

Система адресации и безтабличной маршрутизации в сетях нового поколения

К.Н. Ловцов¹, С. Майхуб¹, С.В. Тимофеева², А.М. Сухов^{1,2}, М.С. Стенгач¹

¹Самарский национальный исследовательский университет им. академика С.П. Королева, Московское шоссе 34А, Самара, Россия, 443086

²Крымский федеральный университет им. В.И. Вернадского, Академика Вернадского 4, Симферополь, Россия, 295007

Аннотация. В статье предложен новый метод битовой иерархической адресации. Для записи составной части иерархического адреса используется два бита служебной информации, которая показывает число блоков стандартной длины. Для поиска маршрута в иерархических адресных средах предложено использовать комплексный метод жадного продвижения с двумя центрами иерархии.

1. Введение

Области применения современных инфокоммуникационных технологий расширяются и захватывают почти все сферы жизни. Концепция Интернет вещей [1] предполагает подключение к сети практически всех бытовых устройств, развиваются интернет медицина, дистанционное образование, производственные и транспортные технологии. В рамках умных городов также требуют множества новых сетевых подключений [2]. Для удовлетворения возрастающих потребностей в надежных сетевых подключениях развиваются новые телекоммуникационные технологии, прежде всего сети пятого поколения.

Для удовлетворения потребности в новых сетевых адресах была предложена шестая версия IP протокола. Казалось бы, эта версия полностью покрывает потребность в сетевых адресах и рассматривать альтернативные версии адресации больше не нужно. Однако для организации маршрутизации в компьютерных сетях одного адреса недостаточно. Для организации маршрутизации в локальной сети кроме IP адреса необходима еще и маска, которая показывает предельное количество сетевых устройств в локальном сегменте, а также начальный и конечный адрес соответствующего сегмента. Фактически для маршрутизации в локальной сети требуется два IP адреса.

Еще хуже дело обстоит с глобальной маршрутизацией. Задание любой маршрутизации за пределами локальной сети требует составления таблицы маршрутизации. Размер таблиц глобальной маршрутизации постоянно растет вместе с ростом числа автономных систем. В настоящее время размер этой таблицы превышает 650 тысяч префиксов. Неконтролируемый рост таблиц глобальной маршрутизации является основной проблемой современной системы сетевой адресации [3]. Поэтому исследования в области новых систем битовой адресации являются актуальными.

Поиски новой формы битовой записи сетевого адреса должны сопровождаться исследованиями в области маршрутизации. Цель этих исследований должна состоять в том, чтобы отказаться от

задания дополнительных переменных и таблиц маршрутизации. Идеально, чтобы сама система адресации позволяла бы находить маршрут.

Подобные исследования давно осуществляются в области самоорганизующихся сетей. Основное требование, чтобы маршрутизация определялась на основе локальной информации, доступной с текущего узла.

2. Альтернативная система сетевой адресации

Рассматриваемая нами в данной работе система адресации основана на принципах иерархии. Впервые такая система была предложена на базе гиперболического преобразования еще в 2007 году [4]. Гиперболическое преобразование отличается тем, что оно позволяет гарантированно находить маршрут для любой односвязной конфигурации сетевых устройств. Применение этого преобразования позволяет забыть о проблеме локального минимума.

Произвольная конфигурация сетевых устройств переводится внутрь круга единичного радиуса, а каждому узлу присваивается две координаты: радиус вектор и угол. Отличите их от обычных полярных координат состоит в том, что расстояние между двумя устройствами вычисляется в гиперболическом пространстве. Однако битовое представление гиперболических координат не так просто.

Параллельно с гиперболическими координатами в 2010 году были предложены иерархические координаты в виде последовательности натуральных чисел m_1, m_2, \dots, m_i , которые описывают положение сетевого устройства внутри иерархических уровней [5].

Было установлено [6], что обе вышеупомянутые системы адресации можно свести к единой форме, используя рекуррентные формулы для преобразования гиперболических радиуса и угла (r_n, φ_n) в последовательность натуральных чисел m_n

$$r_i = \frac{2^i - 1}{2^i}, \quad (1)$$

$$\varphi_i = 2\pi \left(1 - \frac{1}{2^{m_1-1}}\right) + \frac{2\pi}{2^{m_1}} \left(1 - \frac{1}{2^{m_2-1}}\right) + \dots + \frac{2\pi}{2^{m_1+m_2+\dots+m_i-1}} \left(1 - \frac{1}{2^{m_i-1}}\right). \quad (2)$$

Таким образом, необходимо записать в битовой форме иерархические координаты в виде последовательности натуральных чисел m_1, m_2, \dots, m_i . В принципе, для каждой составной части адреса в виде натурального числа m_n битовая форма записи тривиальна. Однако, представление разделительного знака в битовой форме является совсем не тривиальной задачей.

Составная часть IP адреса имеет стандартную длину в 8 бит, поэтому передача разделительного знака не требуется. В случае иерархического адреса составная часть и весь адрес не могут иметь постоянную длину. Длина составной части адреса используется в качестве сетевой маски, то есть определяет количество сетевых устройств в локальной сети. А полная длина иерархического адреса зависит от числа иерархических уровней.

Следует отметить, что в качестве разделительного знака можно использовать любую совокупность символов произвольной длины. При этом каждая составляющая адреса может быть записана в двоичной форме, из которой удалены разделительные знаки. Для удобства будем пользоваться разделительными знаками из 1 нулей, тогда в конце битового значения составляющей адреса должна стоять единица. Иначе распознать разделительный знак будет затруднительно. Такая форма записи фактически увеличивает длину разделительного знака на один бит.

Если использовать совокупность двух нулей 00, то основным недостатком будет слишком частая повторяемость разделительного знака из-за чего слишком много битовых комбинаций должно быть исключено из использования. Если составляющую иерархического адреса m_n записывать в битовой форме, то для четырехбитового числа останется только пять распознанных значения, а для пятибитовой формы только 8.

Поэтому представляется целесообразным использовать в качестве разделительного знака совокупность трех нулей 000, что приведет к росту числа бит служебной информации, но позволит сократить адресное пространство за счет составляющих адреса.

Ниже приведен пример перевода чисел адреса в пятибитовое пространство с учетом разделительной части адреса

Таблица 1. Перевод десятичного числа в битовую форму с учетом разделительного знака.

					Число в десятичном виде
0	0	0	0	1	не используется
0	0	0	1	1	не используется
0	0	1	0	1	1
0	0	1	1	1	2
0	1	0	0	1	3
0	1	0	1	1	4
0	1	0	1	1	5
0	1	1	1	1	6
1	0	0	0	1	не используется
1	0	0	1	1	7
1	0	0	1	1	8
1	0	1	0	1	9
1	0	1	1	1	10
1	1	0	0	1	11
1	1	0	1	1	12
1	1	1	1	1	13

То есть пятибитная запись числа позволяет присваивать 13 реальных адресов, при этом три числа не могут быть использованы. Можно построить следующую таблицу 2, в которой приведены данные о количестве неиспользуемых адресов в зависимости от числа бит, используемых для записи составляющей части адреса.

Таблица 2. Число возможных адресов для трехбитового разделительного знака.

Длина составной части адреса в битах	3	4	5	6	7	8	9
Число не использующихся адресов	0	1	3	7	15	31	63
Максимальное число возможных адресов	4	7	13	25	49	97	193

То есть для нахождения числа не используемых адресов H можно сформулировать следующее утверждение.

Утверждение 1. Если в состав разделительного знака между уровнями иерархии входит l нулей, а битовая длина составной части адреса равна $k(k \geq l - 2)$, то число неиспользуемых адресов H равно:

$$H = 2^{k-l} - 1. \tag{3}$$

Доказательство

Если $l \geq k$, то значимая часть составляющей адреса (то есть часть за исключением последней единицы) имеет меньшее число бит, чем разделительный знак и все 2^{k-1} битовых значения используются.

Если числа l и $k - l$ равны, то только первое битовое значение не будет использовано. При увеличении битовой формы на единицу (до $k = l + 2$), получим еще два неиспользуемых числа. На i ой итерации ($i = k - l + 1$) получим, что число неиспользуемых адресов должно повыситься на 2^i и станет равным:

$$H = 1 + 2^1 + \dots + 2^{k-l-1} = 2^{k-l} - 1, \tag{4}$$

что и требовалось доказать.

Для того, чтобы понять, где заканчивается сетевой адрес целиком установим, что в конце адреса должно располагаться два разделительных знака, то есть $2l$ нулей.

Следует отметить, что есть еще один способ передавать данные о том, где начинается и кончается битовая форма составной части адреса m_n . Первый способ, подробно рассмотренный выше, предполагает запись разделительного знака с помощью определенного битового набора. Этот набор должен быть исключен из битовой формы записи для составной части адреса, что приводит к значительному увеличению длины адреса.

Второй способ предполагает не использовать разделительный знак, а передавать битовую длину составных частей адреса вначале каждой составной части иерархии. В привычной системе IP маршрутизации эту роль играет параметр «маска». Это параметр показывает сколько бит в адресе остается неизменными, то есть задает родительский узел и описывает соответствующий кластер. В нашем случае мы должны передавать минимальное число бит p_n для соответствующего значения составной части иерархического адреса m_n , где $p_n = \lceil \log_2 m_n \rceil + 1$. Это приводит опять к резкому росту длины сетевого адреса.

Можно предложить комбинированный способ передачи информации о длине составной части адреса. Комбинированный способ будет совмещать достоинства фиксированных размеров блоков и переменной длины адресов. В этом случае битовая форма записи составной части адреса предполагает использование от одного до четырех блоков фиксированной длины. Длина такого блока может составить четыре бита, а перед составной частью адреса выделяется два служебных бита для определения длины составной части адреса (см. таблицу 3). То есть минимальная длина составной части адреса составит шесть бит, а максимальная длина – 18 бит, если элементарный блок задается четырьмя битами.

Таблица 3. Пример записи иерархического адреса.

	m_1		.	m_2		...	m_3		
1	1	$m_1(16bits)$	1	0	$m_2(12bits)$...	0	0	$m_i(4bits)$

Таким образом, каждая составная часть адреса увеличивается на два бита и не накладывает никаких ограничений на битовую форму записи адреса. Выше мы показали, что битовое представление с разделительным знаком из двух нулей позволяет использовать чуть больше четверти всех битовых комбинаций.

Теперь мы можем приступить к оценке битовой длины адреса, которая потребуется для нужд технологий IoT. Первым шагом оценим иерархическую глубину IoT. Для этого выделим из адресного пространства глобальную и локальную части. Глобальная часть адреса служит для привязки локального сегмента к глобальной сети. Сейчас эту роль выполняют IPv4 адреса, но их уже давно не хватает, поэтому глобальное адресное пространство было расширено на 2 байта (16 бит) с помощью технологии NAT.

Оценку количества уровней иерархического адресного пространства можно провести исходя из того факта, что число переходов (hop) команды ping ограничено 30. Выбор центрального узла в иерархии автономных систем может сократить это число почти вдвое. В качестве центрального узла целесообразно выбрать самую популярную точку обмена трафиком, расположенную в Нью Йорке, США. При этом достаточно ограничиться 20 уровнями иерархии, чтобы достичь каждого домохозяйства в мире или другого локального узла, к которому будут присоединяться локальные сети для удовлетворения местных потребностей.

Максимальная длина иерархического адреса при таком подходе может быть оценена в 120 бит с учетом разделительных знаков между уровнями иерархии. Это приблизительно столько же, сколько дает IPv6. Однако, если говорить о средней длине адреса, то он будет значительно короче. Заметим, что с удалением от центрального узла длина составной части адреса будет уменьшаться. Абсолютное большинство адресов будет иметь длину до 40 бит. Еще один

огромный плюс подобной маршрутизации – это отсутствие огромных маршрутизирующих таблиц.

Для локальной доставки информации число уровней иерархии может изменяться, как и число устройств на каждом уровне. Пока представляется достаточным 3 уровня иерархии по 8 бит на каждом уровне. Достоинство предложенного подхода состоит в том, что длина адреса не фиксирована и мы можем добавлять уровни иерархии по мере необходимости, так же, как и менять число устройств на каждом уровне.

3. Особенности маршрутизации в иерархических сообществах

В иерархических системах маршрут между двумя любыми узлами существует. Однако, стандартный маршрут, проложенный по иерархическому дереву, далеко не всегда является оптимальным по двум причинам:

- 1) Число участков маршрута по иерархическому дереву чаще всего превышает минимально возможное значение.
- 2) Каналы верхних уровней иерархии сильно перегружены из-за того, что значительная часть трафика от узлов нижних уровней иерархии вынуждена использовать верхние уровни иерархии.

Согласно принципу жадного продвижения пакет, находящийся у устройства $m_1, m_2 \dots m_n$ передается тому из соседних устройств $p_1, p_2 \dots p_l$, которое ближе всего к устройству назначения $g_1, g_2 \dots g_j$. При этом расстояние между устройствами p и g рассчитывается так:

$$L = l + j - 2K, \tag{5}$$

где K это число совпадающих уровней иерархии устройств p и g ($p_i = j_i, i = \overline{1, K}$).

Недостатки иерархической маршрутизации являются следствием того факта, что в иерархическое дерево не включено большое количество горизонтальных каналов, позволяющих устанавливать связь между устройствами одного или соседних уровней иерархии. Принцип жадного продвижения позволяет задействовать горизонтальные каналы связи, однако количество новых каналов связи не превышает 5% [7]. Поэтому требуются дальнейшие поиски новых подходов в иерархической маршрутизации.

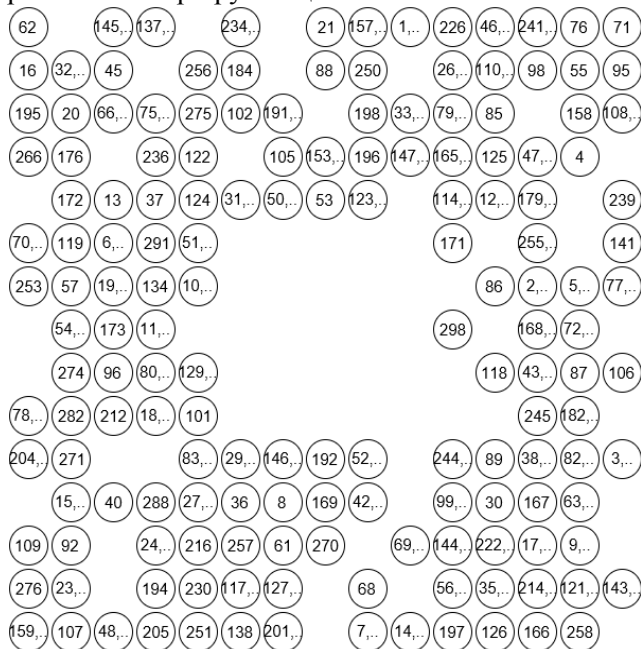


Рисунок 1. Конфигурация сетевых устройств.

В этой работе хотелось бы предложить метод смещения иерархической вершины. Суть его состоит в выборе двух независимых вершин иерархии и использовании принципа жадного

продвижения. То есть мы получаем две независимые системы иерархических адресов, причем эти адреса задаваться как для всех уровней иерархии, так и для небольшого количества верхних уровней. При решении вопроса о выборе маршрута подсчитываются два расстояния в двух иерархических системах согласно уравнению (5). В качестве продолжения маршрута выбирается тот соседний узел, у которого расстояние (5) наименьшее в любой из двух выбранных иерархических систем.

Попытаемся провести небольшое моделирование для данного метода с помощью специально написанного симулятора. Возьмем произвольную конфигурацию сетевых устройств, приведенную на рисунке 1.

Если выбрать узел 118 в качестве центрального узла иерархии и радиус связи, равный 2, то получил следующую иерархическую систему адресов (см. рисунок 2).

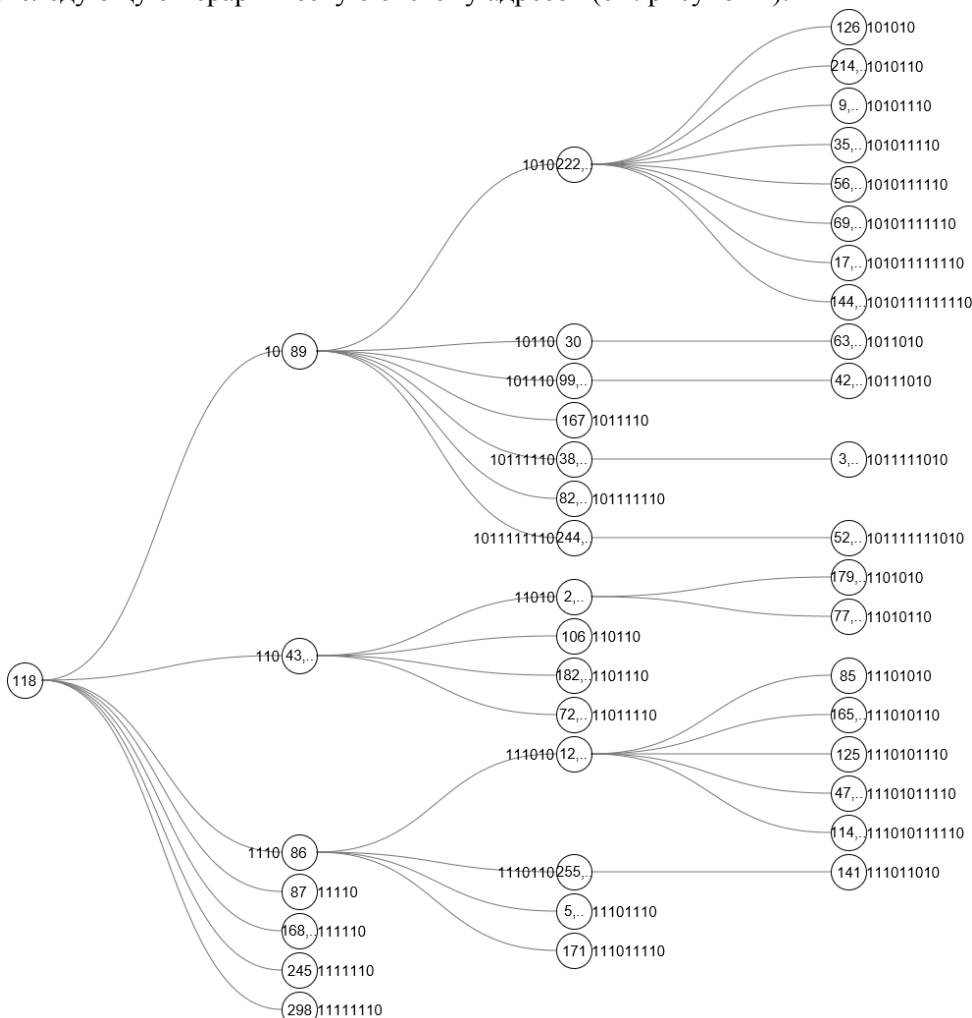


Рисунок 2. Основная иерархическая система адресов.

Кроме основной системы выберем также вспомогательную с центральным узлом иерархии 191. Эта иерархическая система адресов приведена на рисунке 3.

Попробуем теперь найти маршруты между несколькими парами устройств (55-271, 176-222, 282-121) используя три разных метода:

- 1) Метод жадного продвижения в иерархии устройства 118
- 2) Метод жадного продвижения в иерархии устройства 191
- 3) Комплексный метод жадного продвижения

Результаты наших поисков можно свести в следующую таблицу:

Таблица 4. Количество участков маршрута

Метод	55-221	176-222	282-121
1	12	10	11
2	11	11	15
3	11	10	8

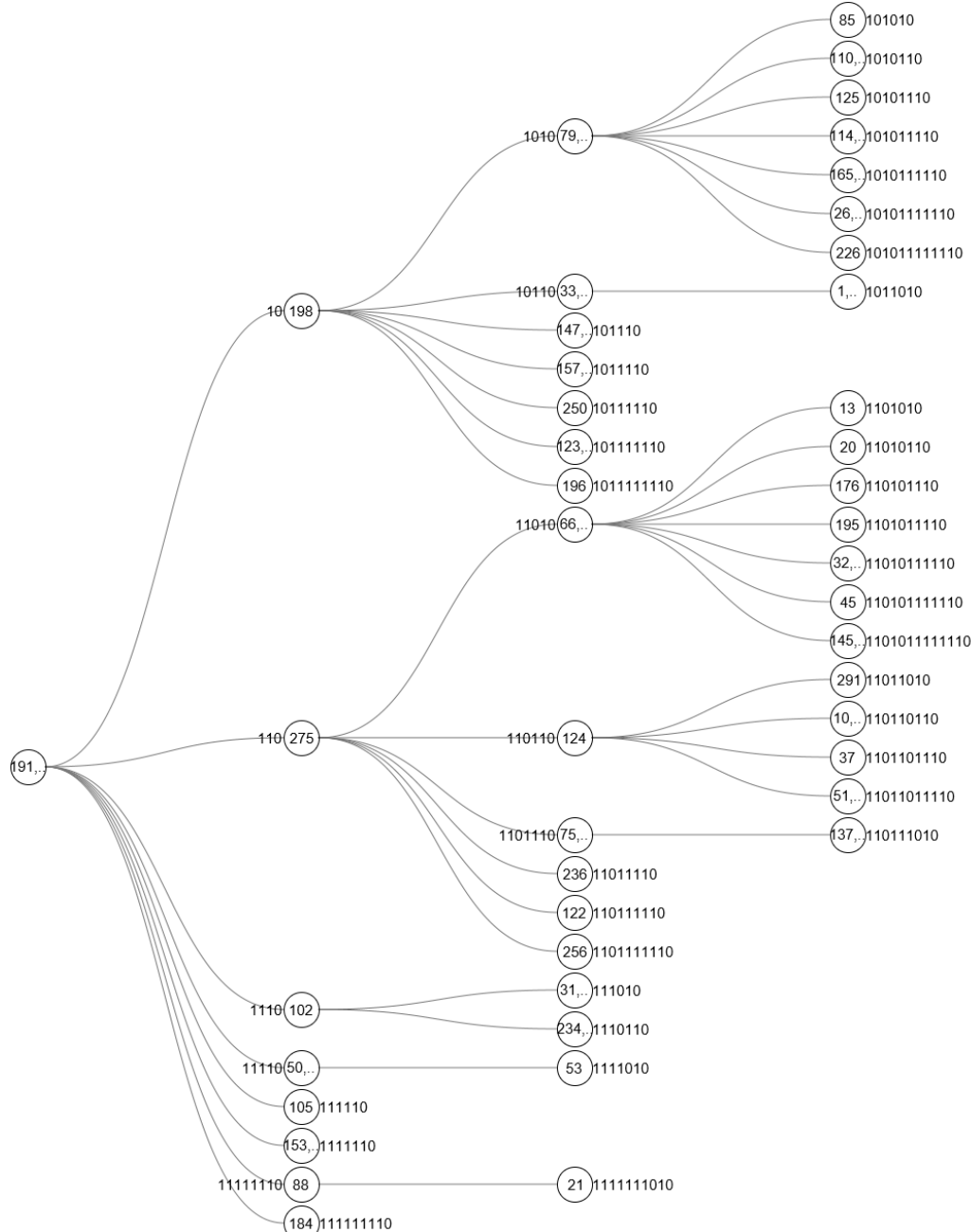


Рисунок 3. Вспомогательная иерархическая система адресов.

4. Выводы

В статье предложена новая система битовой адресации для иерархических адресов. Иерархический адрес представляет последовательность целых чисел, которую требуется перевести в битовую форму. Подробно рассмотрены три варианта. Они включают использование набора символов для разделительного знака, фиксированную длину каждой иерархической составляющей адреса, а также использование блоков стандартной длины и двух бит для задания длины каждой иерархической части адреса.

Показано, что набор символов для разделительной части адреса не может быть меньше 4 бит и существенно удлиняет адрес. Метод переменных блоков добавляет к каждой составной части адреса по два дополнительных бита, зато длина составной части иерархического адреса может быть уменьшена до 6 бит. Максимальный размер составной части может быть 18 бит, из которых 16 бит могут использоваться для записи натурального числа.

В работе обсуждается возможность безтабличной маршрутизации, в простейшем случае иерархические адреса позволяют отказаться от применения дополнительного параметра – сетевой маски. Опираясь на метод жадного продвижения, можно отказаться от таблиц маршрутизации и для конфигураций с большим количеством сетевых устройств. Однако, стандартный метод жадного продвижения не позволяет находить кратчайшие маршруты. Для повышения эффективности маршрутизации предложен и проанализирован метод комплексного жадного продвижения, в котором вводится дополнительная иерархическая структура. При этом следующим узлом маршрута становится соседний узел текущего устройства, которое ближе к устройству назначения в любом из двух уровней иерархии.

5. Литература

- [1] Gubbi, J. Internet of Things (IoT): A vision, architectural elements, and future directions // *Future generation computer systems*. – 2013. – Vol. 29(7). – P. 1645-1660.
- [2] Iraqi, Y. Resource management issues in future wireless multimedia networks / Y. Iraqi, R. Boutaba // *Journal of High Speed Networks*. – 2000. – Vol. 9(3, 4). – P. 231-260.
- [3] Bu, T. On characterizing BGP routing table growth / T. Bu, L. Gao, D. Towsley // *Computer Networks*. – 2004. – Vol. 45(1). – P. 45-54.
- [4] Kleinberg, R. Geographic routing using hyperbolic space // *26th IEEE International Conference on Computer Communications INFOCOM, 2007*. – P. 1902-1909.
- [5] Воробийенко, П.П. Алгоритм динамической адресации объектов телекоммуникационной сети / П.П. Воробийенко, В.И. Тихонов, И.В. Смирнов, У.И. Сопина // *Сборник научных трудов «Цифровые технологии»*. – 2010. – Т. 8.
- [6] Timofeeva S.V. Cluster approach for network routing in hierarchical systems // *Journal of Physics: Conference Series*. – 2018. – Vol. 1096. – P. 012080.
- [7] Chemodanov, D. REBATE: a REpulsive-BAsed Traffic Engineering for Dynamic Scale-Free Networks / D. Chemodanov, F. Esposito, H. Calyam, A. Sukhov // *Submitted to ICDCS, 2019*.

Благодарности

Работа выполнена при финансовой поддержке Минобрнауки РФ в рамках государственного задания № 2.974.2017/4.6.

Addressing system for non-tabular routing in new generation networks

K.N. Lovtsov¹, S. Maihub¹, S.V. Timofeeva², A.M. Sukhov^{1,2}, M.S. Stegach¹

¹Samara National Research University, Moskovskoe Shosse 34A, Samara, Russia, 443086

²V.I. Vernadsky Crimean Federal University, Prospekt Vernadskogo 4, Simferopol, Russia, 295007

Abstract. The article proposes a new method of bit hierarchical addressing. To record the constituent part of the hierarchical address, two bits of service information are used, which indicate the number of blocks of standard length. To search for a route in hierarchical address environments, it is proposed to use the complex method of greedy forwarding with two hierarchy centers.