

HoPP: Robust and Resilient Publish-Subscribe for an Information-Centric Internet of Things

Cenk Gündoğan
HAW Hamburg

{cenk.guendogan, peter.kietzmann}

Peter Kietzmann
HAW Hamburg

@haw-hamburg.de

Thomas C. Schmidt
HAW Hamburg

t.schmidt@haw-hamburg.de

Matthias Wählich
FU Berlin

m.waehlich@fu-berlin.de

Abstract—This paper revisits NDN deployment in the IoT with a special focus on the interaction of sensors and actuators. Such scenarios require high responsiveness and limited control state at the constrained nodes. We argue that the NDN request-response pattern which prevents data push is vital for IoT networks. We contribute HoP-and-Pull (HoPP), a robust publish-subscribe scheme for typical IoT scenarios that targets IoT networks consisting of hundreds of resource constrained devices at intermittent connectivity. Our approach limits the FIB tables to a minimum and naturally supports mobility, temporary network partitioning, data aggregation and near real-time reactivity. We experimentally evaluate the protocol in a real-world deployment using the IoT-Lab testbed with varying numbers of constrained devices, each interconnected via IEEE 802.15.4 wireless LoWPANs. Implementations are built on CCN-lite with RIOT and support experiments using various single- and multi-hop scenarios.

Index Terms—ICN, Industrial Internet of Things, Constrained Environment, DoS Resistance

I. INTRODUCTION

The Internet of Things (IoT) is emerging, and billions of new networked devices are forecasted. However, no common networking technology for the IoT has been agreed upon. Despite of a maturing IETF protocol suite, dozens of incompatible industry solutions are rolled out to meet device and network constraints, as well as application specific needs.

Facing this huge world of mainly constrained devices, it seems worth rethinking its networking paradigm. A very loose coupling appears most appropriate between nodes that often run on battery with long sleep cycles and connect via lossy wireless links. Information-Centric Networking (ICN) [1], [2] decouples content provisioning from data producers in space which makes it a promising candidate. Additional decoupling in time and synchronization is desirable and attainable by a publish-subscribe layer.

Information-centric publish-subscribe networks have been proposed. PSIRP/PURSUIT [3] is an early, prominent candidate. However, its central control architecture seems more suitable for an SDN-type deployment in LANs. Publish-subscribe schemes based on NDN like Content-based pub/sub [4] and COPSS [5] violate the loose coupling principle in their use of name-based routing or forwarding. Facing the current state of the art, we explore the problem of information-centric publish-subscribe for IoT networking open.

In this paper, we take up the challenge and seek for an information-centric IoT networking solution that satisfies all

challenges of real-world sensor-actuator networks and allows for an easy deployment. We base our work on NDN [6] not only because of its widespread availability and implementations on IoT operating systems, but in particular because of its clean request-response scheme that prevents unwanted traffic at the constrained end nodes. We design and evaluate HoP-and-Pull (HoPP), a lean, adaptive publish-subscribe layer that strictly adheres to the NDN communication pattern. Our experimental findings on large IoT testbeds indicate that our system complies indeed to the challenging requirements of IoT use case with promising performance. In particular, reliability and resilience of HoPP largely outperforms previously advised push notifications.

The structure of this paper continues as follows. In Section II, we dive into the design details of our publish-subscribe scheme, including the key aspects of network partitioning and publisher mobility. Implementation and evaluations of our system are described in Section III. Finally, we conclude with an outlook in Section IV.

II. HOP AND PULL: A PUBLISH-SUBSCRIBE APPROACH TO LIGHTWEIGHT ROUTING ON NAMES

We now describe HoP-and-Pull (HoPP), our pub-sub system for lightweight IoT deployment. For a confined IoT environment, we make the common assumption that nodes form a stub network that may be connected to the outside by one or several gateways. Some global prefix is given to a gateway, but (wireless) IoT nodes can reach a gateway without global prefix changes in one or several hops unless they are temporarily disconnected [7]. Internally, nodes may be grouped according to one or several sub-network prefixes (e.g., /lighting).

We select one or several distinguished nodes to serve as Content Proxies (CPs). CPs are typically more stable and more powerful such as gateways or other infrastructural entities. These Content Proxies take the role of data caches and persistent access points. They will be reachable throughout the network by default routes, unless temporary partitioning occurs. Note that one CP can serve several local prefixes, but a local prefix may also belong to several CPs. The latter scenario will lead to replicated caching with higher and faster data availability.

Our publish-subscribe protocol for the IoT is then composed of three core primitives:

- 1) Establishing and maintaining the routing system

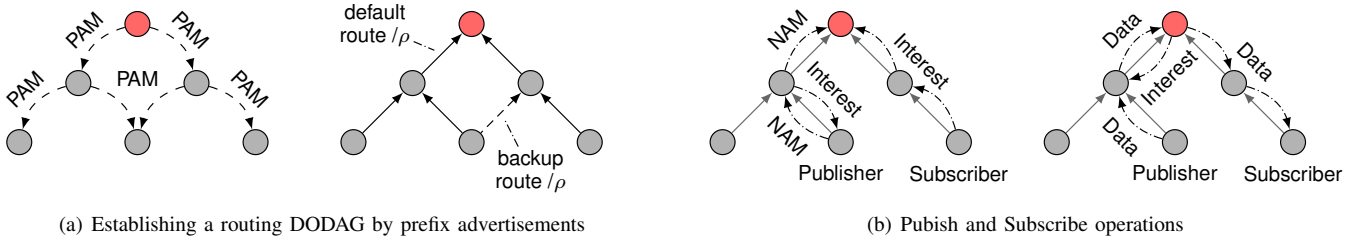


Fig. 1. HoPP protocol operations

- 2) Publishing content to the CPs
- 3) Subscribing content from the CPs

Our following protocol definition strictly complies with the design principles: (a) minimal FIBs that only contain default routes, (b) no push primitive or polling, (c) no broadcast or flooding on the data plane. The HoPP protocol transparently manages consumer and *producer mobility* [8].

A. Prefix-specific default routing

Content Proxies advertise the prefix(es) they own on the control plane to all neighbors in a Prefix Advertisement Message (PAM). Observing nodes will adopt a CP as their parent and re-broadcast the PAM message with an increased distance value. Much like in the core RPL [9], all nodes will be members of a Destination-Oriented Directed Acyclic Graph (DODAG) after routing convergence. Nodes will include the selected best uplink in their FIB as default route to the announced prefix, but may add additional uplinks with lower priority.

Figure 1(a) visualizes the PAM prefix distribution and the corresponding FIB entry for the sample prefix ρ . All nodes establish a default on shortest paths upstream. In addition, nodes learn backup paths of equal hop distance, but lower radio quality.

B. Publishing content

An IoT node (sensor) that has new data to publish will first select a name. It may choose either from a predefined scheme accessible by local controllers, some common standard set, or decide individually. It will advertise this content name to its upstream neighbor via a (unicast) Name Advertisement Message (NAM). It will also associate the content with one or several topic names and adds these to the content metadata.

Under regular network conditions, the upstream neighbor is expected to retrieve the advertised content via the incoming interface of the NAM. It proceeds according to the standard NDN scheme: An Interest requests the name, the data is returned in response. Concurrently, the upstream issues a corresponding NAM to its parent, which in turn pulls the content one hop closer to the CP. This hop-wise content replication proceeds until the data arrives at the Content Proxy.

It is worth noting that the NAM content alerting is situated on the control plane using *link-local unicast* signaling. Neither a data path is established in the PIT, nor are FIBs modified.

The publishing mechanism is depicted in Figure 1(b). A Publisher issues a NAM to its parent, which requests the content and republishes the NAM to the CP in parallel. After arrival of the data, nodes satisfy outstanding Interests up to the CP.

Under irregular network conditions, a node may not receive an Interest that matches its previous name advertisements. This may be due to broken links, failing or deep-sleeping nodes, or enduring overload. After a deployment-specific timeout, the content owner will adapt and try to publish the content on an alternate path by sending a NAM up on a backup link. In case of a complete failure, the content node can follow two strategies: Either it waits and re-advertises according to an exponential back-off, or it solicits a refresh of router advertisements for learning new, operational routes.

C. Subscribing to content

A subscriber in HoPP behaves almost like any content requester in NDN. It issues a regular Interest request up the default route to the CP and awaits the response. There are two deviations from plain NDN, though. First, the subscriber cannot extract content names from its FIB, since FIBs only contain prefixes. Second, it does not expect an immediate reply, but issues Interests with extended lifetimes.

Names are expected to follow an application-specific logic. Names of individual content items can be learned by issuing Interests on topics. The corresponding CP will then answer the request with an empty data chunk that carries available content name(s) as metadata.

Figure 1(b) displays the operations of a subscriber. An Interest for named content is sent up to the proper prefix owner (CP) and remains for a predefined lifetime, if the Content Proxy cannot supply the data. In case content is arriving from a publisher to the CP, data is transferred automatically down the reverse Interest path—as a regular NDN operation. We anticipate that in common sensor-actuator networks of the IoT, the application semantic will define meaningful Interest lifetimes. Otherwise, in regimes of largely fluctuating temporal behaviours or long-lasting subscriptions (e.g., alerts), the subscriber may refresh and maintain the request at its discretion.

Note that in contrast to *long-lived Interests* or the COPSS *subscription tables*, such Interests of extended lifetime are consumed by arriving content and do not open a persistent, uncontrolled data path. Subscribers continue to apply flow control and may discontinue subscriptions to unwanted content.

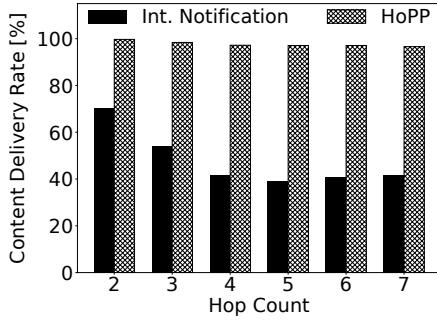


Fig. 2. Success rate of content delivery to one consumer as a function of hop count in the Grenoble multi-hop testbed

III. EVALUATION

A. Basic Testbed Setup

All experiments are conducted in the FIT IoT-LAB testbed [10] to reflect common IoT properties. The testbed consists of several hundreds of class 2 devices equipped with an ARM Cortex-M3 MCU, 64 kB of RAM and 512 kB of ROM, and an IEEE 802.15.4 radio (Atmel AT86RF231). The radio card provides basic MAC layer functions implemented in hardware, such as ACK handling, retransmissions, and CSMA/CA. The software platform is based on RIOT [11] and an extended CCN-lite network stack.

The performance of the HoPP publish-subscribe IoT system is evaluated on the three different topologies:

Paris is a densely connected topology of 69 nodes all within radio reach.

Grenoble (ring) is formed of a closed rectangle with two double-stacked edges. 178 nodes form a heterogeneously meshed network with a maximal hop distance of four.

Grenoble consists of about 350 nodes, where half of them is situated on the rectangle, the other half forms linear extensions leading outwards. This network supports complex topologies with a node distance up to 9 hops.

B. Performance evaluation

The first evaluation inspects the reliability of HoPP as compared to plain Interest notification [12]. We investigate the content reception rate on a given consumer in the Grenoble ring multi-hop topology using a converge cast traffic pattern, where each device generates sensor readings every 30 ± 15 seconds.

Figure 2 compares the reliability of HoPP with the common Interest Notification approach in relation to the hop distance of the consumer. For HoPP, we observe a steady high content delivery rate above 96 % for all hop distances in the topology. NDN Interest Notification admits significantly lower reliability and shows a decline in transmission with increasing hop distance. While a hop count of 1 yields 70 % packet arrivals, success ratio decreases to 41 % for hop distances of 5 and larger. Next, we investigate performance metrics that relate to the temporal behaviour of the protocol. Since deficits of the core protocol, but also different failures of networked elements

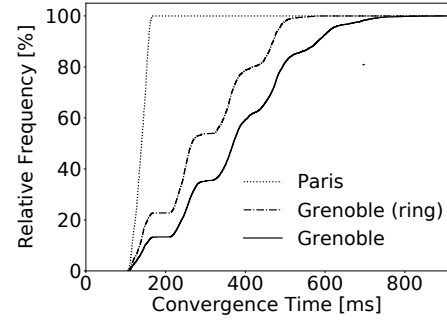


Fig. 3. Routing convergence time for the testbed topologies

(radio/link layer, CCN-layer, pub-sub, and node layer) translate into delays due to retransmissions and re-arrangements, times to completion are a key performance indicators. In detail, we study (i) routing convergence, (ii) times to publish content items, (iii) times to publish under network partitioning, and (iv) times to issue alerts (from publisher to the subscribers).

Routing convergence times in the three testbeds are displayed in Figure 3. Clearly visible is the dependence on hop counts, each counting for an average delay of ≈ 100 ms—the PAM timer. While Paris is single-hop network and exhibits a single step in distribution, multiple steps represent hop count multiplicities in the multi-hop cases. No exceptional delays become visible. This is due to the moderate timing of the routing protocol which causes a low network utilization.

For the evaluation of the times needed to publish a content item, we iterate the following. For each topology, a Content Proxy is positioned in the center of the network, while randomly chosen nodes publish a single, individually named chunk to the network. Publication is initiated every second and depending on the nodes position in the tree, one to several data packets traverse the same sub-paths within this time frame.

Results for the single-hop network (Paris) are displayed in Figure 4(a). Observing round-trip ping values of ≈ 10 ms, the NAM timer (nam_t) of 125 ± 25 ms, and the CCN-lite processing, a mean time to publish of about 135 ms would be expected. Small fluctuations at $\approx 2 \times \text{nam}_t$ indicate additional delays that result from network disturbances and node congestion leading to paths of hop count two.

Similar results become visible from the Grenoble experiments in Figs. 4(b) and 4(c). Clearly pronounced are the first four routing hops, higher hop counts in Fig. 4(c) blur according to increasing fluctuations. These results clearly show the fragility of the lossy wireless regime, but also confirm a majority of these challenging transmissions did complete on the expected time scale.

Finally, the end-to-end delay from the publisher to the subscriber was examined. This corresponds to the use case of issuing alerts between nodes from the local IoT network. The scenarios correspond to the previous measurements of the publishing time, i.e., publishing and subscription requests are issued randomly scattered within the topology at intervals of one second.

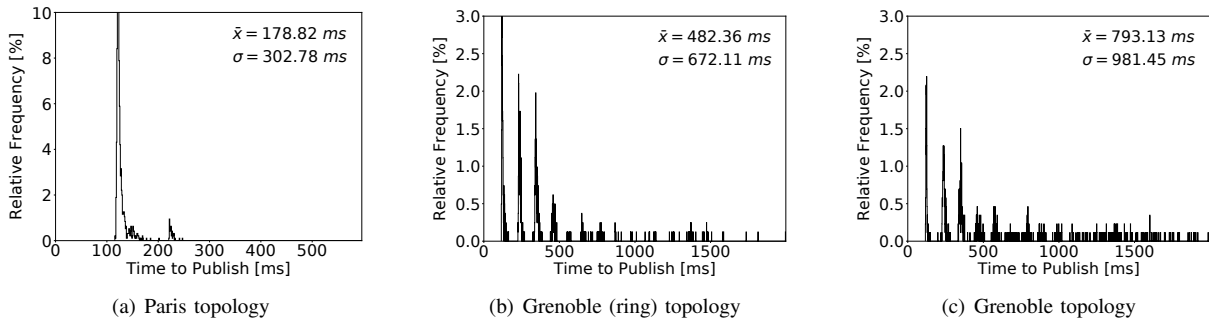


Fig. 4. Time to content publishing

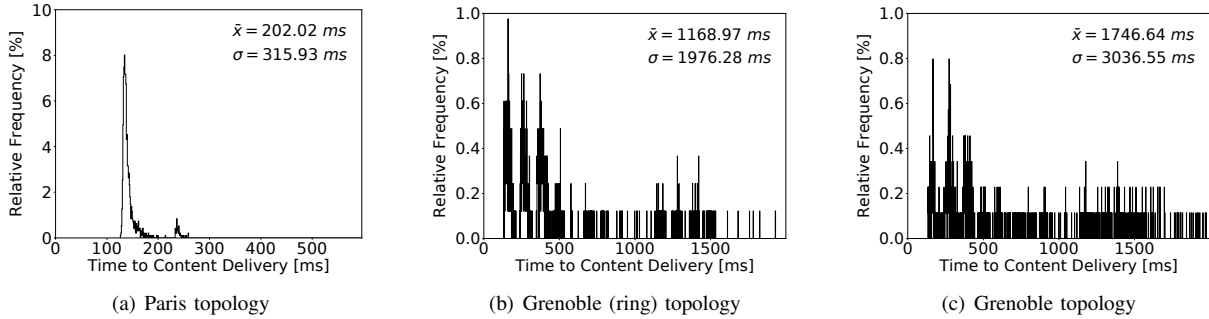


Fig. 5. Time to issue alerts

The experimental output for the three topologies are displayed in Figs. 5(a), 5(b), and 5(c) respectively. As we might expect, blurring fluctuations have enhanced with only a few pronounced signatures of hops and the means increased slightly by the extended paths towards the subscribers. Notably, the single-hop testbed from Paris performed best under the extended communication load, whereas the full Grenoble testbed clearly runs at its limit. The latter can be easily explained by the many hop transitions required at Grenoble, each of which requires an additional packet exchange which potentially impacts on neighbors within radio range.

Low power lossy networks that connect heavily constrained IoT nodes are known to be infeasible for such heavy load. We consider it a success that a notable fraction of the content arrived at its receivers on within about 500 ms – a timescale which is considered normal in multi-hop WPANs. To a certain degree, we account this for the robustness of our hopwise content publishing and replication protocol. Further evaluations [13] confirm these observations.

IV. CONCLUSIONS AND OUTLOOK

In this work, we revisited Information-Centric Networking in the IoT from a variety of perspectives and concluded that (a) publish-subscribe with named topics largely facilitates to manage the complexity of naming data, and (b) NDN without a push option for data has striking advantages for robustness in constrained environments. We introduced HoPP, a lightweight publish-subscribe system that was implemented on RIOT and CCN-lite and experimentally evaluated on large, realistic testbeds. Our findings confirmed that our approach turned out

robust and resilient while performing well in the majority of experiments. In future work, we will work towards prototypic deployment in more intricate use cases. Prior to that, we will study mobility and disruption tolerance in closer detail using multi-proxy set-ups and content redundancy. Adding an analytic model that complements our understanding of the different protocol control loops will be valuable for optimizing parameters and the overall performance.

REFERENCES

- [1] B. Ahlgren, et al., “A Survey of Information-Centric Networking,” *IEEE Communications Magazine*, vol. 50, no. 7, pp. 26–36, July 2012.
- [2] G. Xylomenos, et al., “A Survey of Information-Centric Networking Research,” *IEEE Comm. Surveys and Tutorials*, vol. 16, no. 2, 2014.
- [3] D. Lagutin, K. Visala, and S. Tarkoma, “Publish/Subscribe for Internet: PSIRP Perspective,” *Future Internet Assembly*, vol. 84, 2010.
- [4] A. Carzaniga, M. Papalini, and A. L. Wolf, “Content-based Publish/Subscribe Networking and Information-centric Networking,” in *Proc. of the SIGCOMM WS ICN '11*. ACM, 2011, pp. 56–61.
- [5] J. Chen, et al., “COPSS: An Efficient Content Oriented Publish/Subscribe System,” in *ACM/IEEE ANCS'11*. IEEE, Oct. 2011.
- [6] V. Jacobson, D. K. Smetters, J. D. Thornton, and M. F. Plass, “Networking Named Content,” in *ACM CoNEXT'09*. ACM, Dec. 2009.
- [7] C. Gündogan, et al., “ICN Adaptation to LowPAN Networks (ICN LoWPAN),” IRTF Internet Draft – work in progress 02, July 2018.
- [8] C. Gündogan, et al., “Demo: Seamless Producer Mobility for the Industrial Information-Centric Internet,” in *ACM MobiSys*. June 2018.
- [9] T. Winter, et al., “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks,” IETF, RFC 6550, March 2012.
- [10] “IoT-LAB: a very large scale open testbed,” <https://www.iot-lab.info/>.
- [11] E. Baccelli, et al., “RIOT OS: Towards an OS for the Internet of Things,” in *IEEE INFOCOM. Poster*. IEEE Press, 2013.
- [12] M. Amadeo, et al., “Named data networking for IoT: An architectural perspective,” in *EuCNC*. IEEE, June 2014, pp. 1–5.
- [13] C. Gündogan, et al., “NDN, CoAP, and MQTT: A Comparative Measurement Study in the IoT,” in *ACM ICN*. ACM, September 2018.