THE NS-3 LTE MODULE

Nicola Baldo, Ph.D.
Senior Researcher
Mobile Networks Department
Centre Tecnològic de Telecomunicacions de Catalunya (CTTC)
Castelldefels, Spain
nbaldo@cttc.es
How the LTE module was developed

- Google Summer of Code 2010
- **LENA project**
  - CTTC-Ubiquisys, Jan 2011 to Jun 2013
- Other projects, including:
  - SYMBIOSIS project
  - FP7 ICT-COMBO project
- Community contributions
The LENA project: an open source product-oriented LTE/EPC Network Simulator

- A Product-oriented simulator:
  - designed around an industrial API: the Small Cell Forum MAC Scheduler Interface Specification
  - Allows testing of real code in the simulation
  - Accurate model of the LTE/EPC protocol stack
  - Specific Channel and PHY layer models for LTE macro and small cells

- An Open source simulator:
  - Development open to the community
  - Fosters early adoption and contributions
  - Helps building confidence and trust on simulation model
  - Candidate reference evaluation platform
  - Based on ns-3
  - Free and open source licensing (GPLv2)
LENA: an open source product-oriented LTE/EPC Network Simulator

Target applications for LENA include the design and performance evaluation of:

- DL & UL LTE MAC Schedulers
- Radio Resource Management Algorithms
- Inter-cell interference coordination solutions
- Load Balancing and Mobility Management
- Heterogeneous Network (HetNets) solutions
- End-to-end QoE provisioning
- Multi-RAT network solutions
- Cognitive LTE systems
LENA High level requirements

- Support the evaluation of:
  - Radio-level performance
  - End-to-end QoE

- Allow the prototyping of algorithms for:
  - QoS-aware Packet Scheduling
  - Radio Resource Management
  - Inter-cell Interference Coordination
  - Self Organized Networks
  - Cognitive / Dynamic Spectrum Access

- Scalability requirements:
  - Several 10s to a few 100s of eNBs
  - Several 100s to a few 1000s of UEs
Design approach

- Simulation is a tradeoff between:
  - Detail of the model
  - Implementation complexity and run-time scalability
- Choose min detail that satisfies requirements
  - Minimize implementation complexity
  - Minimize difficulty in using the simulator
(Some) Important Design Choices

- FemtoForum LTE MAC Scheduler API
- Radio signal model granularity: Resource Block
  - Symbol-level model not affordable
  - Simplified Channel & PHY model
- Realistic Data Plane Protocol stack model
  - Realistic RLC, PDCP, S1-U, X2-U
  - Allow proper interaction with IP networking
  - Allow end-to-end QoE evaluations
- Hybrid Control Plane model:
  - Realistic RRC model
  - Simplified S1-C, X2-C and S11 models
- Simplified EPC
  - One MME and one SGW
  - SGW and PGW in the same node (no S5/S8 interface)
- Focus on connected mode
  - RRC connected, EMM Registered, ECM connected
LENA model overview
End-to-end Data Plane protocol stack
End-to-end Data Plane architecture: data flow in downlink
End-to-end Data Plane architecture:
data flow in uplink
LTE Data Plane protocol stack: UE
LTE Data Plane protocol stack: eNB
PHY and Channel architecture: UE
PHY and Channel architecture: eNB
LTE Ctrl Plane protocol stack: UE
LTE Ctrl Plane protocol stack: eNB
EPC Control Plane Architecture
Radio Propagation Models

- Included new models for enabling 3GPP-like scenarios
  - New path loss models (indoor and outdoor)
    - External & internal wall losses
    - Shadowing
    - Pathloss logic
  - Buildings model
    - Add buildings to network topology
  - Antenna models
    - Isotropic, sectorial (cosine & parabolic shape)
  - Fast fading model
    - Pedestrian, vehicular, etc.
Outdoor Radio Propagation models

- Okumura Hata: open area pathloss for distances > 1 Km and frequencies ranging from 150 MHz to 2.0 GHz

- Kun empirical model for 2.6 GHz

- ITU-R P1411 Line-of-Sight (LoS) short range outdoor communication in the frequency range 300 MHz to 100 GHz
  - Used for short communication link (< 200 m.)

- ITU-R P1411 Non-Line-of-Sight (NLoS) short range outdoor communication over rooftops in the frequency range 300 MHz to 100 GHz.
  - Used for communication link < 1 km
Buildings defined as ns3 Box classes with
- xMin, xMax, yMin, yMax, zMin, zMax (inherited by Box)
- Number of floors
- Number of room in x-axis and y-axis (grid topology)

Buildings model that allows to “install” building information to
mobility model of a node:
- the ns3 Box class containing the building
- Position in terms of floors
- Position in the grid of rooms
- Node condition: indoor vs. outdoor

ITU-R P1238 implements building-dependent indoor propagation
loss model as function of the type of building (i.e., residential,
office and commercial)
• External wall losses for penetration loss through walls for indoor to outdoor communications and vice-versa (from COST231)
  – Wood ~ 4 dB
  – Concrete with windows (not metallized) ~ 7 dB
  – Concrete without windows ~ 15 dB (spans between 10 and 20)
  – Stone blocks ~ 12 dB
• Internal wall losses evaluated assuming that each single internal wall has a constant penetration (5 dB) and evaluating the number of walls
• Log-normal shadowing standard deviation as function of the connection characteristics
  – Outdoor $\sigma_o = 7$
  – Indoor $\sigma_i = 10$
  – External walls penetration $\sigma_E = 5$
• Height gain model when transmitting device is on a floor above the ground (2 dB)
• Pathloss logic chooses correct model depending on nodes positions
Antenna models

- Isotropic [default one]
- Sectorial (cosine & parabolic shape)
Fading model

• Fast fading model based on pre calculated traces for maintaining a low computational complexity
  – Matlab script provided in the code using \texttt{rayleighchan} function

• Main parameters:
  – \textbf{users’ speed}: relative speed between users (affects the Doppler frequency)
  – \textbf{number of taps} (and relative power): number of multiple paths considered
  – \textbf{time granularity} of the trace: sampling time of the trace.
  – \textbf{frequency granularity} of the trace: number of RB.
  – \textbf{length of trace}: ideally large as the simulation time, might be reduced by windowing mechanism.

Urban scenario 3 kmph  
Pedestrian scenario 3 kmph
• Only FDD is modeled
• Freq domain granularity: RB
• Time domain granularity: 1 TTI (1 ms)
• The subframe is divided in frequency into DL & UL

• DL part is made of
  – control (RS, PCFICH, PDCCH)
  – data (PDSCH)

• UL part is made of
  – control and data (PUSCH)
  – SRS
Interference and Channel Feedback

- **LTE Spectrum model**: \((f_c, B)\) identifies the radio spectrum usage
  - \(f_c\): LTE Absolute Radio Frequency Channel Number
  - \(B\): Transmission Bandwidth Configuration in number of RB
  - Supports different frequencies and bandwidths per eNB
  - UE will automatically use the spectrum model of the eNB it is attached to

- **Gaussian Interference model**
  - powers of interfering signals (in linear units) are summed up together to determine the overall interference power per RB basis

- **CQI feedback**
  - periodic wideband CQIs: single value representative for the whole B.
  - inband CQIs: a set of value representing the channel state for each RB

- **In DL evaluated according to the SINR of control channel (RS, PDCCH)**
- **In UL evaluated according to the SINR of**
  - SRS signal periodically sent by the UEs.
  - PUSCH with the actual transmitted data.

- **Scheduler can filter the CQI according to their nature:**
  - SRS_UL_CQI for storing only SRS based CQIs.
  - PUSCH_UL_CQI for storing only PUSCH based CQIs.
  - ALL_UL_CQI for storing all the CQIs received.
PHY Data error model

- Signal processing not modeled accurately ⇒ use error model
- Transport Block error model
- Used for PDSCH and PUSCH
- Based on Link-to-System Mapping
  - SINR measured per Resource Block
  - Mutual Information Effective SINR Mapping (MIESM)
  - BLER curves from dedicated link-level LTE simulations
  - Error probability per codeblock
  - Multiple codeblocks per Transport Block
BLER Curves
PHY DL Control error model

- Error model only for downlink, while uplink has an error-free channel
- Based on an evaluation study carried out in the RAN4 (R4-081920)
- Evaluated according to the equivalent SINR perceived in the whole bandwidth of PCFICH+PDCCH with MIESM model
- In case of error correspondent DCIs are discarded and data will not decoded
• Ns3 provides only SISO propagation model
• MIMO has been modeled as SINR gain over SISO according to
• Catreux et al. present the statistical gain of several MIMO solutions respect to the SISO one (in case of no correlation between the antennas as CDF)
• The SINR distribution can be approximated with a log-normal one with different mean and variance as function of the scheme considered (i.e., SISO, MIMO-Alamouti, MIMO-MMSE, MIMO-OSIC-MMSE and MIMO-ZF)
• Variances are not so different and they are approximatively equal to the one of the SISO mode already included in the shadowing component of the BuildingsPropagationLossModel
• MIMO can be modeled as different gains for different TX modes respect to the SISO
UE Measurements

- UE has to report a set of measurements of the eNBs when it receives their physical cell identity (PCI)
  - reference signal received power (RSRP) ~ “average” power across the RBs
  - reference signal received quality (RSRQ) ~ “average” ratio between the power of the cell and the total power received across all the RBs
- Measurements are performed during the reception of the RS
- RSRP is reported by PHY layer in dBm while RSRQ in dB through the C-PHY SAP every 200 ms.
- Layer 1 filtering is performed by averaging the all the measurements collected during the last window slot.
HARQ model

- Model implemented is *soft combining hybrid IR Full incremental redundancy* (also called IR Type II)
- Asynchronous model for DL
  - Dedicated feedback (ideal)
- Synchronous model for UL
  - After 7 ms of the original transmission
- Retransmissions managed by Scheduler
  - Retransmissions are mixed with new one (retx has higher priority)
  - Up to 4 redundancy version (RV) per each HARQ block
- Integrated with error model
  - New rates due to the “soft combination” of the codeblocks
  - Extend the original ones with the ones of RVs with lower $R_{\text{eff}}$ of each modulation order

\[
R_{\text{eff}} = \frac{X}{q} \sum_{i=1}^{q} C_i \\
X \text{ no. of info bits} \\
C_i \text{ no. of coded bits}
\]
MAC & Scheduler model

- Resource allocation model:
  - allocation type 0
  - RBs grouped into RBGs
  - localized mapping approach (2 slots of the RBG to the same UE)

- Transport Block model
  - Mimics 3GPP structure
    - mux RLC PDU onto MAC PDU
  - Virtual MAC Headers and CEs (no real bits)
    - MAC overhead not modeled
    - Consistent with requirements (scheduler neglects MAC OH)
Adaptive Modulation and Coding (AMC)

- Two algorithms working on reported CQI feedback
  - **Piro** model: based on analytical BER (very conservative)
    \[
    \text{BER} = 0.00005, \quad \Gamma = \frac{-\ln(5 \times \text{BER})}{1.5}, \quad \gamma_i \text{ SINR of UE } i, \\
    \eta_i = \log_2 \left( 1 + \frac{\gamma_i}{\Gamma} \right)
    \]
  - **Vienna** model: aim at max 10% BLER as defined in TS 36.213 based on error model curves

- Dynamic TX mode selection supported
  - Interface present in the scheduler interface
  - but no adaptive algorithm currently implemented
Most of the primitives used in LENA are based on the scheduler APIs

- Example: primitive CSCHED_CELL_CONFIG_REQ is translated to
  CschedCellConfigReq method with struct
  CschedCellConfigReqParameters parameters in the ns-3 code
- Control primitives has been implemented through Service Access Points (SAPs)

Control APIs (configuration and update)

- CschedCellConfigReq
- DoCschedUeConfigReq
- DoCschedLcConfigReq
- DoCschedLcReleaseReq
- DoCschedUeReleaseReq
SCF-API MAC Scheduler Interface (2)

- Scheduling primitives
  - DoSchedDlRlcBufferReq
  - DoSchedDlTriggerReq (containing info on HARQ)
  - DoSchedDlRachInfoReq
  - DoSchedDlCqiInfoReq
  - DoSchedUlTriggerReq
  - DoSchedUlCqiInfoReq
MAC Scheduler implementations

- Round Robin (RR)
- Proportional Fair (PF)
- Maximum Throughput (MT)
- Throughput to Average (TTA)
- Blind Average Throughput (BET)
- Token Bank Fair Queue (TBFQ)
- Priority Set Scheduler (PSS)
- Channel and QoS Aware Scheduler (CQA)

- All implementations based on the FemtoForum API
- The above algorithms are for downlink only
- For uplink, all current implementations use the same Round Robin algorithm
- Assumption: HARQ has always higher priority respect to new data
Round Robin

• Divide the available resources among the active UEs (i.e., the ones with at least one LC with buffer \(!= 0\))

• If no. of UEs > no. RBs
  – Circular buffer allocation
Proportional Fair

- Schedule a user when its instantaneous channel quality is high relative to its own average channel condition over time.
- Defines per each UE $i$ the achievable rate as:
  \[
  R_i(k, t) = \frac{S(M_{i,k}(t), 1)}{\tau} \quad M_{i,k}(t) \text{ MCS usable by user on resource block}\]
  \[
  \tau \quad \text{TTI duration}\]

- At RBG $k$ pick the user that maximize
  \[
  \hat{i}_k(t) = \arg\max_{j=1,...,N} \left( \frac{R_j(k, t)}{T_j(t)} \right) \quad T_j(t) \text{ past throughput perceived by the user } j
  \]
  \[
  T_j(t) = (1 - \frac{1}{\alpha})T_j(t-1) + \frac{1}{\alpha}\hat{T}_j(t)
  \]

- Achievable rate ratio
  \[
  \rho_{R,i} = \frac{R_i}{\sum_{j=1}^{N} R_j}
  \]

- Achievable throughput ratio
  \[
  \rho_{T,i} = \frac{T_i}{\sum_{j=1}^{N} T_j}
  \]
Other “theoretical” LTE MAC schedulers

- **Maximum Throughput (MT)**
  
  \[ R_i(k,t) = \frac{S(M_{i,k}(t), 1)}{\tau} \]
  \[ \hat{i}_k(t) = \arg\max_{j=1,...,N} (R_j(k,t)) \]

- **Throughput to Average (TTA)**
  
  \[ \hat{i}_k(t) = \arg\max_{j=1,...,N} \left( \frac{R_j(k,t)}{R_j(t)} \right) \]

- **Blind Average Throughput (BET)**
  
  \[ \hat{i}_k(t) = \arg\max_{j=1,...,N} \left( \frac{1}{T_j(t)} \right) \]

\[ M_{i,k}(t) \] MCS usable by user on resource block
\[ \tau \] TTI duration
\[ T_j(t) \] past throughput perceived by the user j
LTE MAC Schedulers with QoS support

- Token Bank Fair Queue (TBFQ)
  - leaky-bucket mechanism

- Priority Set Scheduler (PSS)
  - controls the fairness among UEs by a specified Target Bit Rate (TBR) defined with QCI bearer primitive

- Channel and QoS Aware (CQA)
  - considers the head of line (HOL) delay, the GBR parameters and channel quality over different subbands.
Random Access model

• Random Access preamble transmission
  – Ideal model: no propagation / error model
  – Collisions modeled with protocol interference model
  – No capture effect ⇒ contention resolution not modeled

• Random Access Response (RAR)
  – Ideal message, no error model
  – Resource consumption can be modeled by scheduler

• message3
  – UL grant allocated by Scheduler
  – PDU with actual bytes, subject to error model

• Supported modes:
  – Contention based (for connection establishment)
  – Non-contention based (for handover)
RLC Model

• Supported modes:
  – RLC TM, UM, AM as per 3GPP specs
  – RLC SM: simplified full-buffer model

• Features
  – PDUs and headers with real bits (following 3GPP specs)
  – Segmentation
  – Fragmentation
  – Reassembly
  – SDU discard
  – Status PDU (AM only)
  – PDU retx (AM only)
PDCP model

- **Simplified model supporting the following:**
  - Headers with real bytes following 3GPP specs
  - Transfer of data (both user and control plane)
  - Maintenance of PDCP SNs
  - Transfer of SN status (for handover)

- **Unsupported features**
  - Header compression and decompression using ROHC
  - In-sequence delivery of upper layer PDUs at re-establishment of lower layers
  - Duplicate elimination of lower layer SDUs at re-establishment of lower layers for radio bearers mapped on RLC AM
  - Ciphering and deciphering of user plane data and control plane data
  - Integrity protection and integrity verification of control plane data
  - Timer based discard
RRC Model features

• System Information (MIB, SIBs)
  – Generation at eNB
  – Reception and processing at UE
• Idle mode cell selection
• RRC Connection Establishment
• RRC Connection Reconfiguration, supporting:
  – SRB1 and DRB setup
  – SRS configuration index reconfiguration
  – PHY TX mode (MIMO) reconfiguration
  – Mobility Control Info (handover)
• UE Measurements
  – Event-based triggering supported (events A1 to A5)
  – Assumption: 1-to-1 PCI to EGCI mapping
  – Only E-UTRA intra-frequency; no measurement gaps
RRC Model architecture

- LteUeRrc: UE RRC logic
- LteEnbRrc + UeManager: eNB RRC logic
- Two models for RRC messages
  - Ideal RRC
    - SRBs not used, no resources consumed, no errors
  - Real RRC
    - actual RRC PDUs transmitted over SRBs
    - with ASN.1 encoding
RRC UE state machine
RRC eNB
State Machine

INITIAL_RANDOM_ACCESS
- rx RRC CONN REQ, Admit = false
- rx RA preamble
- rx X2 HANOVER REQ, Admit = true

HANOVER_JOINING
- rx RRC CONN REQ
- rx RRC CONN RECONF COMPLETED

CONNECTION_SETUP
- rx RRC CONN SETUP COMPLETED
- rx S1 PATH SWITCH REQ ACK

CONNECTION_REJECTED
- connection timeout
- connection rejected timeout

CONNECTED_NORMALLY
- handover trigger
- rx X2 HO PREP FAILURE
- rx RRC CONN RECONF COMPLETED
- reconfiguration trigger

HANOVER_PREPARATION
- rx X2 HO REQ ACK

HANOVER_LEAVING
- handover leaving timeout
- rx X2 UE CONTEXT RELEASE
- context destroyed

no context

handover joining timeout

context destroyed
Handover Support

- API for Handover Algorithms
  - Measurement configuration
  - Measurement report handling
  - Handover triggering

- Available handover algorithms:
  - No-op
  - A2-A4-RSRQ
  - Strongest cell handover (A3-based)
  - <your algorithm here>
Handover example scenario
Handover behavior

![Graph showing handover behavior over time with RSRQ values on the y-axis and time in seconds on the x-axis, with different symbols representing different serving cells and neighbor cells.]
NAS model

• Focus on NAS Active state
  – EMM Registered, ECM connected, RRC connected

• Logical interaction with MME
  – NAS PDUs not implemented

• Functionality
  – UE Attachment (transition to NAS Active state)
  – EPS Bearer activation
  – Multiplexing of data onto active EPS Bearers
    • Based on Traffic Flow Templates
    • Both UDP and TCP over IPv4 are supported

• Unsupported features
  – PLMN and CSG selection
  – Idle mode (tracking area update, paging…)
S1 interface model

- **S1-U (user data plane)**
  - Realistic model including GTP-U implementation
  - Data packets forwarded over GTP/UDP/IPv4
  - Communication over ns3::PointToPoint links

- **S1-C (control plane)**
  - Abstract model, no PDUs exchanged
  - Supported S1-AP primitives:
    - INITIAL UE MESSAGE
    - INITIAL CONTEXT SETUP REQUEST
    - INITIAL CONTEXT SETUP RESPONSE
    - PATH SWITCH REQUEST
    - PATH SWITCH REQUEST ACKNOWLEDGE
    - ERAB RELEASE INDICATION
**X2 interface model**

- **X2-U (data plane)**
  - GTP/UDP/IPv4 over ns3::PointToPoint (similar to S1-U)
- **X2-C (control plane)**
  - Messages as PDUs over ns3::PointToPoint links
  - Handover primitives:
    - HANDOVER REQUEST
    - HANDOVER REQUEST ACK
    - HANDOVER PREPARATION FAILURE
    - SN STATUS TRANSFER
    - UE CONTEXT RELEASE
  - SON primitives:
    - LOAD INFORMATION
    - RESOURCE STATUS UPDATE
S11 interface model

- abstract model
  - no GTP-C PDUs exchanged between MME and SGW
- Supported primitives:
  - CREATE SESSION REQUEST
  - CREATE SESSION RESPONSE
  - MODIFY BEARER REQUEST
  - MODIFY BEARER RESPONSE
  - DELETE BEARER REQUEST
  - DELETE BEARER RESPONSE
Simulation Configuration

• Done via ns-3 attribute system
• Several configurable attributes per LTE object
• Default attribute values can be configured:
  – Via input config file
  – Via command line
  – within simulation program
• Per-instance attribute values can be configured:
  – Within simulation program
  – Using GtkConfigStore
Simulation Output

- Lots of KPIs available at different levels:
  - Channel
    - SINR maps
    - pathloss matrices
  - PHY
    - TB tx / rx traces
    - RSRP/RSRQ traces
  - MAC
    - UL/DL scheduling traces
  - RLC and PDCP
    - Time-averaged PDU tx / rx stats
  - IP and application stats
    - Can be obtained with usual ns-3 means
    - FlowMonitor, PCAP traces, get stats directly from app, etc.
Example: 3GPP dual stripe scenario
Example: 3GPP dual stripe scenario
Execution time performance

LTE+EPC with real RRC

<table>
<thead>
<tr>
<th>UEs/HeNB</th>
<th>elaspedTime/simTime [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>
Memory consumption

LTE+EPC with real RRC

maxMemory [MB]

HeNBs

1 UEs/HeNB
3 UEs/HeNB
6 UEs/HeNB
10 UEs/HeNB
Testing

• Huge effort in testing:
  – Unit tests
    • Checking that a specific module works properly
  – System test
    • Checking that the whole LTE model works properly
  – Validation tests
    • Validating simulation output against theoretical performance in a set of known cases
  – Valgrind test coverage
    • Systematically check for memory errors
      – memory corruption, leaks, etc. due to programming errors
  – Build tests
    • Provided by ns-3 project for stable LENA code
    • Verify correct build on all supported platforms
    • LENA dev code tested daily on ubuntu
• LTE module documentation
  - Part of the ns-3 models library docs
The End

- Questions?