

Article

Partial Bicasting with Buffering for Proxy Mobile IPv6 Mobility Management in CoAP-Based IoT Networks

Moneeb Gohar ^{1,*}, Sajid Anwar ¹, Moazam Ali ¹, Jin-Ghoo Choi ²,
Hani Alquhayz ³ and Seok-Joo Koh ^{4,*}

¹ Department of Computer Science, Bahria University, Islamabad 44000, Pakistan; sajidanwar699@gmail.com (S.A.); moazambui@gmail.com (M.A.)

² Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, Korea; jchoi@yu.ac.kr

³ Department of Computer Science and Information, College of Science in Zulfi, Majmaah University, Al Majma'ah 11932, Saudi Arabia; h.alquhayz@mu.edu.sa

⁴ School of Computer Science and Engineering, Kyungpook National University, Gyeongsan 41566, Korea

* Correspondence: moneebgohar@gmail.com (M.G.); sjkoh@knu.ac.kr (S.-J.K.)

Received: 26 February 2020; Accepted: 30 March 2020; Published: 31 March 2020

Abstract: Constrained application protocol (CoAP) can be used for message delivery in wireless sensor networks. Although CoAP-based proxy mobile internet protocol (PMIP) was proposed for mobility management, it resulted in handover delay and packet loss. Therefore, an enhanced PMIP version 6, with partial bicasting in CoAP-based internet of things (IoT) networks, is proposed. Here, when an IoT device moved into a new network, the corresponding mobile access gateway (MAG) updated the local mobility anchor (LMA) binding. Further, LMA initiated the “partial” bicasting of data packets to the new and the previous MAGs. The data packets were buffered at the new MAG during handover and were forwarded to Mobile Node (MN) after the handover operations. The proposed scheme was compared with the existing scheme, using ns-3 simulations. We demonstrated that the proposed scheme reduced handover delays, packet losses, end-to-end delay, throughput, and energy consumption, compared to the existing scheme.

Keywords: proxy MIPv6; handover; bicasting; buffering; IoT; CoAP

1. Introduction

Wireless networks are the principal mechanisms for establishing computer networks through wireless connections among the nodes of a network [1]. Therefore, wireless networking eliminates the costs of cables. Furthermore, wireless networking is implemented in the physical layer of open systems interconnection (OSI) model networks [2]. Source to destination communications are possible with OSI models. The data are divided into data packets and are distributed by the internet protocol (IP), based on the packet header IP addresses [3]. IP version 4 (IPv4) is the most commonly used IP adaptation. Moreover, IP version 6 (IPv6) is being rendered compatible for these applications. IPv6 supports longer addresses, thereby connecting greater number of internet users. Furthermore, IPv6 integrates IPv4. Therefore, a specific IP is required for mobile devices. The permanent IP addresses must be maintained when moving between different networks [4]. Since mobile IPs (MIPs) are host-based, every movement results in delay, data loss, and signal overload. Therefore, dynamic IPs have been introduced in mobile IP technology. Functionality of this IP is updated by the system responsible for tracking the host's developments and launching the required versatile named tag [5].

The demand for high quality mobile computing service, known as "always and everywhere", will increase in the future. Additionally, the expectation and demand for different types of novel applications and several specific Quality of Services (QoS) mobile computing environments will likely increase. The rapid increase in demand for high-speed "anytime, anywhere" internet-access has become a concern for network operators [6]. Generally, the tendency of central networks is to accommodate the requirements of all mobile IP networks. IP mobile networks, which transmit media (telecommunications) and use internet, organize networks emphatically. Networks, wherein IP operates from a mobile user to access point (AP), link wireless systems to the internet. Mobility management is a principal challenge for next-generation networks [7]. Communication between different devices, through the internet is known as internet of things (IoT) [8]. In simple terms, we discuss the mechanisms of a machine sending and receiving data. Due to advancements in the IoT, the number of communications devices is steadily increasing [9] with the usage and number of devices becoming greater than the world's population. The limit might increase, as human existence on other planets is not confirmed. Therefore, by 2020, as the use of devices increases with people, devices connecting the Web universe will also increase. This has not existed before. Therefore, by 2020, the IoT will likely exceed 50 billion linked gadgets [9].

In 2018, the advent of high-end communication technology and user-friendly devices saw the development of wireless body networks (WBANs) and a dedicated human body network that monitored, directed, and communicated various vital functions, including blood pressure, temperature, and electrocardiogram (ECG), etc. Several sensors were connected to a patient's body and clothing so as to monitor their vital functions. WBANs have a huge range of novel applications, including computer-assisted rehabilitation, an emergency medical response system (EMRS), ubiquitous health monitoring (UHM), and healthy life-style promotion [10]. In general, WBAN in UHM helps in reducing hospital visits. As regular hospital visits are difficult for the general public, the said automation reduces dependency on the specialized health sector workforce. Therefore, this system is recommended in countries with inadequate medical infrastructure and medical personnel, enabling a quicker establishment of a cost-effective health care system. The WBAN is a communication network that integrates human system and computational interface, through portable devices. The common sensor node in a WBAN ensures in identifying correct signals, capturing weak sensor signals, and wirelessly processing these signals at a local processing unit. A special protocol, called a constrained application protocol (CoAP) [9], has been introduced to remotely control WBAN. CoAP is a limited application protocol that transports data in packets from the client to the server. Furthermore, low-weight CoAP devices can be used in small-sized devices with a lower processing capacity and memory. CoAP devices use the user datagram protocol (UDP), which is light, compared to other protocols that support the simultaneous forwarding of messages to different recipients. Among mobile nodes (MNs), mobile access gateway (MAG) and local mobility anchor (LMA), MN moves among networks to facilitate the smooth running of sessions, reduce packet loss, and avoid handover delays in mobility management significance. After entering a new sensor node network domain, from MAG_A to MAG_B , MN changes the point of attachments. MAG_B senses the mobile node detachment and ensues the proxy binding update (PBU) functions, with a local mobility anchor to remove the binding state. This is further linked with the mobile node simultaneously, thereby, resulting in handover delay and packet loss. The research findings can be broadly divided into two phases. The first phase describes the implementation of existing CoAP IoT-based network mobility and the second phase is based on the implementation of partial bicasting with buffering for the IoT scheme. The proposed scheme is compared with the existing scheme by ns-3 simulations. We demonstrate that the proposed scheme could reduce handover delays, packet losses, end-to-end delay, throughput, and energy consumption, compared to the existing scheme.

2. Related Work

The requirement of faster "anywhere, anytime, and anyway" internet has been increasing due to the rapid increase in the number of mobile phone users and the development of portable communication devices, such as cellular phones, smartphones, laptops, and other modern

technologies systems [7]. With recent developments in wireless technologies, such as WCDMA and IEEE 802.16d, IETF, ITU-T, and third-generation partnership project (3GPP), the ubiquitous computing environments have been realized. However, achieving the goals of communication technologies is challenging. Compared to the regular internet, the general communication devices in IoT are turned into smarter devices and modern communication systems are turned into highly informative systems, as IoT possesses intelligent processing. However, IoT communication is through middleware and fundamental protocols [11,12]. Moreover, a functioning connection is important for IoT. The communication between the endpoints should be energy and time efficient. Therefore, IoT procedures must identify the best communication protocol and the CoAP is the principally employed protocol in IoT. CoAP is specified in RFC 7252 and is an open IETF standard compliant. This is a web transmission protocol used in either nodes or restricted networks, such as IoT, Wireless Sensor Network (WSN), etc. The protocol is designed for the resource-constrained IoT, which has lower memory and power consumption. CoAP is also referred to as the "web of things protocol" [13], as it is designed for web applications. It can be used to transport data from a few to thousand bytes in web applications. Principally, CoAP is an efficient RESTful protocol for an integrated web app transfer (CoAP://). The methods used in CoAP are GET, POST, PUT, and DELETE [14,15]. CoAP uses a simple and small 4-byte header. For secure message transmission, CoAP employs certified protection protocols. Additionally, confirmable and non-confirmable messages are used for reliability. The port number used for secure CoAP is 5683 [9]. The message queue telemetry transport (MQTT), an ISO standard method (IEC/ISO PRF 20922) reported in 1999, uses the publish-subscribe based message pattern. MQTT is considered for smaller M2M communications. Although MQTT was established by IBM, it is an open source package. MQTT uses Transmission Control Protocol (TCP) for message transport. The port numbers for MQTT are 1883 and 8883. MQTT works over TCP/IP and provides communication pattern flexibility. MQTT uses a topic-based publish-subscribe architecture [16]. This architecture is based on three components:

1. Publishers: Publishers act as sensors in IoT and communicate with subscribers through brokers. Importantly, publishers can bring the system to sleep-mode, as needed;
2. Brokers: Brokers bridge between publishers and subscribers. The broker is responsible for the categorization of all the information collected from publishers. Further, the brokers transmit sensor data to the subscribers;
3. Subscribers: Subscribers are the application end-users, on whom brokers are interested when the publishers transfer new data to the broker [17].

The secure MQTT (SMQTT) [18] is the improved form of MQTT. Although all the characteristic features in this log work are similar to MQTT, additional security functionality was added to improve the properties of MQTT. This algorithm uses four parts: (a) Set-up, (b) encryption, (c) publication, and (d) decryption. The brokers communicate with subscribers and publishers and receive a passkey. The publishers will encrypt the data to be published. Subscribers receive broker information. Subscribers with the same passkey could, therefore, decrypt the data. Notably, the key generation algorithm is dynamic.

The advanced message queuing protocol (AMQP) is specifically used in the financial sector and works additionally as a MQTT protocol. This publish-subscribe model-based [19] protocol employs a telecommunication protocol. The key components are (1) queues and (2) exchange.

1. Queues: The queues represent the logged-in subjects and subscribers. Therefore, queued data are transmitted to the subscribers;
2. Exchange: Exchange is responsible to retrieve the publishers' data and distribute it to the predefined queues [19].

Furthermore, IoT interconnects physical gadgets and users for implementing specific tasks and sharing data. As IoT hosts separate protocols for communication at different levels and we employed similar protocols. We employed CoAP, as it is the best IoT protocol and employed UDP for the transport layer. As the signal sensors need to be attached on the patients' body, we employed Proxy Mobile IPv6 (PMIPv6) for the network layer, and inserted 6LoWPAN into the network's abstraction layer, which works in conjunction with PMIPv6. Recently, the IETF approved the CoAP as an open

source package for investigating M2M and IoT interaction [9]. Upon clients' request on the server, CoAP employed four methods, PUT, POST, GET, and DELETE, which is similar to HTTP. However, unlike HTTP, CoAP employed UDP as a transport layer protocol to avoid message congestion and TCP-based extended resource requirement. Reliability was ensured through confirmation messages. The client might choose to acknowledge a message. CoAP is a simple and cost-effective protocol developed for low-end microcontrollers and high-bandwidth, high-error-burdened networks, such as 6LoWPANs. Furthermore, CoAP is defined by the open standard IETF RFC 7252 and is the default protocol for UDP. Additionally, CoAP can be implemented in other channels, such as TCP or DTLS. The CoAP is based on the request-response communication model and includes support for resource identification, improved reliability, Uniform Resource Identifiers (URIs), etc. Although the protocol was originally developed for M2M, it has been adapted for gateway-supported IoTs, high-end servers, and business integration. Although CoAP behaves similarly to HTTP in the REST model, with GET, POST, PUT and DELETE commands, it should not be considered as compressed HTTP. For URI, response codes, MIME types, etc., CoAP, however, can easily be connected to HTTP proxy mechanisms, where HTTP clients can communicate with CoAP servers, enabling better web service integration whilst meeting IoT requirements.

The TCP server forwards information to the server and conveys the message to the subscribers, immediately. TCP also executes an error checksum. In UDP, the sender constantly sends information to the receiver, without ensuring its purposeful receipt. For example, live video streaming continues if the next packet is sent via UDP. Although the video transmission is blocked at the particular time, it will accurately resume within milliseconds.

PMIPv6, an IETF-designed network-based mobility management protocol, is defined in RFC 5213. PMIPv6 supports a proxy role for the network game operator for the mobile node in IP mobility reporting. The system's mobility substances follow the mobility signal, the MN movement, and the configuration of the requested routing status. The most important functional units are the MAGs and LMAs. MAG carries out mobility management. The MAG exists on the access link where the mobile node is anchored. LMA maintains the reachability status of the mobile node and is the topological anchor of the IP address of the mobile node. The Cisco wireless LAN controller (WLC) implements the MAG feature. The key objective of this protocol is to provide mobility support for each IPv6 host within the localized and topologically limited network, without the host's participation in mobility signaling. Significant PMIPv6 capabilities are supported for unmodified IPv4 and IPv6 MNs, which efficiently use wireless network resources, are independent interconnected technology, and improve crossover performance. A method was proposed to reduce transmission delays in CoAP [11].

Figure 1 illustrates the "CoAP-PMIP" scheme [11]. The sensor was connected to MAG_A, which sent the PBU to LMA. LMA registered the IP address of the sensor and sent a confirmation as a proxy binding acknowledgement (PBA) to MAG_A (Steps 1, 2, and 3). To convey a communication request, the client sent a binding query to MAG_C, and MAG_C sent the binding query to LMA. Since LMA bears the IP address and other sensor-specific values, it receives a binding query and PBA (Steps 4, 5, and 6). Therefore, to resume a new handover, the first sensor was connected to new MAG_B and its address updated in the LMA table, after handover. After the update, LMA sent the PBA to MAG_B (Steps 8, 9, and 10). Therefore, to communicate with the sensor, the client conveyed a binding acknowledgment query to the LMA with MAG_C. The LMA could communicate, as it contained a new sensor device value (Step 11).

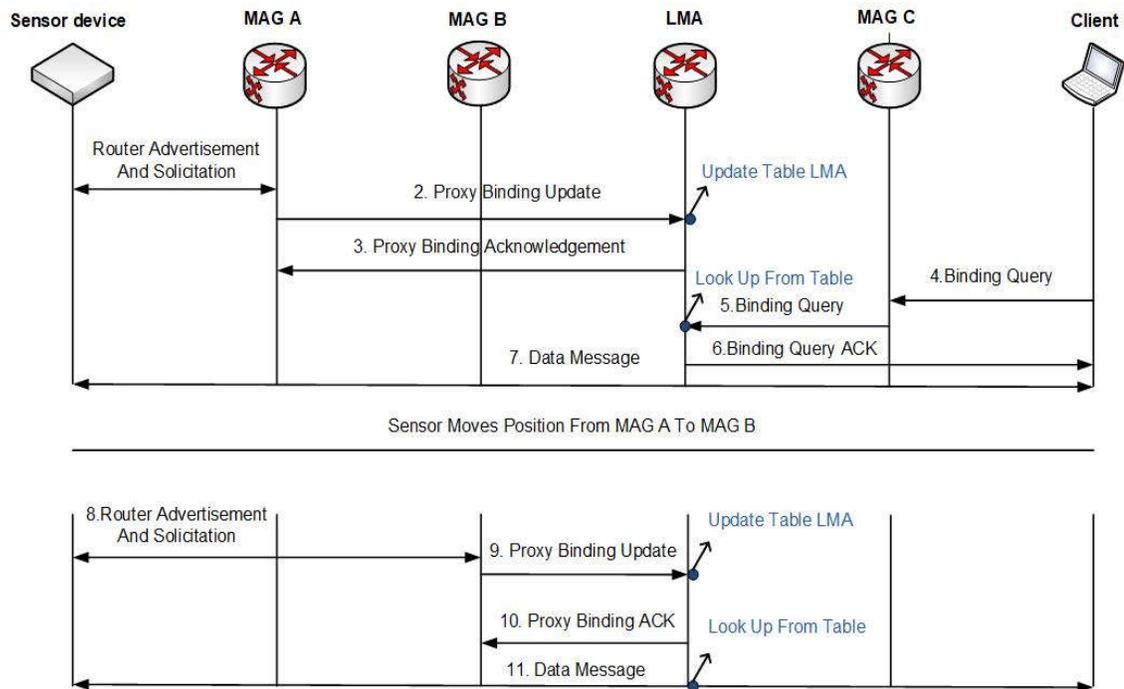


Figure 1. Constrained Application Protocol (CoAP) for Proxy Mobile IPv6 (PMIPv6).

3. Proposed Scheme

In the proposed scheme, an IoT device moved into a new network and its MAG updated the binding to LMA. LMA initiated the partial bicasting of the data packets to the new and previous MAGs. The data packets were buffered at the new MAG, during handover and were forwarded to MN after handover.

Figure 2 illustrates the proposed Partial Bicasting for PMIP (PB-PMIP) handover with IoT-based bicasting. MAGold received a link-layer message from the link-detected. MAGold requested MAGnew to establish the PMIP tunnel with LMA by sending an INIT message. MAGnew sent a PBU to LMA. LMA transmitted the data packets to MAGold and MAGnew. These contained the transmission of handover INIT from the MAGold to the MAGnew, and the PBU and PBA messages between the MAGnew and the LMA. Thus, the bicasting was transmitted in the partial network area between LMA and MAGnew. Upon the receipt of the PBA from the LMA, MAGnew began to buffer data from the LMA and commanded MAGold to terminate bicasting by sending a handover acknowledgement (ACK) message. MAGold released the old PMIP tunnel by sending a PBU message to the LMA. When the new connection was established, MAGnew transferred the buffered data packets to the sensor device. Thus, a normal data was transferred between sensor device and LMA.

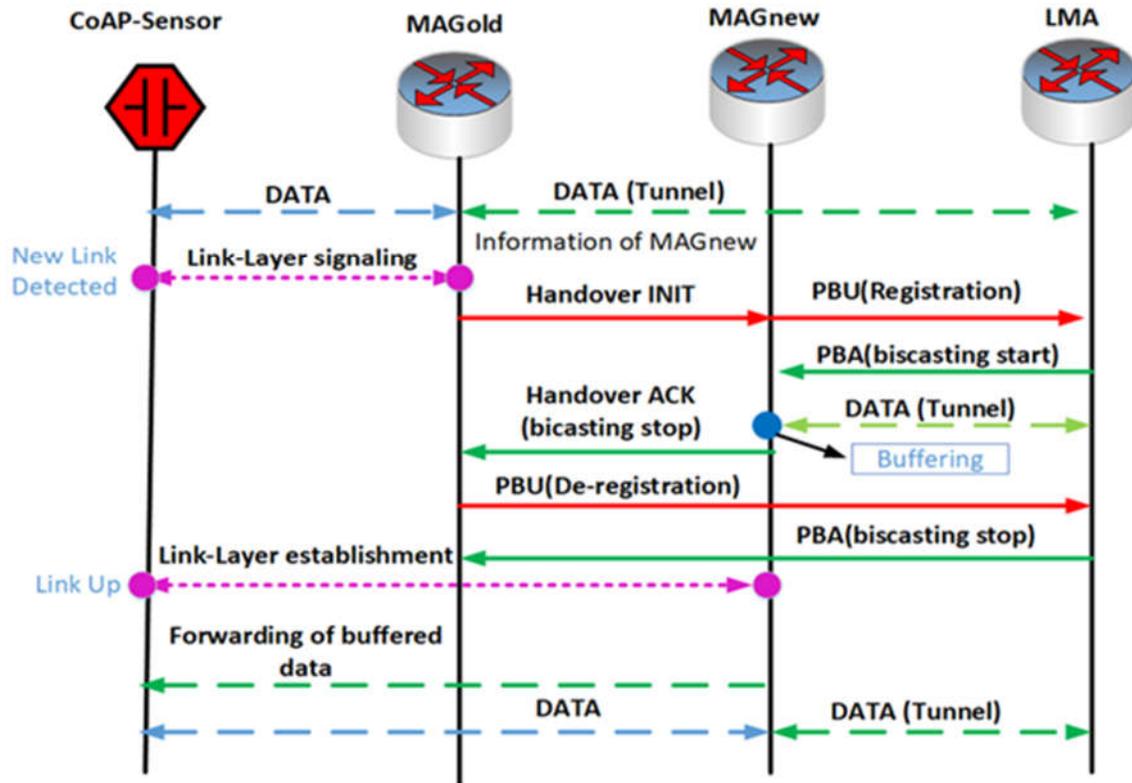


Figure 2. Partial Bicasting for PMIP (PB-PMIP) for internet of things (IoT).

In PB-PMIP handover, the partial region was bicast between the LMA and MAGnew, so that wirelessly interconnected network resources were dispensable during handover. Data loss during handover was reduced by using MAGnew buffering.

Figure 3 depicts the protocol stack of our proposed partial bicasting scheme. CoAP protocol was used in the application layer. CoAP had a lower overhead and was light-weighted due to the UDP. UDP was used in the transport layer for packet delivery.

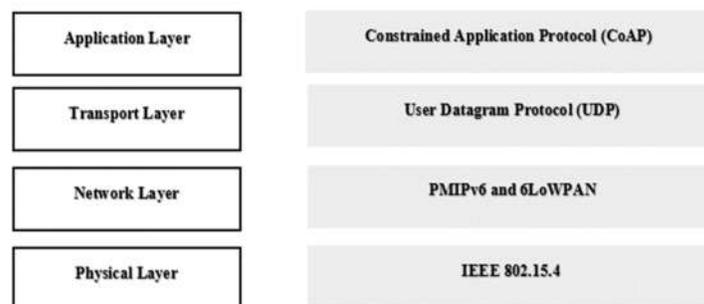


Figure 3. Protocol stack of proposed PB-PMIPv6 for IoT.

4. Simulation Analysis and Results (Experimental Analysis)

4.1. Simulation Analysis

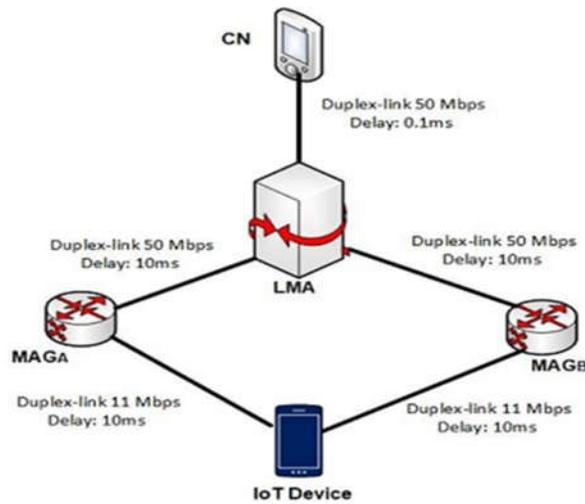


Figure 4. Simulation network model.

NS3 simulation was implemented. Figure 4 illustrates the simulation network model for the proposed and existing schemes.

4.2. Simulation of Existing CoAP-PMIPv6

Figure 5 illustrates the sensor node transmission from MAG_A to the client, via MAG_B and LMA. The communication is as described in the previous section. This simulation was positioned before the handover state. Figure 6 depicts the changed sensor position. The sensor restarted the communication, after the transmission handover.



Figure 5. CoAP-PMIPv6 before handover (NetAnim view).

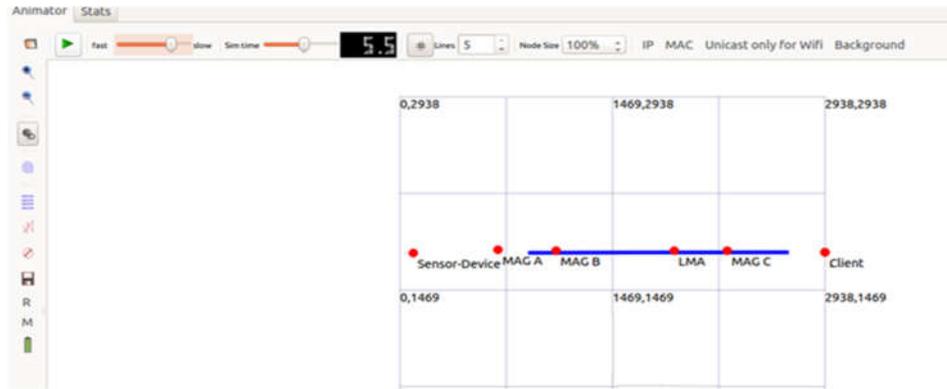


Figure 6. CoAP-PMIPv6 after handover (NetAnim view).

4.3. Simulation of Proposed Scheme

Figure 7 illustrates PB-PMIP for IoT. The sensor was transmitting from MAGold to MAGnew, via PB-LMA. The communication is as described in the previous section. In PB-PMIP handover, bicasting was performed in the partial region between the PB-LMA and the MAGnew, so that wirelessly interconnected network resources were dispensable during handover. Data loss during handover was reduced by using MAGnew buffering.



Figure 7. NetAnim view of PB-PMIP for IoT.

4.4. Results (Experimental Evaluations)

To evaluate the functioning of any proposed scheme, comparing its performance with that of an existing scheme is important. Therefore, we compared the performance of the proposed PB-PMIP scheme to the existing CoAP-PMIP scheme, using the ns-3 simulator. Table 1 lists the simulation parameters.

Table 1. Simulation parameters.

Link between MAGs and LMA	50 Mbps and Delays 10 ms
Link between IoT devices and MAGs	11 Mbps and delays 10 ms
Handover occurs	at the time of 20.5 second
Operating System	Ubuntu 14.04 LTS
Simulation Software	NS 3.19

Animation Viewer	NetAnim
Data Tracing and Graphs Plotting	Wireshark, MATLAB and Excel

4.4.1. Data Packet Traces

Figure 8 illustrates handover delays and packet losses for the two schemes, CoAP-PMIP and PB-PMIP. The transmission of CoAP-PMIP resulted in significant packet loss and handover delays, compared to bicasting handover of PB-PMIP. The proposed PB-PMIP scheme resulted in lower packet loss than the current CoAP-PMIP transmission. The PB-PMIP scheme employed the MAGnew buffer to reduce data loss during transmission.

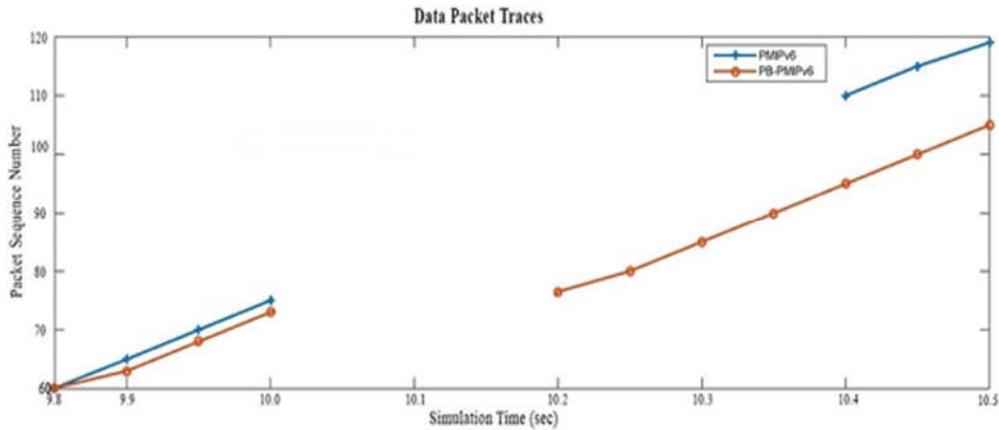


Figure 8. Comparison of data packet trace during simulation.

4.4.2. Handover

Figure 9 illustrates the handover delays of the two entrant schemes for distinct link switching times. As the link switching time for all entrant schemes increased, handover delay also increased, and the PB-PMIP scheme exhibited less handover delays than CoAP-PMIP. PB-PMIP provided substantially similar handover delays for all link switching times.

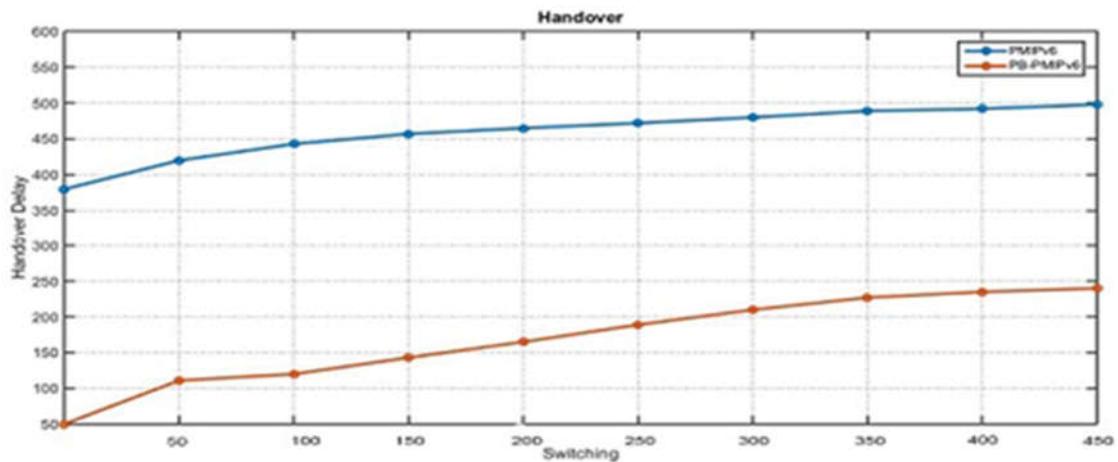


Figure 9. Comparison of handover delays during handover.

4.4.3. Packet Loss during Handover

Figure 10 depicts the packet loss in the existing CoAP-PMIP scheme and increase in packet loss as a function of the link switching time. The PB-PMIP scheme was efficient with a relatively long link switching time. However, packet loss was not observed with the PB-PMIP scheme, even with

increased link-switching time. Here, the data packets were buffered in the MAGnew and were transmitted to the MAGnew-attached MN.

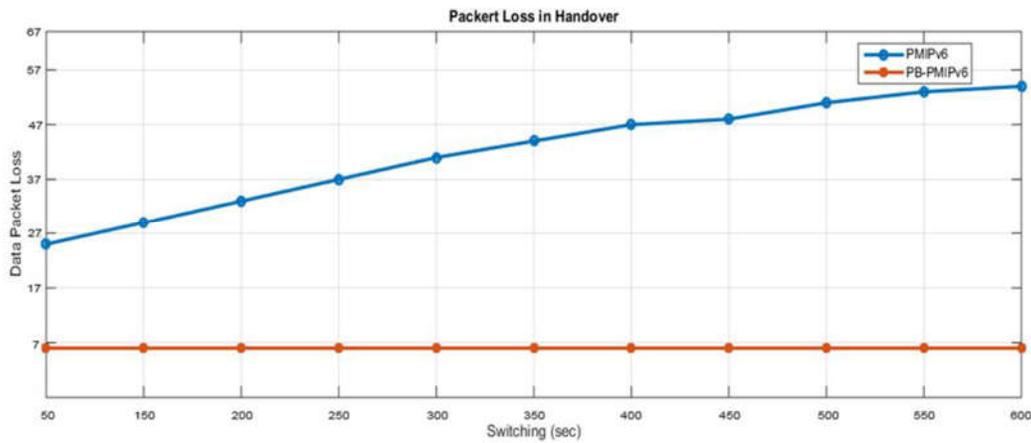


Figure 10. Comparison of lost packets during handover.

4.4.4. Throughput

Throughput is defined as the total number of packets sent in a time period. The exiting scheme exhibited lower throughput, due to the device dysconnectivity (Figure 11). Every device had to reconnect with each other after the handover to establish communication. However the presence of data tunnel between MAGnew and LMA in the proposed scheme increased the throughput.

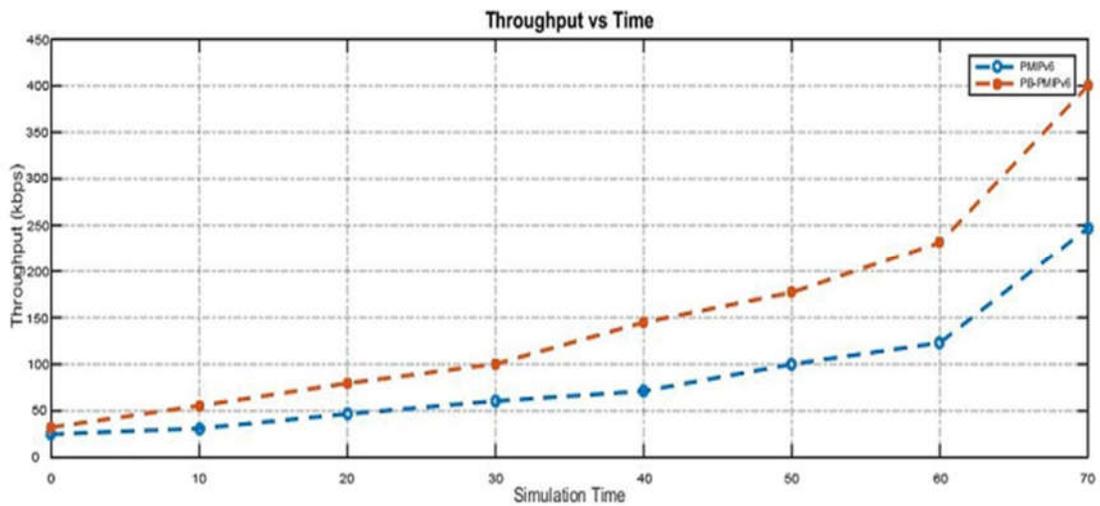


Figure 11. Comparisons of throughput vs time.

4.4.5. End to End Delay

Figure 12 illustrates the end to end delay between the entrant schemes. Due to the lack of communication at the beginning and indifferent delay, both the schemes exhibited a similar delay. When the handover occurred at 20.5 s, the sensors of the existing scheme were disconnected due to mobility and inefficiency. After the handover, every sensor needed to be reconnected to continue communication. However, the proposed scheme exhibited increased delay, which reduced upon employing MAGnew.

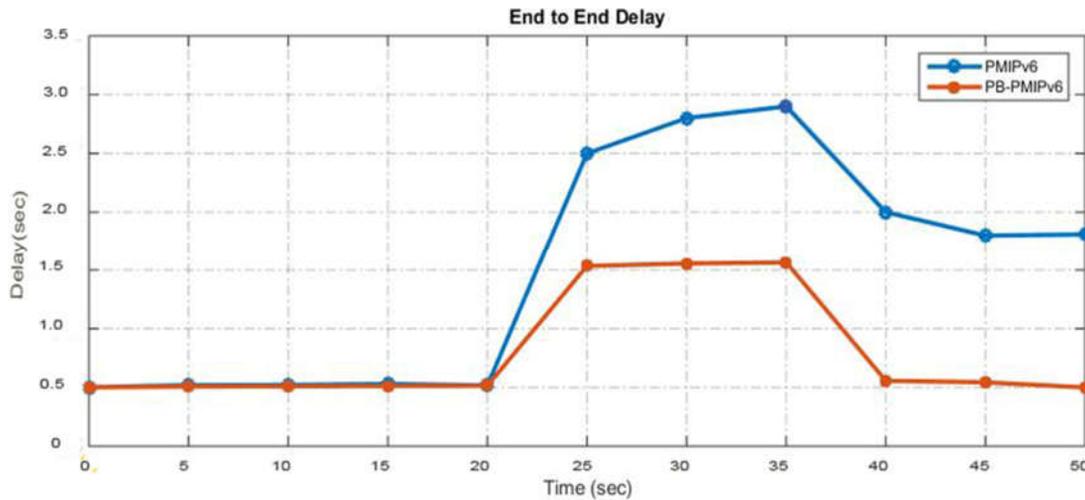


Figure 12. Comparison of end to end delays.

4.4.6. Energy Consumption

Energy utilized by the devices is called energy consumption. Figure 13 depicts the energy consumed during communication in different schemes. PB-PMIPv6 was more energy efficient than CoAP-PMIPv6 (Figure 13). During handover, each sensor was needed to reconnect to other energy-consuming sensors. However, PB-PMIPv6 consumed less energy due to the presence of MAGnew, wherein network resources were not required after handover.

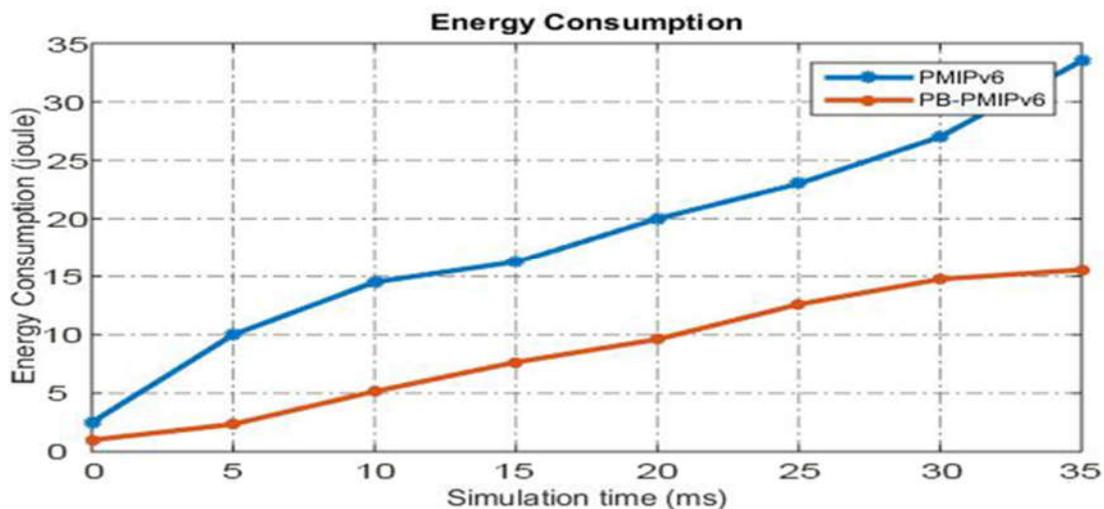


Figure 13. Comparison of energy consumed.

5. Conclusions

Here partial bicasting with buffering to improve performance of the PMIP handover is presented with bicasting in the partial region occurring between the LMA and MAGnew. The data packets were buffered in the MAGnew during handover to reduce delay and packet loss. Therefore, the wireless interconnect network resources were dispensable during handover. Packet loss during handover was reduced by using MAGnew buffering. Simulation results illustrated that the proposed handover scheme was efficient in handover delay, packet loss during handover, end-to-end delay, throughput, energy consumption, and data packet traces compared to the existing scheme. In future, we will implement partial bicasting with the buffering scheme (PB-PMIPv6), for group-based mobility management in IoT.

Author Contributions: Writing-original draft preparation: M.G., S.A. and M.A.; writing—review and editing: J.-G.C. and H.A.; project administration and funding acquisition: S.-J.K. All authors have agreed to the published version of the manuscript.

Funding: This research was supported by the Technology Innovation Program on National Standard of MOTIE in Republic of Korea (20002214).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Atzori, L.; Iera, A.; Morabito, G. The Internet of Things: A survey. *Comput. Netw.* **2010**, *54*, 2787–2805.
2. RFC 7252—The Constrained Application Protocol (CoAP). Available online: <https://tools.ietf.org/html/rfc7252> (accessed on 28 November 2019).
3. Cerf, V.; Icahn, R. A protocol for packet network intercommunication. *Comp. Commun. Rev.* **2005**, *35*, 71.
4. Zolfagharnasab, H. Reducing Packet Overhead in Mobile Ipv6. *Int. J. Distrib. Parallel Syst.* **2012**, *3*, 1–8.
5. Ganz, F.; Li, R.; Barnaghi, P.; Harai, H. A Resource Mobility Scheme for Service-Continuity in the Internet of Things. In Proceedings of the 2012 IEEE International Conference on Green Computing and Communications, Besancon, France, 20–23 November 2012.
6. Malki, K.E.; Soliman, H. Simultaneous Bindings for Mobile IPv6 Fast Handoffs. Available online: https://www.researchgate.net/publication/240242303_Simultaneous_Bindings_for_Mobile_IPv6_Fast_Handovers (accessed on 1 January 2003).
7. Kong, K.-S.; Lee, W.; Han, Y.-H.; Shin, M.-K.; You, H. Mobility management for all-IP mobile networks: Mobile IPv6 vs. proxy mobile IPv6. *IEEE Wirel. Commun.* **2008**, *15*, 36–45.
8. Höller, J.; Höller, J. *From Machine-to-Machine to the Internet of Things*; Academic Press/Elsevier: Amsterdam, Netherlands, 2014.
9. Woolley, S.C.; Howard, P.N. Automation, algorithms, and politics - political communication, computational propaganda, and autonomous agents Introduction. *Int. J. Commun.* **2016**, *10*, 9–14.
10. Sridharan, S.; Shrivastava, H. Excogitation of secure data authentication model for wireless body area network. In Proceedings of the International Conference on Computer Communication and Informatics (ICCCI), IEEE, Coimbatore, India, 3–5 January 2014; pp. 1–7.
11. Choi, S.-I.; Koh, S.-J. Use of proxy mobile IPv6 for mobility management in CoAP-Based internet-of-things networks. *IEEE Commun. Lett.* **2016**, *20*, 2284–2287.
12. Ray, P. A Survey on Internet of Things Architectures. *EAI Endorsed Trans. Internet Things* **2016**, *2*, 151714.
13. What Is CoAP IoT Protocol CoAP Architecture Message Header. Available online: [Rfwireless-world.com](http://rfwireless-world.com) (accessed on 28 November 2019).
14. Khattak, H.A.; Ruta, M.; Sciascio, E.D. CoAP-based healthcare sensor networks: A survey. In Proceedings of the 11th International Bhurban Conference on Applied Sciences and Technology (IBCAST), IEEE, Islamabad, Pakistan, 14–18 January 2014; pp. 499–503.
15. Tapio, L.; Mazhelis, O.; Suomi, H. Comparing the cost-efficiency of CoAP and HTTP in Web of Things applications. *Decis. Support Syst.* **2014**, *63*, 23–38.
16. Thangavel, D.; Ma, X.; Valera, A.; Tan, H.; Tan, C. Performance evaluation of MQTT and CoAP via a common middleware. In Proceedings of the IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Singapore, 21–24 April 2014.
17. Kim, S.-M.; Choi, H.-S.; Rhee, W.-S. IoT home gateway for auto-configuration and management of MQTT devices. In Proceedings of the 2015 IEEE Conference on Wireless Sensors (ICWiSe), IEEE, Melaka, Malaysia 24–26 August 2015; pp. 12–17.

18. Singh, M.; Rajan, M.; Shivraj, V.; Balamuralidhar, P. Secure MQTT for Internet of Things (IoT). In Proceedings of the Fifth International Conference on Communication Systems and Network Technologies, Gwalior, India, 4–6 April 2015.
19. Vasileios, K.; Chatzimisios, P.; Vazquez-Gallego, F.; Alonso-Zarate, J. A survey on application layer protocols for the internet of things. *Trans. IoT Cloud Comput.* **2015**, *3*, 11–17.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).