УДК 004.042 + 519.713.2

В.В. Дорожинский, Г.Н. Жолткевич

Харьковский национальный университет имени В.Н. Каразина, Харьков

РЕГУЛЯРНАЯ ОБРАБОТКА СОБЫТИЙ В УПРАВЛЕНИИ ТЕЛЕКОММУНИКАЦИОННЫМИ СЕТЯМИ

В статье рассматриваются задачи управления трафиком в телекоммуникационных сетях. Существует множество подходов к управлению и оптимизации трафика. В частности, формирование трафика является одним из нескольких методов управления полосой пропускания в телекоммуникационных сетях. В статье показывается, что результаты полученные авторами в предыдущих работах могут широко использоваться при проектировании систем формирования и оптимизации трафика.

Ключевые слова: *телекоммуникационная сеть, машинное обучение, обработка событий, акцептор, регулярный язык, CEP-машина.*

1 Введение

Интенсивное развитие информационных технологий приводит к широкому распространению телекоммуникационных сетей. Развитие таких сетей обусловлено необходимостью передачи большого объема данных между информационными системами, что в свою очередь приводит к увеличению трафика данных передаваемых по сети. Эффективное управление трафиком является ключевым фактором в обеспечении высокой скорости передачи данных, а также их целостности. Термин формирование трафика (так же известный как формирование пакетов) – это техника управления трафиком в компьютерных сетях, которая заключается в задержке некоторых или всех датаграмм (datagrams) для приведения их в оптимальное состояние удовлетворяющее возможностям передающего канала [9,10]. Формирование трафика используется для оптимизации или гарантирования производительности, улучшения скорости передачи, или увеличения используемой полосы пропускания для некоторых типов пакетов путем задержки отправки пакетов других типов [1].

Часто техника формирования трафика используется вместе с техникой мониторинга трафика (Traffic policing), которая заключается в проверке соответствия передаваемых данных неким условиям (политикам). Данные не соответствующие условиям обычно удаляются или маркируются специальным образом [2]. Существует несколько основных подходов к формированию трафика:

— В общем случае, формирование трафика основывается на анализе приложений, которые его генерируют [4]. В случае формирования трафика генерируемого приложениями, вначале используются инструменты определения типов этих приложений (fingerprinting tools), а затем трафик формируется согласно заданным правилам для определенных типов. В некоторых случаях использование формирования трафика, генерируемого приложениями, является противоречивым. Например, изменение ширины полосы пропускания для P2P приложений. Многие прикладные протоколы используют шифрование, чтобы обойти процедуру формирования трафика, т.к. в таком случае невозможно определить тип данных.

— Другой тип формирования трафика основывается на анализе маршрута данных. Т.е. проводится анализ передающих каналов для определения оптимальной пропускной способности. Анализ формирования трафика в узле основанного на маршрутизации обычно проводится с использованием информации о загруженности и пропускной способности предыдущего или следующего канала связи, через которые проходит данный трафик [16]. Формирование пакетов обычно применяется в передающих каналах сети для контроля трафика входящего в сеть, но так же может применяться источниками трафика (например, компьютером или сетевой картой [13]) или любым другим элементом сети.

— Другой способ формирования трафика это ограничение приложениями генерируемого ими трафика. Такие приложения генерируют трафик который никогда не превосходит некоторое максимальное значение. Например, медиа приложения, генерирующие аудио и видео трафик, не могут передавать по сети больше данных в единицу времени, чем позволяет скорость их кодирования [8].

Утилиты формирования трафика работают по принципу задержки измеряемого трафика так, чтобы каждый пакет удовлетворял соответствующим ограничениям налагаемым на трафик. Измерение трафика может быть реализовано с помощью, например, алгоритмов текущего ведра [14] или ведра токенов [15]. При использовании данных алгоритмов передача данных может произойти:

— немедленно, если условия на передачу такого трафика выполняются на момент его поступления;

— после некоторой задержки, данные ожидают в очереди согласно заданного расписания отправки;

— никогда, в случае переполнения очереди.

Все реализации утилит формирования трафика имеют ограниченный объем памяти для входящих данных и должны уметь обрабатывать событие переполнения очереди. Простым и стандартным подходом является удаление входящего трафика в момент переполнения очереди (удаляется хвост очереди). Более сложные реализации могут применять специальные алгоритмы удаления избыточного трафика такие как алгоритм произвольного раннего обнаружения (Random Early Discard, RED) [6]. Существует несколько модификаций алгоритма произвольного раннего обнаружения. Каждая модификация имеет свои достоинства и недостатки [3,7,11].

Другим элементарным способом обработки переполнения является пересылка не сформированного трафика для которого не хватило места в очереди. Очевидно, что такой подход приведет к уменьшению скорости передачи приоритетных данных и это может повлиять на качество обслуживания. С другой стороны, такой подход уменьшает количество повторной передачи данных, что способствует уменьшению пиковых нагрузок. Следует отметить также, что результаты исследования, представленные в [12] показывают, что в интернете увеличивается доля P2P трафика.

Таким образом, решение задач анализа и классификации трафика является одним из ключевых условий улучшения скорости передачи данных, что в свою очередь приводит к улучшению качества обслуживания.

2 Базовые понятия и обозначения

Приведем необходимые понятия и обозначения, используемые в статье. Определим алфавит X как множество символов. В данном контексте каждый символ $x \in X$ будем интерпретировать как атомарный пакет данных, передаваемый по сети. Например, такой пакет может представлять собой фрагмент аудио, видео либо любых других данных. Таким образом, можно определить множество X как объединение:

$$X = \bigcup_{i=1}^{n} \left\{ A^i \bigcup V^i \bigcup D^i \bigcup M^i \right\},$$

где A^i, V^i – множества типов аудио и видео пакетов, генерируемых i-м пользователем сети; D^i - множество пакетов других данных, например, таких как сообщения электронной почты или веб-контент; M^i - множество служебных сообщений. Такие сообщения могут содержать диагностическую информацию о состоянии узла, либо управляющие команды другим узлам сети.

Замечание 1. Заметим, что количество типов данных может быть выбрано произвольно и зависит от требований к качеству передачи для тех или других данных. Так, например, тип пакетов D^i может в свою очередь быть представлен как объединение множеств пакетов представляющих полезный трафик и нежелательный (спам). В нашем случае в дальнейшем в качестве примера будет рассматриваться некоторая сеть мобильной связи в которой качество передачи видео и аудио данных играет ключевую роль, в то время как скорость передачи других данных может ограничиваться с целью максимального улучшения передачи изображения и звука. Другим критерием качества обслуживания может быть зависимость от типа тарифного плана клиента.

Также заметим, что в данном контексте, пользователем может быть как некоторый узел сети (базовая станция), так и некоторое приложение на мобильном устройстве генерирующее пакеты.

Обозначим как X^+ - множество конечных, непустых последовательностей пакетов. ε - пустая последовательность. Тогда $X^* = X^+ \bigcup \varepsilon$. Далее, обозначим как X^ω множество бесконечных потоков пакетов данных. Тогда весь трафик можно обозначить как $X^\infty = X^* \bigcup X^\omega$. Пусть $f : X \xrightarrow{\text{partial}} Y$, обозначает что f является частичным отображением из X в Y. Тогда запись $f(x)\uparrow$ обозначает что отображение $f(x)$ не определено для элемента x из множества X. А запись $f(x)\downarrow$ обозначает что отображение $f(x)$ определено для элемента x из множества X. Следовательно, запись $f(x)\downarrow = y$ обозначает, что $f(x)\downarrow$ и $y = f(x)$ для элемента y из множества Y. Далее, запись $|x|$ обозначает длину конечной последовательности x. А запись $x[0]$ обозначает первый элемент конечной или бесконечной последовательности x. Тогда, запись $x[1:]$ обозначает последовательность, получаемую путем удаления первого элемента последовательности x.

3 Общий вид системы управления трафиком

В данной части приводится общий вид системы анализа, оптимизации и управления трафиком в узле телекоммуникационной сети. В частности, таким узлом сети может быть любой экземпляр виртуальной сетевой функции (VNF). Например, виртуальный сетевой коммутатор.

В общем случае система управления трафиком может состоять из таких компонентов:

1. System Traffic Monitor Plane: отслеживает входящий трафик на предмет важных изменений (событий) в его содержимом, объем и другие возможные аномалии.

2. System Control Plane: компонент принятия решений. Основывается на событиях обнаруженных System Traffic Monitor Plane модулем, устанавливает методы мониторинга, а так же управления трафиком.

3. System Traffic Management Plane: управляет трафиком согласно правилам определенным в System Control Plane.

Схема взаимодействия компонентов системы представлена на рис. 1.

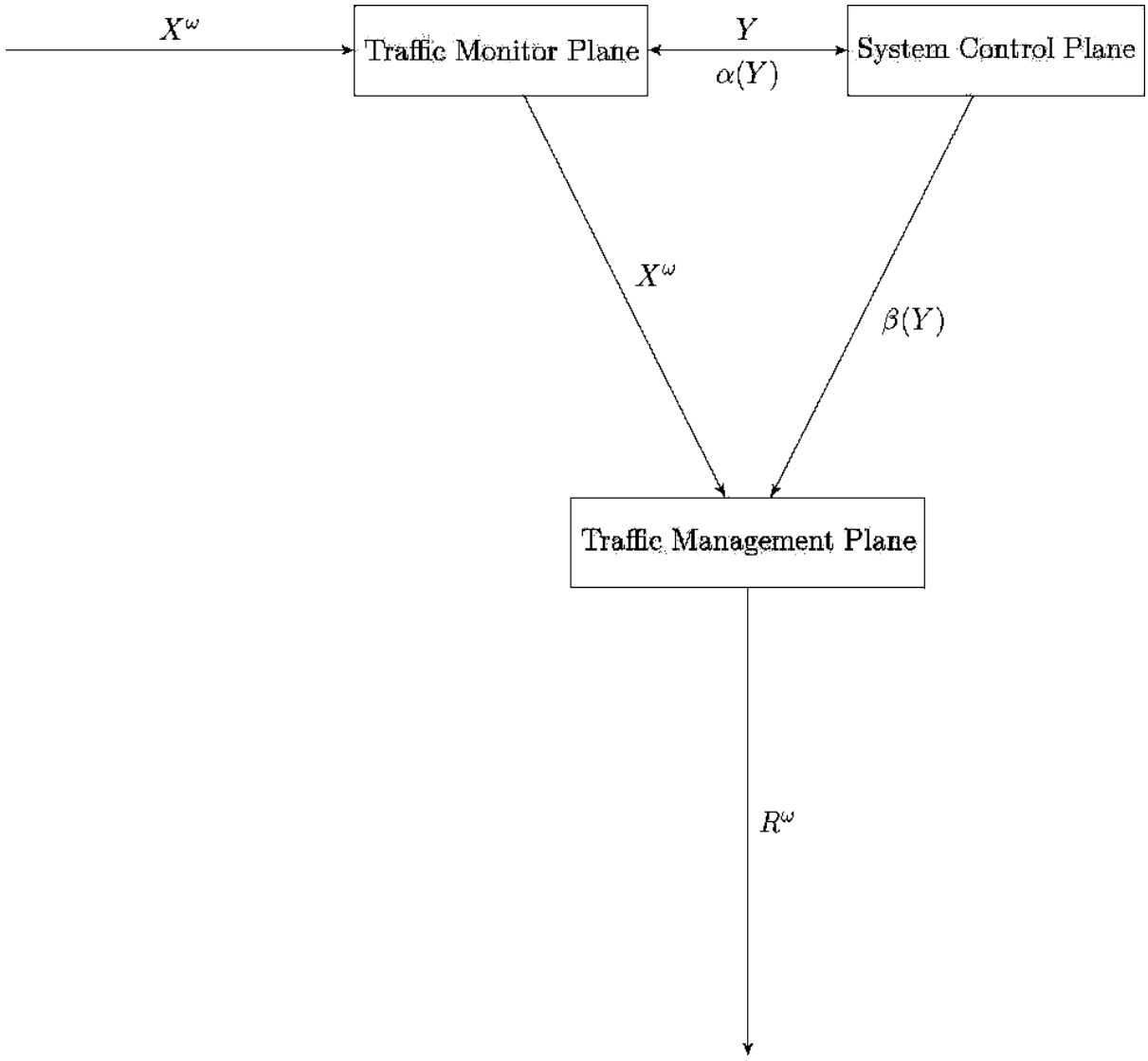

Рис. 1. Схема взаимодействия компонентов системы управления трафиком

На приведенной схеме X^ω обозначает входящий в узел сети трафик, тогда как R^ω - выходящий соответственно. Вначале входящий трафик X^ω анализируется System Traffic Monitor Plane компонентом на предмет возможных событий либо аномалий. Затем System Traffic Monitor Plane модуль перенаправляет данный трафик в System Traffic Management Plane, где с помощью заданных System Control Plane модулем правил он оптимизируется и формируется выходящий трафик R^ω. Если компонент System Traffic Monitor Plane обнаруживает некоторое событие или аномалию во входящем трафике X^ω, то сообщение об этом $y \in Y$ отправляется модулю System Control Plane. Далее, System Control Plane модуль обрабатывает возникшее событие и отправляет управляющие сигналы $\alpha(y)$ и $\beta(y)$, $y \in Y$ компонентам System Traffic Monitor Plane и System Traffic Management Plane соответственно. Управление может заключаться в изменении метода мониторинга входящего трафика X^ω, либо изменении метода оптимизации и формирования выходящего трафика R^ω. Например, может быть запущен механизм масштабирования, увеличивающий объем

доступных виртуальных вычислительных ресурсов, либо активизирующий дополнительные экземпляры виртуальных сетевых функций.

Замечание 2. Отметим, что на физическом уровне входящий трафик X^{ω} может приходить по нескольким каналам связи. Аналогичное утверждение также верно для выходящего трафика R^{ω}. Более того, компонент System Traffic Management Plane должен управлять распределением выходящего трафика по доступным выходным каналам.

4 Взаимосвязь системы управления трафиком и регулярной CEP-машины

Заметим, что каждый приходящий по сети пакет данных можно трактовать как атомарное событие. Как правило, каждое атомарное событие такого вида несет в себе слишком мало информации необходимой для анализа состояния всей сети. Следовательно, необходимо рассматривать последовательности атомарных событий на предмет обнаружения аномалий - необходимости обработки сложных событий (Complex Event Processing). Также заметим, что как правило, последовательности пакетов данных можно описывать регулярными языками. Следовательно, можно использовать регулярные CEP-машины, впервые представленные в работе [5], для обнаружения аномалий в последовательностях пакетов данных приходящих по сети. Напомним, что регулярная CEP-машина это:

Определение 1. (Регулярная машина обработки сложных событий). Любая регулярная CEP-машина это пятерка $M=(X,Y,H,h_0,\alpha)$ состоящая из таких компонентов:

— X - конечное множество (алфавит) атомарных событий;

— Y - конечное множество (алфавит) ответов машины соответственно;

— H - конечное множество регулярных обработчиков. Где любой обработчик $h\in H$ такой, что $h: X^{+} \xrightarrow{\text{partional}} Y$ называется регулярным, если существуют

• некоторое конечное множество Z с отмеченным элементом $z_0\in Z$ и

• некоторое отображение $\delta: Z\times X\to Z\bigcup Y$

такие, что для любого $u\in X^{+}$ и $y\in Y$ условие $h(u)\downarrow = y$ выполняется тогда и только тогда, когда существуют $z_1,...,z_{|u|-1}\in Z$ такие что:

$z_{i+1}=\delta(z_i,u[i])$ для $0\le i\angle|u|-1$ и

$y=\delta\left(z_{|u|-1},u\left[|u|-1\right]\right)$.

— $h_0\in H$ - начальный обработчик;

— $\alpha: Y\to H$ - функция отклика.

Алгоритм 1 описывает поведение любой регулярной CEP-машины. Заметим, что в общем случае, шаг 5 алгоритма может выполняться бесконечно см. работу [17]. Но в случае регулярных CEP-машин алгоритм гарантировано не зависнет на этом шаге.

Алгоритм 1. Operational model of a Regular CEP-machine:

1 def run (M,s):

Require: the studied Regular CEP-machine $M=(X,Y,H,h_0,\alpha)$ and some stream of elementary events $s\in X^{\omega}$

Ensure: printing of the corresponding response stream

2 $\quad$ handler, buff $=h_0,[\ ]$

3 $\quad$ while True:

4 $\quad\quad$ new_event, s $=s[0],s[1:]$

5 $\quad\quad$ buff.append(new_event)

6 $\quad\quad$ if handler(buff)$\uparrow$: continue

7 $\quad\quad$ else:

8 $\quad\quad\quad$ response $=$ handler(buff)

9 $\quad\quad\quad$ print(response) $\quad$ # printing of the current response

10 $\quad\quad\quad$ handler, buff $=\alpha$(response), $[\ ]$

Пусть задана регулярная CEP-машина определяемая пятеркой $M=(X,Y,H,h_0,\alpha)$. В данном контексте множество X является входным алфавитом пакетов, тогда как множество Y - выходной алфавит событий генерируемых как реакция на обнаруженные события или аномалии во входном трафике. Тогда любой регулярный обработчик (частичное отображение) $h\in H$ такой что $h: X^{+} \xrightarrow{\text{partial}} Y$ можно интерпретировать как алгоритм анализа входного трафика предназначенный для обнаружения определенного множества аномалий. А функция отклика $\alpha: Y\to H$ определяет новый алгоритм анализа входного трафика, который должен использоваться в случае обнаружения той или иной аномалии на предыдущем шаге. Например, при обнаружении во входном трафике множества подозрительных пакетов, которые могут содержать некую вредоносную информацию, может быть активирован алгоритм определения генератора такой информации и его блокирования. Другим примером может являться обнаружение последовательности пакетов содержащих информацию о статусе других узлов сети на основании которой данный узел может определить загруженность всей сети и активировать соответствующий метод формирования пользовательского трафика. Таким образом, можно оптимизировать управление балансом загруженности всей сети в автоматическом режиме.

5 Проектирование компонента анализа входного трафика с использованием машинного обучения

Решение задачи анализа входящего трафика является одним из ключевых пунктов при проектировании системы управления трафиком. Для практического решения данной задачи будем использовать метод машинного обучения регулярной CEP-машины. Результаты компьютерного эксперимента подтверждают сходимость этого метода машинного обучения. Приведем ниже идею метода.

Задачу обучения регулярной CEP-машины можно разделить на независимые подзадачи обучения ее обработчиков.

Поскольку каждый регулярный обработчик имеет только один возможный ответ "accepted", ниже будем называть такой обработчик регулярным акцептором.

Тогда задача обучения регулярного акцептора может быть сформулирована следующим образом.

Задача 1. Пусть $E = \{u_1, ..., u_M\} \subset X^+$ это конечное префиксное множество событий, а $C = \{v_1, ..., v_N\} \subset X^+$ - конечное множество слов такие, что $E \bigcap C = \varnothing$.

Тогда будем интерпретировать элементы множества E как примеры, а элементы множества C как контрпримеры;

необходимо найти регулярный акцептор $h : X^+ \xrightarrow{\text{partial}} \{\text{accepted}\}$ такой, что выполняются следующие условия:

1. $h(u_i)\downarrow = \text{accepted}$ для всех $0 \le i < M$;
2. $h(v_i)\uparrow$ для всех $0 \le i < N$; и
3. Регулярный акцептор является минимальным. А именно, соответствующее множество Z имеет наименьшее количество элементов среди всех возможных.

Общий вид метода построения акцептора представлен на рис. 2 в виде UML диаграммы активности.

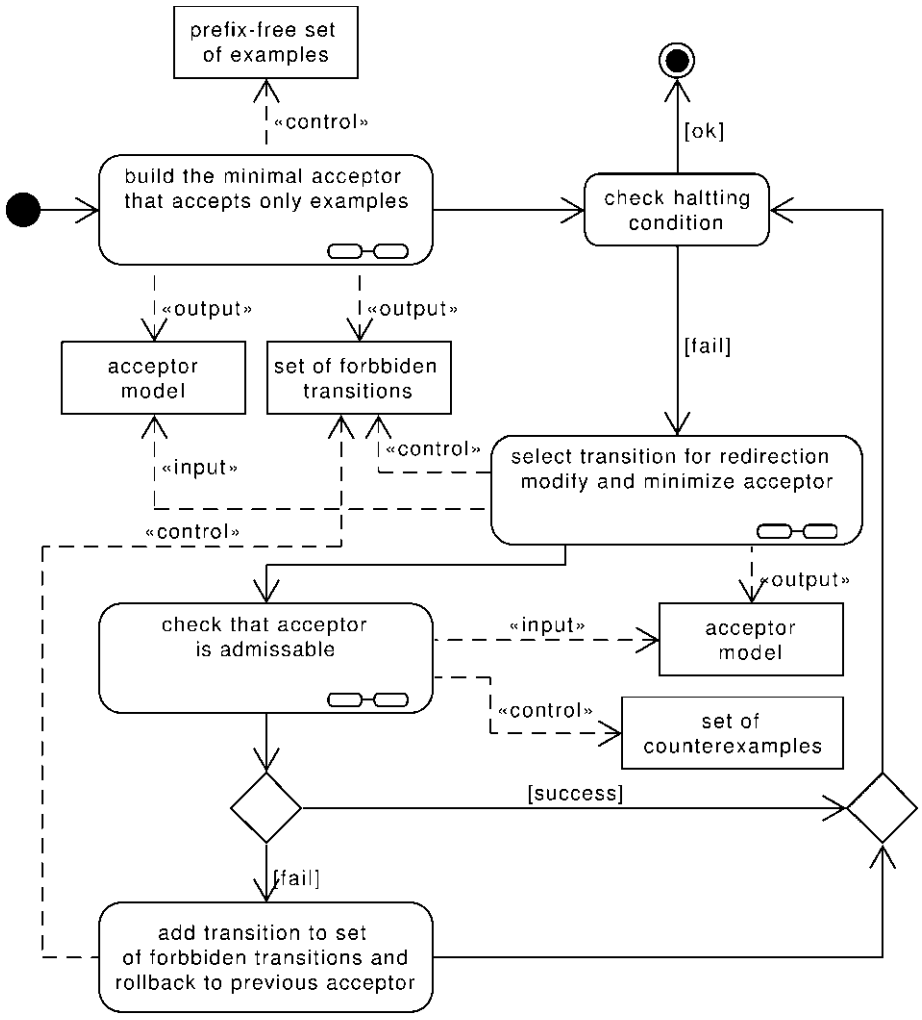

Рис. 2. Общая схема метода обучения регулярного акцептора

Теперь воспользуемся описанным выше методом машинного обучения.

Для этого определим достаточно большое множество примеров событий E - некий набор хорошо известных проблем трафика которые могут привести к сбою или перегрузке сети. А также определим множество контрпримеров C - набор сценариев правильной работы сети

Замечание 3. Множество контрпримеров должно быть максимальным для уменьшения проблемы ложного обнаружения аномалий в трафике.

Для иллюстрации метода приведем пример определения множеств E и C.

Пример 1. Рассмотрим телекоммуникационную сеть состоящую из n узлов (базовых станций), возможно виртуальных.

И пусть множество пакетов характеризуется следующим образом:

$$X = \bigcup_{i=1}^{n} \left\{ A^i \bigcup V^i \bigcup D^i \bigcup M^i \right\},$$

где A^i, V^i, D^i множества аудио, видео и других вебданных сгенерированных базовыми станциями соответственно. Тогда как M^i - множество служебных пакетов, которые могут содержать диагностическую информацию о состоянии станции, уровне загруженности, планируемых действиях (выключение или перезапуск системы, появление в сети нового узла и т.д.).

Тогда начальное множество отслеживаемых событий E в общем случае можно задать следующим образом:

$$E = \left\{ \left(a^* v^* d^* \right)^* m^+ \left(a^* v^* d^* \right)^* \middle| \begin{matrix} \forall a \in A^i, v \in V^i, \\ d \in D^i, m \in M^i \end{matrix} \right\}.$$

Т.е. системе необходимо отслеживать все слова содержащие в себе служебные пакеты и реагировать на те или иные события. Тогда множество контрпримеров можно задать как:

$$C = \left\{ \left(a^* v^* d^* \right)^* \mid \ \forall a \in A^i, v \in V^i, d \in D^i \right\},$$

где $i = 1...n, *$ – звездочка Клини, а $x^+ = x^* \setminus \{\varepsilon\}$, т.е. система не меняет режим работы при прохождении пользовательского трафика.

Другим примером может быть обнаружение ситуации, когда возникает необходимость ограничения трафика генерируемого пользователями, имеющими более низкий приоритет обслуживания. А так же улучшение пропускной способности сети для видео трафика.

Пример 2. Пусть пользователь k имеет более высокий приоритет передачи данных. И так же необходимо обеспечить высокую пропускную способность для видео трафика. Тогда система должна реагировать на ситуации когда входной трафик содержит слишком много пакетов с более низким приоритетом. Следовательно, множество примеров E можно задать следующим образом:

$$E = \left\{ x_i^+ \left(\left(v^k \right)^* \left(a^k \right)^* \left(d^k \right)^* \right)^+ x_j^+ \middle| \forall x_i, x_j \in X^{/}, \right.$$

$$\left. v^k \in V^k, a^k \in A^k, d^k \in D^k \right\} \bigcup$$

$$\bigcup \left\{ x_l^+ v^+ x_m^+ \middle| \forall x_l, x_m \in X^{//}, v \in V \right\},$$

где $X^{/} = X \setminus \left(A^k \bigcup V^k \bigcup D^k \right)$;

$$X^{//} = X \setminus V.$$

Следовательно, система не должна реагировать на ситуации когда в трафике отсутствуют пакеты узла k. Т.е. узел k не генерирует трафик. Либо в сети преобладает видео трафик или трафик генерируемый узлом k.

Тогда множество контрпримеров C можно определить следующим образом:

$$C = \left\{ x^* \middle| \forall x \in X^{/} \right\} \bigcup \left\{ \left(x^k \right)^* \middle| \forall x^k \in X^k \right\} \bigcup \left\{ v^* \middle| \forall v \in V \right\},$$

где $X^k = A^k \bigcup V^k \bigcup D^k$.

Заметим, что в случае ложного обнаружения, обнаруженное событие можно добавить во множество контрпримеров C для исключения ситуации обнаружения такого события в будущем.

Выводы

В статье рассмотрены способы использования регулярных CEP-машин при проектировании систем оптимизации трафика в телекоммуникационных сетях.

Такой подход основывался на строгом математическом анализе задач формирования и оптимизации трафика. Были рассмотрены основные проблемы, возникающие при решении таких задач.

Далее, в статье была представлена общая схема системы формирования и оптимизации трафика. Детально обсуждалась функциональность компонентов такой системы.

Затем обсуждался подход к решению задачи формирования и оптимизации трафика с помощью регулярных CEP-машин. Было показано, что метод машинного обучения, детально описанный авторами в их предыдущих работах, значительно упрощает процесс анализа возникающих в телекоммуникационных сетях аномалий.

В частности, примеры 1 и 2 приводят способ раннего обнаружения возможности перегрузки сети, что помогает в ее автоматической балансировке и тем самым улучшает производительность, посколь-

ку, значительно уменьшается вероятность потери пакетов и их повторной передачи, что в свою очередь является ключевым фактором при обеспечении качества передачи мультимедиа данных.

Также пример 2 показывает возможность управления отдельными потоками данных имеющих более высокий приоритет.

Результаты, полученные в данной работе, могут быть полезны при проектировании и разработке систем формирования и оптимизации трафика данных в телекоммуникационных сетях.

Список литературы

1. ATM Forum Traffic Management Specification, Version 4.0 Approved Specification 0056.00, Section 5.5, Traffic Shaping.
https://www.broadband-forum.org/ftp/ pub/approved-specs/af-tm-0056.000.pdf.

2. Cisco Tech Notes: Comparing Traffic Policing and Traffic Shaping for Bandwidth Limiting. Document ID: 19645. Cisco Systems. Aug 10, (2005)
http://www.cisco.com/c/en/us/support/docs/quality-of-service-qos/ qos-policing/19645-policevsshape.html.

3. Clark, D., D., Wroclawski, J.: An Approach to Service Allocation in the Internet. IETF. p. 12. (July 1997)
https://tools.ietf.org/html/ draft-clark-diff-svc-alloc-00.

4. Dischinger, M., Mislove, A., Haeberlen, A., Gummadi, K.: Detecting bittorrent blocking. IMC '08 Proceedings of the 8th ACM SIGCOMM conference on Internet measurement ISBN: 978-1-60558-334-1 doi>10.1145/1452520.1452523 ACM New York, NY, USA pp 3-8, (2008).

5. Dorozhinsky, V.: Regular Complex Event Processing Machines. Information processing systems. 8, pp. 82-86 (2015).

6. Floyd, S., Jacobson, V.: "Random Early Detection (RED) gateways for Congestion Avoidance". IEEE/ACM Transactions on Networking. 1 (4): pp 397-413. doi:10.1109/90.251892. (August 1993).

7. Gettys, J.: "RED in a Different Light". jg's Ramblings. (17-12-2010) https:// gettys.wordpress.com/2010/12/17/red-in-a-different-light/.

8. Helzer, J., Lisong, Xu: Congestion Control for Multimedia Streaming with Self- Limiting Sources. http : www. ieee- icnp. org/2005/Posters/Helzer. pdf.

9. IETF RFC 2475 "An Architecture for Differentiated Services"section 2.3.3.3 - Internet standard definition of "Shaper"https : //tools. ietf. org/html/rfc2475# section-2.3.3.3

10. ITU-T, Traffic control and congestion control in B ISDN, Recommendation I.371, International Telecommunication Union, Annex A, page 87. (2004)
http://www. itu.int/rec/T-REC-I.371-200403-I/en.

11. Jacobson, V., Nichols, K., Poduri, K.: "RED in a Different Light". (30-09-1999)
http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.22.9406.

12. Leydon, J.: "P2P swamps broadband networks" . http://www.theregister.co. uk/2002/09/12/p2p_swamps_broadband_networks/.

13. Pratt, I., Fraser, K.: a user-accessible gigabit Ethernet interface. IEEE INFOCOM 2001. Arsenic: Computer Laboratory, Cambridge University; Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings Volume 1, pp 67-76 vol.1 (2001)

14. Tanenbaum, A., S.: Computer Networks, Fourth Edition, ISBN 0-13-166836-6, Prentice Hall PTR. (2003)

15. Turner, J.: New directions in communications (or which way to the information age?). Communications Magazine, IEEE 24 (10): 8-15. ISSN 0163-6804. (1986)

16. Ying Zhang, Z. Morley Mao, Ming Zhang Ascertaining the Reality of Network Neutrality Violation in Backbone ISPs HotNets, Association for Computing Machinery, Inc. January 1, (2008) https://www.microsoft.com/en-us/research/publication/ ascertaining-the-reality-of-network-neutrality-violation-in-backbone-isps/

17. Zholtkevych, G., Novikov, B., Dorozhinsky, V.: Pre-Automata and Complex Event Processing. In: V. Ermolayev et al. (eds.) ICTERI 2014. CCIS, vol. 469, pp. 100116. Springer International Publishing, Switzerland (2014).

Надійшла до редколегії 17.06.2015

Рецензент: д-р техн. наук, проф. Г.А. Кучук, Харківський університет Повітряних Сил ім. Кожедуба, Харків.

РЕГУЛЯРНА ОБРОБКА ПОДІЙ В УПРАВЛІННІ ТЕЛЕКОМУНІКАЦІЙНИМИ МЕРЕЖАМИ

Г.М. Жолткевич, В.В. Дорожинський

В статті розглядаються задачі управління трафіком в телекомунікаційних мережах. Існує багато підходів до управління та оптимізації трафіка. Зокрема, формування трафіка є одним із декількох методів управління смугою пропуску у телекомунікаційних мережах. У статті показується, що результати отримані авторами у попередніх працях можуть бути широко використані підчас проектування систем формування та оптимізації трафіка.

Ключові слова: *телекомунікаційна мережа, машинне навчання, обробка подій, акцептор, регулярна мова, CEP-машина.*

REGULAR EVENT PROCESSING IN TELECOMMUNICATION NETWORKS MANAGEMENT

G.N. Zholtkevych, V.V. Dorozhinsky

In the paper the problems of traffic management in telecommunication networks are considered. There are many approaches in the traffic management and optimization. In particular, traffic shaping is one of the methods of the band width management in telecommunication networks. In the paper it is shown that the results obtained by the authors in their previous researches can be widely used in traffic shaping and optimization systems design and development.

Keywords*: Telecommunication Network, Machine Learning, Event Processing, Acceptor, Regular Language, CEP-machine.*