



HAL
open science

Reactive and proactive strategies for content update in content-centric and multi-users WSNs

Ghada Jaber, Rahim Kacimi, Thierry Gayraud

► **To cite this version:**

Ghada Jaber, Rahim Kacimi, Thierry Gayraud. Reactive and proactive strategies for content update in content-centric and multi-users WSNs. 13th International Wireless Communications and Mobile Computing Conference (IWCMC 2017), Jun 2017, Valencia, Spain. pp.1980-1985, 10.1109/IWCMC.2017.7986587. hal-03131867

HAL Id: hal-03131867

<https://hal.science/hal-03131867>

Submitted on 10 Feb 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Reactive and Proactive Strategies for Content Update in Content-Centric and Multi-users WSNs

Ghada Jaber¹, Rahim Kacimi¹, Thierry Gayraud²
IRIT Université de Toulouse, CNRS, INPT, UPS, UT1, UT2J, France
¹{gjaber, kacimi}@irit.fr
²gayraud@laas.fr

Abstract—Unlike other networks, WSNs have the particular characteristic of collecting sensed data and forwarding it to the gateway through a one-way communication protocol. In such data-driven networks, it may seem spontaneous to argue to content-centric networking communication architecture. Through exploiting DFCCN-WSNs implemented previously, we focus on two approaches concerning content update in CCN for Wireless Sensor Networks. Techniques used for content update are either reactive or proactive. In addition, our approaches support multi-user scenarios in order to decrease the propagation of the content object in the network. Through extensive simulations, we demonstrate that the two approaches outperforms traditional CCN in terms of end-to-end delay and energy consumption.

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

I. INTRODUCTION

The Internet of Things interconnects a large number of smart devices, and creates an environment wherein things have interfaces and identities and may communicate through standard and interoperable communication protocols [1]. IoT combines various network infrastructures such as sensor networks, wireless networks and Internet, in order to retrieve useful information from things, to interact with physical world, and to provide various services in different applications.

Wireless Sensors networks (WSNs) are considered as a promising element of the IoT. WSNs are very compliant to the expectations of IoT and this is due to their monitoring and data collecting capacities. For sensors in IoT, 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 [9] is a new architecture of computer networks to replace nowadays host or node centric networking. CCN can run over any technology including IP, and the reverse is also possible, i.e., anything can run over CCN, including IP [5]. Recently, CCN has gained a lot of attention in network research as well as in WSN community.

However, there is a lack of work in CCN that considers the lifetime of contents in cache. In our last paper [8],

we considered this point by integrating lifetime for each content in content store.

The contribution of this paper is twofold. First, we propose proactive and reactive approaches for content update in CCN for sensor networks. The second contribution is more general in that we modify CCNx_Contiki [14] to support multi-user scenarios. We show that our solution ensures reliability and improves end-to-end delay.

The rest of the paper is organized as follows: in the next section, we list previous related work. CCN in a wireless environment and its advantages are described in section III. The use case and the interest generation model are explained in section IV. Section V details the proposed approaches. In section VI, we present extensive simulation results to evaluate the performance of our approaches. Section VII 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 [8] is to integrate lifetime of contents. In this paper, we consider two paradigms that we apply to DFCCN-WSNs: proactive and reactive approaches for content update.

CCNx [2] is based upon the Content-Centric Networking (CCN) architecture. It is built on named data where the content name replaces the location address. Its services include multihop forwarding, flow control, and loop-free multipath forwarding. So, CCNx delivers named content to the user from the nearest cache or content provider. Applications use the CCNx on top of lower layer communication services that can handle packet transmitting [14]. This protocol is flexible and can be deployed in different environments where providing data content is an important concern.

Quevedo *et al.*, in [12], analyze the impact of the inherent multi-node caching mechanism of Information-Centric Networking (ICN) approaches considering the delay between the information generation and its consumption from cache. It also proposes a novel consumer driven information freshness mechanism to overcome the negative effect of the information freshness requirements in IoT-enabled ICN scenarios.

DFCCN-WSNs [8] integrates a lifetime to each content in the network and explores CCN potentialities in WSNs. In this work, we showed that by adding content lifetime, this framework will take into consideration data freshness.

The definition of proactive and reactive routing protocols is detailed in [7]. Proactive and reactive routing strategies are classified according to several criteria. In this paper, proactive and reactive approaches are used for content update in CCN for multi-user WSN.

In [10], authors designed a special query processing method for multi-user scenarios and they used Embedded Network Algorithm. NER resolved the problem of massive queries that do not indicate the query region in multi-users scenario but the event names. The embedded network used in this paper, restricts the distribution of queries within the event region.

In this paper, PDFCCN-WSN and RDFCCN-WSN, two approaches that realize content update and takes into account the interest model generation adapted to the use case, are implemented and evaluated.

III. CONTENT-CENTRIC NETWORKING IN WIRELESS SENSOR NETWORKS

In CCN, the communication is based on two types of message: 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. CCN in a sensor environment

Both interest and content object contain a content name. Interest contains also other information to properly identify the event and content object (CO) contains the requested data. Data must only be transmitted in response to an interest that matches the data.

Current sensor networks are built using different technologies that provide a selection of characteristics to suit any kind of sensor network. In wireless sensor networks research, data or content centric approaches like in-network processing and data aggregation are important [13].

B. Advantage of exploring CCN in WSNs

1) *Content caching*: In WSNs, CCN provides caching to reduce congestion and improve end-to-end delay. Caching insures high content availability, network traffic reduction, and low retrieval latency. Caching data may ensure store-carry-and-forward communications. CCN may cache both interest and content object. A node in the network may keep a copy of an interest in the PIT, so it will be able to response rapidly to the requester if it doesn't have the data. Data can also be found by local search in the Content Store (CS).

2) *Scalability*: CCN is particularly indicated to retrieve data from several nodes in a monitored Sensor Networks area. When the network density increases, CCN may prevent from degradation of the communication quality by controlling forwarding mechanism.

3) *Security*: Many of today's wireless sensor networks (WSN) are completely separated and isolated. CCN security mechanism may protect the information from this by securing the data and not the link.

IV. INTEREST MODEL AND DATA FRESHNESS

A. Use Case

In this work, we focus on everyday lives of the student and personnel on the Paul Sabatier University campus.

As a first step, we start with a neOCampus classroom [3]. This was created on the scientific campus of Paul Sabatier University. We deployed in the classroom sensors to measure temperature, humidity, and luminosity. We also equipped it with sensors that detect the presence of anybody in the classroom and actuators that control shutters. Thus, students and staff may send interests to have an idea about the temperature or the humidity of the room. A technician may have an idea about the state of shutters and if there are someone in the rooms after the end of the courses. The temperature of classrooms can give an idea to the technician if the heating machine is working or not.

B. Interest Generation Model

In the multi-user scenario of WSN, the monitored region is large-scaled with a different numbers of queries issued by various users and users are referred to those who just focus on their interests and ignore the network deployment [10]. To give an example, we consider classrooms which are monitored with the aid of WSN. For instance, students may raise some interests to have an idea about the temperature of classroom or detect the presence of anybody. The number of interests in CCN increases when the number of users increases. In CCN, users raise interests just based on their interest instead on geographic localization [11]. Our use case is designed to work in non-uniform interests environment, hence sometimes there are a lot of interests and sometimes there are few. Poisson distribution is not an appropriate choice for our scenario since its variance is equal to its expectation (equal to λ) [11]. Interests from users are independent among each other and the interest arrival process follows a normal distribution in a multi-user scenario. For instance, a large number of students are expected to send interests to have an idea about the location of classrooms during exams, with this case of example to illustrate the reasonableness of the normal distribution adopted. The number of interest arrivals in the n^{th} round behaves as the Binomial distribution, $B(N_q, p(n))$, where

$$p(n) = \int_{(n-1)\Delta T}^{n\Delta T} \frac{1}{\sqrt{2\pi}} \exp^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \quad (1)$$

is the probability that an interest arrives in the n^{th} round for the normal interest distribution, i.e. $N(\mu, \sigma^2)$, in our scenario. ΔT represents the duration of the round.

Hence, $k(n)$ denotes the expectation of the number of interest arrivals in the n^{th} round, as

$$k(n) = N_q p(n). \quad (2)$$

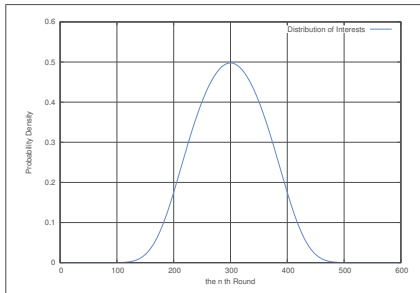


Fig. 1: Interest Distribution Model.

To look in more complicated and practical scenario, we assume that the events are queried with the normal distribution, $N(300, 0.32^2)$ shown in Fig. 1.

The broadcasted interests come from different points of entry into the network. The number of interests launched varies in the large-scaled WSNs to reflect the number of users. As depicted in Fig. 1, the time around the 300th round is the peak time of sending interests, it could be during the break time, while the left side and the right side denote the beginning and the end of the day.

The content lifetime have an impact on the interest generation model, since the response given to the interest dissemination in the network depends on data freshness.

C. Data Freshness

Data freshness concept has a relation with how old is the data? Is it fresh enough with respect to the user expectations? And it has been identified as one of data quality attributes in WSNs. Continuous monitoring applications require periodically refreshed data on the sink node. So it is important for the data to reach the sink node within a certain threshold.

WSNs are generally organized in a multihop topology since messages travel hop by hop; it is important to have high reliability. Data freshness ensures reliability and low latency since users may recover fresh data from the sink.

In [8], we proposed DDFCN-WSNs in which we implemented the lifetime of the Content Object (CO).

V. PROACTIVE AND REACTIVE APPROACHES FOR CONTENT UPDATE

In DFCCN-WSNs 'Data Freshness aware Content-Centric Networking in Wireless Sensor Networks' [8], 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 did not expire yet. 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. In this stage, we have two states: the first one is doing the update of all contents even in intermediate nodes (proactive approach) or waiting for an interest to update contents (reactive approach).

A. Proactive Approach

By proactive approach, we mean that all contents created by the nodes are updates when they expire regardless of the traffic (reception of an Interest). So, when a content expires in the source node, update is realized automatically. Intermediate nodes broadcast an Interest for updated contents when a content in its CS expires. Therefore, we enhance DFCCN-WSNs by adding this mechanism. The resulting protocol is called PDFCCN-WSNs 'Proactive Data Freshness Aware Content-Centric Networking for Wireless Sensor Networks'.

The advantage of proactive approach is that the content is immediately available so delay is optimized when a node needs to send the content. This also helps to optimize flooding.

B. Reactive Approach

For the reactive approach, in contrast, nodes send interests for a fresh content depending of the traffic (reception of an Interest). Hence, content update is realized when we receive an interest from a user. So when a node receives an Interest, 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 is not going to 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 we extend DFCCN-WSNs by RDFCCN-WSNs.

When the traffic is light and when the lifetime of contents is large, it would be intuitive that reactive approach would be energy efficient and inefficient when the traffic is heavy (which is the case in the scenario described previously).

C. Energy consumption

We suppose that a CCN user can forward interests to k CCN neighbors. Since the battery lifetime is very limited in sensors, we study the energy consumption during transmission, reception and sensing. We adopted the model in [6].

a) *Energy spent in transmission:* We compute the dissipated energy when sending interests and Content Objects. The energy spent in transmission is equal to: $e_d b d^\alpha + e_t b$.

Where e_d is the energy dissipated per bit per m^2 is chosen to be 100×10^{-12} , e_t is the energy spent by transmission

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

1 : Receive an Interest message  $I$ ;
   //If matching content found in  $CS$ , checks its lifetime,
   //if it didn't expire yet, send the corresponding
   //content
2 : if ( $Content \in CS$ ) then
3 :   if ( $data\_lifetime$  didn't expire yet ) then
4 :     Matching Content Object is transmitted;
5 :   else
6 :     //Check if it was created locally or received
7 :     if (locally) then
8 :       Data expired;
9 :     else
10 :      Remove Content Object from  $CS$ ;
11 :      Forward Interest;
12 :      if ( $updated$ ) then
13 :        Transmit;
14 : else
15 :   if ( $matching\ Content \in PIT$ ) then
16 :     Add the arrival face to  $PIT$ ;
17 :   else
18 :     if ( $matching\ Content \in FIB$ ) then
19 :       Create an entry in  $PIT$ ;
20 :        $I$  is sent to the destination registered in  $FIB$ ;
```

circuitry per bit and is chosen to be 50×10^{-9} and α is a constant which is upper than 2.

b) *Energy spent in reception*: We compute the dissipated energy when receiving interests and Content Objects. This energy spent in reception is equal to: e_l b.

Where e_l is the energy spent by reception circuitry per bit and is chosen to be 50×10^{-9} .

c) *Energy spent in Sensing*: The energy spent in transmission is equal to: e_s b.

Where e_s is the energy spent by sensing per bit and is chosen to be 50×10^{-9} .

In the next session, we show through simulations that our approaches do not only decreases end-to-end delay and insures reliability but also supports multi-user scenarios.

VI. PERFORMANCE EVALUATION

In this section, we examine the performance of RDFCCN-WSNs and PDFCCN-WSNs to see the impact of the content update on the network performances. The novelty in our scheme is the support of multi-user scenarios since previous works consider only one user via one sink [14]. Hence, the performance of our solution is evaluated under different number of users. To this end, we varied the number of users from 1 to 10 and we plotted the results.

A. Simulation Set-up

For the implementation of our solution, we chose to start with the code of CCNx_Contiki [14] and to modify it to follow the requirements of our approaches. Contiki [4] is

an open source operating system for the Internet of Things with memory-constrained embedded systems and wireless sensor networks.

For users, we suppose that there is an application agent behind who polls interests to ensure reliability.

TABLE I: Simulation Parameters

Simulation Parameters	Value
Number of nodes	30, 60, 90, 120
Number of users	1,3,5,7,10
Lifetime N1	80 s
Lifetime N2	60 s
Lifetime N3	30 s
Poll interest user1	15 s
Poll interest user2	10 s
Poll interest user3	20 s
Poll interest user4	10 s
Poll interest user5	55 s

We consider a wireless sensor network deployed on a surface of 100×100 m. The number of nodes varies from 60 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 types content 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. Simulation parameters are detailed in Table I.

B. Evaluation metrics

The performance metric that we adopted are:

- Propagation delay: Time between the broadcast of the interest and the recovery of the corresponding content object.
- Energy consumption: energy consumed by all nodes in the network.

C. Simulation Results

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

TABLE II: Delay for User with RDFCCN-WSNs

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 II that shows that for the content N2, the delay is equal to 500 ms and this is due to the network deployment. So, this can be explained by the fact that nodes that contain the content N2 are not in the range of the user. As for nodes that contain the content N3, they are in the range but very close to the border.

Ordinarily, during the 2nd round, contents will be available in the cache of nodes close to the user so the delay

must be minimized comparing to the 1st 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, it will depends on the approach that we are going to choose. The user sends an Interest for the content N2 at t=75 s and the lifetime of the content N2 is 60 s. When RDFCCN-WSNs is used, 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 2nd round. For the content N3 also, its lifetime is about 30 s and the interest was broadcast at t=90 s.

Based on the values reported in Table II, we notice that with RDFCCN-WSNs, we lost a little bit in terms of delay but we are sure that the data sent on the network is still fresh.

TABLE III: Delay for User with PDFCCN-WSNs

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 III, we observe that when we adopt PDFCCN-WSNs (realizing the update in the intermediate nodes), the delay decreases comparing to the results presented in Table II. For the content N2, we gain 260 ms in terms of delay. So, for the content N3, we realized a gain of 116 ms.

2) *Multiple users*: Here, we investigate the impact of supporting multiple users. To this end, in this scenario, we have 5 users and 90 nodes that generate 3 contents (N1, N2, and N3) with the same simulation parameters. Note that the old CCNx_Contiki [14] did not support multiple users.

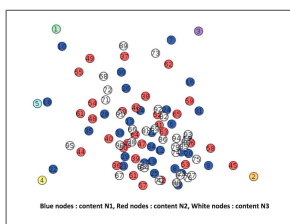


Fig. 2: Network deployment.

Once data collection starts, for every Interest sent by a user, only source nodes having a matching content, may satisfy the user expectation. As long as a Content Object is sent over the network, it is kept in the CS of the intermediate nodes.

Fig. 3 shows different delays for different users in the network. We notice that for users 2 and 4, the delays are very important since they were the two first users that

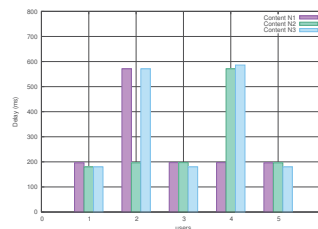


Fig. 3: Delay for different users during 1st round.

broadcast their interest on the network at t=10 s. At that time, only source nodes had their corresponding content. User 2 gets quickly the content N2 because a node that contains this content is in its range. The same applies for the content N1 for user 4. Once contents are exchanged over the network, the users who broadcast their interests later can get a response faster. For instance user 1 waited only for 196 ms to get the content N1. So, if we have multiple users on the network, delay may decrease since other users broadcast frequently interest.

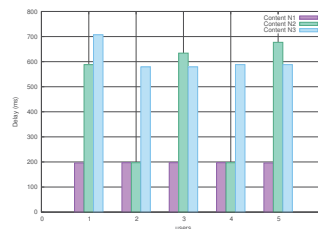


Fig. 4: Delay for different users during the 2nd round when RDFCCN-WSNs.

Usually, during the 2nd round, almost all the nodes have the three different contents in their content store. As a consequence, the delay is not important. However, as depicted in Fig. 4, user 1 waits for 707 ms to get a response for the content N3. This is because user 1 broadcast an interest for the content N3 at t=45 s, while content N3 lifetime is about 30 s. Hence, the nodes that have the content N3 in their CS have outdated data. Since RDFCCN-WSNs approach is adopted here, user 1 waits for more time to get a response from source nodes. Consequently, the delay increases. For the content N1, delay is smaller than other contents since its lifetime is significant and it was requested before its expiration.

When PDFCCN-WSNs is adopted, users may get a response faster. Fig. 5 shows the delays for different users during the second round if content update is realized in intermediate nodes. We notice that the delay for the content N3 for user 1 became 180 ms. So PDFCCN-WSNs approach realizes a gain of 527 ms. As far as for the content N2 for user 3, with RDFCCN-WSNs the delay is about 633 ms and after applying PDFCCN-WSNs it becomes 204 ms.

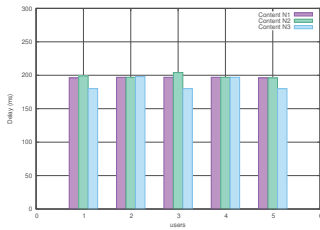


Fig. 5: Delay for different users during the 2nd Round when PDFCCN-WSNs.

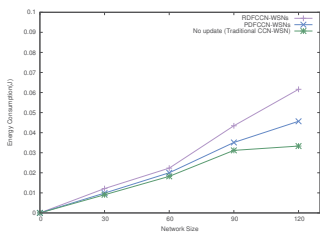


Fig. 6: Energy Consumption.

From the curves shown in Fig. 6, we noticed that the average energy consumption for the three methods increases with the density of the network, which is expected since every node consumes energy. The RDFCCN-WSNs approach consumes more energy because when content expires in an intermediate node, it has to look for content in source node by broadcasting the interest which consumes energy. However, traditional CCNx-Contiki does not consider content lifetime so content will be sent by the intermediate node without considering its freshness. So given the results shown in Fig. 6, we observe that we lost a little bit in terms of energy, for instance, 3 mJ for 30 nodes. But, a gain in term of reliability is realized since content freshness may satisfy the user expectation. Furthermore as mentioned before, in WSNs, continuous monitoring applications require refresh data at the sink node. So, it is important for the data to reach the sink node within a certain threshold. We may notice that for PDFCCNs-WSNs approach consumes less energy than RDFCCNs-WSNs. This is because when a node receives an Interest from a user, if the CO is available on the node, it will be always fresh since the node updated it when it expires so we don't have obsolete data in the node. Hence, we realize a gain in end-to-end delay. PDFCCN-WSNs achieves a gain of 0.02 J in terms of energy consumption comparing to RDFCCN-WSNs. We note also that in case of multi-users scenarios, the energy consumption increases with the number of interests, which is reasonable since every interest consumes energy.

VII. CONCLUSION

In this paper, we presented DFCCN-WSNs implemented previously which integrated the content lifetime in sensor networks. We described the interest generation model adapted to our use case and we proposed two approaches for content update: a proactive and a reactive one. We also demonstrated that PDFCCN-WSNs and RDFCCN-WSNs may realize user satisfaction by decreasing the response delay and energy consumption. We also show that our approaches can support multi-user scenarios unlike the old CCNx_Contiki [14].

As a future work, we plan to implement PDFCCN-WSNs on a the platform of neOCampus classrooms and conduct experiments to investigate real world interest generation models.

ACKNOWLEDGMENT

This work is supported jointly by the neOCampus research grant [3] and the Occitanie Province.

REFERENCES

- [1] 10 technologies that will change the world in the next 10 years. <http://www.networkworld.com/article/2179278/lan-wan/10-technologies-that-will-change-the-world-in-the-next-10-years.html>.
- [2] CCNx. <http://blogs.parc.com/ccnx/>.
- [3] NeocampusLabs. <https://www.irit.fr/neocampus>.
- [4] NeocampusLabs. <http://www.contiki-os.org/>.
- [5] Syed Hassan Ahmed, Safdar Hussain Bouk, and Dongkyun Kim. Content-centric networks (CCN). In *Content-Centric Networks*, pages 35–80. Springer, 2016.
- [6] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on*, pages 10–pp. IEEE, 2000.
- [7] Mohammad Ilyas. *The handbook of ad hoc wireless networks*. CRC press, 2002.
- [8] Ghada Jaber, Rahim Kacimi, and Thierry Gayraud. Data freshness aware content-centric networking in wsns. In *IEEE Wireless Days*, Porto, Portugal, 2017.
- [9] 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.
- [10] Xingcheng Liu, Zaili Ma, Shaochong Liu, and Xutao Cao. Query processing in multi-user scenario for wireless sensor networks. *IEEE Sensors Journal*, 11(10):2533–2541, 2011.
- [11] Xingcheng Liu, Zaili Ma, Shaochong Liu, and Xutao Cao. Query-driven tracing for a multiuser scenario in wireless sensor networks. volume 13, pages 3016–3024, 2013.
- [12] J. Rego, D. Corujo, and R. Aguiar. Consumer driven information freshness approach for content centric networking. In *IEEE The 2nd Workshop on Name Oriented Mobility - INFOCOM 2014 - NOM*, volume -, pages 482–487, April 2014.
- [13] Zhong Ren, Mohamed A Hail, and Horst Hellbrück. /ccn-wsn/-a lightweight, flexible content-centric networking protocol for wireless sensor networks. In *Intelligent Sensors, Sensor Networks and Information Processing, 2013 IEEE Eighth International Conference on*, pages 123–128. IEEE, 2013.
- [14] Bilel Saadallah, Abdelkader Lahmadi, and Olivier Festor. *CCNx for Contiki: implementation details*. PhD thesis, INRIA, 2012.