаются как допущения, упрощающие процесс математического моделирования исследуемой системы [7, с. 366].

Таким образом, в итоге концептуального проектирования модели установлено следующее:

- каждый поступающий на вход системы запрос представляется вектором  $x_i = (i, z_1, \dots, z_j)$ , где  $i$  – номер запроса, а  $z_j$  – конкретное значение  $j$ -того параметра для  $i$ -го запроса;
- запросы поступают на вход системы в виде потока  $X_{(t)}$  с периодичностью запросов  $f(X)$ ;
- множества запросов характеризуются средним значениями  $m(z_j)$  параметров  $z_j$  запросов;
- на выходе системы результаты представляются вектором  $y_i = (i, t_i^{обп}, t_i^{кон})$ , где  $t_i^{обп}$  – время затраченное системой на обработку  $i$ -того запроса, а  $t_i^{кон}$  – время окончания обработки запроса;
- в процессе моделирования определяются и записываются количество  $k_{запр}$  поступивших на вход системы и количество  $k_{обп}$  обработанных запросов за время моделирования  $t_{мод}$ , и, при необходимости, максимальные значения объемов оперативной  $C_{RAM}^{max}$  и дисковой  $C_{HDD}^{max}$  памяти;
- в данной модели не существует аналитических уравнений связывающих ее входные факторы с выходными показателями, поэтому функция выхода системы представляет алгоритм  $f_1: f_1[X_{(t)}, Q_{(t)}] = Y_{(t)}$ , описывающий процесс обработки входного потока запросов  $X_{(t)}$  в поток  $Y_{(t)}$  отклика системы и оценок быстродействия  $W_{(t)}$ ;
- внутреннее состояние  $Q_{(t)}$  системы определяется показателями надежности и занятости ее устройств, поэтому  $Q_{(t+1)} = f_2[X_{(t+1)}, Q_{(t)}]$ , где  $Q_{(t)}$  является, по сути, декартовым произведением множеств текущих  $q_i$  состояний компонентов системы, а  $f_2$  – алгоритмом, описывающим функцию переходов от  $Q_{(t)}$  к  $Q_{(t+1)}$ .

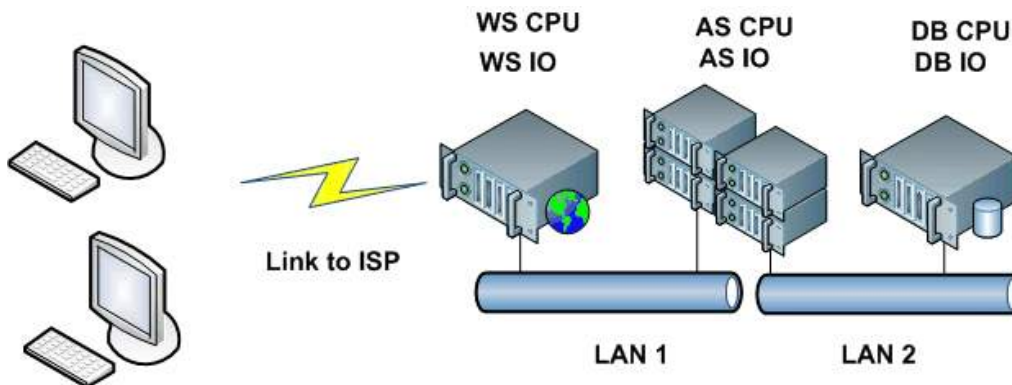


Рис. 3. Компоненты типичной серверной инфраструктуры [разработка автора]

С учетом этих положений обобщенная концептуальная математическая модель процесса обработки запросов для сетевой трехуровневой архитектуры может быть представлена в виде, показанном на рисунке 4:

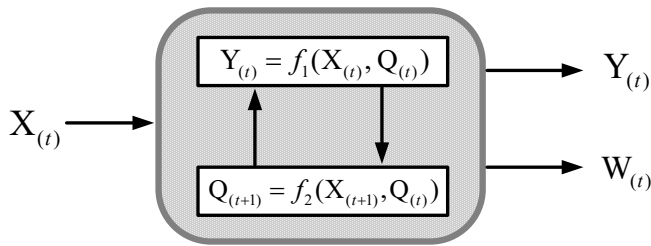


Рис. 4. Обобщенная схема математической модели рассматриваемой системы [5]

Для получения результирующих значений параметров рассматриваемой системы необходимо сформировать набор исходных значений определенных свойств. Такими значениями являются:

1. Общий поток запросов:  $P_{запр} = 90\,000$  запр/ч.
2. Пропускная способность сетей:  $V_{LAN1} = 10$  Мб/с — для сети, обеспечивающей соединения Web-серверов

с серверами приложений,  $V_{LAN2} = 100$  Мб/с — для сети, обеспечивающей соединение серверов приложений и баз данных,  $V_{ISP} = 1,5$  Мб/с — для сети, обеспечивающей подключение серверного пула к глобальной инфраструктуре Интернет.

3. Коэффициенты пропускной способности процессорного блока и блока ввода-вывода серверов системы:

- для Web-сервера:  $KW_{пр}^{CPU} = 1,8$ ,  $KW_{пр}^{IO} = 1$ ;
- для сервера приложений:  $KA_{пр}^{CPU} = 1,4$ ,  $KA_{пр}^{IO} = 1$ ;
- для сервера базы данных:  $KD_{пр}^{CPU} = 1,7$ ,  $KD_{пр}^{IO} = 1$ .

На первом этапе производится расчет параметров серверной компьютерной системы:

1) Для Web-сервера:

$$tW^{CPU} = w_{1,i} \cdot KW_{пр}^{CPU}$$

где  $tW^{CPU}$  — время обслуживания одного запроса для блока CPU;  $w_{1,i}$  — коэффициент нагрузки CPU Web-сервера для отработки i-ой функции электронной коммерции;

$$tW^{IO} = w_{2,i} \cdot KW_{пр}^{IO}$$

где  $tW^{IO}$  — время обслуживания одного запроса для блока ввода-вывода;  $w_{2,i}$  — коэффициент нагрузки IO

Производительность		1000	500	2500	2500	2000	1800	Пропускная способность				
		MIPS	MIPS	MIPS	MIPS	MIPS	MIPS	LAN 1	LAN 2	Link to ISP		
Базовое количество запросов	90 000 запр/час							10	100	1,5		
								Mbps	Mbps	Mbps		
Время обслуживания одного запроса (мс)												
E-business Function	F	Кол-во запросов	WS CPU	WS IO	AS CPU	AS IO	DB CPU	DB IO	LAN 1	LAN 2	Link to ISP	Общее время
Выбор автомобиля	1	90 000	5,00	10,00	21,60	7,60	14,50	16,11	4,92	0,53	16,38	96,64
Выбор опций	2	85 500	4,80	9,60	19,20	19,20	24,00	26,67	3,28	0,35	12,01	119,11
Выбор цвета	3	63 000	4,70	9,40	19,20	11,20	14,00	15,56	2,87	0,33	12,01	89,27
Доп.сервис	4	36 000	5,10	10,20	21,20	9,20	11,50	12,78	2,95	0,49	11,47	84,89
Ввод перс.данных	5	3 600	30,50	7,00	12,20	12,20	0,00	0,00	6,55	0,00	32,77	101,22
Оплата	6	2 700	32,00	6,40	12,80	12,80	0,00	0,00	7,78	0,00	21,85	93,63
Заказ доставки	7	2 700	30,00	6,00	12,00	12,00	0,00	0,00	4,10	0,00	19,11	83,21
Доставка	8	2 700	31,00	6,20	12,40	12,40	15,50	17,22	8,19	0,90	43,69	147,51
Проверка состояния	9	1 800	5,20	10,40	8,84	8,84	11,05	12,28	2,05	0,23	27,31	86,19
Отмена заказа	10	900	5,30	10,60	9,20	9,20	11,50	12,78	2,46	0,26	30,04	91,33

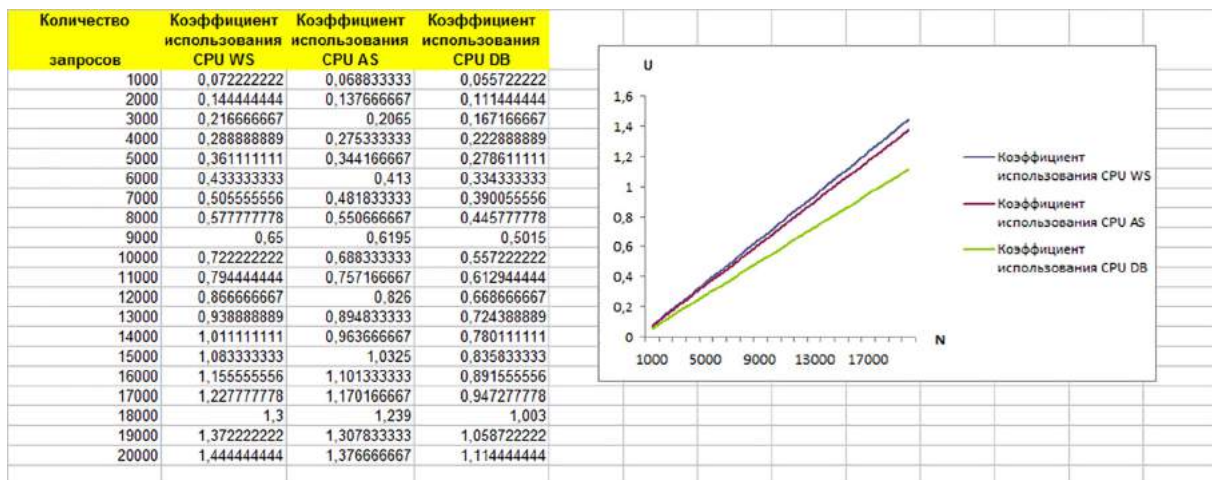


Рис. 5. Реализация расчета факторов, влияющих на эффективность функционирования серверной системы для электронной коммерции, ориентированной на торговлю автомобилями [4, 5]

Web-сервера для обработки  $i$ -ой функции электронной коммерции;

2) Для сервера приложений:

$$tA^{CPU} = a_{1,i} \cdot KA_{np}^{CPU},$$

где  $tA^{CPU}$  — время обслуживания одного запроса для блока CPU;  $a_{1,i}$  — коэффициент нагрузки CPU сервера приложений для обработки  $i$ -ой функции электронной коммерции;

$$tA^{IO} = a_{2,i} \cdot KA_{np}^{IO},$$

где  $tA^{IO}$  — время обслуживания одного запроса для блока ввода-вывода;  $a_{2,i}$  — коэффициент нагрузки IO сервера приложений для обработки  $i$ -ой функции электронной коммерции;

3) Для сервера базы данных:

$$tD^{CPU} = d_{1,i} \cdot KD_{np}^{CPU},$$

где  $tD^{CPU}$  — время обслуживания одного запроса для блока CPU;  $d_{1,i}$  — коэффициент нагрузки CPU сервера БД для обработки  $i$ -ой функции электронной коммерции;

$$tD^{IO} = d_{2,i} \cdot KD_{np}^{IO},$$

где  $tD^{IO}$  — время обслуживания одного запроса для блока ввода-вывода;  $d_{2,i}$  — коэффициент нагрузки IO сервера БД для обработки  $i$ -ой функции электронной коммерции;

На рисунке 5 показана реализация расчета параметров серверной системы для электронной коммерции, ориентированной на торговлю автомобилями, выполненная в среде Excel. Предлагаемый пример предполагает возможность достаточного гибкого задания входных данных.

При представленной в таблице комбинации исходных значений необходимой и достаточной является серверная компьютерная система, состоящая из одного веб-сервера, 4-х серверов приложений и 2-х серверов баз данных.

**Выводы.** Математическое моделирование системы расчета основных параметров серверных вычислительных систем и сетей может быть адаптированным для определения оценки эффективности функционирования комплексных серверных компьютерных систем. В работе предложен новый способ расчета факторов, позволяющий сбалансировать характеристики их компонентов и обеспечить эффективную эксплуатацию.

Предлагаемый вариант расчетной модели предполагает возможность достаточного гибкого задания в качестве исходных значений, как параметров нагрузки, так и характеристик аппаратных средств.

### Литература

1. Самарский А. А. Математическое моделирование. Идеи. Методы. Примеры. / А. А. Самарский, А. П. Михайлов А. П. — 2-е изд., испр. — М.: Физматлит, 2001.
2. Устенко А. С. Основы математического моделирования и алгоритмизации, 2000. Электронный ресурс. Режим доступа: <http://ustenko.fromru.com/index.html>.
3. Менаске Д. Протвостссльность Wcb-служб. Анализ, оценка и планирование: Пер. с англ./ Дэниел А. Менаске, Виргилио А. Ф. Алмейда. — СПб: ООО «ДиаСофтЮП», 2003. — С. 480.
4. Lewandowski S. Frameworks for Component-Based Client/Server Computing, ACM Computing Surveys, vol. 30, no. 1, 1998.
5. Балансировка нагрузки. Материал из Википедии — свободной энциклопедии. Электронный ресурс. Режим доступа: [http://ru.wikipedia.org/wiki/Балансировка\\_нагрузки](http://ru.wikipedia.org/wiki/Балансировка_нагрузки).
6. Аноприенко А. Я. Высокопроизводительные информационно-моделирующие среды для исследования, разработки и сопровождения сложных динамических систем / А. Я. Аноприенко, В. А. Святный // Научные труды Донецкого государственного технического университета. Выпуск 29. Серия «Проблемы моделирования и автоматизации проектирования динамических систем». — Севастополь: «Вебер». — 2001. — С. 346–367.
7. Алиев Т. И. Основы моделирования дискретных систем. — СПб: СПбГУ ИТМО, 2009. — С. 363.
8. Советов Б. Я. Моделирование систем: Учебник для вузов. / Б. Я. Советов, С. А. Яковлев. — 4-е изд., стер. — М.: Высшая школа, 2005. — С. 343.