

Концепция маршрутизации трафика в мобильных ad-hoc сетях с использованием высокоточных измерений доступной полосы пропускания

Д. А. Дугаев, Д. С. Качан, И. С. Федотова

В статье представлена концепция протокола маршрутизации в мобильных ad-hoc сетях, основанная на метрике доступной полосы пропускания. Показано, что наиболее часто используемой метрикой в современных протоколах маршрутизации в ad-hoc сетях является метрика кратчайшего пути, или *hop count*. Однако такой подход может оказаться неэффективным в ряде сетевых приложений, таких как передача файлов или разделение ресурсов мобильной сети. В то же время использование метрики доступной полосы пропускания позволит выбирать более эффективные маршруты для конкретных приложений. В работе показано, что измерение времени, необходимого для оценки доступной полосы пропускания на мобильных устройствах, осуществляется при помощи стандартных библиотек измерения времени, которые могут вносить серьезные погрешности. Поэтому для решения подобной задачи в данной работе предложено использование оригинальной, высокоточной библиотеки измерения времени *HighPerTimer*, значительно повышающей точность и скорость измерения.

Ключевые слова: мобильные ad-hoc сети, доступная полоса пропускания, высокоточное измерение времени, метрики протоколов маршрутизации.

1. Введение

Современные протоколы маршрутизации в беспроводных ad-hoc сетях в большинстве своем ориентированы на нахождение оптимального с точки зрения длины пути маршрута до узла-получателя. Метрику, соответствующую данному критерию оптимальности, чаще представляют количеством промежуточных узлов от узла-источника до узла-получателя, или *hop count*. Данная метрика является наиболее распространенной среди всех актуальных протоколов маршрутизации в таких сетях, например, в протоколах: AODV [1], OLSR [2], DSR [3], BATMAN [4] и т. п. Причина популярности метрики *hop count* основывается на том факте, что число промежуточных линков (связей), лежащих по маршруту источник-получатель, напрямую влияет на почти все важнейшие показатели функционирования соединения, отражающие качество предоставляемой услуги по передаче данных (QoS – Quality of Service). А именно: пропускную способность (throughput) соединения, время задержки передачи пакетов, а также процент потерь пакетов. Очевидно, что каждый дополнительный сегмент в маршруте будет только ухудшать эти показатели: пропускная способность будет деградировать из-за дополнительного времени обработки пакетов в новом узле, а также увеличенного времени распространения сигнала, затраченного на новый линк; общий процент пакетных потерь тоже возрастет, так как всегда существует вероятность потери пакета на новом сегменте.

Таким образом, метрика в виде количества промежуточных узлов (или линков) является наиболее обобщенной метрикой, позволяющей с определенной долей эффективности находить оптимальный маршрут в сети, по крайней мере, с точки зрения вышеописанных параметров качества соединения. Однако в реальных беспроводных сетях, имеющих многоузловую ad-hoc топологию, существует множество факторов, ставящих метрику *hop count* в качестве основного критерия оптимальности в проигрышное положение. Прежде всего, этим фактором является *неравномерность* распределения ресурсов по всем линкам сети и вытекающая из этого *несимметричность* линка [5] (значительно отличающаяся пропускная способность линка в прямом (uplink) и обратном (downlink) направлениях), характерная для подавляющего большинства подобных сетей. Кроме того, сеть может быть уже нагружена трафиком других сетевых приложений, который дополнительно занимает ресурсы некоторых маршрутов. Неравномерное распределение ресурсов по сети означает, что производительность отдельных линков может сильно отличаться друг от друга, и поэтому кратчайший маршрут до получателя не всегда будет оптимальным с точки зрения общей пропускной способности соединения.

2. Актуальность работы

Факт неравномерности распределения сетевых ресурсов становится особенно актуальным, учитывая что современные сетевые приложения (в том числе функционирующие на мобильных устройствах) оперируют все большими объемами трафика. В последнее время было разработано некоторое число концепций/подходов, позволяющих передавать значительные объемы трафика между мобильными устройствами посредством организации беспроводной мобильной ad-hoc сети (mobile ad-hoc networks – MANET) [6].

Одна из таких концепций носит название iShare. Она представлена в работе [7] и подразумевает распределение общих ресурсов сети Интернет среди абонентов, подключенных между собой в единую ad-hoc сеть. Таким образом становится возможным предоставить доступ к сети пользователям с мобильными устройствами, находящимися на удалении от шлюза. В качестве соответствующего протокола маршрутизации в работе рассмотрен протокол OLSR, ориентированный на установление кратчайшего маршрута по метрике *hop count*.

Другой проект, описанный в работе [8], нацелен на разработку протокола обмена файлами между устройствами — Optimized Routing Independent Overlay Network (ORION), который работает поверх текущей маршрутизации (основанной на AODV или DSR) в сети MANET. К этому же направлению работ относится проект wi-Share [9], сфокусированный на предоставлении наиболее «качественного» маршрута для обмена файлами (path-quality-aware protocol). Тем не менее в качестве протоколов маршрутизации по-прежнему используются AODV и OLSR, основанные на метрике *hop count*.

Авторы [10] описывают разработку облачного мобильного сервиса для обмена файлами с элементами MANET сети между мобильными устройствами для оптимизации uplink и downlink направлений передачи данных в сторону и со стороны облачного хранилища соответственно. В качестве протокола маршрутизации авторы использовали разработанный ими протокол, работающий по принципу OLSR.

Как видно из вышеизложенного, подобные проекты часто опираются на уже разработанные протоколы маршрутизации для мобильных ad-hoc сетей, которые, в свою очередь, имеют обобщенную метрику оптимизации, завязанную на длину маршрута. Однако, особенно в случаях передачи файлов, имеет смысл искать маршрут с наиболее высоким значением доступной полосы пропускания, который зачастую отличается от кратчайшего

пути. Исходя из этого, в статье представлена концепция протокола маршрутизации, использующего метрику доступной полосы пропускания как основную.

3. Оценка доступной полосы пропускания

Надёжная и быстрая оценка доступной полосы пропускания в сетевом канале из конца в конец — это важная задача для потоковых приложений, задач Quality of Service (QoS), а также для предотвращения перегрузок в сети. В последние десятилетия было опубликовано множество предложений по оценке доступной полосы пропускания. Предложены такие алгоритмы, как *Pathload* [11], *Calibrated Pathload* [12], *Kite* [13], *PathChirp* [14], *Delphi* [15] и т.п. Для оценки доступной полосы пропускания известными алгоритмами используется передача зондирующего трафика, измеряя характеристики которого можно сделать вывод относительно доступных сетевых ресурсов в соединении из конца в конец.

В каждом из методов по-своему реализован алгоритм послыки зондирующего трафика; так, в методах, основанных на оценке дисперсии пар пакетов (*Probe Gap Model*), таких как *Delphi* и *PathChirp*, используется только один поток зондирующего трафика, что позволяет сделать оценку сравнительно быстрой (единицы секунд), однако качество такой оценки остаётся достаточно низким — зачастую встречаются серьёзные ошибки до 30 %. С другой стороны, методы, основанные на анализе потока с постоянной скоростью передачи (*Probe Rate Model*), такие как *Pathload*, *Calibrated Pathload* и *Kite*, итеративно посылают несколько потоков зондирующего трафика, что позволяет сделать измерение более точным: ошибка измерения алгоритма *Kite* и одноимённого приложения – менее 10 %, однако на измерение требуется больше времени, а кроме этого, для передачи необходим больший объём зондирующего трафика.

Абсолютное большинство техник оценки доступной полосы пропускания предполагает, что технологически возможно точно сделать замер времени, когда пакет был отправлен с посылающего интерфейса, а также когда пакет поступил на принимающий интерфейс. Однако в реальности удаётся оперировать лишь со временем, когда зондирующий пакет был создан и когда зондирующий пакет был доставлен непосредственно в приложение (ядро операционной системы). Эта проблема особенно остро встаёт при использовании беспроводных интерфейсов, например, использующихся на мобильных устройствах в ad-hoc сетях. В этом свете выгодно отличается алгоритм *Kite*, способный производить точную оценку (ошибка оценки менее 10 %) в условиях, когда указанное выше предположение о замерах времени не выполняется. На рис. 1 показаны ошибки измерения инструмента *Kite* и *Yaz* (основанный на алгоритме *Calibrated Pathload*). Исследования [12] показали, что *Yaz* из всех ранее представленных алгоритмов оценивает доступную полосу пропускания наиболее точным образом. Однако в условиях объединения прерываний на приёме ошибка его оценки достигает 80 %, о чем можно судить из рис. 1. Объединение прерываний на приёме, подобно использованию беспроводных интерфейсов, вносит существенные искажения в изменения межпакетных временных интервалов непосредственно на конечных устройствах, что не имеет никакого отношения к доступной полосе пропускания. *Kite* в тех же условиях даёт оценку доступной полосы пропускания с точностью до 10 %. На рис. 1 представлены ошибки оценки доступной полосы пропускания в зависимости от пропускной способности соединения в условиях присутствия агрессивного кросс-трафика в соединении (*ct load*).

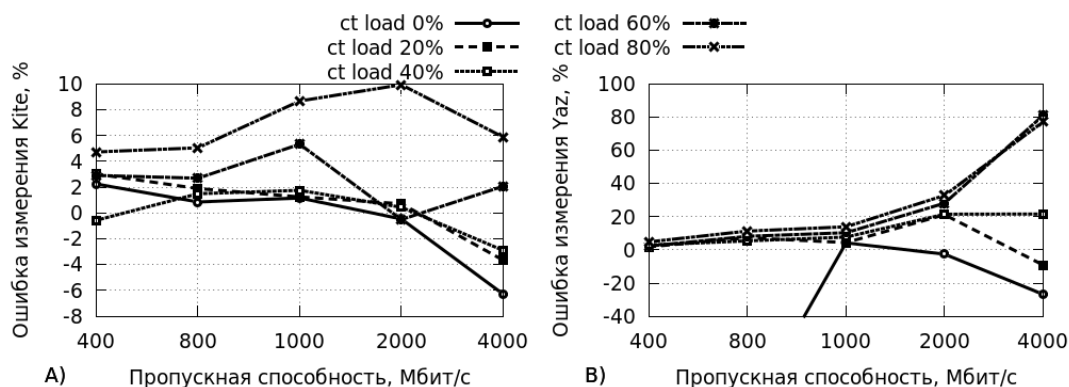


Рис. 1. Сравнение ошибки оценки доступной полосы пропускания

Обмен зондирующим трафиком, который в подавляющем большинстве случаев является набором UDP (User Datagram Protocol) датаграмм, вне зависимости от используемого алгоритма выглядит следующим образом:

На передающей стороне:

1. Процесс передачи зондирующих пакетов ожидает времени, когда необходимо передать следующую пробу.
2. Текущее время и порядковый номер записываются в текущий зондирующий пакет.
3. Пакет отправляется к получателю.

На приёмной стороне:

1. Происходит получение зондирующего пакета.
2. Записывается время получения зондирующего пакета.
3. Из пакета извлекается его порядковый номер и время отправки, которые впоследствии поступают на вход анализирующего модуля.

Суждение о доступной полосе пропускания выносится из анализа изменения межпакетных интервалов, которые были изменены сетью. Поэтому для проведения высококачественной оценки необходимо удостовериться в том, что процесс передачи зондирующих пакетов выполняет ключевые операции, такие как ожидание заданного времени и измерение времени с наибольшей точностью.

4. Высокоточное измерение времени

Задача измерения межпакетного интервала представляет собой классический пример задачи, критичной по времени исполнения, где замер времени может производиться сотни тысяч раз в секунду и чаще. Другими словами, для получения более точной оценки доступной полосы пропускания длительность этого замера, а также используемые ресурсы процессора должны быть минимальны.

Стандартные средства UNIX-подобных операционных систем предоставляют соответствующий интерфейс с набором системных вызовов, которые позволяют получить значение времени напрямую в пространстве пользователя. Хотя на сегодняшний день из-за частого переключения контекста они не являются достаточно эффективными, в частности, где синхронизация времени не должна превышать уровень одной микросекунды. В связи с этим в подобных приложениях часто используется метод прямого обращения к аппаратному счетчику посредством ассемблерных вставок и без использования системных вызовов [16, 17]. Однако если на большинстве современных x86, x86_64 архитектур преобладают относительно известные счётчики TSC, HPET, то архитектуры мобильных устройств требуют более детального изучения.

Рассмотрим процессоры Cortex-A8 и Cortex-A9 архитектуры ARM, наиболее популярной среди встроенных устройств. На платформе ARM Cortex-A8 стандартный интерфейс ОС Linux позволяет получить значение времени за несколько микросекунд, что не всегда является достаточным для выполнения высокоточных измерений. Более того, на платформе Cortex-A9 в качестве стандартного счетчика используется таймер с частотой 32.768 KHz и расширением в 30.518 мкс. Другими словами, это значит, что интерфейс операционной системы не позволяет оперировать значениями времени менее этого интервала. Таким образом, предлагается использовать подход прямого обращения к аппаратному счетчику. Однако следует заметить, что в отличие от архитектуры x86 аппаратные таймеры на встроенных устройствах сильно зависят от семейства, вида процессора и требуют отдельной обработки. Более того, на архитектуре ARM процессоров регистры аппаратных счетчиков находятся в защищенной области ядра ОС, что дополнительно усложняет задачу. В данном исследовании предлагается использовать библиотеку таймерной поддержки *HighPerTimer* [16], [18], которая заменяет системный вызов на прямое обращение к таймеру, объединяя различные аппаратные счетчики, и предоставляет уникальный объект для операций со временем в пространстве пользователя независимо от аппаратной части.

С помощью библиотеки *HighPerTimer* был проведен ряд экспериментов, доказывающий необходимость использования данного подхода при оценках доступной полосы пропускания. Например, прямое обращение к таймеру и сбор значений времени возможен за 400–500 нс. Более того, в случае ожидания процесса передачи пакетов необходимо получить максимально точное измерение задержки заданного времени. Следует заметить, что использование ресурсов процессора при этом сильно ограничено. Вызов системного сна на интервале задержки в 10 мкс показал промах в 130 мкс в случае Cortex-A8 и 85 мкс в случае Cortex-A9. Под промахом здесь понимается отличие ожидаемого времени пробуждения от фактического. Это значит, что стандартный интерфейс с промахом, превышающим ожидаемое время пробуждения в десятки раз, не является приемлемым для выполнения высокоточного сна. При этом, используя метод сна библиотеки *HighPerTimer* [19], ожидание процесса возможно выполнить с промахом в 1 мкс (табл. 1) и затрачивая около 1.89 % ресурсов ЦПУ. На рис. 2–3 рассмотрены оба случая. Таким образом, данный подход позволяет достичь максимальной точности при исследовании полосы пропускания и измерении межпакетной задержки, что в десятки раз эффективнее, чем средствами операционной системы.

Таблица 1. Сравнение результатов промаха сна на 10 мкс

а) на процессоре ARM Cortex A8		
Системный сон	Среднее значение промаха, мкс	Стандартное отклонение, мкс
Сон библиотеки HighPerTimer	130.336	11.375
	1.594	1.647
б) на процессоре ARM Cortex A9		
	Среднее значение промаха, мкс	Стандартное отклонение, мкс
Системный сон	85.086	7.813
Сон библиотеки HighPerTimer	1.131	0.732

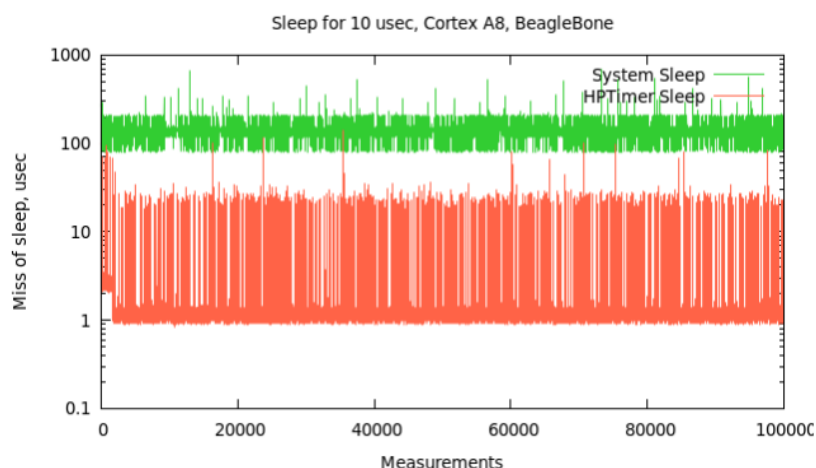


Рис. 2. Промах сна на 10 мкс на процессоре ARM Cortex A8

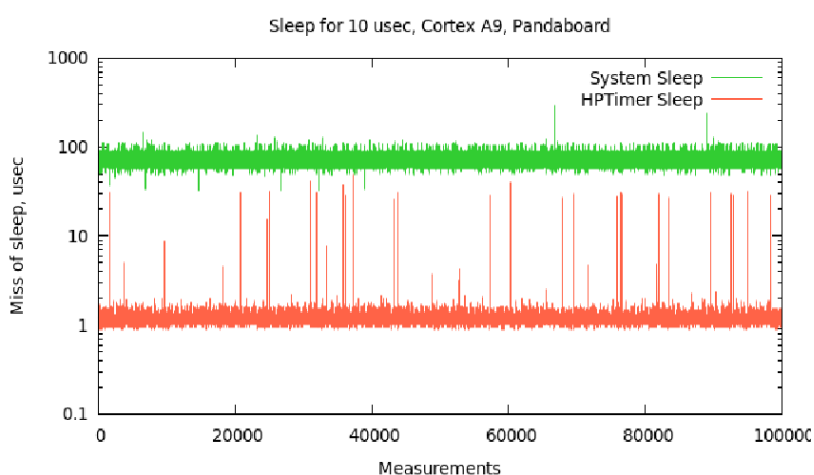


Рис. 3. Промах сна на 10 мкс на процессоре ARM Cortex A9

5. Концепция протокола маршрутизации

Учитывая вышеизложенные подходы по реализации передачи данных между мобильными устройствами посредством организации беспроводной ad-hoc сети, важным является задействовать такую схему маршрутизации трафика, которая позволяла бы максимально эффективно использовать имеющийся ресурс сети по полосе пропускания. Как было указано выше, текущие протоколы, такие как OLSR [2], ориентированы на нахождение кратчайшего пути, что не всегда является наилучшим вариантом.

Исходя из этого, была разработана концепция маршрутизации, пример использования которой представлен на рис. 4, ориентированная на установление маршрута с наибольшей доступной полосой пропускания. Такая схема выглядит следующим образом:

- узлы сети, используя одну из проактивных схем маршрутизации [6] (наподобие схем, применяемых в OLSR и DSDV), создают и постоянно обновляют свою таблицу маршрутов, содержащую в себе информацию о полной и актуальной топологии сети;
- узел, получивший запрос на передачу данных от приложения (узел-источник), используя информацию о всех текущих маршрутах до узла-получателя, запускает процедуру поиска маршрута с максимальной доступной полосой пропускания;

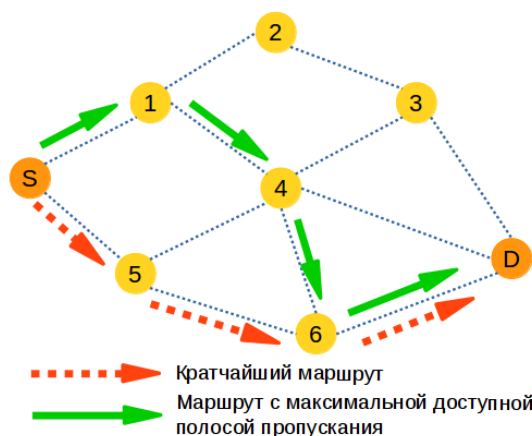


Рис. 4. Подходы к маршрутизации с различными метриками

— после нахождения такого маршрута узел-источник сохраняет его как «маршрут первого выбора» (кэширует), после чего начинает отправку пакетов в сторону узла-получателя по выбранному пути.

Данная схема позволяет динамически находить маршруты с максимальной доступной полосой пропускания, при этом, что немаловажно, учитывая текущую загрузку сети, так как оценка доступной полосы пропускания происходит в реальном времени.

Литература

1. *S. R. Das, E. M. Belding-Royer, C. E. Perkins* Ad hoc On-Demand Distance Vector (AODV) Routing [Электронный ресурс]. URL: <https://tools.ietf.org/html/rfc3561> (дата обращения: 29.11.2015).
2. *P. Jacquet* Optimized Link State Routing Protocol (OLSR) [Электронный ресурс]. URL: <https://tools.ietf.org/html/rfc3626> (дата обращения: 29.11.2015).
3. *D. B. Johnson, D. A. Maltz, J. Broch* DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks // *Ad Hoc Networking*. 2001. pp. 139–172.
4. *J. Chroboczek* A few comments on the BATMAN routing protocol [Электронный ресурс]. URL: <http://lists.athia.debian.org/pipermail/babel-users/2008-August/000151.html> (дата обращения: 29.11.2015).
5. *R. Draves, J. Padhye, B. Zill* Comparison of Routing Metrics for Static Multi-hop Wireless Networks // *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. New York, NY, USA, 2004, pp. 133–144.
6. *D. Dugaev, S. Zinov, E. Siemens V. Shuvalov* A survey and performance evaluation of ad-hoc multi-hop routing protocols for static outdoor networks // *2015 International Siberian Conference on Control and Communications (SIBCON)*. Omsk, Russia, 2015, pp. 1–11.
7. *H.-M. Lin, H.-Y. Lin, A.-C. Pang, C.-W. Hsueh* iShare: An ad-hoc sharing system for internet connectivity // *Wireless Communications and Mobile Computing Conference (IWCMC)*. 2011, pp. 1946–1951.
8. *A. Klemm, C. Lindemann, O. P. Waldhorst* A special-purpose peer-to-peer file sharing system for mobile ad hoc networks // *58th IEEE Vehicular Technology Conference*. 2003, vol. 4, pp. 2758–2763.
9. *E. Karasabun, D. Ertemur, S. Sariyildiz, M. Tekkalmaz, I. Korpeoglu* A path-quality-aware peer-to-peer file sharing protocol for mobile ad-hoc networks: Wi-share // *24th International Symposium on Computer and Information Sciences*. 2009, pp. 322–327.

10. *H. N. Alshareef, D. Grigoras* Mobile Ad-hoc Network Management in the Cloud / 13th IEEE International Symposium on Parallel and Distributed Computing (ISPDC). 2014, pp. 140–147.
11. *M. Jain, C. Dovrolis* Pathload: A Measurement Tool for End-to-End Available Bandwidth // In Proceedings of Passive and Active Measurements (PAM) Workshop. 2002, pp. 14–25.
12. *E. Goldoni, M. Schivi* End-to-end Available Bandwidth Estimation Tools, an Experimental Comparison // Proceedings of the Second International Conference on Traffic Monitoring and Analysis. Berlin, Heidelberg, 2010, pp. 171–182.
13. *D. Kachan, E. Siemens, V. Shuvalov* Available bandwidth measurement for 10 Gbps networks // International Siberian Conference on Control and Communications (SIBCON). Omsk, Russia, 2015, pp. 1–10.
14. *V. J. Ribeiro, R. H. Riedi, R. G. Baraniuk, J. Navratil, L. Cottrell* // pathChirp: Efficient Available Bandwidth Estimation for Network Paths. 2003.
15. *V. Ribeiro, M. Coates, R. Riedi, S. Sarvotham, B. Hendricks, R. Baraniuk* Multifractal cross-traffic estimation // Proceedings ITC Specialist Seminar on IP Trac Measurement, Modeling, and Management. 2000, pp. 15–1.
16. *I. Fedotova, E. Siemens* Self-configurable time source initialization for obtaining high-precision user-space timing // Vestnik SibSUTIS. vol. 4, pp. 22–30.
17. *G.-S. Tian, Y.-C. Tian, C. Fidge* High-Precision Relative Clock Synchronization Using Time Stamp Counters // 13th IEEE International Conference on Engineering of Complex Computer Systems. 2008, pp. 69–78.
18. *I. Fedotova, E. Siemens, H. Hu* A high-precision time handling library // Journal of Communication and Computation. 2013, vol. 10, pp. 1076–1086.
19. *И. Федотова* Реализация новых методов высокоточных ожиданий процессов для ОС Линукс // XV Международная суперкомпьютерная конференция Научный сервис в сети Интернет: все грани параллелизма. Новороссийск, Россия, 2013.

Статья поступила в редакцию 01.12.2015

Дугаев Дмитрий Александрович

аспирант СибГУТИ, 630102, Новосибирск, ул. Кирова, 86

тел. (+49) 157-70-48-4986, e-mail: d.dugaev@emw.hs-anhalt.de

Качан Дмитрий Сергеевич

аспирант СибГУТИ, 630102, Новосибирск, ул. Кирова, 86

тел. (+49) 157-82-34-3452, e-mail: d.kachan@emw.hs-anhalt.de

Федотова Ирина Сергеевна

аспирант СибГУТИ, 630102, Новосибирск, ул. Кирова, 86

тел. (+49) 157-54-09-2309, e-mail: i.fedotova@emw.hs-anhalt.de

Concept of traffic routing in mobile ad-hoc networks based on highly accurate available bandwidth estimations

D. Dugaev, D. Kachan, I. Fedotova

This paper presents a novel concept of routing protocol in mobile ad-hoc networks based on the approach of available bandwidth estimation technique. It is shown, that the most frequently utilized routing metric in such types of networks, the shortest path metric or *hop count*, could be inefficient in certain network applications such as file transfer or any other type of network resource sharing. At the same time, a concept of alternative routing metric, such as current available bandwidth of the end-to-end connection, could significantly benefit in terms of overall network performance since it allows the underlying routing protocol to select the most productive

routes. The paper also shows that the current techniques of system time measurements required for adequate estimation of available bandwidth along certain connection are based on standard time measurement libraries which output the results with high value of measurement error. Therefore, it was suggested to use self-developed highly accurate time estimation library, called *HighPerTimer*, significantly increasing speed and accuracy of time measurement process.

Keywords: mobile ad-hoc networks, available bandwidth, high-precision system time estimation, metrics of routing protocols.