

©Соколов В. А., Корсаков С. В., Смирнов А. В., Башкин В. А., Никитин Е. С., 2015

DOI: 10.18255/1818-1015-2015-4-546-562

УДК 004.72, 004.057.4, 004.023

## Инструментальная система для поддержки разработки и исследования программно-конфигурируемых сетей подвижных объектов

Соколов В. А., Корсаков С. В., Смирнов А. В., Башкин В. А., Никитин Е. С.

*получена 4 сентября 2015*

В данной статье рассмотрены принципы организации беспроводных mesh-сетей — программно-конфигурируемых сетей подвижных объектов. Основное внимание уделяется вопросам построения эффективных алгоритмов маршрутизации для подобных сетей.

Математической моделью системы является стандартная транспортная сеть. В качестве ключевого параметра системы маршрутизации рассматривается коэффициент доступности узла — функция, зависящая от ряда основных и дополнительных параметров («mesh-факторов»), характеризующих маршрут между двумя узлами сети. Каждой паре (дуга, узел) сопоставляется композитный параметр, характеризующий «доступность» узла по маршруту, начинающемуся данной дугой. Лучшим («кратчайшим») маршрутом между двумя узлами считается маршрут с наибольшим коэффициентом доступности.

Описаны правила построения и обновления таблиц маршрутизации узлами сети. Получая анонс от соседа, узел имеет сведения об энергетике соединения, надежности соединения, времени получения анонса, отсутствии промежуточных узлов, а также располагаемой пропускной способности. На основании этой информации ко всем маршрутам, проходящим через данного соседа, может быть применена пенализация (наложение штрафа) или поощрение (увеличение коэффициента доступности). Указанная схема пенализации / поощрения складывается из отдельных аспектов:

1. Пенализация за актуальность информации.
2. Пенализация / вознаграждение за надежность узла.
3. Пенализация за энергетiku соединения.
4. Пенализация за располагаемую пропускную способность.

На основе предложенных эвристических алгоритмов маршрутизации построен симулятор беспроводной mesh-сети подвижных объектов, описание и характеристики которого приведены в статье. Также рассмотрены особенности программной реализации симулятора.

**Ключевые слова:** mesh-сеть, сетевой протокол, маршрутизация, пенализация, эвристический алгоритм, симулятор

**Для цитирования:** Соколов В. А., Корсаков С. В., Смирнов А. В., Башкин В. А., Никитин Е. С., "Инструментальная система для поддержки разработки и исследования программно-конфигурируемых сетей подвижных объектов", *Моделирование и анализ информационных систем*, **22:4** (2015), 546–562.

### Об авторах:

Соколов Валерий Анатольевич, [orcid.org/0000-0003-1427-4937](https://orcid.org/0000-0003-1427-4937),  
докт. физ.-мат. наук., профессор, заведующий кафедрой теоретической информатики,  
Ярославский государственный университет им. П.Г. Демидова,  
ул. Советская, 14, г. Ярославль, 150000 Россия, e-mail: [valery-sokolov@yandex.ru](mailto:valery-sokolov@yandex.ru)

Корсаков Станислав Валентинович, [orcid.org/0000-0003-2349-7980](https://orcid.org/0000-0003-2349-7980),  
директор, ООО «Нетше лаб»,  
ул. Белинского, 28-75, г. Ярославль, 150047 Россия,  
ассистент кафедры теоретической информатики,

Ярославский государственный университет им. П.Г. Демидова,  
ул. Советская, 14, г. Ярославль, 150000 Россия, e-mail: sta@stasoft.net

Смирнов Александр Валерьевич, [orcid.org/0000-0002-0980-2507](https://orcid.org/0000-0002-0980-2507),  
канд. физ.-мат. наук., доцент кафедры теоретической информатики,  
Ярославский государственный университет им. П.Г. Демидова,  
ул. Советская, 14, г. Ярославль, 150000 Россия, e-mail: alexander\_sm@mail.ru

Башкин Владимир Анатольевич, [orcid.org/0000-0002-2534-1026](https://orcid.org/0000-0002-2534-1026),  
докт. физ.-мат. наук., доцент, доцент кафедры теоретической информатики,  
Ярославский государственный университет им. П.Г. Демидова,  
ул. Советская, 14, г. Ярославль, 150000 Россия, e-mail: v\_bashkin@mail.ru

Никитин Евгений Сергеевич, [orcid.org/0000-0002-2341-9950](https://orcid.org/0000-0002-2341-9950), студент,  
Ярославский государственный университет им. П.Г. Демидова,  
ул. Советская, 14, г. Ярославль, 150000 Россия, e-mail: nik.zhenya@gmail.com

**Благодарности:**

Работа выполнена при поддержке Российского фонда фундаментальных исследований (проект № 15-07-03038 А).

## Введение

Стремительное развитие Интернета вещей, автономных либо удаленно управляемых беспилотных аппаратов (исследовательских, спасательных и других робототехнических комплексов) и средств телеприсутствия переводит в разряд важнейших проблем организацию и поддержание эффективной работы беспроводных сетей передачи данных, узлами которых являются подвижные объекты.

Подобные подвижные беспроводные сети обладают рядом существенных особенностей, таких как:

- множественность узлов;
- перемещение узлов сети относительно друг друга и относительно неподвижного наблюдателя с некоторой скоростью;
- недетерминированное во времени, а возможно, и в пространстве появление новых узлов сети и отключение имеющихся;
- неоднородность физической среды в пределах сети;
- изменяющиеся во времени условия взаимной радиовидимости между узлами такой сети;
- возможные топологии сети с двумя и более возможными маршрутами между узлами, а также с узлами, достижимыми исключительно через транзитные узлы;
- равноправность (в сетевом смысле) узлов в сетях;
- ограниченность вычислительных ресурсов узлов.

В то же время к рассматриваемым нами сетям предъявляются достаточно «стандартные» требования:

- эффективное использование частотного ресурса и имеющейся пропускной способности сети;
- потребность в значительной располагаемой пропускной способности на всем протяжении маршрута передачи информации в связи с необходимостью передачи видео, голосовой, телеметрической и иных видов информации.

Классические централизованные беспроводные сети (сети с главным устройством — точкой доступа или базовой станцией) непригодны для подобных ситуаций ввиду потери работоспособности при отключении центрального устройства. Однако такие сети широко применяются в системах телеприсутствия и автономных аппаратах в конфигурациях «одно устройство — один пульт управления», т. е. для сетей топологии «точка-точка».

Классические беспроводные одноранговые (они же Ad-Hoc) сети непригодны для подобных ситуаций ввиду отсутствия механизмов маршрутизации и, как следствие, невозможности передачи информации между узлами, не являющимися соседями.

Модификация одноранговых беспроводных сетей посредством внедрения в них функции маршрутизации выглядит наиболее перспективным вариантом решения проблемы. На данный момент существует немало решений для организации беспроводных сетей передачи данных с равноправными узлами (mesh-сетей). Большинство из них ориентировано на организацию и поддержку функционирования неподвижных mesh-сетей. Среди наиболее известных и доступных можно назвать *batman\_adv* [1] и стандарт 802.11s [2]. Их недостатком является учет ограниченного числа факторов (число хопов, пропускная способность) выбора маршрута. В

них совершенно не учитываются факторы изменения топологии сети и надежности узлов.

Существуют реализации (см. [3] – [4]) средств управления подвижными mesh-сетями. Однако все известные решения либо являются закрытыми и в настоящее время ориентированными на военное применение, либо являются предметом академических исследований. По причине закрытости работ над средствами управления подвижными mesh-сетями практически невозможно судить о методах организации процесса маршрутизации, используемого в этих решениях.

Тем не менее, анализ решений для неподвижных mesh-сетей, а также математические соображения подсказывают, что в процессе маршрутизации для подвижных mesh-сетей оптимальное решение должно учитывать факторы времени (возраста маршрутной информации) и поведения узлов сети (предсказание состояния узла на основе статистических данных) наряду с такими классическими факторами, как число хопов и пропускная способность.

В данной статье мы рассмотрим принципы построения mesh-сетей указанного вида.

В разделе 1 предложены алгоритмы маршрутизации в mesh-сети.

Разделы 2–3 посвящены принципам организации и особенностям программной реализации симулятора mesh-сети.

Напомним также несколько определений, используемых для рассматриваемых нами mesh-сетей.

*Тайм-слот* — временной интервал, в течение которого беспроводным компьютерным устройством будет на выбранной скорости гарантированно завершён процесс передачи одного кадра-маяка и, как минимум, одного кадра данных максимального размера с защитными интервалами перед передачей кадра и после нее. Тайм-слот (продолжительность этого временного интервала) фиксирован для любого узла сети. Любые два действующих узла в сети могут использовать один и тот же тайм-слот при условии, что они не принадлежат одному фрагменту сети. Для тайм-слотов в сети определены три состояния: свободный, занятый, спорный.

Сеть оперирует  $N$  тайм-слотами, где  $N$  может принимать значения от 2 до 64. Количество тайм-слотов неизменно для конкретной работающей сети и может быть изменено только перезапуском сети с иными настройками.

*Раунд* — время, включающее в себя все  $N$  тайм-слотов.

*Доступ к среде с временным мультиплексированием* — механизм неколлизионного доступа к среде, при котором каждому узлу сети для передачи назначаются определенные, как правило, непересекающиеся временные интервалы (тайм-слоты).

## 1. Эвристические алгоритмы как часть инструментальной системы для исследования сетей подвижных объектов

Существует большое количество протоколов маршрутизации в мобильных сетях (см., например, обзор [5]), тем или иным способом решающих проблему выбора оптимального маршрута. Разнообразие подходов объясняется разнородностью об-

ластей применения рассматриваемых сетей и, соответственно, различиями в целевых функциях, характеристиках среды, природе помех, скорости изменения сетевой конфигурации и т.д. Для сравнения успешности того или иного подхода в конкретной системе используются, как правило, численные и натурные эксперименты. В качестве примера можно привести исследование [6], посвященное сравнению наиболее известных алгоритмов маршрутизации в мобильных сетях (DSDV, TORA, DSR и AODV).

Особенностью рассматриваемого нами случая является достаточно высокая скорость изменения топологии сети и энергетических характеристик соединений между узлами, а также используемая схема разделения времени соединения на основе тайм-слотов. Необходимость «быстрого» установления соединения и достаточно небольшие временные интервалы устойчивости конфигурации сети требуют высокой адаптивности системы маршрутизации. Кроме того, отсутствие диспетчера в сети приводит к обязательности использования децентрализованных (мультиагентных) алгоритмов.

Для классических ad hoc сетей адаптивный децентрализованный подход к маршрутизации не является чем-то новым, однако известные нам работы рассматривают достаточно специфические предметные области. В частности, в [7] предлагается основанный на случайных блужданиях алгоритм для сенсорных сетей (передача данных в основном от сенсора к серверу); в [8] рассматривается локальный подход, использующий оценку остаточной пропускной способности соединения (для проводных сетей); в [9] приводится основанный на потенциалах алгоритм организации доступа внутренних узлов mesh-сети к внешним шлюзам.

Перечисленные эвристики обладают рядом важных достоинств, однако, на наш взгляд, ни одна из них не подходит для случая беспроводных сетей подвижных объектов. Основной причиной является то, что в наших сетях возникает целый ряд новых факторов, влияние которых мы хотим каким-то образом учитывать в алгоритме при выборе того или иного маршрута. Кроме того, заметим, что характеристики «выгодности» можно описать множеством способов. Именно в этой множественности проявляется принципиальная трудность формализации задачи маршрутизации в подвижных mesh-сетях. На практике возможен лишь подбор эмпирических формул и коэффициентов, в «нужной» степени учитывающих «нужные» характеристики реальной системы.

Для упрощения задачи в наших алгоритмах мы используем единый композитный (вычисляемый) параметр, отвечающий за «оптимальность» маршрута с точки зрения «mesh-факторов» (его характеристики будут сформулированы далее). Если называть этот параметр «ценой» («длиной») маршрута, то задача маршрутизации сводится к известной проблеме поиска потока с наименьшими затратами (наименьшей длины).

Основными критериями при выборе алгоритма являлись:

- простота реализации;
- эффективность (в среднем);
- возможность мультиагентной реализации (одноранговость сети, отсутствие узла-диспетчера);
- скорость реагирования на изменения конфигурации сети (как топологии сети, так и загруженности узлов/соединений).

Были рассмотрены следующие варианты:

1. Алгоритмы на основе метода проталкивания предпотока (ПП) — push-relabel.
2. Алгоритмы на основе метода последовательного насыщения кратчайших путей (КП) — successive shortest path.

Алгоритмы на основе проталкивания предпотока в настоящее время считаются одними из наиболее эффективных среди полиномиальных по времени. Описания таких алгоритмов можно найти во множестве классических книг (например, [10]). К их недостаткам можно отнести достаточно сложную реализацию и большое число предварительных служебных сообщений (согласований) в мультиагентном варианте алгоритма.

Алгоритмы последовательного насыщения кратчайших путей, напротив, имеют достаточно простую реализацию и не требуют подробного описания. Поток пускается по кратчайшему (самому дешевому) маршруту до его насыщения, далее по следующему кратчайшему маршруту до его насыщения, и т.д. до полного исчерпания потока.

При предварительном сравнительном анализе двух подходов выяснились следующие преимущества КП перед ПП:

1. Объем хранимой на узле информации для КП может быть существенно меньше, поскольку его можно реализовать с использованием только лишь таблицы маршрутизации (не храня на узле всю известную структуру оптимальных путей). Схемы на основе таблиц, возможно, менее универсальны, однако достаточно эффективны. Например, в [11] показано, что «табличный» алгоритм AODV эффективнее более «интеллектуального» DSR в зашумленных и загруженных сетях, а также в сетях с динамически изменяющейся топологией. Кроме того, как показано в [12], для табличных схем маршрутизации достаточно просто реализуются методы защиты от атак «византийского» типа (когда в качестве злоумышленника выступают внутренние узлы mesh-сети).

2. При мультиагентной реализации КП позволяет начать передачу данных сразу после обнаружения адреса узла-получателя узлом-отправителем в хотя бы одной из таблиц маршрутизации узлов-соседей (и выбранный маршрут будет достаточно эффективен) — то есть скорость реагирования на смену конфигурации сети максимальна.

3. Алгоритм КП легко адаптировать для сети с большим количеством разнонаправленных потоков (с учетом изменения коэффициентов доступности узлов и неполного насыщения кратчайших путей). Локальные методы оценки оптимальности маршрутов уже достаточно хорошо себя зарекомендовали на практике [13].

Однако очень существенным недостатком КП по сравнению с ПП и другими полиномиальными алгоритмами является его неэффективность: в худшем случае требуется экспоненциальное число операций для нахождения оптимального потока между двумя узлами. Тем не менее, в ходе обзора литературы были обнаружены следующие факты:

1. В работе [14] исследованы проблемы эффективного перераспределения электроэнергии в сетях, состоящих из множества активных агентов (производителей и потребителей). Как и в нашем случае, элементы сети (узлы и дуги) обладают и пропускной способностью, и «ценой» (производства, потребления и передачи электроэнергии). Авторы при помощи математического моделирования сравнивают стра-

тегии маршрутизации потоков, основанные на ПП и КП, и на основе анализа статистики приходят к выводу, что в среднем ПП является более эффективным для «радиальных» сетей (древовидной топологии), а КП — для «mesh-сетей» (так в статье названы графы произвольной топологии).

2. В работе [15] проведен анализ КП методами теории гладкого анализа вычислительной сложности алгоритмов (Smoothed Analysis). Эта теория возникла относительно недавно [16] и позволяет не учитывать при оценке трудоемкости алгоритмов крайние (вырожденные) случаи индивидуальных алгоритмических проблем. Авторы работы [15] доказывают, что с точки зрения Smoothed Analysis алгоритм КП является полиномиальным (более того, он даже более эффективен, чем многие считающиеся эффективными классические полиномиальные алгоритмы).

На основании проведенного анализа в качестве основы ПО маршрутизации в подвижных mesh-сетях предлагается использовать алгоритм последовательного насыщения кратчайших путей.

Математической моделью системы является стандартная транспортная сеть. Задача маршрутизации формализуется как задача поиска потока в сети с минимальными затратами. В качестве алгоритма решения выбран алгоритм последовательного насыщения кратчайших путей, где в качестве аналога «длины» пути рассматривается коэффициент доступности маршрута на первой транзитной вершине.

Ключевой параметр системы маршрутизации — *коэффициент доступности* (КД) узла — функция, зависящая от ряда основных и дополнительных параметров («mesh-факторов»), характеризующих маршрут между двумя узлами сети. Каждой паре (дуга, узел) сопоставляется композитный параметр, характеризующий «доступность» узла по маршруту, начинающемуся данной дугой. Таким образом, у каждой дуги есть ровно столько таких характеристик (коэффициентов доступности узлов), сколько узлов достижимы от вершины-окончания данной дуги (т.е. если эта вершина выступает в качестве первой транзитной вершины всех соответствующих маршрутов). Неформально, если КД пары  $((s, u), t)$  выше, чем КД пары  $((s, v), t)$ , то, исходя из комбинации дополнительных факторов, при передаче данных из  $s$  в  $t$  «лучше» использовать в качестве первой транзитной вершины  $u$ , а не  $v$ .

Рассмотрим структуру функции вычисления КД узла. Отдельные компоненты данной функции будем называть *штрафами (пенализацией)*. Объектами наложения штрафов могут являться:

- *надежность узла* —  $F_1(f) = F_f$ ;
- *число промежуточных узлов* —  $F_2(n) = F_n$ ;
- *энергетика соединения* с учетом истории ее изменения —  $F_3(e) = F_e$ ;
- *располагаемая пропускная способность* с учетом ее изменения —  $F_4(p) = F_p$ ;
- *возраст данных о маршруте* —  $F_5(t) = F_t$ . Заметим, что поскольку мы имеем дело с подвижной сетью, а процесс распространения данных растянут по времени, наиболее свежие данные должны иметь преимущество.

Таким образом, коэффициент доступности узла можно выразить абстрактной формулой

$$K = F(F_f, F_n, F_e, F_p, F_t).$$

Будем считать, что значение данной функции находится в некотором интервале  $[K_{\min}, K_{\max}]$ , где  $K_{\max}$  — значение, соответствующее максимально возможной до-

ступности узла. Маршруты, достигшие значения  $K_{\min}$ , будем исключать из таблицы маршрутизации (либо игнорировать) и, соответственно, не будем анонсировать.

Лучшим («кратчайшим») маршрутом между двумя узлами будем считать маршрут с наибольшим коэффициентом доступности. Именно этот маршрут будет насыщаться первым по выбранной стратегии алгоритма последовательного насыщения кратчайших путей.

Ключевые структурные характеристики архитектуры системы маршрутизации:

- информация для определения топологии сети и построения маршрутов в реактивном режиме распространяется в кадрах-маяках;
- перерасчет таблицы маршрутизации происходит каждый раз по приходу кадра-маяка, кадра данных, а также квитанции о доставке кадра;
- локальная таблица маршрутизации узла имеет ограниченное время жизни, записи в таблице маршрутизации постоянно «устаревают»;
- на каждом узле-ретрансляторе выполняется проактивная часть маршрутизации.

Используется минимальный по расходуемым ресурсам вариант таблицы маршрутизации, в котором для каждого МАС-адреса в сети, известного конкретному узлу, поставлен в соответствие МАС-адрес соседа-ретранслятора, обеспечивающего (по мнению этого самого узла) маршрут до получателя, а также некоторое свойство (коэффициент доступности), позволяющее определять этот «лучший маршрут».

Исходя из ограничений на целочисленную математику и 32-разрядную архитектуру, в качестве коэффициента доступности используется 16-разрядная величина, а для штрафов — два 16-разрядных значения (числитель и знаменатель). Все операции укладываются в 32-разрядные требования.

Опишем процесс формирования сведений о сети и маршрутах в ней.

Для начала рассмотрим процесс формирования информации на отдельно взятом узле сети, не имеющем локально подключенных устройств и не имеющем связи с другими устройствами в сети. Такое устройство просто анонсирует свой МАС-адрес в сеть с максимальным коэффициентом доступности  $K_{\max}$ .

Теперь рассмотрим процесс формирования информации на отдельно взятом узле сети, имеющем локально подключенные устройства и не имеющем связи с другими устройствами в сети. Данный узел анонсирует свой МАС-адрес с коэффициентом  $K_{\max}$ , а также МАС-адреса всех локально подключенных устройств с тем же коэффициентом  $K_{\max}$ .

Рассмотрим поведение соседа, который получает анонсы от вышеуказанного узла.

Получая анонс, узел имеет сведения об энергетике соединения, надежности соединения, времени получения анонса, отсутствии промежуточных узлов, а также располагаемой пропускной способности. Т.е. сосед может рассчитать все штрафы, налагаемые на такое соединение, а как следствие, на маршруты, доступные через данного соседа. Соответственно, если мы немного изменим функцию расчета коэффициента достижимости для учета предыдущего значения, мы сможем пересчитывать коэффициент, получаемый от соседа  $K' = F(F_f, F_n(1), F_e, F_p, K)$ .

Поскольку мы заявили принцип преимущества «свежих данных», нам требуется функция перерасчета коэффициента от времени  $K'' = F'(F_t(n), K)$ , где  $n$  — число микросекунд, прошедших с момента последнего пересчета по времени.



Рассмотрим теперь *реактивную часть* процесса маршрутизации.

Каждый узел анонсирует собственную таблицу маршрутизации, в виде доступных через него МАС-адресов и коэффициентов достижимости, где свой МАС-адрес и МАС-адреса локально подключенных устройств имеют коэффициент  $K_{\max}$ , а другие узлы имеют текущий расчетный коэффициент, находящийся в интервале  $K_{\min} < K < K_{\max}$ .

При получении анонса узел производит перерасчет коэффициентов доступности для узлов, имеющих коэффициент меньше  $K_{\max}$ , из локальной таблицы маршрутизации по формуле  $K'' = F'(F_t(n), K)$  (где  $n$  — число микросекунд с момента предыдущего пересчета).

Производится исключение маршрутов, достигших значения коэффициента  $K_{\min}$ .

Коэффициенты для маршрутов из анонса пересчитываются по формуле  $K' = F(F_f, F_n(1), F_e, F_p, K)$ , т.е. применяются штрафы.

Далее выполняется объединение двух таблиц (локальной и анонсируемой) по правилу:

- если анонсируемый МАС-адрес отсутствует в локальной таблице маршрутизации, он копируется из анонса вместе с обновленным коэффициентом доступности. Транзитным узлом для него назначается анонсирующий данный адрес сосед;
- если для анонсируемого МАС-адреса имеется маршрут, то маршрут из анонса добавляется в таблицу маршрутизации;
- обновляются временные метки в локальной таблице маршрутизации.

В результате работы предлагаемой схемы на каждом узле сети имеется таблица маршрутизации, в которой для каждого МАС-адреса в сети имеется минимум одна запись с МАС-адресом непосредственного соседа узла, начинающего маршрут к данному МАС-адресу.

В предлагаемой схеме используется пенализация (наложение штрафа) и поощрение (увеличение коэффициента доступности) соседей в зависимости от их «надежности». Под надежностью узла в данном контексте понимается пропуск кадров-маяков от узла-соседа, а также отсутствие квитанций от соседа о доставке кадров.

В предлагаемой схеме отсутствует обновление маршрутов через соседа при пропуске кадра-маяка и, как следствие, уменьшается коэффициент доступности зависимых маршрутов ввиду пересчета по времени. Уменьшение коэффициента доступности проводится также при неполучении квитанции от узла соседа. При этом данное уменьшение весьма существенно.

В качестве поощрения вводится увеличение коэффициента доступности маршрута при получении квитанции о доставке кадра от узла-соседа. Поощрение пропорционально размеру доставленного кадра.

Данная стратегия позволяет быстро реагировать на изменение условий соединения: прекратить использование маршрута в случаях выпадения кадров либо выявить и активно использовать соединение с высокой пропускной способностью.

Ниже рассмотрим отдельные виды пенализации.

#### 1. Пенализация за актуальность информации.

Поскольку мы имеем дело с подвижной mesh-сетью, для определения лучшего маршрута нам нужна максимально свежая информация. В связи с этим логичной выглядит пенализация маршрутов по мере устаревания информации о них.

Предлагается ввести пенализацию маршрутов за каждую микросекунду, прошедшую с момента их получения. Степень пенализации должна зависеть от времени (прямо пропорциональна) и расчетной средней скорости взаимного перемещения объектов в целевой сети (прямо пропорциональна). Она является эмпирической величиной и должна быть установлена в ходе экспериментов (в том числе на симуляторе). Например, для малоподвижной сети (ССВПО до 20 км/ч) и при размере тайм-слота в 10 миллисекунд, максимальная пенализация за раунд не должна превышать 10% от  $K_{\max}$ .

## 2. Пенализация / вознаграждение за надежность узла.

Основывается на статистических данных по пропуску кадров-маяков от узла и доставке / недоставке кадров к узлу. Эмпирическая величина, устанавливаемая в ходе экспериментов. В связи с подвижностью сети, нет смысла собирать сведения о «надежности» узла в течение значительного времени (по меркам сети).

Содержит два компонента пенализации:

- *временной* (все маршруты через узел значительно штрафуются при непоступлении крайнего кадра-маяка);
  - *отсутствие квитанции* (маршруты через узел значительно штрафуются при неполучении квитанции о доставке кадра)
- и один компонент поощрения:
- *получение квитанции* (коэффициенты доступности для маршрутов через узел увеличиваются пропорционально размеру успешно переданного кадра).

## 3. Пенализация за энергетику соединения.

Данный вид пенализации является наиболее важной и сложной частью в системе штрафов.

Для вычисления штрафов предлагается оперировать четырьмя сущностями:

- 1) средний уровень сигнала;
- 2) тренд изменения среднего уровня сигнала;
- 3) текущий уровень сигнала;
- 4) скорость и направление изменения уровня сигнала.

Уровень сигнала — это величина, находящаяся в определенном диапазоне. Приближение к границам диапазона вредно, поскольку ведет к переусилению сигнала или потере связи. Соответственно, пенализация при приближении текущего уровня сигнала к границам диапазона должна быть максимальной. Пенализация при приближении среднего уровня к границам диапазона должна быть значительной. В обоих случаях при нахождении уровней в середине диапазона пенализация должна отсутствовать. Характер соответствующих функций нелинейный. Правила пенализации по энергетике:

- 1) устойчивый тренд на снижение среднего уровня сигнала должен приводить к увеличению штрафа; устойчивый тренд на рост — к уменьшению штрафа; отсутствие изменений — не влиять на штраф;
- 2) высокая скорость уменьшения сигнала при наличии сигнала в нижней части диапазона должна приводить к высокому штрафу;
- 3) высокая скорость уменьшения сигнала при наличии сигнала в верхней части диапазона должна приводить к среднему штрафу;
- 4) высокая скорость уменьшения сигнала при наличии сигнала в середине диа-

пазона должна приводить к среднему штрафу. При этом уровень штрафа должен быть больше предыдущего;

5) средняя скорость уменьшения сигнала при наличии сигнала в нижней части диапазона должна приводить к высокому штрафу. При этом штраф должен быть меньше аналогичного за высокую скорость;

6) средняя скорость уменьшения сигнала при наличии сигнала в верхней части диапазона не должна приводить к штрафу;

7) средняя скорость уменьшения сигнала при наличии сигнала в середине диапазона должна приводить к среднему штрафу. При этом уровень штрафа должен быть меньше соответствующего за высокую скорость;

8) низкая скорость уменьшения сигнала при наличии сигнала в нижней части диапазона должна приводить к среднему штрафу;

9) низкая скорость уменьшения сигнала при наличии сигнала в верхней части диапазона не должна приводить к штрафу;

10) низкая скорость уменьшения сигнала при наличии сигнала в середине диапазона должна приводить к малому штрафу;

11) высокая скорость увеличения сигнала при наличии сигнала в нижней части диапазона должна приводить к среднему штрафу;

12) высокая скорость увеличения сигнала при наличии сигнала в верхней части диапазона должна приводить к высокому штрафу;

13) высокая скорость увеличения сигнала при наличии сигнала в середине диапазона должна приводить к малому штрафу;

14) средняя скорость увеличения сигнала при наличии сигнала в нижней части диапазона должна приводить к малому штрафу;

15) средняя скорость увеличения сигнала при наличии сигнала в верхней части диапазона должна приводить к среднему штрафу;

16) средняя скорость увеличения сигнала при наличии сигнала в середине диапазона не должна приводить к штрафу;

17) низкая скорость увеличения сигнала при наличии сигнала в нижней части диапазона должна приводить к среднему штрафу;

18) низкая скорость увеличения сигнала при наличии сигнала в верхней части диапазона не должна приводить к штрафу;

19) низкая скорость увеличения сигнала при наличии сигнала в середине диапазона не должна приводить к штрафу.

Уровни штрафных коэффициентов являются эмпирическими величинами и устанавливаются в процессе экспериментов.

#### 4. Пенализация за располагаемую пропускную способность.

Пенализация за располагаемую пропускную способность соединения на данном этапе выглядит достаточно простой — штраф обратно пропорционален располагаемой пропускной способности.

Для сети с временным разделением доступа к среде предложенная схема реактивного режима одновременно является практически полной реализацией *практического режима*.

На начало процесса передачи кадра данных на узле сформирована таблица маршрутизации, имеющая «лучший маршрут» до узла назначения.

Полная реализация проактивного режима заключается в реализации процедуры замены адреса ретранслятора в заголовке кадра на соответствующий адрес из локальной таблицы маршрутизации, а также установки актуальной скорости передачи непосредственно перед передачей кадра.

## 2. Характеристика и принципы организации симулятора

Для моделирования различных ситуаций, возникающих в mesh-сети указанного вида, был разработан симулятор, предназначенный для работы под операционными системами Ubuntu 14.04 LTS и Windows 7. Он состоит из двух компонентов:

- «Редактор»;
- «Интерпретатор».

Компонент «Редактор» позволяет определять и изменять конфигурацию сети, для которой необходимо выполнить моделирование. Конфигурация сети представляется с помощью набора параметров:

- 1) *количество узлов сети* (от 3 до 16);
- 2) *матрица смежности / энергетики* — квадратная матрица, в которой для каждой пары узлов задается величина энергетики соединения между ними (целое число в диапазоне от  $-127$  до  $12$  dBm); в случае, когда соединение отсутствует, указывается значение 13. При этом для любой пары узлов  $\{a, b\}$  допускается несоответствие значений энергетики на соединениях  $(a, b)$  и  $(b, a)$ ;
- 3) *вектор длин очередей* (неотрицательные целые значения) — показывает, сколько кадров находится в очереди на каждом узле;
- 4) *вектор времени* (целые значения в диапазоне от 0 до 8388 мс) — возраст информации о каждом узле.

Компонент «Интерпретатор» позволяет выполнять моделирование различных ситуаций в mesh-сети указанной конфигурации. Функциональные возможности «Интерпретатора»:

- 1) построение таблиц маршрутизации для отдельного узла либо для всех узлов в сети;
- 2) отслеживание продвижения одного кадра между двумя выбранными узлами сети.

Компонент «Интерпретатор» является консольной утилитой, поддерживающей 4 режима работы:

1. *Инициализация сети с консольным выводом.* Инициализация сети по конфигурации из файла; вывод осуществляется на консоль; результатом выполнения является набор таблиц маршрутизации для всех узлов сети.
2. *Инициализация сети с файловым выводом.* Инициализация сети по конфигурации из файла; вывод осуществляется в файл; результатом выполнения является набор таблиц маршрутизации для всех узлов сети.
3. *Инициализация узла с файловым выводом.* Инициализация сети по конфигурации из файла; вывод осуществляется в файл; результатом выполнения является таблица маршрутизации для выбранного узла сети.

4. *Отслеживание кадра.* Инициализация сети по конфигурации из файла и отслеживание продвижения информационного кадра от узла-отправителя к узлу-получателю; вывод осуществляется в файл; результатом выполнения является набор таблиц маршрутизации для всех узлов сети и информация по отслеживанию кадра.

Разработанный симулятор предназначен прежде всего для анализа и оценки используемой модели пенализации и алгоритма маршрутизации без проведения натурных испытаний. При этом все методы расчета таблиц маршрутизации и выбора конкретного маршрута эквивалентны тем, что применяются в реальных устройствах (описание методов см. в разделе 1.).

В текущей версии симулятора инициализация сети и построение таблиц маршрутизации осуществляется следующим образом.

1. Узлы поочередно «включаются», получая при этом индивидуальный номер тайм-слота.

2. Каждый узел анонсирует себя путем отправки кадра-маяка в момент включения, а также в начале своего тайм-слота в каждом раунде. Кадр-маяк содержит информацию об узле, в том числе таблицу маршрутизации, получаемую после применения всех штрафов на этом узле.

3. Все узлы-соседи, включенные в момент передачи кадра-маяка, получают этот кадр и обновляют свои таблицы маршрутизации на основе полученной информации (правила обновления таблицы см. в разделе 1.). Узел  $b$  считается соседом узла  $a$ , если в матрице смежности / энергетике указано наличие соединения  $(a, b)$ .

После включения всех узлов сети шаги 2–3 повторяются циклически до завершения симуляции.

При обновлении таблиц маршрутизации к записям применяются следующие элементы пенализации:

1. Пенализация записей по времени на основе вектора времени. Чем больше значение некоторого компонента вектора времени, тем «старее» информация о соответствующем узле и больше штраф.

2. Пенализация по энергетике соединения. Поскольку в данной версии мы работаем только с одним снимком сети (конфигурация связей и энергетика не меняются), то используется простая пенализация по энергетике, не учитывающая характер изменения топологии сети и энергетике соединений во времени.

3. Пенализация за пропускную способность соединения. В данной версии симулятора предусмотрена возможность передачи только одного информационного кадра, поэтому пенализация за пропускную способность сводится к пенализации по длине очереди (чем больше очередь на узле, тем меньше пропускная способность соединения). Соответственно, размер накладываемого штрафа определяется вектором длин очередей — чем больше значение, тем больше пенализация.

В симуляторе не используется пенализация / поощрение за «надежность» узла, поскольку в данной реализации невозможно собрать статистику по передаче информационных кадров (отслеживается только один кадр).

При работе симулятора в режиме «Отслеживание кадра» используется следующий алгоритм продвижения информационного кадра.

Пусть необходимо передать кадр от узла  $a$  узлу  $b$ . С данным кадром ассоции-

руется параметр  $TTL$  (Time To Live, время жизни), в начальный момент времени равный 255.

Пусть в некоторый момент времени кадр находится на некотором узле  $x$  и  $TTL \neq 0$ . Во время тайм-слота, соответствующего узлу  $x$ , этот узел ищет в своей таблице маршрутизации все маршруты до узла  $b$ , выбирает среди них тот, который обладает максимальным коэффициентом доступности, и отправляет кадр тому узлу  $y$ , который указан в качестве ретранслятора для выбранного в таблице маршрута.

Получая кадр, узел  $y$  проверяет его. Если  $y = b$ , то кадр доставлен. Иначе узел  $y$  уменьшает значение  $TTL$  на 1. Если  $TTL = 0$ , кадр уничтожается.

### 3. Особенности программной реализации

Как уже отмечалось выше, нами был реализован симулятор, состоящий из двух компонентов: «Редактор» и «Интерпретатор».

Компонент «Редактор» является приложением с графическим интерфейсом. Основная задача этого приложения — создание и редактирование конфигураций имитируемой сети. Исходный код компонента «Редактор» представляет собой проект на языке C++ с использованием кроссплатформенного инструментария Qt и соответствует известному шаблону проектирования “Model–View–Controller” («Модель–Вид–Контроллер»).

Напомним, что согласно данному шаблону основная логика программы делится на модель, вид и контроллер. Модель в данном случае подразумевает под собой внутреннюю логику программного обеспечения, вид отвечает за отображение состояния модели пользователю, а контроллер обеспечивает связь между пользователем и системой.

Компонент «Интерпретатор» реализован в виде консольного приложения. Данный компонент позволяет выполнять симуляцию сети на основе файла конфигурации, созданного при помощи приложения «Редактор». Исходный код интерпретатора написан на языке C. Функционал программы логически разделен на уровни (таблица 1).

Таблица 1. Разделение логики в компоненте «Интерпретатор»

Файл	Функционал
nsutils.h	Подключение необходимых системных библиотек
list.h	Linux-реализация двусвязных списков
nstypes.h	Описание структур, используемых в программе
nsalloc.h	Реализация работы с памятью
nsio.h	Реализация операций ввода-вывода информации
nsrouting.h	Реализация работы узлов и обмена кадрами
nssimulator.h	Симуляция сети
tested.h	Тестируемый функционал

Остановимся подробнее на реализации работы узлов и симуляции работы сети в целом.

Интерпретатор в начале своей работы производит инициализацию сети, включающую в себя поочередное включение узлов сети с последующим обменом маршрутной информацией между включенными узлами.

Инициализация, как и симуляция работы сети, производится пошагово, один шаг соответствует одному тайм-слоту. По прохождению тайм-слотов всех узлов отсчитывается раунд. Сеть считается проинициализированной, если с момента включения всех узлов в сети прошло количество раундов, равное количеству задействованных в симуляции узлов. После инициализации в сети применяются временные задержки на основе вектора времени. На этом инициализация завершается.

Симуляция сети проходит тем же образом, что и инициализация. Во время симуляции работы сети производится генерация и обмен кадрами между узлами с сохранением маршрутов передачи сгенерированных кадров. Симуляция считается законченной, если в сети не происходит обмена ни одним сгенерированным кадром и в то же время отсутствуют записи о генерации кадров в будущем. Рассмотрим симуляцию сети на уровне обработки узлов. Следует заметить, что каждый узел в нашей реализации содержит следующие элементы:

- два потока (входной и выходной потоки), реализованные при помощи двусвязного списка;
- номер-идентификатор в сети;
- таблицу маршрутизации.

Во время текущего тайм-слота производится обработка узла, за которым закреплен данный тайм-слот. Обработка узла реализована в виде последовательности шагов:

1. *Обработка входящих кадров.*

На данном этапе происходит обработка кадров, так или иначе оказавшихся во входном потоке. В случае появления кадров, которые необходимо переслать другому узлу, производится перенос такого кадра в выходной поток. Если же во входном потоке встречается кадр-маяк, то его содержимое используется для обновления таблицы маршрутизации узла.

2. *Обновление таблицы маршрутизации.*

На данном этапе производится обновление таблицы маршрутизации узла, пенализация и сортировка записей в таблице.

3. *Генерация кадра-маяка.*

В процессе генерации создается кадр, содержащий информацию о наиболее выгодных маршрутах от данного узла. Данный кадр помещается в выходной поток. Поскольку этот кадр всегда рассматривается как первоочередной, он помещается в начало очереди.

4. *Обработка исходящих кадров.*

На данном этапе производится отправка кадров из выходного потока. В случае, если связность между узлом-отправителем и узлом-получателем в данный момент нарушена, кадр удаляется, иначе кадр попадает во входной поток узла-получателя.

Если узел в данный момент неактивен, то его обработка пропускается.

## Список литературы / References

- [1] “B.A.T.M.A.N. Advanced Documentation Overview”, 2014, <http://www.open-mesh.org/projects/batman-adv/wiki>.
- [2] Conner W.S. et al., “IEEE 802.11s Tutorial”, 2006, [http://www.ieee802.org/802\\_tutorials/06-November/802.11s\\_Tutorial\\_r5.pdf](http://www.ieee802.org/802_tutorials/06-November/802.11s_Tutorial_r5.pdf).
- [3] “Mobile Mesh Networks for Military, Defense and Public Safety”, 2015, <http://meshdynamics.com/military-mesh-networks.html>.
- [4] Shen W.L. et al., “Autonomous Mobile Mesh Networks”, *IEEE Transactions on Mobile Computing*, **13**:2 (2014), 364–376.
- [5] Boukerche A. et al., “Routing protocols in ad hoc networks: A survey”, *Computer Networks*, **55**:13 (2011), 3032–3080.
- [6] Broch J. et al., “A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols”, *Proc. of ACM/IEEE International Conference on Mobile Computing and Networking*, 1998, 85–97.
- [7] Servetto S.D., Barrenechea G., “Constrained Random Walks on Random Graphs: Routing Algorithms for Large Scale Wireless Sensor Networks”, *Proc. of 1st ACM International Workshop on Wireless Sensor Networks and Applications*, 2002, 12–21.
- [8] Nelakuditi S. et al., “Adaptive Proportional Routing: A Localized QoS Routing Approach”, *IEEE/ACM Transactions on Networking*, **10**:6 (2002), 790–804.
- [9] Baumann R. et al., “Routing in Large-Scale Wireless Mesh Networks Using Temperature Fields”, *IEEE Network*, 2008, 25–31.
- [10] Cormen T.H. et al., *Introduction to Algorithms*, MIT Press, 2009.
- [11] Perkins Ch. E. et al., “Performance Comparison of Two On-demand Routing protocols for Ad Hoc Networks”, *IEEE Personal Communications*, 2001, 16–28.
- [12] Awerbuch B. et al., “On the Survivability of Routing Protocols in Ad Hoc Wireless Networks”, *CERIAS Tech Report 2005–121*, 2005.
- [13] Nelakuditi S., Zhang Zh.-L., “On Selection of Paths for Multipath Routing”, *Proc. of IWQoS*, 2001, 170–186.
- [14] Nguyen Ph. H. et al., “Distributed routing algorithms to manage power flow in agent-based active distribution network”, *Innovative Smart Grid Technologies Conference Europe (ISGT Europe)*, IEEE PES, 2010, 1–7.
- [15] Brunsch T. et al., “Smoothed Analysis of the Successive Shortest Path Algorithm”, 2015, arXiv:1501.05493v1.
- [16] Spielman D.A., Teng Sh.-H., “Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time”, *Journal of the ACM*, **51**:3 (2004), 385–463.



DOI: 10.18255/1818-1015-2015-4-546-562

## Instrumental Supporting System for Developing and Analysis of Software-Defined Networks of Mobile Objects

Sokolov V. A., Korsakov S. V., Smirnov A. V., Bashkin V. A., Nikitin E. S.

*Received September 4, 2015*

This article describes the organization principles for wireless mesh-networks (software-defined networks of mobile objects). The emphasis is on the questions of getting effective routing algorithms for such networks. The mathematical model of the system is the standard transportation network. The key parameter of the routing system is the node reachability coefficient — the function depending on several basic and additional parameters (“mesh-factors”), which characterize the route between two network nodes. Each pair (arc, node) is juxtaposed to a composite parameter which characterizes the “reachability” of the node by the route which begins with this arc. The best (“shortest”) route between two nodes is the route with the maximum reachability coefficient. The rules of building and refreshing the routing tables by the network nodes are described. With the announcement from the neighbor the node gets the information about the connection energy and reliability, the announcement time of receipt, the absence of transitional nodes and also about the connection capability. On the basis of this information the node applies the penalization (decreasing the reachability coefficient) or the reward (increasing the reachability coefficient) to all routes through this neighbor node. The penalization / reward scheme has some separate aspects: 1. Penalization for the actuality of information. 2. Penalization / reward for the reliability of a node. 3. Penalization for the connection energy. 4. Penalization for the present connection capability. The simulator of the wireless mesh-network of mobile objects is written. It is based on the suggested heuristic algorithms. The description and characteristics of the simulator are stated in the article. The peculiarities of its program realization are also examined.

**Keywords:** mesh-network, network protocol, routing, penalization, heuristic algorithm, simulator

**For citation:** Sokolov V. A., Korsakov S. V., Smirnov A. V., Bashkin V. A., Nikitin E. S., "Instrumental Supporting System for Developing and Analysis of Software-Defined Networks of Mobile Objects", *Modeling and Analysis of Information Systems*, **22**:4 (2015), 546–562.

**On the authors:**

Sokolov Valery Anatolyevich, [orcid.org/0000-0003-1427-4937](https://orcid.org/0000-0003-1427-4937), Doctor, Professor, P.G. Demidov Yaroslavl State University, Sovetskaya str., 14, Yaroslavl, 150000, Russia, e-mail: [valery-sokolov@yandex.ru](mailto:valery-sokolov@yandex.ru)

Korsakov Stanislav Valentinovich, [orcid.org/0000-0003-2349-7980](https://orcid.org/0000-0003-2349-7980), CEO, NETSHe Lab Ltd, Belinskogo str., 28-75, Yaroslavl, 150047, Russia, Assistant Professor, P.G. Demidov Yaroslavl State University, Sovetskaya str., 14, Yaroslavl, 150000, Russia, e-mail: [sta@stasoft.net](mailto:sta@stasoft.net)

Smirnov Alexander Valeryevich, [orcid.org/0000-0002-0980-2507](https://orcid.org/0000-0002-0980-2507), PhD, Associate Professor, P.G. Demidov Yaroslavl State University, Sovetskaya str., 14, Yaroslavl, 150000, Russia, e-mail: [alexander\\_sm@mail.ru](mailto:alexander_sm@mail.ru)

Bashkin Vladimir Anatolyevich, [orcid.org/0000-0002-2534-1026](https://orcid.org/0000-0002-2534-1026), Doctor, Associate Professor, P.G. Demidov Yaroslavl State University, Sovetskaya str., 14, Yaroslavl, 150000, Russia, e-mail: [v\\_bashkin@mail.ru](mailto:v_bashkin@mail.ru)

Nikitin Evgeny Sergeevich, [orcid.org/0000-0002-2341-9950](https://orcid.org/0000-0002-2341-9950), student, P.G. Demidov Yaroslavl State University, Sovetskaya str., 14, Yaroslavl, 150000, Russia, e-mail: [nik.zhenya@gmail.com](mailto:nik.zhenya@gmail.com)

**Acknowledgments:**

This work was supported by the Russian Foundation for Basic Research under the Grant No 15-07-03038 A.