

АРТЕМ ВЛАДИМИРОВИЧ ЖУКОВ

кандидат технических наук, старший преподаватель кафедры прикладной математики и кибернетики математического факультета, Петрозаводский государственный университет

zhukov_a_v@rbcmail.ru

ОЛЕГ ВАЛЕРЬЕВИЧ ГУСЕВ

преподаватель кафедры прикладной математики и кибернетики математического факультета, Петрозаводский государственный университет

eleset@gmail.com

АЛГОРИТМ ФИЛЬТРАЦИИ ЗАПРОСОВ К РЕСУРСОЕМКИМ WEB-СЕРВИСАМ НА ОСНОВЕ АЛГЕБРАИЧЕСКИХ СООТНОШЕНИЙ

В статье описаны проблемы обеспечения устойчивой работы web-сервисов в режимах предельных нагрузок, когда количество направляемых к серверу запросов превышает имеющиеся возможности по их обработке. Показано, что предложенные ранее методы управления запросами не всегда приемлемы ввиду использования моделей линейной оптимизации. Предложены алгоритмы, позволяющие в ряде случаев упростить задачу фильтрации за счет использования простых математических соотношений.

Ключевые слова: перегрузка, управление запросами, фильтрация трафика, сервер, компьютерные сети

ВВЕДЕНИЕ

Одним из важнейших требований к любой сетевой системе, ориентированной на предоставление информационных услуг широкому кругу пользователей, является обеспечение устойчивой бесперебойной работы, в том числе и в режимах нагрузки, близкой к предельным возможностям системы по количеству обрабатываемых запросов [2], [5]. Ситуации, когда система не справляется с обработкой поступающих запросов из-за недостатка имеющихся аппаратных ресурсов, будем называть перегрузкой. В частности, данная проблема актуальна для корпоративных информационных систем, построенных на базе интерактивных web-технологий, где функционал системы является отражением бизнес-функций предприятия.

Основные меры, направленные на предотвращение перегрузок, сводятся либо к наращиванию общей производительности системы [3], либо к управлению запросами, то есть к отклонению части из них на этапе поступления в систему в соответствии с некоторыми критериями. Надо отметить, что в силу специфики функционирования web-сервисов подходы, успешно применяемые при функционировании некоторых транспортных протоколов, например TCP [8], и основанные на обратной связи, позволяющей предотвратить перегрузку путем направления соответствующего сообщения отправляющему запросы клиенту, неприменимы, что вынуждает искать иные альтернативы.

© Жуков А. В., Гусев О. В., 2011

Поскольку только управление запросами позволяет предотвращать ситуации перегрузок даже при недостатке имеющихся вычислительных ресурсов, в рамках данной статьи сосредоточим внимание именно на этом методе.

В работах [4], [7] рассматривалось управление запросами к ресурсоемким web-сервисам (системам дистанционного обучения и сетевым системам моделирования). В частности, было предложено производить управление запросами путем отсечения части из них согласно долям, определенным соответствующими коэффициентами для каждого сетевого сервиса. Пересчет коэффициентов производится регулярно с периодичностью, позволяющей учесть изменения характеристик потока поступающих в систему запросов.

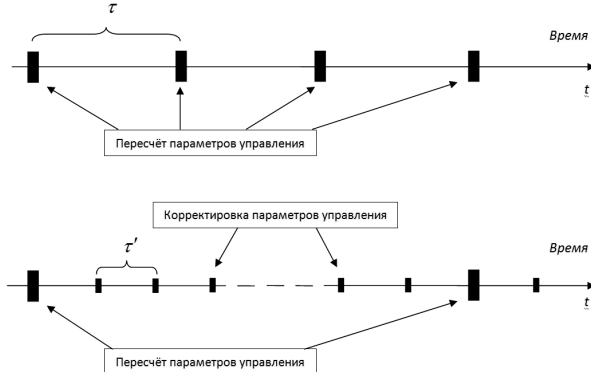
Несмотря на явные достоинства предложенных моделей управления запросами, они не лишены недостатков: при поиске параметров управления запросами необходимо всякий раз решать оптимизационную задачу, что при регулярном пересчете и большом количестве учитываемых факторов может стать ощутимой дополнительной нагрузкой на сервер и ограничивает как сферу применения предложенных моделей, так и возможности их развития.

Поэтому основными задачами исследования стали поиск возможностей для снижения вычислительной сложности рассмотренных схем управления запросами и повышение эффективности управления в ситуациях значительной флуктуации интенсивности поступающих запросов.

ИЗМЕНЕНИЕ ПАРАМЕТРОВ УПРАВЛЕНИЯ ЗАПРОСАМИ

В идеале параметры управления запросами должны в точности соответствовать характеристикам потока поступающих запросов, однако на практике такое соответствие поддерживать затруднительно: в общем случае предугадать поведение сетевого трафика крайне сложно, к тому же изменение параметров управления запросами требует учета большого количества влияющих факторов, что неизбежно сказывается на вычислительной сложности алгоритма.

В предложенных моделях в целях актуализации параметров управления их пересчет предлагается производить регулярно через интервал времени τ , выбираемый индивидуально для каждой системы (см. рисунок). Фактически параметры управления запросами изменяются дискретно, в то время как параметры потока запросов – непрерывно.



Моменты изменения параметров управления запросами

Учитывая, что пересчет является затратной с вычислительной точки зрения операций, один из возможных путей развития данной схемы заключается в том, чтобы взамен полного пересчета осуществлять корректировку параметров через меньший интервал времени τ' . Алгоритм корректировки должен позволять получать пусть и не строго оптимальный результат, но близкий к нему, при этом требуется гораздо меньше вычислительных ресурсов. Это позволит снизить нагрузку на сервер за счет увеличения временного интервала между полными пересчетами параметров управления без ущерба для их адекватности.

Достижение таких требований возможно, если алгоритм будет учитывать лишь наиболее значимые факторы при принятии решения об обслуживании.

УЧЕТ ТОЛЬКО ОДНОГО ВИДА РЕСУРСОВ

Возможности системы по обработке всех поступающих запросов определяются количеством имеющихся доступных вычислительных ресурсов. В информационных системах в роли таких ресурсов, как правило, выступают производи-

тельность процессора, доступный объем оперативной памяти, пропускная способность линий связи, доступный объем дисковой памяти и т. д.

На практике перегрузка обычно возникает вследствие недостатка ресурсов только одного-двух видов, в то время как остальные не востребованы полностью. Поэтому в ряде случаев можно говорить о наличии в системе наиболее востребованного ресурса, запасы которого будут истощены в первую очередь. При наличии в системе такого ресурса его достаточность фактически будет определять предел общей производительности системы, а ограничения на объем остальных можно пренебречь.

Пусть в системе имеется множество сетевых сервисов M ($I \in M$) и множество видов вычислительных ресурсов K ($j \in K$). Будем считать известными интенсивности потоков запросов к сервисам λ_i и потребности сервисов в ресурсах каждого вида S_i^j . Доступное количество имеющихся в системе ресурсов каждого вида ограничим соответствующим значением V^j .

Введем в рассмотрение степень востребованности j -го ресурса w^j , представляющую собой отношение необходимого количества данного ресурса для обслуживания всего потока запросов к его доступному количеству в системе.

$$w^j = \frac{\sum_{i \in M} \lambda_i \cdot S_i^j}{V^j}, \quad j \in K$$

Если $w^j \leq 1$, можно говорить о том, что данный ресурс не израсходуется полностью и не может стать причиной перегрузки, а потому не влияет на принятие решения об отклонении запросов. Если $w^j > 1$, доступного количества данного ресурса недостаточно для обслуживания всех поступающих запросов, что должно быть учтено при определении параметров управления.

Чем больше степень востребованности w^j , тем в большей мере ресурс j ограничивает возможности системы по обработке запросов. Таким образом, выбор наиболее востребованного ресурса может происходить по формуле

$$w^q = \max \{w^j \mid w^j > 1\}.$$

Поскольку потребность в ресурсах различных видов для работы разных сервисов различна, а управление запросами не всегда предполагает пропорциональное сокращение трафика ко всем сервисам, приведенная формула должна быть усилена в части выполнения ограничений по имеющимся объемам для ресурсов всех видов.

В наиболее общем случае достаточно, чтобы значение w^q наиболее востребованного ресурса значительно превышало значения потребности в ресурсах других видов.

$$\begin{cases} w^q = \max \{w^j \mid w^j > 1\} & j \in K \\ w^q \gg \max \{w^j\} & j \in K \setminus q \end{cases}$$

Таким образом, при соблюдении ряда условий число ограничений на количество доступных вычислительных ресурсов в модели может быть сокращено за счет исключения тех из них, которые не влияют на итоговый результат.

СПОСОБ КОРРЕКТИРОВКИ БЕЗ РЕШЕНИЯ ЗАДАЧИ ОПТИМИЗАЦИИ

Рассмотрим способ поиска параметров управления запросами, основанный на отказе от формулирования задачи математического программирования. Для предотвращения заторов при передаче дейтаграмм в IP-сетях успешно применяется метод взвешенной максиминной схемы равномерного распределения ресурсов [1]. Ниже представлена адаптация данного алгоритма применительно к управлению запросами к web-сервисам. Будем исходить из того, что в системе имеется лишь один ресурс, количество которого определяет ее возможности по обработке запросов.

Введем основные обозначения модели. Пусть в системе имеется множество сетевых сервисов M ($i \in M$). Будем считать известными интенсивности потоков запросов без управления к сервисам на предыдущем $\lambda = \{\lambda_1, \dots, \lambda_n\}$ и текущем $\lambda = \{\lambda_1, \dots, \lambda_n\}$ этапах определения (корректировки) параметров управления. Общий объем разделяемого ресурса ограничим значением V . Через S_i обозначим объем ресурса, необходимый для выполнения сервиса i .

Результатом работы алгоритма будем считать определение компонент вектора $\mu = \{\mu_1, \dots, \mu_n\}$ как максимально возможных интенсивности запросов к соответствующим сетевым сервисам, не вызывающим перегрузки. Зная $\mu = \{\mu_1, \dots, \mu_n\}$ и $\lambda = \{\lambda_1, \dots, \lambda_n\}$, легко найти коэффициенты, определяющие долю запросов, которые необходимо отклонить для предотвращения перегрузки.

Будем считать, что определенный для каждого сетевого сервиса приоритет P_i влияет на определение доли ресурса. Объем ресурса V_i , гарантированно доступный для i -го сервиса, может быть найден по формуле

$$V_i = V \frac{P_i}{\sum_{k \in M} P_k} = V \cdot P_i.$$

Дополнительно в модели делается допущение, что объемы имеющихся вычислительных ресурсов являются постоянными величинами. Данный подход не вполне соответствует реальным системам, где эти величины зависят от многих факторов [6]. К тому же при одновременном выполнении нескольких запросов общий требуемый объем ресурсов не всегда может быть представлен в виде алгебраических сумм потребностей каждого запроса в отдельности.

Из-за указанных допущений результат работы алгоритма не является строго оптималь-

ным, что особенно может проявиться в случаях, когда количество активных сервисов невелико, а характеристики потока запросов подвержены значительным флуктуациям. Это позволяет рассматривать предложенный способ как пригодный только для корректировки имеющихся параметров управления, а их расчет должен производиться по другим схемам.

Для начала рассмотрим систему, в которую поступают запросы только к одному сервису. Интенсивность потока с управлением μ либо совпадает с интенсивностью входного потока без управления λ , если имеющихся ресурсов достаточно для обслуживания всех поступающих запросов, либо ограничена имеющимися возможностями системы. Зависимость между λ и μ можно представить в виде

$$\lambda = \{\lambda_1\} \quad \mu = \{\mu_1\}$$

$$\mu_1 = \min \left\{ \frac{V}{S_1}, \lambda_1 \right\}.$$

Рассмотрим систему, в которой функционируют два сервиса. Интенсивность потока запросов с управлением к сервису в этом случае ограничена либо интенсивностью потока запросов без управления λ , либо гарантированно выделяемым этому сервису объемом ресурса V_i в совокупности с объемом ресурса, незадействованного вторым сервисом, если тот не использует свой лимит полностью.

$$\lambda = \{\lambda_1, \lambda_2\} \quad \mu = \{\mu_1, \mu_2\}$$

$$\mu_1 = \min \left\{ \lambda_1, \max \left\{ \frac{V_1}{S_1}, \frac{V - \min \{S_2 \lambda_2, V_2\}}{S_1} \right\} \right\},$$

$$\mu_2 = \min \left\{ \lambda_2, \max \left\{ \frac{V_2}{S_2}, \frac{V - \min \{S_1 \lambda_1, V_1\}}{S_2} \right\} \right\}.$$

Поскольку при работе 2 ресурсов $V = V_1 + V_2$, справедливо неравенство

$V - \min \{S_2 \lambda_2, V_2\} = V_1 + V_2 - \min \{S_2 \lambda_2, V_2\} \geq V_1$, а формулы могут быть приведены к виду

$$\mu_1 = \min \left\{ \lambda_1, \frac{V - \min \{S_2 \lambda_2, V_2\}}{S_1} \right\} = \min \left\{ \lambda_1, \frac{V - \min \{S_2 \lambda_2, V \cdot \frac{P_2}{P_1 + P_2}\}}{S_1} \right\}$$

$$\mu_2 = \min \left\{ \lambda_2, \frac{V - \min \{S_1 \lambda_1, V_1\}}{S_2} \right\} = \min \left\{ \lambda_2, \frac{V - \min \{S_1 \lambda_1, V \cdot \frac{P_1}{P_1 + P_2}\}}{S_2} \right\}.$$

Рассмотрим систему, в которой запросы поступают к нескольким сервисам. Для начала разделим все имеющиеся сервисы на два множества:

$$1) M^{ДОСТ} = \{i \mid S_i \cdot \lambda_i \leq V_i\} \quad i \in M, \quad (1)$$

если сервису достаточно предоставляемого гарантированного объема ресурсов V_i для обслуживания потока запросов λ_i в полном объеме ($\lambda_i = \mu_i$). В таком случае объем предоставляемых сервису ресурсов может быть сокращен до необходимого значения (произведения интенсивнос-

ти входящего потока λ_i и сложности выполнения этого одного запроса S_i).

2) $M^{НЕДОСТ} = \{i \mid S_i \cdot \lambda_i > V_i\}$ $i \in M$, если сервису недостаточно предоставляемого гарантированного объема ресурсов V_i для обслуживания потока запросов λ_i в полном объеме.

Учитывая, что сервисы из множества $M^{ДОСТ}$ не используют полностью предоставляемые им вычислительные ресурсы, образующиеся резервы могут быть перераспределены в пользу сервисов из множества $M^{НЕДОСТ}$. Найдем объем неустраиваемых сервисами из множества $M^{ДОСТ}$ резервов $V^{СВОБ}$ через разность между гарантированно предоставляемыми объемами ресурса и фактическим его использованием:

$$\begin{aligned} V^{СВОБ} &= \sum_{j \in M^{ДОСТ}} (V_j - S_j \cdot \lambda_j) = V - \sum_{j \in M^{ДОСТ}} S_j \cdot \lambda_j - \sum_{j \in M^{НЕДОСТ}} V_j = \\ &= V - \sum_{j \in M} \min\{S_j \cdot \lambda_j, V_j\} = V - \sum_{j \in M} \min\{S_j \cdot \lambda_j, V \cdot P_j\}. \end{aligned} \quad (2)$$

Полученный объем разделим между нуждающимися сервисами $i \in M^{НЕДОСТ}$ согласно принципам приоритетов. Тогда объем дополнительно выделяемых ресурсов для каждого конкретного сервиса $V_i^{ДОП}$ ($i \in M^{НЕДОСТ}$) может быть найден по формуле:

$$V_i^{ДОП} = V^{СВОБ} \cdot \frac{P_i}{\sum_{k \in M^{НЕДОСТ}} P_k} = \left(V - \sum_{j \in M} \min\{S_j \cdot \lambda_j, V \cdot P_j\} \right) \cdot \frac{P_i}{\sum_{k \in M^{НЕДОСТ}} P_k}$$

$i \in M^{НЕДОСТ}$.

В результате для сервисов из множества $M^{ДОСТ}$ интенсивность потока μ_i будет определяться интенсивностью потока λ_i , а для сервисов из множества $M^{НЕДОСТ}$ — отношением объема предоставляемых ресурсов (гарантированно предоставляемый объем V_i и дополнительно выделяемый из неиспользуемых резервов объем $V_i^{ДОП}$) к сложности выполнения запросов S_i . Итоговая формула примет вид:

$$\begin{aligned} \lambda &= \{\lambda_1, \lambda_2, \dots, \lambda_n\} \quad \mu = \{\mu_1, \mu_2, \dots, \mu_n\} \\ \mu_i &= \min\left\{\lambda_i, \frac{V_i + V_i^{ДОП}}{S_i}\right\} = \\ &= \min\left\{\lambda_i, \frac{V \cdot P_i + \left(V - \sum_{j \in M} \min\{S_j \cdot \lambda_j, V \cdot P_j\}\right) \cdot \frac{P_i}{\sum_{k \in M^{НЕДОСТ}} P_k}}{S_i}\right\}. \end{aligned} \quad (3)$$

Коэффициент, отражающий долю запросов, которые необходимо пропустить для выполнения к конкретному сервису, может быть легко найден через соотношение:

$$x_i = \begin{cases} \frac{\mu_i}{\lambda_i} & \lambda_i > 0 \\ 0 & \lambda_i = 0 \end{cases}.$$

Алгоритм управления запросами:

Шаг 1. Определение необходимости в управлении запросами

Оценивается разница $V^{НЕДОСТ}$ между общим количеством имеющегося ресурса и фактической потребностью в нем для обслуживания всех поступающих в систему запросов.

$$V^{НЕДОСТ} = \sum_{i \in M} \lambda_i \cdot S_i - V.$$

В зависимости от значения:

1) Если $V^{НЕДОСТ} \leq 0 \Rightarrow \mu_i = 1 \quad \forall i \in M$, име-

ющегося в системе объема вычислительного ресурса достаточно для выполнения всех поступающих запросов, управление запросами не требуется. Работа алгоритма прекращается.

2) Если $V^{НЕДОСТ} > 0$, имеющегося в системе объема вычислительного ресурса недостаточно для выполнения всех поступающих запросов, требуется фильтрация части запросов (переход к шагу 2).

Шаг 2. Определение необходимости в корректировке

Определяется необходимость корректировки существующих параметров управления запросами исходя из изменения характеристик потока запросов. Составляется перечень сервисов, интенсивность запросов к которым существенно не изменилась. Введем поправочный коэффициент $\omega > 0$ для устранения незначительных флуктуаций параметров трафика.

$$M^{НЕИЗМ} = \bigcup \{i \mid (1 - \omega) \cdot \bar{\lambda}_i < \lambda_i < (1 + \omega) \cdot \bar{\lambda}_i\} \quad i \in M.$$

Если $M^{НЕИЗМ} = M$, можно говорить о том, что характеристики потока запросов изменились незначительно, а существующие параметры управления запросами не нуждаются в корректировке. В остальных случаях необходима корректировка (переход к шагу 3).

Шаг 3. Корректировка параметров управления запросами

Производится определение параметров управления запросами в зависимости от сложности их выполнения, уровня приоритета и интенсивности поступления.

1) По формуле (1) определяется перечень сетевых сервисов, интенсивность запросов к которым не превышает выделенные для них квоты.

2) По формуле (2) определяются объемы дополнительно распределяемого ресурса для каждого сетевого сервиса.

3) По результатам полученных значений с использованием соотношения (3) на основе имеющейся информации об интенсивности запросов к каждому сервису определяется доля запросов, которые должны быть отклонены с целью предотвращения возникновения перегрузки.

Шаг 4. Фильтрация трафика

На основе полученных коэффициентов и длительности периода управления запросами определяется предельное число запросов для каждого сетевого сервиса. Запросы, превышающие установленные квоты, отклоняются.

ЗАКЛЮЧЕНИЕ

Предложенный алгоритм позволяет упростить процесс управления запросами по сравнению с ранее предлагаемыми моделями за счет учета лишь наиболее значимых факторов, а также дает возможность более оперативно реагировать на изменение характеристик потока поступающих запросов. Управление запросами может дать положительный эффект, однако результат во многом зависит от динамики изменения характеристик потока запросов и правильности выбора критериев управления.

На наш взгляд, развитие подобных алгоритмов должно следовать по пути уменьшения вычислительной сложности при обеспечении достаточной эффективности управления запросами. Вопросы применимости в реальных системах, оценки эффективности предложенных моделей являются темой дальнейших исследований и оставлены за рамками данной статьи.

В заключение стоит отметить, что управление запросами не является полноценным восполнением недостающих вычислительных способностей системы, хотя и позволяет системе с минимальными потерями преодолевать ситуации, когда возможность качественного оказания информационной услуги наиболее уязвима. Гарантированное обслуживание подразумевает наращивание вычислительных ресурсов сервера и/или уменьшение сложности выполнения запросов.

СПИСОК ЛИТЕРАТУРЫ

1. Вегешна Ш. Качество обслуживания в сетях IP: Пер. с англ. М.: Издательский дом «Вильямс», 2003. 368 с.
2. Гусев О. В., Поляков В. В., Жуков А. В., Поляков С. В. Проблема адекватной оценки производительности веб-серверов в корпоративных сетях на предприятиях ЦБП // Материалы 6-й науч.-техн. конф. «Новые информационные технологии в ЦБП и энергетике». Петрозаводск, 2004. С. 84–87.
3. Дудченко В. Управление производительностью прикладных систем // Открытые системы. СУБД. 2005. № 3. С. 50–53.
4. Жуков А. В., Аминова И. В., Поляков В. В. О некоторых механизмах предотвращения перегрузок на web-серверах. Петрозаводск, 2005. 26 с. Деп. в ВИНТИ 04.05.05, № 654-B2005.
5. Кучерявый Е. А. Управление трафиком и качеством обслуживания в сети Интернет. СПб.: Наука и техника, 2004.
6. Менаске Д., Алмейда В. А. Ф. Производительность Web-служб. Анализ, оценка и планирование: Пер. с англ. СПб.: ООО «ДиаСофтЮП», 2003. 480 с.
7. Поляков В. В. Стратегия обслуживания запросов к Web-сервисам в условиях ограниченных ресурсов вычислительной системы // Труды ПетрГУ. Сер. «Прикладная математика и информатика». Вып. 12. Петрозаводск, 2007.
8. Postel J. Transmission Control Protocol // RFC 793. 1981.