

On the Performance of Caching and Forwarding in Information-Centric Networking for the IoT

Mohamed Hail, Marica Amadeo, Antonella Molinaro, Stefan Fischer

► **To cite this version:**

Mohamed Hail, Marica Amadeo, Antonella Molinaro, Stefan Fischer. On the Performance of Caching and Forwarding in Information-Centric Networking for the IoT. 13th International Conference on Wired/Wireless Internet Communication (WWIC), May 2015, Malaga, Spain. pp.313-326, 10.1007/978-3-319-22572-2_23 . hal-01728806

HAL Id: hal-01728806

<https://hal.inria.fr/hal-01728806>

Submitted on 12 Mar 2018

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.



On the Performance of Caching and Forwarding in Information-Centric Networking for the IoT

Mohamed Ahmed M. Hail¹, Marica Amadeo² *,
Antonella Molinaro², and Stefan Fischer¹

¹ Telematics Institute University of Lübeck,
Ratzeburger Allee 160, 23562 Lübeck, Germany
{hail,fischer}@itm.uni-luebeck.de

² University Mediterranea of Reggio Calabria,
Via Graziella Loc. Feo di Vito, 89100 Reggio Calabria, Italy
{marica.amadeo,antonella.molinaro}@unirc.it

Abstract. *Information-Centric Networking* (ICN) is a communication paradigm for the future Internet that matches the pattern of information-oriented applications and provides in-network caching. Thanks to such properties, ICN has been recently considered as a solution for the IoT. However, IoT shows challenging features: many devices are resource-constrained and applications usually originate transient information. As a consequence, ICN-IoT solutions requires new design choices. Among them, it is crucial understanding if constrained devices can successfully implement ICN caching and forwarding functionalities, and what are the delivery performance.

To provide hints in such a context, in this paper we evaluate the performance of different caching/replacement policies and forwarding strategies in an ICN-IoT wireless network. A novel forwarding scheme is also defined to reduce the load on constrained devices. Simulation results show that the devised caching and forwarding operations are beneficial for delivery performance: they reduce the traffic volume and save the devices' energy resources.

Keywords: Information-Centric Networking, Internet of Things, Caching, Forwarding

1 Introduction

The Internet of Things (IoT) envisions a future in which everyday objects undergo a transformation into smart entities able to interact with each other and/or with humans therefore enabling new classes of applications in different domains, like smart home, smart city and smart transportation [13]. But unlike traditional networks, IoT is a highly heterogeneous and challenging environment, characterized by both resource-constrained and more powerful devices, heterogeneous wireless and wired access technologies and peculiar traffic patterns.

* Corresponding author.

Several research efforts and standardization activities are underway to define proper networking solutions for the IoT. In addition to *evolutionary* IP-based approaches, e.g., 6LoWPAN, the research community is considering a novel communication paradigm called Information Centric Networking (ICN), aka Content Centric Networking (CCN), which has been specifically conceived for data dissemination from information-oriented applications [4] and may also offer several advantages in IoT. In ICN each content is named uniquely and independently from the location of the producer thus facilitating caching at intermediate nodes. A consumer requests a content by sending the so-called *Interest* packet, which carries the content name; the producer or any node that maintains a copy of that content may answer the request. By enabling anycast, connectionless delivery and in-network caching, ICN allows to decouple senders and receivers, speed up data retrieval and limit the massive data access to the same producer [5].

But research on ICN-IoT systems is still at its infancy and there are several open challenges to address. IoT data are usually *transient*: they can be frequently updated by the producer, and also the consumers could be interested in data with a specific *freshness* requirement [15]. This complicates the caching operation. Moreover, the ICN networking functionalities, namely advanced packet processing, caching and forwarding, can be quite expensive for *constrained wireless IoT nodes* like sensors. When a multi-hop scenario is considered, in fact, an ICN node receiving an Interest must decide whether *forwarding* it or not towards the producer. If the Interest is transmitted and a content packet is then received, the node should decide if *caching* the content before *forwarding* it back to the original consumer. When the content is cached, the node can act as a source and answer to future requests. As a result, ICN communication could be highly energy consuming for resource-constrained devices.

The ICN Research Group (ICNRG) of the Internet Research Task Force (IRTF) is currently investigating what could be the role of constrained nodes in ICN-IoT systems. As discussed in [11], a design choice may be to separate IoT functionalities (e.g., sensing and measurement transmission) from ICN functionalities. The constrained node could implement only the IoT functionalities and let ICN tasks to the powerful nodes. However, results in [8] obtained with a preliminary testbed showed that caching is very useful in IoT because it shortens the path length to the content and reduces the network load. In summary, to enable Interest forwarding and consequent data caching in constrained nodes has pros and cons that deserve to be investigated. This is indeed the objective of this paper, which explores the potentials of in-network caching and forwarding in wireless ICN-IoT systems.

In this context, it is the authors' opinion that IoT nodes should perform a *simple distributed caching scheme* and an *adaptive Interest forwarding (AIF) strategy* that considers the status of the node resources (e.g., energy level), which vary over time. Nodes with higher capabilities (e.g., fully charged) have priority in the Interest forwarding and they are candidate to cache the data; constrained nodes, instead, can implement ICN functionalities only if it is strictly necessary,

e.g., in a sparse network, where their participation is fundamental to connect different network segments.

After discussing caching and forwarding strategies in ICN-IoT systems by referring related literature, in this paper we present the AIF strategy and deploy an extensive simulation campaign by means of the reference simulation tool of the ICN community, ndnSIM [3]. Obtained results show that a simple probabilistic caching scheme coupled with the Least Recently Used (LRU) policy can be particularly beneficial in ICN-IoT systems, where consumers are usually interested in the freshest copy of the content. But the benefits of caching can be further improved by using proper forwarding schemes like AIF, which is aware of the node's resources.

The rest of the paper is organized as follows. Section 2 describes the ICN paradigm; Section 3 discusses ICN caching and its applicability in IoT, while Section 4 introduces the forwarding process and presents the AIF strategy. Section 5 discusses the simulation results. Finally, Section 6 concludes the paper.

2 Information Centric Networking: general concepts

ICN replaces the *host-centric conversational model* of the TCP/IP protocol suite with *information-centric communications* where named data are disseminated in the network in a connectionless fashion. Different ICN architectures have been proposed so far [4], which share the same networking principle. Without loss of generality, in the following we refer to the Named Data Networking (NDN) [20] architecture to show the content retrieval procedure by focusing on forwarding and caching operations.

NDN communication is initiated by the consumer which transmits an Interest to request a Data packet by name. Every node maintains three data structures: *(i)* the Content Store (CS) that caches incoming Data; *(ii)* the Pending Interest Table (PIT) that keeps track of the forwarded Interest(s) that are not yet satisfied; *(iii)* the Forwarding Information Base (FIB) that is used as a routing table to relay Interests towards content source(s). When an Interest is received, the node first looks in its CS to find a matching Data packet and send it back. If the Data is not found in the CS, the node looks in the PIT. If a matching PIT entry is found, it means that an equal request has been already forwarded and the node is waiting for the response. Therefore, the entry is updated with the arrival interface and the Interest is *discarded*. Otherwise, a new PIT entry is created and the Interest is forwarded to the interface stored in the FIB.

The Data packet follows the chain of PIT entries back to the consumer(s), and it can be cached in intermediated nodes.

The NDN forwarding fabric together with the caching operation can provide several benefits in IoT scenarios. The PIT look-up is specifically designed to deal with massive data access and reduce the traffic load. In fact, by identifying multiple Interests for the same Data and discarding duplications, intermediate nodes forward only a single request to the producer. Retrieval performance is

further improved by in-network caching, which makes data available closer to consumers also under intermittent connectivity, e.g., due to low-power operation.

However, forwarding and caching mechanisms need to be further investigated and properly extended to cope with the challenges of IoT wireless systems, as discussed in the following.

3 In-network caching

3.1 Caching strategies and replacement policies

The caching system in a node consists of two main parts: the caching decision strategy and the replacement policy.

The default caching strategy in ICN is the so-called ALWAYS (aka universal) caching, where all incoming packets are cached by all nodes without any actual selection criterion. This simple strategy, however, has been shown inefficient because it generates high redundancy without maximizing the data diversity in the network [9]. Therefore, other solutions are usually used in the literature as benchmark, including PROBABILISTIC caching [17], where the node randomly caches incoming Data with a certain probability p .

Several advanced caching strategies have been proposed so far to minimize redundancy and maximize the cache hit ratio [19]. However, they are not designed for IoT scenarios, they mainly target Internet contents with Zipf distributed request patterns and propose schemes with *implicit* or *explicit* coordination, which leverage information about network topology or content characteristics, e.g., popularity. For instance, in [14], a probabilistic strategy is proposed where the caching probability is inversely proportional to the distance between the requester and the caching node. If this latter is closer to the requester, the content will be copied with higher probability thus quickly pushing it to the network edge and reducing the number of copies. Vice versa, an explicit coordination scheme called Cooperative In-Network Caching is proposed in [10], where nodes in the neighbourhood cooperate to cache the data without duplications. When a Data packet arrives at a caching node, it uses a hash function to determine in which of its neighbours (including itself) to cache the packet. However, in [16], the authors argue that explicit coordination policies would likely violate ICN line of speed constraint, while simple randomized caching policies could perform almost as well as more complex ones.

The most common replacement policies in ICN are LEAST RECENTLY USED (LRU), LEAST FREQUENTLY USED (LFU) and RANDOM. LFU is computationally more expensive than LRU, which is therefore considered for line speed operations. To maintain ICN simple and scalable, the RANDOM replacement, which is agnostic of the frequency and timing of requests, is sometimes preferred [16].

3.2 Caching in ICN-IoT

Caching strategies defined for Internet contents and wired networks are not generally suited to IoT. As a matter of fact, IoT contents are different from

Internet contents because they are usually transient and small, and IoT devices are different from Internet hosts because they have usually constraints in terms of energy and storage capabilities, e.g., the CS size is only of a few kbytes [8].

So far, only a very limited number of works have studied in-network caching and related issues in IoT.

The importance of the freshness parameter from the perspectives of both producers and consumers have been recently discussed in [15]. The authors argue that in presence of transient IoT contents both Interest and Data packets should carry a freshness value to perform accurate caching and retrieval operations. On one hand, stale contents are automatically discarded from a CS that honours the freshness declared by the producer; on the other hand, a CS matching can be considered valid if the freshness requested by the consumer is in accordance with the freshness of the cached data.

The work in [18] proposes a distributed probabilistic caching algorithm for IoT contents in the Internet, where routers dynamically update their caching probability by considering information about the network topology, the data freshness, the rate and freshness requirement of incoming requests. However, by taking into consideration those large set of parameters and routers with large storage capacity, the proposed coordination scheme is not well suited for low-power networks and constrained devices. Vice versa, in [8], the focus is on a wireless IoT system where nodes have limited storage. By leveraging a real testbed, the authors compare data retrieval with ALWAYS caching and without caching. They find that, thanks to in-network caching, the number of radio transmissions is reduced by up 50% thus proving that the caching operation is useful even in presence of nodes with limited capabilities.

In this paper, we extend through simulations the analysis started in [8] with the objective of understanding the performance of the standard caching and replacement policies in wireless ICN-IoT systems.

4 The forwarding decision

4.1 Controlled flooding

IoT systems are often characterized by battery-powered, sometimes mobile, nodes that communicate through the unreliable broadcast shared medium in a multi-hop fashion. In such environments, FIBs are usually not populated in advance by routing information and Interests are broadcasted hop-by-hop with controlled flooding mechanisms, as discussed in [6], [8], [7] and reference therein. Flooding is indeed a simple and robust solution in finding a path to the requested content even under intermittent connectivity: the Interest packet is broadcasted over the wireless medium by the consumer and re-broadcasted by intermediate nodes until a producer is located; then, the Data packet is transmitted by the producer and forwarded by intermediate nodes that maintain a PIT entry.

To reduce overhead and redundancy, the forwarding strategy is enhanced with timer-based packet suppression techniques that exploit packet overhearing,

i.e., the node defers the packet of a specific delay and drops it if the same packet is overheard over the broadcast medium. The defer (and listening) time can be selected in different ways: it can be randomly computed, or based on some specific information about the network topology, if available. For instance, if nodes are aware of their distance to the producer, the forwarding scheme can prioritize the nodes closer to the producer with shorter defer times [7], [12].

Controlled flooding can also be used as the initial discovery phase to set a temporary FIB entry on the reverse path taken by the first Data packet [8]. After the producer is discovered, subsequent Interests can be unicast using the established path, while flooding is used again only if the producer becomes unreachable. Unicast transmissions, however, hinder the possibility of sharing information over the wireless channel.

4.2 Adaptive Interest forwarding

In this work, we present a simple forwarding strategy that implements controlled flooding but uses a novel technique for the calculation of the defer time. Our focus, in fact, is on constrained IoT devices, which should implement ICN forwarding depending of their capabilities like the energy level, whose value varies over time making the device less (or more) prone to participate in ICN functionalities.

The rationale of the designed defer time calculation is to prioritize forwarding from the nodes with higher capabilities, while the constrained nodes calculate longer defer times and transmit only if they do not overhear the Interest transmission from the neighbours. This should limit the number of packet transmissions, which are indeed the main cause of energy consumption.

The proposed *Adaptive Interest Forwarding* (AIF) strategy extends the so-called Blind Forwarding (BF) presented in [6]. It assumes that Interest and Data rebroadcasting events are deferred by $T_{Interest}$ and T_{Data} time values, respectively, and Data have always higher access priority than Interests. Both defer times are randomly computed by considering a defer window (DW), an integer value that indicates the maximum length of the time intervals, as follows:

$$T_{Data} = rand[0; DW - 1] * DeferSlot \quad (1)$$

$$T_{Interest} = (DW + rand[0; DW]) * DeferSlot \quad (2)$$

where *DeferSlot* is a fixed, short time interval³.

During $T_{Interest}$, the potential forwarder listens to the channel: if it overhears the requested Data or the same Interest, then it cancels its own transmission and the related PIT entry.

The AIF strategy maintains the logic of the Blind Forwarding but modifies the $T_{Interest}$ calculation in order to set the defer time of a constrained node in a way that is proportional to its available resources, whose values are usually

³ Different values for DW and *DeferSlot* parameters are considered in [6]. In this paper, the selected values are reported in Table 1.

monitored by the node itself. We do not consider additional parameters (e.g., about the network topology or the neighbourhood) because we do not want to introduce additional complexity (e.g., signalling or coordination).

In this paper we focus on a single resource, the energy level of the node, En , but the algorithm can be easily extended to include also other inputs. En is modelled as a normalized parameter with $0 \leq En \leq 1$, where the values 0 and 1 mean that the battery is fully discharged or charged, respectively. The Interest defer time is then calculated as:

$$T_{Interest} = (DW + \Gamma_{En} + rand(0; DW)) * DeferSlot \quad (3)$$

where the term $\Gamma_{En} = (1 - En) * DW$ is the defer time extension related to the energy level of the node.

By following the above computation, the nodes with higher energy level will transmit with higher priority, while the others will not participate in the forwarding and, consequently, they will not be involved in the caching decision of the correspondent Data packet. Such advantages come at the expense of only a little energy consumption due to the overhearing operation.

5 Performance evaluation

5.1 System description

To evaluate the impact of caching and forwarding in wireless ICN-IoT systems, we use the ns-3 simulator [1] and the open-source ndnSIM [3] module, specifically designed to support NDN networks.

Our scenario is a $400m \times 400m$ area with 36 nodes deployed in a lattice topology, as shown in Fig. 1. Each node uses the IEEE 802.11g radio interface, where physical and medium access control parameters are set according to [2], a small form factor, low power networking module available on the market. The distance between two adjacent nodes is 80m and the nominal coverage range of each node is about 120m. However, the received signal power can vary during the simulation because the Rayleigh fading distribution is selected to reproduce realistic propagation in urban environments.

Four consumers, each one located in a corner of the square, are interested in the retrieval of contents with specific *freshness* values from a set of nodes in the topology. Six nodes are randomly selected as producers, which generate *transient* contents, each one composed by N_d Data packets, i.e., the consumer has to transmit N_d different Interests to obtain the overall information. Each consumer periodically sends novel Interests to obtain the fresher (novel) information. We assume the update period is 1 minute (e.g., the consumer requests the content at $t = 0s$, $t = 60s$, $t = 120s$, etc., and each time the Interests also carry the freshness requirement).

Performance evaluation is divided into two main parts.

First, we focus on the caching performance in presence of devices with very small CSs. Since our study targets constrained wireless devices, we do not consider burdensome caching schemes based on explicit or implicit cooperation. We

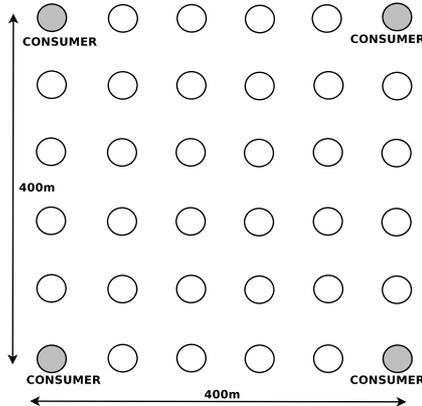


Fig. 1: Simulation scenario: lattice topology with 36 nodes.

instead focus on two simple decision strategies, ALWAYS and PROBABILISTIC with $p = 0.5$, coupled with two replacement policies, LRU and RANDOM. The LFU policy is not included in the analysis since it is computationally more complex than LRU and RANDOM.

Then, we focus on the forwarding schemes (BF and AIF) and their effects on the traffic load and the network energy consumption. To this purpose, the energy model in ns-3 is used to calculate the energy consumption in the simulated scenario. Simulation results are averaged over 10 independent runs.

Table 1 summarizes the main simulation parameters.

5.2 In-network caching analysis

The objective of this first study is to evaluate the impact of small CSs over content retrieval. Therefore, we vary the CS size from 1 to 4 kbytes. Nodes are fully charged and do not run out of energy during the simulation, which lasts 10 minutes during which the four consumers request contents according to the previously described pattern. Packet forwarding is based on BF, without the AIF extension.

As performance metrics, we consider: (i) the *cache hit ratio*, which indicates (in percentage) the number of times a Data packet is served by a CS instead of the original producer; (ii) the average *Data retrieval delay*, which is computed as the time since the consumer has sent the Interest to the reception of the Data and includes possible Interest retransmissions; (iii) the average *Interest retransmissions*, which indicates (in percentage) the number of times a consumer has to re-transmit an unsatisfied Interest.

Fig. 2 shows the performance metrics when varying the CS size and the caching/replacement policies. As expected, the greater the CS size the better the performance of the considered caching strategies, e.g., the cache hit ratio with ALWAYS(LRU) passes from 6% to 31%, when the CS size varies from 1 to

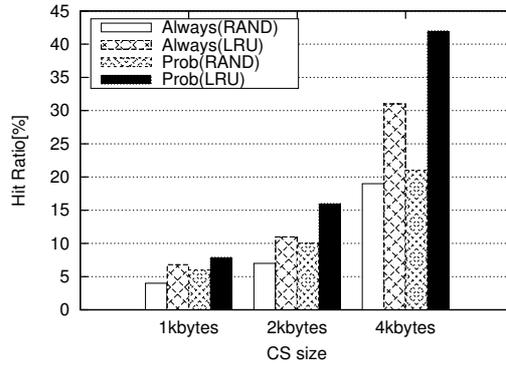
Table 1: Simulation Parameters

Category	Parameter	Value
Application	Data payload size	256 bytes
	Data freshness	rand(5:20)
	Consumer's freshness	rand(1:10)
	N_d	15
	Update period	1 minute
Scenario	Coverage area	400m x 400m
	Number of nodes	36
	Distance between nodes	80m
	Number of consumers	4
	Number of producers	6
NDN	CS size	1-4 kbytes
	Caching strategy	ALWAYS, PROBABILISTIC (with $p = 0.5$)
	Replacement policy	LRU, RANDOM
	Forwarding strategy	Controlled flooding with (or w/o) AIF $DW = 255$ $DeferSlot = 28\mu s$ (= IEEE 802.11g DIFS)
Access	Technology	IEEE 802.11g
	Rx sensivity	-84dBm
	Propagation model	Rayleigh

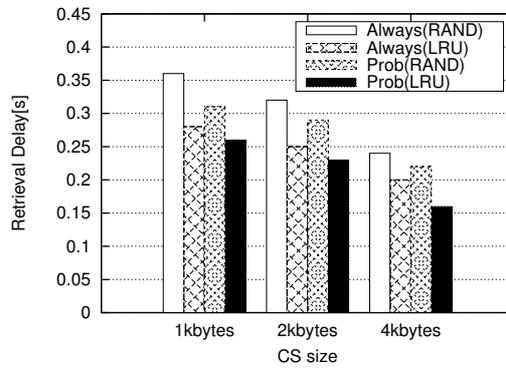
4 kbytes, respectively (Fig. 2(a)). Larger CSs allow nodes to cache more data and increase the possibility to find a cached copy closer to the consumer thus shortening the retrieval delays (Fig. 2(b)). As a further result, the transmissions over the lossy wireless channel and the packet collision probability are reduced, which correspond to limited Interest retransmissions (Fig. 2(c)).

When focusing on the replacement policies, an important trend is that LRU performs better than RANDOM when both PROBABILISTIC (shortened as *Prob* in Fig. 2) and ALWAYS strategies are considered. This is due to the fact that IoT consumers are generally interested in fresher data, and the LRU policy naturally favours fresher contents, by discarding the least recently used packets. Therefore, a LRU policy is recommended in IoT systems that honour the freshness requirement.

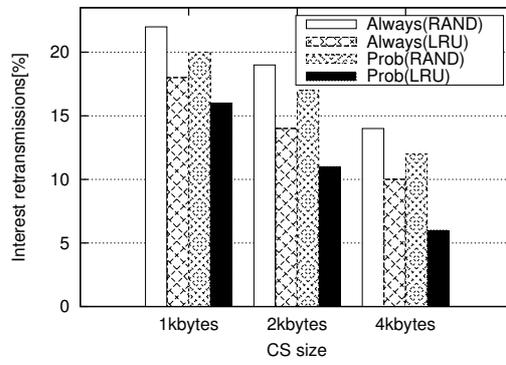
Under the same replacement policy, it can be observed that PROBABILISTIC caching always performs better than ALWAYS. In particular, PROBABILISTIC caching coupled with LRU gives the highest performance in terms of hit ratio and, consequently, in terms of retrieval delay and Interest retransmissions. This result is due to the fact that the ALWAYS strategy generates high redundancy in the network, a condition already pinpointed in general literature about ICN caching [17]. In presence of small CSs, the negative effects of the ALWAYS strategy are even exacerbated. Vice versa, a PROBABILISTIC scheme improves the data diversity in the network and provides a more efficient utilisation of available cache resources.



(a) Cache hit ratio.



(b) Data retrieval delay.



(c) Interest retransmissions.

Fig. 2: Caching analysis: metrics varying the cache size, the caching strategy and the replacement policy.

5.3 Forwarding analysis

Our second evaluation study compares BF and AIF schemes and it is structured into two parts: (i) energy efficiency and (ii) delivery performance analysis.

We select as caching strategy the PROBABILISTIC scheme coupled with LRU, which obtained the highest performance in the previous analysis. The CS size is fixed to 2kbytes.

Our first focus is to estimate the energy efficiency of the forwarding by considering the *network lifetime*, which is defined as the time interval during which the network is connected and there is at least one path between the consumers and the requested contents. When nodes start to run out of energy the communication opportunities reduce as well and, eventually, the data retrieval becomes totally infeasible. The ns-3 energy model allows to specify the initial energy of each device and during the simulation such value decreases on the basis of the activity of the node. In the experiment, the four consumers are fully charged, while the rest of the nodes are provided with low levels of initial energy and they are meant to quickly run out of energy. Simulation ends when such nodes have exhausted their energy and data retrieval becomes infeasible.

Table 2 shows the network lifetime together with the retrieval delay and the percentage of Interest retransmissions for BF and AIF schemes.

Table 2: Comparison of BF and AIF in low-energy conditions

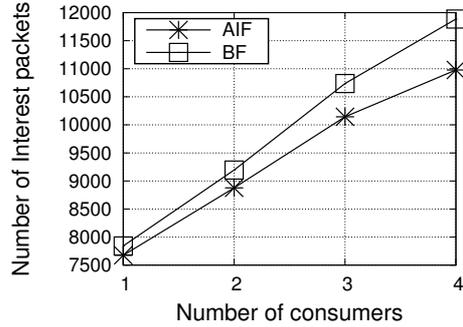
Metric	BF	AIF
Network lifetime [s]	366	425
Retrieval delay [s]	0.23	0.24
Interest retransmissions	12%	9%

It can be observed that AIF, by enabling an energy aware forwarding, can improve the network lifetime w.r.t. BF. In fact, nodes with a higher energy level transmit first and let the others to save their resources. AIF can also automatically *balance* the charge in the network because when a node decreases its energy level, due to a high number of transmission operations, its defer time $T_{Interest}$ increases as well. As a result, its transmission opportunities decrease.

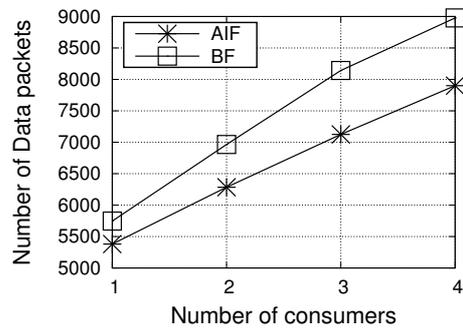
This advantage is at the expense of a slightly higher retrieval delay, which is due to generally longer defer times. However, longer delays have a beneficial side effect: they limit the packet collision probability and therefore reduce the number of Interest retransmissions from consumers.

To obtain further insights into the delivery performance of AIF, we consider a different simulation campaign where we vary the number of consumers from 1 to 4. We assume that nodes are provided with different levels of initial energy, which is sufficiently high to guarantee that they do not run out of energy during the simulation, which last 10 minutes.

Fig. 3 shows the number of Interest and Data packets transmitted by all the nodes in the network (i.e., consumers, forwarders, producers) during the simula-



(a) Number of Interest packets.



(b) Number of Data packets.

Fig. 3: Traffic load varying the number of consumers.

tion. It can be observed that AIF is able to reduce the overall traffic load since it originates a lower number of Interest and Data packets, if compared with BF. This is due to the fact that the $T_{Interest}$ computation includes the additional term Γ_{En} , which increases the listening period and reduces the number of concurrent transmissions. Less transmissions also imply less energy consumption, with consequent advantages in terms of network lifetime.

The price to pay, as briefly discussed above, is the slightly higher retrieval delay. However, this is not a big matter since, as shown in Fig. 4, the difference in the AIF and BF retrieval delay is less than $10ms$.

6 Conclusion

In this paper, we have studied in-network caching and forwarding in wireless ICN-IoT systems characterized by constrained devices. Simulation results obtained with ndnSIM have shown that ICN caching is very useful in IoT, even if the content store of a constrained device is only of a few kilobytes. In particular, by facilitating the data diversity in the network, PROBABILISTIC caching can

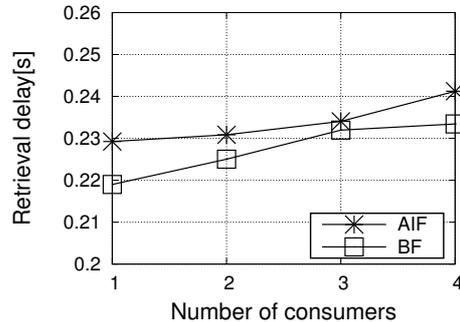


Fig. 4: Data retrieval delay when varying the number of consumers.

improve the cache hit ratio w.r.t. the ALWAYS caching strategy, while the LRU replacement policy results very suited to IoT because it naturally matches the freshness requirements of consumers.

To improve the performance of Interest forwarding in presence of constrained devices, the AIF scheme has been proposed, which considers the node energy level. Such awareness allows AIF to outperform the Blind Forwarding algorithm by guaranteeing longer network lifetime and reducing the traffic load.

As future work, we plan to further extend the simulation campaign by considering mobile scenarios and different content request patterns and node density values.

7 Acknowledgement

This work was supported by a fellowship within the FITweltweit programme of the German Academic Exchange Service (DAAD) and by the national research project PON03PE_00050 DOMUS “Home automation systems for a cooperative energy brokerage service”.

References

1. The network simulator-3 (ns-3), <http://www.nsnam.org/>
2. RN-171 802.11 b/g Wireless LAN Module, Rowing Networks, Datasheet available on line at: <http://www.microchip.com/>.
3. Afanasyev, A., Moiseenko, I., Zhang, L.: ndnSIM: NDN simulator for NS-3. Technical Report NDN-0005, NDN (October 2012)
4. Ahlgren, B., Dannewitz, C., Imbrenda, C., Kutscher, D., Ohlman, B.: A Survey of Information-Centric Networking. *IEEE Communications Magazine* 50(7), 26–36 (2012)
5. Amadeo, M., Campolo, C., Iera, A., Molinaro, A.: Named Data Networking for IoT: an Architectural Perspective. In: European Conference on Networks and Communications (EuCNC). Bologna, Italy (2014)

6. Amadeo, M., Campolo, C., Molinaro, A.: Forwarding strategies in named data wireless ad hoc networks: Design and evaluation. Elsevier Journal of Network and Computer Applications (2015)
7. Amadeo, M., Campolo, C., Molinaro, A., Ruggeri, G.: Content-centric wireless networking: A survey. Computer Networks 72, 1–13 (2014)
8. Baccelli, E., Mehlis, C., Hahm, O., Schmidt, T.C., Wählisch, M.: Information Centric Networking in the IoT: Experiments with NDN in the Wild. ACM Conference on Information-Centric Networking (ICN-2014) (2014)
9. Chai, W.K., He, D., Psaras, I., Pavlou, G.: Cache "less for more" in Information-Centric Networks (extended version). Elsevier Computer Communications 36(7), 758–770 (2013)
10. Li, Z., Simon, G.: Time-shifted TV in Content Centric Networks: the Case for Cooperative In-Network Caching. In: IEEE ICC (2011)
11. Lindgren, A., et al.: Applicability and Tradeoffs of Information-Centric Networking for Efficient IoT. In: Internet-Draft (January 2015)
12. Meisel, M., Pappas, V., Zhang, L.: Ad hoc networking via named data. In: ACM international workshop on Mobility in the evolving internet architecture. pp. 3–8 (2010)
13. Miorandi, D., Sicari, S., De Pellegrini, F., Chlamtac, I.: Internet of Things: vision, applications and research challenges. Elsevier Ad Hoc Networks 10(7), 1497–1516 (2012)
14. Psaras, I., Chai, W.K., Pavlou, G.: Probabilistic In-network Caching for Information-centric Networks. In: ICN Workshop on Information-centric Networking. pp. 55–60 (2012)
15. Quevedo, J., Corujo, D., Aguiar, R.: Consumer Driven Information Freshness Approach for Content Centric Networking. In: IEEE Infocom NOM Workshop (2014)
16. Rossi, D., Rossini, G.: Evaluating CCN multi-path Interest forwarding strategies. vol. 36, pp. 771–778 (2013)
17. Tarnoi, S., Suksomboon, K., Kumwilaisak, W., Ji, Y.: Performance of Probabilistic Caching and Cache Replacement Policies for Content-Centric Networks. In: IEEE 39th Conference on Local Computer Networks (LCN) (2014)
18. Vural, S., Navaratnam, P., Wang, N., Wang, C., Dong, L., Tafazolli, R.: In-network Caching of Internet-of-Things Data. In: IEEE International Conference on Communications (ICC). pp. 3185–3190 (2014)
19. Zhang, G., Li, Y., Lin, T.: Caching in Information Centric Networking: a Survey. Computer Networks 57(16), 3128 – 3141 (2013)
20. Zhang, L., et al.: Named Data Networking (NDN) Project. Tech. Rep. NDN-0001, PARC (October 2010)