УДК. 621

А.А. Полков

ОБЕСПЕЧЕНИЕ ТРЕБУЕМОГО ВРЕМЕНИ ОТВЕТА ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ ПУТЕМ БАЛАНСА НАГРУЗОК

Полков Андрей Андреевич, *окончил факультет прикладной информатики (в экономике) Московской академии рынка труда и информационных технологий. Ведущий инженер-программист ОАО «НИИАА». [e-mail: andr.polkov@yandex.ru].*

Аннотация

В статье рассматривается имитационная модель информационно-вычислительной системы с клиент-серверной архитектурой. Разработана методика моделирования, в основе которой лежат задачи обнаружения и устранения «узких мест» и обеспечения сбалансированности нагрузок.

Ключевые слова: информационно-вычислительная система, имитационная модель, «узкие места», баланс нагрузок.

Введение

Неудовлетворительное время ответа информационно-вычислительных систем (ИВС) часто связано с их несбалансированностью по нагрузкам или с возникновением «узких мест». Существуют различные способы преодоления этой проблемы:

1) подбор требуемой производительности отдельных модулей, перераспределение между ними решаемых задач и др.;

2) устранение существующих «узких мест».

Эксперименты по выявлению указанных проблем в работающей системе, как правило, требуют значительных затрат. Одним из широко практикуемых подходов является использование для этих целей математических моделей, которые позволяют выполнить множество экспериментов по определению требуемых режимов функционирования, обоснованию решений по устранению «узких мест» и повышению уровня сбалансированности. В качестве таких моделей могут использоваться аналитические, имитационные и комбинированные модели [1, 2]. Несмотря на множество различных моделей, проблема разрешения «узких мест» для систем реального времени, в частности, систем, имеющих клиент-серверную архитектуру, все еще продолжает оставаться актуальной [3, 4]. Наибольшее практическое применение находят имитационные модели, в связи с тем что в них легко отразить особенности исследуемых систем. В данной статье приведена имитационная модель многоуровневой клиент-серверной системы, разработана методика ее использования и прокомментированы результаты моделирования. Применение этой модели позволило не только оценить время реакции исследуемой системы при различных условиях ее функционирования, но и разрешить часто возникающие критические ситуации, когда интуиция не может подсказать правильный ход

действий. Например, увеличение производительности одного из узлов системы, являющегося «узким местом», в распределенных системах часто является обычным решением; однако оно может привести не к уменьшению, а к увеличению времени ответа на запросы пользователей.

Описание системы и модели

Рассмотрение подходов к построению и использованию моделей ИВС удобней всего вести на конкретном примере. Далее для этих целей будем рассматривать систему с клиент-серверной архитектурой (рис. 1).

Считаем, что в систему поступает три класса потоков запросов:

– локальные запросы (терминальный модуль 1), связанные с управлением комплексом средств автоматизации (КСА);

– информационные сообщения (терминальный модуль 2);

– фоновый поток, создающий нагрузку на серверы КСА.

Локальные запросы с первого терминального модуля поступают непосредственно на соответствующие серверы через локальную сеть (LAN).

Сообщения (2-й класс запросов) поступают с удаленных терминалов через территориальную и локальную сети (WAN и LAN) на веб-сервер и далее распределяются по другим серверам.

Фоновый поток работ аналогичным образом доводится до серверов приложений и баз данных (БД) и выполняется ими в фоновом режиме.

Потоки запросов и время их выполнения являются независимыми друг от друга случайными величинами с произвольными законами распределения. В приведенных ниже расчетах для простоты интерпретации использовались экспоненциальные законы времени выполнения запросов в отдельных устройствах обслуживания. В каждом сервере организовано приоритетное выполнение запросов: 1-й класс запросов имеет наивысший приоритет; следующий, 2-й класс имеет

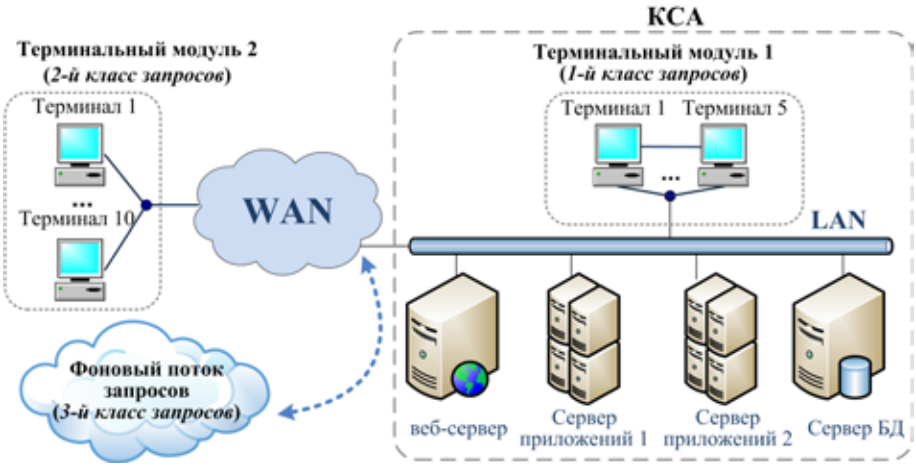

Рис. 1. Структура исследуемой системы

более низкий приоритет по сравнению с 1-м. Фоновые работы выполняются в периоды отсутствия запросов 1-го и 2-го классов.

Структура имитационной модели системы на данном уровне рассмотрения совпадает со структурой, указанной на рисунке 1. Приоритеты обслуживания запросов неодинаковы в различных узлах: выполнение запросов на центральных процессорах (ЦП) ведется с абсолютным приоритетом, а на дисковых накопителях – с относительными приоритетами.

В таблице 1 приведены основные характеристики модели и некоторые исходные данные для имитационных экспериментов и интерпретации результатов.

Таблица 1

Характеристики узлов имитационной модели

Узел	Характеристики
Терминальный модуль 1	1) поток запросов ограничен количеством терминалов (число терминалов – 5); 2) среднее время обдумывания полученного ответа – 15 с
Терминальный модуль 2	1) поток запросов ограничен количеством терминалов (число терминалов – 10); 2) среднее время обдумывания полученного ответа – 20 с
Фоновый поток запросов	Поток запросов неограниченный (экспоненциальный) с интенсивностью 3,7 запросов / с
Территориальная сеть (WAN)	Среднее время передачи запросов: 1) 2-й класс – 0,5 с; 2) 3-й класс – 1 с
Локальная сеть (LAN)	Среднее время передачи запроса – 0,001 с
Веб-сервер	Состоит из ЦП (2 ядра) и одного дискового накопителя
Сервер приложений 1	Состоит из ЦП (2 ядра) и двух дисковых накопителей
Сервер приложений 2	Состоит из ЦП (4 ядра) и двух дисковых накопителей
Сервер БД	Состоит из ЦП (4 ядра) и шести дисковых накопителей

Помимо параметров законов распределения потоков и обслуживания в сетевой модели задаются также маршрутные вероятностные матрицы, регулирующие движение запросов по сети. На рисунке 2 приведена типовая структура сервера, в которой для примера указаны отдельные вероятности маршрутной матрицы.

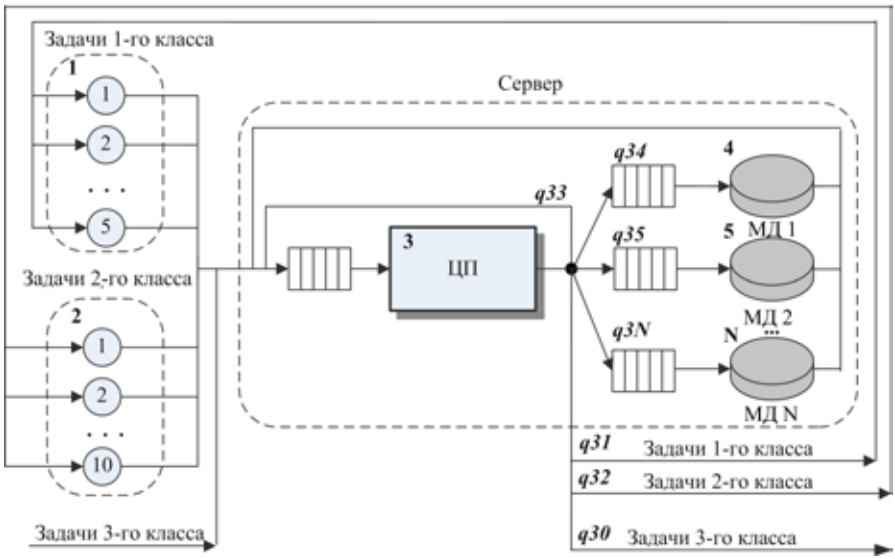

Рис. 2. Типовая структура сервера

Методика обеспечения баланса нагрузки в ИВС с использованием имитационной модели

Методика использования имитационной модели для оценки «узких мест» и обеспечения сбалансированности состоит из выполнения следующих укрупненных фаз.

Фаза 1. Установление на модели значений основных параметров исходя из требований к системе и принятых проектных решений и оценка времени ответа системы на поступающие запросы разных классов [5]. При невыполнении требований к этому времени осуществляется переход к выявлению возможных причин наличия в системе «узких мест» и несбалансированности нагрузок в результате моделирования исходного варианта.

Фаза 2. Итерационное исключение «узких мест» в системе по результатам моделирования:

1) обнаружение «узких мест» по уровням нагрузки на отдельные устройства сети («узким местом» считается устройство, нагрузка на которое превышает 98 %);

2) принимается одно из решений, которое может привести к разгрузке указанного устройства (в качестве решений могут выступать: увеличение производительности устройства, снижение нагрузки от менее важных задач, перенаправление потоков запросов сети и др.);

3) выполнение моделирования ИВС при выбранном новом сценарии и оценка времени ответа; при нарушении требований к нему производится возврат к первому пункту данной фазы;

4) в случае отсутствия «узких мест» осуществляется переход к следующей фазе.

Фаза 3. Обеспечение сбалансированности нагрузок в системе при неудовлетворительном времени ответа и отсутствии «узких мест»:

1) рассматриваются последние результаты моделирования и проводится анализ возможностей уменьшения задержек в наиболее загруженных узлах сетевой модели;

2) проводятся мероприятия по снижению времени задержек в нагруженных узлах (увеличение производительности соответствующих процессорных устройств и обеспечение более высокого параллелизма процессов, расширение дискового пространства, перераспределение информации и потоков между узлами сети и др. Крайней мерой может служить решение о прекращении обслуживания менее важных потоков запросов).

Фаза 4. Исключение излишних ресурсов по результатам моделирования и оценка стоимости результирующего варианта ИВС.

Результаты моделирования ИВС

Рассмотрим более детально фазы методики на примере описанной выше имитационной модели ИВС, выполненной в моделирующей системе AnyLogic [6]. Считаем, что ко времени обработки сообщений (запросы 2-го класса) в сети выдвинуто внешнее требование – это время в среднем не должно превышать 8 секунд. Время выполнения остальных классов запросов в данном примере не анализируется. Получение вероятностно-временных характеристик выполнения запросов производится по той же схеме. С целью сокращения объема статьи результаты моделирования и анализа приводятся только на примере серверов приложений и общего времени ответа системы.

Фаза 1. Устанавливаются исходные параметры распределений входных потоков и интервалов обслуживания, и задаются маршрутные матрицы сетевой имитационной модели.

Полученное время ответа на модели оказалось равным 10,4 с. Причиной явилась перегрузка сервера приложений 1 – «узкое место» (табл. 2).

Таблица 2

Показатели моделирования итерации 1

Нагрузка						Среднее время ответа		
сервер приложений 1			сервер приложений 2			1-й класс запросов	2-й класс запросов	3-й класс запросов
ЦП	диск 1	диск 2	ЦП	диск 1	диск 2			
99 %	12 %	13 %	22 %	85 %	85 %	3,5	10,4	442

Так как обнаружено «узкое место» в сети, то осуществляется переход к выполнению фазы 2 методики. Одним из способов повышения производительности ЦП сервера приложений 1 является распараллеливание задач и увеличение количества ядер (с двух до шести). В результате после моделирования данного сценария указанное «узкое место» устранено (загрузка ЦП снизилась до уровня 69 %, табл. 3). Однако среднее время ответа на запросы 2-го класса не снизилось, а увеличилось до 12,3 с. Анализ результатов моделирования показал, что

в системе возникло новое «узкое место» – дисковые накопители сервера приложения 2. Причиной является то, что часть потока, задерживаемого ранее в сервере приложений 1, за счет устранения в нем «узкого места» перераспределилась на сервер приложений 2, увеличив загрузку дисковых накопителей (табл. 3). Кроме того, влияние оказывает то, что дисковые накопители работают с относительным приоритетом.

Таблица 3

Показатели моделирования итерации 2

Нагрузка						Среднее время ответа		
сервер приложений 1			сервер приложений 2			1-й класс запросов	2-й класс запросов	3-й класс запросов
ЦП	диск 1	диск 2	ЦП	диск 1	диск 2			
69 %	35 %	35 %	39 %	99 %	99 %	4,2	12,3	467

Указанные «узкие места» устраняются путем увеличения количества дисковых накопителей и перераспределения между ними данных. В результате очередного моделирования данного варианта получаем новые значения нагрузок и времени ответа (табл. 4). Из полученных результатов моделирования видно, что среднее время ответа удовлетворяет требуемому значению (равно 7,6 с при требуемом значении 8 с). При этом загрузка всех устройств серверов приложений становится практически сбалансированной (загрузка устройств лежит в пределах от 57 % до 74 %).

Таблица 4

Показатели моделирования итерации 3

Нагрузка								Среднее время ответа		
сервер приложений 1			сервер приложений 2					1-й класс запросов	2-й класс запросов	3-й класс запросов
ЦП	диск 1	диск 2	ЦП	диск 1	диск 2	диск 3	диск 4			
70 %	57 %	57 %	69 %	74 %	74 %	64 %	65 %	3,3	7,6	11,4

В результате того, что исключение «узких мест» привело к требуемому времени ответа для 2-го класса запросов и одновременно обеспечило баланс нагрузок, последующие две фазы методики в данном примере не выполняются.

Выводы

Существует большое число методов и методик оценки производительности информационно-вычислительных систем. В настоящей статье рассматривается апробированная на практике методика балансирования нагрузок, обнаружения и устранения «узких мест» в ИВС. Указанные в методике фазы анализа согласованы с технологией моделирования Any Logic, с использованием которой и была разработана соответствующая модель. Особенностью данной модели является легкость

в перенастройке ее структуры и гибкость интерфейсов, облегчающих и ускоряющих проведение имитационных экспериментов.

Рассмотренный пример применения методики поиска «узких мест» и обоснования проектных решений в сложных информационных системах указывает на важность использования моделей для анализа вычислительного процесса и обоснования решений, не всегда достижимых на интуитивном уровне.

СПИСОК ЛИТЕРАТУРЫ

1. Denning P.J., Buzen J.P. Operational analysis of queueing network models // Computing Surveys. – 1978, Vol. 10, No. 3. – pp. 225–261.
2. Литвин В.Г., Аладышев В.П., Винническо А.И. Анализ производительности мультипрограммных ЭВМ. – М. : Финансы и статистика, 1984. – С. 263.
3. Лукьянов В.С., Слесарев Г.В. Проектирование компьютерных сетей методами имитационного моделирования. – Волгоград, 2001. – С. 65.
4. Monte Carlo and Quasi-Monte Carlo Methods 2010 // Springer, New York, 2010. – P. 732.
5. Елизабет Х., Кен Д., Джереми Д. Разработка и управление требованиями. – М. : Комания Telelogic, 2005. – С. 240.
6. Карпов Ю. Имитационное моделирование систем. Введение в моделирование с AnyLogic 5. – СПб. : БХВ Петербург, 2005. – С. 400.