

# Peningkatan *Throughput* pada LTE-wifi Menggunakan Balia Multipath-TCP pada *Software Defined Wireless Network (SDWN)*

1<sup>st</sup> Priyagung Elza Yogitama  
*Teknologi Informasi*  
*Fakultas Informatika*  
Bandung, Indonesia  
priyagungyogi@students.telkomuni-  
versity.ac.id

2<sup>nd</sup> Endro Ariyanto  
*Teknologi Informasi*  
*Fakultas Informatika*  
Bandung, Indonesia  
endroa@telkomuniversity.ac.id

3<sup>rd</sup> Hilal Hudan Nuha  
*Teknologi Informasi*  
*Fakultas Informatika*  
Bandung, Indonesia  
hilalnuha@telkomuniversity.ac.id

## Abstrak

Jaringan Long-term Evolution (LTE) mendominasi ada jaringan telekomunikasi Indonesia seiring perkembangan zaman, menurut Kominfo perkembangan hingga 90%. Perangkat yang menunjang jaringan nirkabel seperti teknologi Wi-Fi dengan standar 802.11 meningkat dari jumlah pengguna, base station dan trafik. Penggunaan Multipath Transport Control Protocol (MP-TCP) dapat meningkatkan throughput dengan lebih baik dari single path Transmission Control Protocol (TCP). Perangkat yang sesuai dengan MPTCP hal ini memungkinkan untuk memaksimalkan utilitas jaringan throughput hingga 60% dengan menggunakan sejumlah interfaces jaringan. Penelitian ini tentang performansi algoritma congestion control Balance linked Adaption (BALIA) dan Linked Increase Algorithm (LIA) pada jaringan Long-term Evolution (LTE) dan wifi pada mobile device dengan topologi sederhana yang menggunakan MP-TCP congestion control. Parameter yang dianalisis adalah throughput. Mininet-wifi dapat mensimulasikan jaringan SDWN. Dari hasil pengujian terlihat bahwa BALIA dapat meningkatkan throughput sebesar 206% atau dua kali lebih besar dibanding LIA.

**Kata kunci:** multipath, multihoming, performansi, LTE, 802.11

## Abstract

Long-term Evolution (LTE) networks dominate the Indonesian telecommunications network over the times, according to the Communications and Information Technology, the development is up to 90%. devices that support wireless networks such as Wi-Fi technology with the 802.11 standard increase from the number of users, base stations and traffic. The use of Multipath Transport Control Protocol (MP-TCP) can increase throughput better than single path Transmission Control Protocol (TCP).

*MPTCP compliant devices make it possible to maximize network utility by up to 60% by using this interface. This study discusses the scalability and performance of congestion control algorithms balance linked increase (BALIA) and linked increase (LIA) on Long-term Evolution (LTE) and wifi networks on mobile devices with simple topology using MP-TCP congestion control. The parameter analyzed is throughput. Mininet-wifi can simulate SDWN network. From the test results, it can be seen that BALIA can increase throughput by 206% or two times greater than LIA.*

**Keywords:** multipath, multihoming, performace, LTE, 802.11

## I. PENDAHULUAN

### a. Latar Belakang

Perkembangan teknologi nirkabel seperti perangkat *smartphone* dan laptop memungkinkan memiliki multiple *interfaces* jaringan, tantangan seperti banyaknya pengguna perangkat pintar yang diperkirakan mengalami lonjakan 5,3 miliar pengguna (66 persen dari populasi) pada tahun 2023[8]. Oleh karena itu juga akan mempengaruhi pada jaringan LTE. Jaringan LTE mengalami peningkatan yang juga akan berakibat meningkatnya *base station*[6]. *Multipath-TCP (MPTCP)* memperluas *layer transport* dengan tambahan fungsional di atas *Transport Control Protocol (TCP)*. Koneksi MPTCP yang berada di bawahnya terdiri dari beberapa koneksi TCP, sementara pada *upper layer* MPTCP hanya terdiri satu koneksi. Pengaturan *traffic* yang terjadi ketika adanya *congestion* berlaku pada MPTCP. *Congestion control* yang tersedia pada MPTCP kernel antara lain *Linked Increase Algorithm (LIA)* dan *Balance Increase Algorithm (BALIA)*.

*Congestion control LIA* memiliki keunggulan dengan meningkatkan *throughput* dengan baik akan tetapi tidak memenuhi faktor keseimbangan pada *subflow* yang tersedia. LIA terkadang bertindak terlalu agresif terhadap pengguna *single path TCP* tanpa ada keuntungan bagi pengguna MPTCP. Untuk mengatasi

permasalahan pada LIA, BALIA hadir untuk memenuhi tujuan dari MPTCP[5]. Menyamaratakan pengiriman data pada setiap *subflow* juga berguna untuk meningkatkan *throughput*. Teknologi yang membantu untuk mengurangi kompleksitas arsitektur jaringan konvensional ialah *Software-defined wireless network (SDWN)*.

Tujuan penelitian untuk menganalisis perbandingan performansi dari algoritma BALIA dan LIA pada SDWN. *Congestion control* pada MPTCP telah banyak dikembangkan. MPTCP pada awalnya menggunakan *congestion control LIA* yang merupakan adopsi dari standar *congestion control TCP*. Sedangkan BALIA pengembangan lebih lanjut dari MPTCP yang mencakup aspek *responsiveness* untuk meningkatkan *throughput* dengan pembaruan nilai *window* dari setiap *subflow*. BALIA dan LIA mempunyai perilaku terhadap *packet loss* sebagai *congestion signal* dengan penanganan *congestion window (CWND)* yang berbeda dimana BALIA mempertimbangkan nilai RTT pada setiap *subflow* sehingga lebih responsif. Integrasi jaringan LTE dan SDWN agar menyederhanakan arsitektur secara fleksibel dan mengoptimalkan kebutuhan sumber daya. MPTCP secara efektif menggunakan ketersediaan *bandwidth* dibandingkan dengan penggunaan TCP biasa[10].

Data yang dikirim akan melalui semua *subflow* yang terhubung secara bersamaan. *Mininet-wifi* digunakan sebagai simulator pada penelitian ini karena mendukung *protocol* wifi 802.11 dan nirkabel.

### a. Topik dan Batasannya

Topik yang diangkat meliputi analisis perbandingan performansi dari algoritma BALIA dan LIA pada SDWN. Berdasarkan topik tersebut ada beberapa batasan masalah antara lain:

- pengujian menggunakan dua virtual mesin salah satu bertugas sebagai *data plane* dan *controller*
- topologi dibangun menggunakan *mininet-wifi*
- *controller* menggunakan *Ryu*
- tidak membahas keamanan pada jaringan MPTCP
- menggunakan standar wireless 802.11
- tidak membahas parameter seperti gain dan pola radiasi

### b. Tujuan

Untuk menganalisis peningkatan *throughput* ketika menggunakan *congestion control* antara BALIA dan LIA. Simulasi dilakukan menggunakan *mininet-wifi*. Protokol yang digunakan menggunakan standar wifi 802.11.

### c. Organisasi Tulisan

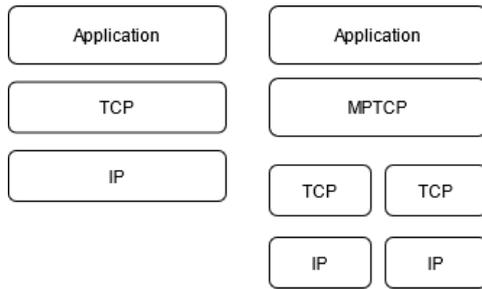
Adapun pembagian penulisan pada penelitian ini, bab pertama menjelaskan abstrak, latar belakang, rumusan masalah, bab dua menjelaskan studi terkait, skenario percobaan pada bab tiga, bab empat hasil percobaan, sedangkan bab 5 kesimpulan.

### d. Studi Terkait

### e. Multipath-TCP

Multipath-TCP adalah upaya pengembangan dari TCP biasa dengan menggunakan sistem *load balancing* yang terdiri dari dua atau lebih *interfaces* jaringan, setiap *interfaces* jaringan disebut *subflow* yang berbeda *bandwidth* dan adanya *congestion control*, setiap *interfaces* yang terhubung harus menjalankan koneksi *transport* yang dibangun kepada penerima[4].

Kelebihan ketika menggunakan MPTCP antara lain seperti, meningkatkan utilitas jaringan seperti *throughput*, *latency*, menghindari kegagalan pada salah satu *interfaces* jaringan, tidak membebani kinerja dari *interfaces* yang terhubung, perbandingan gambar TCP dan MPTCP[5].



**Gambar 1. Tradisional TCP dan MPTCP**

Gambar 1 membandingkan antara struktur TCP biasa dengan MPTCP. TCP biasa hanya terdiri dari satu jenis *internet protocol* (IP) yang terhubung. MPTCP menggunakan lebih dari satu koneksi TCP atau disebut *subflow*. Subflow pada MPTCP bertugas layaknya koneksi TCP biasa yang digunakan untuk menghubungkan antar *host* atau perangkat

f. Linked Increase Algorithm

*Linked Increase Algorithm* (LIA) algoritma *congestion control* yang berjalan secara dinamis berdasarkan *congestion window* agar tidak melewati *subflow* yang terjadi *congestion* [2][4]. Jika setiap *subflow* mengirimkan data yang lebih maka akan mengirim melalui jalur yang paling tidak padat, *traffikan* menjauhi jalur yang padat [9]. LIA tidak menyediakan efisiensi terhadap *load balancing*[10]. LIA memiliki kekurangan terhadap *unfairness* dan *unresponsive* terhadap perubahan jaringan [4]. Berikut perhitungan pada LIA :

g. Balance Linked Algorithm

*Balance Linked Algorithm* (BALIA) mendukung keseimbangan antar *interfaces* agar tidak membebani kinerja disalah satu *interfaces*, selain itu BALIA juga mendukung *responsiveness* dan *tcp-friendliness* karena LIA tidak mendukung *balance link* yang tidak cukup baik, BALIA mengadopsi algoritma *congestion TCP Reno*[2][4]. Berikut perilaku peningkatan ataupun penurunan *congestion window (CWND)* pada BALIA:

Persamaan 4 dan 5 menunjukkan bahwa apabila terdapat hanya satu *subflow* maka  $\alpha$  bernilai 1,

sehingga penambahan atau pengurangan nilai *cwnd* sama seperti algoritma TCP Reno.

h. *Software Defined Wireless Network*

*Software Defined Wireless Network (SDWN)* menyediakan pengelolaan pemrograman jaringan dari *base station* seperti *Access Point (AP)* untuk menangani paket yang dikirim melalui *controller*. Pewarisan dari *software defined network (SDN)* dengan pemisahan *control plane* dan *data plane* yang memperbolehkan administrasi jaringan memberi perilaku terhadap jaringan serta menyederhanakan arsitektur [3][7].

i. *Mininet-wifi*

*Mininet-wifi* sebagai ekstensi dari emulator jaringan SDN yang mendukung teknologi *wireless* dengan standar 802.11 penambahan virtualisasi *Wifi stations* dan *access point(AP)* berdasarkan linux driver 80211\_hwsim[11]. Sebagian besar driver linux yang berkembang sudah mendukung hwsim80211/softmac fitur *wireless NIC* mendukung pengiriman paket jaringan nirkabel [1].

j. *Ryu*

*Ryu* merupakan *framework controller* yang terdiri dari beberapa komponen pendukung jaringan serta dirancang untuk meningkatkan adaptasi jaringan terhadap simplisitas, kecepatan jaringan. *Ryu* juga dengan membuat mudah dalam manajemen serta adaptasi terhadap trafik [12].

## II. KAJIAN TEORI

### a. Topologi Sistem

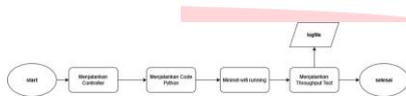
Desain topologi yang dibangun dengan dua virtual mesin dengan sistem operasi ubuntu kernel MPTCP, salah satu virtual mesin sebagai *controller* dan *control plane*. *Mininet-wifi* bertugas sebagai data plane sebagai simulator jaringan nirkabel dengan penambahan komponen 802\_11hswim. *Controller Ryu* sebagai *control plane* yang akan menjalankan *L2 learning switch* pada *access point* yang terhubung. Simulasi yang menggunakan jaringan heterogen LTE dan Wifi. Simulasi jaringan LTE dengan memodifikasi *interfaces* seperti *bandwidth*, *loss* serta *latency* menggunakan linux *traffic control (TC)*. *Latency* pada jaringan LTE lebih panjang dibanding dengan jaringan wifi[11]. Modifikasi *interfaces* jaringan LTE dengan *bandwidth* 3Mb, *latency*

75ms dan *loss* sebesar 1%. Jaringan wifi dimodifikasi dengan kapasitas *bandwidth* 5Mb, *latency* 10ms serta *loss* 2%.

Gambar 2. Topologi jaringan

Pada gambar 2 terdapat host MPTCP client dengan terhubung base station dan access point. Base station dan access point terhubung dengan controller yang mereduksi kongesti dan menambah fungsi traffic management[11]. Router terhubung dengan MPTCP server yang digunakan sebagai destinasi tujuan paket.

a. Skema Pengujian



Gambar 3. Skema Pengujian

Skenario pengujian dibagi menjadi dua dengan penggunaan metode LIA dan BALIA, dengan pengukuran parameter *throughput client* dan *background traffic*.

*Throughput* digunakan sebagai parameter data yang dikirim dalam satuan waktu tertentu mencapai tujuan. Dalam penelitian menggunakan tools *iperf* sebagai pengukur *throughput*. Host client bertindak sebagai pengirim lalu MPTCP server bertindak sebagai penerima.

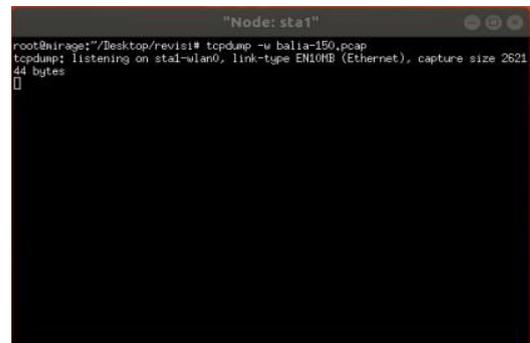
*Background traffic* digunakan sebagai penambah beban pada jalur agar memberi dampak congestion pada traffic yang terkirim oleh *host client*.

Analisis performansi untuk perbandingan *congestion control* LIA dan BALIA pada jaringan Wifi dan LTE seperti yang ada pada gambar 3, pada alur skenario pengujian, yang pertama menjalankan *controller Ryu*, lalu mengeksekusi program python sekaligus menjalankan konfigurasi *congestion control* MPTCP pada *script* python yang terhubung dengan *controller ryu* yang sudah dinyalakan, setelah program berjalan maka menjalankan *throughput test* dengan *iperf* pada *xterm* pada *host client* dan *server* sekaligus dengan dibuatnya *log file* hasil pengujian. Data yang dihasilkan lalu dibersihkan dengan menggunakan utilisasi text processing menggunakan *bash*.

*Host client* terhubung dengan dua *access point*, yang memiliki jaringan heterogen atau yang memiliki subnet yang berbeda, masing masing AP terhubung dengan router lalu menghubungkan dengan MPTCP *server*.

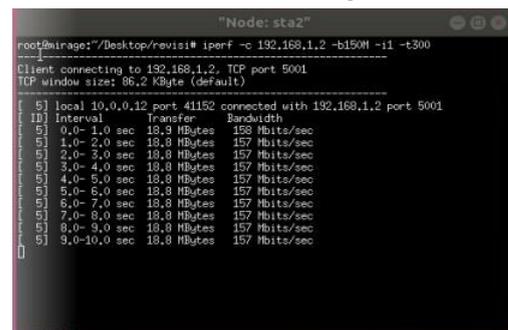
a. Evaluasi

Berikut merupakan perintah untuk menangkap data dengan menggunakan tool *tcpdump* pada sisi *client*.



Gambar 4. Perintah *tcpdump* pada sisi client

*Tcpdump* digunakan sebagai penangkap data pada sisi *client* (sta1). Setelah didapatkan data menggunakan *tcpdump* maka diperoleh file *pcap*. File *pcap* dibuka dengan *wireshark*, filter dengan *tcp* lalu *export* data dengan format *csv* diambil nilai besaran *window* dan dibuat grafik *cwnd*.

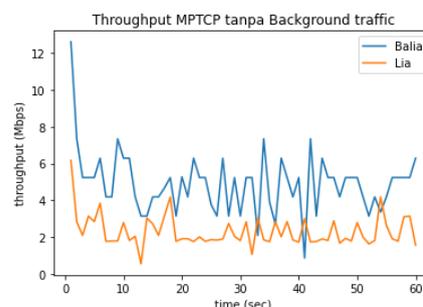


Gambar 5. Perintah *iperf* untuk *background traffic*

*Iperf* pada terminal host generator (sta2) digunakan untuk pemberian beban *traffic*

4.1 Hasil Pengujian *Throughput*

Di bawah ini merupakan grafik pengujian *throughput* setelah beberapa kali percobaan *iperf* tanpa menggunakan beban *traffic* yang berjalan pada *background*:



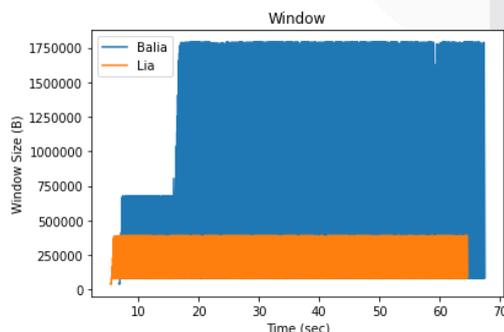
Gambar 6. Grafik MPTCP tanpa adanya *background traffic*

### 4.3 Analisis Hasil Pengujian

Berdasarkan hasil pengujian *throughput* didapatkan hasil bahwa LIA dengan rata-rata 2,31 Mbps sedangkan BALIA didapatkan rata-rata *throughput* 4,90 Mbps pada tabel 1 tanpa adanya *background traffic* atau mengalami peningkatan sekitar 212% dari penggunaan *congestion control* BALIA. Pada gambar 6 BALIA menghasilkan *throughput* yang dominan lebih baik daripada LIA. Pada detik ke 41 mengalami penurunan *throughput* tetapi setelah itu mengalami peningkatan yang lebih besar dibanding beberapa waktu sebelumnya.

Berdasarkan hasil pengujian dengan *background traffic* BALIA dapat meningkatkan *throughput* sebesar 206%. Rata-rata *throughput* yang didapatkan oleh BALIA 4,55 Mbps dan LIA 2,20 Mbps. Pada gambar 7, grafik *throughput* BALIA lebih bagus pada setiap detiknya. Sedangkan grafik LIA ketika ada *background traffic* cenderung mendapatkan hasil yang buruk. Pada gambar 7, *Background traffic* terhadap jaringan memberi dampak terhadap *congestion control* dengan penanganan yang berbeda seperti peningkatan *cwnd* ketika mendapatkan ACK dan penurunan *cwnd* ketika mendeteksi *loss*.

*Background traffic* yang semakin tinggi membuat jalur mengalami *congestion* dengan jumlah *throughput* yang dikirim, karena data yang dikirim hilang ketika menuju tujuan. BALIA dengan jumlah *background traffic* yang berbeda seperti pada tabel 3 masih lebih baik dari LIA terhadap penanganan *congestion*. Paket yang dikirim ketika menggunakan BALIA lebih baik terhadap penanganan *congestion avoidance* karena BALIA mendeteksi *congestion* pada *subflow* BALIA akan melakukan perpindahan ke jalur yang minim *congestion*. Perpindahan bisa terjadi dari *subflow* LTE ataupun *subflow* Wifi. Sedangkan LIA bekerja dari penggunaan pengaturan kongesti pada TCP biasa, sehingga lebih rendah dibanding BALIA.



Gambar 11. Perbandingan Grafik peningkatan *window* pada BALIA dan LIA

Gambar 11 menunjukkan peningkatan ukuran *window* dari MPTCP LIA dan BALIA selama pengujian. Dari gambar tersebut terlihat bahwa

ukuran *window* pada LIA cenderung stabil dibandingkan dengan BALIA. Namun BALIA menghasilkan ukuran *window* yang lebih tinggi daripada LIA, sehingga *throughput* yang dihasilkan lebih banyak seperti yang ditunjukkan pada gambar 6 dan 7.

## V. KESIMPULAN

Berdasarkan hasil yang dilakukan pada pengujian, maka dapat disimpulkan bahwa:

1. *Mininet-wifi* mendukung simulasi jaringan SDWN dibantu dengan protokol 80211\_hwsim pada linux wireless driver serta desain jaringan nirkabel.
2. BALIA lebih baik dalam hal penanganan terhadap *congestion* dengan cara menaikkan ukuran *window* sehingga *throughput* yang dihasilkan lebih tinggi yaitu sebesar 206% atau dua kali lebih besar.

### b. Saran

Saran yang dapat diberikan untuk pengembangan berikutnya adalah penambahan parameter jumlah host atau *loss* jaringan yang berbeda-beda.

## REFERENSI

- [1] R. R. Fontes, S. Afzal, S. H. B. Brito, M. A. S. Santos and C. E. Rothenberg, "Mininet-WiFi: Emulating software-defined wireless networks," 2015 11th International Conference on Network and Service Management (CNSM), Barcelona, 2015, pp. 384-389, doi: 10.1109/CNSM.2015.7367387
- [2] Y. Thomas, M. Karaliopoulos, G. Xylomenos and G. C. Polyzos, "Low Latency Friendliness for Multipath TCP," in IEEE/ACM Transactions on Networking, vol. 28, no. 1, pp. 248-261, Feb. 2020, doi: 10.1109/TNET.2019.2961759.
- [3] Fontes, Ramon & Esteve Rothenberg, Christian. (2016). Mininet-WiFi: A Platform for Hybrid Physical-Virtual Software-Defined Wireless Networking Research. 607-608. 10.1145/2934872.2959070.

- [4] Hijawi, Hamzah & Hamarsheh, Mohammad. (2016). Performance Analysis of Multi-Path TCP Network. *International journal of Computer Networks & Communications*. 8. 145-157. 10.5121/ijnc.2016.8213.
- [5] Kato, Toshihiko & Adhikari, Diwakar & Yamamoto, Ryo & Ohzahata, Satoshi & Suzuki, Nobuo. (2019). EXPERIMENTAL ANALYSIS OF MPTCP CONGESTION CONTROL ALGORITHMS; LIA, OLIA AND BALIA. 135-142. 10.33965/tpmc2019\_201907L017
- [6] Costa-Requena, Jose. (2014). SDN integration in LTE mobile backhaul networks. 264-269. 10.1109/ICOIN.2014.6799479
- [7] Costanzo, S., Galluccio, L., Morabito, G., & Palazzo, S. (2012). Software Defined Wireless Networks: Unbridling SDNs. *2012 European Workshop on Software Defined Networking*, 1-6
- [8] "Cisco Annual Internet Report (2018–2023) White Paper." <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>
- [9] Wischik, Damon & Handley, Mark & Braun, Marcelo. (2008). The Resource Pooling Principle. *Computer Communication Review*. 38. 47-52. 10.1145/1452335.1452342.
- [10] R. Khalili, N. Gast, M. Popovic and J. Le Boudec, "MPTCP Is Not Pareto-Optimal: Performance Issues and a Possible Solution," in *IEEE/ACM Transactions on Networking*, vol. 21, no. 5, pp. 1651-1665, Oct. 2013, doi: 10.1109/TNET.2013.2274462.





[11] P. Du, Q. Zhao and M. Gerla, "A Software Defined Multi-Path Traffic Offloading System for Heterogeneous LTE-WiFi Networks," 2019 IEEE 20th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM), 2019, pp. 1-9, doi: 10.1109/WoWMoM.2019.8793045.

[12] sdxcentral studios. (2016, June 24). What is a ryu controller? — sdxcentral .com. sdx central. Retrieved February 11, 2022, from <https://www.sdxcentral.com/networking/sdn/definitions/what-is-ryu-controller/>

[13] Kimura, Bruno & Loureiro, Antonio. (2018). MPTCP Linux Kernel Congestion Controls.

