Information Centric Networking in IoT scenarios: the Case of a Smart Home

Marica Amadeo, Claudia Campolo, Antonio Iera, Antonella Molinaro
University "Mediterranea" of Reggio Calabria - DIIES Department
Email: {name.surname}@unirc.it

***Abstract*—The Information-Centric Networking (ICN) paradigm for the future Internet is fundamentally different from the classic host-centric Internet Protocol (IP). By leveraging unique, persistent and location-independent content names, ICN provides native multicast support, content-based security, in-network caching, and easy data access, which can be especially useful in the Internet of Things (IoT).**

In this paper, the attention is on the design of an ICN framework tailored to the *smart home* domain, considered as a major representative of IoT scenarios. The proposed solution encompasses the definition of a flexible and expressive naming scheme that supports data/command exchanges and configuration/management operations, and also fits the common service models in the smart home domain (i.e., push, pull, multi-party). Use cases are provided to shed light on the system behaviour and preliminarily assess its potential and performance.

***Index Terms*—Information Centric Networking, Internet of Things, Smart Home, Named Data Networking**

I. Introduction

The proliferation of low-cost embedded sensors/actuators and the advancement in wireless technologies enabling the connection of small devices to the Internet have pushed towards the *Internet of Things (IoT)* vision, where every-day physical objects turn into smart things that sense, understand, and react to their context.

So far, different networking solutions have been proposed to bring IoT to life. In addition to the *evolutionary host-centric* approach based on the Internet Protocol (IP) [1], also revolutionary solutions, such as Information Centric Networking (ICN), have been recently considered [2].

ICN defines a new communication paradigm centred around *named contents* [3]: end users and network nodes do not use IP addresses for content retrieval, but they directly leverage content names. This makes every ICN content packet a self-identifying and self-authenticating unit, with a unique, persistent, location-independent name. As a consequence, in-network data caching is enabled and any node in the network may provide the requested content, so that content consumers and producers do not need to be simultaneously connected.

In our previous work [4], we proposed ICN as an enabling paradigm for IoT and showed the main benefits and open issues related to the deployment of a high-level IoT architecture based on named contents. We pointed out that the main ICN building blocks (e.g., naming, forwarding, routing), originally conceived for content dissemination in the Internet core, must be extended and customized to address the IoT traffic patterns and the presence of resource-constrained devices with little-to-none caching capabilities.

In this paper, the smart home has been chosen as a representative IoT use case, where monitoring and control functions need to be supported for a wide range of applications (e.g., user comfort/care, smart energy management) [5]. Almost the totality of protocols designed for smart home environments are based on proprietary solutions, thus sacrificing interoperability. Efforts for open, IP-based, standards are underway to ensure global access to services and information, also in home environments [6]. The 6LoWPAN stack [7] and the Constrained Application Protocol (CoAP) [8] are aimed, respectively, to support IPv6 in low-power and lossy networks (LLNs) and to extend the HyperText Transfer Protocol (HTTP) web service paradigm into LLNs. However, such solutions could inherit some of the drawbacks of IP when dealing with intermittent connectivity and challenged networks [1], [9]. At the same time, ICN holds great promise for building smart home management systems, because it simplifies network configuration, data retrieval, and service access, and also provides inherently security at the network layer [10], [11].

In such a context, after identifying the main features and requirements of the smart home ecosystem (Section II), this paper contributes to *(i)* surveying existing ICN works that target smart-home networking and automation (Section III); *(ii)* designing an ICN framework, which we refer to as *NDOMUS (Named Data netwOrking for sMart home aUtomation Systems)*, that lays its foundations in the ICN architecture for IoT we abstracted in [4] and focuses on the definition of a flexible naming scheme and forwarding procedures to support the typical service models in a smart home (Section IV); *(iii)* referring some concrete use cases to figure out how NDOMUS will work in practice and providing a preliminary performance assessment against a CoAP solution (Section V) prior to conclude (Section VI).

II. The Smart Home: an Overview

A smart home is a very heterogeneous environment, characterized by several types of devices, different connectivity technologies, applications, and service patterns.

In the house, a high number of small *end devices* (EDs), like sensors or actuators, are embedded in home appliances.

This work has been carried out within the national research project PON03PE_00050 DOMUS "Home automation systems for a cooperative energy brokerage service".

978-1-4673-6432-4/15/$31.00 ©2015 IEEE

They produce information or execute tasks according to their capabilities. A *home server (HS)* interacts with EDs (either directly or through other intermediate nodes) and runs the application logic devoted to monitor/control the status of the house, by taking decisions to satisfy the user preferences. It is linked to a *home gateway (HG)* that enables connectivity to the Internet. It may also interact with the smart meter that collects local data about power consumption.

In the following, we report a set of basic features and demands from a networking perspective, which must be considered in the design of the ICN framework targeting the delivery of smart home applications.

Local and global connectivity. Data produced in the smart home may be *locally consumed* and shown to the house's owner on the control panel's screen, or they are processed in order to trigger actions (e.g., switching on the air conditioning if the temperature is above a specified threshold).

Data may also be delivered to a *remote consumer* for further processing and decision. For example, they can be visualized on the smartphone of the home owner when she is outside. Data about energy usage may be delivered to the utility provider, e.g., to optimize the electrical power generation and distribution to its subscribers.

Wireless networking. The usage of wireless technologies provides new opportunities in terms of flexibility. However, the home is typically a multipath environment with also high interference, which could induce high packet error rates and losses. Therefore, robust transmission schemes should be conceived if reliability is demanded by the application.

Security. Smart home applications may rely on highly sensitive information, which requires privacy, integrity and authentication support. The same data could be requested by different consumers which are located in different administrative domains; thus, a strong security support is a must to protect information independently from the channel/connection over which it travels. At the same time, EDs must be protected from external intrusions, which could create several malfunctioning, like denial-of-service (DoS) attacks.

Service models. Smart home applications exhibit different service models with specific quality constraints. Some applications can tolerate variable *delays* up to a few seconds, while others require real-time interactions. Without loss of generality, we can identify two basic service models, namely *pull* and *push*. A pull service captures a wide range of *(i)* control applications, where the execution of actions is required, and *(ii)* monitoring applications, where sensing information is required. Vice versa, a push service encompasses unsolicited data transmissions, like real-time alarm messages.

Multi-party communications. Depending on the set of entities involved in the communication, we can identify four different transmission modes: *(i)* single consumer-single source (1C:1S), e.g., a switch transmits a command to a light fixture; *(ii)* single consumer-multiple sources (1C:MS), e.g., the HS asks all temperature information from sensors in the house; *(iii)* multiple consumers-single source (MC:1S), e.g., a set of remote applications/entities is interested in the *same data* produced within the home network; *(iv)* multiple consumers-multiple sources (MC:MS), combining *(ii)* and *(iii)*.

III. ICN in the Smart Home Domain

So far, a few very recent works have been proposed that leverage ICN to enable building automation systems and/or home networking [10]–[13]. Among ICN instantiations [3], they rely on the *Named Data Networking* (NDN) [14], which defines a *receiver-driven pull-based* communication model based on the exchange of two packets types, *Interest* and *Data*, used to request and transfer the content, respectively.

NDN packets carry *hierarchical* (sometimes user-friendly) content names, which have a variable number of components with unbounded lengths. Thus, different ad hoc defined namespaces can be designed for smart home applications [10], [11]. Although traditionally ICN names are related to content objects, in the IoT domains they can also be used to logically identify the functionalities of devices, without the need to resolve them into devices' addresses. NDN effectively moves the process of name resolution from the application layer to the network layer, thus facilitating service discovery and delivery.

In [10], the initial design of a secure Building Management System (BMS) is presented, with focus on a data sensor acquisition system that implements encryption-based access control. The system is composed of a hybrid IP-NDN network: it uses a proprietary protocol over IP for the communication between sensors and the gateway that collects the sensing data. The gateway publishes data in NDN repositories and responds to the Interests of authorized users, uniquely identified by public keys. The case of securing a lighting control system running over NDN is discussed in [12]. The proposed framework includes a *configuration manager*, which assigns fixtures and applications their NDN namespace and identity, represented by a unique public/private key pair; and an *authorization manager*, which determines the applications allowed to access each fixture, signs applications' public keys and issues signed access control lists.

Security is also the main topic of the work in [13], where a named data home energy management system (iHEMS) is proposed. A new publish-subscribe layer is created above NDN to allow the producer advertising and publishing its contents, and the subscribers getting the content whenever a new one is available.

In [11], the initial design of a ICN based homenet is presented with a focus on naming and service configuration and a comparison against an IPv6-based home network is shown. Names are a hierarchy of components that identify the service, the device and the scope, thus reflecting service access restrictions. An ICN prototype of the discovery protocol for low-cost configuration is implemented. However, the ICN logic is only in the home gateway and interior routers, i.e., the *powerful* nodes, but not in the sensors. Unlike previous literature, our study has the threefold objective of:

- presenting an ICN framework, NDOMUS, that brings NDN in the home domain, even in constrained sensing and actuator devices;

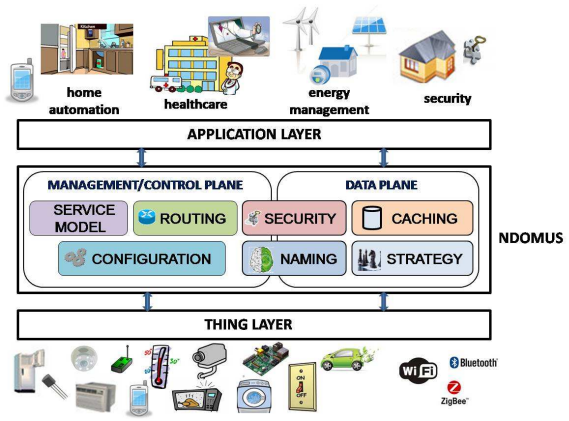

Fig. 1. The NDOMUS framework.

- dissecting the namespace definition and the support of different service models (i.e., push and pull) and multi-party communications (i.e., 1C:MS and MC:1S) as required by smart home applications;

- illustrating the benefits of NDOMUS in supporting the delivery of smart home applications through some representative use cases and a preliminary evaluation, against the plain NDN and a CoAP solution.

IV. Our NDOMUS proposal

The NDOMUS framework in Fig. 1 is based on the NDN-IoT architecture that we introduced in [4]. In this paper, the NDN-IoT functionalities, originally abstracted at a high level, are specified for the smart home domain.

The reference home scenario is depicted in Fig. 2, where the HS (for the sake of simplicity and without loss of generality, co-located with the HG) communicates with a set of EDs. All nodes implement the NDN building blocks designed for the conceived *Data* and *Management/Control* planes, but with different restrictions that depend on their role and resource availability. The Data plane is responsible for handling the individual packets, both Interest and Data, and operations on top of them (e.g., naming, caching, forwarding, retransmissions, security). while the Management/Control plane accounts for configuration and control functions onto the Data plane.

In the following, we focus on three main NDOMUS features: naming scheme (a *cross-plane* functionality strictly related to configuration and security operations), service model, and strategy for multi-party communications.

A. Naming scheme

In NDOMUS, application-specific and human-readable names support both sensing/action and management/configuration operations in the house. For this purpose, we identify two sub-namespace classes: *(i) Configuration and management* namespace, identified by the prefix */conf*, used for home network initialization, configuration updates and management operations; and *(ii) Task* namespace, identified by the prefix */task*, used to identify and enable all the control and monitoring operations.

With such a classification in mind, we build a *name tree* to represent a highly flexible and extensible global namespace, which is also meant to easily support multi-party communications. The root node is a logical name component that uniquely identifies the house referred in the following as to */homeID*. It can be related to the geographic location of the house and/or to a owner identification number. According to the NDN routing protocol, the HS may announce the */homeID* prefix in the core network to advertise the availability of its smart home services and allow global reachability.

The */homeID* prefix is followed by two main branches for */conf* and */task* sub-namespaces, under which different names can be composed, as explained below.

Configuration and management sub-namespace. Set-up operations in the home network, including EDs discovery and configuration, are identified by the prefix "*/homeID/conf/*". Similarly to the solution in [12], we assume that a configuration and authorization manager is in charge of registering the EDs by assigning the namespaces under which they can operate, together with the related keys and other information for security management (e.g., access control list, encryption mechanism). We assume that the manager is co-located in the HS, which maintains a list of active EDs and their parameters. EDs periodically broadcast *keep-alive* messages in Interest packets with a refresh rate of the order of some minutes or more. The HS removes a device from its EDs list if it does not hear from it for a certain interval and, eventually, notifies the owner. Any application (including the owner's one) that wants to access the home network to collect secure sensitive data or to trigger an action, must also interact with the HS to receive configuration instructions.

Task sub-namespace. Task names describe the type of operations to be carried out by EDs, such as measurement reporting or action execution. These names are used in both Interest and Data packets.

NDOMUS adopts the following basic name structure: */homeID/task/type/subtype/location/*, where the *type* component specifies the task type (*sensing* or *action*); the *subtype* component describes which specific sensing or action task must be performed (e.g., temperature sensing); and the *location* component identifies the physical position of the ED in the house. The proposed name structure can be easily extended by including new sub-task and locations.

Fig. 3 shows an easy example of task name tree, where light

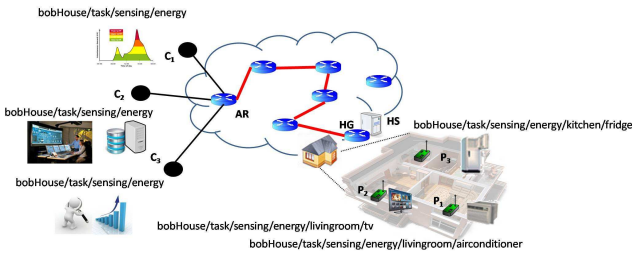

Fig. 2. Reference smart home scenario with several consumers and producers.

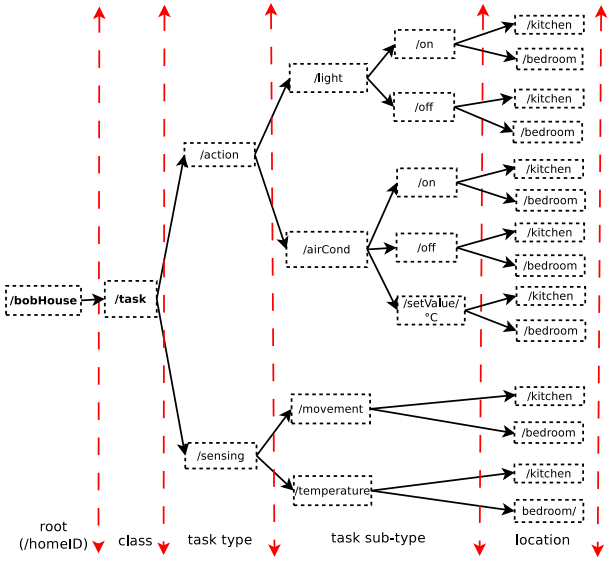

Fig. 3. Name tree example for task sub-namespace.

and air conditioning are associated as action subtype, while movement and temperature are considered as sensing subtype. An Interest with name */bobHouse/task/action/light/on/kitchen* is issued to require the kitchen light fixture to turn on. After the task execution, the ED can send a Data packet with the same name while the payload includes a boolean value as a result of the operation. In case of failure, the payload can also include a specific explanation code, e.g., the bulb burnt out.

Although NDN potentially imposes no restrictions on the name structure, it is worth noticing that some IoT access technologies (e.g., IEEE 802.15.4) support small payloads. Thus, names should be maintained *thin* and preferibly not exceed the length of the required content also to avoid packet fragmentation. To this aim, when HS and EDs agree upon the namespace in the initial configuration stage, they can also share a dictionary storing the mapping between long names and shorter name versions to be used locally.

B. Service model

From the service model perspective, smart home applications can be classified in three main categories: pull, *periodic* push, and *event-triggered* push. NDOMUS supports all the three models and provides Interest retransmission routines to guarantee reliable delivery, as defined by the strategy function.

1. Pull. This service model is naturally supported by NDN: the consumer application sends an Interest to ask for a measurement or an action, and waits for a Data packet from the ED containing the requested parameter value or the result of the triggered action. For security purposes, both Interests and Data are *authenticated* with proper encryption mechanisms. For a guaranteed deliver, in case no Data are received within the Interest timeout expiration, the Interest is retransmitted by the consumer. In NDOMUS, the HS is in charge of autonomously retransmitting unsatisfied Interests which are originated by a remote application. The Data is processed by the HS and, eventually, cached and forwarded to the remote consumer, which shall use longer Interest retransmission timeouts.

2. Periodic pushing. Many home applications require measurements at fixed intervals, typically minutes (e.g., 15 minutes for energy consumption values). Data may be stored for long-term statistics and optimizations. This service can be modelled in two ways: *(i)* the HS periodically pulls the ED with an Interest; *(ii)* the ED autonomously sends an unsolicited packet at regular intervals. While the first strategy works according to the vanilla NDN pull logic, the second one apparently violates the NDN primitives because information is sent without any Interest solicitation. However, the second solution can be implemented by embedding data in the Interest itself; i.e., the ED sends the generated information, which is typically very short, as the last component of the global name. For this reason, the packet is re-named as *Interest Notification* [4]. After sending the notification, the ED waits for a dummy Data packet that is sent by the HS as an acknowledgement, otherwise it retransmits the notification.

3. Event-triggered pushing. The occurrence of some events, like the detection of the owner entering the house, can originate the unsolicited transmission of Data, e.g., for turning on the lights. If the event is an alarm, it must be reported with the highest priority and timeliness. A periodic pulling from the consumer application is therefore unfeasible in this context, since it could lead to intolerable delays and would waste resouces. Vice versa, an *Interest Notification* is again the viable strategy.

C. Strategy for multi-party communications

In addition to the 1C:1S communication pattern, multi-party communications are quite common in smart home applications.

Multi-consumer (MC:1S) communication is natively supported in NDN through Interest aggregation and Data caching. In particular, the HS, which bridges the entire home network to the Internet, can receive many different requests from external consumers interested in getting information about the house and the status of its appliances. The HS may answer by using the Data cached in its Content Store, if not stale, with the twofold purpose of speeding-up data retrieval and limiting the access to the EDs, thus also saving network and energy resources and protecting EDs from DoS attacks.

Multi-source (1C:MS) communication in vanilla NDN requires the transmission of multiple separate Interests to retrieve Data from different producers, because *one Data is intended to consume only one pending Interest*. In NDOMUS, instead, the designed namespace enables the transmission of *a single request* to order tasks to different producers. We assume that the producers share a common name prefix and are distinguished by different producer-specific name components. For instance, an Interest with name */homeID/task/action/light/off* commands to turn off any light in the house. Triggered nodes answer with a Data packet, where the producer-specific part is appended to the common prefix name, e.g., */homeID/task/light/off/kitchen*, */homeID/task/light/off/bedroom*. By doing so, the identity of

Fig. 4. NDOMUS local communications: Interest/Data packets exchange and relevant naming.

nodes executing the task can be detected and selective retransmissions adequately triggered [15], if needed. This solution is also viable when the number of devices matching a given request is not known in advance (e.g., the set of appliances whose energy consumption is above a given threshold).

Finally, for multi-consumers multi-sources scenarios (MC:MS) the above approaches can be easily combined.

V. Use cases

In order to better illustrate the behaviour and benefits of our ICN-based framework, we provide *(i)* a description of NDOMUS primitives in the local home domain, and *(ii)* a simple comparison of the proposed NDOMUS against the plain NDN and a CoAP-like solution, representative of an IP-based host-centric approach.

NDOMUS in the local home domain. Let us refer to the following use case. The surveillance system detects the presence of the owner back from work and sends an Interest notification to the HS, which in turn confirms the reception of the packet. To warmly welcome her, the HS is programmed to check the temperature status in all the rooms and adequately set the air conditioning. To this aim, it issues a single 1C:MS Interest packet and gets back the required measurements from the sensors in the house. The application logic infers that the temperature is too high in the bedroom and asks the air conditioning to be switched on, by setting the temperature value according to owner's preferences. The overall exchange of Interest/Data packets is shown in Fig. 4.

NDOMUS in a global scenario. We consider the reference use case in Fig. 2 for an initial performance comparison between NDOMUS and potential alternatives (CoAP, NDN) in terms of number of exchanged packets. This simple metric is considered critical in the reference scenario, since it gives information about both the used network resources (limited in wireless environments and close to be exhausted in the core network) and the devices' energy consumption (serious for resource-constrained devices). For the sake of simplicity, we assume that *(i)* no packet losses occur and *(ii)* the size of each exchanged packet fits the typical payload constraints of IoT access technologies and is similar for CoAP (including 6LoWPAN/UDP header), NDN, and NDOMUS, by neglecting security aspects [9], thus represents not a decisive factor in the differences observed in the following experiments.

Indeed, the analysis is not intended to be extensive but to get preliminary quantitative insights into the behaviour of the proposed scheme when considering a global scenario.

A number of C remote consumers request home energy consumption data for different purposes (e.g., real-time energy billing and distribution purposes, long-term analytics and profiling). We assume that the consumers are connected to the same access router (AR), which is linked to the HS through N wired links of the core network[1]. Upon receiving the requests from the remote consumers, the HS queries all P metering devices embedded into home appliances and sends back a packet conveying the collected data, aggregated according to some rules agreed with the utility companies. In the considered example in Fig. 2, C=3, P=3 and N=5.

When a CoAP-like solution [8] running over 6LoWPAN is implemented, the typical HTTP request/response interaction is used, which is based on the standard GET and PUT methods. The requested resource (i.e., the measurement from a given meter) is identified by a Uniform Resource Identifier (URI). Some resources can be saved over the wireless link since in CoAP the HS issues a single request to simultaneously get responses from multiple producers, by leveraging IP multicasting. Moreover, the HS acts as a *CoAP proxy* and can cache data to satisfy requests from different consumers. The resulting number of exchanged packets is: $n_{CoAP} = 2C + 2CN + (1 + P)$. The first term accounts for request/response packets over the consumer-AR link, the second one for the request/response messages over the N common wired links for each consumer (IP is data-agnostic and does not identify requests to the same resource object), and the third contribution accounts for the number of requests and replies (respectively, 1 and P, thanks to caching in the HS) over the wireless link.

In the case of vanilla NDN, an Interest with the name *bobHouse/task/sensing/energy* is issued by each remote consumer. The AR aggregates the Interests with the same name, so that they are not duplicated on the commnon N links; hence a single Interest reaches the HS. This is different from CoAP that, being data-agnostic, does not allow aggregation of requests at the AR. P Interests are then transmitted by the HS to retrieve data *from each producer*, so the amount of exchanged packets is: $n_{NDN} = 2C + 2N + 2P$.

NDOMUS operates like the plain NDN until the single aggregated Interest arrives to the HS, then the HS with a single *Interest* broadcasted over the wireless link gets back individual replies from the P metering devices in the house. Each producer replies with a Data packet carrying a name that combines

[1]The core network topology could be quite more complex in real scenarios, but a simple abstraction is deemed sufficient to get a first rough performance comparison.

the prefix in the Interest *bobHouse/task/sensing/energy/* with the producer-specific component that uniquely identifies it, e.g., *kitchen/fridge*. Once the HS receives all the Data from the P producers (whose identity and number is supposed known from the configuration phase), it sends back a single aggregated Data packet to the remote consumers. The number of exchanged packets is: $n_{NDOMUS} = 2C + 2N + 1 + P$.

Fig. 5 shows the overall number of transmitted packets in the above analysed cases when P is variable and C=1 and 3, N=0 and 5.

When P=0 (i.e., single source) and N=0, the three schemes exhibit the same performance: no aggregation of requests can be performed in NDN and NDOMUS and a single request is issued to query the producer. As P increases, being C=1, NDOMUS behaves like CoAP (solid blue line) and outperforms the plain NDN: a single Interest is issued to retrieve Data from multiple producers. With increasing values of C and N, the advantages of NDOMUS clearly emerge: *(i)* by aggregating common Interests at intermediate points, as in the plain NDN, it reduces the load over the core network compared to the CoAP case, and *(ii)* by issuing a single Interest to retrieve Data from multiple producers, through a 1C:MS Interest, similarly to CoAP, NDOMUS allows halving the load over the wireless link w.r.t. NDN, so reducing the stress on (likely) constrained producer devices.

Overall, NDOMUS needs the smallest number of packet transmissions to collect Data from all the producers, both in the core and in the wireless segments. This is an important achievement in the IoT domain, where *big data* are expected to characterize the future traffic demands.

It is also worth remarking that, despite the operation similarity over the wireless hop towards end-devices, CoAP and NDOMUS exhibit some differences:

• CoAP leverages the IP multicast capability to reach the end-devices, and this requires additional signalling (not included in derived formulas) for the join/leave operation of each producer to the multicast group and for allocation of the group 's URI. NDOMUS easily allows groups to be created without additional signalling, thanks to the expressiveness and granularity of the conceived naming scheme.

• Being CoAP messages transported over the connectionless User Datagram Protocol (UDP), message delivery is not guaranteed in the case of multicasting, whereas Interest retransmission routines like in [15] can achieve reliable message transmissions in NDOMUS.

• The successful delivery of a CoAP message is conditioned on the resolution of the URI into the IP address of the device; this phase is not needed in NDOMUS.

As an additional remark, it would be expected the RAM/ROM footprint of NDOMUS, relying on NDN, to be lower than a 6LoWPAN/CoAP solution [1], [9].

VI. Conclusion

In this paper we have investigated ICN as a viable solution to support smart home applications. The study capitalizes on the design of an ICN framework that lays its foundations into the NDN instantiation, and is adequately conceived to enable typical operations and traffic patterns required in the smart home domain. Common use cases have been presented and a preliminary evaluation has been also reported to provide a rough estimate of the footprint of NDOMUS packet transmissions. NDOMUS exhibits encouraging performance motivating a more elaborated comparison as a future work, together with a test-bed deployment that will leverage low-cost off-the-shelf devices, like Arduino and Raspberry Pi, and open source NDN software tools.

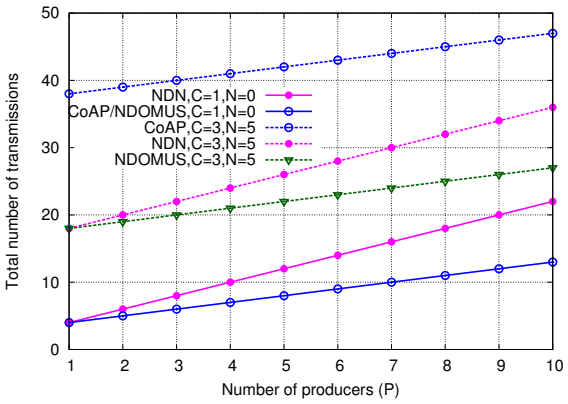

Fig. 5. Total number of transmissions when varying the number of producers P and for C=1 and 3 and common links N=0 and 5

References

[1] Z. Sheng, *et al.*,"A Survey on the IETF Protocol Suite for the Internet of Things: Standards, Challenges, and Opportunities," *Wireless Communications, IEEE*, vol. 20, no. 6, pp. 91–98, 2013.

[2] A. Lindgren *et al.*, "Applicability and Trade-offs of Information-Centric Networking for Efficient IoT," in *Internet-Draft*, December 2014.

[3] B. Ahlgren, *et al.*, "A Survey of Information-Centric Networking," *Communications Magazine, IEEE*, vol. 50, no. 7, pp. 26–36, 2012.

[4] M. Amadeo, C. Campolo, A. Iera, and A. Molinaro, "Named Data Networking for IoT: an Architectural Perspective," in *IFIP EuCNC*, 2014.

[5] M. A. Zamora-Izquierdo, J. Santa, and A. F. Gómez-Skarmeta, "An integral and networked home automation solution for indoor ambient intelligence," *Pervasive Computing, IEEE*, vol. 9, no. 4, pp. 66–77, 2010.

[6] O. Bergmann, K. T. Hillmann, and S. Gerdes, "A CoAP-Gateway for Smart Homes," in *IEEE ICNC* 2012, pp. 446–450.

[7] Z. Shelby and C. Bormann, *6LoWPAN: The Wireless Embedded Internet*. Wiley, 2009.

[8] Z. Shelby, K. Hartke, C, Bormann, and B. Frank, "Constrained Application Protocol (CoAP), draft," *RFC 7252, The Internet Engineering Task Force–IETF*, 2014.

[9] E. Baccelli, *et al.*, "Information Centric Networking in the IoT: Experiments with NDN in the Wild," in *ACM ICN*, 2014.

[10] W. Shang, *et al.*, "Securing Building Management Systems Using Named Data Networking," *IEEE Network*, 2014.

[11] R. Ravindran, *et al.*, "Information-Centric Networking based Homenet," in *IFIP/IEEE ManFI Workshop*, 2013.

[12] W. Shang, *et al.*, "Securing Instrumented Environments over Content-Centric Networking: the Case of Lighting Control and NDN," in *IEEE Infocom NOMEN Workshop*, 2013.

[13] J. Zhang, Q. Li, and E. M. Schooler, "iHEMS: an Information-Centric Approach to Secure Home Energy Management," in *IEEE SmartGridComm*, 2012.

[14] L. Zhang *et al.*, "Named Data Networking (NDN) Project," PARC, Tech. Rep. NDN-0001, October 2010.

[15] M. Amadeo, C. Campolo, and A. Molinaro, "Multi-source Data Retrieval in IoT via Named Data Networking," in *ACM ICN*, 2014.