

УДК 004.942.00

АНАЛИЗ ПРОИЗВОДИТЕЛЬНОСТИ РАСПРЕДЕЛЕННЫХ  
ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМ  
МЕТОДАМИ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ

А.С. Овдина, И.Е. Жигалов

*Владимирский государственный университет, г. Владимир*

В статье рассмотрены альтернативные подходы к анализу производительности телекоммуникационных систем. Описан подход к анализу производительности распределенных телекоммуникационных систем методом имитационного моделирования на основе раскрашенных сетей Петри. Была принята верифицированная модель для исследований возможностей системы с точки зрения производительности. при исследовании характеристик поведения не только существующей системы в различных режимах работы, но и при исследовании еще не реализованных решений. Экспериментально получены зависимости количества занятых потоков от количества запросов клиента и количества ответов сервера.

*Моделирование распределенных систем, раскрашенные сети Петри*

**Введение**

Производительность часто является основной проблемой при проектировании, разработке и конфигурировании систем. Не всегда достаточно, чтобы

система работала, необходимо, чтобы система отвечала требованиям производительности. Соответственно, решение проблем связанных с производительностью, так или иначе, влияет на конечную стоимость или на время разработки программной системы. Чтобы это влияние было предсказуемым, применяют технику превентивного управления производительностью, одним из важных аспектов которой является анализ производительности.

### **Постановка задачи**

Существует три альтернативных подхода к анализу производительности системы: измерение, аналитические модели и имитационные модели. Каждый из этих подходов обладает рядом преимуществ и недостатков.

Измерение может обеспечить точные ответы относительно производительности системы. При этом система исследуется без упрощения ее поведения и без исключения из процесса измерения каких-либо частей системы. С другой стороны, измерение актуально только при наличии исследуемой системы, в случае же отсутствия исследуемой системы данный подход не может быть применен.

Аналитические модели, такие, в частности, как Марковские модели, позволяют получить достаточно точные результаты для анализа производительности, однако результаты могут быть неполными, в зависимости от упрощений, сделанных при их создании.

Анализ производительности на основе имитационного моделирования может применяться в качестве альтернативы аналитическому моделированию. Имитационное моделирование редко дает точные результаты, однако существует возможность рассчитать их точность. Более того, большие и более сложные модели могут быть созданы и проанализированы без ограничивающих упрощений. Однако, имитационное моделирование ко времени моделирования; также при исследовании производительности с использованием имитационного моделирования довольно сложно достичь необходимой точности моделирования.

Большое количество работ в этой области основано на использовании формальных методов анализа производительности. Под формальными методами подразумеваются математические языки, методы и средства для спецификации и верификации системы. Формальные методы обладают рядом существенных недостатков: нотации часто неясные, методы не масштабируемы, средства на основе формальных методов трудны в использовании.

### **Теоретический базис**

Тема анализа производительности на основе различных классов сетей Петри достаточно широко представлена в научных трудах. Большинство исследований фокусируется на использовании аналитических моделей, генерируемых из моделей сетей Петри. Однако следует отметить, что размер, сложность и по-

нятие времени в раскрашенных сетях Петри (РСП) препятствуют генерации и использованию аналитических моделей из РСП-моделей, в связи с чем анализ производительности с применением РСП основывается на имитационном моделировании, используемом для получения данных о производительности модели. Несмотря на это, РСП предоставляют возможность создания модели со значительно меньшим количеством упрощений, что позволяет получать более реалистичные результаты анализа моделирования.

В настоящее время РСП используется для моделирования систем промышленного масштаба. При этом, по сравнению с использованием аналитических моделей, РСП позволяют анализировать производительность таких систем, предъявляя существенно меньшие требования к ресурсам.

В качестве примера использования анализа производительности на основе имитационного моделирования рассмотрим одностраничное Web приложение (SPA, Single Page Application). Суть подхода заключается в создании такой клиент-серверной архитектуры приложения, при которой страница с сервера на клиент загружается один раз при старте приложения (т.е. в момент первого обращения клиента к серверу). В зависимости от дальнейших действий пользователя клиентская часть приложения асинхронно подгружает необходимые данные с сервера, при этом пользователь продолжает работать.

Пусть было разработано SPA приложение. После проведения тестовой эксплуатации заказчик выдвинул ряд новых требований, среди которых было необходимость получения количественных характеристик производительности приложения, а также способы повышения производительности.

Измерение производительности с помощью профайлеров показало, что система удовлетворяет требованиям. Но количественных характеристик собрать в полной мере не удалось в связи с ограниченными возможностями профайлеров. Более того, нет возможности проверить способы повышения производительности без их реализации.

Был разработан ряд РСП-моделей для проведения имитационного моделирования, в том числе модель для исследования производительности подсистемы, отвечающей за передачу данных между клиентом и сервером SPA приложения.

Основными показателями, влияющими на производительность этой подсистемы, являются: количество передаваемых сообщений, скорость передачи сообщений, время обработки сообщений сервером. Количество передаваемых сообщений напрямую зависит от количества пользователей. В SPA приложении данные запрашиваются клиентом асинхронно по мере необходимости. Пусть архитектура клиента была разработана таким образом, что отображаемая пользователю информация группируется в виде представлений. Пользователь может одновременно просматривать только одно представление. Представления в

свою очередь сформированы из графических элементов пользовательского интерфейса, данные для которых асинхронно запрашиваются у сервера.

Примером такого представления может послужить группа элементов отвечающих за отображение информации о сотрудниках: список сотрудников, имя выбранного из списка сотрудника, список отделов выбранного сотрудника, список подчиненных выбранного сотрудника, иерархия проектов выбранного сотрудника. Когда пользователь выбирает нужного сотрудника в списке сотрудников, все остальные элементы графического интерфейса запрашивают необходимую им информацию у сервера. Таким образом, каждое представление можно представить в виде, набора запросов серверу. Информация, полученная от сервера, обрабатывается асинхронно и отображается в соответствующих графических элементах. При выборе нового сотрудника в списке все элементы вновь обновляются. В силу того, что это распределенное приложение, иногда на запрос элемента представления ответ не получен. Тогда через определенный интервал времени (таймаут) элемент представления информирует пользователя, что данные не получены, и пользователь может обновить представление, чтобы повторно загрузить информацию.

В имитационной модели отображаются представления, имеющиеся в системе: набор запросов серверу, имитация реального сценария работы пользователя. Сформированные запросы помещаются в очередь запросов клиента, которая работает по принципу FIFO. В данной модели важными параметрами запроса являются: идентификатор запроса, который обеспечивает уникальность запроса, идентификатор представления, элемент которого инициировал запрос, и команда (функция, результат выполнения которой должен отослать сервер в виде ответа). В случае передачи запроса серверу, появляется еще один важный параметр – время передачи. Максимальное время ожидания клиентом ответа сервера задается требованием к системе. Переход выполняет отправку запроса, помещая его в буфер сети, откуда запрос через определенное время (время передачи запроса по сети) помещается в позицию, в которой содержатся запросы ожидающие ответ сервера, а также устанавливается признак того, что запрос еще не получил ответ сервера. Если по истечении таймаута ответ на запрос не получен, тогда формируется новый набор запросов серверу, имитируя действия пользователя, который обновляет представления пытаясь обновить информацию. При этом сбрасывается признак того, что запрос еще не получил ответ от сервера, и вновь сформированные запросы помещаются в очередь запросов клиента.

Ответы сервера помещаются в буфер ответов. Таким образом в ходе моделирования будут накапливаться запросы, ответы для которых получены за интервал времени меньший чем таймаут, и будут аккумулировать просроченные ответы. Скорость передачи сообщений регулируется с помощью задания правила определяющего временную задержку передачи по сети.

Время обработки сообщений сервером зависит от времени затрачиваемых сервером на обработку конкретного сообщения, а также от количества свободных потоков, выделяемых сервером для обработки сообщений. Выделяемые сервером потоки обработки сообщений образуют пул потока сервера. В случае не пустой очереди запросов занимается один из свободных потоков и он помещается в позицию занятых (выполняемых) потоков, откуда поток может быть изъят по истечении интервала времени определяющего время, затрачиваемое на выполнение текущего задания.

Таким образом, проходит общий цикл получения информации клиентом от сервера. Данная модель не учитывает потери запросов или ответов в сети. Это может быть сделано в качестве упрощения, либо в силу, того что вероятность потери ничтожно мала по сравнению с общим количеством отправленных пакетов, и поэтому, на первый план выдвигается проблема перегруженности сервера, и как правило появление просроченных ответов на стороне клиента.

Основными параметрами модели являются: количество пользователей (выражается в количестве представлений), максимальное время ожидания ответа, задержка в сети передаваемых запросов, задержка в сети передаваемых ответов, количество запросов на каждое представление, время затрачиваемое на обработку сервером каждого запроса, количество выделенных сервером потоков для обработки запросов. Эти параметры можно выставить на основе статистическом данных, которые были получены в ходе тестовой эксплуатации системы, с помощью ведения журналов действий пользователя и журналов событий в системе. Эти же данные позволили провести верификацию и аттестацию моделей после проведения имитационного моделирования.

### **Результаты экспериментов**

Верифицированная модель представляет интерес для дальнейших исследований возможностей системы с точки зрения производительности. Например, для исследования пиковой нагрузки на сервер были установлены следующие параметры модели. Количество пользователей: 600 пользовательских одновременных обращений (пиковая нагрузка) и затем 400 постоянных пользователей, таймаут: 4 секунды, задержка в сети передаваемых запросов и ответов: 0.2-0.3 секунды, время затрачиваемое сервером на обработку запросов: 0.3-1.0 секунды, количество выделенных сервером потоков для обработки запросов: 150. Результаты имитационного моделирования представлены на рисунке 1.

Можно отметить, что максимальная нагрузка сервера приходится на момент одновременного обращения 600 пользователей. При этом количество необходимых в обработке запросов равно 1100. Пиковая нагрузка спадает на 3 секундах, работа сервера постепенно нормализуется, но часть запросов не была принята в течение 4 секунд после отправки, пользователи получившие сообщение о том, что информация не получена, обновляют представления, посылая

еще 230 запросов (около 90 повторных обновленных представлений) серверу. Эти запросы создают очередную пиковую нагрузку на сервер, которая продолжается 1 секунду, после работа сервера окончательно нормализуется. Все оставшееся время моделирования сервер занимал до 41 потоков для обработки запросов клиента, обеспечивая бесперебойную работу 400 пользователей.

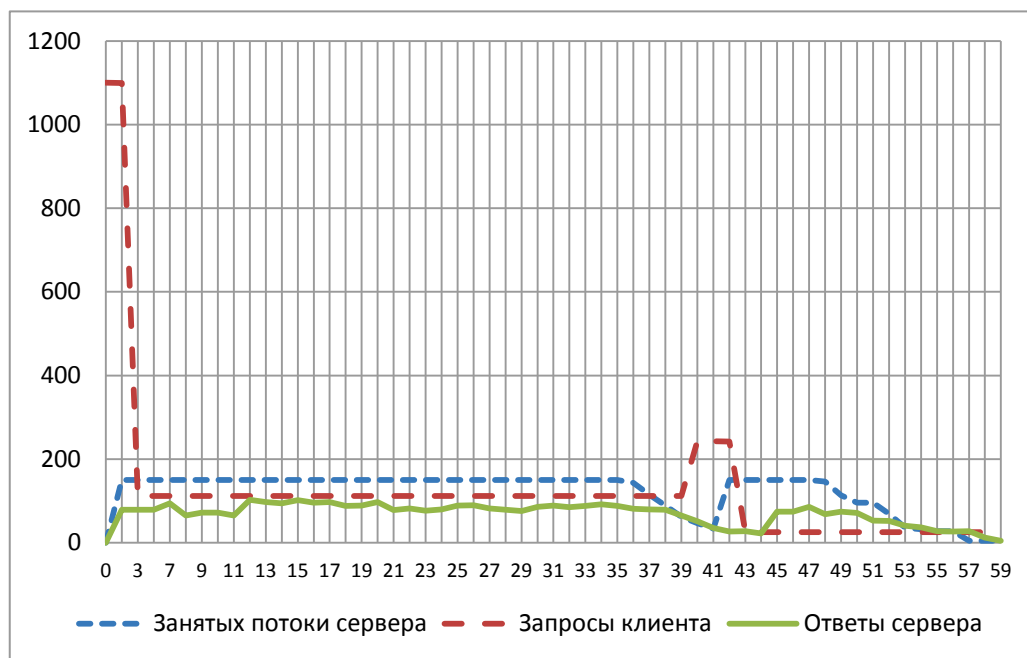


Рис. 1. Зависимость количества занятых потоков от количества запросов клиента и количества ответов сервера

### Обсуждение результатов

Можно сделать вывод текущие настройки не обеспечивают бесперебойную работу при пиковой нагрузке (600 пользователей), система восстанавливается после таких нагрузок в течение 5 секунд. В обычном режиме (400 пользователей) система работает, используя не более 30% выделенных серверных ресурсов.

Такие выводы после анализа результатов моделирования говорят о некоторых диспропорциях в конфигурировании распределенной системы и о неверных принятых решениях при проектировании системы. Так, например, вторую пиковую нагрузку можно избежать путем обновления не всего представления, а только тех графических элементов, данные для которых не были получены. Это сократит количество вторично отправляемых запросов до 90. 150 выделенных сервером потоков не справляются с нагрузкой, т.к. 90 пользователей получают не полную информацию. Самым простым решением является увеличение максимального времени ожидания ответа. Тем не менее, если в обычном режиме сервер используется всего на 30%. В качестве решения, можно уменьшить общее количество выделенных сервером потоков, а пиковую нагрузку на сервер уменьшить за счет подключения к серверу средств кэширования на уровне запросов и кэширования на уровне выполнения. Такой подход позволит миними-

зировать время выполнения сервером запросов, за счет дополнительных затрат на ресурсы для хранения кэшированных запросов. Все перечисленные нововведения можно опробовать на имеющейся РСП-модели, и варьируя различные параметры РСП-модели, добиться оптимальной работы системы с точки зрения производительности.

### **Выводы и заключение**

В качестве вывода следует отметить, что имитационное моделирование обеспечивает большую гибкость как при исследовании характеристик поведения существующей системы в различных режимах работы, так и при исследовании еще не реализованных решений, что позволяет выбрать наиболее выгодные по заданным критериям (качество, время, цена) решения еще до практической реализации. С помощью имитационного моделирования обеспечивается предсказуемость производительности системы. В течение всего жизненного цикла информация о системе постоянно уточняется, что позволяет получать более точные результаты моделирования и, соответственно, более точную информацию о последствиях принимаемых решений.

### **Список литературы**

1. Венгжин А., Короткевич А. Г. Использование раскрашенных сетей Петри для моделирования и верификации параллельных устройств логического управления // Радиоэлектроника и информатика. 2000. № 3. С. 70–74.
2. Дворянский Л. В., Ломазова И. А. Имитационное моделирование и верификация вложенных сетей Петри с использованием CPNTools // Моделирование и анализ информационных систем. 2012. № 5. С. 115–130.