Open access • Proceedings Article • DOI:10.1109/WD.2017.7918152

Data Freshness Aware Content-Centric Networking in WSNs — Source link

Ghada Jaber, Rahim Kacimi, Thierry Gayraud

Institutions: University of Toulouse

Published on: 29 Mar 2017 - IFIP Wireless Days

Topics: Content centric networking, Key distribution in wireless sensor networks, Wireless sensor network, Mobile wireless sensor network and The Internet

Data Freshness Aware Content-Centric Networking in WSNs

Ghada Jaber [1], Rahim Kacimi [1], Thierry Gayraud [2]
IRIT Université de Toulouse, CNRS, INPT, UPS, UT1, UT2J, France
[1]{gjaber, kacimi}@irit.fr
[2] gayraud@laas.fr

***Abstract*—The Content-Centric Networking (CCN) concept is a significant approach of several future Internet research activities. CCN in Wireless Sensor Networks present a promised technique that may ensure data routing based on content. In this paper, we focus on the lifetime of the Content Object exchanged in the network and we explain the idea of its integration in CCN for WSNs to better highlight the relevance of its exploitation. To this end, we implement DFCCN-WSNs 'Data Freshness aware Content-Centric Networking in Wireless Sensor Networks' a protocol that implements the data lifetime. Through extensive simulations, we demonstrate that DFCCN-WSNs outperforms traditional CCN in terms of end-to-end delay.**

***Index Terms*—Wireless sensor networks, content-centric networking, interest, data freshness, data-lifetime.**

I. Introduction

Wireless Sensors networks (WSNs) are considered as a promising element of the Internet o Things (IoT). WSNs in IoT are connected to the internet through gateways so information may be produced either on demand when another sensor requests it (query-based) or proactively sent to multiple subscribers (event-based).

Content-centric networking [5] is a new architecture of computer networks to replace nowadays host or node centric networking. It is built on named data where the identification and the transport of contents rely on their names and not on their location. Recently, CCN has gained a lot of attention in network research also in WSN community.

There is a lack of work in CCN that considers the lifetime of contents in cache. Indeed, nodes exchange data that may be obsolete in the network. In this paper, we are interested in improving CCN for wireless sensor networks by adding the lifetime of contents. We show that our solution improves end-to-end delay.

The remainder of this paper is organized as follows: in the next section, we list previous related work. CCN and its structure are described in section III. In section IV, we present simulation results to evaluate the performance of our solution. Section VI summarizes and concludes the paper.

II. Related Work

Different from recent popular proposed work of Content-Centric Networking (CCN), the main focus of the proposed DFCCN-WSNs is to integrate lifetime of contents.

CCNx [1] is based upon the Content-Centric Networking (CCN) architecture. It is built on named data where the content name replaces the location address. Every packet in the network can be cached at any CCNx router. So, CCNx delivers named content to the user from the nearest cache or content provider. This protocol is flexible and can be deployed in different environments where providing data content is an important concern.

NDN for WSNs [3] explores NDN potentialities in WSNs. In this work, the authors showed that NDN features match the use cases and applications developed on top of sensors and well cope with their potential constraints. They also enhanced NDN with packet overhearing to reduce collisions and duplicated transmissions. Besides, they extended it by principles inspired by the data-centric directed diffusion routing technique.

In [6], Yichao *et al.*, proposed the Content-Centric Routing (CCR) technology, where routing paths are determined by content. CCR provides content based information flow and optimized in-network data aggregation to avoid transmission of redundant network traffic, reduce delay, and realize energy saving. CCR is a distributed process, its objective is to optimize a routing structure.

In this paper, DFCCN-WSN, a protocol that integrates lifetime of content is implemented and evaluated.

III. Content-Centric Networking in Wireless Sensor Networks

In CCN, communication is based on two message types: Interest and Content Object. So when a node requests a content, it broadcasts an interest and the node that has the corresponding content responds with the content object.

A. Node model

In each node, some data structure are maintained to properly forward interest in the network:

- *Interface*: In CCN for WSNs, an interface is configured to send and receive a broadcast packet.
- *Content Store (CS)*: The Content Store is a buffer memory or cache where data is stored. CS is not

a persistent store, it holds content created locally or content object received from other nodes in the network. When finding a match in the CS, processing stops and the interest message is discarded.

- *Forwarding Information Base (FIB)*: If the node does not find a matching content in the CS it moves to check the FIB. FIB gives the interface on which the interest should be sent to retrieve a matching data. If finding a match in FIB, an entry is created in the PIT and the message is transmitted to the destination.
- *Pending Interest Table (PIT)* : PIT stores unsatisfied forwarded interests and contains a list of interfaces from which these interests have been received.

IV. DATA FRESHNESS AWARE CONTENT-CENTRIC NETWORKING

A. Data Freshness

For CCN-WSN, every node in the network has its own Content Store in which it caches contents. Initially, nodes create their content measured locally. When a user transmits an interest it starts by checking its CS and hereafter it transmits the interest to the other nodes in the network and the node with the corresponding content responds with the data located in its cache. If we don't check its lifetime, there is a risk that when it reaches the user it may be already expired and doesn't satisfy the user expectation.

In this context, we propose DDFCN-WSNs in which we implement the lifetime of the Content object (CO) recommended in the last version of CCNx but this time, it is for the adapted CCN for wireless sensor networks. We study its impact on the global system in terms of end-to-end delay.

B. Principle of DFCCN-WSNs

In DFCCN-WSNs 'Data Freshness aware Content-Centric Networking in Wireless Sensor Networks', a user broadcasts an Interest packet in the network. When the request arrives at a node, it begins by checking if it has the adequate content in its CS, if yes it checks if its lifetime didn't expire yet. If the data is still fresh it responds the user with this content object else it verifies if the content was created locally or received from another node. If it was created locally, content won't be sent since it is expired. Otherwise, it removes the content from its CS and broadcasts it in the network to look for it in the nodes in which it was created if it is updated.
So, data freshness has to be taken into account when matching an interest with the corresponding content. If the node finds the right content and if it is still fresh, intermediate node can send it. Otherwise, the user has to address his interest to the source which creates the content to recover the data. The different steps are detailed in Algorithm 1.

Algorithm 1: Requesting a fresh Content Object

```
input  : I: Interest message
         CS: Content Store
         PIT: Pending Interest Table
         FIB: Forwarding Interest Table
         data_lifetime: Content lifetime
output : CO: Content Object
Receive an Interest message I;
//If matching content found in CS, checks its lifetime,
  if it didn't expire yet, send the corresponding
  content
if (Content ∈ CS) then
    if ( data_lifetime didn't expire yet ) then
        Matching Content Object is transmitted;
    else
        //Check if it was created locally or received
        if (locally) then
            Data expired;
        else
            Remove Content Object from CS;
            Forward Interest;
            if (updated) then
                Transmit;
else
    if (matching Content ∈ PIT) then
        Add the arrival face to PIT;
    else
        if (matching Content ∈ FIB) then
            Create an entry in PIT;
            I is sent to the destination registered in FIB;
```

1) Impact of adding Content lifetime on Temporal redundancy: Temporal redundancy may be defined as the achievement of a specific action more than once over time followed by a checking result to increase reliability [4]. It is used to improve the accuracy of measurements by the sensor nodes to withstand transient faults during detection and communication. Redundancy provides robust and faulty tolerance information but results in energy wasting. Consequently, attributing a lifetime to a content may result in avoiding temporal redundancy thus saving energy. In the next session, we show through simulations that our solution does not only decreases end-to-end delay.

V. PERFORMANCE EVALUATION

In this section, we examine the performance of DFCCN-WSNs to see the impact of the lifetime on the delay.

A. Simulation Set-up

For the implementation of our solution, we chose to start with the code of ccnx_contiki [7] and to modify it to follow the requirements of our approach.

We consider a wireless sensor network deployed on a surface of 100×100 m. The number of nodes varies from 30 to 120. All the sensor nodes are static and have the same radius detection. In addition, all communication links are bidirectional. We consider that we have three different contents generated by nodes in the network N1, N2, and N3. Contents have a limited lifetime. When the lifetime expires, the nodes update their content. Lifetimes are 80s, 60s and 30s corresponding to N1, N2, and N3 respectively. We suppose that the clock nodes are synchronized.

The performance metric that we chose is the propagation delay that represents time between the broadcast of the interest and the recovery of the corresponding content object.

B. Simulation Results

In this scenario, we have one user who broadcasts its interests on the network and 30 nodes (10*N1, 10*N2, and 10*N3).

TABLE I: Delay for User1

Delay	1^{st} round	2^{nd} round
N1	165 ms	156 ms
N2	500 ms	447 ms
N3	274 ms	313 ms

An interesting observation comes from Table I that shows that for the content N2, the delay is equal to 500 ms and this is due to the network deployment.

Ordinarily, during the 2^{nd} round, contents will be available in the cache of nodes close to the user so the delay must be minimized comparing to the 1^{st} round. Hence, users can get the requested content from many nodes (the content's source or the intermediate nodes). But with the add of the content lifetime, this may not be always the case. The user sends an Interest for the content N2 at t=75 s and the lifetime of the content N2 is 60 s. Hence, it is found in the cache of intermediate nodes with expired lifetime so the interest is relayed to the source nodes which explains the delay for the content N2 during the 2^{nd} round.

TABLE II: Delay for User 1 if updates

Delay	1^{st} round	2^{nd} round
N1	165 ms	156 ms
N2	500 ms	187 ms
N3	274 ms	197 ms

Given the results shown in Table II, we observe if the update is realized in the intermediate nodes, the delay decreases comparing to the results presented in Table I. For the content N2, we gain 260 ms in terms of delay. So, for the content N3, we realized a gain of 116 ms.

Fig. 1 represents the average delay for interests and data response sent by nodes under varying network sizes 30, 60, 90, and 120 nodes. We note that the average delay during the first round for 30 nodes is about 183 ms and stays stable around 310 ms while increasing the network density.

If the content is just updated in source nodes, the user waits for about 236 ms to get the corresponding data when the network size is 30 and 325 ms when the density of the network increases. It is also worth noting that, if the update is also done in the intermediate nodes, the delay is about 152 ms for 30 nodes and 180ms for 90 nodes. Usually, during the 2^{nd} round, the user may get data

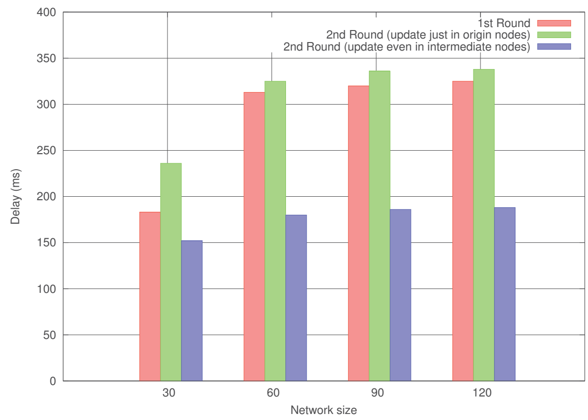

Fig. 1: Average delay

from the intermediate node faster. However, when it is not always fresh, it takes more time to get it from a source node where data was created locally.

VI. Conclusion

In this paper, we presented Content-Centric Networking for WSN and its structure. We also explained the concept of the content lifetime in sensor networks and we described our approach in which we proposed to add the lifetime of each content in the network so we satisfy the interest of user by ensuring the data freshness. We demonstrated that DFCCN-WSNs may realize user satisfaction by decreasing the response delay.

As a future work, we plan to implement DFCCN-WSNs on a real platform of neOCampus classrooms and conduct experiments to investigate interest generation model. Besides, we plan also to explore proactive and reactive solutions when the lifetime of contents expires.

Acknowledgment

This work is supported jointly by the neOCampus research grant [2] and the Occitanie Province.

References

[1] CCNx. http://blogs.parc.com/ccnx/.

[2] NeocampusLabs. https://www.irit.fr/neocampus.

[3] Marica Amadeo, Claudia Campo, Antonella Molinaro, and Nathalie Mitton. Named Data Networking: a Natural Design for Data Collection in Wireless Sensor Networks. In *IEEE Wireless Days (WD)*, Valencia, Spain, November 2013.

[4] Daniel-Ioan Curiac, Constantin Volosencu, Dan Pescaru, Lucian Jurca, and Alexa Doboli. Redundancy and its applications in wireless sensor networks: a survey. volume 8, pages 705–714. World Scientific and Engineering Academy and Society (WSEAS), 2009.

[5] Van Jacobson, Diana K Smetters, James D Thornton, Michael F Plass, Nicholas H Briggs, and Rebecca L Braynard. Networking named content. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, pages 1–12. ACM, 2009.

[6] Yichao Jin, Sedat Gormus, Parag Kulkarni, and Mahesh Sooriyabandara. Content centric routing in iot networks and its integration in RPL. *Computer Communications*, 2016.

[7] Bilel Saadallah, Abdelkader Lahmadi, and Olivier Festor. *CCNx for Contiki: implementation details.* PhD thesis, INRIA, 2012.