

Combination Synchronization Protocol with Center Nodes in Device-to-Device Communication System

Jiangsheng Fan , Yafeng Wang* , Lu Youxiong

Wireless Theory & Technology lab (WT&T), Beijing University of Posts and Telecommunications
Beijing, China

Wireless Theory & Technology lab (WT&T), Beijing University of Posts and Telecommunications
Beijing, China

Wireless Product R&D Institute ZTE Corporation Xi'an, China

jsfan@bupt.edu.cn, wangyf@bupt.edu.cn, lu.youxiong@zte.com.cn

Keywords: Combination; Cluster; Pre- synchronization.

Abstract: In this paper, we propose a hybrid synchronous algorithm based on the combination of distributed and centralized algorithm. First of all, we put forward the concept of local cluster and generalized local cluster. The cluster is a collection of synchronous UE meeting certain conditions and has a center UE. Through this mechanism, we can finish many synchronous parts in advance what we call pre- synchronization which can reduce system latency. When we need system synchronization, we will use the flooding time synchronous protocol. The algorithm has low latency as well as high accuracy and good robustness.

Introduction

Device-to-Device (D2D) [1] communication is one of the latest technology in recent years. Compared with the traditional communication system, D2D communication has great advantages in both spectrum efficiency and coverage. Many academic institutions pay much attention to D2D related technology research. And the synchronization problem [2] without cellular network coverage is one of the key issue in the D2D communication research nowadays [3,4].

Time synchronization is the precondition of communication and this technology has been widely used in the traditional network. In 1991, professor Mills put forward the network time synchronization protocol (NTP) [5] which is useful to solve synchronization problem in the internet. In 2002, Jeremy Elson and Kay Romer put forward the problem of time synchronization in wireless sensor networks (WSN) [6] at an international conference for the first time. Shortly after, a protocol called RBS[7,8] (Reference Broadcast Synchronization) came into the public view and the technical means for the time synchronization problem of wireless sensor network has opened up a train of thought. In June 2003, Ping S put forward the delay measurement time synchronization (DMTS) mechanism. In November, Ganeriwal proposed time synchronization protocol for sensor networks (TPSN) [9,10]. In November 2004, Maroti put forward the flooding time synchronization protocol (FTSP)[11,12], the agreement considers the energy awareness, scalability, robustness, stability and convergence of synchronization requirements, and single jump time synchronization error can be within 1 us and it has a good application prospect.

In this paper, we proposed a combination time-synchronization protocol with center points (CTPCN) in Device-to-Device communication system. This technology can be realized without cellular network coverage. Using center node synchronization, we can transform multiple hop synchronous problem into a several single hop synchronous problem. Because one hop synchronization problem can be solved through our synchronous cluster, the system can achieve synchronization through the time synchronization protocol FTSP one hop to another. Analysis and simulation show that this algorithm has good robustness, high precision, and small time delay characteristic.

The rest of this paper is organized as follows: in section II we present the system model. In section III we show the process of this algorithm. In section IV we present the simulation results and finally we conclude in section V.

System Model

In this algorithm, all the nodes are random distribution at the beginning and they are equivalent. If T1 overflows, a searching signal will be broadcasted, any node which receives the searching signal will give back an ACK signal which contains its own signal confirmation ID and other information. Through the limited conditions, we choose the center nodes and divide network nodes into several sub-networks to establish a local cluster, generalized local cluster and finish pre-synchronization process. As a result, before the source node launching a synchronous request, the network has already reached a semi-autonomous partial synchronization network state, so pre-synchronization process saves a lot of time overhead. System model diagram is shown in figure 1.

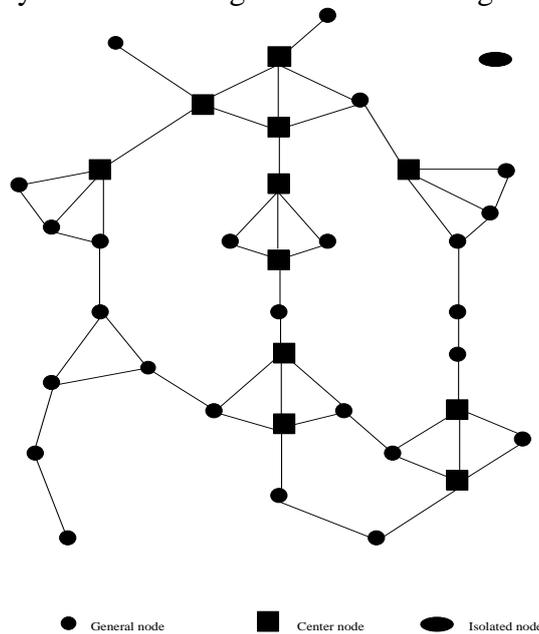


Fig.1. System model diagram

Here the local cluster refers to the center node as well as its neighbor nodes; the generalized local cluster refers to a local cluster as well as its neighbor local cluster or nodes which don't belong to any local cluster.

The algorithm description

Node searching

Node searching function is mainly used to get relevant data to judge the center node including the number of neighbor nodes and the distance between neighbor nodes and the searching nodes.

Specifically, the number of neighbor nodes and average distance is obtained as follows:

Referring to the process of TPSN protocol, we can get the following data acquisition process. First of all, the node (A) broadcast searching signal and record signal transmission time W_1 at the same time. Then neighbor nodes (B) which receive broadcasting signal record signal arriving time W_2 . At the moment of W_3 , B node responses an ACK signal. At W_4 , A node received the ACK signal from B, so we can get:

$$D = [(W_2 - W_1) + (W_3 - W_4)]/2$$

$$d = [(W_2 - W_1) - (W_3 - W_4)]/2 \tag{1}$$

Here the W_1, W_4 are local time of node A; W_2, W_3 are the local time of node B. D and d means the transmission time between A, B and clock skew between the nodes A, B respectively.

Then we can get:

$$L_{AB} = D * c \tag{2}$$

C is the speed of light.

L_{AB} is saved in node A until node A gets all the neighbor nodes distance information data group L . And through L , we can get the average distance L_0 . The number of the neighbor nodes can get by a counter K . At the beginning $K = 0$, when node A receives an ACK signal, counter K adds 1 automatically. The above process is the method to get neighbor nodes number and average distance, when T_1 overflows, the process above will start again.

Data saved in node A is as follows in Table 1:

Table1. Data saved in node A

Node array	User's sign	Center node sign	Neighbor's number	Distance array	Average distance
IDG	ID	S	K	L	L_0

IDG: Used to judge the node state. Whenever the ACK signal are containing the new user ID, a record will be send into IDG array;

S: Used to judge the center node. $S = 1$ is corresponding to center node and $S = 0$ means general node. Initially $S = 0$, when L_0 or K updates, judge center node again;

K: Used for neighbor nodes statistics, when receive a new ID user ACK signal, K add 1 automatically;

L: Used to store the calculated the distance between the node and all its neighbor nodes, and changes with K at the same time;

L_0 : Used to get the node itself with all the average distance of the neighbor nodes, the decision condition is as follows:

1) The number K of neighbor nodes around the node is greater than or equal to n (usually takes more than 3 integer);

$$2) L_0 \leq \forall L_i \in \{L_1, L_2, \dots, L_k\} \quad i \in 1, 2, \dots, k$$

L_0 : the average distance between node A and all the neighbor nodes;

k : the number of neighbor nodes;

L_i : the average distance between neighbor node i and the neighbor node i 's all neighbor nodes;

ACK: ACK signal stands for neighbor nodes signal response, including the following specific content:

Table2. ACK signal

User's sign	Average distance
ID	L_0

Center node information is obtained by the above steps. After the judgment of center node, the center node uses TPSN synchronization protocol to reach synchronization with all its neighbor nodes, completing the construction of the local group.

Pre- synchronization

Node searching part:

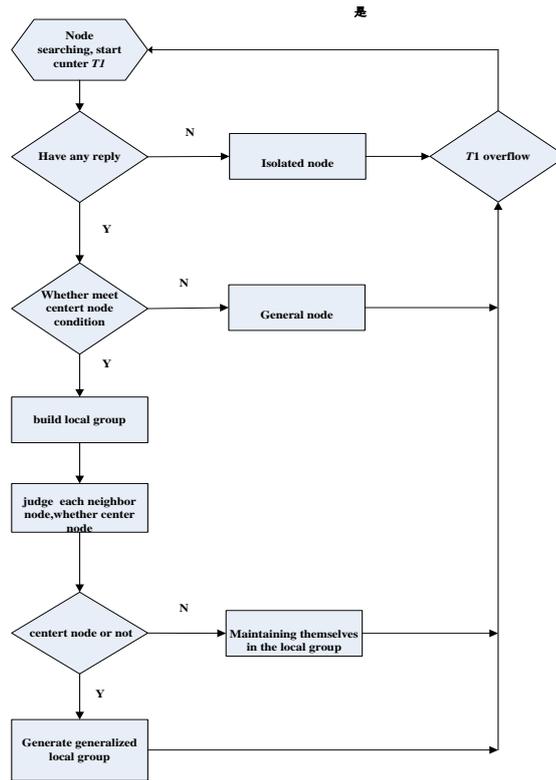


Fig.2. Node searching part

The synchronization process:

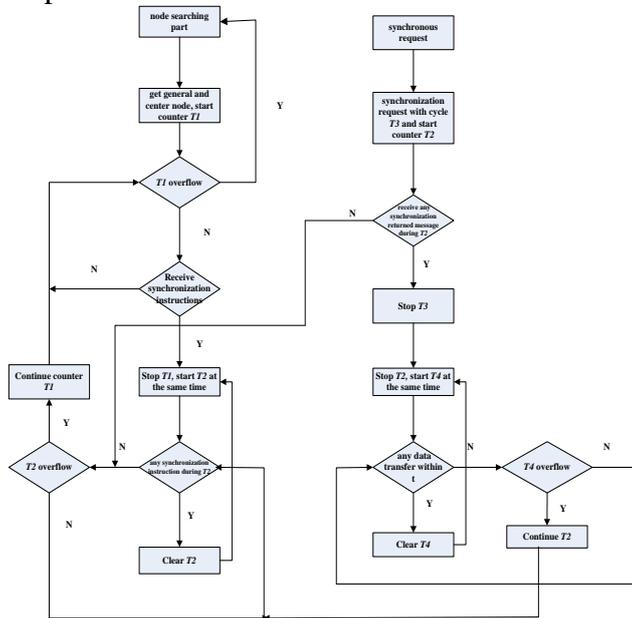


Fig.3. The synchronization process

Simulation Results

In this section, numerical results of the proposed algorithms are presented. We get the latency and accuracy of the system, and the contrast algorithm is tree algorithm (LTS) without center nodes. The simulation parameters are as follows in Table 3:

Table3. Simulation parameters

Scenario	Area(m)	Cycles	Maximum communication distance(m)	User number
Without base station	L=80 W=80	100	30	N

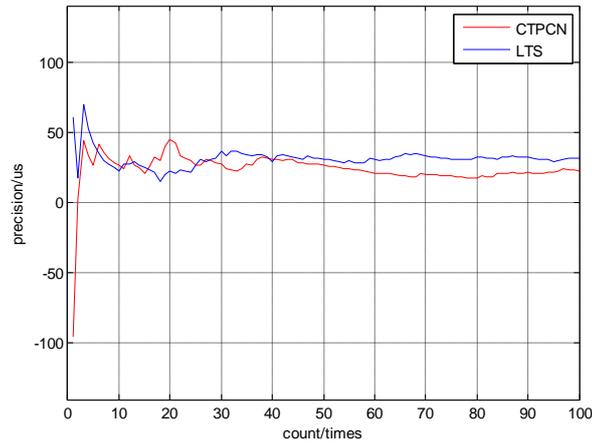


Fig.4. Synchronization accuracy for N=40

As we can see from Fig. 4, with the growth of count, the synchronization accuracy of the system tends to be stable. The accuracy of CTPCN algorithm is about 23 us, while the accuracy of LTS algorithm is about 30 us, so the proposed method has good performance than LTS. This result fully shows the benefit from the pre-synchronization process.

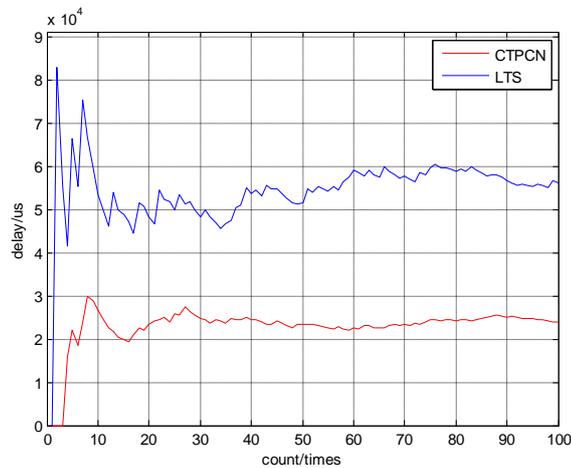


Fig.5. Synchronization time delay for N=40

As we can see from Fig. 5, with the growth of count, the synchronization time delay of the system tends to be stable. The time delay of LTS algorithm is about 57ms, while the time delay of CTPCN algorithm is about 24ms, so the proposed method has good performance than LTS. This result fully shows the benefit from the pre-synchronization process.

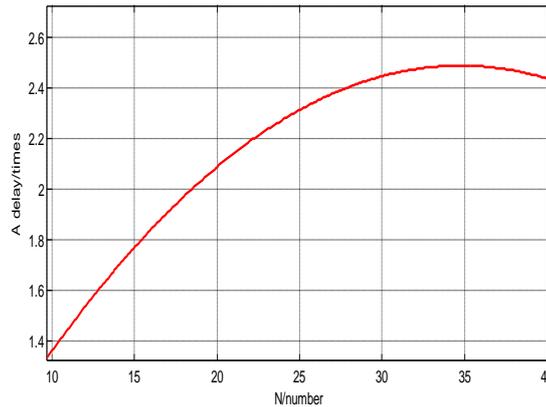


Fig.6. Synchronization time delay gain with N

Fig. 6 reflects the system time delay gain with the growth of UE number. We can know when UE is less than 35, the gain is increasing with the increasing of UE, but the gain falls when UE increase further. This is because the system information exists the possibility of collisions, and the possibility of the system information retransmission will increase with the increasing of UE, which leads to a decline in the gain.

Summary

In this paper, we introduce a combination time-synchronization protocol with center points (CTPCN) in Device-to-Device communication system. This technology can be realized without cellular network coverage. Through this mechanism, we can get many synchronous parts in advance which we call pre-synchronization to reduce system latency. Because one hop synchronization problem can be solved through our synchronous cluster, the system can achieve synchronization through the time synchronization protocol FTSP between different hop. The algorithm has good robustness, high precision, small time delay.

Acknowledgement

This paper is sponsored by ZTE Research Fund and National Key Technology R&D Program of China under grant No. 2014ZX03003011-004.

References

- [1] C.-H. Yu, et al., "On the performance of device-to-device underlay communication with simple power control," in Proc. IEEE Vehicular Technology Conference, Apr. 2009.
- [2] X. Wu, et al., "FlashLinQ: A synchronous distributed scheduler for peer-to-peer ad hoc networks," in Proc. Allerton, Oct. 2010.
- [3] J. J. van de Beek, M. Sandell, and P. O. Borjesson, "ML estimation of time and frequency offset in OFDM systems," IEEE Trans. Signal Processing, vol. 45, pp. 1800-1805, July 1997.
- [4] B. Chen, et al., "Blind estimation of OFDM carrier frequency offset via oversampling," IEEE Trans. Signal Processing, vol. 52, no.7, pp. 2047-2057, July 2004.
- [5] David L. Mills, "Internet Time Synchronization: The Network Time Protocol," IEEE Transactions on Communications, Vol. 39, No. 10, Oct 1991.
- [6] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless Sensor Networks: A Survey. Computer Networks, 38(4):393-422, March 2002.
- [7] Saurabh Ganeriwal, Ram Kumar, Sachin Adlakha and Mani Srivastava, Network-wide Time Synchronization in Sensor Networks, Technical Report UCLA, April 2002.
- [8] D.L. Mills, Improved Algorithms for Synchronizing Computer Network Clocks, Proc. of ACM SIGCOMM 94 Symposium, September 1994.

- [9] P. V. Rangan , H. M. Vin and S. Ramanathan "Communication Architectures and Algorithms for Media Mixing in Multimedia Conferences", IEEE/ACM Transactions on Networking, vol. 1, no. 1, 1993
- [10] A. Mutazono, M. Sugano, and M. Murata, "Evaluation of Robustness in Time Synchronization for Sensor Networks," in Bio-Inspired Models of Network, Information and Computing Systems, 2007. Bionetics 2007. 2 nd, pp. 89-92, 10-12 Dec. 2007.
- [11] C. H. Rentel and T. Kunz, "A mutual network synchronization method for wireless ad hoc and sensor networks," IEEE Transactions on Mobile Computing, vol. 7, no. 5, pp. 633-646, May 2008.
- [12] F. Ren, C. Lin, and F. Liu, "Self-correcting time synchronization using reference broadcast in wireless sensor network," IEEE Wireless Communications Magazine, vol. 15, no. 4, pp. 79-85, Aug. 2008.