IoT Applications and Services in Space Information Networks

Manlio Bacco, Luca Boero, Pietro Cassará, Marco Colucci, Alberto Gotta, Mario Marchese, Fabio Patrone

✦

Abstract—The rate of traffic generated by Machine-type Communications (MTC) is growing at a very fast pace, putting under strong pressure existing network infrastructures. One of the most challenging tasks at today is being able to support machine-to-machine (M2M)/Internet of Things (IoT) data exchanges by providing connectivity between any pair of M2M devices all over the world. For this reason, we analyse the role of Space Information Networks (SINs) in supporting MTC, in this work. Horizontal solutions are analysed herein in order to allow interworking by acting as relay entities among different protocol stacks and services, vertically implemented over different network segments. We analyse the still pending challenges hampering interworking, and propose a possible protocol stack for M2M/IoT communications based on the oneM2M standard. Eventually, this paper compares the performance achievable by using two of the most diffused application protocols, Constrained Application Protocol (CoAP) and Message Queuing Telemetry Transport (MQTT), shedding light on their efficiency and differences.

Index Terms—Internet of Things, Machine-to-Machine, CoAP, MQTT, performance evaluation, openM2M, openMTC

1 INTRODUCTION

SPACE Information Networks are complex network infrastructures relying on different network segments as a whole implemented by space platforms, such as satellites, Unmanned Aerial Vehicles (UAVs), High Altitude Platforms (HAPs) and airships, able to support data acquisition and processing in a plethora of application domains [1]. SINs can provide worldwide coverage, thus playing a key role in supporting many different applications: connectivity for otherwise disconnected areas, emergency communications, environmental monitoring, Massive Machine-type Communications (mMTC), and interplanetary communications [2], to cite a few. The network segments composing a SIN show different requirements and characteristics, so that *interworking* must be considered as one of the main objectives. For instance, the recent improvement in small-satellite technologies is making appealing the employment of small-satellite-based solutions in different use cases, included IoT. One practical example among others is the D-SAT project [3], where a flexible Cubesat-based system has been designed and tested to broadcast data generated by sensors spreaded in a certain coverage area. At this time, connectivity between any pair of M2M devices all over the world is one of the most challenging tasks. Such a challenge, which is inherently multifaceted, is discussed in this work according to two different viewpoints: network connectivity, and interoperability for services and applications.

As discussed in [1] and [2], network connectivity challenge should be faced at first by creating a backbone able to exchange data (*i*) among ground stations and (*ii*) among space platforms and ground stations, in both cases with minimal delay. The main objective herein is in broadening the observation area with respect to the capabilities of a single network segment, such as the single satellite portion. Integrating different network segments as seamlessly as possible can prove challenging [4], as well as providing applications and services typically deployed as vertical solutions on top of single network segments. Earth observation, Internet connectivity, cellular connectivity via satellite, environmental monitoring, wide area measurement systems [5] are examples of services that use a single network segment at a time in almost all deployments. If we look at the IoT ecosystem, the absence of a widely adopted standard for MTC is an example of this fragmentation: a commonly adopted horizontal architectural solution enabling *interoperability* among a plethora of application stacks, hardware, and services is still missing [6]. As anticipated, a multifaceted challenge should be tackled in order to move towards a unified network vision, able to glue together heterogeneous hardware and software components.

We describe the network segments composing a SIN in Section 2, considering and discussing the open challenges. In Section 3, we focus on MTC services and applications. We underline the increasing need for open standards, in order to effectively support interoperable MTC scenarios connecting remote things and we propose a relay solution to connect remote M2M/IoT devices in a SIN-based heterogeneous network. In Section 4, we qualitatively compare the achievable performance level of two application protocols in SINs, in order to highlight the different features and their possible role in such networks. The conclusions are drawn in Section 5.

2 NETWORK SEGMENTS COMPOSING A SIN

SINs rely on heterogeneous communication infrastructures composed of sub-networks. Each sub-network includes dif-

---

Manlio Bacco, Pietro Cassará, Marco Colucci, and Alberto Gotta are with the Institute of Information Science and Technologies (ISTI), CNR, Pisa - e-mails: {name.surname}@isti.cnr.it

Luca Boero, Mario Marchese, and Fabio Patrone are with the University of Genoa, Italy - e-mails: luca.boero@edu.unige.it, mario.marchese@unige.it, f.patrone@edu.unige.it

TABLE 1: Comparison of the main characteristics of the network segments composing a SIN.

	UAV	HAP	LEO	GEO
Operating altitude	variable according to the payload weight, engine and wing type: 0.1 - 20 km	stratosphere: 17 - 25 km	above the denser part of the atmosphere and below the inner Van Allen radiation belt: 200 - 2 000 km	35 786 km
One-way propagation delay	very low	low	1 - 15 ms	120 - 140 ms
Operational cost	variable according to the payload type	maintenance may require recovery and redeployment	low, LEOs do not require physical maintenance and external power supply	low, GEOs do not require physical maintenance and external power supply
Deployment timing	very fast deployment	rapid deployment, depending on launch platforms	variable depending on the number of deployed objects per launch and orbital parameters	variable depending on the deployment strategy
Endurance	variable according to the engine type, e.g. battery-powered vs gasoline: 30 min - 1 day	multiple days or even weeks if solar energy generation is used	from few years up to 10 - 15 years	10 - 15 years

ferent network components such as satellites, HAPs, and UAVs which constitute the enabling platforms of the SIN infrastructure, and is composed of different layers with different functionality and features [1], as summarized in Table 1 and detailed in the following.

At the highest altitudes, geosynchronous (GEO) satellites offer the largest coverage, high bandwidth, and permanent availability at the cost of high propagation delays. These features make GEO satellites a core part of the space-based backbone network conveying data and control information coming from the network segments at lower altitudes. GEO satellites can also contribute to gather and exploit real-time information about the status of the underlying networks in order to apply real-time control strategies, such as dynamic resource management and allocation, directly on-board or at a terrestrial control station [1], [2]. Owing to their inherent broadcast capabilities, GEO satellites can be exploited by several kinds of IoT application services. An example comes from the information delivery to a large number of nodes at the same time, as in the cases of Over The Air (OTA) and updates delivery by *broadcasting* or *multicasting* communication modes, and *geocasting*, i.e., transmitting to a set of nodes in a precise geographical area.

Below GEO satellites, Low Earth Orbit (LEO) ones offer communication services with lower latency, but at the cost of reduced coverage and, typically, data rate [7], even if the use of transceivers operating in the band of millimeter-frequencies (E-band) can increase the achievable data spped [1] and allow obtaining transmission rates in the order of tens of Gbps. LEO satellites are periodically visible from the ground only for fixed time windows with respect to GEO satellites. In order to increase the number of transmission/reception opportunities and, consequently, the amount of exchanged data, more LEO satellites can be deployed in constellations and equipped with antennas suitable for Inter-Satellite Links (ISLs). This kind of constellations are very attractive, because new generations of LEO satellites weight just a few kilograms and have reduced Operating Expenditures (OPEX) and Capital Expenditures (CAPEX) with respect to GEO ones. These characteristics make them really appealing to deploy wide-area IoT services [5]. On the other hand, LEO satellite computational power, storage capacity, and energy resources are limited. These limitations become stricter, when the satellite size decreases [7].

Below LEO satellites, HAPs can be grouped in two categories: fixed-wing aircrafts (manned or unmanned) and airships [8]. Over the past years, HAPs have been engaged in several research projects for civil applications, such as broadband wireless access services based on networks of *base stations in the sky*, Earth observation, and disaster monitoring [9]. Project Loon[1], e.g., aims at providing Internet access to rural and remote areas through a network of stratospheric balloons acting as cellular base-stations. HAPs operate in the stratosphere and the covered area is therefore smaller to that of LEO satellites. HAPs are characterized by reduced operational costs with respect to satellite-based solutions, since neither a rocket launch nor a complex terrestrial infrastructure are required. Maintenance tasks require less resources as well: for instance, in the case of malfunctioning of a balloon, it can be recovered, fixed or substituted, and relaunched in a short amount of time, thus guaranteeing a rapid recovery of the service. Most HAPs suffer less of resource limitations related to computational power, storage capacity, and available energy, because they can carry heavier payloads than LEO satellites and are endowed with solar battery charging systems.

The possible deployment of UAVs is located below HAPs. UAVs can cooperate to achieve specific tasks dynamically. If such a cooperation is enabled by an underlying network architecture, the literature refers to it as Flying Ad-Hoc Network (FANET) [10]. FANETs are multi-vehicle networks or *swarms* with an arbitrary network topology (e.g., star, mesh, hierarchical), which provide advantages when compared to single-UAV systems: larger covered area, increased system redundancy, and, on average, lower time needed to complete a given mission. Depending on the employment, UAVs may be equipped with different sensors and actuators and clustered in different classes, even if a classification methodology is not straightforward. The International Telecommunication Union (ITU) proposes a

1. Details about Project Loon can be found at x.company/loon.

three-clusters classification based on the weight, maximum operational altitude, speed, and endurance provided by vehicles: small, medium, and large UAVs. Low Altitude Short Endurance (LASEs) and Low Altitude Long Endurance (LALEs) vehicles, both operating within a 5 km altitude, fall in the first two classes, respectively. Vehicles belonging to the large class are used at medium and high altitudes, within a 20 km altitude, and are referred to as Medium Altitude Long Endurance (MALE) and High Altitude Long Endurance (HALE) UAVs. The flight safeness is a fundamental requirement. Physical collision avoidance is a critical aspect, along with the flying formation. Unlike HAPs, UAVs can typically transport limited loads and, because of low-capacity batteries, they have a rather short endurance. Despite these disadvantages, UAVs play a key role in a variety of application scenarios, such as disaster monitoring, coverage extension, and pipeline inspection [10] owing to low CAPEX and OPEX.

Summarizing, relying on a SIN infrastructure allows achieving communication services by planning the role of each network segment accordingly to its own features and limitations. For example, in a search and rescue scenario characterized by the absence of a terrestrial communication infrastructure, a swarm of UAVs can be employed to closely and repeatedly scan the area of interest. The collected data can be transmitted to the emergency operating center through HAPs and/or a constellation of LEO satellites, according to the urgency and to the available resources. HAPs and LEO satellites can be exploited to provide Beyond Line of Sight (BLoS) coverage between a FANET and its ground control station. Above these altitude layers, GEO satellites can provide control and supervision by gathering and delivering information about ongoing operations to a remote operating center. A graphical representation of a SIN is reported in Figure 1, which also visualizes some examples of services and applications together with the possible interconnection among different network segments acting at different altitudes. We elaborate on possible applications and services in Section 3.

2.1 Open challenges in SINs

Despite the potential advantages offered by SINs, some open challenges still need to be addressed [1], [9].

- *Coverage*: this is one of the most appealing features offered by SINs, but national and international regulations pose some limitations because of the non-uniform spectrum allocation and the different frequency allocation schemes that force the usage of a specific subset of access technologies or frequencies.
- *Airspace regulation*: mainly impacts the network segment elements operating at lower altitudes. Although in the last few years civil aviation organizations have refined the flight rules for UAVs, a cross-country agreement is yet to come.
- *Handoff*: design of effective handoff procedures among heterogeneous access technologies. For instance, the UAV communication payload can be designed to support a primary UAV-to-HAP link, but in the case of tampering or malfunction of the link, the infrastructure may switch the communication to an UAV-to-LEO link.
- *Softwarization*: due to the complexity of these interoperability procedures and on the needed hardware, the definition of SIN-based communication infrastructures is a rather complex task. The use of softwarization techniques can act as a game changer. Software Defined Networking (SDN)/Network Function Virtualization (NFV) paradigms can provide *gluing approaches* able to hide the complexity of the underlying physical network, also providing energy efficiency [11]. SDN/NFV paradigms can support the reduction of resource consumption, by allocating the required resources only for the needed time in an optimized way, and also by relieving flying objects of processing load by centralizing the operative functions. Such an approach is in line with the envisioned architecture to integrate satellite and flying technologies with the upcoming 5G terrestrial network [11].
- *Network stack definition*: Last but not least, M2M/IoT resource-constrained devices poses other limitations, especially in terms of low-power consumption, low computational and storage capabilities, and, for certain applications, long communication range. This requires the definition or the adaption of communication protocols. Large effort has been already spent on standardization and commercialization activities. The result is a vast plethora of defined/ under-definition standards and strictly proprietary solutions [6]. This situation rose the challenge related to the implementation of flexible and lightweight communication protocols, providing Quality of Service (QoS) support, security, and energy efficiency. Issues still to be solved involve all stack layers, from the physical ones, concerning spectrum allocation strategies, modulation techniques, and channel access solutions aim to lower energy consumption and increase throughput, to the higher ones, referring to routing in heterogeneous networks, address allocation, reliability through acknowledgments.
- *Interoperability* among devices of different kinds and producers is a very open issue which is discussed in Section 3.

3 MTC APPLICATIONS AND SERVICES

The upcoming fifth cellular generation will bring a continuous flow of innovative services and applications to be supported by the physical and virtual Radio Access Network (RAN). Softwarization and virtualization of RANs will be implemented thanks to aforementioned SDN and NFV paradigms. In order to enable IoT massive internetworking, the complementary use of aerospace networks will be necessary to deal with such a huge amount of traffic, in particular in areas typically disconnected or poorly served. In this context, SINs may be of great help to integrate different network segments. According to the literature and to some market reports, M2M/IoT markets are still fragmented, since horizontal solutions [6], acting as relay entities to integrate different vertical protocol stacks implemented, even privately, over separate network segments, are not so common. Vertical full turnkey solutions are often privileged, also for commercial reasons. Therefore, a better integration and interoperability is a need for both different network

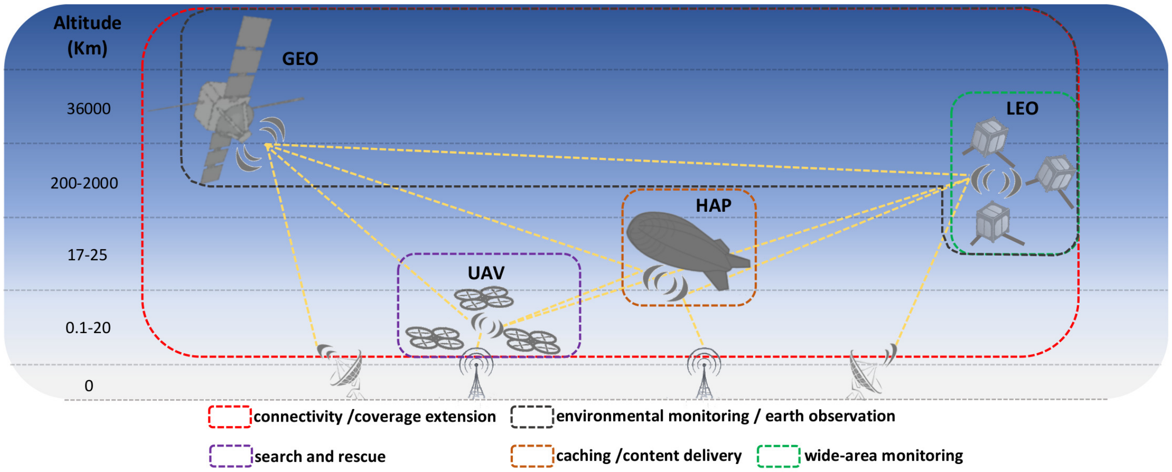

Fig. 1: Network segments composing a SIN at different altitudes. Examples of applications and services are provided, showing the possible interconnections among different network segments.

segments and application scenarios on top of them. An architectural approach is introduced in [4] and applied to IoT in [12].

As anticipated in Section 1, connectivity between any pair of MTC devices all over the world should be considered as one of the most challenging tasks at today. On the one hand, the SIN paradigm is aimed at providing an overall communication network by relying on different network segments as a whole, thus being able to connect any couple of remote endpoints all over the world. Different aspects, such as routing, security, and protocol stacks, need to be tackled through careful studies and policies, thus meeting different requirements and regulations all over the world. On the other hand, there is the need for an open standard to exchange M2M/IoT data. The survey in [13] describes the challenges brought by the need of interoperability of multiple layers of the end-to-end protocol stack, making such a requirement as one of the most significant for the success of this ecosystem. The oneM2M organization has the mission to ensure the alignment of (the babel of) M2M standards. It defines a service layer to exchange data among M2M entities in an agnostic way with respect to the underlying network. In more words, oneM2M develops technical specifications for a service layer connecting the plethora of M2M devices all around the world in an interoperable way, addressing the requirements of different business domains (smart cities, smart factories, smart villages, and so on) sharing such a need. Concerning compliant software implementations of oneM2M, openMTC provides an open source implementation in the form of a *middleware*, which can be used to integrate different devices in a horizontal way. Figure 2 depicts the logical architecture for connecting M2M/IoT devices all over the world; in this scenario, openMTC is used as a middleware solution performing translation from non-oneM2M domains to oneM2M-compliant domains [13] within gateways giving access to separate network segments. openMTC is agnostic of the underlying network, which we assume composed of both terrestrial and space networks. In the proposed framework, M2M/IoT nodes can be either resource-rich or resource-constrained devices. The former have a full TCI/IP protocol stack on-board. In the latter, the data are sent directly over data link or a lightweight network layer. In both cases, generated or collected data by M2M/IoT devices will belong to a specific domain and accordingly formatted [13]. However, in case of resource-constrained devices, an additional gateway (not shown in Figure 2) must de-encapsulate the data and re-encapsulate it within the TCP/IP stack. By using openMTC gateways, data can be sent to or accessed by services and applications after the translation to a oneM2M-compliant data format. The openMTC gateway can be also logically co-located with a Ground Control Station (GCS) used to communicate with space network segments, providing horizontal translation services for vertical application domains and transparent connectivity through different network segments.

Concerning applications, when classifying M2M traffic, 3GPP relies on two classes: time-driven and event-driven. The former is typically related to telemetry(-like) services, where small amounts of data are expected at regular time intervals; the latter is often associated to alarm(-like) services, which generate a small amount of data at less predictable time instants. While telemetry traffic can tolerate small/moderate data loss and delivery delays, the same cannot be said in the case of alarm traffic, whose delivery is subject to stringent time and reliability constraints. Two key concepts emerge: loss and delay tolerance, which can be used to categorize different application classes. Referring to the tolerance to delays, M2M/IoT applications can be categorized into four classes: class 1 - *elastic applications*, class 2 - *hard real-time applications*, class 3 - *delay-adaptive applications*, and class 4 - *rate-adaptive applications*. Such a classification is reported in Table 2, which also provides some examples of M2M/IoT applications for each case, considering loss-sensitive and loss-tolerant scenarios. More specifically, concerning IP-based data exchange solutions, the two most diffused application protocols are CoAP and MQTT, which are briefly described and compared as shown in Table 3.

TABLE 2: Classification of traffic according to delay tolerance and examples of each class. Green stands for moderate tolerance; red for very little tolerance; yellow for little/moderate tolerance; blue for adaptive.

	delay-tolerant	examples of applications and services	
elastic applications (class 1)	🟢	OTA / firmware update	(loss-sensitive)
		caching / content delivery network	(loss-tolerant)
hard real-time applications (class 2)	🔴	industrial automation	(loss-sensitive)
		wide-area monitoring systems	(loss-tolerant)
delay-adaptive applications (class 3)	🔵	transportation and logistics	(loss-sensitive)
		interplanetary communications	(loss-tolerant)
rate-adaptive applications (class 4)	🟡	connectivity / coverage extension	(loss-sensitive)
		environmental monitoring	(loss-tolerant)

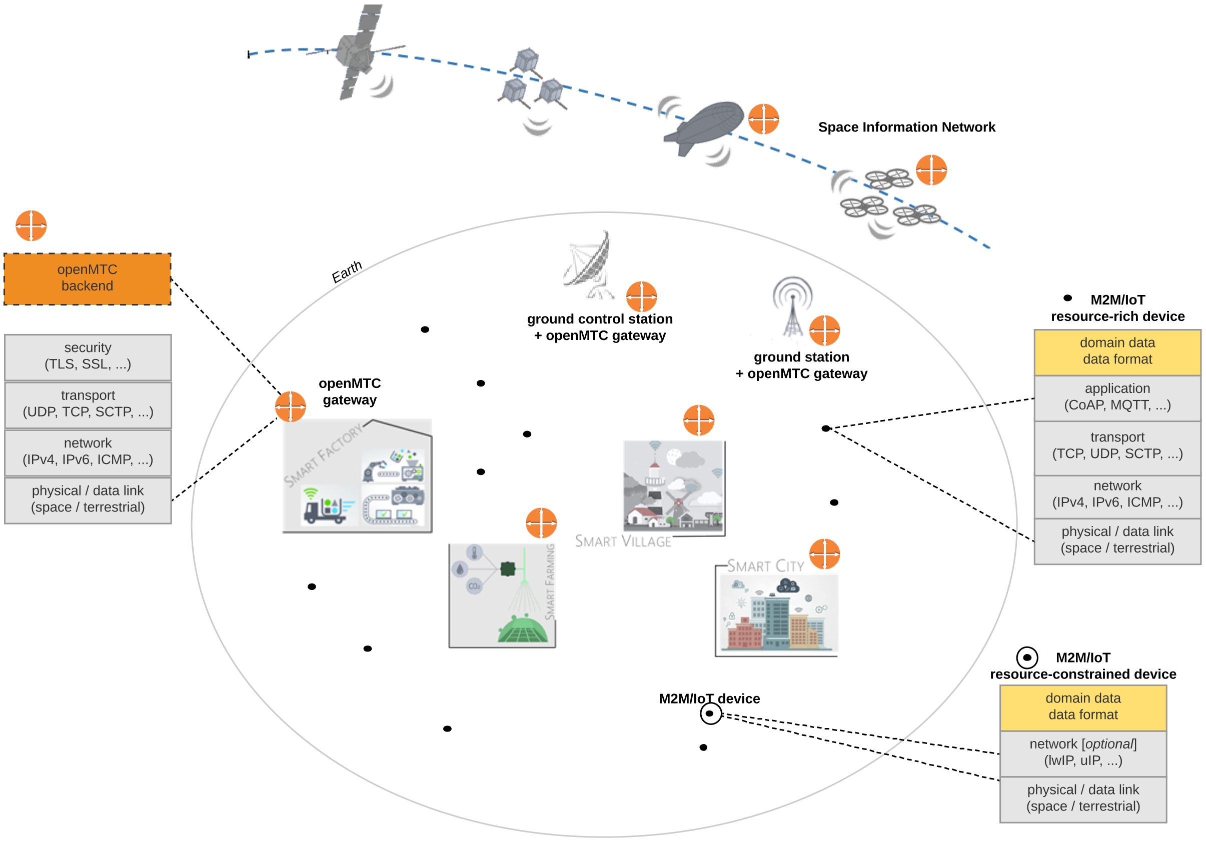

Fig. 2: OpenMTC, a gluing solution for connecting remote M2M/IoT devices by exploiting terrestrial networks and SINs.

3.1 MQTT protocol

MQTT was designed by IBM in 1999 for satellite networks, but largely exploited in terrestrial ones. MQTT implements the Publish / Subscribe (PUB/SUB) paradigm in which data producers (*publishers*), and data consumers (*subscribers*) are decoupled by means of a rendez-vous entity called *broker*. Data are organized into logical flows called *topics*. Each data packet is sent to the broker that maintains the list of the active subscriptions and topics. MQTT provides reliable data exchanges because it is TCP-based. For this reason, the energy efficiency of MQTT is lower than the one provided by CoAP, as detailed below. Default settings imply a protocol overhead of ∼10 bytes. End to end security is achieved through Transport Layer Security (TLS) / Secure Sockets Layer (SSL).

3.2 CoAP protocol

CoAP adheres to a Representational State Transfer (REST) architectural style, providing support to resource-constrained environments. Typical settings imply a lower overhead (∼8 bytes) with respect to MQTT. IPSEC / DTLS can be used for security purposes. Resources are encapsulated by CoAP servers and addressable by Uniform Resource Identifiers (URIs). To enquiry a resource (server), a CoAP client sends either a *confirmable* (i.e., an acknowledgment is expected) or a *non-confirmable* request. Being CoAP UDP-based, reliability is not foreseen by default. However, it outclasses MQTT in terms of energy efficiency, thanks to UDP, the lower overhead, and the larger flexibility. While MQTT exploits TCP capabilities on congestion control and automatic retransmissions, CoAP implementation must take charge of both functionalities. As discussed in detail in [14], PUB/SUB-like data exchanges are possible, by exploiting the observer pattern in RFC 7641 and the proxy functionality in RFC 7252. Implementing both functionalities allows decoupling data producers and data consumers in a transparent way. Because of that, M2M/IoT endpoints are relieved from maintaining local information, thus shifting complexity to the rendez-vous node.

4 PERFORMANCE OF COAP AND MQTT IN SINS

In this section, CoAP and MQTT performance are compared, when put on top of synchronous data link protocols in long delay Random Access CHannels (RACHs) by leveraging on the performance evaluations in [14], [15]. The former takes into account the case of medium/high traffic load, and the latter the case of low traffic. Here, a full overview is provided, by considering *goodput* instead of throughput as performance metric, with additional considerations on the overhead, complexity, and easiness to deploy.

Figure 3 depicts the achievable normalized goodput at the application layer, when relying on the use of CoAP and MQTT. For both protocols, the average goodput value is shown for normalized loads between 0 and 1. Both 0.25 and 0.75 quantiles are plotted around the average values (straight lines), showing that MQTT has larger variations than CoAP around its average values due to the TCP bandwidth probing. To better understand the meaning of the results shown in Figure 3, we discuss separately low, medium, and high load conditions, since different loss rates are experienced at the application layer. Loss rate is negligible with low loads, so opening to very simple strategies providing high energy efficiency and very low complexity for reliable exchanges, while it is not so for higher loads. We consider medium load conditions as the ones compatible with the system working points in the presence of TFRC-like congestion control algorithms [14]; and high load conditions as the ones suffering from severe loss rates. The load in Figure 3 is directly related to the number of devices transmitting at the same time in a RACH: in this case, the devices can be CoAP proxies or MQTT brokers. CoAP can exploit the simple congestion control mechanism natively provided by its plain implementation, in case of low traffic. Its simplicity results in a low computational load on available resources, which is a necessary condition in the case of resource-constrained devices. Increasing NSTART (the CoAP parameter limiting the number of simultaneous outstanding interactions that clients maintain to a given server) makes the goodput increasing almost linearly at low loads [15], due to the very low contention level. In the case of MQTT, TCP congestion control algorithm governs the sending rate, from which the achievable goodput is computed. To summarize, it is possible to identify a value for NSTART able to provide a comparable performance between CoAP and MQTT. In [15], the authors empirically establish this value by setting NSTART=5. In case of medium/high loads, the native use of NSTART is no more sufficient due to the increased contention on the RACH and a performance degradation should be expected as the load increases. Therefore, the use of a modified TFRC congestion control algorithm for CoAP, namely TFRC-s, coupled with a selective-repeat Automatic Repeat reQuest (ARQ) algorithm is suggested in [14]. The proposed implementation impacts only on rendez-vous nodes (i.e., CoAP proxies), leaving unmodified both publishers and producers. The use of a TFRC-based algorithm allows having a performance level comparable or even higher than the one achieved by the use of MQTT because of the lower protocol overhead and of TFRC-s.

Eventually, MQTT provides an easily deployable solution, benefiting of the reliability of a TCP-based stack. A larger diffusion in real uses than CoAP must be mentioned, partially due to its prior appearance in the market, and to the fact it inherently provides a reliable solution. On the other hand, CoAP is gaining attraction if compared to MQTT. It represents a more flexible solution, with optional features providing additional functionalities at the cost of an increasing complexity. In other words, it allows an incremental approach that is desirable in a vast market ecosystem of M2M/IoT devices with different requirements.

5 CONCLUSIONS

In this work, the network infrastructure of SINs has been presented, focusing on the different involved network segments. Such a complex software and hardware architecture can provide connectivity all over the world, actually enabling data exchanges between any couple of M2M/IoT devices. This paper provides examples of application scenarios that can benefit from global coverage, as well as of the still open challenges to be solved. In order to really push forward towards ubiquitous MTC connectivity, open horizontal standards are necessary to provide interoperability, overcoming vertical solutions. In this view, we present a possible horizontal relay solution to support M2M/IoT-based application scenarios. Eventually, we analyze two of the most diffused application protocols, i.e., CoAP and MQTT, comparing the achievable performance level when reliable data exchanges are desired in long-delay networks.

REFERENCES

[1] Q. Yu, J. Wang, and L. Bai, “Architecture and Critical Technologies of Space Information Networks,” *Journal of Communications and Information Networks*, vol. 1, no. 3, pp. 1–9, 2016.

[2] J. Mukherjee and B. Ramamurthy, “Communication Technologies and Architectures for Space Network and Interplanetary Internet,” *IEEE communications surveys & tutorials*, vol. 15, no. 2, pp. 881–897, 2013.

[3] S. Jayousi, S. Morosi, L. S. Ronga, E. D. Re, A. Fanfani, and L. Rossettini, “Flexible Cubesat-based System for Data Broadcasting,” *IEEE Aerospace and Electronic Systems Magazine*, vol. 33, no. 5-6, pp. 56–65, May 2018.

[4] M. Marchese, *QoS Over Heterogeneous Networks*. Wiley Publishing, 2007.

[5] Z. Qu, G. Zhang, H. Cao, and J. Xie, “LEO Satellite Constellation for Internet of Things,” *IEEE Access*, vol. 5, pp. 18 391–18 401, 2017.

[6] A. Al-Fuqaha, A. Khreishah, M. Guizani, A. Rayes, and M. Mohammadi, “Toward Better Horizontal Integration Among IoT Services,” *IEEE Communications Magazine*, vol. 53, no. 9, pp. 72–79, 2015.

[7] M. N. Sweeting, “Modern Small Satellites - Changing the Economics of Space,” *Proceedings of the IEEE*, vol. 106, no. 3, pp. 343–361, 2018.

[8] F. A. d’Oliveira, F. C. L. d. Melo, and T. C. Devezas, “High-Altitude Platforms - Present Situation and Technology Trends,” *Journal of Aerospace Technology and Management*, vol. 8, pp. 249 – 262, 09 2016.

[9] S. Chandrasekharan, K. Gomez, A. Al-Hourani, S. Kandeepan, T. Rasheed, L. Goratti, L. Reynaud, D. Grace, I. Bucaille, T. Wirth *et al.*, “Designing and Implementing Future Aerial Communication Networks,” *IEEE Communications Magazine*, vol. 54, no. 5, pp. 26–34, 2016.

TABLE 3: Comparison of features and implemented communication paradigms of MQTT and CoAP application protocols for M2M/IoT scenarios.

	Brokered	**Broker-less**	**Data Exchanges**		**Protocol Overhead**	**Energy Efficiency**	**Security**
			reliable	unreliable	default settings		
MQTT	✓	✗	✓	✗	10 bytes	low/medium	TLS / SSL
CoAP	✓	✓	✓	✓	8 bytes	medium/high	IPSEC / DTLS

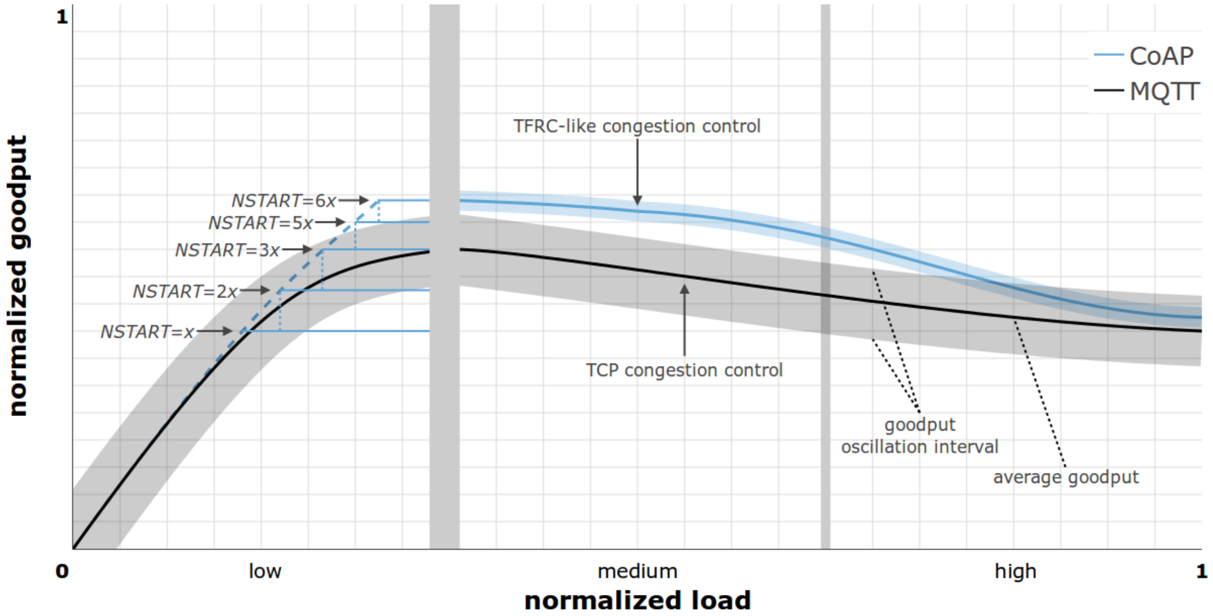

Fig. 3: Aggregated normalized goodput of CoAP and MQTT vs. normalized load.

[10] N. H. Motlagh, T. Taleb, and O. Arouk, "Low-Altitude Unmanned Aerial Vehicles-Based Internet of Things Services: Comprehensive Survey and Future Perspectives," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 899–922, Dec 2016.

[11] J. Zhang, X. Zhang, M. A. Imran, B. Evans, Y. Zhang, and W. Wang, "Energy Efficient Hybrid Satellite Terrestrial 5G Networks with Software Defined Features," *Journal of Communications and Networks*, vol. 19, no. 2, pp. 147–161, 2017.

[12] M. De Sanctis, E. Cianca, G. Araniti, I. Bisio, and R. Prasad, "Satellite Communications Supporting Internet of Remote Things," *IEEE Internet of Things Journal*, vol. 3, no. 1, pp. 113–123, 2016.

[13] V. Gazis, "A Survey of Standards for Machine-to-Machine and the Internet of Things," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 482–511, 2017.

[14] M. Bacco, P. Cassarà, M. Colucci, and A. Gotta, "Modeling Reliable M2M/IoT Traffic over Random Access Satellite Links in Non-saturated Conditions," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 4, 2018.

[15] M. Bacco, M. Colucci, and A. Gotta, "Application Protocols enabling Internet of Remote Things via Random Access Satellite Channels," in *International Conference on Communications (ICC)*. IEEE, 2017, pp. 1–6.