iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** III **Month of publication:** March 2022

DOI: https://doi.org/10.22214/ijraset.2022.40528

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue III Mar 2022- Available at www.ijraset.com

Simulating an LTE Network Using OMNeT++

D. Yamini Reddy[1], K. Hari Hruthvik[2], M. Harsha Vardhan Reddy[3], Dr. Vyomal Naishadhkumar Pandya[4]
[1, 2, 3]*B.Tech (ECE), Sreenidhi Institute Of Science and Technology, Telangana, India*
[4]*PhD (ECE), M.E.(E.C.S), MIEEE Associate Professor, ECE Department, Sreenidhi Institute Of Science and Technology, Telangana, India*

Abstract: ***Long Term Evolution emerged from the previous 3GPP system known as the Universal Mobile Telecommunication System. This technology is an entire IP-based network that provides an IP-based mobile content called the Evolved Packet Core. The main goals of LTE are to increase data rates, provide lower latency on the Radio Access Network, improve bandwidth and optimal performance. The LTE architecture combines a new basic channel called eNodeB with a single core called the Evolved Packet Core that provides access to both voice and IP data resources.***

Long Term Evolution (LTE) talks about a promising new invention that gives broadband internet access to the universe. Therefore, many test groups are trying to simplify their show. Shockingly, from now on, apparently, there are no open source simulation systems, which can be used to emulate high-density user platforms like India, which are unrestricted. Lack of a standard reference system for use by assessment students and researchers is a loss. To overcome this, we came up with a project that not only imitates LTE but can also be used to replicate any OFDM package programs such as LTE pro advanced, and was launched to provide a comprehensive presentation of the LTE environment.

Our simulation project has been contemplated to reproduce uplink and downlink editing methods in multi-cell / multi-user environments, considering customer portability, radio asset development, reusable strategies, flexible balance and coding module, and various ideas that work best for aspiring students. researchers.

Using simulation software and framework,After researching many simulation software we concluded that OMNet ++ would be the best suit for our project. We have used the Inet 4.3 framework in the required modules.

I. INTRODUCTION

LTE (Long Term Evolution) was developed from the previous 3GPP framework known as the Universal Mobile Telecommunication System. This initiative is an IP-based organization that provides a multimedia IP service called the Evolved Packet Core. The key objectives of LTE are to create high levels of information, to provide low efficiency in the Radio Access network, to improve data transfer capacity and to produce phantom. The LTE design incorporates one basic channel called eNodeB and one single channel called Evolved Packet Core that provides access to both IP-based voice control and information.

OMNeT ++ (Objective Modular Network Testbed in C ++) is a limited library and component, based on C ++-based components, primarily for building network

testing systems. OMNeT ++ itself is a hobby program without models for network protocols such as IP or HTTP.

The INET Framework is an OMNeT ++ open source climate module model library. Provides contracts, experts and various models for analysts and students working with corresponding organizations. INET is extremely important when planning and approving new agreements, or investigating new or unusual situations. A few other entertainment agencies embrace INET as a base, and expand it into explicit topics, such as automotive organizations, clustered / shared organizations, or LTE.

SimuLTE is a tool that enables the redevelopment of the LTE / LTE-Advanced organizational framework within OMNeT ++. It is structured in such a way that it can be well positioned within network components such as an additional Network Interface Card (NIC) card for those around it provided by the INET system (for example Wi-Fi).

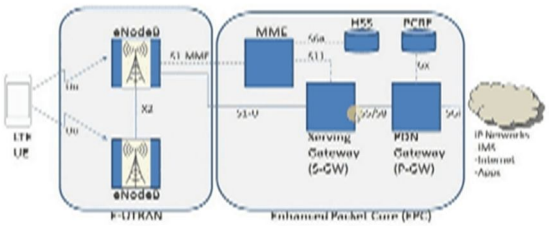


FIG:1.1 Basic LTE structure.

II. DESCRIPTION

1) *LTE:* Long Term Evolution. LTE is a norm for remote broadband correspondence for cell phones and information terminals. It expands the limit and speed utilizing an alternate radio interface along with upgrades.
2) *ENodeB:* EUTRAN Node B. It is the component in the E-UTRA of LTE that is the advancement of the component Node B.
3) *MME:* Mobility Management Entity.It is a control hub for the LTE access organization. HSS-Home Subsriber Server. HSS is the connection of the HLR and the Authentication Center.
4) *PCRF:* Policy and Charging Rules Function. It is the product hub assigned progressively to decide strategy in a media network.
5) *SGW:* Serving GateWay. The SGW lives in the client plane where it advances and courses parcels to and from the eNodeB and bundle information network door..
6) *PGW:* Packet Data Network Gateway. The Packet Data Network Gateway (PDN Gateway, additionally PGW) gives availability from the User Equipment (UE) to outer parcel information organizations (PDNs) by being its place of exit and passage of traffic

Entity	Description
UE	A UE connects to an eNB over the LTE-Uu interface.
eNB	An eNB provides users with the radio interfaces and performs Radio Resource Management (RRM) functions such as dynamic resource allocation (scheduler), eNB measurement configuration and provision, radio admission control, connection mobility control and Radio Bearer (RB) control and Inter-Cell Interference Coordination (ICIC).

FIG:2.1 eNb entities & their functions

Entity	Description
MME	An MME is the main control entity for the E-UTRAN. It communicates with an HSS for user authentication and user profile download, and provides UEs with EPS Mobility Management (EMM) and EPS Session Management (ESM) functions using NAS signaling. The main functions supported by a MME are as follows: • NAS signaling (EMM, ESM and NAS Security) • User authentication and roaming with HSS over the S6a interface • Mobility management (paging, Tracking Area List (TAI) management and handover management) • EPS bearer management
S-GW	An S-GW terminates the interface towards an E-UTRAN. It serves as the local mobility anchor point of data connections for inter-eNB handover and inter-3GPP handover.
P-GW	A P-GW provides a UE with access to a PDN by assigning an IP address from the address space of the PDN. The P-GW serves as the mobility anchor point for handover between 3GPP and non-3GPP. It also performs policy enforcement, packet filtering and charging based on the PCC rules provided by a PCRF. The main functions supported by a P-GW are as follows: • IP routing and forwarding • Per-SDF/Per-User based packet filtering • UE IP address allocation • Mobility anchoring between 3GPP and non-3GPP • PCEF functions • Charging per-SDF/per-User
HSS	An HSS is the central DB where user profiles are stored. It provides user authentication information and user profiles to the MME.
PCRF	A PCRF is the policy and charging control entity. It makes policy decisions for SDFs and provides the PCC rules (QoS and charging rules) to the PCEF (P-GW).
SPR	A SPR provides subscription information (access profile per subscriber) to the PCRF. Receiving the information, the PCRF performs subscriber-based policy and creates PCC rules.

FIG:2.2EPC entities & their functions

A. *Choosing a Simulating Software*

Any Simulation tool is used to predict, test and verify the performance of a wireless network. These tools are reliable and provide a good balance between accuracy and sophistication. Also, the simulation results are simple because it is possible to log data in important areas that can be used to diagnose network protocols. These tools overcome the limit of various analytics tools, marketing simulation tools and testing.

After passing, Venkataramanan, V; Lakshmi, S; "Examples of Multiple Wireless Network Simulation Tools", We have learned that there are three basic types of simulations such as Monte Carlo Simulation, Trace-Driven Simulation and Discrete-Event Simulations. Network Simulators develop a variety of applications. They are used to understand configuration changes in one and many links. Also, they are used to understand errors in the network and bring out the same solutions. In order to predict the results in extreme cases, it is necessary to configure measurable and reliable tools, namely Network simulations.

These simulation tools are easy and fast to use. In fact, they measure network performance and performance before deployment. These templates can be classified as commercial or free. Some of the most common examples of commercial network operators are the Optimized Network Evaluation Tool (OPNET) and QualNet. On the other hand, open source or free network templates have their advantages and disadvantages. While the downside is that there are no special people who can work with an improved version with different text, the good thing is that everyone or the organization can work on the network and help find errors. Other open source network simulations include Network Simulator, Objective Modular Network Test bed in C ++ (OMNeT ++), Scalable Simulation Framework and its extension (SSFNet), Network Simulator and Emulator (NCTUns), and J-Sim (Java based Simulator).).

NS is a computer simulation for a separate Internet Systems event.

OMNET ++ supports wireless and mobile simulation and is well suited for any system that integrates compatible devices.

SSFNet simulations are ideal for large networks and include horizontal structures. This type of simulation focuses on computer simulation and simulation simulation.

NCTUns is a network simulator that can mimic both wireless or wireless Internet Protocol (IP).

J-Sim is a scalable network simulator that promotes wireless network simulation. It uses a compatible discrete-event simulation network and its network structure is similar to Open Systems Interconnection (OSI) with seven layers.

Among all of the above, OMNET ++ and NCTUns have Graphical User Interface (GUI) streaming, J-Sim with limited GUI feature and NS-2 with no GUI streaming.

After reviewing A.Varga's "OMNeT ++ simulation environment", he suggested that in order to simulate a network environment it should have certain features namely pre-defined framework, various operating modules and continuous software updates to meet network development. They also say that simulation software should be modular, flexible and should allow blocking of simulations in large applications such as network editing software. In addition it should allow us to simulate a large scale with reusable components (modules). Due to these requirements, it is safe to say that OMNeT ++ is the most important software you can use to simulate network space.

B. Framework and Source Code

After running through many simulation source codes namely Netsim, SimuLTE, LTE simulation and many other open ware simulation source codes. We have decided to take the framework of SimuLTE.

As per “Simulating LTE/LTE-Advanced networks with SimuLTE " by Antonio Virdis, Giovanni Stea, Giovanni Nardini . SimuLTE reproduces the information plane of the LTE/LTE-A Radio Access Network and Evolved Packet Core. . SimuLTE permits reenactment of LTE/LTE A in Frequency Division Duplexing (FDD) mode, with heterogeneous eNBs (large scale, miniature, pico and so forth), utilizing omni directional and anisotropic receiving wires, conceivably imparting through the X2 interface. Sensible channel models, MAC and asset planning for the two bearings are upheld.

The framework level testing system is based on OMNeT ++ and INET properties. The basic concept of OMNeT ++ specification: simulation is performed by making various modules that communicate with each other using messages. INET offers a wide range of elements and reproduction meetings for both wireless and remote organizations. Specifically, it provides an overview of Network Interface Card (NIC) modules. The latter can be inserted into various modules to show the various connections between network gadgets. For example, gadgets can be blessed with Wi-Fi modules or Point-to-Point Protocol (PPP) NIC, or both. SIMLTE is abusing this component, as it is designed as an extension of the remote NIC module. This allows one to add LTE power to the recalled memory hub. SIMLTE provides models for both UE and eNB.

Additionally, eNB has an Internet interface using PPP and can also be linked to other eNBs using the X2 interface. LTE NIC in both UE and eNB makes the whole LTE conference stack, as one module for each layer, will be defined Packet Data Convergence Protocol (PDCP), Radio Link Control (RLC), MAC and PHY. As UE and eNB perform a variety of functions within the assembly stack, SIMI LTE abuses the OMNeT ++ asset concept by pointing to the structure and function of each layer.

Specifically, each partner has standard tasks, which are achieved through clear UE and eNB operations, each. Communication between the various layers occurs through text trading, similar to the transmission of information between UE and eNBs.

Then again, storage is removed from the transmission of information. A dedicated module, called Binder, screens what assets, for example Asset Blocks (RBs), used by eNBs (for downlink transmissions) and UEs (for uplink transmissions). Binder can be considered a prophet for the LTE organization, as all LTE facilities can access it to share general information on specific strategic calls.

C. Different Types of Simulations

SimuLTE itself offers UE and eNB models, which can mimic D2D connections on a mobile network, and can be used in conjunction with other Internet devices. Take the cables from INET. However, the use of SimuLTE NIC described in the previous section may be used on any network device. For example, VeinsLTE uses SimuLTE to simulate a separate car network, in which the car is fitted with Wi-Fi and LTE network cards. Following this guide, nodes have multiple network connections, LTE, Wi-Fi, Bluetooth, etc. We now explain how to set up D2D-related modeling parameters in a .ini configuration file, and how to use different parameters to test different scenarios. TCP connection. For CQI calculations, a consistent CQI may be used for each D2D transmission.

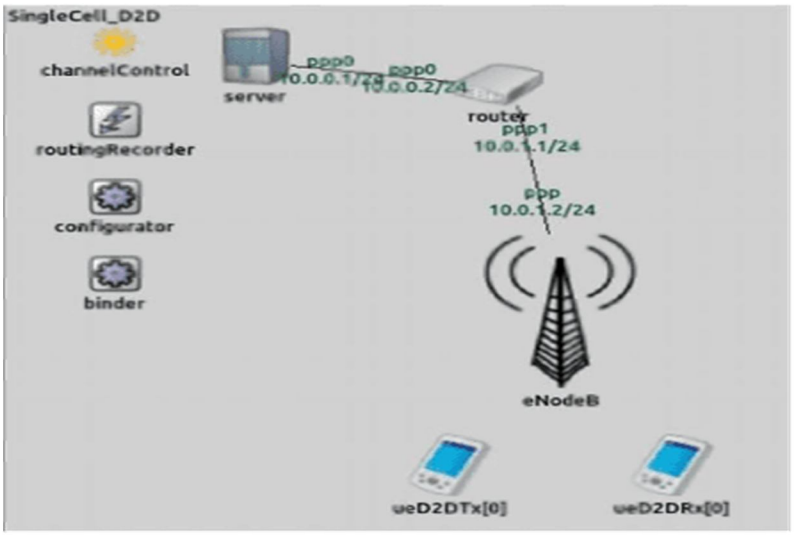


FIG:3.1 D-D connection.

1) *One-to-One D2D:* In order to enable D2D communication between two UEs, they must be identified as D2D capable UEs and eNBs. For this, the parameter d2dCapable of the node involved must be set in the .ini file. In the eNB, the D2D specific modulation and adaptive coding module (AMC) must be used to calculate the SL transmission parameters. Next, we need to configure the receiver as a possible D2D pair of the sender. The d2dPeerAddresses parameter is a space- separated list of IP addresses or UE aliases. Please note that the peer-to-peer interconnection is unidirectional, so for bidirectional streams, for example, a fixed CQI set at the beginning of the simulation can be used for each D2D transmission. Otherwise, the sending UE may be allowed to report the CQI measured for all its peer UEs on the D2D channel, that is, those in the list d2dPeerAddresses. The latter mode makes the SL line suitable for UL and DL communications. On the other hand, since CQI reporting and the number of peer-to-peer connections of the UE can be performed periodically, this method is more computationally intensive. Use different transmission capacities for UL and D2D.For this, the parameters ueTxPower and d2dTxPower must be configured accordingly. This makes it possible, for example, to estimate the maximum distance or power consumed by direct communication with certain combinations of CQI and transmit power. Since D2D supports spatial frequency reuse, different settings will also affect the interference between UEs allocated on the same resource, so as to evaluate the performance of planning algorithms that use frequency reuse and coexistence between D2D and traditional infrastructure communications.
2) *One-to-many D2D:* Multicast messages are generated at the application layer of the node and sent towards a multicast IP address. Thus, they can be received only by UEs that subscribed to the addressed IP multicast group. The latter has to be defined within the XML configuration file of the IPv4NetworkConfigurator module. In the example of below Figure , this is accomplished as follows, where all UEs belong to the group having address 224.0.0.10.The configuration of the omnetpp.ini file presents just a few changes w.r.t. the one-to-one case described before. However, for one-to-many communications, CQI can be configured only in fixed mode, i.e. setting the use Pre-configured TxParams. The selected CQI affects the transmission range of the communication. In fact, smaller CQIs allow the transmission to be received at farther distances at the cost of higher resource consumption, as more RBs are required to send a message. This can be exploited, for example, in vehicular networks for broadcasting traffic and/or collision information, even in multi-hop scenarios. In this case, different CQIs allow the packet to reach a different number of vehicles, affecting the performance of the dissemination in terms of latency and resource consumption.

III. MODULES , FEATURES AND SOURCE CODES

A. Basic Element Modules

1) *UE Module:* The design of the LTE radio convention stack model of the UE is addressed in the figures LTE radio convention stack engineering for the UE on the information plane and LTE radio convention stack design for the UE on the control plane which feature separately the information plane and the control plane.

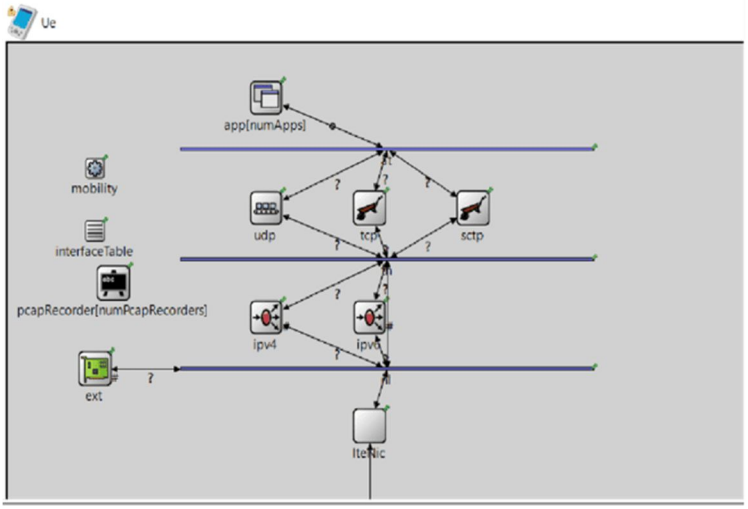


FIG: 4.1 UE MODULE

2) *eNB Architecture:* The eNodeB gives the radio interface and performs radio asset the executives for Long-Term Evolution (LTE), including radio conveyor control, radio affirmation control, and planning of uplink and downlink radio assets for singular UEs. The eNodeB likewise upholds IP header pressure and encryption of the client plane information. eNodeBs are interconnected to each other through an interface called X2; this interface has a few uses, for example handover. eNodeBs are likewise associated with the EPC through the S1 interface, which is separated into the client plane and the control plane. The control-plane interface is alluded to as S1-MME and ends in the MME. The S1-U interface, then, ends at the Serving GW and handles client plane traffic. The S1 interface upholds pooling.This permits administrators to share the radio organization, for example the eNodeBs, while keeping up their own EPC organizations.

3) *P-GW:* The P-GW acts as the anchor for the user's flight. User traffic can be filtered into P-GW for service quality (QoS) distribution between multiple package packets. P-GW collects charging data and submits these Charging Data Records (CDRs) for processing.

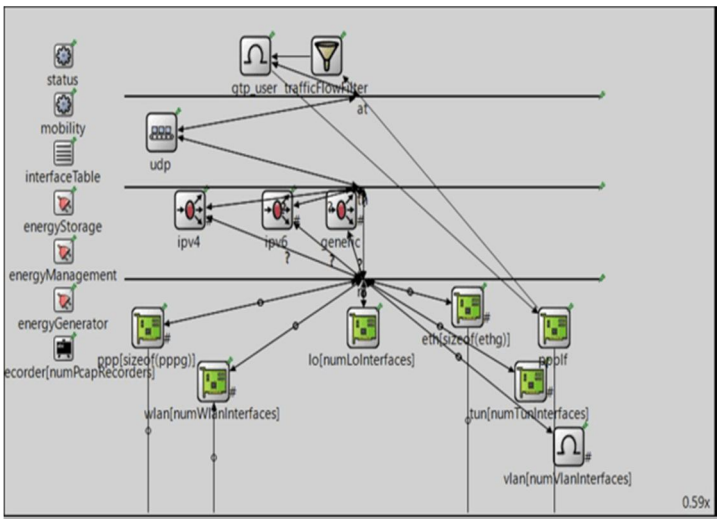


FIG: 4.2 P-GW MODULE.

4) SGW: The serving gateway (SGW) stays in the user plane where it forwards and routes packets to and from the eNodeB and packet data network gateway (PGW). The SGW serves as the local mobility anchor for inter-eNodeB handover and mobility between 3GPP networks.

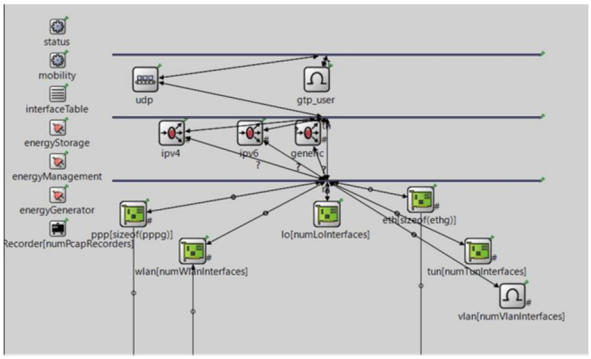


FIG: 4.3 SGW Module

5) Router: A router can be a network tool that transmits data packets between computer networks. Runners perform network traffic control tasks. Data sent via the net, such as a web page or email, is within the data package. Packages are usually transferred from one route to a different route using networks that build the Internet (e.g. the Internet) to the destination. , or wireless transmission. It can also support different levels of network layer transfer. Each network interface is used to enable data packets to be transferred from one transmission to another. Routers may also routinely connect two or more sensible groups to computer systems called subnet, each with a different network base.

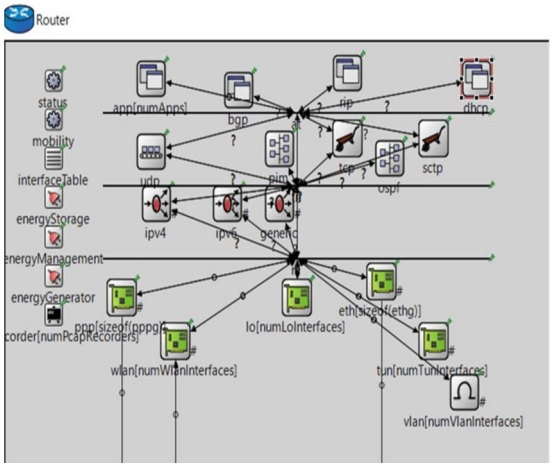


FIG:4.4 Router

B. LTE Layers

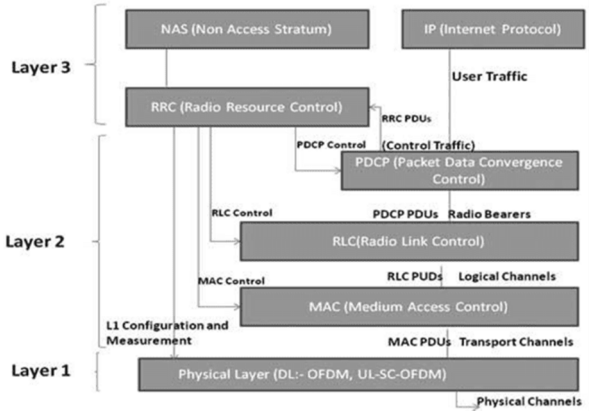


FIG:4.5 Layers

1) Physicallayer

The actual layer transfers all data to the MAC transport channels in the airspace. It deals with the modification of connectivity (AMC), power control, cell appearance (initial compatibility with delivery objectives), and the different dimensions (within the LTE framework and between frameworks) of the RRC framework.

2) Medium Access Layer

The MAC layer is responsible for Mapping between fixed channels and transport channels, Duplication of MAC SDUs channels from one or more sensible channels to transport blocks (TB) to be transmitted to a real layer of transport channels, duplicate single-channel or MAC SDU duplicate channels. from transport blocks (TB) transmitted from the actual layer of transport channels, Editing data disclosure, HARQ error review, Important experience among UE with unique bookings, Important care between official single UE channels, prioritizing Reasonable Channel.

3) Radio Link Control

RLC operates 3 modes of operation: Clear Mode (TM), Unknown Mode (UM), and Allow Mode (AM).

The RLC layer is responsible for PDU exchange over the top layer, error correction with ARQ (For AM Only Information Transfer), Consolidation, Separation, and Reassembling RLC SDUs (UM and AM-only information transfers).

The RLC is also responsible for the redistribution of PDU information for RLC (Transport for AM information only), for the reorganization of information for RLC PDUs (for UM and AM only information), for identifying a copy (for UM information only), the RLC SDU disclaims (only UM and AM information flow), RLC restarting, and conference error detection (For AM-only information transmission).

4) Non-Access Stratum Protocol

Reachable Layout Principles (NAS) form the most significant layer of control aircraft between client computing (UE) and MME computing.The NAS agreements support UE flexibility and board system assembly to establish and maintain IP availability between UE and PDN GW.

5) PDCP (Packet Data Convergence Protocol)

In the LTE radio conference stack, the PDCP layer is placed above the RLC layer and below the IP (client plane) layer or RRC layer (control plane). 3GPP provides PDCP commitment to [1]. From UMTS, the capacity of the PDCP layer system has been under pressure at the top of the IP piles - this is the reason why it is known as the Packet Data Convergence Protocol.

In LTE, the PDCP layer is designed to support security function; that is, confirmation of honor and coding. To provide time-varying attributes in the defense function, the PDCP party number is presented in the PDCP layer. The PDCP program number is integrated into each PDU PDCP, and is used to create a separate protection output for each PDU PDCP. Due to the PDCP program number, the PDCP layer can develop ARQ-related capabilities, which can improve radio efficiency in delivery.

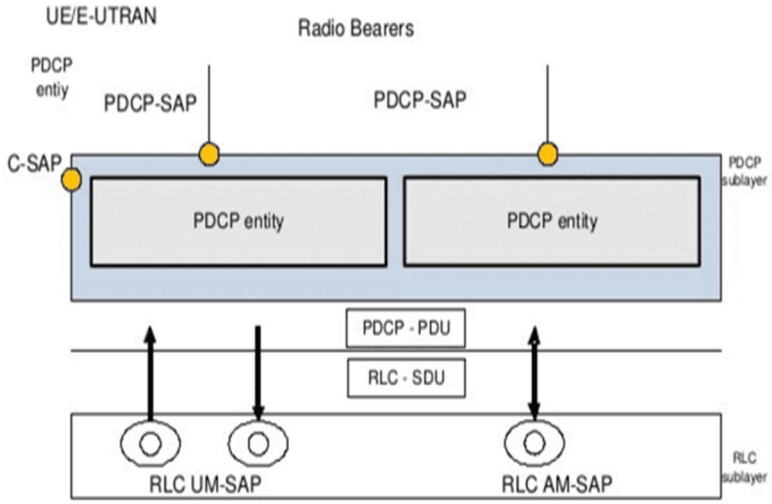


FIG:4.6 PDCP LAYER VIEW

C. RRC AND MAC (Interface Controls)

1) RRC(Radio Resource Control)

The Radio Resource Control (RRC) convention is utilized in UMTS, LTE, and 5G on the Air interface. It is a layer 3 (Network Layer) convention utilized among UE and Base Station. This convention is determined by 3GPP in TS 25.331 for UMTS, in TS 36.331 for LTE, and in TS38.33 for 5G New Radio. RRC messages are moved through the PDCP-Protocol.

The significant elements of the RRC convention incorporate association foundation and delivery capacities, broadcast of framework data, radio conveyor foundation, reconfiguration and delivery, RRC association versatility methodology, paging warning and discharge, and external circle power control. Through the flagging capacities, the RRC arranges the client and control planes as indicated by the organization status and takes into account Radio Resource Management procedures to be executed.

The activity of the RRC is guided by a state machine that characterizes certain particular expresses that a UE might be available in. The various states in this state machine have various measures of radio assets related to them and these are the assets that the UE may utilize when it is available in a given explicit state. Since various measures of assets are accessible at various states the nature of the help that the client encounters and the energy utilization of the UE are impacted by this state machine. The RRC convention is the flagging traded between the portable and the developed Node Base station (eNB) over the LTE-Uu radio interface.

RRC(Radio Resource Control)

The Radio Resource Control (RRC) convention is utilized in UMTS, LTE, and 5G on the Air interface. It is a layer 3 (Network Layer) convention utilized among UE and Base Station. This convention is determined by 3GPP in TS 25.331 for UMTS, in TS 36.331 for LTE, and in TS38.33 for 5G New Radio. The significant elements of the RRC convention incorporate association foundation and delivery capacities, broadcast of framework data, radio conveyor foundation, reconfiguration and delivery, RRC association versatility methodology, paging warning and discharge, and external circle power control. Through the flagging capacities, the RRC arranges the client and control planes as indicated by the organization status and takes into account Radio Resource Management procedures to be executed.

The activity of the RRC is guided by a state machine that characterizes certain particular expresses that a UE might be available in. The various states in this state machine have various measures of radio assets related to them and these are the assets that the UE may utilize when it is available in a given explicit state. Since various measures of assets are accessible at various states the nature of the help that the client encounters and the energy utilization of the UE are impacted by this state machine. The RRC convention is the flagging traded between the portable and the developed Node Base station (eNB) over the LTE-Uu radio interface.

The GSM framework is the most common framework for the second year. It uses the establishment of TDMA to assist portable clients in a variety of situations. This framework was originally standardized and shipped to Europe, exported worldwide. The entire vehicle has a frequency of 200 kHz, which can hold eight full-volume regional voice clients using eight TDMA network port holes. It also enhances the circuit information capacity of 9.6 Kbit / s. To aid high levels of knowledge, ETSI has developed a GPRS standard, which uses flexible coding and multislot functionality to maximize high value. Using package access further enhances frame performance and overall performance. In any case, the maximum GPRS rate is limited to approximately 144 Kbit / s.

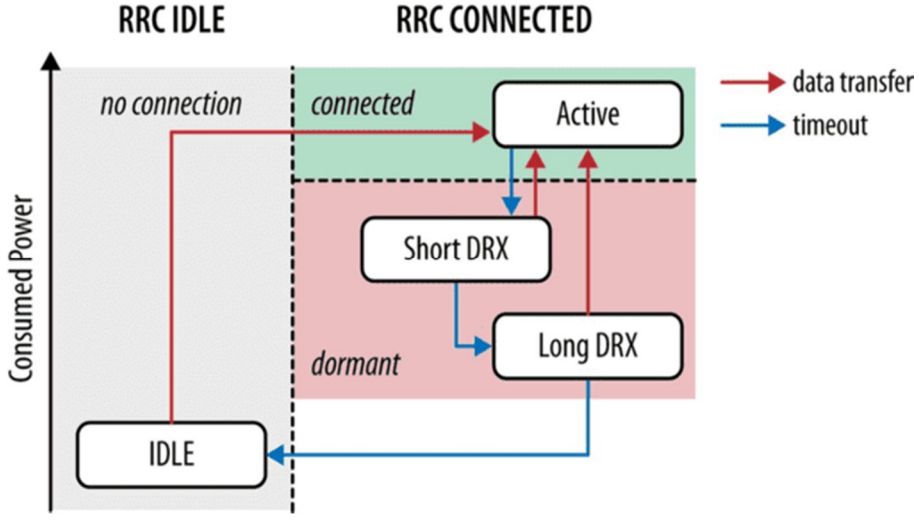


FIG:4.6 Radio Resource Control

2) Medium Access Control(MAC)

The fundamental administrations and elements of the MAC sub layer include:

a) Planning between sensible channels and transport channels

b) Multiplexing/demultiplexing of MAC SDUs having a place with one or distinctive legitimate channels into/from transport blocks (TB) conveyed to/from the actual layer on transport channels.

c) Booking Information Reporting.

d) Mistake amendment through HARQ.

e) Need dealing with between UEs through unique planning.

f) Need dealing with between sensible channels of one UE through coherent channel prioritization

g) A solitary MAC substance can uphold one or various numerologies or potentially TTI lengths and planning limitations in sensible channel prioritization controls which numerology and additionally TTI span a coherent channel can utilize

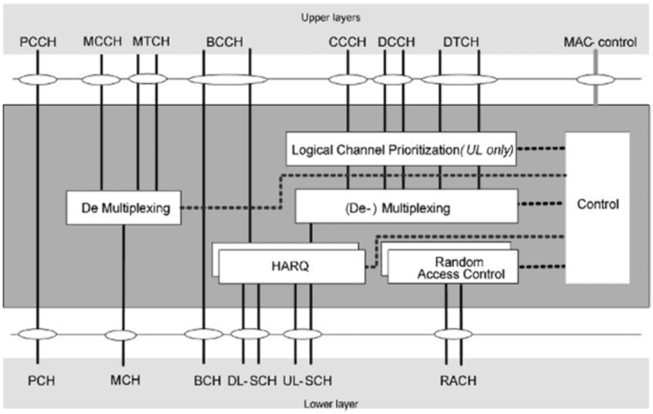


FIG:4.7 MAC architecture

D. Compression And Decompression

Header Compression

The header pressure assembly creates two types of crop packages:

-combined piles, each connected to a single PDCP SDU

- independent packages not related to PDCP SDU, for example a lot of ROHC criticism scattered.

The integrated package is related to the same PDCP SN and COUNT credentials as a linked PDCP SDU. Scattered ROHC scores are not related to PDCP SDU. They are not related to PDCP SN. Moreover, they are not encoded. Transmission of full head pressure (ROHC) to information packages may be kept away. Typically, ROHC is used to pack a large TCP / UDP / IP header in order to remotely transmit large batch headings over a radio connection. The ROHC topic has been cut off from the central organization, in order to teach information packages to the ultimate goal. Due to the misuse of RAN engineering, the ROHC header can be fully attempted to not improve cloud-RAN performance with network address translation (NAT) capabilities. In this case, the cloud-RAN may play the NAT method, that is, the location definition by adding the remote employee location to be reached, by considering the personality of the sending gadget and the type of information.

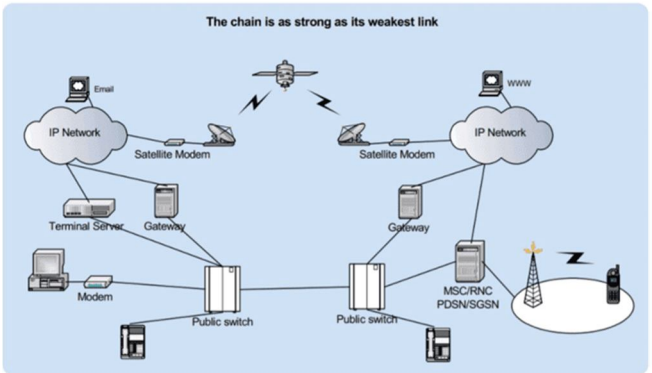


FIG:4.8 Header compression structure.

E. Header Decompression

A header decompression contraption utilized with a hub device that gets a convention information unit on a multi-facet convention stack, including a procurement segment that obtains coordinated data on pressure of multi-facet header data remembered for said convention information unit; and a decompression area that decompresses said multi-facet header data dependent on the obtained incorporated information.

A header decompression strategy for a header decompression contraption utilized with a hub device that gets a convention information unit on a multi-facet convention stack, including an obtaining step of procuring coordinated data on pressure of multi-facet header data remembered for said convention information unit; and a decompression step of decompressing said multi- facet header data dependent on the gained incorporated data.

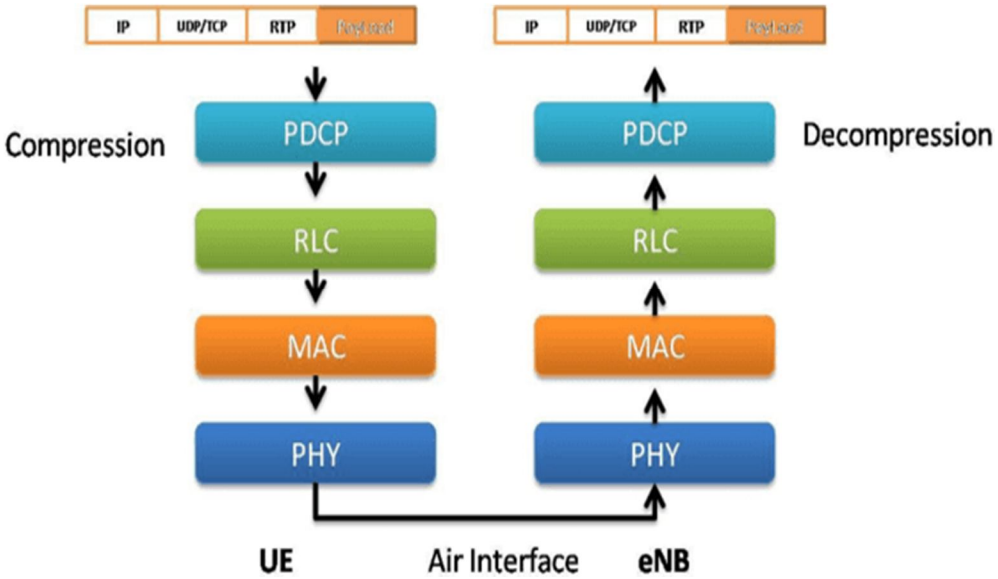


FIG:4.9 Compression and decompression interface.

F. Features

1) Radio Link Control

Elements of the RLC sub layer are performed by RLC substances. For a RLC substance designed at the eNB, there is a friend RLC substance arranged at the UE and the other way around. For a RLC substance designed at the communicating UE for STCH or SBCCH there is a friend RLC element designed at each getting UE for STCH or SBCCH.

A RLC substance gets/conveys RLC SDUs from/to upper layer and sends/gets RLC PDUs to/from its companion RLC element through lower layers. A RLC PDU can either be a RLC information PDU or a RLC control PDU. In the event that a RLC element gets RLC SDUs from upper layer, it gets them through a solitary SAP between RLC what's more, upper layer, and subsequent to shaping RLC information PDUs from the got RLC SDUs, the RLC element conveys the RLC information

PDUs to bring down layer through a solitary legitimate channel. On the off chance that a RLC substance gets RLC information PDUs from lower layer, it gets them through a solitary legitimate channel, and subsequent to shaping RLC SDUs from the got RLC information PDUs, the RLC substance conveys the RLC SDUs to upper layer through a solitary SAP among

RLC and upper layer. On the off chance that a RLC element conveys/gets RLC control PDUs to/from lower layer, it conveys/gets them through a similar legitimate channel it conveys/gets the RLC information PDUs through.

2) Retransmission

The RLC retransmission component is likewise liable for giving blunder free conveyance of information to higher layers. To achieve this, a retransmission convention works between the RLC substances in the beneficiary and transmitter. By observing the succession numbers showed in the headers of the approaching PDUs, the getting RLC can recognize missing PDUs (the RLC arrangement number is free of the PDCP grouping number). Status reports are taken care of back to the communicating RLC element, mentioning retransmission of missing PDUs. In view of the got status report, the RLC substance at the transmitter can make the proper move and retransmit the missing PDUs if necessary.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

3) Multiplexing and DeMultiplexing of MAC SDUs

The multiplexing and de-multiplexing substance is accountable for creating and deteriorating the MAC PDUs and performs (de-) multiplexing of information from a few sensible channels into/from one vehicle channel.

Consistent channel prioritization substance

At the point when the radio assets for another transmission are dispensed, the sensible channel prioritization element educates the multiplexing and de-multiplexing element to create MAC PDUs from the MAC SDUs.

The consistent channel prioritization element additionally chooses how much information from each arranged intelligent direct ought to be remembered for every MAC PDU at whatever point radio asset for another transmission is accessible. As expressed over, this choice is conveyed to the multiplexing and demultiplexing element.

IV. PROCEDURE

A. OMNeT Basics

Include path: Source folders in referenced projects are going to be automatically added to the include path of your makefile if the Add include paths exported from referenced projects option on the Compile tab is checked where the referenced projects also enable the Export include path for other projects option.

Linking: If the Link with libraries exported from referenced projects option on the Link tab is enabled, then the makefile target are going to be linked with those libraries in referenced projects that have the Export this shared/static library for other projects option checked on the Target tab.

NED types: NED types defined in a very referenced project are automatically available in referencing projects.

B. Simulation Basics

This part alludes to the model reenactments in the recreations/instructional exercise index. We make a General design in the omnetpp.ini document, which determines some normal boundaries of our situations.

For instance, we can indicate the quantity of Resource Blocks to be utilized in the recreations:

//.deployer.numRbDl = 6

//.deployer.numRbUl = 6

Parameters identified with the channel model and input calculation are situated in a different XML document that should be pointed as follows:

//.nic.phy.channelModel = xmldoc("config_channel.xml")

//.feedbackComputation = xmldoc("config_channel.xml")

These records are utilized to decide actual layer boundaries like radio wire acquire, warm commotion, target BLER and so forth and to empower/cripple the calculation of shadowing, blurring, obstruction from different cells, etc.

simple scenario

We are prepared for setting up our first reenactment situation. In the omnetpp.ini record, we make another setup called SingleCell. The organization model (characterized in the recreations/networks registry) contains one eNodeB serving a variable number of UEs.

[Config SingleCell]

network = lte.simulations.networks.SingleCell

UEs need to speak with a worker situated on the web. Along these lines, we design the worker so as it runs however many applications as the quantity of UEs. UEs should be related with the eNodeB, by setting macCellId and dominated boundaries (eNodeB IDs are doled out continuously, beginning from 1). Portability boundaries (i.e., starting position and versatility type) are likewise determined both for the eNodeB and the UEs.

/.ue[*].numUdpApps = 1

/.server.numUdpApps = ${numUEs}

connect each UE to the eNB

//.ue[*].macCellId = 1

//.ue[*].masterId = 1

Positioning and mobility

/.eNodeB.mobility.initFromDisplayString = false

/.eNodeB.mobility.initialX = 300m

/.eNodeB.mobility.initialY = 300m

/.ue[*].mobility.constraintAreaMaxX = 600m

/.ue[*].mobility.constraintAreaMaxY = 600m

/.ue[*].mobility.constraintAreaMinX = 0m

/.ue[*].mobility.constraintAreaMinY = 0m

/.ue[*].mobility.initFromDisplayString = false

/.ue[*].mobility.initialX = uniform(0m,600m)

/.ue[*].mobility.initialY = uniform(0m,600m)

/.ue[*].mobility.speed = 1mps

/.ue[*].mobilityType = "LinearMobility"

We have not yet characterized the sort of the traffic. We get two distinct arrangements by expanding the past one: one for the downlink course and one for the uplink bearing.

[config singlecell-ul]

extend=singlecell

/.ue[*].udpApp[0].typename = "VoIPReceiver"

/.server.udpApp[*].destAddress ="ue["+string(ancestorIndex(0))+"]"

/.server.udpApp[*].localPort = 3088+ancestorIndex(0)

/.server.udpApp[*].typename = "VoIPSender"

/.server.udpApp[*].startTime = uniform(0s,0.02s)

[config=singlecell-dl]

extend=singlecell

/.server.udpApp[*].typename = "VoIPReceiver"

/.server.udpApp[*].localPort = 3000+ancestorIndex(0)

/.ue[*].udpApp[/].destAddress = "server"

*.ue[*].udpApp[*].destPort = 3000+ancestorIndex(1)

/.ue[*].udpApp[*].localPort = 3088

/.ue[*].udpApp[*].typename = "VoIPSender"

/.ue[*].udpApp[*].startTime = uniform(0s,0.02s)

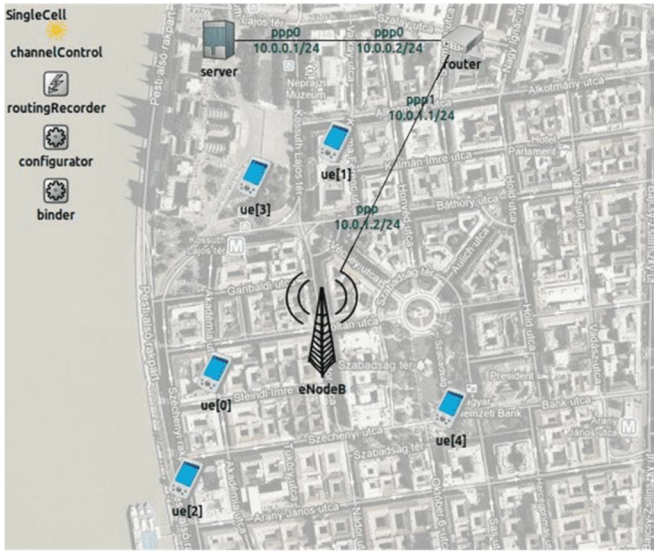


FIG:5.1 Single cell multiple devices.

C. Simulating More than One Cell

Further to present another setup, called Multi Cell. The organization model (characterized in the simulations*networks index) contains two eNodeBs, each serving two UEs. Once more, UEs need to speak with a worker.

[Config MultiCell]

network = lte.simulations.networks.MultiCell

The configuration is very similar to that of the single cell scenario. Remember to associate UEs with the correct eNodeB

//.ue1*.macCellId = 1

//.ue1*.masterId = 1

//.ue2*.macCellId = 2

//.ue2*.masterId = 2

In addition, we need to tell the IPv4 Network Configurator module that UEs served by eNodeB1 and UEs served by eNodeB2 have a place with various organizations. We do this by making a setup document for the IPv4 Network Configurator called multi.xml. The name of the arrangement document should be determined in the NED meaning of the organization, as a boundary of the IPv4NetworkConfigurator submodule.

<config>

<interfacehosts="*"address="10.x.x.x" netmask="255.x.x.x"*>

<wireless hosts="eNodeB1 ue1*"*>

<wireless hosts="eNodeB2 ue2*"*>

</config>

When utilizing at least two eNodeBs, almost certainly, we need to mull over the presence of between cell interference. This component should be empowered in the channel model arrangement. In this manner, we change the multi cell-interference boundary in the config_channel.xml document as follows:

<parameter name="multiCell-interference" type="bool" value="true"/>

Similar to single cell basis, we extend the configuration to specify the traffic type is follows:

[config Multicel-DL]

extends=multicell

Application Setup

/.ue/.udpApp[*].typename = "VoIPReceiver"

/.server.udpApp[*].typename = "VoIPSender"

/.server.udpApp[0].destAddress = "ue11"

/.server.udpApp[1].destAddress = "ue12"

/.server.udpApp[2].destAddress = "ue21"

/.server.udpApp[3].destAddress = "ue22"

/.server.udpApp[*].localPort = 3088+ancestorIndex(0)

/.server.udpApp[*].startTime = uniform(0s,0.02s)

Here's a snapshot of the graphical runtime environment:

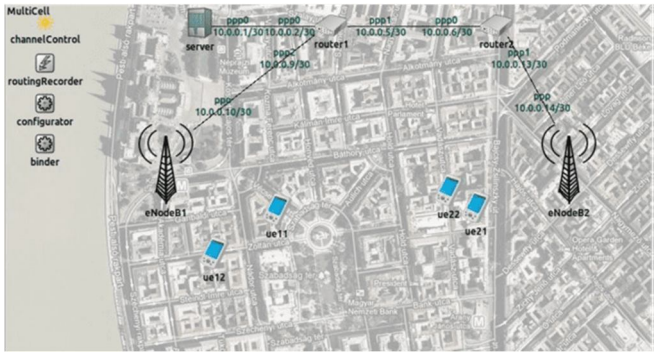


FIG:5.2Multicell with multiple devices

D. Adding External Cells to the Network

An outer cell is an improved on eNodeB that has no UEs to serve except for produces obstruction to UEs conveyed in the organization. This is a valuable component to reproduce downlink between cell impedance without adding increasingly more eNodeBs, along with their served UEs. We characterize a MultiCell-DL-ExtCell arrangement as follows:

[congig=multicell-DL-extcells]

extends-multicell-DL
/.numExtCells = 2
Configuration
/.extCell[/].txPower = 20

/.extCell[/].txDirection = "ANISOTROPIC"

/.extCell[/].bandAllocationType = "RANDOM_ALLOC"

/.extCell[/].bandUtilization = 0.5

Positioning

/.extCell[0].position_x = 100m

/.extCell[0].position_y = 600m

/.extCell[0].txAngle = 315

/.extCell[1].position_x = 600m

/.extCell[1].position_y = 600m

/.extCell[1].txAngle = 225

For every outside cell, we characterize the position, the transmission power, and the transmission point (if anisotropic cells are utilized). In addition, bandAllocationType and band use boundaries determine how outside cells dispense Resource Blocks over the accessible transfer speed (e.g., utilizing bordering RBs or arbitrarily picked RBs over the entire transmission capacity) and the level of involved Resource Blocks, individually. Once more, to empower obstruction from outside cells, we need to set the extCell-impedance boundary in the config_channel.xml record as follows:
<parameter name="extCell-interference" type="bool" value="true"/>

E. Using X2 Interface

Whenever we have fabricated an organization made out of various eNodeBs, almost certainly, we are keen on permitting correspondence among the last through the X2 interface.
For lucidity, you can discover the model portrayed in this part in another catalog, for example recreations/x2.
Let us to consider an organization with three eNodeBs. As a matter of first importance, the X2Enabled banner should be set.
To set up a X2 correspondence with a companion, the eNodeB should have one X2 application for each friend (two, for this situation). Make sure to allocate distinctive port numbers to various X2 applications.

```
//.x2Enabled = ${x2=true}
/.eNodeB/.numX2Apps = 2
/.eNodeB/.x2App[*].server.localPort = 5000 + ancestorIndex(1)
```

Note that a X2 application is made out of two autonomous modules: a worker (sending messages) and a customer (getting messages). Subsequently, modules should be arranged independently (thus the x2App[*].server documentation).

Since X2 applications run on top of the SCTP transport convention, we may two or three boundaries of the last mentioned. For instance, we can impair the Nagle calculation, which defers the transmissions of little bundles (this could be a bothersome conduct for the X2). To accelerate the recreation, we could choose to debilitate the transmission of pulses bundles (SCTP utilizes these messages to test the reachability of a location).

```
//.sctp.nagleEnabled = false
//.sctp.enableHeartbeats = false
```

Presently, we need to arrangement the customer module of the X2 applications, for example objective location of the peering eNodeB. Be that as it may, this relies upon how the eNodeBs are associated with one another (would they say they are associated through highlight point interfaces or would they say they are associated with a backhaul network?). The current variant backings just a X2 backhauling associated as a full-network geography.

F. Building a Full-Mesh Topology

We create an configuration called X2-MeshTopology, where each and every couple of eNodeBs is linked via PPP links. Again, the network definition is in the *simulations/networks* directory.

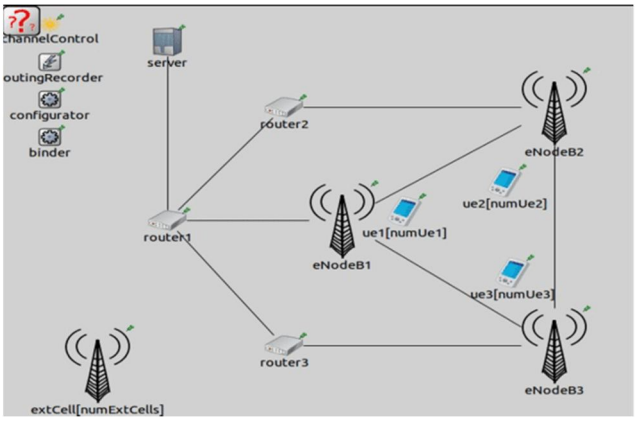


FIG:5.3 X-2 Mesh topology

```
[config x2-MeshTopology]
network=lte.simulations.network.multicell_X2mesh

  /.eNodeB1.x2App[0].client.connectAddress= "eNodeB2%x2ppp0"

  /.eNodeB1.x2App[1].client.connectAddress= "eNodeB3%x2ppp0"

  /.eNodeB2.x2App[0].client.connectAddress= "eNodeB1%x2ppp0"

  /.eNodeB2.x2App[1].client.connectAddress= "eNodeB3%x2ppp1"

  /.eNodeB3.x2App[0].client.connectAddress= "eNodeB1%x2ppp1"

  /.eNodeB3.x2App[1].client.connectAddress= "eNodeB2%x2ppp1"
```

Associating eNodeBs by means of a full-network geography is slightly more mind boggling. Truth be told, each eNodeB has one X2 interface for every peering eNodeB, consequently we need to indicate an alternate door record for every association. In this model, eNodeB2 associates with the door 0 of eNodeB1 and to the entryway 1 of eNodeB3.

V. WHY OMNET++?

OMNeT++ is an extensible, modular, component-based C++ simulation library and framework, primarily for building network simulators

1) OMNeT++ provides a powerful GUI. This strong GUI makes the tracing and debugging much easier than using other simulators.

2) Although initial OMNeT++ do not support the module library which is specifically used for WSNs simulation, with the consciously contribution of the supporting team, now OMNeT++ has a mobility framework. This simulator can support MAC protocols as well as some localized protocols in WSN. People can use OMNeT++ to simulate channel controls in WSNs. In addition,

3) OMNeT++ can simulate power consumption problems in WSNs.

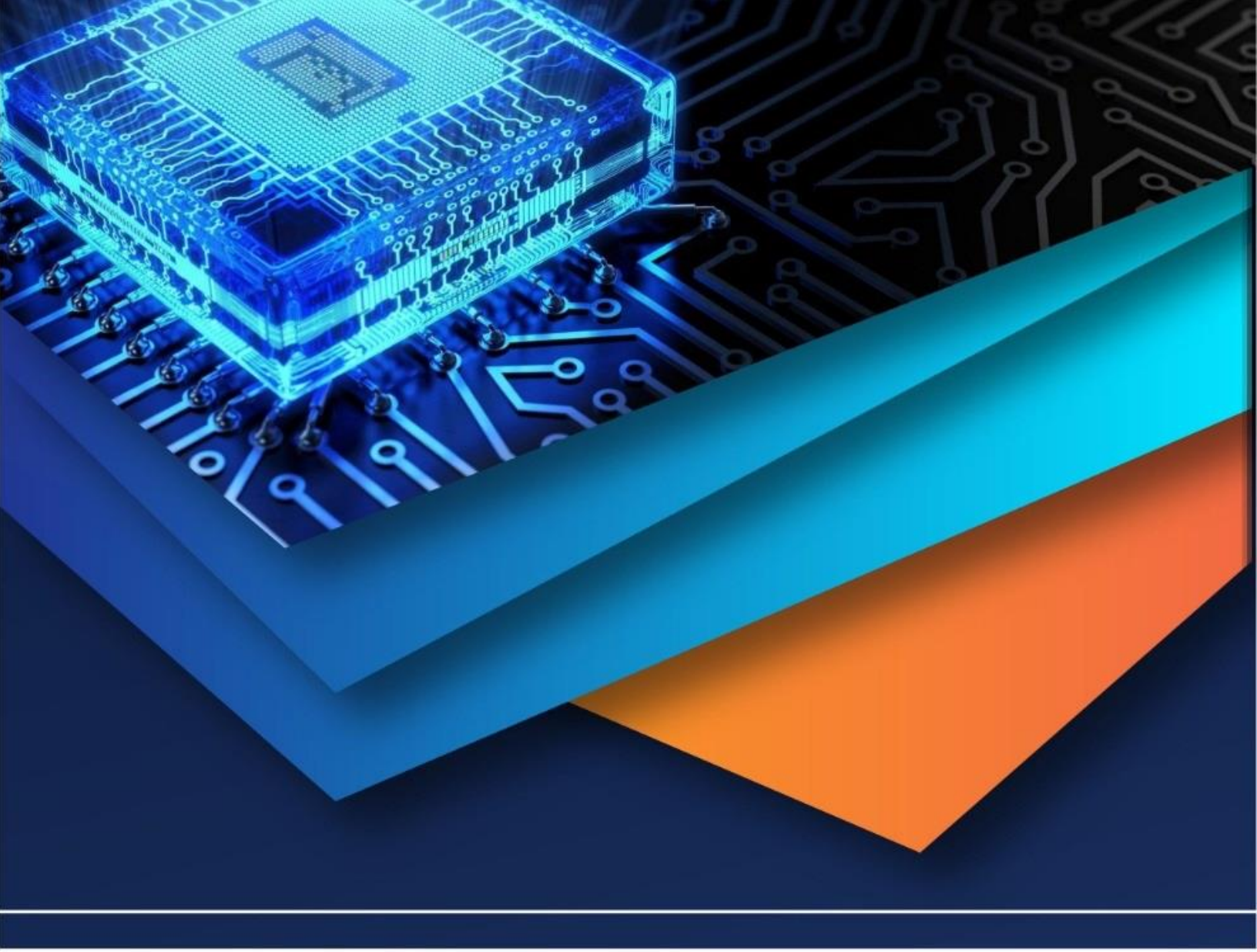
4) Strong structured. Extremely modular. Not limited to network protocol simulation. Source code is openly available. Simulation model for internet, IPV6, mobility is also accessible.

VI. CONCLUSION

In this work we present SimuLTE, an OMNeT++-based simulator for LTE and LTE-Advanced networks. We have simulated an LTE network under various scenarios . Measured throughputs and observed the inner implications of an network environmentFollowing well-established OMNeT++ programming practices, SimuLTE exhibits a fully modular structure, which makes it easy to be extended, verified and integrated. Moreover, it inherits all the benefits of such a widely-used and versatile simulation framework as OMNeT++, i.e., experiment support and seamless integration with the OMNeT++ network modules, such as INET. This allows SimuLTE users to build up mixed scenarios where LTE is only a part of a wider network. This paper describes the architecture of SimuLTE, with particular emphasis on the modeling choices at the MAC layer, where resource scheduling is located. Furthermore, we describe some of the verification and validation efforts and present an example of the performance analysis that can be carried out with SimuLTE.

REFERENCES

[1] Varga, R. Hornig, “An overview of the OMNeT++ simulation environment”, in Proc. SIMUTools '08, Marseille, FR, March 2008.
[2] The INET framework, https://inet.omnetpp.org/
[3] A. Virdis, G. Stea, G. Nardini, “Simulating LTE/LTE-Advanced Networks with SimuLTE”, DOI 10.1007/978-3-319-26470-7_5, in: Advances in Intelligent Systems and Computing, Vol. 402, pp. 83-105, Springer, 15 January 2016.
[4] 3GPP - TS 36.843 v12.0.1, “Study on LTE Device-to-device Proximity Services: Radio aspects (Release 12)”, March 2014.
[5] Virdis, G. Nardini, G. Stea, "Modeling unicast device-to-device communications with SimuLTE", IWSLS2 2016, Vienna, July 1st, 2016.
[6] Venkataramanan, V; Lakshmi, S; " A Case Study of Various Wireless Network Simulation Tools", AND Wikipedia



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)