

Rancang Bangun Perangkat Mobile Berbasis *Delay Tolerant Network* Sebagai Perantara Pengiriman Data Sensor Dari Lapangan Ke Pusat Data

Hendra Danu Laksana¹, Adhitya Bhawiyuga², Kasyful Amron³

Program Studi Teknik Komputer, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹hendradanu.official@gmail.com, ²bhawiyuga@ub.ac.id, ³kasyful@ub.ac.id

Abstrak

Pada pengembangan teknologi berbasis *Internet of Things* (IoT) terdapat 3 komponen, yaitu *hardware*, *middleware*, dan *presentation*. Permasalahan yang muncul adalah *hardware* yang dipakai pada *sensor node* cenderung terbatas dari segi pemroses maupun memori. Maka dari itu, *sensor node* hanya bertugas untuk proses pembacaan data sensor dan mengirimkannya ke pusat data untuk penganalisisan data dan pengambilan keputusan. Pengiriman data pada sistem IoT dari *sensor node* ke pusat data dapat menggunakan berbagai macam protokol, tetapi dari berbagai macam protokol tersebut belum ada yang dapat mengatasi permasalahan pengiriman data pada kondisi jaringan yang memiliki konektivitas yang intermiten. Sebagai upaya untuk mengatasi permasalahan tersebut, maka pada penelitian ini dirancang sebuah mekanisme komunikasi yang mengimplementasikan konsep *store-carry-forward*. Salah satu mekanisme komunikasi yang menerapkan konsep tersebut adalah *Delay Tolerant Network*. Mekanisme yang dirancang akan diterapkan pada perangkat *mobile* berbasis sistem operasi Android. Hasil pengujian menyatakan bahwa sistem dapat mengirimkan pesan dengan tingkat keberhasilan 100% bahkan dalam kondisi sambungan antar-perangkat yang intermiten dan rata-rata *delay* pengiriman sebesar 51,8 detik. Hasil pengujian juga menyatakan bahwa sistem tidak terlalu membebani perangkat karena spesifikasi perangkat saat ini sangat mumpuni untuk menampung beban tersebut.

Kata kunci: *Delay Tolerant Network*, *IoT*, *Store Carry Forward*, *Sensor*, *Cloud*.

Abstract

Internet of Things (IoT) consist of three main components: hardware, middleware, and presentation. The emerging problem is used hardware in sensor node have constrained processor and memory ability. Therefore, sensor node only functionate as sensor data reader and sender to data center for data analysis and decision making. Data sending in IoT system from sensor node to data center can use several protocol, but no one of those procotols can be used in intermittent connected network. Attempting to overcome these problems, a system that implements the mechanism of store-carry-forward data transmission is designed in this research. One of communication mechanism that implement store-carry-forward is Delay Tolerant Network. The designed mechanism will be applied to Android OS-based mobile device. The test results state that the system can transmit messages with a 100% success rate even when inter-device connection conditions are intermittently connected and the average delay is 51.8 seconds. The test result also states that the system is not too burdensome device because the device specifications that exist today is very qualified to accommodate the load.

Keywords: *Delay Tolerant Network*, *IoT*, *Store Carry Forward*, *Sensor*, *Cloud*.

1. PENDAHULUAN

Pada pengembangan teknologi berbasis *Internet of Things* (IoT) terdapat 3 komponen, yaitu *hardware*, *middleware*, dan *presentation*. Pada komponen *hardware* terdiri dari sensor, aktuator, dan *hardware* komunikasi. Pada komponen *middleware* terdiri dari *on demand*

storage dan fitur untuk analisis data. Pada komponen *presentation* terdiri dari pusat data yang memiliki fitur visualisasi dan interpretasi yang dapat diakses pada berbagai *platform* dan dapat dirancang untuk berbagai penerapan (Gubbi, et al., 2013).

Terdapat beberapa masalah yang muncul pada kemampuan *hardware sensor node*.

Permasalahan pertama adalah *hardware* yang dipakai pada *sensor node* cenderung terbatas dari segi pemroses maupun memori (Hauke, et al., 2014). Maka dari itu, *sensor node* hanya bertugas untuk proses pembacaan data sensor dan mengirimkannya ke pusat data untuk penganalisisan data dan pengambilan keputusan (Munich, et al., 2016).

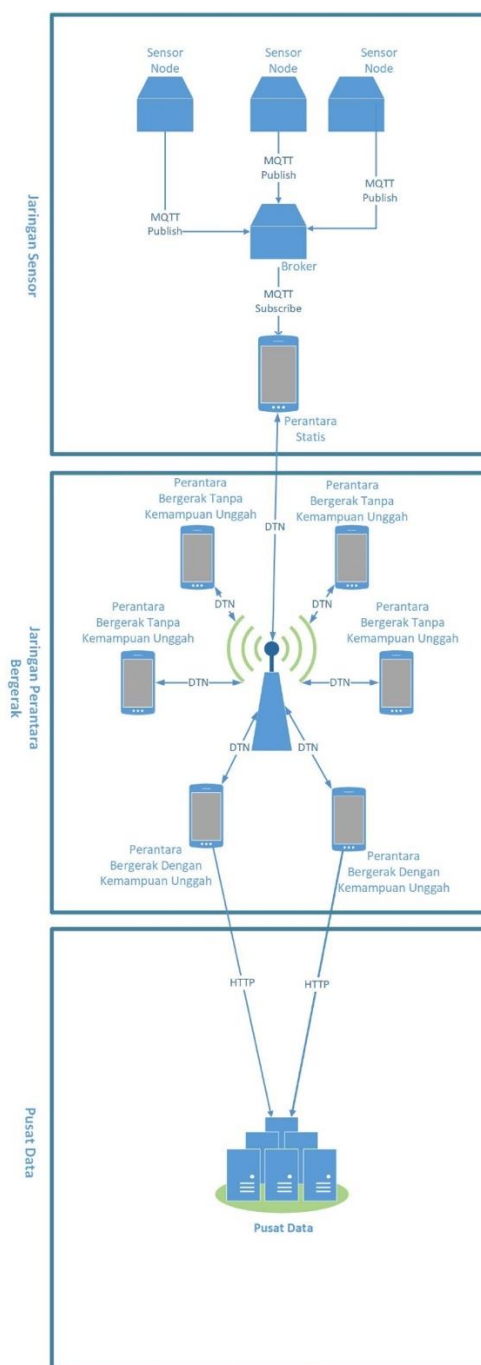
Pengiriman data pada sistem IoT dari *sensor node* ke pusat data dapat menggunakan berbagai macam protokol, seperti MQTT, CoAP, AMQP dan HTTP. Berbagai protokol tersebut memiliki kelebihan dan kelemahan masing-masing, tetapi belum ada yang dapat mengatasi permasalahan pengiriman data pada kondisi jaringan yang memiliki konektivitas yang intermiten karena protokol-protokol tersebut harus membangun *end-to-end path* sebelum mengirim data (Naik & Nitin, 2017). Kondisi tersebut biasa ditemui ketika *sensor node* berada di pedesaan atau pegunungan dan pusat data berada di perkotaan.

Sebagai upaya untuk mengatasi permasalahan tersebut, maka pada penelitian ini dirancang sebuah mekanisme komunikasi yang mengimplementasikan konsep *store-carry-forward*, sehingga memungkinkan data dapat dikirim meskipun *end-to-end path* atau jalur koneksi ke tujuan tidak selalu ada. Salah satu mekanisme komunikasi yang menerapkan konsep tersebut adalah DTN (*Delay Tolerant Network*). DTN menggunakan *bundle protocol* sebagai standar komunikasinya. Dengan konsep komunikasi semacam ini, ketidaktersediaannya jaringan internet dapat ditoleransi karena pesan disimpan, dibawa, dan diteruskan dari perangkat ke perangkat (Phearin, et al., 2013).

Mekanisme yang dirancang akan diterapkan pada perangkat *mobile* berbasis sistem operasi Android yang kemudian akan disebut sebagai perantara. Perantara dibagi menjadi 2, yaitu perantara statis dan perantara bergerak. Perantara statis adalah perantara yang berada di lokasi *sensor node* dan bertugas menjembatani komunikasi *sensor node* dan perantara bergerak. Perantara bergerak dibagi menjadi 2, yaitu perantara bergerak dengan kemampuan unggah dan tanpa kemampuan unggah. Perantara bergerak tanpa kemampuan unggah adalah perantara yang bergerak dan menyalurkan pesan ke perantara bergerak lain hingga pesan sampai pada perantara bergerak dengan kemampuan unggah, sehingga selanjutnya data dapat diunggah ke pusat data. Diharapkan mekanisme komunikasi ini dapat mengatasi permasalahan komunikasi pada sistem IoT.

2. PERANCANGAN

2.1 Arsitektur Sistem



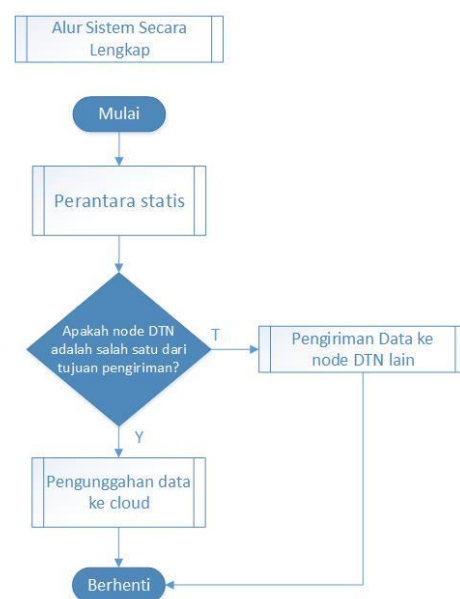
Gambar 1. Perancangan Arsitektur Sistem

Pada Gambar 1 dapat dilihat bahwa sistem terdiri dari 3 bagian yang masing-masing bagiannya memiliki komponen yang berbeda. Penjelasan dan fungsi dari masing-masing komponen adalah sebagai berikut:

- 1) *Sensor node* merupakan perangkat yang berfungsi untuk membaca sensor dan mengirimkan datanya ke broker menggunakan protokol MQTT (*publish*).

- 2) *Broker* merupakan perangkat yang berfungsi untuk menerima pesan dari *publisher*, mem-*filter* pesan yang masuk, dan meneruskan pesan kepada *subscriber* sesuai topik pesan melalui WiFi.
- 3) Perantara pada dasarnya merupakan perangkat *mobile* berbasis sistem operasi Android yang terpasang perangkat lunak IBR-DTN. Perangkat IBR-DTN terdiri dari 2 perangkat lunak, yaitu IBR-DTN *service* dan IBR-DTN *client*. Berikut adalah jenis-jenis perantara berdasarkan fungsinya:
 - a) Perantara statis merupakan perangkat *mobile* yang berfungsi untuk menerima pesan dari broker menggunakan protokol MQTT (*subscribe*), meneruskannya ke perantara bergerak melalui *Access Point* (AP) menggunakan protokol DTN melalui WiFi.
 - b) Perantara bergerak tanpa kemampuan unggah merupakan perangkat *mobile* yang berfungsi untuk menerima pesan dari perantara statis atau perantara bergerak lainnya dan meneruskannya ke perantara bergerak lainnya melalui *Access Point* (AP) menggunakan protokol DTN melalui WiFi. Khusus pada perantara jenis ini tidak dibutuhkan IBR-DTN *client*.
 - c) Perantara bergerak dengan kemampuan unggah merupakan perangkat *mobile* yang berfungsi untuk menerima pesan dari perantara statis atau perantara bergerak lainnya dan mengunggahnya ke pusat data menggunakan protokol HTTP ketika tersedia jaringan seluler dan menyimpannya ketika tidak ada jaringan seluler untuk selanjutnya dikirim setelah jaringan seluler tersedia kembali.
- 4) *Access point* merupakan perangkat yang berfungsi untuk menjembatani koneksi antar-perantara agar bisa saling bertukar pesan.
- 5) Pusat data merupakan server yang memiliki 2 layanan, yaitu *webapp* untuk menampilkan data dan *webservice* untuk jalur pengiriman data

- 1) Jalankan alur kerja perantara statis, yaitu membentuk suatu mekanisme pengiriman data dari *sensor node* menuju perantara bergerak
- 2) Ketika data sampai pada perantara bergerak, periksa apakah perantara bergerak tersebut adalah salah satu dari perantara bergerak dengan kemampuan unggah yang telah didaftarkan pada perantara statis sebagai tujuan pengiriman data.
- 3) Jika benar, maka unggah data ke pusat data melalui Webservice API.
- 4) Jika salah, maka kirim data ke perantara bergerak lain



Gambar 2 Diagram Alur Sistem Secara Lengkap

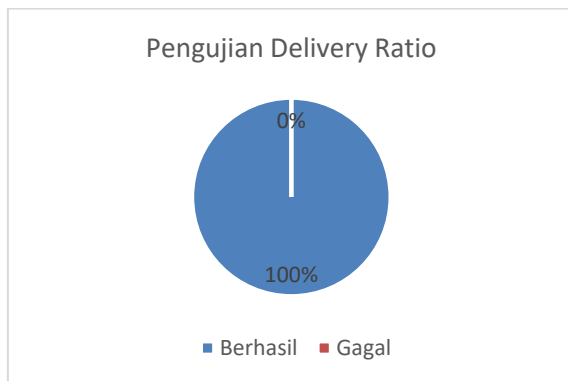
3. HASIL PENGUJIAN

Pengujian dilakukan untuk mengetahui performa dari mekanisme komunikasi yang dirancang. beberapa parameter performa yang diuji adalah *delivery ratio*, *delivery latency*, dan *delivery cost*.

2.2 Alur Kerja Sistem

Penjelasan dari Gambar 2 yang menggambarkan alur kerja sistem secara lengkap adalah sebagai berikut:

3.1 Pengujian Delivery Ratio

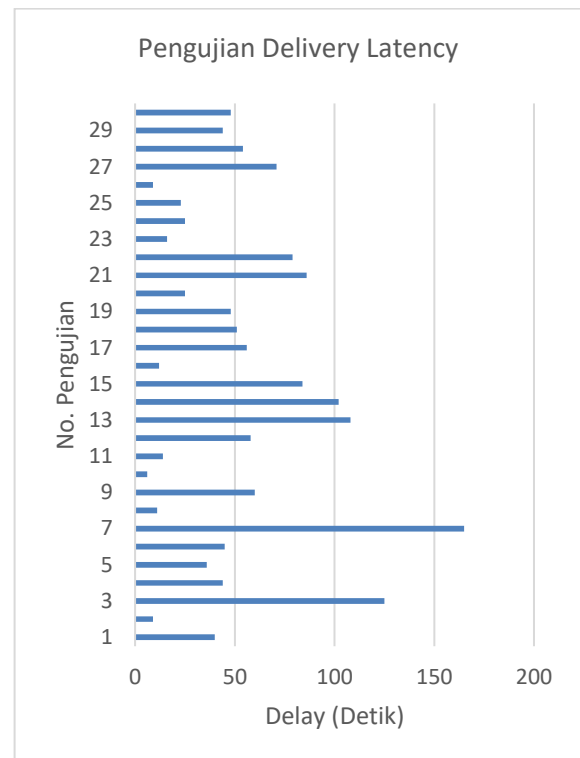


Gambar 3 Grafik Hasil Pengujian *Delivery Ratio*

Berdasarkan hasil pengujian *delivery ratio* yang telah dilakukan, pada Gambar 3 dapat dilihat bahwa performa *delivery ratio* yang dihasilkan dapat dikatakan baik walaupun dalam kondisi ketersediaan jalur yang intermiten dari *sensor node* ke pusat data. Pada saat pengujian, jalur komunikasi dari *sensor node* ke pusat data diputus. *Sensor node* hanya terhubung ke sebuah perantara bergerak tanpa kemampuan unggah. Setelah pengiriman pesan dilakukan, kemudian jalur dari *sensor node* ke perantara bergerak tanpa kemampuan unggah diputus dan perantara tersebut disambungkan ke pusat data melalui sebuah perantara bergerak dengan kemampuan unggah. Dari skenario tersebut, dilakukan 20 kali percobaan dan menghasilkan 20 kali keberhasilan.

Hasil yang didapat tersebut membuktikan bahwa pada mekanisme komunikasi ini, konsep *store-carry-forward* yang dimiliki DTN telah berfungsi dengan baik dan dapat diandalkan.

3.2 Pengujian Delivery Latency



Gambar 4 Grafik Hasil Pengujian *Delivery Latency*

Berdasarkan hasil pengujian *delivery latency* yang dilakukan, dapat dilihat pada Gambar 4 bahwa performa *delivery latency* yang dihasilkan dapat dikatakan baik karena *delay* pengiriman cukup rendah. Pada saat pengujian, *sensor node* mengirim pesan ke pusat data melalui 2 buah perantara dengan kemampuan unggah dengan jalur komunikasi yang stabil. Dari skenario tersebut, dilakukan 30 kali percobaan dan menghasilkan nilai *delay* tercepat 6 detik, terlama 2 menit 45 detik, dan rata-rata 51,8 detik. Hasil yang didapat tersebut membuktikan bahwa *delivery latency* yang dimiliki mekanisme komunikasi ini dapat diterima.

3.3 Pengujian Delivery Cost

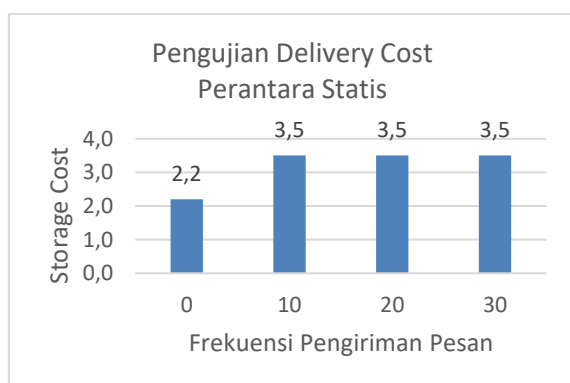
Berdasarkan hasil pengujian yang *delivery cost* yang dilakukan, dapat dilihat bahwa performa *delivery cost* yang dihasilkan dapat dikatakan baik karena tidak terlalu banyak mengonsumsi *resource* penyimpanan perangkat yang dipakai. Pada saat pengujian, *sensor node* mengirim pesan ke pusat data melalui perantara dengan kemampuan unggah. Selain itu, pada jaringan juga dihubungkan sebuah perantara tanpa kemampuan unggah.

Percobaan dilakukan 10 kali pada beberapa variabel uji, yaitu frekuensi pengiriman dan

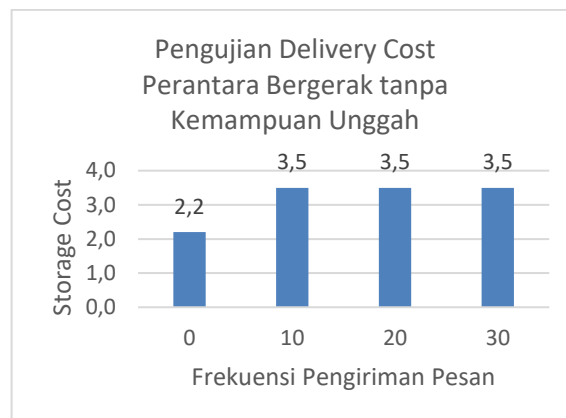
panjang pesan pada perantara statis, perantara tanpa kemampuan unggah, dan perantara dengan kemampuan unggah. Pada variabel frekuensi pengiriman dikirim 0, 10, 20, dan 30 pesan per 3 menit dengan panjang pesan 5 byte. Pada variabel panjang pesan dikirim 10 pesan per 3 menit dengan panjang 5, 10, 20, dan 40 byte.

Pada variabel panjang pesan didapatkan hasil pemakaian penyimpanan terbesar adalah 4,8 KB, yaitu ketika mengirim 10 pesan per 3 menit dengan panjang 40 byte pada ketiga perangkat yang diuji. Pada variabel frekuensi pengiriman didapatkan hasil pemakaian penyimpanan terbesar adalah 3,6 KB, yaitu ketika mengirim 30 pesan per 3 menit dengan panjang 5 byte pada ketiga perangkat yang diuji. Hasil yang didapat membuktikan bahwa pada beberapa kondisi pengujian yang dilakukan, kapasitas penyimpanan yang terpakai tidak membebani perangkat karena perangkat yang dipakai adalah perangkat *mobile Android* yang rata-rata memiliki kapasitas penyimpanan mencapai 8 GB atau 8.388.608 KB.

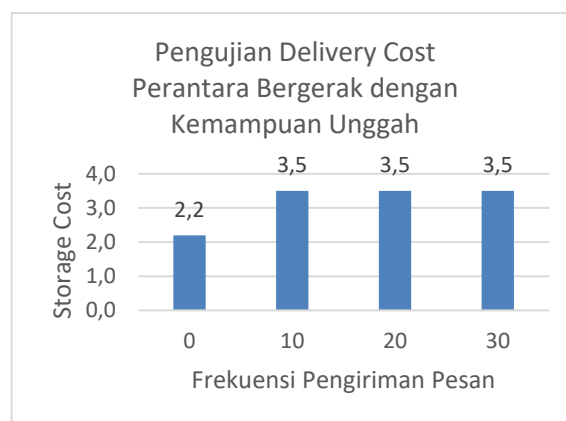
Sebagai penguat hasil pengujian, dilakukan prediksi jika nilai variabel baik frekuensi pengiriman maupun panjang pesan ditingkatkan untuk mengukur apakah mempunyai kemungkinan membebani perangkat. Prediksi dilakukan dengan mencari selisih hasil terbesar pada tiap kelompok variasi pengujian dari kedua variabel yang diuji.



Gambar 5 Hasil Pengujian Delivery Cost dengan Variasi Frekuensi Pengiriman Pesan pada Perantara Statis

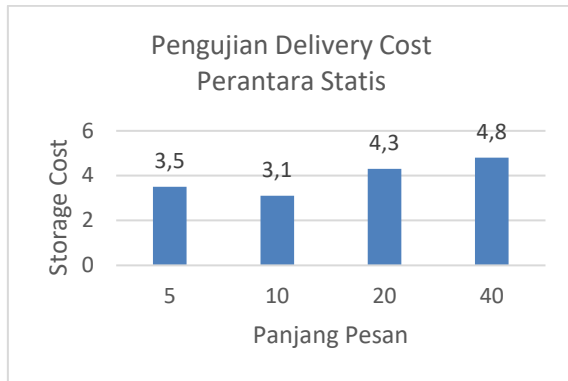


Gambar 6 Hasil Pengujian Delivery Cost dengan Variasi Frekuensi Pengiriman Pesan pada Perantara Bergerak tanpa Kemampuan Unggah

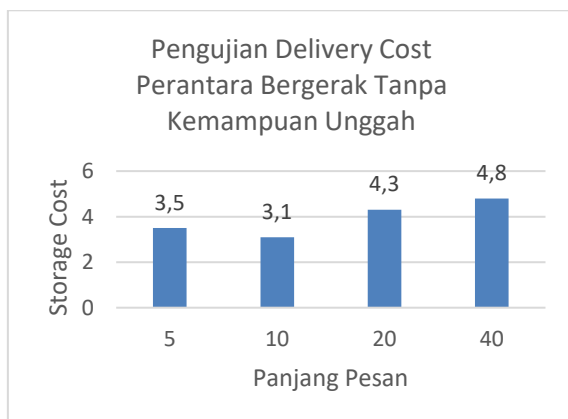


Gambar 7 Hasil Pengujian Delivery Cost dengan Variasi Frekuensi Pengiriman Pesan pada Perantara Bergerak dengan Kemampuan Unggah

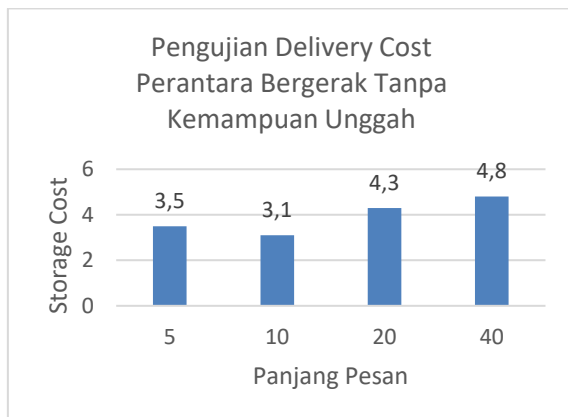
Pada perantara statis dengan variabel frekuensi pengiriman 0 kali, didapat hasil 2,2 KB. Pada frekuensi 10 kali, meningkat sebanyak 1,3 KB menjadi 3,5 KB. Pada frekuensi 20 kali, tidak terjadi peningkatan. Pada frekuensi 30 kali, tidak terjadi peningkatan. Pada perantara bergerak tanpa kemampuan unggah dengan variabel frekuensi pengiriman 0 kali, didapat hasil 2,2 KB. Pada frekuensi 10 kali, meningkat sebanyak 1,3 KB menjadi 3,5 KB. Pada frekuensi 20 kali, tidak terjadi peningkatan. Pada frekuensi 30 kali, tidak terjadi peningkatan. Pada perantara bergerak dengan kemampuan unggah dengan variabel frekuensi pengiriman 0 kali, didapat hasil 2,2 KB. Pada frekuensi 10 kali, meningkat sebanyak 1,3 KB menjadi 3,5 KB. Pada frekuensi 20 kali, tidak terjadi peningkatan. Pada frekuensi 30 kali, tidak terjadi peningkatan.



Gambar 8 Hasil Pengujian Delivery Cost dengan Variasi Panjang Pesan pada Perantara Statis



Gambar 9 Hasil Pengujian Delivery Cost dengan Variasi Panjang Pesan pada Perantara Bergerak tanpa Kemampuan Unggah



Gambar 10 Hasil Pengujian Delivery Cost dengan Variasi Panjang Pesan pada Perantara Bergerak dengan Kemampuan Unggah

Pada perantara statis dengan variabel panjang pesan 5 karakter, didapat hasil 3,5 KB. Pada panjang pesan 10 karakter, menurun sebanyak 0,4 KB menjadi 3,1 KB. Pada panjang pesan 20 karakter, meningkat sebanyak 1,2 KB menjadi 4,3 KB. Pada panjang pesan 40 karakter, meningkat sebanyak 0,5 KB menjadi 4,8 KB. Pada perantara bergerak tanpa kemampuan unggah dengan variabel panjang

pesan 5 karakter, didapat hasil 3,5 KB. Pada panjang pesan 10 karakter, menurun sebanyak 0,4 KB menjadi 3,1 KB. Pada panjang pesan 20 karakter, meningkat sebanyak 1,2 KB menjadi 4,3 KB. Pada panjang pesan 40 karakter, meningkat sebanyak 0,5 KB menjadi 4,8 KB. Pada perantara bergerak dengan kemampuan unggah dengan variabel panjang pesan 5 karakter, didapat hasil 3,5 KB. Pada panjang pesan 10 karakter, menurun sebanyak 0,4 KB menjadi 3,1 KB. Pada panjang pesan 20 karakter, meningkat sebanyak 1,2 KB menjadi 4,3 KB. Pada panjang pesan 40 karakter, meningkat sebanyak 0,5 KB menjadi 4,8 KB.

Hasil tersebut menunjukkan bahwa peningkatan nilai variabel tidak terlalu beresiko menimbulkan beban yang berarti pada perangkat atau dengan kata lain, perangkat akan dapat menampung beban tersebut dalam jangka waktu yang lama.

4. KESIMPULAN

- 1) Konsep *delay-tolerant network* dapat diimplementasikan pada mekanisme komunikasi antar-perangkat *mobile* Android dengan memanfaatkan IBR-DTN *service* dan IBR-DTN *client*. IBR-DTN *service* berkomunikasi dengan IBR-DTN *service* pada perangkat lain melalui WiFi.
- 2) Mekanisme komunikasi antara *sensor node* dan perantara bergerak dilakukan menggunakan perantara statis. Perantara statis adalah perangkat yang berperan sebagai *subscriber* yang akan menerima *publish* pesan dari *sensor node* pada jaringan sensor berbasis MQTT. Selain itu, perantara statis juga berperan sebagai *node* DTN yang akan meneruskan pesan yang diterimanya dari *sensor node* ke perantara bergerak menggunakan konsep DTN.
- 3) Mekanisme komunikasi antara perangkat bergerak ke pusat data dilakukan menggunakan protokol HTTP. Perangkat bergerak dengan kemampuan unggah berperan sebagai HTTP *client* ketika berkomunikasi dengan pusat data. Perangkat bergerak dengan kemampuan unggah mengirim data ke pusat data dengan cara mengakses Webservice API yang telah disediakan melalui jaringan seluler.
- 4) Berdasarkan hasil pengujian, mekanisme komunikasi yang dirancang memiliki performa *delivery ratio* yang baik yaitu mencapai 100%. Mekanisme komunikasi

yang dirancang juga memiliki performa *delivery latency* yang baik yaitu rata-rata 51,8 detik. Selain itu berdasarkan hasil pengujian, perangkat lunak yang mengimplementasikan mekanisme komunikasi yang dirancang juga diprediksi tidak akan membebani perangkat yang digunakan dalam jangka waktu yang lama.

DAFTAR PUSTAKA

- DE Munich, D. W. J. T., 2016. IoT 2020: Smart and secure IoT platform. *International Electrotechnical Commission*, pp. 1-181.
- Fielding, R., Irvine, U., Gettys, J. & Masinter, L., 1999. Hypertext Transfer Protocol -- HTTP/1.1. *RFC 2616*, p. 7/114.
- Hauke Petersen, E. B. M. W., 2014. Interoperable Services on Constrained Devices in the Internet of Things. pp. 1-4.
- Jayavardhana Gubbi, R. B. S. M. M. P., 2013. Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions. pp. 1-28.
- Lindgren, A., Doria, A., Davies, E. & Grasic, S., 2012. *Probabilistic Routing Protocol for Intermittently Connected Networks*. swedia: PROPHET.
- M2M, 2013. *MQTT*. s.l.:M2M.
- Naik, N., 2017. Choice of Effective Messaging Protocols for IoT Systems: MQTT, CoAP, AMQP and HTTP. *IEEE*, pp. 1-7.
- Phearin, S., Seryvuth, T. & Keecheon, K., 2013. PROPHET Routing Protocol based on Neighbor Node Distance Using a Community Mobility Model in Delay Tolerant Networks. *IEEE*, pp. 1-8.