Information Centric Networking in the IoT: Experiments with NDN in the Wild

Emmanuel Baccelli, Christian Mehlis, Oliver Hahm, Thomas Schmidt, Matthias Wählisch

Information Centric Networking in the IoT: Experiments with NDN in the Wild

E. Baccelli, C. Mehlis, O. Hahm, T. C. Schmidt, M. Wählisch

ISSN 0249-6399 ISRN INRIA/RR--8551--FR+ENG

RESEARCH REPORT

N° 8551

June 2014

Project-Team HiPERCOM

Information Centric Networking in the IoT: Experiments with NDN in the Wild

E. Baccelli, C. Mehlis, O. Hahm, T. C. Schmidt, M. Wählisch

Project-Team HiPERCOM

Research Report n° 8551 — June 2014 — 18 pages

Abstract: This report explores the feasibility, advantages, and challenges of an ICN-based approach in the Internet of Things. We report on the first NDN experiments in a life-size IoT deployment, spread over tens of rooms on several floors of a building. Based on the insights gained with these experiments, the report analyses the shortcomings of CCN applied to IoT. Several interoperable CCN enhancements are then proposed and evaluated. We significantly decreased control traffic (i.e., interest messages) and leverage data path and caching to match IoT requirements in terms of energy and bandwidth constraints. Our optimizations increase content availability in case of IoT nodes with intermittent activity. This report also provides the first experimental comparison of CCN with the common IoT standards 6LoWPAN/RPL/UDP.

Key-words: Network, Information-centric, ICN, CCN, NDN, internet, things, objects, IoT, routing, OS, energy, efficient, operating system, protocol, IPv6, wireless, radio, constrained, embedded

RESEARCH CENTRE
SACLAY – ÎLE-DE-FRANCE

Parc Orsay Université
4 rue Jacques Monod
91893 Orsay Cedex

Réseaux Centrés sur les Contenus dans l'Internet des Objets: Experiences Grandeur Nature avec NDN

Résumé : Ce rapport étudie la faisabilité, les avantages et les vérrous scientifiques affér- rants à une approche réseau centré sur les contenus dans l'Internet des Objets. Cette étude est notamment basée sur des expériences avec NDN, sur un déploiement grandeur nature.

Mots-clés : réseaux, internet, objets, routage, radio, sans-fil, capteurs, ICN, CCN, NDN, système d'exploitation, protocole, OS, efficace, énergie, contraint, embarqué

1 Introduction

The Internet is currently evolving in several ways. On one hand, by going beyond end-to-end streams with Peer-to-Peer, CDNs and now ICN [1], which use information access models where endpoints try to access named content, without direct mapping to a transport layer session. On the other hand, by going beyond traditional user terminal vs. router dichotomy: machine-to-machine (M2M) communications do not involve human source or destination, and interconnected machines include billions of cheap tiny communicating objects which play both the roles of host and router in spontaneous wireless networks, i.e. the Internet of Things [2]. In this dual context, this report explores the feasibility, advantages and challenges of an ICN-based approach in the Internet of Things.

1.1 The Next Billion of Connected Machines

The next billions of interconnected machines are expected to consist in a variety of heterogeneous devices, ranging from wireless sensors to actuators, wearables, Radio-Frequency IDentification (RFID) tags, smart home appliances and many other types of machines that were typically not networked so far. The interconnection of these devices is what has been coined the Internet of Things (IoT), which is expected to profoundly transform our environment.

Most IoT devices will be very limited in terms of memory, CPU as well as power capacities, running on batteries. The term *constrained devices* [3] was recently introduced to define a category of connected devices that are resource-challenged compared to PCs, smartphones or laptops. Constraints include (i) orders of magnitude less power consumption measured in mWatt instead of Watt, (ii) orders of magnitude less computation power measured in MegaFLOPS instead of TeraFLOPS, and (iii) orders of magnitude less memory measured in Kilobytes instead of Gigabytes. For cost reasons, and due to the specific nature of the envisioned (massive) deployments of IoT devices, such constraints are expected to remain the norm in this domain, in the foreseeable future.

To interconnect IoT devices, different approaches have been designed to fit the characteristics of constrained devices in terms of memory, CPU and power capacities. These approaches leverage both traditional, infrastructure-based network paradigms, and spontaneous wireless network paradigms [4] which allow device autoconfiguration and dynamic self-organization to relay data towards destination – even without the help of infrastructure and pre-provisioned access points. Current approaches fall into two categories: silo approaches such as Zigbee [5], and approaches based on open standards protocol stacks such as IPv6 with 6LoWPAN [6] and RPL [7]. In the long run, one can expect that for the same reasons that led IP to prevail, an approach based on open standards and on a layered protocol stack could prevail in the IoT. Thus, in the following, we will consider 6LoWPAN/IPv6/RPL as the reference networking solution for constrained devices in the IoT, with which ICN should measure up.

1.2 ICN for the Internet of Things

Information-centric networking was recently mentioned as a potential alternative networking solution for the IoT [8]. ICN leverages in-network storage and multiparty communication through replication and interaction models such as publish-subscribe to provide efficient and reliable distribution of content.

While several ICN approaches have already been developed, including NDN [9], PSIRP [10], Netinf [11], DONA [12], a number of key aspects remain challenges for ICN [1]. One example of such challenge is the design of routing schemes enabling automatic, efficient and scalable forwarding information configuration on each ICN device. Recent work in this context proposed routing approaches based on proactive, link-state mechanisms [13] and OSPF [14]. However, such approaches may not be directly applicable in the IoT, where constrained devices impose different requirements in terms of memory and power capacities. For instance, requirements for home, industrial and building automation [15] led to the design of a specific routing protocol (i.e. RPL [7]) which can be more energy and memory efficient than standard link-state approaches, by not requiring periodic flooding and by allowing partial topology knowledge.

Nevertheless an ICN paradigm to interconnect IoT devices would provide a number of advantageous characteristics. For example, in-network caching enabled by ICN may save energy and increase local content availability while content producers are in power-save mode. Furthermore, an ICN paradigm could natively accommodate publish-subscribe traffic, which represents a large part of the expected IoT traffic. Last, but not least, by blurring the distinction between several mechanisms across layers, an ICN approach might (i) offer opportunities to efficiently factorize functionalities e.g., caching and buffering for error control (ii) drastically reduce the complexity of autoconfiguration mechanisms compared to an approach based on a layered protocol stack, and (iii) achieve a smaller memory footprint compared to 6LoWPAN/IPv6/RPL.

1.3 Related Work

Recent work has thus started to study ICN paradigms in IoT scenarios or similar contexts (e.g. mobile ad hoc networks). In [16], authors reports on early efforts to provide constrained devices with a CCN communication layer in practice. This implementation is however not interoperable with the full-blown, reference CCN implementation. This initial implementation was used in [17] to showcase a health monitoring application prototype in the context of a small home network. Several architecture design proposals emerged recently for ICN in the Internet of Things, such as [18] which proposes an overlay ICN architecture designed over the M2M ETSI standard, or [19] which identifies high-level requirements of ICN for IoT and proposes a network architecture for IoT based on ICN. Other efforts have proposed enhancements to tackle various issues with ICN in wireless scenarios. For instance, [20] focuses on MANETs scenarios and mobile nodes using ICN and proposes a mechanism reducing the overhead of NDN packet forwarding. On the other hand, [21] focuses on WSN and data collection from a data sink, and proposes in this context an NDN extension for directed diffusion with new packet types and neighbor distinction. This implementation is however not interoperable with the reference CCN implementation. In [22] authors propose a push mechanism for CCN targeting sensor networks. In [23] a gossip mechanism for CCN is introduced, targeting wireless ad hoc networks. Another category of efforts have focused on tackling security and naming issues with ICN in the IoT, such as [24] which studies such issues with CCN in the context of lighting systems and building automation.

However, the above prior work only studied ICN approaches via theoretical analysis and simulations. In [17] and [16], preliminary tests are reported on small, toy networks. But to the best of our knowledge, there are no reports to date on larger scale deployments on IoT hardware, in environments matching requirements described by the industry e.g. in [15]. Furthermore, prior work in this domain has either (i) focused on MANET, where machines are not constrained devices, or (ii) focused on wireless sensor networks and sink-centric data traffic, which is not representative of the whole IoT, where other types of devices participate, and other types of data traffic is significant, such as sensor-to-sensor traffic (as opposed to sensor-to-sink or sink-to-sensor), which represents a substantial part of the traffic in building automation scenarios, e.g. for lighting systems.

1.4 Contributions of this report

In this report, we report on the first CCN experiments in a life-size IoT deployment, spread over tens of offices on several floors of a building, matching characteristics and requirements from building automation as specified in [15]. Based on the insights gained with these experiments, the report analyses the shortcomings of NDN applied to IoT. Several interoperable CCN enhancements are then proposed and evaluated, which decrease interest traffic and focus data path and caching to match IoT requirements in terms of energy and bandwidth constraints, and increase content availability in case of IoT nodes with intermittent activity. This report also provide the first experimental comparison of CCN with the alternative dominant approach in IoT based on 6LoWPAN/RPL/UDP. In addition to our real-world experiments, we discuss ICN in the context of IoT based on an extensive literature survey.

The remainder of this report is organized as follows. First, in Section 2 we will compare IoT requirements with basic ICN characteristics in view to identify the core mismatches and challenges one faces with ICN in the Internet of Things. Then, in Section 3 we will describe in more detail our ICN implementation for the IoT and our deployment setup in a building automation context. Based on insights gained from our experiments with the CCN implementation in this deployment, we will propose and evaluate in Section 4 several interoperable enhancements for CCN operation in the Internet of Things. We present lessons learned in Section 5. Finally, we conclude and discuss future steps in Section 6.

2 A Priori Challenges of ICN in IoT: Memory Requirements

Limited memory resources are fundamental in IoT scenarios. Before an ICN solution can be deployed, it needs to be aligned with these constraints. In this section, we discuss memory requirements introduced by ICN and how we overcome this basic challenge. We separately discuss aspects concerning caching, protocol stack architecture, and routing schemes. For challenges we derived based on our experiments, we refer to § 5.

2.1 Implications on Caching Capabilities

One of the fundamental aspects of ICN is in-network caching, which requires memory dedicated to content cache on nodes in the network. On constrained devices, available RAM is very limited and usually in the order of 10 kBytes [3]. This memory is shared by all processes running on the device, including the operating system, the full network stack, the application(s). Considering typical sizes of these software components in the IoT, the remaining cache size for content on constrained devices is at most in the order of 1 kByte. This is extremely small compared to cache sizes expected on other types of devices initially targeted by ICN [25, 26]. Furthermore, readings of sensor values are ephemeral information by nature: sensor data are continuously replaced by new data, which might allow to disable caching at all. However, caching is doable in the IoT even with limited resources and beneficial.

First, a significant part of the data is expected to consist in small size content. Several sensor values, for example, fit in a single cache. Distributed caching strategies could coordinate multiple devices to achieve in-network caching of all the chunks for medium-sized content (i.e., of size in the order of n kBytes, where n is the number of nodes in the network). Second, in contrast to simple sensor scenarios with a single sink, the IoT envisions multiple consumers (e.g., crowd computing [27]). Then, caching ephemeral content may significantly increase content availability because (i) nodes typically sleep as often as possible to save energy, and (ii) multi-hop wireless paths towards content producers can be very lossy. We will study the effect of caching in Section 4.

2.2 Implications on Overlay Applicability

Deploying only the IP stack on constrained devices is already a challenge in terms of RAM and ROM. ICN approaches that work on top of IP may be impossible due to the additive memory requirements of both the ICN stack and the IP stack. This observation points towards ICN implementations that can function directly above the link layer. For the experiments reported in this report, we have thus used an ICN approach running directly above the MAC layer (see Sections 3 and 4).

2.3 Implications on Routing Approaches

Reduced memory of constrained devices also limits applicability of ICN routing approaches. Current routing proposals usually route either directly on names or indirectly via name resolution. Based on the above observation, name resolution which relies on an ICN overlay on top of IP is not possible. However, even pure name-based routing, such as [13] and [14] challenge the IoT environment as they require an IP underlay or use proactive link state algorithms. Link state routing requires both (i) a significant amount of control traffic, whether or not there is data traffic to carry in the network, and (ii) a significant amount of memory, typically in O(n), where n is the number of nodes in the network. These characteristics do not match the memory and energy resources of constrained devices.

Routing protocols running on IoT devices should aim for O(1) routing state and minimal control traffic – ideally none, especially when there is no data traffic to carry [28]. In this report (see Section 4), we introduce an ICN routing scheme with these properties.

3 Steps to Enable ICN in the IoT

In order to gain a full understanding how ICN operates in the Internet of Things, it is inevitable to conduct experiments in real-world deployments and/or testbeds that reflect properties of such deployments, i.e., avoiding topologies and densities that are too artificial, too regular, or too isolated from the real word which includes external interferences resulting from other radio networks, electrical devices, or simple human activity. The first step towards such experiments is implementing ICN code that runs on IoT hardware.

3.1 Porting CCN-Lite to RIOT

We have ported CCN-Lite [29], a bare-bone Linux open source implementation of NDN, to RIOT [30], an operating system for the constrained devices. Among ICN approaches, we have chosen NDN because it can easily operate directly above the link layer – a requirement we identified in Section 2. We chose to base ourselves on CCN-Lite because this implementation is compliant with the reference NDN implementation (CCNx) while being very compact: less than 1000 lines of C code and low memory footprint. And we chose RIOT as operating system to run on constrained devices because it is open source and fits IoT devices memory requirements, while allowing plain C code with all the standard headers, and being based on a (multi-)threading model comparable to POSIX. These characteristics guaranteed that porting Linux code to RIOT is straightforward. We also leveraged RIOT support for popular debugging tools such as Valgrind,

Module	ROM	RAM
RPL + 6LoWPAN	53412 bytes	27739 bytes
CCN-Lite	16628 bytes	5112 bytes

Table 1: Comparing memory resources for RIOT on MSBA2

Module	ROM	RAM
RPL + 6LoWPAN	52131 bytes	21057 bytes
CCNX	13005 bytes	5769 bytes

Table 2: Comparing memory resources for Contiki on Redbee-Econotag

Wireshark, gdb, and nativenet. Our implementation is open source and available online in GitHub [31].

Tables 1 and 2 compare the ROM and RAM sizes of the binaries compiled for NDN network stacks and for 6LoWPAN/RPL network stacks, built upon state-of-the-art IoT operating systems (RIOT and Contiki), for state-of-the-art IoT hardware (Redbee Econotag board and MSB-A2 board). We observe that an ICN approach can significantly outperform common IoT protocols in terms of ROM size (down to 60% less) and RAM size (down to 80% less).

3.2 NDN Deployment on Campus

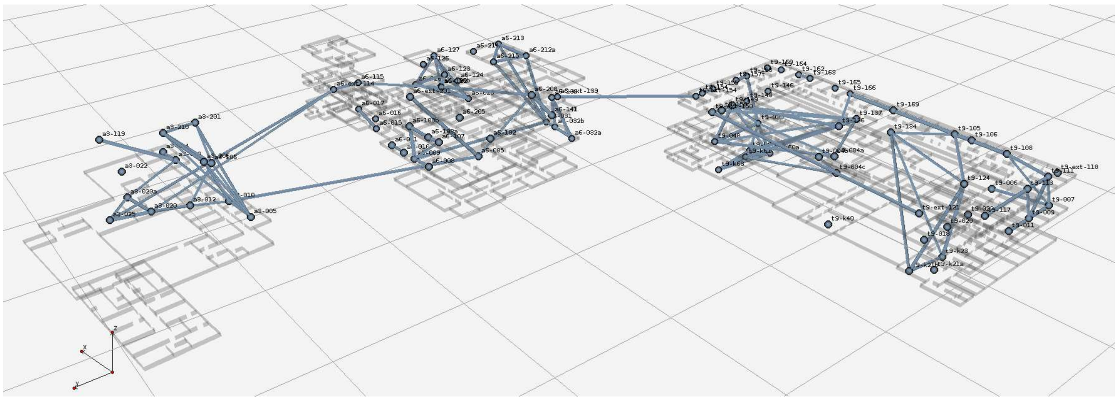


Figure 1: 3D visualization of the topology of the deployment, consisting in 60 nodes that interconnect via wireless communications (sub-GHz) and that are physically distributed in multiple rooms, multiple floors, and multiple buildings.

Typical IoT application scenarios, include building and home automation [15, 32], smart metering [33], or environment monitoring [34, 35]. For the NDN experiments, we deployed our ICN IoT implementation on the campus of *Freie Universität Berlin*, consisting in 60 nodes distributed in various rooms, on several floors, and in several building, as shown in Figure 1. Each node is equipped with a CC1100 radio chip operating at 868MHz, and sensors that can measure various parameters including room temperature, humidity etc. For more details we refer to [36]. Most of the nodes are deployed inside rooms, while a few nodes are deployed outdoor to better interconnect nodes in different buildings. Nodes interconnect via their wireless interface, which offers a maximum link layer frame size of 64 Bytes.

In order to monitor closely energy consumption, verify individual node behavior, and manage experiments on this deployment (e.g., flash nodes, gather results) each node is furthermore accompanied by its own docking station. Docking stations are interconnected via an Ethernet backbone [37]. However, these docking stations are used only to monitor and manage the nodes. Nodes operate autonomously, i.e., each node can only use its own CPU, its own memory, and its own wireless interface to communicate with other nodes. A 3-dimensional snapshot of the resulting wireless topology is shown in Figure 1.

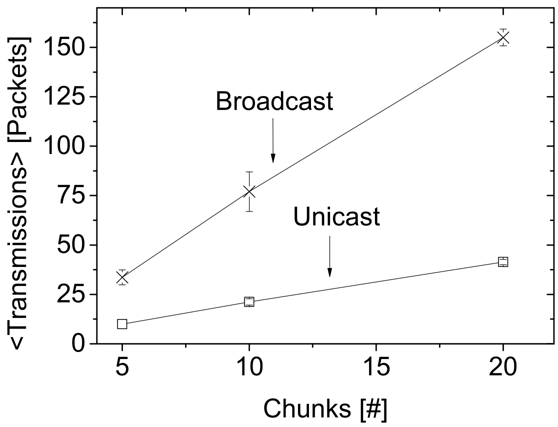
4 NDN Experiments and Optimizations for IoT Deployment

In order to obtain a fully functional NDN network stack for the IoT, a FIB autoconfiguration mechanism is needed: in IoT scenarios, even less than in others, one cannot expect humans in the loop, so manual configuration is out of the question. As mentioned in Section 2, existing ICN routing approaches are not appropriate for constrained devices in the IoT: alternative routing mechanisms must be used in this context, which require drastically less state. In the following, we will describe and evaluate several routing alternatives, as well as other aspects of NDN in the wild, such as the effect of caching in IoT.

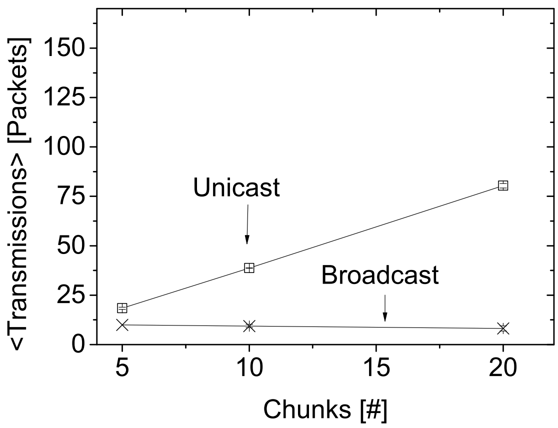
4.1 Vanilla Interest Flooding (VIF)

The simplest routing approach that requires minimal states is *interest flooding*, whereby each node in the network repeats an interest, upon first reception. In the following, we will call this simple mechanism Vanilla Interest Flooding (VIF). Using VIF, a consumer with an empty FIB can nevertheless disseminate its interest in content, and the flooded interest will reach the producer which can then send the content on the reverse path. VIF fits the constraints of IoT devices in that (i) it does not rely on any control traffic, (ii) it requires minimal state, i.e., only temporary pending interests on the reverse path of content that is sought after.

Figure 2(a) shows the results of an experiment using a network of ten nodes and NDN with VIF, on the deployment described in Section 3.2. In this experiment, one consumer accesses 3 types of content, consisting in 5, 10, and 20 chunks of data, all of which produced by another constrained node in the network, at a minimum hop distance of 2. While the experiment is successful in that NDN was demonstrated to operate on IoT hardware (therefore meeting memory requirements) and the consumer could fetch the content, Figure 2(a) shows that quite a lot of packets were transmitted in order to fetch content, compared to the size of the content fetched. This is due to the fact that each chunk fetched requires network-wide flooding. Thus, in a network of n nodes, and for k chunks of content, the number of transmissions is $k \cdot ((n-1)+\sqrt{n})$, assuming the average path length approximation $\sqrt{n}$. Thus, while VIF is simple and works, it does not scale in terms of number radio transmissions when the network or the content grows in size. Radio transmission and reception are however very costly in terms of energy for battery-powered IoT devices. In the following, we have thus designed and tested enhancements reducing the number of radio transmissions and receptions in IoT environment.



(a) Vanilla Interest Flooding



(b) Reactive Optimistic Name-based Routing

Figure 2: NDN performance for different routing schemes. Average number of packets transmitted in a network of 10 nodes to fetch content of various size.

4.2 Reactive Optimistic Name-based Routing (RONR)

In order to reduce the number of radio transmissions compared to basic interest flooding, a possible enhancement is RONR, which automatically configure a temporary FIB entry on the reverse path taken by the first content chunk. That way, in case the FIB is empty (e.g., after booting) or if no FIB entry matches the name/prefix of the content in which the consumer is interested, only a single initial interest flooding is needed, while subsequent interests for chunks of that content can be unicast using the FIB entries thus auto-configured along the path. RONR is optimistic because it first assumes that the whole content is stare in a single node (a cached replica or the original producer), which may not be the case in general. However, this assumption makes sense in the IoT because typical content size is small. Furthermore, FIB entries timeout ensure that if the configured FIB entries do not lead to a node with the full content, the consumer will eventually revert to interest flooding and discover another node with the rest of the content, install new temporary FIB entries etc.

In Figure 2(b), we show the results of an experiment using NDN with RONR, for the exact same topology and scenario as for Figure 2(a). We observe that as expected the number of radio transmissions decrease drastically to NDN with VIF, about 50% less. A quick back-of-the-envelope analysis shows that in a network of n nodes, and for k chunks of content, the number of transmissions is $(n-1)+2(k-\frac{1}{2})\sqrt{n}$, assuming again the average path length approximation $\sqrt{n}$. Therefore, RONR scales much better than VIF when network size or content size grows. RONR thus better fits IoT devices energy requirements compared to VIF, while still fitting other requirements of constrained devices by (i) not relying on any control traffic, and (ii) requiring minimal state, i.e., only temporary FIB entries on the reverse path of content that is sought after (not counting PIT state, of course).

4.3 Multiple Consumers & Impact of Caching

In this section we evaluate experimentally the impact of caching on a larger topology, with multiple consumers for the same content. In Figure 3(a) we show the results of an experiment using NDN with a cache size of 0 (i.e. caching is disabled), on a network of 20 nodes. In this experiment, the same content (20 chunks) is accessed alternatively by 1, 2, or 3 consumers that are topologically close to one another (minimum hop distance is 2). We observe that, as expected, the number of radio transmissions scales almost linearly with the number of consumers. In a network of n nodes, and for k chunks of content and m consumers within radio reach, the number of transmissions is $m \cdot ((n-1)+2(k-\frac{1}{2})\sqrt{n})$, still assuming the average path length approximation $\sqrt{n}$.

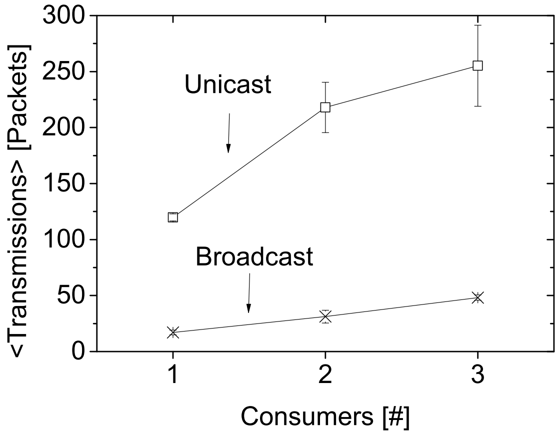
On the other hand, in Figure 3(b) we show the results we obtained for the exact same topology and scenario as for Figure 3(a), except that the cache size is set to a capacity of 20 chunks (which corresponds to RAM usage of 2kBytes). We observe that the number of radio transmissions needed to retrieve the content is drastically reduced, by up to 50% in this scenario. In the best case, if the initial flood for subsequent consumers can be reduced to a local broadcast because only neighbors with cached content receive the interest, the number of transmissions becomes $2(k-\frac{1}{2})(\sqrt{n}+n-1)+n+m-2$.

4.4 Comparison with 6LoWPAN/IPv6/RPL/UDP

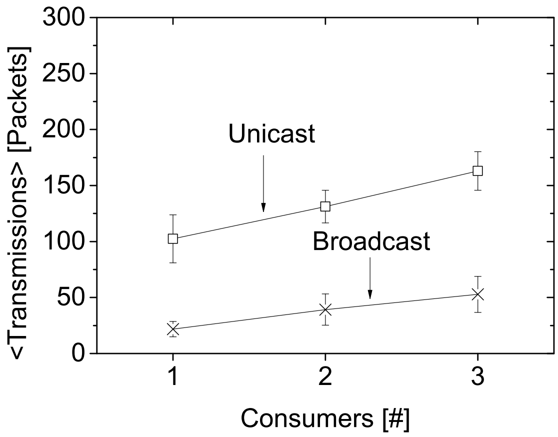
In this section, we compare NDN using RONR with 6LoWPAN/RPL/UDP, a common protocol suite for the current IoT. For RONR, we configure a cache size of 2 kByte. In Figure 4, we show the results we obtained for the exact same topology and scenario as for Figure 3(b), except the network stack used was 6LoWPAN/RPL/UDP with default settings instead of NDN. For fairness, we first let the network converge to a point where the root and the routing entries are installed in nodes, before we start the experiment. Then, following the scenarios from Section 4.3, the same content (20 chunks) is accessed alternatively by 1, 2, or 3 consumers. We observe that, the 6LoWPAN/RPL/UDP network stack yields much more transmissions compared to what we measured for NDN in Figure 3(b), approximately three times more. In particular, the amount of control traffic is a big penalty. So we can conclude that NDN may be potential alternative to 6LoWPAN/RPL/UDP, which should be studied more in the context of IoT in future work.

5 A Posteriori Challenges: What are the Lessons Learned

In this section, we gathered further considerations and observations concerning ICN in the Internet of Things, based on our practical experience with NDN implementation and deployment. In the following, we distinguish energy consumption aspects, wireless connectivity aspects and communication model aspects.



(a) Without caching



(b) With caching

Figure 3: NDN performance for differnet cache schemes. Average number of packets transmitted in a network of 20 nodes, where a variable number of consumer fetch the same content.

5.1 Energy Consumption

Energy consumption is mainly impacted by network transmissions, which are affected by content naming, content caching, network flooding, and local wireless broadcast.

Impact of Names. Routing information about names and prefixes should dynamically be auto-configured in IoT devices. The resulting overhead not only depends on the routing protocol but also on the size of names to be processed in ICN packets. In our experiments, we deployed VIF, a very basic approach based on *flooding*, whereby each node in the network repeats (on all interfaces) each flooded packet upon first reception (on any interface).

Flooding is used (i) to disseminate an interest message when no forwarding information is available, or (ii) to disseminate names and topology information, e.g., with link state routing approaches [13] [14]. However, flooding is costly in terms of energy since each flood requires O(n) packet transmissions and O(nm) packet receptions, where n is the number of nodes in the network and m is the average node degree. Each packet received will not only be costly in

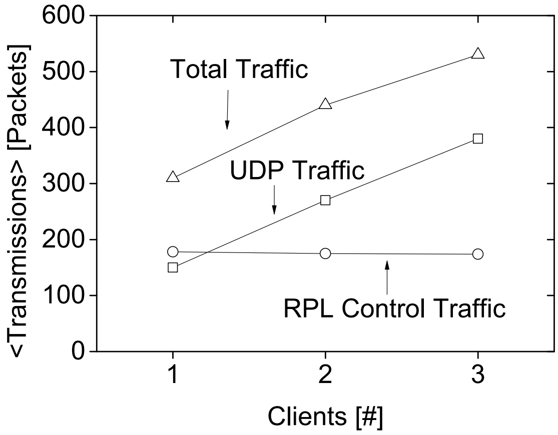


Figure 4: 6LoWPAN/RPL/UDP. Average number of packets transmitted in a network of 20 nodes, where a variable number of consumer fetch the same content.

# of instructions	**Function**
14,002,814	memcmp_ssse3
7,525,050	ccnl_nonce_find_or_append
4,062,659	ccnl_i_prefixof_c
1,462,304	dehead
956,238	ccnl_core_RX_i_or_c
895,590	ccnl_extract_prefix_nonce_ppkd
845,042	memcpy_ssse3

Table 3: CPU cycles per CCN function.

terms of pure packet reception but will also trigger its processing, which includes CPU-expensive string comparisons with variable lengths, trying to match received names with names stored locally. Furthermore, recent work [38] identifies ICN packet processing as a CPU bottleneck, serious enough to provide DOS attack opportunities. This processing is even more costly on constrained devices since their CPU typically does not benefit from advanced functionalities such as prefetching or super scalar instruction set, and thus needs one cycle per byte compared. Table 3 shows a benchmark for the number of required CPU cycles per CCNlite operation for our implementation in RIOT. The top 3 functions, which represent 85% of the CPU cycles, have to do with string comparison and name matching.

These observations thus call for (i) the least possible recourse to flooding and (ii) the shortest possible names. Note that human-readable names may not be required or useful in a context of machine-to-machine communication. Also note that shorter names should however not sacrifice prefix aggregability so that scalability remains in terms of number of nodes in the network vs. routing state.

Impact of Caching. The impact of in-network caching on energy aspects with ICN approaches has been studied by recent work such as [39], which indicates that energy consumption incurred by caching reduces energy efficiency. But on the other hand, studies such as [40] show that CCN can be more energy efficient than other content delivery approaches such as CDN and P2P by leveraging the most energy efficient devices in the network. It remains to be seen at large scale on the Internet which ICN approaches introduce low overhead in terms of energy consumption. In the IoT, to the best of our knowledge, there are no studies yet that focused on energy aspects of ICN due to the use of caching.

In Section 4, we demonstrated experimentally that savings in terms of energy consumption are possible thanks to (even small) in-network caching since (i) on-path or near-path caching can decrease the number of intermediate energy-challenged devices on the path to reach content in some scenarios, and (ii) content producers such as sensors could sleep more while their content could still be available in other caches in the network.

Impact of Local Wireless Broadcast. In case of multiple PIT hits, the NDN stack could use a single multicast transmission if all matching neighbors are reachable through the same wireless interface – which is the case in most IoT scenarios where nodes only have a single interface (omnidirectional radio). We have thus enhanced our NDN implementation with such a link-local multicast awareness mechanism called Content Forwarding Aggregation (CFA). In scenarios where multiple geographically close consumers are interested in the same content at approximately the same time, CFA leads to substantial gains in terms of number of radio transmissions necessary to deliver the content. Another opportunity to leverage the multicast nature of IoT devices' wireless interface concerns caching. Very often, a node will overhear unsolicited chunks of content that are being transmitted in its radio vicinity. In such case, instead of discarding this content, the node could cache this unsolicited in its content store, if there is space left, with a lower priority than solicited content. We have thus enhanced our NDN implementation with such a mechanism, called Opportunistic Near-Path Caching (ONPC). ONPC enables to further reduce the number of radio transmissions in case of several consumers of the same content. Due to lack of space, we do not show experimental results with CFA or ONPC in this paper.

5.2 Wireless Connectivity

Although ICN is applicable in wireless networks, several issues arise when applied to wireless regime in the IoT. In the following, we distinguish aspects concerning frame size, fragmentation, and bidirectional links.

Frame Size and Packet Fragmentation. Several link layer technologies are currently used in the IoT, and it is likely that multiple technologies will be used in the future, too. Currently, the dominant IoT link layer in the field of building automation and industrial automation is IEEE 802.15.4. The maximum frame size is very small (127 bytes). Other popular wireless link layers such as Dash7 provide an even smaller maximum frame size (64 bytes), and Bluetooth Low Energy [41] typically allows a payload of 23 bytes. These frame sizes are more are ten to a hundred times smaller compared to traditional Ethernet or WiFi frames. Consequently, fragmentation and reassembly mechanisms are necessary. While Bluetooth provides its own, IEEE 802.15.4 or DASH7 do not. To bridge this gap, 6LoWPAN introduced (i) a standard header compression scheme, and a (ii) standard fragmentation and reassembly mechanism for IPv6 operation in the IoT, both on top of IEEE 802.15.4 link layer. It is worth noting that ICN cannot benefit from these mechanisms because overlay architectures conflict with memory constrains in the IoT (cf., Section 2.2).

In our real-world deployment, we demonstrated that NDN can be implemented directly on top of an IoT link layer, without compression/fragmentation mechanisms (see Sections 3 and 4). Omitting these optimizations is suitable for basic scenarios in which small enough names and small enough chunks can be used in the first place. Our results give confidence that we can already start with ICN in the IoT. However, in the future, ICN approaches for the IoT need an equivalent of what 6LoWPAN is providing for IPv6. For illustration, NDN relies on a

30-40 bytes header, which is negligible in the common Internet (≈2% of the capacity of standard 1500 bytes MTU) but occupies ≈28% of the capacity of standard 802.15.4 frames. It neither can be expected that all chunk sizes on all ICN networks will be defined by IEEE 802.15.4 frame size, which would be inefficient, nor can we expect that names indicated in interest packets will always be short enough to fit in a single 802.15.4 frame of 127 bytes, for example. Note that fragmentation approaches need to take into account that altered chunks can break security and naming schemes.

Bidirectional links. Many ICN approaches assume bidirectional links. This is not true in general in spontaneous wireless networks [4], and thus this assumption does not hold in the IoT. In such context, a high proportion of links are asymmetric, e.g., 10% loss rate from A to B and 80% loss rate from B to A. In reality, a substantial fraction of the links are unidirectional, i.e., loss rate strictly below 100% in one direction, and 100% loss rate in the reverse direction. Last but not least, wireless link quality between two nodes A and B can vary significantly over time, even at small time scales [42] – that we also experienced in our experiments.

The above wireless connectivity characteristics lead to the following observations. ICN routing protocols running on constrained devices need to satisfy conflicting requirements (i) negligible control traffic to reduce energy consumption and small state to fit memory constraints, and at the same time (ii) dynamic tracking of wireless link to avoid non-functional paths. The goal is to not forward an interest in the first place if reverse link is not "good enough". The overhead for failing is a reverse path taken by content which often fails and will lead to PIT time-outs, interest flooding, etc. Finally, this might lead to the same failing reverse path – and thus be very inefficient both in terms of energy and delay.

5.3 Different Communication Models

The ICN communication model is based on a *pull* paradigm: in a first phase, a node expresses interest in some content, and in a second phase, the node should receive this content. However, this communication model alone is not sufficient to accommodate typical traffic patterns in the IoT. Aside of pull, these patterns include for instance *push* paradigms (e.g., for actuators), and *observe* paradigms [43] whereby a node can register for updates from a given content producer (e.g., a sensor measuring in real-time the evolution of a given parameter). Note that explicit acknowledgements are also typically used in this context, for example patterns such as push+ACK, or request+reply+ACK are the norm in this domain. Recent work has started to integrate these patterns in ICN, such as [22] which proposes a push mechanism for CCN on sensor networks.

Furthermore, the simplified communication model at the base of ICN was initially designed with the assumption that the number of consumers is much larger than the number of producers, targeting use cases that are comparable to the scenarios CDNs aim for. Such assumption does not hold in general in the IoT, where consumers (e.g., a data sink) are often outnumbered by producers (e.g., sensors). In consequence, content caching strategies designed for scenarios similar to CDN will not be efficient in the IoT, and thus, specific alternative strategies should be designed for content replication and content cache replacement in the IoT with ICN.

6 Conclusion and Perspective

ICN has recently been mentioned as a potential alternative network paradigm for the Internet of Things. In this report we have carried out experiments with an ICN approach on a real IoT deployment consisting in tens of constrained nodes in multiple rooms of multiple buildings. Based on this experience, we have shown that ICN is indeed applicable in the IoT, and that it can offer advantages over an approach based on 6LoWPAN/IPv6/RPL in terms of energy consumption, as well as in terms of RAM and ROM footprint. We have proposed several interoperable NDN enhancements to decrease energy consumption and routing state. We have have furthermore identified several areas where future work is needed. Topics include (i) an efficient fragmentation/reassembly adaptation layer below NDN to fit typically small frame sizes, (ii) IoT-specific content replication and cache replacement strategies, (iii) enhancements of the basic ICN communication model to accommodate IoT traffic patterns, (iv) further studies on the impact of caching on content availability in the context of sleeping nodes, and (v) short naming schemes optimized for constrained devices.

References

[1] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, “A survey of information-centric networking,” *IEEE Communications Magazine*, vol. 50, no. 7, pp. 26–36, 2012.

[2] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey,” *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[3] C. Bormann, M. Ersue, and A. Keranen, “Terminology for Constrained-Node Networks,” RFC 7228 (Informational), Internet Engineering Task Force, May 2014. [Online]. Available: http://www.ietf.org/rfc/rfc7228.txt

[4] J. Cordero, J. Yi, T. Clausen, and E. Baccelli, “Enabling Multihop Communication in Spontaneous Wireless Networks,” in *ACM SIGCOMM eBook on "Recent Advances in Networking", Volume 1, Chapter 9, pp. 413-457.* ACM, 2013.

[5] ZigBee Alliance, “ZigBee Specifications,” 2012.

[6] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, “Transmission of IPv6 Packets over IEEE 802.15.4 Networks,” RFC 4944 (Proposed Standard), Internet Engineering Task Force, Sep. 2007, updated by RFCs 6282, 6775. [Online]. Available: http://www.ietf.org/rfc/rfc4944.txt

[7] T. Winter and P. Thubert, “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks,” RFC 6550 (Proposed Standard), Internet Engineering Task Force, Mar. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6550.txt

[8] A. Ghodsi, B. Ohlman, J. Ott, I. Solis, and M. Wählisch, “Information-centric networking: Ready for the real world?” *Dagstuhl Reports (Seminar 12361)*, vol. 2, no. 9, pp. 1–14, 2012.

[9] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, “Networking named content,” pp. 1–12, 2009.

[10] N. Fotiou, D. Trossen, and G. Polyzos, “Illustrating a publish-subscribe internet architecture,” *Telecommunication Systems*, vol. 51, no. 4, pp. 233–245, 2012.

[11] C. Dannewitz, D. Kutscher, B. Ohlman, S. Farrell, B. Ahlgren, and H. Karl, “Network of information (netinf), an information-centric networking architecture,” *Computer Communications*, vol. 36, no. 7, pp. 721 – 735, 2013.

[12] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, “A data-oriented (and beyond) network architecture,” *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 181–192, Aug. 2007.

[13] A. K. M. M. Hoque, S. O. Amin, A. Alyyan, B. Zhang, L. Zhang, and L. Wang, “NLSR: Named-data Link State Routing Protocol,” in *Proc. of the 3rd ACM SIGCOMM WS on ICN*. New York, NY, USA: ACM, 2013, pp. 15–20.

[14] 2012, “Ospfn: An ospf based routing protocol for named data networking,” Lan Wang, Mahmudul Hoque, Cheng Yi, Adam Alyyan and Beichuan Zhang. [Online]. Available: http://www.named-data.net/techreport/TR003-OSPFN.pdf

[15] J. Martocci, P. D. Mil, N. Riou, and W. Vermeylen, "Building Automation Routing Requirements in Low-Power and Lossy Networks," RFC 5867 (Informational), Internet Engineering Task Force, Jun. 2010. [Online]. Available: http://www.ietf.org/rfc/rfc5867.txt

[16] B. Saadallah, A. Lahmadi, O. Festor, "CCNx for Contiki: implementation details," in *INRIA, Tech. Report RT-0432*. INRIA, 2012.

[17] T. Biswas et al., "Contextualized information-centric home network," in *ACM SIGCOMM*, 2013.

[18] L. Grieco et al., "Architecting information centric etsi-m2m systems," in *Proc. of PERCOM*, 2014.

[19] Y. Zhang, D. Raychadhuri, R. Ravindran, G. Wang, "ICN based Architecture for IoT," in *IETF Internet Draft draft-zhang-iot-icn-architecture-00*. IRTF, 2013.

[20] Y.-T. Yu, R. B. Dilmaghani, S. Calo, M. Sanadidi, and M. Gerla, "Interest propagation in named data manets," in *Proc. of IEEE ICNC*, 2013, pp. 1118–1122.

[21] M. Amadeo, C. Campolo, A. Molinaro, and N. Mitton, "Named data networking: A natural design for data collection in wireless sensor networks," in *Wireless Days (WD), 2013 IFIP*. IEEE, 2013, pp. 1–6.

[22] J. Francois, T. Cholez, T. Engel, "CCN Traffic Optimization for IoT," in *International Conference on Network of the Future (NoF)*, 2013.

[23] F. Angius, M. Gerla, and G. Pau, "Bloogo: Bloom filter based gossip algorithm for wireless ndn," in *Proc. of the 1st ACM NoM Workshop*. New York, NY, USA: ACM, 2012, pp. 25–30.

[24] J. Burke, P. Gasti, N. Nathan, and G. Tsudik, "Securing instrumented environments over content-centric networking: the case of lighting control," *arXiv preprint arXiv:1208.1336*, 2012.

[25] S. Arianfar, P. Nikander, and J. Ott, "On Content-Centric Router Design and Implications," in *Proc. of ReARCH workshop*. USA: ACM, 2010.

[26] D. Perino and M. Varvello, "A Reality Check for Content Centric Networking," in *Proc. of the ACM SIGCOMM WS on ICN*. USA: ACM, 2011, pp. 44–49.

[27] D. G. Murray, E. Yoneki, J. Crowcroft, and S. Hand, "The Case for Crowd Computing," in *Proc. of the 2nd ACM SIGCOMM MobiHeld WS*. New York, NY, USA: ACM, 2010, pp. 39–44.

[28] P. Levis, A. Tavakoli, and S. Dawson-Haggerty, "Overview of existing routing protocols for low power and lossy networks," *IETF Internet Draft draft-ietf-roll-protocols-survey*, 2009.

[29] "CCN Lite: Lightweight implementation of the Content Centric Networking protocol," 2014. [Online]. Available: http://ccn-lite.net

[30] E. Baccelli, O. Hahm, M. Günes, M. Wählisch, and T. C. Schmidt, "RIOT OS: Towards an OS for the Internet of Things," in *Proc. of the IEEE INFOCOM. Poster*, NJ, USA: IEEE Press, 2013.

[31] "RIOT open source code on GitHub," 2014. [Online]. Available: https://github.com/RIOT-OS/RIOT

[32] A. Brandt, J. Buron, and G. Porcu, "Home Automation Routing Requirements in Low-Power and Lossy Networks," IETF, RFC 5826, 2010.

[33] Z. Fan, G. Kalogridis, C. Efthymiou, M. Sooriyabandara, M. Serizawa, and J. McGeehan, "The new frontier of communications research: Smart grid and smart metering," in *Proc. of ACM e-Energy*. New York, NY, USA: ACM, 2010, pp. 115–118.

[34] T. Naumowicz, R. Freeman, A. Heil, M. Calsyn, E. Hellmich, A. Brändle, T. Guilford, and J. Schiller, "Autonomous monitoring of vulnerable habitats using a wireless sensor network," in *Proc. of REALWSN*. USA: ACM, 2008, pp. 51–55.

[35] G. Wittenburg *et al.*, "Fence Monitoring - Experimental Evaluation of a Use Case for Wireless Sensor Networks," in *Proc. of EWSN*, 2007.

[36] Michael Baar et al., "The ScatterWeb MSB-A2 Platform for Wireless Sensor Networks," Freie Universität Berlin Tech. Rep, Tech. Rep., 2008.

[37] M. Günes et al., "Practical Issues of Implementing a Hybrid Multi-NIC Wireless Mesh-Network," Tech. Rep., 2008.

[38] M. Wählisch, T. C. Schmidt, and M. Vahlenkamp, "Backscatter from the Data Plane – Threats to Stability and Security in Information-Centric Network Infrastructure," *Computer Networks*, vol. 57, no. 16, pp. 3192–3206, Nov. 2013.

[39] N. Choi, K. Guan, D. C. Kilper, and G. Atkinson, "In-network caching effect on optimal energy consumption in content-centric networking," in *Proc. of IEEE ICC*, 2012, pp. 2889–2894.

[40] U. Lee et al., "Greening the internet with content-centric networking," in *Proc. of the ACM e-Energy*, 2010.

[41] M. Isomaki et al., "Transmission of IPv6 Packets over BLUETOOTH Low Energy," *IETF Internet Draft draft-ietf-6lo-btle*, 2014.

[42] E. Baccelli and C. Perkins, "Multi-hop Ad Hoc Wireless Communication," IETF, Internet-Draft – work in progress 03, March 2014.

[43] Z. Shelby, K. Hartke, C. Bormann, and B. Frank, "Constrained application protocol (coap)," *IETF Internet Draft draft-ietf-core-coap*, 2014.

RESEARCH CENTRE
SACLAY – ÎLE-DE-FRANCE

Parc Orsay Université
4 rue Jacques Monod
91893 Orsay Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399