

ПРИНЦИПЫ АДАПТАЦИИ В МЕХАНИЗМАХ АКТИВНОГО УПРАВЛЕНИЯ ОЧЕРЕДЯМИ

Введение

В свете значительного увеличения количества и спектра предоставляемых услуг, а также стремительного роста скорости абонентского доступа к услугам глобальных сетей и Интернет остро стоит проблема перегруженности магистральных участков провайдеров услуг и внутренних сетей крупных предприятий [1]. Одним из первых принципов борьбы с перегрузками было создание буферов на телекоммуникационных узлах (коммутаторах и маршрутизаторах), которые позволяют сгладить кратковременные всплески нагрузки. Однако очереди больших размеров, которые загружены полностью большую часть времени, являются источниками увеличения значения средней задержки в сети. В то же время очереди малых размеров не позволяют полностью адаптироваться к всплескам нагрузки, что приводит к падению коэффициента использования полосы пропускания канала.

Таким образом, возникает необходимость иметь механизм, который поддерживал бы высокую пропускную способность и в то же время поддерживал средний размер очереди малым. Данная задача решается с помощью методов активного управления очередями, которые используются в дополнение к стандартным методам планирования и обслуживания программных очередей.

На сегодняшний день основным методом активного управления очередями, применяемым в сетях, является механизм DropTail, популярность которого обусловлена простотой реализации и минимальной нагрузкой на вычислительные ресурсы устройств. Суть работы механизма DropTail состоит в отбрасывании всех пакетов, которые приходят на постановку в переполненную программную очередь. Основным его недостатком, а также причиной разработки новых методов активного управления очередями, является эффект глобальной синхронизации (рис. 1, а), который наиболее ярко проявляется при использовании протокола Tahoe TCP [2, 3]. Кроме того, механизм DropTail не поддерживает приоритезацию трафика, что накладывает ограничения на его применение в рамках модели DiffServ.

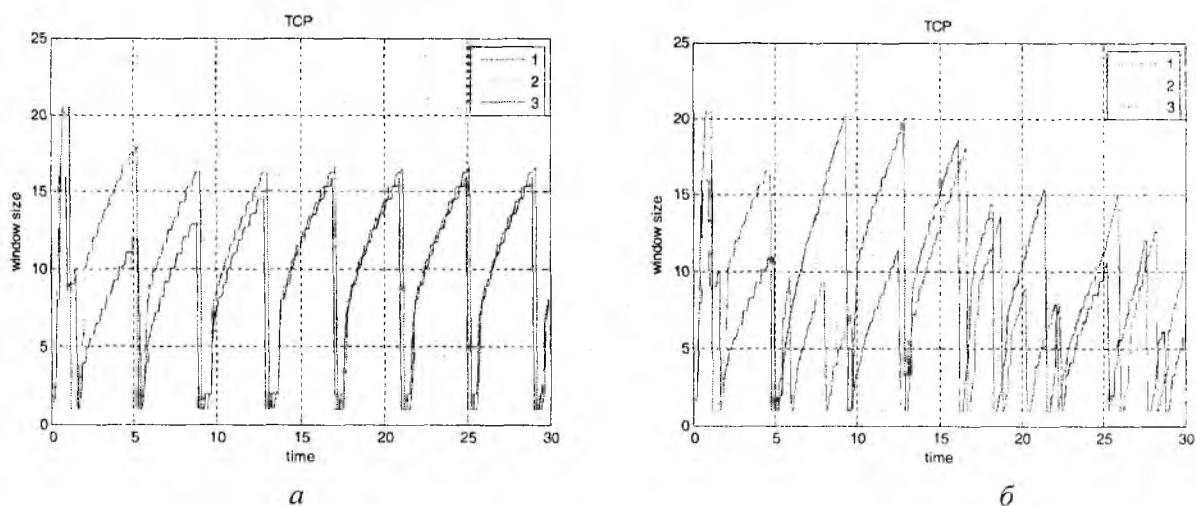


Рис. 1

Механизм раннего произвольного обнаружения

С целью устранить недостатки DropTail и повысить качество обслуживания в TCP-сетях был разработан эмпирический механизм произвольного раннего отбрасывания (Random Early Detect, RED). Механизм RED на основе мониторинга среднего размера каждой исходящей очереди реализует функцию случайного отбрасывания пакетов, за счет чего некоторые, произвольно выбранные TCP-соединения уведомляются о перегрузке. В ответ источники, чьи пакеты были отброшены, уменьшают размеры TCP-окна, снижая тем самым свою скорость передачи. Благодаря тому, что отбрасываются пакеты лишь некоторых соединений, удается избежать эффекта глобальной синхронизации (рис. 1, б).

Использование механизма RED в телекоммуникационных сетях решает следующие задачи:

- минимизацию дрожания задержки (джиттера) пакетов посредством контроля среднего размера очереди;
- предотвращение эффекта глобальной синхронизации TCP-трафика;
- обеспечение справедливого обслуживания трафика, который характеризуется кратковременными всплесками;
- строгое ограничение максимального среднего размера очереди.

Главный результат применения механизма RED состоит в минимизации среднего размера очереди, а следовательно, и результирующей задержки. Кроме того, вероятность P_{drop} , с которой соединение будет уведомлено о перегрузке, пропорциональна доле пропускной способности, используемой этим соединением на узле, что дополнительно решает еще одну пространственную проблему протокола TCP – TCP starvation; состоящую в «узурпации» пропускной способности канала одним или несколькими соединениями, которые быстрее адаптируются к индикации перегрузки либо вообще не реагируют на данного рода уведомления.

В рамках механизма RED вводится ряд параметров:

- Θ_{min} – минимальное граничное значение размера очереди, при превышении которого начинается процесс отбрасывания пакетов;
- Θ_{max} – максимальное граничное значение, при превышении которого отбрасываются все пакеты, пришедшие на постановку в очередь;
- β – экспоненциальный весовой коэффициент, который принимает участие в расчете среднего размера очереди q_{av} и определяет относительное влияние предыдущего среднего q_{old} и текущего размера очереди q_{cur} (мгновенный размер) в новом среднем размере очереди, который вычисляется по формуле

$$q_{av} = q_{old}(1 - 2^{-\beta}) + q_{cur}2^{-\beta}; \quad (1)$$

- δ – знаменатель граничной вероятности (mark probability denominator), определяет вероятность отбрасывания пакетов при достижении средним размером очереди максимального граничного значения Θ_{max} . В общем случае для $\Theta_{min} < q_{av} < \Theta_{max}$

$$P_{drop} = \frac{q_{av} - \Theta_{min}}{\Theta_{max} - \Theta_{min}} \cdot P_{max}, \quad (2)$$

где $P_{max} = 1/\delta$ – максимальная вероятность отбрасывания.

Все перечисленные параметры механизма RED в стандартном варианте [4] являются статическими и задаются администратором сети. При этом правила их выбора не формализованы и носят рекомендательный характер. Так, инженерами компании Cisco рекомендуются следующие значения: $\Theta_{min} = 5$ пакетов, $\Theta_{max} = (2...3)\Theta_{min}$, $\beta = 9$, $\delta = 10$.

При реализации алгоритма RED в зависимости от соотношения средней длины очереди q_{av} и введенных граничных значений возможны три ситуации (рис. 2). Если $q_{av} < \Theta_{min}$, пакеты, пришедшие на постановку в очередь, не отбрасываются, $P_{drop} = 0$. Если оценка среднего размера очереди находится посреди интервалов, $\Theta_{min} \leq q_{av} \leq \Theta_{max}$, вероятность отбрасывания P_{drop} прямо пропорциональна увеличению среднего размера очереди q_{av} от минимального Θ_{min} до максимально-го Θ_{max} граничных значений. P_{drop} изменяется от 0 до P_{max} .

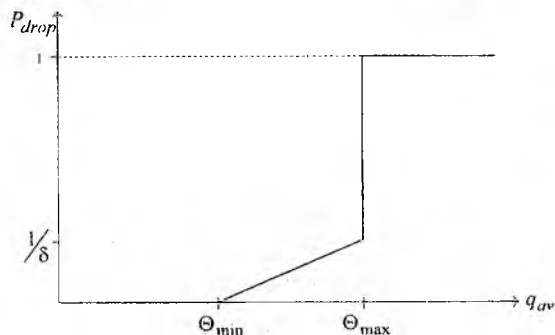


Рис. 2

В случае, когда средний размер очереди превышает верхнее пороговое значение $q_{av} > \Theta_{max}$, все прибывающие пакеты отбрасываются, $P_{drop} = 1$.

Адаптивный подход в механизмах активного управления очередями

Несмотря на очевидные преимущества стандартного алгоритма RED, было показано [5], что он обладает рядом недостатков, связанных в первую очередь со статичностью настраиваемых параметров, отвечающих за реакцию на перегрузку в сети. Так, если канал не загружен и/или значение параметра P_{max} велико, большую часть времени средний размер очереди будет приблизительно равен нижнему порогу Θ_{min} . Если же канал загружен сильно либо значение параметра P_{max} мало, средняя очередь будет приблизительно равна верхнему порогу Θ_{max} или даже превышать его. Как результат, средняя задержка пребывания пакетов в очереди, которая является одним из основных показателей качества обслуживания, зависит, во-первых, от загрузки канала, во-вторых, от правильности выбора администратором значений параметров, а потому слабо поддается предсказанию [6]. Тем не менее, операторы сетей, предоставляющие услуги гарантированного качества, должны иметь возможность априори оценить задержки в сети, что указывает на важность выбора параметров RED с учетом характеристик трактов передачи и передаваемого через них трафика. В идеале это должны быть текущие, а не усредненные характеристики, что определяет необходимость перехода от статически конфигурируемых параметров к динамически подстраиваемым.

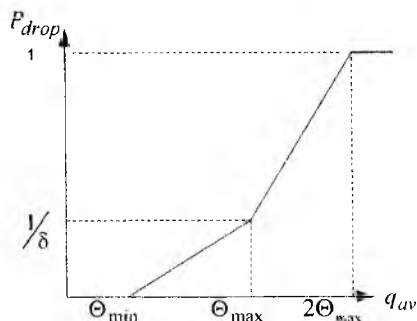


Рис. 3

Первой попыткой устранить выявленные в процессе эксплуатации недостатки стандартного RED было введение трех порогов вместо двух и добавление «плавного режима» (gentle mode) (рис. 3).

В настоящее время инженерами предложен ряд других, более эффективных решений, связанных с динамическим реконфигурированием параметров механизма отбрасывания. В основном [6–8] предложения по улучшению работы алгоритма стандартного RED базируются на сохранении его основных принципов функционирования, и состоят в том, чтобы путем внесения минимальных изменений производить автоматическую подстройку параметра P_{max} . Например, в работе [6] предполагается за счет динамического и адаптивного изменения параметра максимальной вероятности P_{max} добиться такой работы узла, при которой средний размер его очереди не просто принимает значения от Θ_{min} до Θ_{max} , но при этом еще и нахо-

дится в так называемой «целевой области» посередине Θ_{\min} и Θ_{\max} . С этой целью максимальная вероятность ограничивается диапазоном от 1 до 50 %, и изменяется фиксированными временными интервалами, превышающими среднее время обращения сегмента в сети (RTT).

Для изменения P_{\max} используется метод аддитивного увеличения мультипликативного уменьшения. Общий алгоритм работы адаптивного варианта механизма RED (Adaptive RED, ARED), предложенного в [6], можно коротко описать следующим образом (рис. 4).

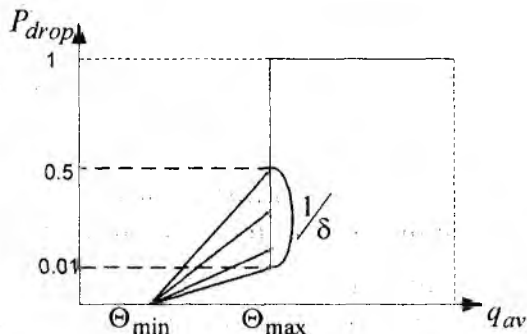


Рис. 4

Целью механизма ARED является удержание среднего размера очереди в интервале от $[\Theta_{\min} + 0,4 * (\Theta_{\max} - \Theta_{\min})]$ до $[\Theta_{\min} + 0,6 * (\Theta_{\max} - \Theta_{\min})]$. Тогда, если рассчитанный средний размер очереди q_{av} превышает свое целевое значение, и текущее значение максимальной вероятности отбрасывания $P_{\max} \leq 0,5$, то новое значение максимальной вероятности отбрасывания рассчитывается согласно

$$P_{\max} = P_{\max} + \alpha, \quad (3)$$

где α – аддитивный коэффициент увеличения вероятности отбрасывания, $\alpha = \min(0,01; 1/4P_{\max})$.

В ином случае, если рассчитанный средний размер очереди q_{av} меньше целевого значения и максимальная вероятность отбрасывания $P_{\max} \geq 0,01$, новое значение максимальной вероятности отбрасывания вычисляется как

$$P_{\max} = P_{\max} * \varepsilon, \quad (4)$$

где ε – мультипликативный параметр уменьшения вероятности отбрасывания, $\varepsilon = 0.9$.

Рекомендации относительно выбора весового коэффициента процедуры оценки β , общие для RED и ARED, $\beta = 1 - \exp(-1/C)$, где C – скорость канала передачи в пакетах в секунду.

Как видно из рис. 4, для определения вероятности отбрасывания пакета, алгоритм ARED использует набор линейно изменяющихся зависимостей $P_{\max}(q_{av})$. После пересчета средней длины очереди q_{av} и сравнения его с целевым значением вычисляется новая максимальная вероятность отбрасывания P_{\max} , которая по сути определяет наклон характеристики и конечную вероятность маркировки пакета. Стабильность ARED достигается путем медленных и нечастых изменений P_{\max} . Однако именно с этим связан главный недостаток алгоритма ARED, предложенного в [6]: при стремительных изменениях уровня загруженности сети медленные изменения P_{\max} не смогут предотвратить лавинообразное отбрасывание пакетов, т.е. допустят возникновение перегрузки сети.

Постановка задач адаптивного управления

С точки зрения теории оптимального управления механизм RED в стандартном его варианте [4] представляет собой практическую реализацию стохастического управления. Согласно общепринятому подходу [9] весь процесс решения подобного рода задач состоит из двух этапов. Первый связан со стохастическим оцениванием текущего состояния системы, а второй представляет собой детерминированную процедуру выработки управляющего воздействия. В терминологии RED состояние системы отражает величина среднего размера

очереди q_{av} , а ее вычисление (1) представляет собой процедуру стохастического оценивания случайной величины, известную как процедура Роббинса–Монро [10]. Дальнейшее принятие решения об отбрасывании по имеющемуся значению q_{av} носит чисто детерминированный характер (рис. 5). Однако, как было показано выше, проявившиеся недостатки стандартного



Рис. 5

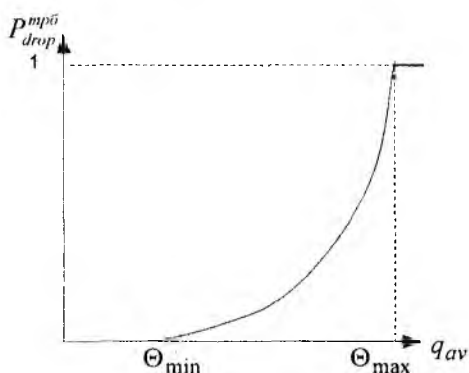


Рис. 6



Рис. 7

управления τ , – вектор подстраиваемых параметров.

Если оставить уравнение управления стандартного RED (2) без изменений, придерживаясь принципа минимума вносимых изменений в уже существующие разработки, т.е. $u = P_{drop}$, то в качестве подстраиваемых параметров могут выступать: максимальная вероятность отбрасывания P_{max} , минимальный Θ_{min} и максимальный Θ_{max} пороги, весовой коэффициент β . Тогда задача адаптации будет состоять в формировании векторов перечислен-

механизма RED, связанные со статичностью параметров в процессе функционирования сети, приводят к необходимости пересмотра самой постановки задачи.

Сформулируем задачу активного управления очередью следующим образом: путем регулирования количества отбрасываемых пакетов обеспечить максимально возможное значение полезной производительности сети G с точки зрения пользователя (Goodput) в условиях стохастически изменяющегося текущего размера очереди q_{cur} .

Как известно, в наибольшей степени значение производительности сети G снижается по причине лавинообразных отбрасываний пакетов на одном или нескольких узлах.

Для достижения максимальной производительности сети зависимость $P_{drop}^{mpb}(q_{av})$ на узлах сети должна носить нелинейный характер, при котором в области малых размеров очередей отбрасывание незначительно либо отсутствует, а при размере очереди, близком к перегрузке ($q_{av} \rightarrow \Theta_{max}$) вероятность отбрасывания увеличивается, но не скачкообразно (рис. 6).

Таким образом, цель управления может быть формализована как

$$|P_{drop}(q_{av}) - P_{drop}^{mpb}(q_{av})| \leq \Delta, \quad (5)$$

где Δ – допустимое отклонение, $\Delta \geq 0$.

Уравнение управления в общем виде выглядит следующим образом

$$u_i = U(q_{av_i}, \tau_i), \quad (6)$$

где u_i – управляющее воздействие для i -го цикла управления; $U(\bullet)$ – правила, согласно которым определяется текущее управление; q_{av_i} – оценка состояния сети в i -м цикле

ных подстраиваемых параметров, а задача управления остается без изменений и заключается в расчете числа пакетов, подлежащих отбрасыванию (рис. 7). Правила формирования подстраиваемых параметров в полной мере определяются зависимостью $P_{drop}^{nrb}(q_{av})$, конкретный вид которой представляет собой предмет дальнейших исследований.

Выводы

Широко распространенный в практике эксплуатации сетей механизм раннего произвольного отбрасывания пакетов на узлах сети, призванный поддерживать средний размер очереди в заданных рамках и противостоять эффекту глобальной синхронизации, обладает рядом недостатков.

Главным образом они обусловлены статическим характером параметров механизма, которые задаются администратором при настройке сети и в дальнейшем не изменяются.

В связи с этим возникает необходимость перехода от статических к динамически подстраиваемым параметрам, а с точки зрения теории оптимального управления приводит к пересмотру самой постановки задачи отбрасывания и трактовке ее как задачи адаптивного управления.

Список литературы: 1. Нехаенко А. Интернет: перезагрузка // <http://www.computerra.ru/vision/388550/>. 2008. 2. Jacobson V. Congestion Avoidance and Control // Proceedings of SIGCOMM '88. 1988. P.314-329. 3. Stevens W. RFC 2001 TCP Slow Start, Congestion Avoidance, Fast Retransmit. 4. Braden B., Clark D., Crowcroft J. and others RFC 2309 Recommendations on Queue Management and Congestion Avoidance in the Internet. 5. May M., Bolot J., Diot C., Lyles B. Reasons Not to Deploy RED // Proc. of 7th. International Workshop on Quality of Service (IWQoS'99). 1999. pp.260-262. 6. Floyd S., Gummadi R., Shenker S. Adaptive RED: An Algorithm for Increasing the Robustness of RED's Active Queue Management // AT&T Center for Internet Research at ICSI. 2001. 7. Feng W., Kandlur D., Saha D., Shin K. Techniques for Eliminating Packet Loss in Congested TCP/IP Network // Technical report U. Michigan CSE 349-97. 1997. 8. Feng W., Kandlur D., Saha D., Shin K. A Self-Configuring RED Gateway // Proceedings of Infocom. 1999. 9. Сейдж Э.П., Уайт Ч.С. III Оптимальное управление системами. М.: Радио и связь, 1982. 392 с. 10. Математичні основи теорії телекомунікаційних систем. / За заг. ред. Поповського В.В. Харків: ТОВ «Компанія СМІТ», 2006. 564 с.

Харьковский национальный
университет радиоэлектроники

Поступила в редколлегию 15.09.2009