Enhanced ZigBee Tree Addressing for Flexible Network Topologies

Uttam Kumar Roy[1], Parama Bhaumik[2]
Dept. of IT, Jadavpur University
Block-LB/8, Sector-III, Salt Lake City
Kolkata, India
royuttam@gmail.com[1], bhaumikparama@gmail.com[2]

***Abstract*—Since, IEEE 802.15.4 complaint devices are resource constrained, traditional heavyweight routing algorithms are not suitable. Although, ZigBee provided an addressing scheme and a table-free Tree routing algorithm for such devices, it has several limitations.**

In this paper, we have proposed an extended version of ZigBee addressing scheme. The proposed scheme has all the properties of original addressing scheme and hence the same table-free routing algorithm can be applied. In addition, it supports asymmetric tree networks which original scheme cannot cope up with. It does not also use the parameter, *maximum network depth* (L_m), which is often very difficult to choose before forming the network. Simulation shows that proposed addressing scheme works seamlessly with existing ZigBee tree routing.

Keywords—ZigBee; Tree Networks; Address; Routing

I. INTRODUCTION

IEEE 802.15.4 [1] is novel standard to promote pervasive computing. Since it suggests low cost devices; there by resource constrained, ZigBee [2] provided a simple, lightweight, table-free routing algorithm together with an addressing scheme. The strength of this algorithm is that it uses only some mathematical equations to take routing decisions.

The performance of this standard for **W**ireless **P**ersonal **A**rea **N**etworks (WPANs) is now unquestionable. Since it is a low cost technology, researchers are now trying to apply it beyond WPANs, where networks are often asymmetric. However, ZigBee address assignment scheme cannot support asymmetric networks. This is because all devices in a tree network use same values of network parameters C_m, R_m and L_m and hence same C_{skip}. Moreover, the value of L_m (maximum network depth) is often very difficult to select before actually forming the network.

We have already addressed those issues in [3, 4, 5, 6, 7, 8] and provided several potential solutions. In this paper, we have proposed a new recursive expression for C_{skip}. We have shown that it actually degenerates to original C_{skip} and hence original table-free routing algorithm can be applied seamlessly. This implies that the proposed scheme has all advantages the original ZigBee scheme had. Moreover, the expression is free from L_m and hence there is no hazard to select it prior to the network is formed. It also accepts different values of C_m and R_m at different network depth and hence supports asymmetric networks.

II. RELATED WORK

Wireless networks have rapidly gained popularity since their introduction in 1970s. However, an investigation into low-cost, low-rate, low-power PAN is relatively new.

Our investigation [9] revealed that the routing protocols in ZigBee networks is essentially a combination of tree routing and AODV with several optimizations done in consideration of stationary wireless topologies. In [9], we also developed an NS2 simulator for ZigBee PAN to analyze and optimize both tree and mesh routing in a ZigBee network. We performed several experiments to study its various features, including: (i) the amount of routing packets generated, (ii) the packet-delivery ratio, (iii) the number of hops taken by application packets from source to destination. The packet-delivery ratio is found to be excellent, although the number of hops taken is not least always. We make several interesting optimizations in the implementation to control the huge burst of control packets produced for route discovery.

In [3], we provided a unified address borrowing scheme which can be easily applied to grow the network beyond 16 hops and overcome the address exhaustion problem by borrowing address. A routing algorithm based on mobile IP, was also proposed.

In [4], we extended the Tree routing proposed by ZigBee for the networks to be harsh and asymmetric.

In [5], we provided a unified multi-channel routing scheme which can be easily applied to tree network so that the network can be used in dynamic application space and overcome the link disruption problem by multi channeling but without adding any extra overhead of having a routing table.

Performance evaluation of routing protocols is a challenging issue and requires a well-planned, reliable and if possible standardized test bed. The VINT project is a landmark in the arena of network simulation research. The concept of split-level programming, its merits and applicability are discussed in [10].

In [11], authors explored the complex behaviour of a large number of low-power sensor nodes. Energy aware operation of wireless devices is the dominant theme in [12, 11]

Routing in wireless networks has been a fascinating topic of research for long. These routing protocols deal with the challenges of wireless networks, namely low-bandwidth, high error-rates and often energy and memory constraints. They are either table-driven (e.g. DSDV [13], WRP [14]) or source-initiated, that is demand-driven (e.g. DSR [15], AODV [16], [13]). A comprehensive survey of these protocols has been done in [18].

There is relatively scant literature on 802.15.4/ZigBee although its applications have been discussed in [Comprehensive, 19]. Authors provide one of the first studies of the MAC sub layer while the recent paper [1] is a comprehensive performance evaluation of 802.15.4.

III. LIMITATIONS OF ZIGBEE ADDRESS ASSIGNMENT

ZigBee distributed address assignment scheme has several limitations. We already have addressed those issues in [3, 4, 5, 6] and provided several potential solutions using different methodologies. Let us quickly review those problems so that we can appreciate the need of a new addressing scheme.

Given the following network parameters:

C_m = max. no. of children of a device
R_m = max. no. of router-capable-children of a device
L_m = max. depth in the network

ZigBee provided a function $C_{skip}(d)$, which is essentially the size of the address sub-block distributed by each parent at depth d to each of its router-capable child devices, as follows:

$$C_{skip}(d) = \begin{cases} 0 & : d = L_m \\ 1 + C_m(L_m - d - 1) & : R_m = 1, d \neq L_m \\ \dfrac{1 + C_m - R_m - C_m R_m^{L_m-d-1}}{1 - R_m} & : R_m \neq 1, d \neq L_m \end{cases} \quad (1)$$

ZigBee specified that the "*coordinator determines*" important network L_m. However, it remains silent about how to determine it. Since, before forming the network, we have very little or no idea about how large a network will be it is almost impossible to find favourably good value of this parameter.

Note that improper value of C_m and R_m may result wastage of network addresses. This happens since all routers use same C_m and R_m and once set they remain unchanged. Given those parameters, essentially a virtual symmetric tree network skeleton gets created. Devices may sit in an empty location and use its address. So, it may happen that a device can not join at a desired position even if addresses are available.

The size of unused address sub-block may be very large for larger value of L_m. Consider C_m=4, R_m=2 and L_m=14. The number of addresses, a router R at depth 1 can distribute to each of its 2 children routers is $C_{skip}(1)$ [=16381]. Because R can have maximum 2 (two) end devices and 2 (two) routers children, total number of addresses it can distribute is 2+2*16381(=32764). If R has no children (this may happen when no device is within the transmission range of R), large number (32764) of addresses that R could have distributed would remain unused. This implies straight wastage of approximately 50% of the total 2^{16} (=65536, for 16-bit address) possible addresses.

This situation becomes even worse if we attempt to form networks in mines, road-side etc. For C_m=4 and R_m=3, it can be shown [see Eqn. 1] that, the maximum value of L_m is 9. This means no device can exist at depth beyond 9.

Another major problem is that the addressing scheme essentially limits network depth. $C_{skip}(0)$ is the address sub-block being distributed by coordinator (at depth 0) to each of its R_m routers [Eqn 1]. So total number of addresses distributed to all of its routers is $R_m C_{skip}(0)$. Then, total possible addresses is the sum of $R_m C_{skip}(0)$ and number of end devices E_m(=C_m-R_m) of the coordinator and 1(one) for its own address. Because network address is a 16-bit address, assuming $R_m>1$, the following equality must be valid:

$$R_m C_{skip}(0) + (C_m - R_m) + 1 \leq 2^{16}$$

$$\Rightarrow R_m \frac{1 + C_m - R_m - C_m R_m^{L_m - 1}}{1 - R_m} + (C_m - R_m) + 1 \leq 2^{16}$$

$$\Rightarrow \frac{1 + C_m - R_m - C_m R_m^{L_m}}{1 - R_m} \leq 2^{16}$$

$$\Rightarrow L_m \leq \frac{\log_{10}\left(\dfrac{(2^{16} - 1)(R_m - 1)}{C_m} + 1 \right)}{\log_{10} R_m} \quad (1)$$

For example if $C_m = 8$ and $R_m = 4$, maximum possible depth $L_m = 7$ only. This proves the impossibility of creating a very long network.

IV. PROPOSED ADDRESSING SCHEME

Note that the concept of C_{skip} is fundamental in tree routing. It helps us to take routing decisions using only some mathematical/logical calculations. Let's try to remember the concept.

Each router gets a block of addresses from its parent. The root (coordinator) gets all possible addresses. The size of this block is called C_{skip}. The router, in turn, assigns a sub-block of addresses (given to it) to each of its router children. It also assigns addresses to its end-device children. This address assignment is done in a predictable manner. This process continues until a router does not have enough addresses to distribute further.

Since addresses are distributed predefined manner, each router knows who its descendants are. Accordingly, making a routing decision becomes possible without consulting any routing tables.

However, the value of C_{skip} should be determined such that we can have flexible network topologies. As mentioned in the previous section, that ZigBee's C_{skip} uses same C_m and R_m. This means every router is capable to have same number of children. In the following section, we shall device a new expression for C_{skip} (we call it C'_{skip}) that may accept different C_m and R_m values at different level.

A. Proposed Cskip

We define the following parameters:

D^d = A device at dept d.
C_m^d = max. no. of children of D^d.
R_m^d= max. no. of router-capable-children of D^d.
$E_m^d[= (C_m^d - R_m^d)]$ = max. no. of end-device children of D^d.
b = no. of bits in network address

For b-bit address, the coordinator (at level 0) has 2^b number of addresses to use. It keeps one address for itself and E_m^0 addresses for its end-device children. It then distributes rest $2^b - (E_m^0 + 1)$ addresses equitably among its R_m^0 router-capable children. So, the number of addresses, each router at level 1 obtains is

$$C'_{skip}(0) = \left\lfloor \frac{2^b - (E_m^0 + 1)}{R_m^0} \right\rfloor \qquad d = 0, R_m^0 \neq 0, 2^b \geq (E_m^0 + 1)$$

Where $\lfloor n \rfloor$ is the largest integer less than or equal to n. A router at level 1 now has $C'_{skip}(0)$ addresses to use. It keeps one address for itself and E_m^1 addresses for its end-device children and distributes rest $C'_{skip}(0) - (E_m^1 + 1)$ addresses equitably among its R_m^1 router-capable children. So, the number of addresses, each router at level 2 obtains is

$$C'_{skip}(1) = \left\lfloor \frac{C'_{skip}(0) - (E_m^1 + 1)}{R_m^1} \right\rfloor$$

$$: d > 0, R_m^d \neq 0, C'_{skip}(d-1) \geq (E_m^d + 1)$$

This process continues until the value C'_{skip} becomes less than 1. If the value before applying floor function is less than 1, it indicates that the router cannot distribute any more addresses from the poll of addresses assigned to it. In that case, the value of $C'_{skip}(d)$ is set to 0 to make it compatible with original convention. So, the function $C'_{skip}(d)$ can recursively be expressed as follows:

$$C'_{skip}(d) \begin{cases} \left\lfloor \dfrac{2^b - (E_m^0 + 1)}{R_m^0} \right\rfloor : d = 0, R_m^0 \neq 0, 2^b \geq (E_m^0 + 1) \\ \left\lfloor \dfrac{C'_{skip}(d-1) - (E_m^d + 1)}{R_m^d} \right\rfloor \\ \quad : d > 0, R_m^d \neq 0, C'_{skip}(d-1) \geq (E_m^d + 1) \\ 0 \qquad \text{for all other cases} \end{cases} \tag{2}$$

The interesting part of this expression is that it is free from L_m. Moreover, C'_{skip} is calculated per level basis. This implies that routers at different level may have different number of C_m, R_m values. Fig. 1 pictorially illustrates this.

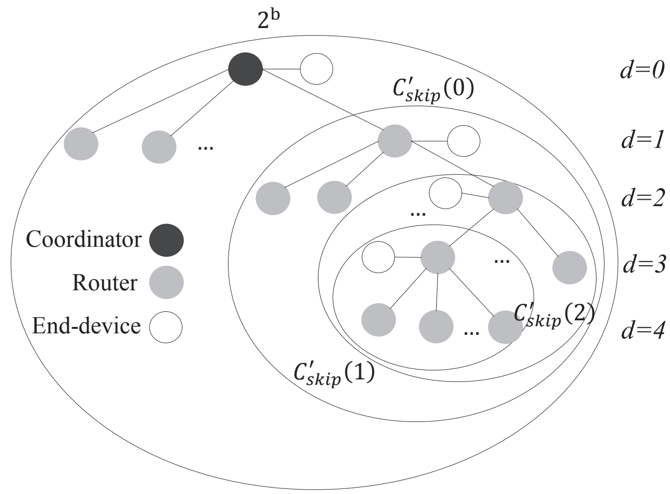


Fig. 1. Calculation of C'_{skip}

The coordinator gets all possible addresses which is shown using the outer most ellipse. For b-bit address, the size of this address block is 2^b. Coordinator then distributes $C'_{skip}(0)$ addresses to each of its router child at level 1 using rules as described in the next section. The same procedure is followed by the subsequent routers.

The function C'_{skip} works same way as original one. In fact, for some cases C'_{skip} degenerates to C_{skip}. TABLE I illustrates this for various set of parameters:

TABLE I. $C_{skip}(d)$ AND $C'_{skip}(d)$

d	b=8, L_m=6, C_m=4, R_m=2		b=9, L_m=4, C_m=6, R_m=4		b=10, L_m=5, C_m=5, R_m=3		b=16, L_m=7, C_m=8, R_m=4	
	$C_{skip}(d)$	$C'_{skip}(d)$	$C_{skip}(d)$	$C'_{skip}(d)$	$C_{skip}(d)$	$C'_{skip}(d)$	$C_{skip}(d)$	$C'_{skip}(d)$
0	125	126	127	127	201	340	10921	16382
1	61	61	31	31	66	112	2729	4094
2	29	29	7	7	21	36	681	1022
3	13	13	1	1	6	11	169	254
4	5	5	0	0	1	2	41	62
5	1	1			0	0	9	14
6	0	0					1	2
7							0	0

B. Address Assignment

Note that a router assigns network addresses to its children using some rules which allow it to make routing decision later. The rules are as follows:

The coordinator has address 0. A router (including coordinator) X having address A_x at depth d assigns address A_n to its n^{th} child at depth $d+1$ as follows:

FOR ROUTER-CAPABLE CHILD:

$$A_n = A_x + (n-1) \times C'_{skip}(d) + 1 \quad : 1 \leq n \leq R_m^d \tag{3}$$

FOR END-DEVICE CHILD:

$$A_n = A_x + R_m^d \times C'_{skip}(d) + n \quad : 1 \leq n \leq E_m^d \tag{4}$$

A careful inspection of the above equations helps us to understand that a router X assigns addresses to its router children from the beginning of the address block assigned to it. The first router-capable child gets address A_x+1 [see (3)] and subsequent router-capable children get addresses separated (skipped) by $C'_{skip}(d)$. Hence the suffix 'skip' was used in C_{skip}.

The end-devices, in turn, get addresses linearly from the end of the address sub-block. In fact, they get addresses when all the addresses for router children (i.e. $R_m^d \times C'_{skip}(d)$) are exhausted. They get addresses starting from $A_x + R_m^d \times C'_{skip}(d) + 1$ [see (4)] separated by 1. An example of such address assignment is shown in Fig. 2.

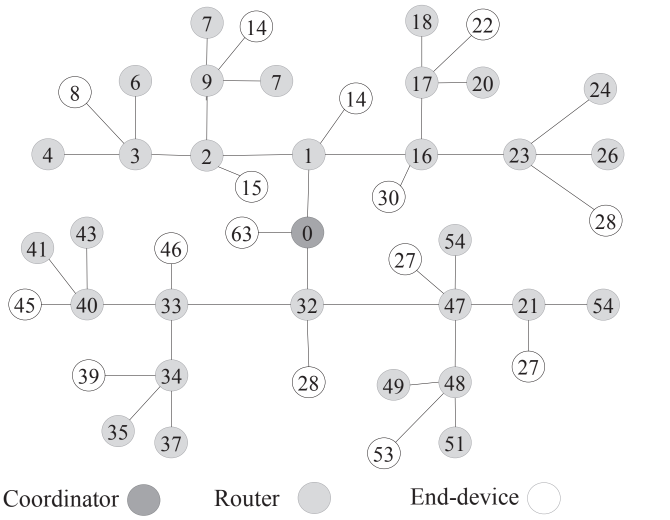


Fig. 2. Addresses assignment, b=6, $C_m^d = 3, R_m^d = 2 \quad \forall d$, $C'_{skip}(\mathbf{0}) = \mathbf{31}$, $C'_{skip}(\mathbf{1}) = \mathbf{14}$, $C'_{skip}(\mathbf{2}) = \mathbf{6}$, $C'_{skip}(\mathbf{3}) = \mathbf{2}$, $C'_{skip}(\mathbf{4}) = \mathbf{0}$

Note that, ZigBee uses 16-bit short and 64-bit long network address. For simplicity of the figure, we have considered 6-bit network address. The scheme works for arbitrary values of b.

The values of $C_m^d = 3, R_m^d$ need not be identical for all levels. They can be set depending upon the number of devices available at that level. If a router at a level cannot accommodate those children, the C'_{skip} automatically becomes 0. Note that the value 0 of C_{skip} indicates that the router can not accept any children further. One such example is shown in TABLE II

TABLE II. VARIABLE NETWORK PARAMETERS

$\boldsymbol{b=5}$				
d	0	1	2	3
C_m^d	4	4	3	3
R_m^d	2	3	2	2
$C'_{skip}(d)$	14	4	1	0

A sample network for these parameters is shown in Fig. 3.

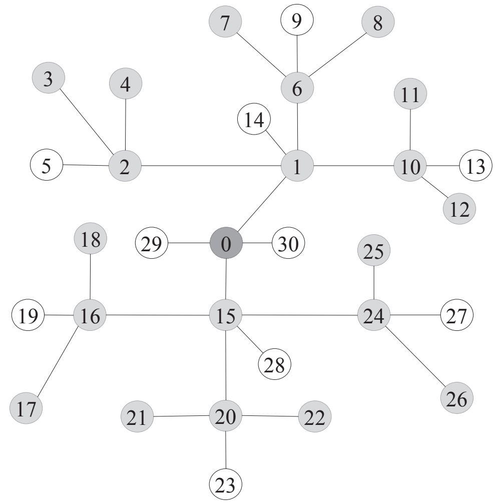


Fig. 3. Network addresses assignment.

Note that to calculate $C'_{skip}(d)$, a router needs $C'_{skip}(\mathrm{d}-1)$, which the C'_{skip} for its parent. So, during the address assignment, a parent does not only provide a block of addresses, it provides its own C'_{skip} also.

C. Tree Routing Mechanism

For hierarchical routing, if the destination is a descendant of a device, the device shall route the frame to the appropriate child. Trivially, every other device is a descendant of the ZigBee Coordinator and no device is a descendant of any ZigBee end-device.

Suppose, at any point of time a device with address A has obtained a packet targeted to the address *D*. The packet is processed as shown in Fig. 4.

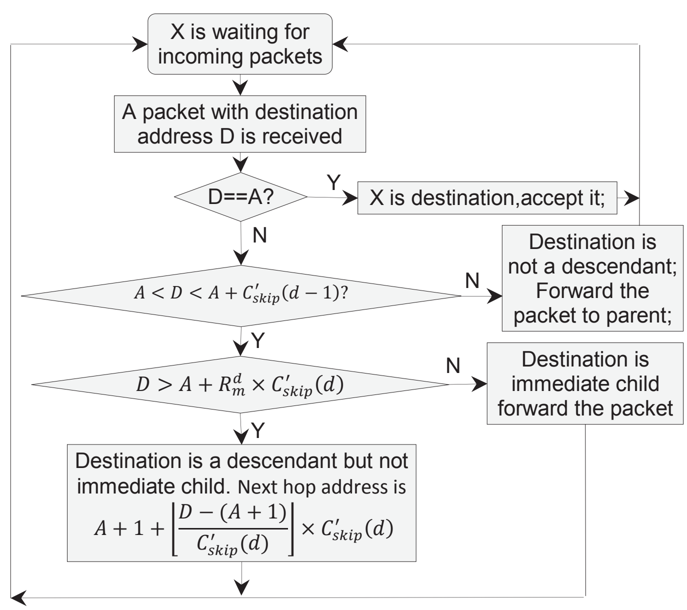


Fig. 4. Routing algorithm

D. Proof of Correctness

Refer to (2), (3) and (4), taking a routing decision is very simple. A target device with address D is a descendant of a ZigBee router X with address A at depth d if

$$A < D < A + C'_{skip}(d-1) \qquad (5)$$

This happens since X's parent [at depth *(d-1)*] gives the address A to X and the address $A + C'_{skip}(d-1)$ to the "closest" router-capable-sibling of *X*. This means, the address sub-block $[A+1,\ A + C'_{skip}(d-1) - 1]$ is reserved for X's descendants. So, above inequality must be true for a device D to be a descendant of A. Note that X uses its parent's C'_{skip}. How will X know this? X's parent, when sending address sub-block, should additionally send this value to X. The coordinator, as it does not have any parent, uses 2^b as its parent's C'_{skip}.

If this inequality is not valid, D is not a descendant of A. In that case, A has nothing to do except forwarding the packet to its parent. The packet traverses this way up until it reaches to the coordinator or some other router which is an ancestor of D. Two situations can happen. D may be a direct end-device child of A or D may be reached through some child router.

In the former case following inequality must be valid

$$D > A + R_m^d \times C'_{skip}(d)$$

This happens because, child end-devices gets address linearly when all the addresses for router children (i.e. $R_m^d \times C'_{skip}(d)$) are exhausted. Other wise the address *N* of the next hop device is determined as:

$$N = A + 1 + \left\lfloor \frac{D-(A+1)}{C'_{skip}(d)} \right\rfloor \times C'_{skip}(d) \qquad (6)$$

We look below at the derivation of formula (6). Refer to (3), consider the next-hop address is

$N = A+1+k \times C'_{skip}(d)$ for some k to be determined.

i.e. the packet is to be routed to the router-capable-child with address $A+1+k\, C'_{skip}(d)$ if $A+1+k\, C'_{skip}(d) \le D < A+1+(k+1)\, C'_{skip}(d)$

This is because the device with address $A+1+kC_{skip}(d)$ gets address sub-block $[A+1+kC'_{skip}(d),\ A+1+(k+1)\, C'_{skip}(d)-1]$

i.e. $N = A+1+k\, C'_{skip}(d)$, where $k \le \frac{D-A-1}{C'_{skip}(d)} < k+1$

$$\Rightarrow k = \left\lfloor \frac{D-(A+1)}{C'_{skip}(d)} \right\rfloor$$

So, next hop router can be determined by using only some equations and the complexity of this scheme is constant. Moreover, there is no routing table and that way searching procedure is completely eliminated.

V. Example Scenario

To understand how routing actually takes places, consider the network in Fig 3.

A. Example 1- router 3 wants to send a packet to another router 26.

Refer to Fig. 4, since $26 \neq 3$, it comes to the next step where it checks the expression $3 < 26 < 3+1$. Since, this evaluates to false, device 3 forwards the packet to its parent 2, which in turn does the same and forwards the packet to its parent 1. This way the packet eventually reaches to the coordinator with address 0.

Since, every device is a descendant of coordinator, the device 26 must also be a descendant (direct or indirect). Coordinator evaluates the expression $26 > 0 + 2 \times 14$ and finds false. This implies that the destination is not an end-device child of coordinator and must be forwarded to some router child. It finds the next hop address as $0+1+\left\lfloor \frac{26-(0+1)}{14} \right\rfloor \times 14 = 15$. The device 15, proceeds the same way and finds the next hop address as $15+1+\left\lfloor \frac{26-(15+1)}{4} \right\rfloor \times 4 = 24$.

Finally the device 24 calculates the next hop address $24+1+\left\lfloor \frac{26-(24+1)}{1} \right\rfloor \times 1 = 26$ and the packet reaches to the destination.

B. Example 2- router 17 wants to send a packet to end-device 27.

Refer to Fig. 4, since $27 \neq 17$, it checks the expression $17 < 27 < 17+1$. Since, this evaluates to false, device 17 forwards the packet to its parent 16 which, in turn does the same and forwards the packet to its parent 15. The device now evaluates the following expression $15 < 27 < 15+14$ which results true now. It evaluates the expression $27 > 15 + 3 \times 4$ and finds false. This implies that the destination is not an end-device child of 15 and must be forwarded to some router child. So, device 15 finds next hop router address as $15+1+\left\lfloor \frac{27-(15+1)}{4} \right\rfloor \times 4 = 24$.

Finally the device 24 calculates the following expression $27 > 24 + 2 \times 1$ and finds true. This indicates that the destination is an immediate child. So device 24 forwards the packet directly.

C. Example 3- end-device 9 wants to send a packet to another end-device 5.

Refer to Fig. 4, since $5 \neq 9$ and source is an end-device, it simply forwards the packet to its parent 6. The device 6 is not also the destination 6; it checks the expression $6 < 5 < 6+4$. Since, this evaluates to false, device 6 forwards the packet to its parent 1. The device 1 now evaluates the following expression $1 < 5 < 1+14$ which results true now. It evaluates the expression $5 > 1 + 3 \times 4$ and finds false. This implies that the destination is not an end-device child of 1 and must be forwarded to some router child. So, device 1 finds next hop router address as $1+1+\left\lfloor \frac{5-(1+1)}{4} \right\rfloor \times 4 = 2$.

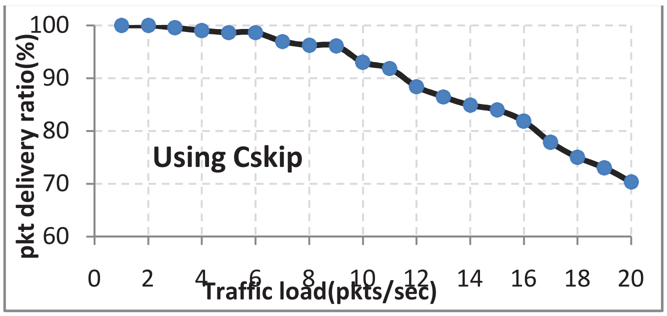
Finally the device 2 calculates the following expression $2 < 5 < 2+4$ and finds true and knows that 5 is its descendant. It then evaluates $5 > 2 + 2 \times 1$ and finds true. This indicates that the destination is an immediate child. So device 2 forwards the packet directly.

VI. EXPERIMENTAL RESULT

Since, we our modified Cskip works exactly like original one and the same routing algorithm is used, the performance of this proposed scheme is unquestionable. Moreover, we have already proved its correctness. In spite of that we have made several experiments in NS2 simulator to check its performance.

The experiment was carried out on a network as shown in Fig. 2. The source and destination pair was chosen randomly 10,000 times. Then a packet was sent from the source device to destination using our proposed method.

The same experiment was carried out using original tree routing. The result obtained is shown in Fig. 5.



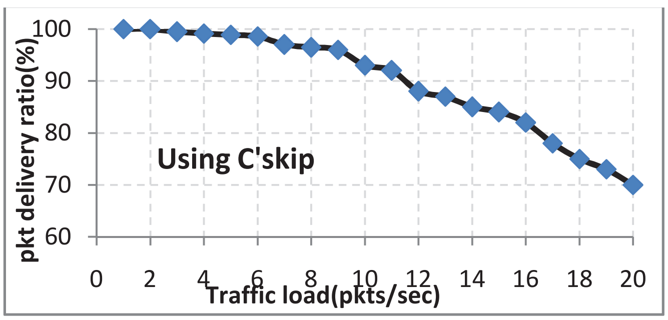


Fig. 5. Experimental result

The packet delivery ratio at the network layer, under low load, is 100%. This also proves the correctness of the protocol. However, under high load, the packet delivery ratio at the MAC sub-layer falls sharply with packet-rate due to increased number of collisions (which we do not model), as evidenced by the above curves.

Then we performed the same experiment the topology as shown in Fig. 3 using our proposed method. Since ZigBee tree routing does allow devices to have different C_m and R_m values, we could not carry out the experiment using it. The packet delivery ratio under low load was also found to be 100%.

VII. CONCLUSION

In this paper, we proposed a new recursive expression for C_{skip}. We demonstrated that it actually degenerates to original C_{skip} and hence original table-free routing algorithm can be applied seamlessly. The proposed scheme has all advantages the original ZigBee scheme had. Moreover, the expression is free from L_m and hence there is no hazard to select it prior to the network is formed. It also accepts different values of C_m and R_m at different network depth and hence supports asymmetric networks.

REFERENCES

[1] IEEE 802.15.4, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)

[2] ZigBee Alliance (ZigBee Document 02130r7) Draft Version 0.90: Network Specification, July 2004.

[3] Debabrato Giri, Uttam Kumar Roy, “Address Borrowing In Wireless Personal Area Network,” Proc. of IEEE International Anvanced Computing Conference, (IACC ’09, March 6-7), Patiala, India, page no 1074-1079

[4] Debabrato Giri, Uttam Kumar Roy, “Single Level Address Reorganization In Wireless Personal Area Network,” 4th International Conference on Computers & Devices for Communication (CODEC-09), December 14-16, 2009, Calcutta University, India.

[5] Debabrato Giri, Uttam Kumar Roy, "Multi Channel Personal Area Network(MCPAN) Formation and Routing," International Conference on Industrial Engineering Science and Applications (IESA 2014), April 2-4, 2014

[6] Uttam Kumar Roy, Debabrato Giri “WPAN Routing Using Huffman Technique,” Journal of Engineering Sciences, Vol (2), No (2), February, 2013 Pages 49-61.

[7] Uttam Kumar Roy, “Light-Weight Addressing and Routing Schemes for Resource-Constrained WPAN Devices,” Proc. Of 8th International Conference on Sensing Technology (ICST 2014), September 2-4, 2014, Liverpool, UK., ISSN-1178-5608, p 46-51

[8] Uttam Kumar Roy, “Extending ZigBee Tree Routing Protocol for Resource-Constined Devices,” Asia Pacific Conference Wireless and Mobile (APWiMob2014), August 28-30, 2014, Bali, Indonesia, p-48-53

[9] Uttam Kumar Roy, Debarshi Kumar Sanyal, Sudeepta Ray, “Analysis and Optimization of Routing Protocols in IEEE802.15.4,” Asian International Mobile Computing Conference (AMOC 2006), Jadavpur University, Kolkata, India

[10] Gang Lu, Bhaskar Krishnamachari, Cauligi S. Raghavendra, “Performance Evaluation of the IEEE 802.15.4 MAC for Low-Rate Low-Power Wireless Networks”, IEEE International Conference on Performance, Computing, and Communications, 2004.

[11] D. Ganeshan, B. Krishnamachari, “Complex Behavior at Scale: An Experimental Study of Low-Power Wireless Sensor Networks”, UCLA/CSD-TR 02-0013, UCLA Computer Science, 2002.

[12] J. Heidemann, W. Ye and D. Estrin.”An Energy-Efficient MAC Protocol for Wireless Sensor Networks”, Proceedings of the 21st International Conference of the IEEE Computer and Communications Societies (INFOCOM 2002), New York, NY, June 2002.

[13] C. E. Perkins and P. Bhagwat, “Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers”, Proceedings of ACM SIGCOMM, 1994.

[14] C. Schurgers, S. Park and M. B. Srivastava, “Energy-Aware Wireless Microsensor Networks”, IEEE Signal Processing Magazine, Volume: 19, Issue: 2, March 2002.

[15] D. B. Johnson, D. B. Maltz, “Dynamic Source Routing in Ad-hoc Wireless Networks”, Mobile Computing, T. Imielinski, H. Korth, Eds. Kluwer Academic Publishers, 1996, ch. 5, pp. 153-181.

[16] C. E. Perkins, E. Belding-Royer, and S. R. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing”, http://www.ietf.org/rfc/rfc3561.txt, July 2003. RFC 3561.

[17] C. E. Perkins and Elizabeth Royer, “Ad-hoc On-Demand Distance Vector Routing”, Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, New Orleans, LA, February 1999. [AODV_2]

[18] Elizabeth Royer and C-K Toh, “A Review of Current Routing Protocols for Ad-Hoc Mobile Wireless Networks”, IEEE Personal Communications Magazine, April 1999.

[19] Ed Callaway, Paul Gorday, Lance Hester, Jose A. Gutierrez, Marco Naeve, Bob Heile and Venkat Bahl. “Home Networking with IEEE 802.15.4: A Developing Standard for Low-Rate Wireless Personal Area Networks”,IEEE Communications Magazine August 2002.