

## BAB III METODOLOGI PENELITIAN

### 3.1 Perancangan Skenario Simulasi

Perancangan dengan menggunakan sebuah sistem eNodeB. Sebuah sistem terdiri dari Arsitektur LTE yang menggunakan sebuah eNodeB dan sepuluh (10) UE. Sistem yang pertama akan dilakukan konfigurasi dengan menggunakan *Scheduling Throughput to Average* dan sistem kedua menggunakan *Proportional Fair*. Penggunaan sebuah eNodeB dilakukan karena pengamatan kinerja *scheduling* dititikberatkan pada QoS yang didapatkan setiap pengguna yang dipengaruhi oleh pengaturan *Packet Scheduling*. Untuk itu diperlukan perancangan parameter yang akan dikonfigurasi pada eNodeB menggunakan *software* simulasi NS3 versi 3.24.1 yang berjalan pada Sistem Operasi Ubuntu 14.04. Skema parameter pengaturan pada E-UTRAN ditunjukkan pada tabel 3.1

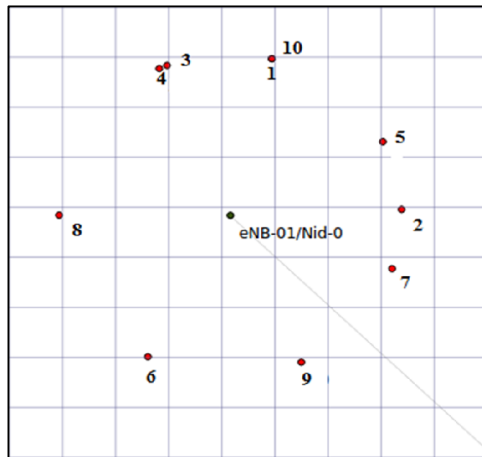
Tabel 3.1 Parameter Pengaturan pada E-UTRAN

<b>Parameter</b>	<b>Keterangan</b>
Waktu Simulasi	3 detik
eNB	1
Antena	<i>Omni Directional</i>
Jarak Terjauh UE	1000 m
Bandwidth	10 MHz
RB	50
Waktu <i>Scheduling</i>	1 ms
Struktur <i>Frame</i>	FDD
Jumlah UE	10
Jarak UE	Variatif
Trafik Data	UDP <i>Packet Size = Variative</i> Interval 20 ms
Frekuensi	1800 MHz

Berdasarkan parameter pada tabel 3.1 penulis menggunakan dua skenario yang memiliki perbedaan yaitu parameter jarak antara UE dengan ENB. Berikut adalah beberapa skenario yang akan digunakan dalam melakukan pengujian menggunakan *software* NS-3.

1. Skenario Satu

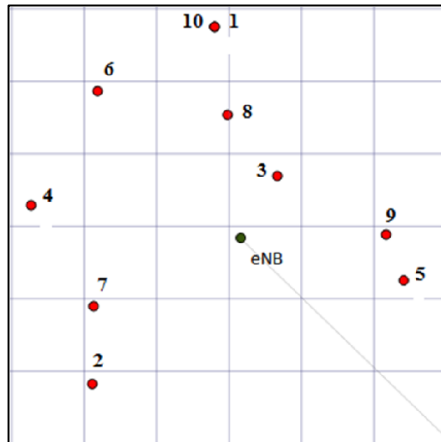
Skenario satu yang digunakan adalah dengan mengubah nilai jarak semua UE dengan menggunakan paket dengan ukuran variatif yaitu 100 Bytes, 500 Bytes, 900 Bytes, 1300 Bytes dan 1500 Bytes dengan interval 20 ms. Perubahan nilai jarak yang dilakukan berlaku sama untuk setiap UE. Jarak yang digunakan menggunakan variasi 500 meter dan 1000 meter. Pengujian ini bertujuan untuk melihat pengaruh dari perubahan jarak dan pengalokasian *radio resource* terhadap setiap UE dalam keadaan penerimaan daya yang sama. Titik persebaran UE pada skenario satu ditunjukkan oleh gambar 3.1.



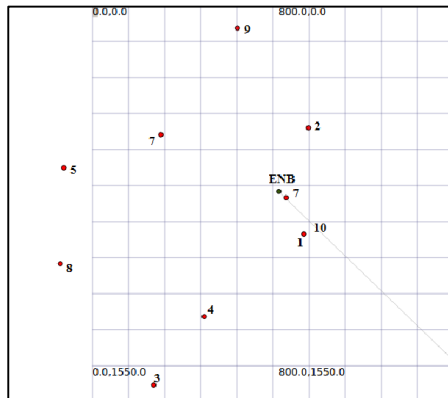
Gambar 3.1 Persebaran UE pada Skenario Satu

2. Skenario Dua.

Skenario dua yang digunakan adalah dengan membedakan jarak semua UE dari ENodeB. Metode penempatan UE acak dari jarak 0 meter hingga 500 meter dan jarak 0 meter hingga 1000 meter. Pengujian ini bertujuan untuk melihat pengaruh dari perbedaan jarak di setiap UE. Titik persebaran UE pada skenario jarak 0 – 500 meter ditunjukkan pada gambar 3.2 dan titik persebaran UE pada skenario jarak 0-1000 meter ditunjukkan pada gambar 3.3



Gambar 3.2 Persebaran UE Jarak 0 – 500 meter



Gambar 3.3 Persebaran UE Jarak 0 – 1000 meter

### 3.2 Parameter Analisis

Pada skenario simulasi *packet scheduling* ini, performa dihitung berdasarkan analisis parameter berikut ini :

#### 1. Throughput

*Throughput* adalah jumlah paket data yang sampai ke tujuan dalam satuan waktu per detik

$$x = \frac{\sum_{i=0}^n RXi}{t} \dots\dots\dots(3.1)$$

Dimana x merupakan nilai *throughput*, i merupakan pengguna, RX merupakan bit data yang diterima, n merupakan jumlah pengguna dan t adalah waktu simulasi

2. *Packet loss ratio*

*Packet loss ratio* adalah nilai presentase data yang gagal dikirimkan selama proses pengiriman berlangsung.

$$PLR = 1 - \frac{\sum_{i=0}^n RX_i}{\sum_{i=0}^n TX_i} \dots\dots\dots(3.2)$$

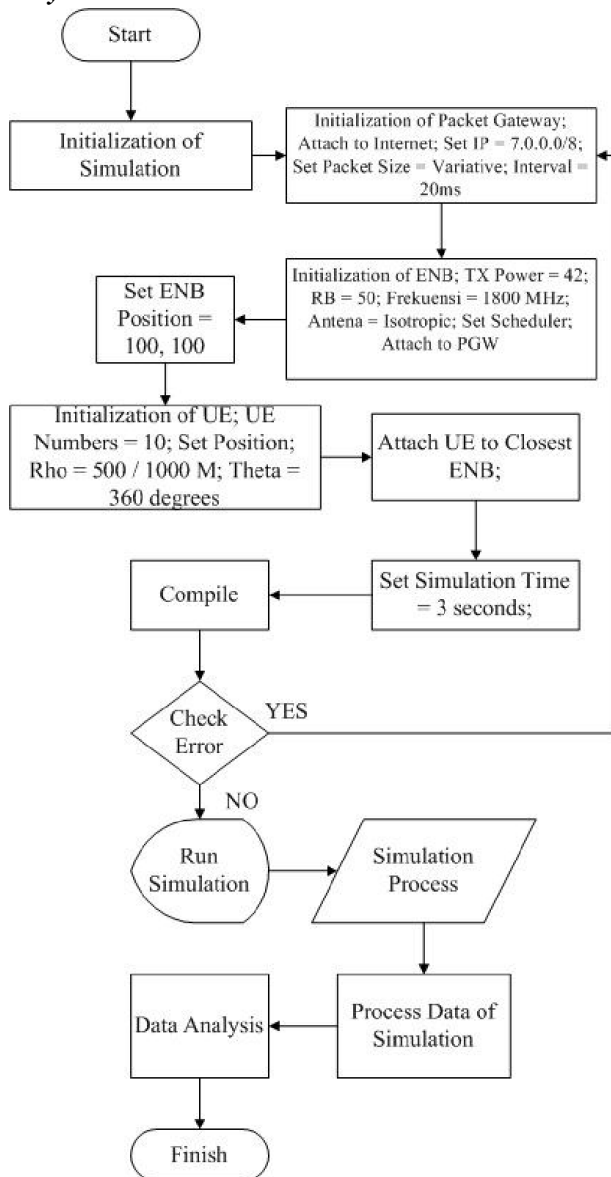
Dimana *i* adalah pengguna, *n* adalah jumlah pengguna, *RX* merupakan data yang berhasil diterima dan *TX* adalah data yang dikirimkan

3. *Delay*

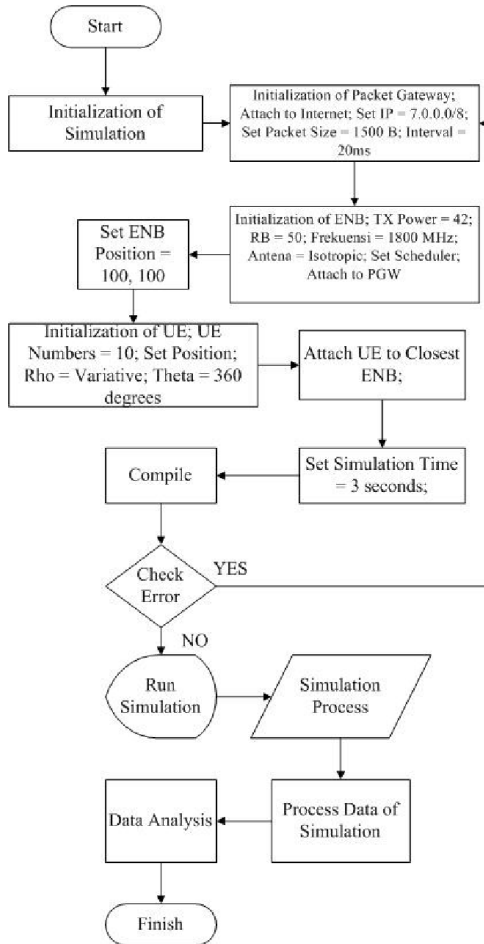
*Delay* adalah selisih waktu yang dibutuhkan saat data dikirimkan hingga data diterima.

4. *Jitter* adalah nilai variasi *delay* atau waktu respon dalam satuan *millisecond* (ms). *Jitter* merupakan indikator dari kestabilan sebuah jaringan. Nilai *jitter* yang semakin besar merupakan indikator bahwa *delay* yang terjadi dalam jaringan mengalami fluktuasi.

### 3.3 Flowchart System



Gambar 3.4 Flowchart System pada Skenario Satu



Gambar 3.5 *Flowchart System* pada Skenario Dua

Berdasarkan gambar 3.1 dan gambar 3.2 perancangan model simulasi digunakan untuk menentukan proses berjalannya simulasi yang menggunakan *software* NS3.24.1 yang menggunakan bahasa pemrograman C++ dikonfigurasi menggunakan langkah – langkah berikut ini :

1. Perancangan simulasi dimulai dari proses inialisasi simulasi, yaitu proses instalasi *software* NS3 versi 3.24.1 pada sistem operasi Ubuntu 14.04 Thrusty Tahr.

2. Berdasarkan pengaturan pada konfigurasi program NS-3, deklarasi pertama yang dikonfigurasi adalah pada sisi UE. Pengaturan yang dilakukan dengan mengatur jumlah UE (UENum). UENum adalah deklarasi yang digunakan untuk menentukan jumlah UE yang akan digunakan yaitu sebanyak 10 UE dengan posisi yang diatur dengan variabel sudut (theta) dan jarak (rho). Sudut yang digunakan adalah dengan sistem acak dari 0 derajat hingga 360 derajat. Pada skenario satu untuk jarak yang digunakan adalah tetap yaitu 500 meter dan 1000 meter sehingga rho diatur sebesar 500 meter atau 1000 meter. Sementara pada skenario dua rho yang digunakan adalah dengan menggunakan angka Minimum dan Maksimum. Untuk jarak 0 – 500 meter digunakan angka min = 0 dan max = 500 meter.
3. Kemudian selanjutnya adalah deklarasi untuk sistem ENB pada program NS-3, pengaturan yang dilakukan adalah konfigurasi *transmitter power* (`ns3::LteEnbPhy::TxPower`) yaitu 42 mW. TxPower digunakan untuk menentukan kekuatan daya pancar yang digunakan pada ENodeB. Selanjutnya adalah deklarasi frekuensi pembawa (carrier) yaitu 1800 MHz ("Frequency", DoubleValue (1.8e9)) dengan *bandwidth* yang digunakan dideklarasikan dengan jumlah RB ("DlBandwidth", UIntegerValue (50)) dikarenakan dengan lebar bandwidth 10 MHz terdapat RB dengan jumlah 50.
4. Pengaturan antena dengan menggunakan jenis antena yang dapat memancarkan daya ke segala arah yaitu jenis antena *isotropic* ("ns3::IsotropicAntennaModel"). Antena isotropic sering disebut sebagai antena Omni Directional. Antena jenis ini memancarkan gelombang elektromagnetik dengan sudut 360 derajat.
5. Selanjutnya adalah pengaturan *scheduling* yang digunakan dengan deklarasi pada program NS-3. Pada skenario *scheduling Throughput to Average*, *syntax* yang digunakan adalah (SetSchedulerType = ns3::TtaFfMacScheduler"); dan untuk *scheduling* Proportional Fair adalah (SetSchedulerType = ns3::PffMacScheduler"). Pada setiap program, *scheduling* yang digunakan hanya berjumlah satu *scheduling* saja.
6. Kemudian UE dan ENB yang telah dideklarasikan, dikonfigurasi agar saling tersambung sehingga proses transmisi data dapat dilakukan, pada program dilakukan dengan menggunakan perintah (AttachToClosestEnb(ueDev.Get(n),enbDev) ).

7. Berdasarkan arsitektur jaringan LTE, paket data yang dikirimkan bersumber dari *Packet Data Network-Gateway* (PGW) yang terhubung dengan internet. Oleh karena itu PGW dikonfigurasi dengan deklarasi pada program NS-3.
8. Pengaturan pada PGW dengan memberikan pengaturan IP *address* antara PGW dengan internet yaitu 13.0.0.0 dengan subnet mask 255.0.0.0.
9. Kemudian menentukan paket yang dikirimkan dari PGW yaitu paket berbasis UDP dengan ukuran paket variatif pada skenario satu yaitu 100B, 500B, 900B, 1300B dan 1500B sementara pada skenario dua yaitu 1500B dengan interval pengiriman paket setiap 20 ms.
10. PGW dikonfigurasi penyambungan dengan ENB. Dan dilakukan *routing* antara IP Internet Host dengan IP pada UE. IP pada UE adalah 7.0.0.0 dengan subnet mask 255.0.0.0 yaitu IP standar yang digunakan pada simulasi program NS-3.
11. Program yang telah selesai dilakukan *compile* yang akan menghasilkan data *trace file* yang selanjutnya dapat diolah menggunakan program pengolah angka yang menghasilkan data berupa bentuk grafik dan kemudian dapat dianalisis.

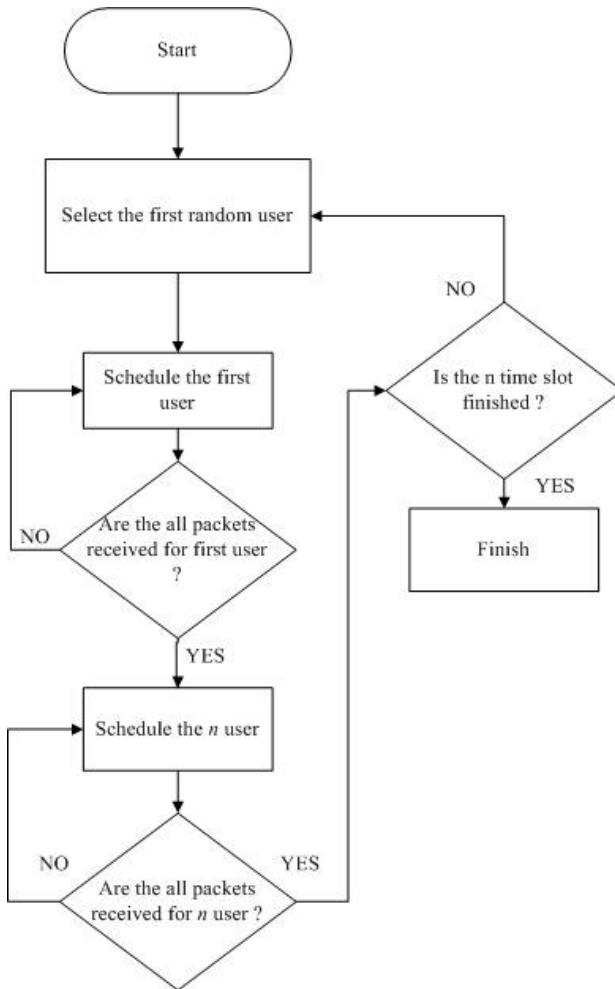
### 3.4 Algoritma *Scheduling*

Sebuah algoritma *Scheduling* harus mampu mengatur distribusi saluran pada jaringan agar mampu mencapai efisiensi maksimum. *Orthogonal Frequency Division Multiple Access* (OFDMA) pada sisi *downlink* jaringan LTE memberikan pilihan metode pengalokasian *bandwidth*. Domain waktu dan frekuensi dapat dialokasikan secara fleksibel dan dinamis dengan tujuan mencapai *maximum throughput*, *delay minimum* dan memenuhi persyaratan QoS. Berikut adalah algoritma *scheduling* yang digunakan dalam penelitian ini :

#### 3.4.1 Algoritma *Scheduling Throughput to Average*

Prinsip kerja *Throughput to Average* melakukan kuantifikasi pengalokasian *Resource block* kepada semua pengguna yang tersambung, sehingga menjamin RB terbaik dialokasikan kepada setiap pengguna. Hasil uji yang diprediksi dari TTA adalah untuk memberikan *throughput* yang sama untuk semua UE di dalam jaringan ENB, meskipun memiliki kondisi kanal yang sama maupun berbeda. [10] Diagram alir (*flowchart*) dalam algoritma *scheduling* TTA ditunjukkan pada gambar 3.5.





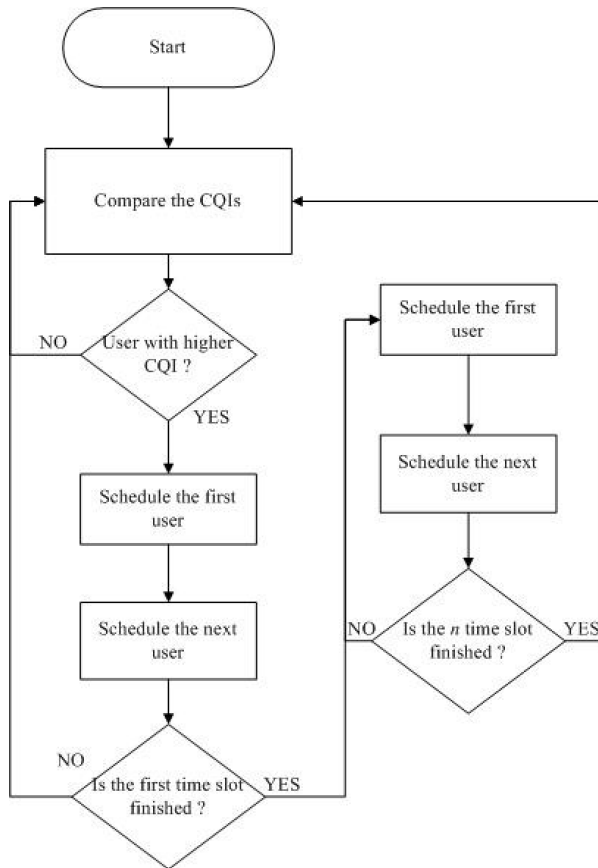
Gambar 3.6 *Flowchart Scheduling TTA* [16]

Berdasarkan gambar 3.6, diagram alir cara kerja *scheduling* TTA adalah dengan melakukan pemilihan UE dengan urutan pertama secara acak tanpa melihat dari kualitas sinyal yang diterima oleh UE tersebut. Jika UE telah terpilih maka akan dilakukan penjadwalan paket kepada UE tersebut hingga semua paket yang akan diterima, jika paket belum diterima seluruhnya maka penjadwalan akan terus dilakukan. Jika seluruh paket telah diterima UE, maka penjadwalan akan dilakukan kepada UE selanjutnya hingga seluruh paket diterima oleh UE

tersebut. Penjadwalan akan dilakukan kepada seluruh UE hingga *timeslot* yang ditentukan berakhir.

### 3.4.2 Algoritma Scheduling Proportional Fair

*Proportional Fair* (PF) adalah *packet scheduler* yang bertujuan memaksimalkan *throughput cell* sekaligus meningkatkan nilai keadilan *fairness*. PF menghitung nilai metric berdasarkan penghitungan nilai data rate dan nilai *throughput* rata-rata yang diperoleh dari penghitungan metric terakhir oleh flow yang sama. [1] Diagram alir (*flowchart*) dalam algoritma *scheduling* PF ditunjukkan pada gambar 3.6.



Gambar 3.7 Flowchart Scheduling PF [16]

Berdasarkan gambar 3.7, diagram alir cara kerja *scheduling* PF adalah dengan melakukan pemilihan UE dengan urutan pertama dengan melihat dari kualitas sinyal yang diterima oleh UE tersebut. Jika UE dengan kualitas sinyal terbaik telah terpilih maka akan dilakukan penjadwalan paket kepada UE selanjutnya dengan kualitas sinyal lebih rendah daripada UE sebelumnya. Penjadwalan akan dilakukan kepada seluruh UE secara berjenjang hingga *timeslot* yang ditentukan berakhir. Ketika *timeslot* sudah berakhir maka perbandingan kualitas sinyal akan dilakukan kembali.

### 3.5 Perangkat Simulasi

#### 3.5.1 Spesifikasi Komputer

Spesifikasi komputer yang digunakan dalam simulasi ini adalah

<i>Notebook</i>	: Lenovo G400
CPU	: Intel Celeron 1005M 2.05 GHz
RAM	: 2GB
Harddisk	: 500 GB
VGA	: Intel HD Graphics
Sistem Operasi	: Ubuntu 14.04 Trusty Tahr 32bit

#### 3.5.2 *Software*

##### A. Ubuntu 14.04 Trusty Tahr

Ubuntu Versi 14.04 “Trusty Tahr” merupakan distribusi Linux yang paling populer menggunakan user interface Unity yang khas dan disesuaikan. Trusty Tahr merupakan edisi dengan dukungan jangka panjang “*Long Term Support*” (LTS) selama 5 tahun, berupa dukungan keamanan berikut jalur upgrade yang lebih mudah dibandingkan rilis versi LTS (12.04) sebelumnya.

Rilis dengan dukungan jangka panjang (LTS) menjanjikan pemeliharaan yang didukung oleh Canonical selama lima tahun, agar senantiasa stabil, handal, aman dan sekaligus berdampak hemat biaya, terutama untuk desktop dengan penyebaran skala besar, baik pada perusahaan, sektor publik dan pemerintahan.

Mulai dari rilis ini, Canonical berikrar untuk fokus pada konvergensi Ubuntu dalam membangun infrastruktur sistem operasi open source OS tunggal yang akan berjalan secara berkesinambungan baik di PC, tablet, ponsel dan bahkan pada TV.

Diantara fitur-fitur terpenting pada Ubuntu 14.04 LTS ini, termasuk perangkat tambahan untuk AppArmor, software yang memberikan keamanan

ekstra untuk Ubuntu. Rilis Ubuntu ini datang dengan versi kernel Linux 3.13.0-19.39 terbaru yang telah menggondong fitur-fitur yang diperlukan untuk mendukung mobile dan komputasi awan (*Cloud Computing*), serta *Big Data*. [16]

### **B. NS-3.24.1**

NS-3 adalah simulator jaringan berdasarkan waktu yang bersifat diskrit, ditargetkan untuk penelitian dan penggunaan pendidikan. ns-3 adalah perangkat lunak gratis, dilisensikan di bawah lisensi GNU GPLv2, dan tersedia untuk umum untuk penelitian, pengembangan, dan penggunaan publik.

Tujuan dari proyek ns-3 adalah untuk mengembangkan, pilihan lingkungan simulasi terbuka untuk penelitian jaringan dengan kebutuhan simulasi jaringan penelitian modern dan harus mendorong kontribusi masyarakat, *peer review*, dan validasi perangkat lunak.

Proyek NS-3 berkomitmen untuk membangun simulasi yang solid yang didokumentasikan dengan baik, mudah digunakan, dukungan perbaikan oleh publik, dan melayani kebutuhan seluruh simulasi alur kerja, dari konfigurasi simulasi untuk proses analisis.

Selain itu, infrastruktur ns-3 mendorong pengembangan model simulasi yang cukup realistis dengan tujuan ns-3 yang akan digunakan sebagai *emulator* jaringan *realtime*, yang saling berhubungan dengan dunia nyata dan yang memungkinkan banyak implementasi yang ada pada protokol dunia nyata untuk digunakan kembali dalam ns-3.

Ns-3 mendukung penelitian di jaringan non-IP dan jaringan berbasis IP. Namun, sebagian besar penggunaannya berfokus pada simulasi nirkabel / IP yang melibatkan model untuk Wi-Fi, WiMAX, atau LTE untuk lapisan 1 dan 2. [17]