RELIABLE MULTICAST DISRUPTION TOLERANT NETWORKING PROTOCOL FOR INTERNET OF THINGS

Khang-Siang Wong Tat-Chee Wan

October 30, 2018

Abstract

The Internet of Things (IoT) is a new emerging network-based paradigm to interconnect almost everything to the Internet for specific purposes such as environmental monitoring and Supervisory control and data acquisition (SCADA) data logging. However, network disruption may occur anytime and causes IoT applications failed to work as expected. Delay Disruption Networking (DTN) is devised as a class of network architectures that address the lack of continuous network connectivity. DTN helps to bridge the gaps between partitioned networks and enable IoT devices to store-carry-forward data whenever communication link is available. IoT applications with DTN-enabled are robust and efficient even in scenarios with an intermittent and unstable network connection. Many potential IoT applications often require group based reliable data delivery services for data collection and data dissemination especially critical configuration updates from the control center. In this research, Reliable Multicast Disruption Tolerant Networking (RMDTN) routing protocol is proposed to ensure data bundles are delivered reliably and efficiently to the group of receivers and sender able to verify the data bundles have been delivering properly to all intended recipients. The preliminary conceptual implementation of RMDTN protocol using message ferry is presented and the results show that RMDTN protocol performs better than epidemic and flooding DTN routing protocols and better storage utilization for non-member node.

1 INTRODUCTION

The goal of the Internet of Things (IoT) [1] is to interconnect smart objects with embedded computing capability where each of them is unique on the Internet and identifiable to each other. IoT-enabled devices, sensors, appliances, wearables, routers and even human are comprised as part of this largely scalable IoT network and aim to integrate the physical world into intelligent computer systems. Researchers believe this realization able to improve efficiency, increase economic benefits and further reduce human interventions in an automated system. Some common IoT applications are smart home, environmental monitoring and industry 4.0. IoT will be eventually integrated as part of our life and bring our daily life to a new dimension.

Currently, IoT devices work in a way that they continuously collect and exchange data over the network which often required to establish end-to-end network path between the data source and the destination using the conventional Transmission Control Protocol (TCP) Internet Protocol. They are expected to work seamlessly but the performance is unsatisfactory when it involves the opportunistic network. The nodes in the opportunistic network have limited and intermittent communication opportunity where an end-to-end network path between the source and the destination might never be established. Challenged and stressed environment always presents such intermittent network connectivity issues to IoT devices. IoT devices tend to discard messages more frequently when no path is found to reach the destination under the existing networking protocol.

Disruption Tolerant Networking (DTN) [2] is the potential enabler of IoT in the network that suffers from frequent network disruption with intermittent network connectivity. DTN employ store-carry-forward message switching mechanism to facilitate message forwarding even without established network path between source and destination. Although plenty of research has been done on DTN [3] in common computers and embedded systems, the integration of DTN in IoT still post great challenge due to the resource-constrained nature of IoT devices with low computing power, limited storage capability, and short battery life. Provisioning multicast services in DTN improve the routing efficiency and result in better resources utilization. For example, the sender only needs to send out one message addressed to a group of destinations (i.e. multicast) instead of sending one message to each of the destination in the group (i.e. unicast) where the latter consume more bandwidth to transmit messages which may lead to network congestion, more storage is used up to store a large number of messages and increase batteries usage due to high processing requirement.

The reliability requirement of multicast support in DTN further increase the complexity of IoT applications as IoT devices proliferates especially in the intermittently connected network. Data collection, as well as management and configuration of devices, require a certain level of reliability to ensure data is transmitted efficiently to the group of receivers and the sender able to verify data are received by intended receivers correctly. For example, manual reconfiguration of hundreds or thousands of environmental sensors in the field is time-consuming and require significant manpower. The availability of Over-The-Air (OTA) update helps to ease the reconfiguration burden to deliver a software update to the sensors. However, the lack of reliable wireless network connectivity in remote areas causes the updating task to be error-prone and difficult to verify that each device was updated properly. Reliable data collection, data dissemination and configuration updates from the control center to multiple participating IoT nodes become more complex in such network with intermittent connectivity.

The remainder of this paper is organized as follows. Section 2 introduces the main idea of the research work and Section 3 discusses preliminary results of experiments.

2 PROPOSED WORK

DTN is introduced as an experimental network architecture to address connectivity disruption in challenged environments result from limited wireless radio range of communication technologies, the sparsity of mobile nodes or resource-constrained network. DTN use store-carry-forward message switching to bridge the connection between intermittent and partitioned network by overlaying a Bundle Protocol (BP) [4] on the top transport layer. The architecture of DTN is described in Request for Comments (RFC) 4838 [5].

In the IoT application scenario, sensors and devices can be distributed over large geographical areas to collect environmental data. IoT devices generate data as usual although they experience frequent network disruption. They store the generated data as a message bundle in their persistent storage and wait for any communication link to be available to transmit over DTN. For example, workers or security patrols are scheduled to pass through the monitored area periodically by carrying communication devices that act as a data mule to provide network connectivity. Bundles queued in the nodes are forwarded to the data mule via low power short range air interfaces such as Bluetooth Low Energy (BLE) and data mule further forwards bundles to control center automatically via available network links such as two-way radio devices, General Packet Radio Service (GPRS), or Long-Term Evolution (LTE).

This research focuses on provisioning a new multicast service in DTN-enabled IoT network with the reliability requirement to ensure reliable data collection and dissemination especially in delivering OTA configuration updates to the pool of devices. IoT applications require multicasting service because group messages need to be delivered to intended receivers eventually regardless of the long delay and dynamic network [6]. Existing DTN protocols didn't address the multicast issues in IoT context. Furthermore, the reliable multicast capability is added to the existing DTN to enhance reliability and network efficiency by devising a new routing protocol namely Reliable Multicast Disruption Tolerant Networking (RMDTN) protocol. Figure 1 shows the RMDTN framework consists of static nodes, mobile nodes, control center and end users. As in the previous example, message bundles are queued and wait for the communication link to be available for transmission when the workers or security patrols visit the areas without network connectivity.

RMDTN protocol is a common system-level protocol that manages bundles forwarding and retransmission over intermittent links when disruption occurs. By incorporating Reliable Multicast capability into DTN, data would be sent as RMDTN bundles to all the intended recipients. Reliable Multicast ensures that the message bundles are transmitted efficiently to the group of receivers, where message bundles are received by all intended recipients correctly and sender able to verify each message bundle's delivery status. RMDTN is well suited for performing this kind of data collection and update dissemination task for intermittently connected IoT devices especially the updated configuration settings from the control center to a group of sensors. Furthermore, the RMDTN middleware reduces the complexity to develop IoT applications by removing the need to implement error transmission error recovery mechanism in each application hence simplify the development process.

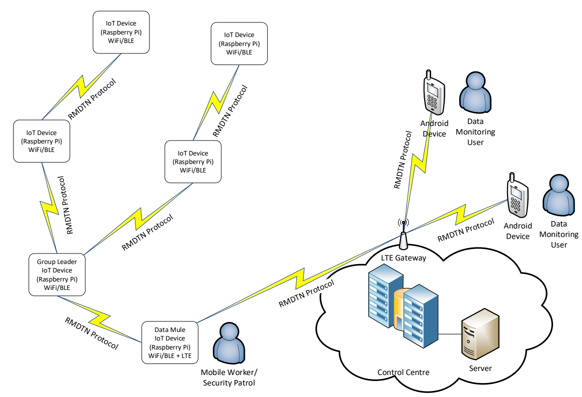

Figure 1 RMDTN framework [7]

3 RESULTS

The RMDTN protocol is implemented as an additional routing extension in IBR-DTN [8] with full BP implement using the lightweight C uClibc library for an embedded system. IBR-DTN is a middleware library for DTN bundle routing and communication sit on top of the transport layer and below the application layer. Each device running IBR-DTN with RMDTN extension will be able to multicast data in DTN reliably with higher efficiency and low resource overhead. Two modules are added to IBR-DTN to enable RMDTN routing: Group management module and Multicast routing module. RMDTN enables runtime multicast group management by exchanging group membership map upon each encounter with RMDTN-enabled nodes. The multicast routing module handles all the multicast transmission based on the group membership map and select relays if the encountered nodes are not the destinations [9].

We setup an ethernet testbed to examine the feasibility of RMDTN protocol using Raspberry Pis as shown in Figure 2. Each Raspberry Pi is installed with IBR-DTN and running with the RMDTN routing extension. Three Virtual Local Area Network (VLAN) is configured to represent three partitioned networks in the remote area and the ferry will switch to different VLAN to represent periodic visits to the partitioned networks. The ferry is a switch to VLAN 201, VLAN 202 and VLAN 203 respectively and repeated every five minutes using a python script by changing the port's VLAN connecting to the ferry. Node 1 sends an image with a file size of 361 KB every half minute to the group dtn://temperature for 60 minutes and the ferry act as a forwarding node to deliver the image file to nodes in different VLANs. The goal of this experiment is to prove the advantages of RMDTN protocol using message ferry.

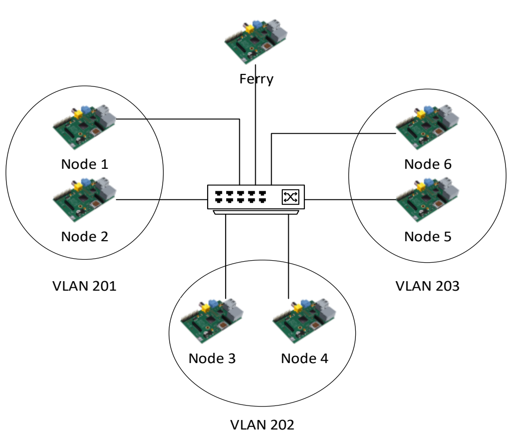

Figure 2 RMDTN experiment setup [7]

Figure 3 and figure 4 show the experiment results. Node 1 is the data source of the group message and Node 2, Node 4, as well as Node 6, are subscribed to the same group as Node 1. Ferry node has the highest number of received bytes due to its responsibility as data mule to accept all the bundles from the encountered node. When using RMDTN protocol, Node 3 and Node 5 which are not subscribed to the group have almost zero MB throughput for a number of bytes received and transmitted from the network interface. This infers that the non-group members are free from storage exhaustion when using the RMDTN protocol. RMDTN protocol performs better compared to flooding protocol but slightly lower to epidemic protocol. Flooding tends to increase the delivery rate by forwarding every bundle to every encountered node and this result in the highest resource consumption among the three protocols. RMDTN protocol makes forwarding decisions based on the group membership maps to the destinations and in selecting relays.

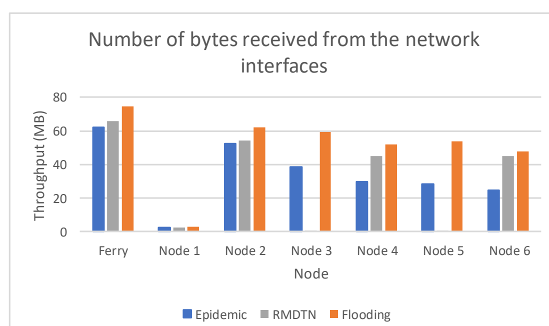

Figure 3 The number of bytes received from the network [7]

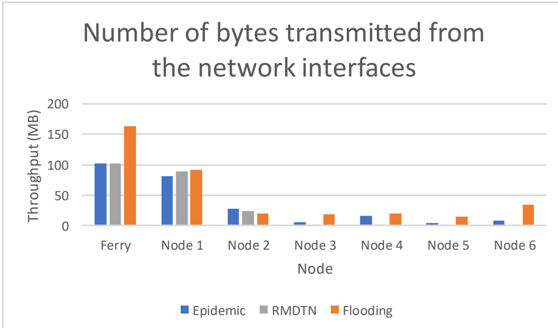

Figure 4 The number of bytes transmitted from the network interfaces [7]

Figure 5 shows the bundles stored in each node when the experiment is done. The delivery ratio for RMDTN protocol is 100% while flooding protocol with higher resource consumption slightly lower than RMDTN protocol. However, the delivery ratio for Epidemic routing is only around 60-70%. Node 3 and Node 5 still suffer from storage exhaustion by receiving bundles that are not designated to them.

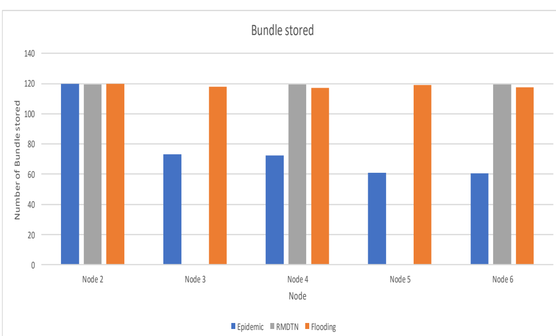

Figure 5 Bundle stored for each node at the end of the experiment [7]

References

[1] J. Holler, V. Tsiatsis, C. Mulligan, S. Avesand, S. Karnouskos, and D. Boyle, From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence, 1st ed. Academic Press, 2014.

[2] K. Fall, "A Delay-tolerant Network Architecture for Challenged Internets," in Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, 2003, pp. 27–34.

[3] S. CC, V. Raychoudhury, G. Marfia, and A. Singla, "A survey of routing and data dissemination in Delay Tolerant Networks," J. Netw. Comput. Appl., vol. 67, pp. 128–146, 2015.

[4] K. L. Scott and S. Burleigh, "Bundle protocol specification," 2007.

[5] V. Cerf et al., "Delay-Tolerant Networking Architecture," 2007.

[6] W. Zhao, M. Ammar, and E. Zegura, "Multicasting in Delay Tolerant Networks: Semantic Models and Routing Algorithms," in Proceeding of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking, 2005, pp. 268–275.[7]K.S. Wong and T.C. Wan, "Reliable Multicast Disruption Tolerant Networking : Conceptual Implementation Using Message Ferry," pp. 1817–1822, 2017.

[8] S. Schildt, J. Morgenroth, and L. Wolf, "IBR-DTN: A lightweight, modular and highly portable Bundle Protocol implementation," vol. 37, 2011.

[9] F. Franke, "Implementation of Multicast in I BR-DTN," 2013.