IReHMo: An Efficient IoT-Based Remote Health Monitoring System for Smart Regions

Ngo Manh Khoi, Saguna Saguna, Karan Mitra and Christer Åhlund
Department of Computer Science, Electrical and Space Engineering
Luleå University of Technology
Skellefteå, Sweden
Email: manngo-4@student.ltu.se,{saguna.saguna, karan.mitra, christer.ahlund}@ltu.se

***Abstract*—The ageing population worldwide is constantly rising, both in urban and regional areas. There is a need for IoT-based remote health monitoring systems that take care of the health of elderly people without compromising their convenience and preference of staying at home. However, such systems may generate large amounts of data. The key research challenge addressed in this paper is to efficiently transmit healthcare data within the limit of the existing network infrastructure, especially in remote areas. In this paper, we identified the key network requirements of a typical remote health monitoring system in terms of real-time event update, bandwidth requirements and data generation. Furthermore, we studied the network communication protocols such as CoAP, MQTT and HTTP to understand the needs of such a system, in particular the bandwidth requirements and the volume of generated data. Subsequently, we have proposed IReHMo - an IoT-based remote health monitoring architecture that efficiently delivers healthcare data to the servers. The CoAP-based IReHMo implementation helps to reduce up to 90% volume of generated data for a single sensor event and up to 56% required bandwidth for a healthcare scenario. Finally, we conducted a scalability analysis to determine the feasibility of deploying IReHMo in large numbers in regions of north Sweden.**

***Keywords*—remote health monitoring, Internet of Things, sensor communication, eHealth, smart city, smart region.**

I. Introduction

The world aging population comprises an important part of the world society. Due to the worldwide improvements in society, economy and healthcare in the last decades, the average life expectancy has increased substantially while the mortality rate decreased. As a direct consequence, the number of elderly people worldwide has risen constantly. Today, the average percentage of elderly people (a person who is 65 years old or more) worldwide of 7% [1]. Moreover, in many countries the percentage of people over 65 years old exceeds the world average, such as 18% in Sweden, 18.5% in Finland and 15% average for countries in Organisation for Economic Co-operation and Development (OECD) group. Furthermore, this percentage is likely to rise in the future. It is predicted that by 2050, 24% of the Swedish population will be elderly people, among them 10% will be 80 or over [2]. This situation poses challenges to the government as well as local municipalities, whose responsibilities are to maintain the health of elderly people and enhance their quality of life.

Among the health care facilities for elderly people such as home care, hospitals, health centers, home for elderly people and service house for elderly, home care are preferred by the majority of the elderly people [3]. Furthermore, in some countries, the government makes the target of increasing the possibilities that elderly people can stay in their home and receive same care as they go to care facilities dedicated for them [4].

Considering an elderly person with Alzheimer staying in his/her house, it is very beneficial to deploy a remote health monitoring system there. The system will allow the caregiver to know if a person is in the room or opens a door, and sends alarm if the stove is on for too long or a person walks out in the middle of night. The healthcare service provider can deploy three scenarios that can help monitoring the patient remotely. In scenario 1, the system offers basic monitoring capability by using an emergency button with voice recorder, door sensor, motion sensor and fire alarm. In scenario 2 and 3, the system offers enhanced monitoring capability with video stream by using a set of sensors including an emergency button with voice recorder, motion sensor, IP camera and different numbers of wall plugs. However, this system alone can generate large amounts of data, especially when a large number of sensors are included.

An efficient remote health monitoring system is needed as it offers healthcare providers the ability to constantly monitor the behaviours and wellbeing of the elderly people. At the same time, the system gives them the convenience and peace of living in their own house, knowing that they will get assistance immediately when they need. The expected system should perform tasks such as detecting and preventing accidents and transmitting body parameters to the processing place. Body parameters range from non-time-critical information such as periodic check of heart rate, body temperature, blood pressure, blood glucose level to time-critical information such as ECG signal. This paper seeks to study the network communication needs of IoT devices in the context of remote health monitoring for smart regions. The major contributions of our work are:

- Studying the network communication requirements of an IoT-based remote health monitoring application.
- Proposing an overall remote health monitoring architecture to evaluate and compare several network communication protocols.
- Realizing the architecture into a prototype that can reduce up to 90% volume of generated data for a single sensor event and up to 56 % required bandwidth for a healthcare scenario compared with an existing commercial product.

The paper is organized as follows: In Section 2, IoT architecture and network communication are reviewed; in Section 3, the components of the existing commercial product and the architecture of the proposed IReHMo are described; in Section 4, the results and comparisons of the performance of different protocols are analyzed; in Section 5 the conclusion and future work are presented.

II. Background and Related Work

A. IoT architecture

The last decade has witnessed the developments of many IoT-based healthcare applications [5], [6], [7]. Researchers have worked on a reference model for IoT implementations. For example, Jin et al. [8] proposed a generic framework for creating IoT implementations, including smart healthcare. According to this article, there are three main viewpoints that guide the building of an IoT implementation: network-centric IoT, cloud-centric IoT and data-centric IoT. Each viewpoint has several building blocks, and each building block in a viewpoint corresponds to one or more blocks in the other viewpoints. Therefore, the three viewpoints have strong relation with each other. All these three viewpoints have influenced the implementation of different IoT applications.

Furthermore, this reference model is reflected in other literature works, especially in the field of IoT-based healthcare. Fengou et al. [9] proposed an architecture of the e-Health telemonitoring system, which has several components performing data collection, data management and data interpretation. It represents the data-centric IoT viewpoint that is discussed in [8]. In [10], the data-centric IoT viewpoint is highlighted as the authors explicitly described the data flow from sensors to intermediate gateways and hubs and eventually to cloud-based data stores. In [11], the authors described an overall system design of e-Health applications, with focus on the interaction between several components of the system such as Body Sensor Network (BSN), Zigbee, smart house and medical call center. In [12], an open IoT platform was proposed. The platform is designed as a self-management model for chronic diseases, but the architecture can be extended to have remote health monitoring capabilities.

B. IoT communication

In the application layer, there are many network communication protocols for IoT devices such as HTTP [13] or novel protocols dedicated to the field of IoT such as MQTT [14] or CoAP [15]. The comparison between these protocols have been discussed in recent literature work. In [16], the authors suggested that MQTT is a better candidate for applications requiring sophisticated functionalities such as different level of QoS and message persistence and multicast. However, quantitative analysis indicates that CoAP performs better than MQTT in terms of bandwidth requirement and round trip time (RTT). For reliability, it depends on the actual scenario; MQTT achieves better results in scenarios where message exchange happens very frequently, otherwise the difference is not much. The ability of CoAP to fragment large messages makes it more efficient when transmitting large messages. In [17], CoAP has been compared against HTTP in terms of energy consumption and volume of generated data. CoAP energy consumption is 50% and 83% less than HTTP in push mode and pull mode, respectively. In term of volume of generated data, the CoAP-based system generated data which is 85% less than the HTTP-based system (62 GB vs 434 GB). In [18], the author gave a qualitative analysis on the competing IoT protocols (HTTP, MQTT, CoAP, AMQP). The criteria used were architecture, security mechanism, QoS schemes and communication pattern (inquiries, telemetry, commands and notification). The authors concluded that each and every protocol has its own strengths and drawbacks. It depends on the scenario and the requirements to choose the most appropriate protocol. Furthermore, it is possible for a complex system to incorporate several protocols together.

III. IReHMo- IoT-based Remote Health Monitoring System

A. Architecture

In order to realize a system which combines several IoT protocols in the lower layer and efficiently transmits data to the remote servers, the paper proposed an architecture called IoT-based Remote Health Monitoring (IReHMo). The overall architecture of IReHMo consists of five layers, namely sensing layer, home gateway, network infrastructure, cloud computing and application layer, as described in figure 1.

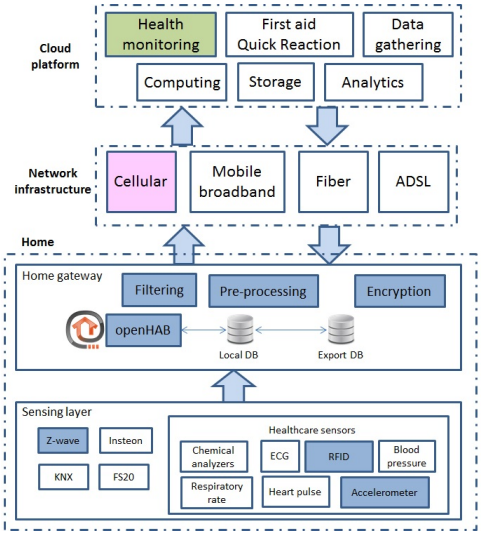

Fig. 1: The proposed IReHMo architecture.

The sensing layer comprises of home automation devices, such as sensors (temperature, humidity, smoke and CO, water leak) and actuators (power switch, lock, dimmer). For the purpose of remote health monitoring, IoT sensors such as RFID sensors, accelerometers can be included in this layer. Today, there is a long list of biosensors or healthcare sensors that measure body temperature, blood pressure, heart pulse, ECG, respiratory rate, glucose level. Another source of healthcare information may come from smartphones which act as the gateway of the wireless body area network. It is possible to collect all these health-related information in this sensing layer and forward it to higher layers for further processing.

In the home gateway layer, the focus is on the data collection, filtering, pre-processing and encryption, which are highlighted in blue in figure 1. Sensor data are collected and

stored locally as well as transmitted to the remote servers. Using openHAB - a middleware for integrating several home automation solutions, the home gateway can interact with several types of IoT protocols such as Z-wave, KNX and Insteon. Other medical sensors stream their data to local databases. In the home gateway, activities such as filtering or pre-processing can take place, to select and improve the data for the next stage of data transmission. These activities can help reducing the bandwidth required or the volume of data to be transmitted. Different databases are presenting in this layer. Local database is in charged of storing raw data, while export database is for storing processed data, ready to be transmitted. Encryption is carried out in this layer, as health data is sensitive and need to be protected. At the gateway layer, the selection of IoT application layer protocol is crucial as it decides the performance of the system, which is discussed in the next section.

Data from the home gateway is transmitted to the monitoring side where it is consumed using the **network infrastructure**. To date, different networking technologies are available, namely cellular (3G, 4G), mobile broadband, fiber and ADSL. The transmission can include WiFi and Ethernet technologies. Depending on the required bandwidth of the implementation, certain technologies are preferred to ensure the best performance and efficiency.

The cloud is the receiving end of the data flow from the sensors. Further processing is done in cloud computing facilities. Here the sensor data is transformed into meaningful knowledge and actions, using algorithms and dedicated softwares. In this layer, the main activities are further processing of data, data storage and analytics. Efficient and elastic data storage can be achieved by cloud services such as Amazon S3 or Microsoft Azure.

The top-most layer of the stack is the application layer, where applications interact with users through web interfaces. Since sensor data is collected, stored and processed continuously, users can get a holistic picture of the situation and deliver actions accordingly. Typical applications based on health data collected by sensors can be remote health monitoring, quick reaction to emergency situations or health data gathering and statistics.

B. Prototype implementation

An actual implementation of IReHMo has been carried out. In the sensing layer, the implementation includes Z-wave sensors for monitoring parameters such as room temperature, the presence of a person, the state of the door/windows, the presence of smoke. Healthcare sensors are RFID readers and embedded accelerometers from iPhone for the purpose of activity recognition. Encryption is carried out at the home gateway, using AES 128-bit encryption algorithm as shown in figure 2. This encryption method is selected due to its simplicity in implementation. Furthermore, the monetary per-bit cost of AES is several orders of magnitude lower than that of other encryption methods (RSA, DSA and ECDSA)[19]. At the patient's house, the healthcare data is encrypted into a string, which is then transmitted over the public network. As shown in figure 2, at the monitoring side, the decryption of the received string takes place; as a result the original healthcare data is obtained. This security feature comes at a small additional price (more CPU cycles at both home gateway and monitoring side), however AES decryption at cloud facilities is much more efficient and cost-effective than other smaller facilities (2.37E+01 picocent vs 1.42E+03 picocent) [19].

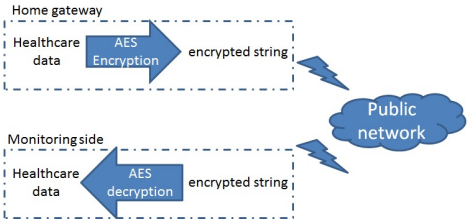

Fig. 2: The encryption process.

In the home gateway layer, the sensed data are stored locally and secured by user authentication as well as transmitted to the monitoring side, which acts as the receiving end of the data flow. HTTP and CoAP were used to transport sensor data to the monitoring side. The amount of generated data depends on the monitoring policy and the health situation of the patient. Although the actuation of home automation devices is not in the scope of the paper, it can be done using the REST API of openHAB. The command can be sent from the monitoring side to the openHAB middleware using POST command of HTTP and CoAP. The details of the implementation are depicted in the following figure 3.

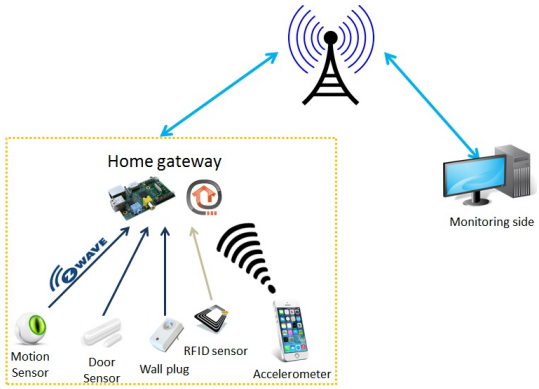

Fig. 3: The actual IReHMo implementation.

IV. Results and Analysis

A. Results

The IReHMo architecture allows to deploy several application layer protocols at the home gateway layer. In this paper, CoAP and HTTP were deployed, evaluated and compared.

1) CoAP-based IReHMo implementation: The CoAP-based IReHMo implementation can deliver sensor data in three different modes using CoAP GET, PUT and Observe method. Furthermore, CoAP has two QoS mechanisms, namely Confirmable (CON) and Non-confirmable (NON). In Confirmable mode, each packet sent is acknowledged by an ACK packet from the receiver. If the receiver has to reply to a request, the response is piggybacked in that ACK packet. In case the sender doesn't receive the ACK message, the sender resends

the packet after a backoff time. In contrast, the receiver will not acknowledge the packet transmission in Non-confirmable mode. The response for a NON request is usually a NON reply.

Considering a healthcare value, for example the room temperature of 19.9°C. After encryption, the value is turned into a hexadecimal string of length 32 bytes. This encrypted string will be delivered by CoAP, using its methods of GET, PUT or Observe. The following table I summarizes the traffic generated and bitrate when the encrypted string is delivered using CoAP CON and NON mode.

TABLE I: Volume of generated data and bitrate required by CoAP-based IReHMo implementation in CON and NON mode.

CoAP method	Bytes UL/DL	Bytes (Total)	Bitrate UL/DL (kbps)
GET (CON & NON)	82 / 60	142	0.656 / 0.48
PUT (CON & NON)	80 / 60	140	0.64 / 0.48
Observe (CON)	85 / 60	145	0.68 / 0.48
Observe (NON)	85 / 0	85	0.68 / 0

Using CoAP, the overhead for any small healthcare payload is from 48 to 53 bytes in each packet, depending on the method.

2) HTTP-based IReHMo implementation: The HTTP-based IReHMo implementation can deliver sensor data using HTTP GET method. To get a sensor value, the monitoring side has to establish an HTTP session with the home gateway, therefore HTTP packets can be exchanged. From the measurements, for the whole session (which lasted for 30 seconds) it took 845 bytes in the downlink and 494 bytes in the uplink to get a sensor reading (voltage level) from the wall plug. In that session, the following events occurred: 3-way handshake, HTTP GET, HTTP/1.1 200 OK, TCP Keep-Alive and session closing. While the 3-way handshake, HTTP GET and HTTP/1.1 200 OK happened immediately after each other, the TCP Keep-Alive and session closing happened later in the session. We took the two events of HTTP GET and HTTP/1.1 200 OK for comparison with the CoAP-based IReHMo implementation. The following table II summarizes the traffic generated and bitrate when a small sensor data (voltage level) is delivered using HTTP get.

TABLE II: HTTP GET method.

Event	Bytes UL/DL	Bytes (Total)	Bitrate UL/DL (kbps)
Request / Response	188 / 359	547	1.504 / 2.872
Whole session	494 / 845	1339	Not available

3) The commercial product: The commercial product, having a similar implementation contains, contains a door sensor, a motion sensor, smoke sensor, IP camera and emergency button. MQTT was used to deliver sensor data from the home gateway to the remote servers. There are two traffic patterns produced by this product: the periodic traffic pattern and the traffic pattern from sensor data. The periodic traffic pattern is the packet exchange between the gateway and the server (the frequency depends on the pattern), to maintain the connection. The following table III summarizes the periodic traffic pattern.

For each sensor event, the gateway sends several packets to the remote server. Each outgoing packet is followed by acknowledgement packets from the remote server and the gateway, due to TCP acknowledgement mechanism. The following table IV summarizes the traffic pattern from sensor data.

TABLE III: Periodic traffic pattern.

Pattern	Bytes Uplink/Downlink	Bitrate UL/DL (kbps)
Pattern 1	206 / 140	1.648 / 1.12
Pattern 2	1036 / 66	8.288 / 0.528
Pattern 3	204 / 102	1.632 / 0.816
Pattern 4	420 / 564	3.36 / 4.512
Pattern 5	506 / 140	4.048 / 1.12
Pattern 6	66 / 66	0.528 / 0.528

TABLE IV: Traffic pattern from sensor data.

Event	Bytes UL/DL	Bytes (Total)	Bitrate UL/DL (kbps)
Door closing	644 / 420	1064	5.152 / 3.36
Door opening	556 / 280	836	4.448 / 2.24
Door sensor tampering	334 / 140	474	2.672 / 1.12
Wall plug ON/OFF	636 / 280	916	5.088 / 2.24
Emergency button pressed	938 / 420	1358	7.504 / 3.36
Smoke alarm ON	890 / 420	1310	7.12 / 3.36
Smoke alarm OFF	1084 / 700	1784	8.672 / 5.6
High temperature	318 / 140	458	2.544 / 1.12
Smoke sensor tampered	668 / 280	948	5.344 / 2.24
Motion sensor ON	842 / 420	1262	6.736 / 3.36
Motion sensor OFF	1084 / 700	1784	8.672 / 5.6
Strong vibration	668 / 280	948	5.344 / 2.24

The bitrate for the voice connection is 32 kbps for each uplink and downlink channel, while the bitrate for video stream is 167.46 kbps for the uplink and 12.328 kbps for the downlink channel.

B. Discussion

In a typical remote health monitoring scenario, a sudden event will trigger sensors to sense and produce data. This data is collected by the home gateway and forwarded to the remote servers for further processing and knowledge extraction. Due to its characteristics, the remote health monitoring has to reliably deliver sensor data to the monitoring side. To achieve this goal, we have to calculate the required bandwidth of the system in different conditions. In this paper, we evaluate scenario 1, 2 and 3 as described in Section 1. Scenario 1 contains an emergency button, a door sensor, motion sensor and fire alarm. Scenario 2 contains an emergency button, a motion sensor, IP camera and one wall plug. Scenario 3 contains an emergency button, a motion sensor, IP camera and several wall plugs. To ensure that the system performs in all cases, the worst situation is considered when all sensor events happened at the same time. The bandwidth in this worst-case scenario is the sum of all bitrates from the sensors included in that event (for example door opened, motion sensor on, smoke alarm on, emergency button pressed) plus the bitrate of the periodic traffic patterns. The following figure 4 details the required bandwidth for scenario 1, 2 and 3 for the commercial product and the CoAP-based IReHMo implementation.

For each single sensor event, the HTTP-based IReHMo implementation reduced the required bandwidth in the uplink, compared with the commercial product. However, the volume of generated data remains the same, especially when sensor events happen intermittently. In this case, HTTP sessions are established and closed consecutively, which produce unnecessary traffic. From the results, it is clear that CoAP delivers sensor data very efficiently due to its small protocol overhead.

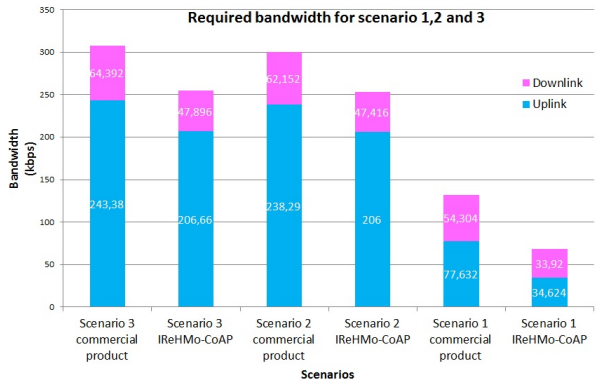

Fig. 4: The required bandwidth for different scenarios.

For a single sensor data, CoAP reduces the volume of generated data up to 90 % in the uplink channel (85 bytes vs 842 bytes), using several methods such as GET, PUT and Observe as shown in figure 5. This efficiency allows the implementation to reduce significantly the required bandwidth, especially in the uplink, and the volume of generated data.

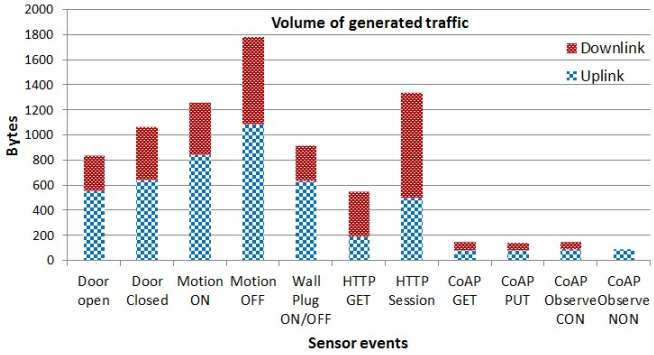

Fig. 5: Comparison between volume of generated data from various sensor events.

Furthermore, it is advantageous to deploy the CoAP-based IReHMo implementation compared with the commercial product since the communication pattern from IReHMo can be either telemetry (CoAP Observe method) or request-response (CoAP GET and PUT methods). If real-time update is the priority then the telemetry pattern is preferred. Otherwise, the monitoring side can request the home gateway for a particular value of sensor on demand, thus reducing bandwidth and volume of generated data both in the uplink and downlink. In contrast, the commercial product offers only the telemetry communication pattern, which sends data even when it is not needed. When all the sensor events are combined in a scenario, the CoAP-based IMeHRo implementation helps reducing the total required bandwidth up to 56% in the uplink channel.

From the tables, it is clear that the video stream consumes most of the bandwidth, compared with sensor data. Instead of using the video stream, we can deploy several types of sensors that can gather more information and provide them to the monitoring side. The second biggest bandwidth consumer is the voice connection. If we can offload voice traffic from the remote health monitoring system to the GSM network (which has the same coverage) then a large portion of the uplink bandwidth can be used for healthcare data.

C. Scalability analysis

As suggested by the IReHMo high level architecture, the healthcare data can be delivered using various network technologies such as cellular, mobile broadband, fiber and ADSL. Among them, the cellular network is used by the vast majority of the population. However, the network offers limited capacity and it can change very dynamically. As a result, this scalability analysis studies the feasibility of deploying remote health monitoring systems in remote rural areas, where the cellular network infrastructure offers limited capacity. Moreover, the importance of these systems are best shown here where people have limited access to regular healthcare service. The analysis needs the information on cellular network quality and the number of users for these systems.

The information on cellular network quality is retrieved from the coverage map of a service provider in Sweden. Based on this, we have selected several villages/residential areas that have moderate to low quality cellular network. To estimate the available bandwidth, we set up a testbed to measure the average download/upload speed to our selected server. In order to verify the available bandwidth, we cross-check our obtained results with a well-established speedtest app. The comparisons showed that both results converged with small discrepancies. For a single user, the upload bandwidth is in the range of 1.35-3.61 Mbps, while the range for download bandwidth is 8.15-15.364 Mbps, depending on the measurement locations. For the DC-HSPA+ technology that are being used in these locations, the upper bound of cell capacity in the uplink channel is 6.2 Mbps [20].

The selected communities have diverse population, ranging from several dozens to several hundreds. Using the statistics from the regional government, it is safe to say that at least 25% of the population are at least 65 years old. We assume that all these elderly people are using a remote health monitoring system, with a mix of healthcare scenarios. The following table V shows the required bandwidth from selected villages/residential areas, with different combination of scenario 1, 2 and 3 (all scenario 3, 30% scenario 3 - 40% scenario 2 - 30% scenario 1, 10% scenario 3 - 20% scenario 2 - 70% scenario 1 and all scenario 1) when using the commercial product.

TABLE V: Scalability analysis of the commercial product.

Name	Elderly / Total	Scenario 3 (Mbps)	30/40/30 (Mbps)	10/20/70 (Mbps)	Scenario 1 (Mbps)	Available bandwidth
Gumboda	13/55	3.34	2.63	1.73	1.06	6.2 Mbps
Renström	17/67	4.07	3.20	2.11	1.30	6.2 Mbps
Långträsk	27/109	6.63	5.22	3.44	2.11	6.2 Mbps
Moskosel	58/232	14.11	11.11	7.32	4.50	6.2 Mbps
Bastuträsk	98/392	23.85	18.77	12.38	7.60	6.2 Mbps
Backe	150/599	36.44	28.69	18.91	11.62	6.2 Mbps
Jörn	199/797	48.49	38.18	25.17	15.46	6.2 Mbps
Boliden	391/1566	95.28	75.01	49.46	30.39	6.2 Mbps

Table VI shows the required bandwidth from selected villages/residential areas, with different combination of scenario 1, 2 and 3 when using the CoAP-based IReHMo implementation.

TABLE VI: Reduced bandwidth requirement by IReHMo and CoAP.

Name	Elderly / Size	Scenario 3 (Mbps)	30/40/30 (Mbps)	10/20/70 (Mbps)	Scenario 1 (Mbps)	Available bandwidth
Gumboda	13/55	2.84	2.12	1.18	0.47	6.2 Mbps
Renström	17/67	3.46	2.59	1.44	0.57	6.2 Mbps
Långträsk	27/109	5.63	4.21	2.34	0.94	6.2 Mbps
Moskosel	58/232	11.98	8.97	4.99	2.00	6.2 Mbps
Bastuträsk	98/392	20.25	15.16	8.43	3.39	6.2 Mbps
Backe	150/599	30.94	23.17	12.89	5.18	6.2 Mbps
Jörn	199/797	41.17	30.84	17.15	6.89	6.2 Mbps
Boliden	391/1566	80.90	60.59	33.70	13.55	6.2 Mbps

It is evident that the bandwidth reduction from CoAP helps to deploy more remote health monitoring systems into the existing cellular network in rural areas. For example, in most of the selected villages/residential areas, scenario 1 can be deployed to all elderly people in that area.

V. Conclusion and future work

The paper has identified several network-related requirements of a remote health monitoring system, such as low bandwidth consumption, especially upload bandwidth so that it can fit in and scale up in areas where network infrastructure is limited and reduce volume of generated data so that it will not stress the existing network infrastructure as well as induce unnecessary costs to users. The paper proposed and evaluated an architecture called IReHMo. IReHMo is capable of incorporating several types of home automation sensors and healthcare IoT devices in the sensing layer. An IReHMo implementation using CoAP significantly reduced the bandwidth requirements and volume of generated data. For each small size healthcare data, IReHMo significantly reduced the number of packets being sent, the required bandwidth and the volume of generated data compared to the commercial product. This will translate to a large saving in bandwidth, volume of generated data and round trip time; the system reduces up to 56% of the required bandwidth for a remote health monitoring scenario. Finally, the scalability analysis showed that the combination of IReHMo and CoAP make it possible to deploy larger number of remote health monitoring systems compared to the existing commercial product. This is of paramount importance in remote rural areas where the network capacity is low and local people have limited access to regular healthcare services. In the future, IoT based healthcare architectures such as IReHMo will be used to recognize complex activities [21] of patients to enable improved remote healthcare services. Further, such complex activity recognition will generate large-scale data that needs to be processed using cloud services available closest to the end-user [22] to minimize end-to-end latency and ensure timely response from healthcare services.

Acknowledgment

The authors would like to thank Erasmus Mundus PERCCOM program(Pervasive Computing and Communications for Sustainable Development) and Erasmus+ program of European Commission for providing funding for this research work. Also we would like to thank Nina from Västerbotten regional government for our interesting and insightful discussion.

References

[1] Population Reference Bureau, "2014 World Population Data Sheet". [Online]. Available: http://www.prb.org/Publications/Datasheets/2014/2014-world-population-data-sheet.aspx

[2] OECD/European Commission "Sweden - A good life in old age? Monitoring and improving quality in long-term care", June 2013. OECD. [Online]. Available: http://www.oecd.org/els/health-systems/Sweden-OECD-EC-Good-Time-in-Old-Age.pdf

[3] T. Bengtsson, *Population Ageing-a Threat to the Welfare State?: The Case of Sweden*, 2010th ed. Springer Science & Business Media, 2010.

[4] P. Böckerman, E. Johansson, and S. Saarni, "Institutionalisation and quality of life for elderly people in Finland"," *ENEPRI Research Report No. 92*, August 2011.

[5] V. M. Rohokale, N. R. Prasad, and R. Prasad, "A cooperative internet of things (IoT) for rural healthcare monitoring and control," in *Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE), Second International Conference on*. IEEE, 2011, pp. 1–6.

[6] C. Doukas and I. Maglogiannis, "Bringing IoT and cloud computing towards pervasive healthcare," in *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), Sixth International Conference on*. IEEE, 2012, pp. 922–926.

[7] S. Amendola, R. Lodato, S. Manzari, C. Occhiuzzi, and G. Marrocco, "RFID technology for IoT-based personal healthcare in smart spaces," *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 144–152, 2014.

[8] J. Jin, J. Gubbi, S. Marusic, and M. Palaniswami, "An information framework for creating a smart city through internet of things," *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 112–121, 2014.

[9] M. Fengou, G. Mantas, D. Lymberopoulos, N. Komninos, S. Fengos, and N. Lazarou, "A new framework architecture for next generation e-health services," *Biomedical and Health Informatics, IEEE Journal of*, vol. 17, no. 1, pp. 9–18, 2013.

[10] D. Lake, R. Milito, M. Morrow, and R. Vargheese, "Internet of things: Architectural framework for ehealth security," *Journal of ICT Standardization, River Publishing*, vol. 1, 2014.

[11] A. Chehri, H. Mouftah, and G. Jeon, "A smart network architecture for e-health applications," in *Intelligent Interactive Multimedia Systems and Services*. Springer Berlin Heidelberg, 2010, pp. 157–166.

[12] B. M. Lee, "Design requirements for IoT healthcare model using an open IoT platform," *Computer*, vol. 4, p. 5, 2014.

[13] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee, "Hypertext transfer protocol–http/1.1," 1999.

[14] A. N. Andy Stanford-Clark, *"MQTT Version 3.1.1"*, OASIS Std., October 2014. [Online]. Available: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html

[15] Z. Shelby, K. Hartke, and C. Bormann, "The constrained application protocol (CoAP)," 2014.

[16] N. De Caro, W. Colitti, K. Steenhaut, G. Mangino, and G. Reali, "Comparison of two lightweight protocols for smartphone-based sensing," in *Communications and Vehicular Technology in the Benelux (SCVT), Twentieth Symposium on*. IEEE, 2013, pp. 1–6.

[17] T. Levä, O. Mazhelis, and H. Suomi, "Comparing the cost-efficiency of CoAP and HTTP in web of things applications," *Decision Support Systems*, vol. 63, pp. 23–38, 2014.

[18] P. Patierno. (2014, June) "IoT Protocols Landscape". [Online]. Available: http://www.slideshare.net/paolopat/io-t-protocols-landscape

[19] Y. Chen and R. Sion, "Costs and security in clouds," in *Secure Cloud Computing*. Springer, 2014, pp. 31–56.

[20] M. Valtonen. (2010, March) "The bitrate limits of HSPA+ enhanced uplink". [Online]. Available: http://omnitele-com.s3.frantic.com/2011/05/the_bitrate_limits_of_hspa_enhanced_uplink.pdf

[21] S. Saguna, A. Zaslavsky, and D. Chakraborty, "Complex activity recognition using context-driven activity theory and activity signatures," *ACM Transactions on Computer-Human Interaction (TOCHI)*, vol. 20, no. 6, p. 32, 2013.

[22] K. Mitra, S. Saguna, C. Åhlund, and D. Granlund, "M2C2: a mobility management system for mobile cloud computing," in *2015 IEEE Wireless Communications and Networking Conference (WCNC): - Track 3: Mobile and Wireless Networks (IEEE WCNC 2015 - Track 3- Mobile and Wireless Networks)*, New Orleans, USA, Mar. 2015, pp. 1626–1631.