Addressing system and routing without tables in new generation networks

K N Lovtsov¹, S Mayhoub¹, S V Timofeeva², A M Sukhov¹,², and M S Stengach¹

¹Samara National Research University, Moskovskoe Shosse 34, Samara, Russia, 443086
²V.I. Vernadsky Crimean Federal University, Prospekt Vernadskogo, 4, Simferopol, Russia, 295007

e-mail: sukhov@ssau.ru

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

1. Introduction

The fields of application of modern information and communication technologies are expanding and grasping almost all spheres of life. The concept of the Internet of Things [1] involves connecting to the network of almost all household devices, developing Internet medicine, distance education, production and transport technologies. Within smart cities, many new network connections are also required [2]. To meet the growing need for reliable network connections, new telecommunication technologies [3] are being developed, primarily the fifth-generation networks.

To realize the need for new network addresses, the IPv6 protocol was proposed. It would seem that this version completely covers the necessity for network addresses, and it is no longer necessary to consider alternative versions of addressing. However, for the organization of routing in computer networks, one address is not enough. For the organization of routing on the local network, in addition to the IP address, a mask is also required. It shows the maximum number of network devices in the local network, as well as the starting and ending addresses of the corresponding segment. In fact, two IP addresses are required for routing on the local network.

Even worse is global routing. Setting any routing outside the local network requires creating a routing table. The size of global routing tables is constantly expanding in conjunction with the growth of number of autonomous systems. Currently, the size of this table exceeds 650 thousand prefixes. The uncontrolled growth of global routing tables is the main problem of the modern network addressing system [4]. Therefore, research in the field of new bit addressing systems is relevant.

Searches for a new bit form of network address should be accompanied by routing research [5]. The purpose of these studies should be to abandon additional variables and routing tables. Ideally, the addressing system allows the user to independently find the route. Such studies have long been carried out in the field of self-organizing networks. The basic requirement is that routing be determined based on local information available from the current node.
2. Alternative network addressing system

The addressing system we are considering in this paper is based on the principles of hierarchy. For the first time such a system was proposed on the basis of a hyperbolic transformation back in 2007 [6]. Hyperbolic transformation allows us to guarantee the finding of a route for any single-connected configuration of network devices. The use of this transformation allows to forget about the problem of a local minimum.

An arbitrary configuration of network devices is transferred inside a circle of a single radius, and two coordinates are assigned to each node: the radius vector and the angle. What distinguishes them from ordinary polar coordinates is that the distance between the two devices is calculated in hyperbolic space. However, the bit representation of hyperbolic coordinates is not so simple.

In 2010, hierarchical coordinates were proposed as a sequence of natural numbers \( m_1, m_2, \ldots, m_i \) in parallel with the hyperbolic coordinates. They describe the position of the network device inside the hierarchical levels [7].

It was established [8] that both the above-mentioned addressing systems can be reduced to a united form, using recurrent formulas to convert hyperbolic radius and angle \((r_n, \varphi_n)\) into a sequence of natural numbers \( m_n \).

\[
\begin{align*}
  r_i &= \frac{2^i - 1}{2^i}, \\
  \varphi_i &= 2\pi \left(1 - \frac{1}{2^{m_1-1}}\right) + \frac{2\pi}{2^{m_1}} \left(1 - \frac{1}{2^{m_2-1}}\right) + \ldots + \frac{2\pi}{2^{m_1+m_2+\ldots+m_i-1}} \left(1 - \frac{1}{2^{m_i-1}}\right),
\end{align*}
\]

Thus, it is necessary to write the hierarchical coordinates in bit form as a sequence of natural numbers \( m_1, m_2, \ldots, m_i \). In principle, the bit form for each component of the address in the form of a natural number \( m_n \) is trivial. However, representing a separator in bit form is not a trivial task at all.

Each part of the IP address has a standard length of 8 bits, so the transfer of the separator sign is not required. In the case of a hierarchical address, the component and the entire address cannot have a constant length. The length of the component part of the address is used as a network mask; that is, it determines the number of network devices in the local network. The total length of the hierarchical address then depends on the number of hierarchical levels.

It should be noted that any combination of characters of arbitrary length can be used as a separator sign. In addition, each component of the address can be written in binary form, from which the separators are removed. For convenience, we will use the separator sign of \( l \) zeros and ones at the end (0...01). Otherwise, it will be difficult to recognize the beginning of the separator sign. This form of recording increases the length of the separation sign by one bit.

If we use a set of two zeros 001, then the main drawback would be a frequent recurrence of the separator sign. Because of this, too many bit combinations should be excluded from use. If the component of the hierarchical address \( m_n \) is written in bit form, then for the four-bit number only five recognized values remain, and the five-bit form is only 8.

Thus, it seems appropriate to use a set of three zeros 0001 as a separator sign. This will lead to an increase in the number of bits of service information but will reduce the address space due to the address components.

Below is an example of the translation of the address component into a five-bit space, considering the separation part of the address.

That is, a five-bit record allow to assign 13 real numbers, and three numbers cannot be used (see Table 1). We can build the following Table 2, in which data on the number of unused
Table 1. Conversion of a decimal number to a bit form, considering the separator sign.

| Number in decimal |   |
|-------------------|---|
| 0 0 0 0 1         | not used |
| 0 0 0 1 1         | not used |
| 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 1 0 1         | 5       |
| 0 1 1 1 1         | 6       |
| 1 0 0 0 1         | not used |
| 1 0 0 1 1         | 7       |
| 1 0 1 0 1         | 8       |
| 1 0 1 1 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      |

addresses is given depending on the number of bits used to write the component part of the address.

Table 2. The number of possible addresses for the three-bit separator character.

| The length of the component part of the address in bits | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|--------------------------------------------------------|---|---|---|---|---|---|---|
| Number of unused addresses                             | 0 | 1 | 3 | 7 | 15| 31| 63|
| Maximum number of possible addresses                   | 4 | 7 | 13| 25| 49| 97|193|

That is, to find the number of unused addresses, \( H \), we can formulate the following statement:

**Statement 1.** If the separator sign between the levels of the hierarchy includes \( l \) zeros, and the bit length of the address component is \( k \) \((k \geq l - 2)\), then the number of unused addresses \( H \) is equal to:

\[
H = 2^{k-l} - 1, \tag{3}
\]

**Proof**

If \( l \geq k \), then the significant part of the component address (that is, the part except for the last one) has fewer bits than the separator and all \( 2^{k-1} \) bit values are used.

If the numbers \( l \) and \( k - 1 \) are equal, then only the first bit value will not be used. If the numbers \( l \) and \( k - 1 \) are equal, then only the first bit value will not be used. By increasing the bit form by one (up to \( k = l + 2 \)), we get two more unused numbers. At the \( i \)th iteration \((i = k - l + 1)\), we find that the number of unused addresses should increase by \( 2^i \) and become equal to

\[
H = 1 + 2^1 + \ldots + 2^{k-l-1} = 2^{k-l} - 1, \tag{4}
\]

Q.E.D.
In order to understand where the network address ends, we completely establish that at the end of the address there should be two separators (that is, 2l zeros).

It should be noted that there is another way to transmit data regarding where the bit form of the component part of the $m_n$ address begins and ends. The first method, discussed in detail above, involves writing a separator sign using a specific bit set. This set should be excluded from the bit record form for the component part of the address, which leads to a significant increase in the address length.

The second method assumes not to use a separator sign, but to transmit the bit length of the component parts of the address at the beginning of each component part of the hierarchy. In the usual IP routing system, the mask parameter plays this role. This parameter shows how many bits in the address remain unchanged (that is, it sets the parent node and describes the corresponding cluster). In our case, we must transmit the minimum number of bits $p_n$ for the corresponding value of the component part of the hierarchical address $m_n$, where $p_n = \left[ \log_2 m_n \right] + 1$. This again leads to a sharp increase in the length of the network address.

It is possible to propose a combined method of transmitting information about the length of an address component. The combined method will combine the advantages of fixed block sizes and variable length addresses. In this case, the bit form of the recording of the component part involves the use of one to four blocks of fixed length. The length of such a block can be four bits, and before the component part of the address, two service bits are allocated to determine this length (see Table 3). That is, the minimum length of the constituent part of the address is six bits, and the maximum length is 18 bits if the elementary block is specified by four bits.

If we take the length of the standard block as being equal to 3 bits, then the maximum size of the component part of the address is 12 bits. This is not enough, as now there are autonomous systems with about 4 thousand neighbors.

| Table 3. Example of a hierarchical address record. |
|--------------------------------------------------|
| $m_1$ 1 1 $m_1$ (16 bits) 1 0 $m_2$ (12 bits) ... 0 0 $m_3$ (4 bits) |

Thus, each component of the address is increased by two bits and does not impose any restrictions on the bit form of the address. Above, we have shown that a bit representation with a two-zero separator allows the use of little more than a quarter of all bit combinations.

Now we can proceed to estimating the bit length of the address that will be required for the needs of IoT technologies. The first step is to evaluate the hierarchical depth IoT. To do this, select the global and local parts from the address space. The global part of the address is used to bind the local segment to the global network. IPv4 addresses usually fulfill this role, but they have long been lacking, so the global address space was extended by 2 bytes (16 bits) using NAT technology.

Evaluation of the number of levels of the hierarchical address space can be made since the number of hops of the ping utility is limited to 30. The choice of the central node in the hierarchy of autonomous systems can reduce this number by almost half. As a central node, it is advisable to choose the most popular traffic exchange point located in New York, USA. At the same time, it suffices to limit ourselves to 20 levels of hierarchy in order to reach every household in the world or another local node to which local networks will join in order to meet local needs.

The maximum length of the hierarchical address with this approach can be estimated at 120 bits, considering the separator sign between levels of the hierarchy. This is about the same as IPv6; however, if we talk about the average length of the address, it will be much shorter. Note
that, with an increasing distance from the central node, the length of the address component will decrease. Many addresses will be up to 40 bits long. Another huge plus of such routing is the absence of huge routing tables.

For local information delivery, the number of hierarchy levels may vary, as well as the number of devices at each level. So far, 3 levels of hierarchy of 8 bits at each level are enough. The advantage of the proposed approach is that the length of the address is not fixed and we can add hierarchy levels as necessary, as well as change the number of devices at each level.

3. Features of routing in hierarchical communities

In hierarchical systems, a route between any two nodes exists. However, the standard route laid on a hierarchical tree is not always the optimal one for two reasons:

- the number of route hops along a hierarchical tree most often exceeds the minimum possible value
- channels of the upper hierarchy levels are heavily overloaded because a significant part of the traffic from the nodes of the lower hierarchy levels is forced to use the upper levels of the hierarchy

According to the principle of greedy forwarding, the packet located at the device \( m_1.m_2...m_n \) is transmitted to that of the neighboring devices \( p_1.p_2...p_l \), which is closest to the destination device \( g_1.g_2...g_j \). The distance between the devices \( p \) and \( g \) is calculated as

\[ L = l + j - 2K, \quad (5) \]

where \( K \) is the number of matching levels of the device hierarchy \( p \) and \( g \) (\( p_i = g_i \ i = (1, K) \)).

The disadvantages of hierarchical routing are a consequence of the fact that many horizontal channels are not included in the hierarchical tree. These channels allow for the establishment of communication between devices of the same or neighboring levels of the hierarchy. The principle of greedy forwarding allows the use of horizontal communication channels, but the number of new communication channels does not exceed 5% [9]. Therefore, further searches for new approaches in hierarchical routing are required.

In this paper, we would like to suggest a method for shifting the hierarchical vertex. Its essence lies in the choice of two independent vertices of the hierarchy and the use of the principle of greedy forwarding. That is, we get two independent systems of hierarchical addresses, and these addresses can be set for all levels of the hierarchy, as well as for a small number of upper levels. When deciding on the choice of route, two distances are calculated in two hierarchical systems according to equation (5). As a continuation of the route, the neighboring node is selected, whose distance (5) is the shortest in any of the two selected hierarchical systems.

We will try to conduct a small simulation for this method using a specially written simulator. Take an arbitrary configuration of network devices, shown in Figure 1.

If node 118 is selected as the central node, then we obtain the following hierarchical system of addresses (see Figure 2)

In addition to the main system, we also choose an auxiliary one with the central node of the hierarchy 191. This hierarchical system of addresses is shown in Figure 3.

Now let’s try to find routes between several pairs of devices (55-271, 176-222, 282-121) using three different methods:

- The method of greedy forwarding the hierarchy of the device 118
- The method of greedy forwarding in the hierarchy of the device 191
- The comprehensive greedy forwarding method

The results of our searches can be summarized in the following table.
Figure 1. Network device configuration.

Figure 2. Basic hierarchical address system.
4. Conclusions
The article proposes a new bit addressing system for hierarchical addresses. A hierarchical address represents a sequence of natural numbers that we want to convert to bit form. Three options are considered in detail. These include the use of a character set for the separator sign, the fixed length of each hierarchical component of the address, and the use of blocks of standard length and two bits to set the length of each hierarchical part of the address.

It is shown that the character set for the separation sign cannot be less than 4 bits and significantly lengthens the address. The variable block method adds two bits to each addresses components, but their length can be reduced to 6 bits. The maximum size of a component can be 18 bits, of which 16 bits can be used to record a natural number.

This paper discusses the possibility of routing without tables. In the simplest case, hierarchical addresses allow us to abandon the use of an additional parameter, the network mask. Based on the method of greedy forwarding, we can abandon the routing tables for configurations.
with a large number of network devices. However, the standard method of greedy advancement does not allow for finding the shortest routes. To improve the efficiency of routing, a method of comprehensive greedy forwarding was proposed and analyzed in which an additional hierarchical structure was introduced. At the same time, the next node of the route becomes the neighboring node of the current device, which is closer to the destination device in any of the two levels of the hierarchy.

5. References
[1] Gubbi J, Buyya R, Marusic S and Palaniswami M 2013 Internet of Things (IoT): A vision, architectural elements, and future directions Future generation computer systems 29(7) 1645-1660
[2] Iraqi Y and Boutaba R 2000 Resource management issues in future wireless multimedia networks Journal of High Speed Networks 9(3, 4) 231-260
[3] Agafonov A and Myasnikov V 2018 Numerical route reservation method in the geoinformatic task of autonomous vehicle routing Computer Optics 42(5) 912-920 DOI: 10.18287/2412-6179-2018-42-5-912-920
[4] Bu T, Gao L. and Towsley D 2002 On characterizing BGP routing table growth Global Telecommunications Conference 3 2185-2189
[5] Balliu A and Fraigniaud P 2018 Certification of compact low-stretch routing schemes The Computer Journal 62(5) 730-746
[6] Kleinberg, R 2007 Geographic routing using hyperbolic space 26th IEEE International Conference on Computer Communications 1902-1909
[7] Vorobienko P, Tikhonov V, Smirnov I and Sopina U 2010 Dynamic addressing algorithm for the telecommunication network object Tsifrovye technologii 8 11-18
[8] Timofeeva S, Sarkisyan D, Alzinskaya N and Sukhov A 2018 Cluster approach for network routing in hierarchical systems Journal of Physics: Conference Series 1096(1) 012080
[9] Chemodanov D, Esposito F, Calyam P and Sukhov A 2019 REBATE: a REpulsive-BAsed Traffic Engineering for Dynamic Scale-Free Networks (in press)

Acknowledgements
The work was done with the financial support of the Ministry of Science and Higher Education of the Russian Federation within the framework of state task No. 2.974.2017/4.6.