



Design and Implementation of Tree Topology Algorithm for Power Line Communication Network

Guan-Jen Huang^(✉), Jih-Ching Chiu, and Yueh-Lin Li

Department of Electronic Engineering, National Sun Yat-sen University,
Kaohsiung, Taiwan

{m053010049, chiujihc}@ee.nsysu.edu.tw,
m063010112@student.nsysu.edu.tw

Abstract. The concept of smart grid has been proposed for years. Many countries, such as United State, England and Japan, have been replacing traditional electric meters with smart electric meters in recent years. There are lots of communication methods used in smart grid, power line communication (PLC) is an important one among them. G3-PLC is a widely-used specification for long-distance PLC, however, PLC is sensitive to old power lines and the interference caused by large electric current flowing through the power line. Moreover, although G3-PLC has stable performance of communication, the AODV routing protocol and the complex startup procedures results in taking long time for G3-PLC devices to finish the whole startup procedures.

To reduce the time to finish the startup procedures in G3-PLC, in this paper, a tree topology algorithm is proposed. In the tree topology algorithm, a simple startup procedure is provided, by setting up the parent-child relationship between nodes, the routes for nodes in networks are simplified. Furthermore, a maintain procedure is also provide in the algorithm, in the maintain procedure, the nodes in networks can use Check Alive mechanism to check the connection between their parent nodes and child nodes, when a broken connection is found by a node, the node will use Recovery mechanism to rescue the isolated nodes.

In this paper, the feasibility of the tree topology algorithm is verified by NS-3 platform, and suggestions of the suitable value of the parameters in the algorithm are proposed. In addition, the algorithm is implemented on Atmel SAM4CP16C evolution kits, the measurement results show that the time for PLC devices to finish the startup procedures of the tree topology algorithm is at least 6.7 times less than G3-PLC specification.

Keywords: PLC · Tree topology · Startup procedure · Maintain procedure

1 Introduction

THE Smart Grid works by establishing the Advanced Metering Infrastructure(AMI). As shown in Fig. 1, AMI includes Meter Data Management System (MDMS), Smart Meter, Communication systems and Power supply-consumption management [1]. There are lots of communication methods used in smart grid, however, considering the

wide establishment of Smart Meters in the future, how to arrange transmission lines for Smart Meters appropriately is important. To adapt different geographic environments and to consider the economic benefit, power line communication (PLC) is an appropriate way for communication of Smart Grid because the new arrangement of power line is needless. G3-PLC is a widely-used specification for long-distance PLC, however, although G3-PLC has stable performance of communication, the AODV routing protocol and the complex startup procedures results in taking long time for G3-PLC devices to finish the whole startup procedures. To reduce the time to finish the startup procedures in G3-PLC, in this paper, a tree topology algorithm is proposed. In the tree topology algorithm, a simple startup procedure and a maintain procedure are provided.

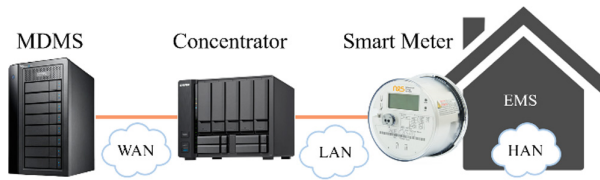


Fig. 1. Structure of AMI

2 Background Knowledge

According to the frequency of carrier used in PLC technology, there are broadband PLC and narrowband PLC. The carrier frequency of narrowband PLC ranges from tens of kHz to hundreds of kHz, and the lower carrier frequency results in the longer transmission distance. G3-PLC [2] and PRIME [3] are the widely-used narrowband PLC specification nowadays, both specifications use OFDM as the signal modulation method in physical layer [4–7] because OFDM provides low interrupt between sub-carriers and increases bandwidth usage rate which can improve the low transmission quality of old power line. In addition, every subcarrier uses PSK to modulate digital signal to analog signal, although both specifications use OFDM and PSK as modulation methods, different encode way between them makes G3-PLC have better bit error rate (BER) than PRIME [8–10]. The OSI model of G3-PLC is shown in Fig. 2, IEEE 802.15.4 protocol [11] is used in MAC layer, and 6LoWPAN protocol is used in Adaption sublayer to connect IEEE802.15.4 with IPv6.

To compare the startup procedure and routing way of G3-PLC with the tree topology algorithm proposed in this paper, the introduction of the startup procedure and routing protocol of G3-PLC are as follows.

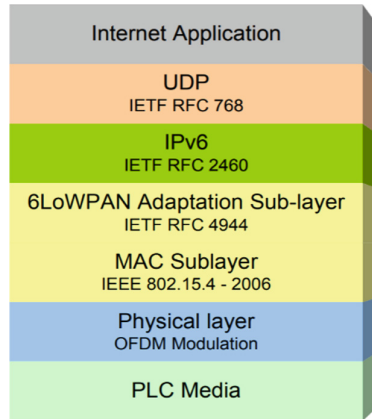


Fig. 2. OSI model of G3-PLC

2.1 Startup Procedure: LoWPAN Bootstrapping Protocol

The LoWPAN Bootstrapping Protocol (LBP) [12] defines the startup procedure of every 6LoWPAN device. There are three proper nouns in LBP, include LoWPAN BootStrapping Device (LBD) which represent the device that not yet join a PAN, LoWPAN BootStrapping Server (LBS) is the coordinator of PAN which in charge of all devices, and LoWPAN BootStrapping Agent (LBA) which represent the device that already join a PAN. Every device has its unique EUI-64 which is the MAC Address defined in IEEE 802.15.4, in addition, LBAs need to help LBDs or LBS transfer data. The following steps are the startup procedure of G3-PLC.

1. LBD starts and resets.
2. LBD checks if there is any devices nearby, so broadcasts “Beacon request” during active scan, any LBS or LBA receive this frame needs to reply “Beacon”.
3. LBD starts processing LBP. LBD broadcast “LBA Solicitation Message” to find LBA who can help it transfer data to LBS.
4. After receiving “LBA Advertisement” from LBAs or LBS, LBD chooses a best LBA(LBS) among them. The way to choose the best LBA is as following: (1) If there is LBS who reply “LBA Advertisement”, LBS is the one to choose, (2) the LBA that has the smallest hops from LBD to LBS is the one to choose.
5. LBD and LBS starts to exchange some frames through the LBA chosen in step 4. The sequence chart is shown in Fig. 3. “Joining” represents the frame transmitted from LBD to LBS, and “Challenge” represents the frame transmitted from LBS to LBD, and “Accept” is the frame that LBS decides to accept this LBD join this PAN. Every LBD needs at least three back and forth data exchange between itself and LBS to finish joining a PAN.

Furthermore, every G3-PLC device has a Neighbor Table which contains the connection situation between itself and other devices.

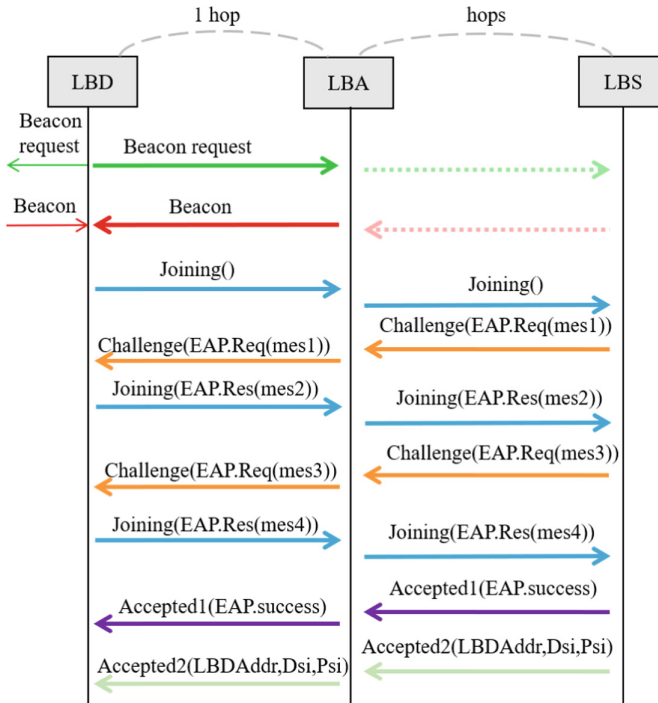


Fig. 3. Sequence chart of LBD to register

2.2 Routing Protocol: 6LoWPAN Ad Hoc On-Demand Distance Vector Routing

G3-PLC uses 6LoWPAN Ad Hoc On-Demand Distance Vector Routing (LOAD) [13, 14] as its routing protocol. A registered node will process LOAD before its data transmission to find the route between itself and the destination node. The following steps are the routing protocol of G3-PLC.

1. Source node broadcast “Route Request (RREQ)” which contains the information of the destination node.
2. A relay node that receives RREQ adds the source node’s information to its “Routing Request Table”, meaning itself is now helping the source node to find routes to destination node. And then this relay node also adds the source node’s information to its “Routing Table”, helping the source node to find routes by broadcast RREQ.
3. Until the destination node receives the RREQ of the source node from any relay nodes and updates its Routing Table, it replies “Route reply (RREP)” to the source node through reply nodes.

LOAD not only can establish routes but repair routes. The repair mechanism will process when any relay node in a route is off. The upstream node of the offline node broadcasts RREQ again to find a new route to the destination node, besides, the frame that failed to transmit will be save for next try after new route established. If new route

failed to build, the upstream node of the offline node will transmit “Local Repair and Route Error (RRER)” to the source node, when the source node receives RRER, a new routing process of the destination node will start by broadcasts RREQ.

From what has been discussed above, G3-PLC not only needs long time to register and many frames exchange in startup procedure but using the LOAD as its routing protocol results in many frame collisions. To improve those problem, the tree topology algorithm is proposed in this paper. The tree topology could establish simple routes between each node to reduce the frame collision.

3 Tree Topology Algorithm for Power Line Communication Network

The tree topology algorithm proposed in this paper contains startup procedure and maintain procedure, and it’s suitable for tree topology, besides, the Modbus transfer protocol is used for data transfer after the startup procedure. In tree topology, every node has their own parent node and child nodes. The names of the characters used in this algorithm are as following: Coordinator oversees a PAN, Device is a node that wants to join a PAN, and Agent is a node that already join a PAN, in addition, Agent also helps Coordinator and Devices to transfer messages if needed.

To build a tree topology, the routing tables are needed, so in this algorithm, there are three tables for Coordinators, they are Registered Node’s MAC Address Table, Registered Node’s last Agent Node’s MAC Address Table, and Registered Node’s Modbus Address Table, furthermore, there are four tables for Device, they are Parent Node’s MAC Address Table, Child Node’s MAC Address Table, Multilevel-Child Node’s MAC Address Table, and Multilevel-Child Node’s last Agent Node’s MAC Address Table. The introduction of startup procedure and maintain procedure are as following sections.

3.1 Startup Procedure

In this section, bold font represents the name of frame used in startup procedure, and bottom line represents the name of tables. The illustration of startup procedure is shown in Fig. 4. The details of Startup Procedure in Fig. 4 is as following steps:

1. Device2 broadcasts **Routing Request** following the “Resend Mechanism of Routing Request” to look for replies from any registered devices nearby.
2. Device1 receives Routing Request from Device2, and it’s already registered, so it transmits **Routing Reply** to Device2. After Device2 receives Routing Reply, it can start to transmit the register-related frame through Device1.
3. Device2 transmits **Direct Register Request** to Device1, since Device1 is not Coordinator, Device1 needs to add Device2 to its Child Node’s MAC Address Table and form **Indirect Register Request** which includes the information of Device2 and transmit to its parent node. “Direct” here means a device wants to join a PAN is accepted directly from Coordinator, not other devices. After Coordinator receives **Indirect Register Request** which includes the information of Device2, it adds Device2 to its Registered Node’s MAC Address Table, meaning it accepts Device2 to join this PAN.

- Coordinator then transmits **Indirect Register Reply** to the last Agent Node of Device2-Device1, and Device1 add Device2 to its Child Node's MAC Address Table. When Device2 receives **Indirect Register Reply**, it adds Device1 to its Parent Node's MAC Address Table, and the register procedure of Device2 is finished.

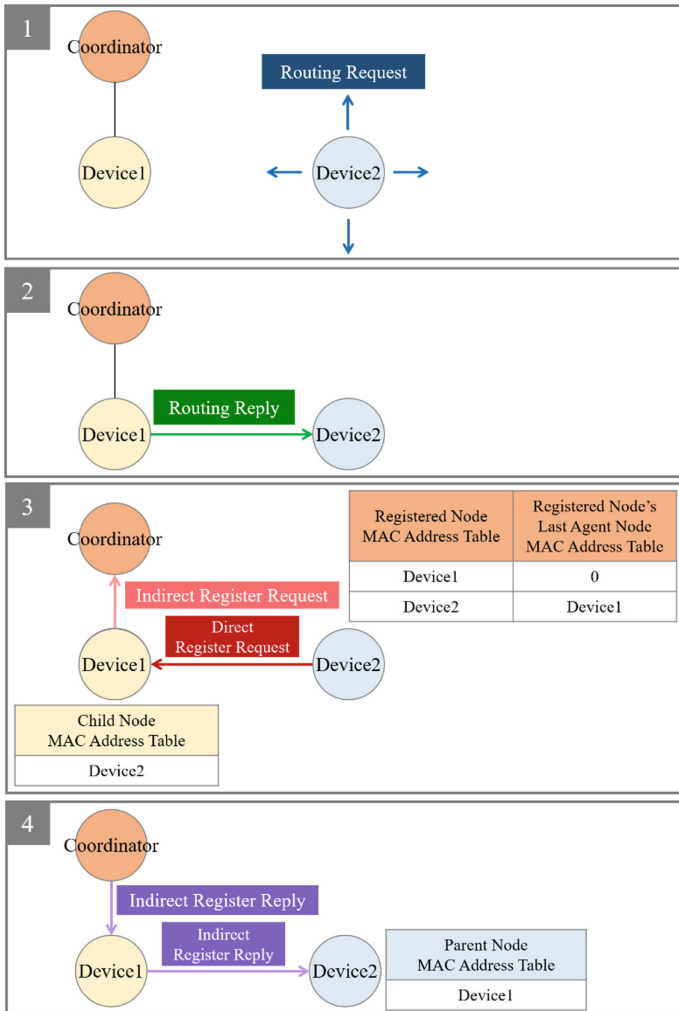


Fig. 4. The illustration of startup procedure

3.2 Resend Mechanism of Routing Request

In startup procedure, when a not-registered device broadcasts Routing Request but doesn't get any reply, the register procedure fails, that is, the not-registered node needs to restart the register procedure. The Resend Mechanism is proposed in this section to make

sure every device can register successfully. The illustration of the Resend Mechanism is shown in Fig. 5. A check period is the product of Duration of check receiving Routing Reply (t) and Times of check receiving Routing Reply (p). At the beginning of a check period, device which wants to register will check whether itself received **Routing Reply** or not, if not, it will resend **Routing Request** or do nothing until next inspection time. After p times inspection, the device will check itself registered or not, if not, it will resend Routing Request and restart a new check period or stop the Resend Mechanism. How to choose the value of t and p will be discussed in chapter IV.

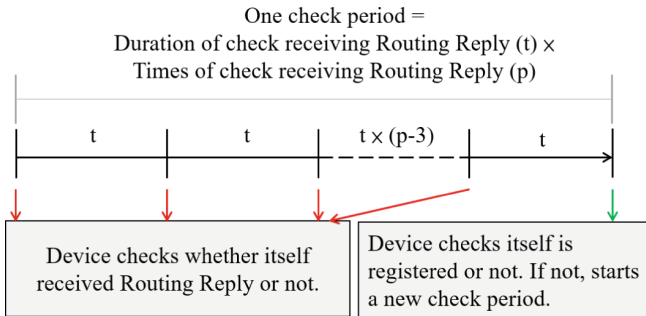


Fig. 5. Illustration of Resend Mechanism of Routing Request

3.3 Maintain Procedure

Since power line is sensitive to ambient interference, how to maintain a network build with power line is also important. In this algorithm, a Maintain Procedure is proposed. The procedure has Check Alive Mechanism and Recovery Mechanism respectively to periodic check the connection between itself and its related nodes and rescue the isolated nodes.

The definition of the characters in Maintain Procedure is shown in Fig. 6. There are three new defined character, they are Missing Child Node, Missing Parent Node and Isolated Child Node. The left side diagram in Fig. 6, when Device1 Finds that one of its child nodes-Device2 is disconnect, Device2 is called Missing Child Node, in addition, Device3 and Device4 are called Isolated Child Node. Since Device2 is disconnected, Device3 and Device4 are disconnected with Coordinator, too, that's why the Recovery Mechanism is needed, for Device1 to rescue the Isolated Child Nodes. Furthermore, the other side diagram in Fig. 6, when Device2 finds its parent node is disconnected, it uses the Recovery Mechanism to seek for a new parent node, in order to connect to coordinator again.

The details of Maintain Procedure will be introduced from the viewpoint of Parent Node and Child point individually. The illustration of the viewpoint of Parent Node is shown in Figs. 7 and 8, and the details are as following steps. The bold fonts represent name of frames, the bottom line represent name of tables, and the italics font represent parameters.

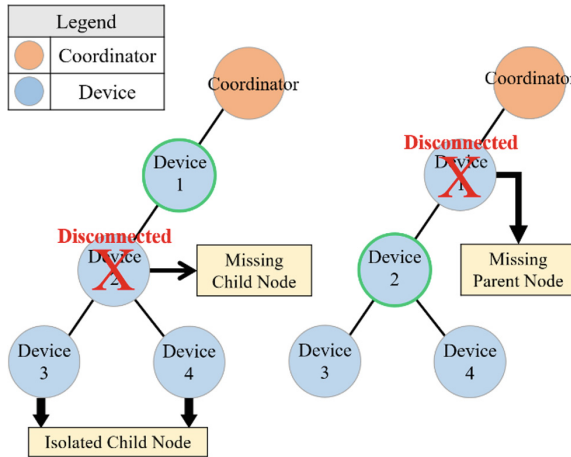


Fig. 6. Definition of the characters in Maintain Procedure

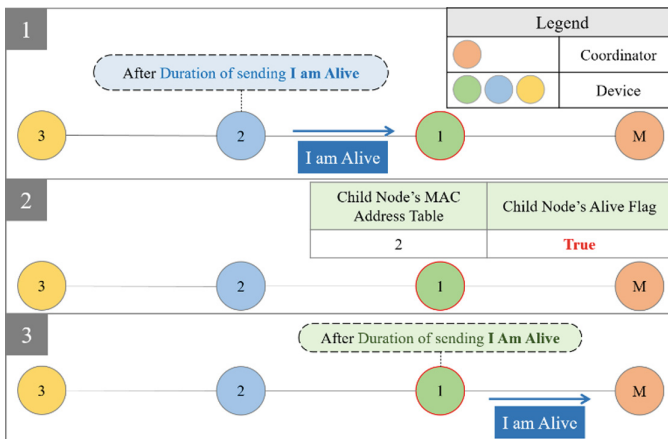


Fig. 7. Illustration of Maintain Procedure by viewpoint of Parent Node (1)

1. Every Child Node of Device1 transmit **I am Alive** to its Parent Node when its timer counts to *Duration of sending I am Alive*.
2. When Device1 receives **I am Alive** from Device2, it updates the *Child Node's Alive Flag* in *Child Node's MAC Address Table* to True.
3. After the timer of Device1 counts to *Duration of sending I am Alive*, it transmits **I am Alive** to its Parent Node-Coordinator.
4. And then Device1 Check the *Child Node's Alive Flag* of its *Child Node's MAC Address Table* with the yellow table in step 4 of Fig. 8.
5. If the *Times of Child Node is Not Alive* has counted to the *Child Node Not Alive Value*, it means that Device1 considered Device2 is disconnect, and need to start the Recovery Mechanism to rescue the isolated Child Node.

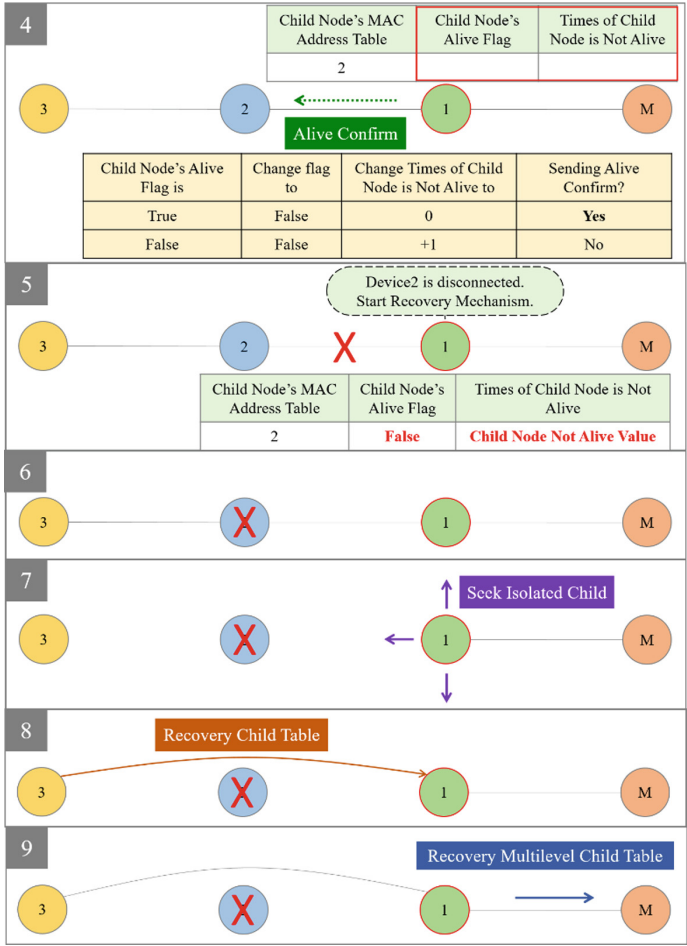


Fig. 8. Illustration of Maintain Procedure by viewpoint of Parent Node (2)

6. Device2 has been considered disconnect by Device1.
7. The Recovery Mechanism starts, Device1 broadcasts **Seek Isolated Child** to seek for replies from isolated child nodes.
8. If Device3 is able to receive **Seek Isolated Child**, it replies **Recovery Child Table** with all the information of its Child Nodes and its Multilevel Child Node, in order to inform Device1 to update the Child Node's MAC Address Table.
9. After updating its Child Node's MAC Address Table, Device1 transmits **Recovery Multilevel Child Table** to its Parent Node, informing its Parent Node to update Multilevel-Child Node's MAC Address Table, and then the Recovery Mechanism is finished.

On the other hand, the illustration of the viewpoint of Child Node is shown in, and the details are as following steps.

1. When Device2 receives **Alive Confirm** from Device1, it updates the *Parent Node's Alive Flag* in Parent Node's MAC Address Table to True, and after the timer of Device1 counts to *Duration of sending I am Alive*, it transmits **I am Alive** to its Parent Node-Device1.
2. Then Device2 Check the *Parent Node's Alive Flag* of its Parent Node's MAC Address Table with the yellow table in step 2 of Fig. 9.

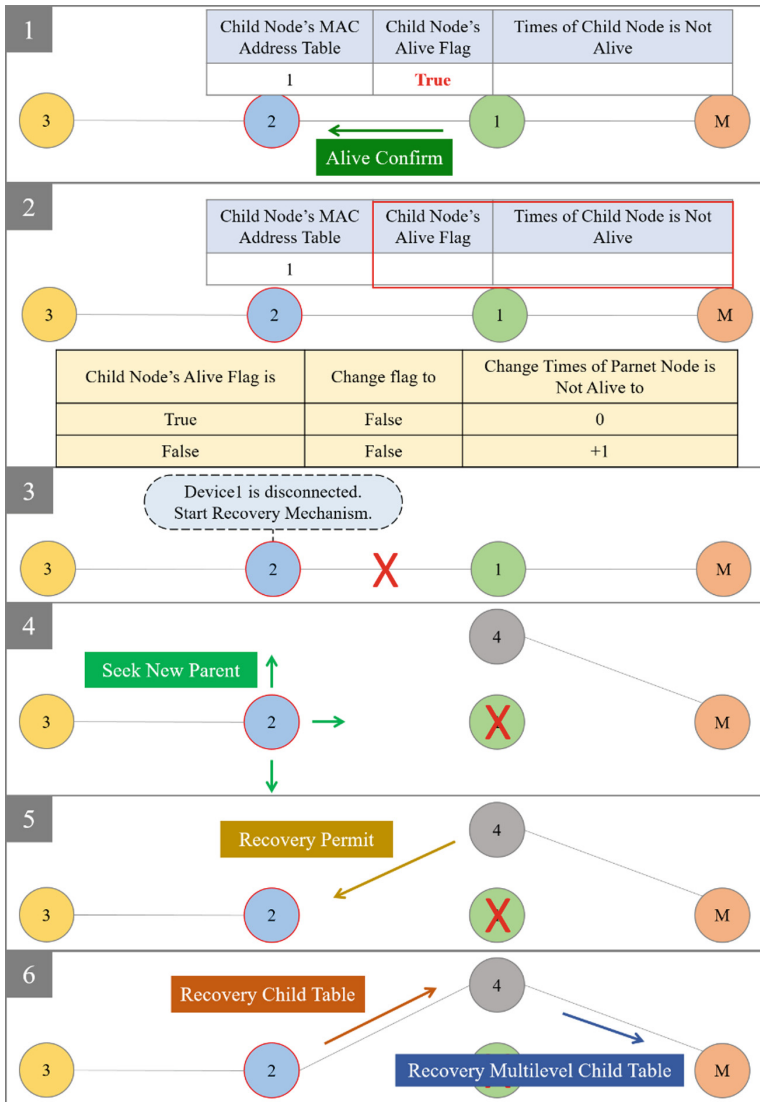


Fig. 9. Illustration of Maintain Procedure by viewpoint of Child Node

3. If the *Times of Parent Node is Not Alive* has counted to the *Parent Node Not Alive Value*, it means that Device2 considered Device1 is disconnect, and need to start the Recovery Mechanism to find a new Parent Node.
4. The Recovery Mechanism starts, Device2 broadcasts **Seek New Parent** to seek for replies from other registered nodes.
5. If Device4 is able to receive **Seek New Parent**, it replies **Recovery Permit** to Device2 telling Device2 that I'm your new Parent Node.
6. Then, Device2 replies **Recovery Child Table** with all the information of its Child Nodes and its Multilevel Child Node, in order to inform Device4 to update the Child Node's MAC Address Table. After updating its Child Node's MAC Address Table, Device4 transmits **Recovery Multilevel Child Table** to its Parent Node, informing its Parent Node to update Multilevel-Child Node's MAC Address Table, and then the Recovery Mechanism is finished.

4 Simulation and Implementation

This Tree Topology Algorithm is verified with NS-3 [15, 16] platform, and the appropriate value of parameters is proposed, and this algorithm is also implemented on evolution module of G3-PLC.

4.1 Verified with NS-3

The following discussion is about the factor that influence the duration of whole startup procedure. The verification uses different value of t and p of Resend Mechanism of Routing Request, and there is some conclusion about those tests. The most important factor is the times of Resending Routing Request. There are some reasons about a node has to resend. First, the node who is able to receive the Routing Request from the node wants to register is not yet registered, so it can't transfer frame for the node wants to register, results in the node want to register has to resend Routing Request. Second, the node wants to register already received Routing Reply, but the following register-related frames lost in progress, so it's needed to restart the register procedure.

Besides, the tree level of the topology doesn't influence the duration of a single node to finish to Startup Procedure, the factor that influence the most is the position of Agent node and the workload of the Agent load.

Moreover, the value of t and p is also an important factor. After many tests, $p = 2$ and $p = 3$ is better than other value of p . So the following tests of different value of t are under $p = 2$ and $p = 3$. Figure 10 shows the diagram of recommend value of t vs. duration of Startup Procedure, the node numbers of simulations are from 10 to 80 nodes, and the curve of duration of Startup Procedure with different node number is linear, furthermore, the recommended value of t is stuck in 3200 ms after the node number is over 40 nodes. From this relation diagram, the duration of 80 nodes is 208 s. The equation of the curve of recommended value of t is shown as Eq. (1).

$$t = \begin{cases} 800 + 70 \times (\text{node number} - 10) \text{ ms}, & 10 \leq \text{node number} < 40 \\ 3200 & \text{ms}, \text{ node number} \geq 40 \end{cases} \quad (1)$$

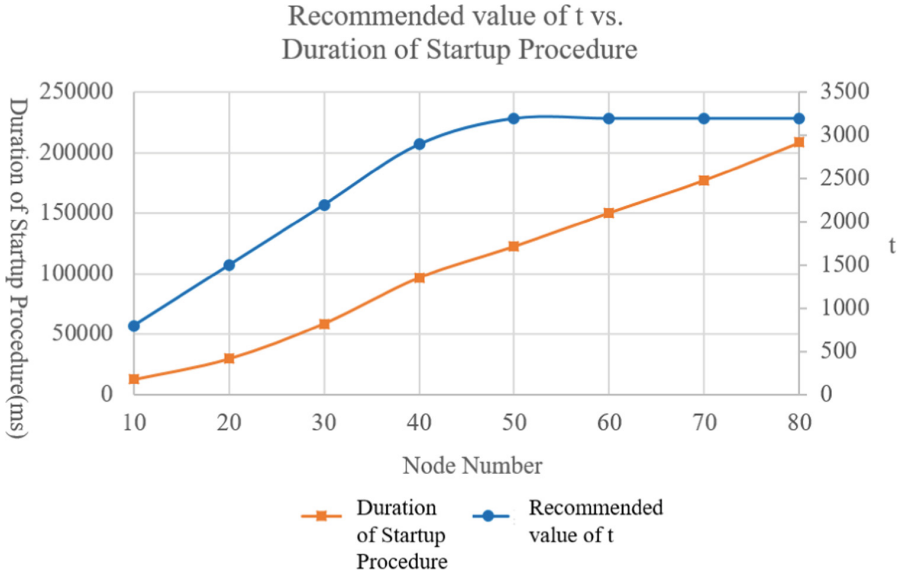


Fig. 10. Recommend value of t vs. Duration of Startup Procedure

4.2 Implementation on Atmel SAM4CP16C-EK [17]

This algorithm is also implemented on Atmel SAM4CP16C-EK. To test the feasibility of the algorithm, using 1 node to 6 nodes, each node executes Startup Procedure. To verify the recommended value of t and p mentioned last section, four different values of t and p are used, from No. 1 to No. 4, the values of p and t are 3/800 ms, 1/800 ms, 3/1200 ms, 3/2000 ms. The values of No. 3 is match with the recommended values mentioned last section. The test results of the duration of Startup procedure are shown in Fig. 11. The results show that the duration which used the recommended value of t and p have the smallest duration of the Startup Procedure. The recommended values are verified.

Furthermore, the comparison result of the tree topology and G3-PLC is in Fig. 12. The duration of this tree topology algorithm is at least 6.7 times less than G3-PLC specification.

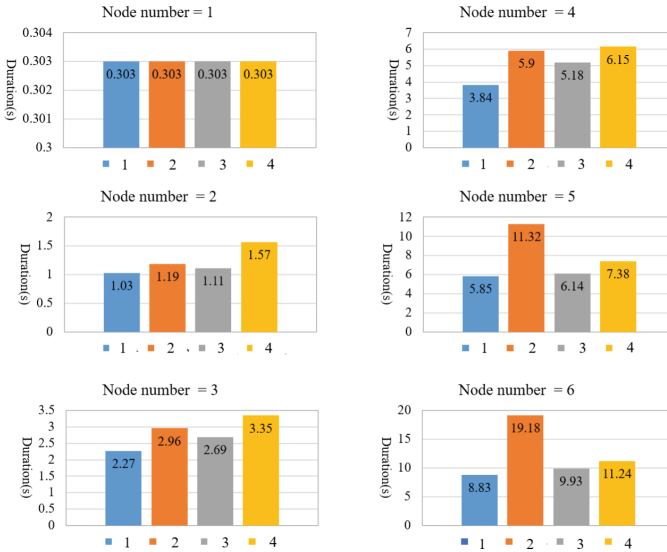


Fig. 11. Tests results of different values of t and p

Duration of Startup procedure (s)	Node = 1	Node = 2	Node = 3	Node = 4	Node = 5	Node = 6
Tree Topology Algorithm	0.303	1.03	2.27	3.84	5.85	8.83
G3-PLC	15	20	26	33	47	60

Fig. 12. The results of implementation

5 Conclusion

The tree topology algorithm is proposed in this paper, by exchanging frames between nodes, a tree topology and routing table is established. And a Maintain Procedure is also proposed to prevent the network from broken connection. The verification and the recommended values of parameters are proposed, and the results of implementation is also proposed, the results show that the duration of this tree topology algorithm is at least 6.7 times less than G3-PLC specification.

References

1. Author, F.: Article title. Journal **2**(5), 99–110 (2016)
2. Author, F., Author, S.: Title of a proceedings paper. In: Editor, F., Editor, S. (eds.) Conference 2016, LNCS, vol. 9999, pp. 1–13. Springer, Heidelberg (2016)
3. Author, F., Author, S., Author, T.: Book title, 2nd edn. Publisher, Location (1999)

4. Author, F.: Contribution title. In: 9th International Proceedings on Proceedings, pp. 1–2. Publisher, Location (2010)
5. LNCS Homepage. <http://www.springer.com/lncs>. Accessed 21 Nov 2016
6. Kabalci, Y.: A survey on smart metering and smart grid communication. *Renew. Sustain. Energy Rev.* **57**, 302–318 (2016)
7. G3-PLC Alliance. <http://www.g3-plc.com/home/>
8. PRIME Alliance. <http://www.prime-alliance.org/>
9. PLC G3 Physical Layer Specification, Électricité de France S.A. <https://www.edf.fr/en/meta-home>
10. Specification for PowerLine Intelligent Metering Evolution, PRIME Alliance Technical Working Group. <http://www.prime-alliance.org/>
11. Narrowband orthogonal frequency division multiplexing power line communication transceivers – Power spectral density specification, ITU-T G.9901, Telecommunication Standardization sector of ITU. <https://www.itu.int/en/pages/default.aspx>
12. Narrowband orthogonal frequency division multiplexing power line communication transceivers for G3-PLC networks, ITU-T G.9903, Telecommunication Standardization Sector of ITU. <https://www.itu.int/en/pages/default.aspx>
13. Hoch, M.: Comparison of PLC G3 and PRIME. Presented at IEEE International Symposium on Power Line Communications and Its Applications, Udine, Italy, April 2011
14. Matanza, J., Alexandres, S., Rodriguez-Morcillo, C.: Performance evaluation of two narrowband PLC systems: PRIME and G3. *Comput. Stand. Interfaces* **36**(1), 198–208 (2013)
15. Sadowski, Z.: Comparison of PLC-PRIME and PLC-G3 protocols. Presented at International School on Nonsinusoidal Currents and Compensation (ISNCC), Lagow, Poland, June 2015
16. IEEE Computer Society, IEEE 802.15.4-2006: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs). <http://www.ieee802.org/15/pub/TG4.html>
17. Network Working Group of IETF, Commissioning in 6LoWPAN. <https://www.ietf.org/>
18. Network Working Group of IETF, 6LoWPAN Ad Hoc On-Demand Distance Vector Routing (LOAD). <https://www.ietf.org/>
19. Ramírez, D.F., Céspedes, S.: Routing in neighborhood area networks: a survey in the context of AMI communications. *J. Netw. Comput. Appl.* **55**, 68–80 (2015)
20. Network Simulator-3. <https://www.nsnam.org/>
21. Low-Rate Wireless Personal Area Network (LR-WPAN). <https://www.nsnam.org/docs/models/html/lr-wpan.html>
22. Microchip, Microchip_G3-PLC Firmware Stack User Guide. <http://www.microchip.com/design-centers/smart-energy-products/power-line-communications/g3-based-plc-solutions>