

CN Lab (17CSL57)

Exp 05-06: Performance Study of GSM/CDMA

Dr. Ram P Rustagi
Dept of CSE, KSIT
KRP-KSGI
rprustagi@ksit.edu.in

Ex07 Resources

- References:
 - <https://www.isi.edu/nsnam/ns/tutorial/index.html>
 - https://www.nsnam.org/docs/release/3.8/tutorial/tutorial_27.html#Building-a-Wireless-Network-Topology
 - <https://intronetworks.cs.luc.edu/current/html/ns2.html>
 - <https://www.isi.edu/nsnam/ns/doc/node29.html>
 - <https://www.cs.helsinki.fi/u/gurtov/papers/wireless-report.ps>
 - github.com/rprustagi/VTU-CNLab/Exp05_06

Lab05/06 Program

- Program 05

- Implement and study the performance of GSM on NS2/NS3 (Using MAC layer) or equivalent environment

- Program 06

- Implement and study the performance of CDMA on NS2/NS3 (Using MAC layer) or equivalent environment

Wireless Links and Performance

- Congestion:
 - TCP works on basic premise that congestion in the network causes packet loss.
 - In wireless networks, packet loss primarily occurs because of packet corruption
- Wireless links have high variability of delay and bandwidth
 - Cellular links have high latency and thus high RTT.
 - High link level queues also adds to RTT
 - Link level error recovery causes delay variation
- A sudden increase in delay causes TCP timeout

Cellular links

- GSM network
 - Experimentally found even good radio network has moderate jitter.
 - A moderate jitter results in increased TCP retransmit timer (timeout)
 - Prevents detection of packet loss quickly, and
 - increases recovery time.
- Handover
 - causes high jitter
 - results in spurious timeouts
- Desiderata:
 - Choose model parameters to reflect real work scenarios.

Modeling of Cellular Networks

- Purpose
 - Evaluate the effect of link level mechanisms on end to end transport protocol e.g. TCP
 - We don't want to represent link level transmissions and handoff.
 - Purpose is not to build a complex model of link level transmissions and handoffs.
- Modeling requirements for Transport layer performance
 - Change link characteristics
 - Introduce packet loss
 - Introduce delays to traffic

Wireless Simulation

- Wireless networks
 - There are no point to point links (as in wired n/w)
 - All configuration work requires
 - Setting up of nodes
 - Setting up of traffic
 - Wireless behaviour (wireless specific attributes), e.g.
 - Antenna type
 - Radio propagation model
 - Nodes themselves need to take care of queueing
 - In wired n/w, queueing is with link
 - Nodes have positions coordinates
 - Nodes move (need to define velocity)
 - Signal loss varies with distance

Modeling Cellular Links

- Typical GSM/GPRS
 - Downlink
 - Latency: 400-500ms
 - Bandwidth: 9.6kbps - 40kbps
 - Uplink: Latency/BW: 200ms, 10kbps
 - Cell coverage: cities: 100+ meters, rural: 1+ Kms.
- CDMA links
 - Latency: 100ms-150ms
 - BW: 500kbps - 1Mbps
 - Cell coverage: 100+m to few Kms
- Radio propagation
 - Acquiring a channel also causes delay
 - Every packet may require new channel allocation
 - Preserving battery power is equally a challenge

Performance Metrics

- Performance metrics for wireless links
 - Throughput
 - Delay
 - Fairness
- Important metric:
 - Goodput
 - defined as fraction of useful data from all data delivered
 - Goodput affects energy efficiency.
 - High goodput implies effective use of radio spectrum

Wireless Networks Attributes

- Propagation delay:
 - Distance / speed of light
 - In wired network, this is specified explicitly
- Bandwidth
 - Depends upon wireless model chosen, need to be specified (set) accordingly
 - Mac/802_11
 - defined by attribute `dataRate_`
 - default value in ns2 is 1mb.
- Adhoc wireless networks
 - Routing protocol must be configured
 - Needed to find the path from one node to another
 - example: AODV, DSDV, DSR

Wireless Networks Attributes

- Defining wireless config attributes (related to each other)
 - Use attributes of a single tcl object `opt`
- Wireless channel
 - `Channel/WirelessChannel`
 - Defines physical terrestrial wireless medium
 - `Channel/Sat`
 - Define satellite radio
- Radio propagation
 - `Propagation/TwoRayGround`
 - Takes into account ground reflection
 - Received power level is $1/d^4$
 - `Propagation/FreeSpace`
 - Received power level is $1/d^2$

Wireless Networks Attributes

- **Wireless node interface**
 - `Phy/WirelessPhy`
 - **Defines standard wireless interface**
 - `Phy/WirelessPhyEst`
 - **Standard wireless interface with additional options**
 - `Phy/Sat`
 - **Satellite specific**
 - `Mac/802_11`
 - **Specifies Wi-Fi (WLAN) Behaviour**
 - **Other values:** `Mac/Csma/Ca`, `Mac/802_3`
- **Queueing behaviour for each node**
 - `Queue/DropTail/PrioQueue`
 - `opt(ifqlen)` **corresponds to** `queue-limit` **for wired networks**

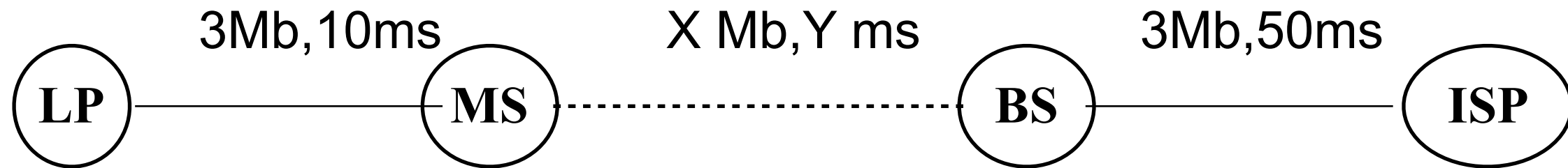
Wireless Networks Attributes

- Link Layer – `lltype`
 - LL specified behaviour of ARP packets
- Antenna types
 - `Antenna/OmniAntenna`
 - Defines standard omnidirectional antenna
 - Other antenna types
 - Parabolic, waveguide
- Layout
 - `opt(x)` and `opt(y)` defines layout dimension
 - By default these values are in meters

Wireless Networks Attributes

- Node movement
 - Nodes goes out of range when power loss becomes high, e.g.
 - At 250 m ($x:200, y:150 \rightarrow 200^2 + 150^2 = 250^2$)
 - Node moves out of range of current node
 - Default routing protocol kicks in
 - Searches for a new path
 - AODV takes about 50ms
 - DSR, DSDV may take much longer (seconds)

Modeling Cellular Networks



- Cellular network is between MS and BS
- Apply the cellular network characteristics parameters to link MS \longleftrightarrow BS
- TCP traffic between LP (low power) and ISP
 - Link between MS and LP is high speed, compared to Cellular link

Cellular Network Parameters

- Queue Management
 - RED (Random Early Detection)
 - Min Threshold (`minthresh`): 0
 - Max Threshold (`maxthresh`): 30
 - Adaptive (dynamically adjusts pkt drop rate) vs Plain
- Downlink details (BS→MS)
 - DL bandwidth [`bwDL (gsm)`] : 9600 bps
 - DL propagation delay [`propDL (gsm)`] : 500ms
- Uplink details (MS→BS)
 - UL bandwidth [`bwUL (gsm)`] : 9600 bps
 - UL propagation delay [`propUL (gsm)`] : 500ms

Cellular Network Parameters

- Link Characteristics
 - Length/duration of delays: (not used for good GSM)
 - `delayLen` : 0.3
 - Interval between delays (not used for good GSM)
 - `delayInt` : 0.3

Cellular Network Parameters

- Downlink details (BS→MS)
 - DL bandwidth $[bw_{DL} (gsm)] : 9600 \text{ bps}$
 - DL propagation delay $[prop_{DL} (gsm)] : 500ms$
 - Channel allocation delay: Uniform or Exponential
 - Uniform requires: Min and Max range
(0.16–0.19)
 - Exponential requires average value
 $allocDelayDL_{Avg} : 0.17$
 - Channel Hold Delay: Uniform or Exponential
 - $allocHoldDL_{Avg} : 3$

Cellular Network Parameters

- Downlink details (BS→MS)
 - Rate of errors (currently not configured)
 - `errRateDL`: 0
 - Burstiness coefficients of errors (currently not configured)
 - `errBurstDL`: 0 (0 - no burst,)
 - Queue Length of packets

Code Snippets (cellular.tcl)

```
# Active Queue Mgmt (AQM parameters)
# import config parameters GSM or CDMA radio
source ./ $prefix.params
set minthresh 30; # used for queueing RED
set maxthresh 0; # used for queueing RED
set adaptive 1; ; # used for queueing RED

set delayLen ""
set delayInt ""

# queue size
set ql(gsm) 10
```

Code Snippets (cellular.tcl)

```
# RED and TCP parameters
```

```
Queue/RED set adaptive_ $adaptive  
Queue/RED set q_weight_ 0.0  
Queue/RED set thresh_ $minthresh  
Queue/RED set maxthresh_ $maxthresh
```

```
# Set up TCP connection characteristics  
set pktSize 1460  
Agent/TCP set window_ $window  
Agent/TCP set packetSize_ $pktSize
```

Code Snippets (cellular.tcl)

```
# node creation
set nodes(isp) [$ns node]
$nodes(isp) label "isp"
set nodes(ms) [$ns node]
$nodes(ms) label "ms"
set nodes(bs) [$ns node]
$nodes(bs) label "bs"
set nodes(lp) [$ns node]; # low power device
$nodes(lp) label "lp"
```

Code Snippets (cellular.tcl)

```
# node creation
```

```
# link between Mobile node and Base station (mimicking)
```

```
$ns duplex-link $nodes(ms) $nodes(bs) 1 1  
RED
```

```
# link between Mobile node and low power (e.g. bluetooth)
```

```
$ns duplex-link $nodes(lp) $nodes(ms)  
3Mbps 10ms DropTail
```

```
# link between Base station and ISP (wired)
```

```
$ns duplex-link $nodes(bs) $nodes(isp)  
3Mbps 50ms DropTail
```

Code Snippets (cellular.tcl)

```
# mimic wireless link (radio) params
$ns bandwidth $nodes(bs) $nodes(ms)
$bwDL(gsm) simplex
$ns bandwidth $nodes(ms) $nodes(bs)
$bwUL(gsm) simplex
$ns delay $nodes(bs) $nodes(ms) $propDL(gsm)
simplex
$ns delay $nodes(ms) $nodes(bs) $propUL(gsm)
simplex
$ns queue-limit $nodes(bs) $nodes(ms)
$ql(gsm)
# delay characteristics setup
set delayerDL [new Delayer]
set delayerUL [new Delayer]
$ns insert-delayer $nodes(bs) $nodes(ms) $delayerDL
$ns insert-delayer $nodes(ms) $nodes(bs) $delayerUL
```



Code Snippets (cellular.tcl)

```
#set_delay values
set al_dl [new RandomVariable/Exponential]
$al_dl set avg_ $allocDelayDLAvg
set ah_dl [new RandomVariable/Exponential]
$ah_dl set avg_ $allocHoldDLAvg
#
set al_ul [new RandomVariable/Exponential]
$al_ul set avg_ $allocDelayULAvg
set ah_ul [new RandomVariable/Exponential]
$ah_ul set avg_ $allocHoldULAvg

$delayerDL alloc $ah_dl $al_dl
$delayerUL alloc $ah_ul $al_ul
```

Code Snippets (cellular.tcl)

```
#delay length and interval
set dist_len [new RandomVariable/
Exponential]
set dist_int [new RandomVariable/
Exponential]
if {$delayLen != "" && $delayInt != ""} {
    $dist_len set avg_ $delayLen
    $dist_int set avg_ $delayInt
    $ns after [$dist_int value] "insertDelay"
}

# define procedure `insertDelay'
```

Code Snippets (cellular.tcl)

```
proc insertDelay {} {  
    global ns dist_len dist_int delayerDL  
    delayerUL  
  
    $delayerDL block  
    $delayerUL block  
  
    set len [$dist_len value]  
    $ns after $len "$delayerUL unblock"  
    $ns after $len "$delayerDL unblock"  
    set next [expr $len + [$dist_int value]]  
    $ns after $next "insertDelay"  
}
```

Code Snippets (cellular.tcl)

```
# Setting up traffic
# create TCP Connection.
# Use selective Ack (like Select Repeat)
set tcp1 [$ns create-connection TCP/Sack1
$nodes(isp) TCPSink/Sack1 $nodes(lp) 0]

# define FTP Application
set ftp1 [[set tcp1] attach-app FTP]

# start the FTP application
$ns at 1.0 "[set ftp1] start"
```

External Programs

- Need 2 external programs
 - `getrc`
 - `raw2xg`
- These programs are in `ns-allinone-2.35.tar.gz`
 - `ns-allinone-2.35/ns-2.35/tcl/ex/wireless-scripts`
- Copy these programs in `bin` subdir of `$HOME`
- For general wireless study, use the programs
 - `mtp.tcl`, `runall-tr.cmd`

Code Snippets (cellular.tcl)

```
proc stop {} {  
    global nodes opt tf prefix  
    set wrap $opt(wrap)  
    set sid [$nodes($opt(srcTrace)) id]  
    set did [$nodes($opt(dstTrace)) id]  
    set GETRC "~/bin/getrc"  
    set RAW2XG "~/bin/raw2xg"  
    exec $GETRC -s $sid -d $did -f 0 $prefix.tr |\  
        $RAW2XG -s 0.01 -q > gsmplot.xgr  
    exec $GETRC -s $did -d $sid -f 0 $prefix.tr |\  
        $RAW2XG -a -s 0.01 -q >> gsmplot.xgr  
    #exec xgraph -t $prefix -x time -y packets  
    [set prefix]plot.xgr &  
    exit 0  
}
```

Config GSM (gsm.params)

```
set type gsm;      #type of link:
set allocDelayDLAvg 0.17;  # Avg time to
allocate a downlink channel
set allocHoldDLAvg 3;  # Avg time to hold a
downlink channel
set allocDelayULAvg 0.5;  # Avg time to
allocate a uplink channel
set allocHoldULAvg 0.2;  # Avg time to
hold a uplink channel
set bwDL($type) 9600
set bwUL($type) 9600
set propDL($type) 500ms
set propUL($type) 500ms
```

Config CDMA (cdma.params)

```
set type cdma;      #type of link:
set allocDelayDLAvg "";      # Avg time to
allocate a downlink channel
set allocHoldDLAvg "";      # Avg time to
hold a downlink channel
set allocDelayULAvg "";      # Avg time to
allocate a uplink channel
set allocHoldULAvg "";      # Avg time to
hold a uplink channel
set bwDL($type) 384000
set bwUL($type) 64000
set propDL($type) 100ms
set propUL($type) 150ms
```


GPRS Simulations

- **Good GPRS**

- allocLenDL 'U(0.16,0.19)' \
 - allocHoldDL 'U(2,5)' \
 - allocLenUL 'U(0.5,0.6)' \
 - allocHoldUL 'U(0.01,0.4)'

- **Mediocre GPRS**

- allocLenDL 'U(0.16,0.19)' \
 - allocHoldDL 'U(2,5)' \
 - allocLenUL 'U(0.5,0.6)' \
 - allocHoldUL 'U(0.01,0.4)' \
 - delayInt 'E(0.1)' \
 - delayLen 'E(0.1)' \

GPRS Simulations

- **Poor GPRS**

```
-allocLenDL 'U(0.16,0.19)' \  
-allocHoldDL 'U(2,5)' \  
-allocLenUL 'U(0.5,0.6)' \  
-allocHoldUL 'U(0.01,0.4)' \  
-delayInt 'E(0.3)' -delayLen 'E(0.3)' \  
-errRateUL 0.01 \  
-errBurstUL 0.3 \  
-errSlotUL 3 \  
-errRateDL 0.01 \  
-errBurstDL 0.3 \  
-errSlotDL 3
```

CDMA Parameters

- **Good CDMA params**

```
set bwDL(cdma) 384000
```

```
set bwUL(cdma) 64000
```

```
set propDL(cdma) .150
```

```
set propUL(cdma) .150
```

```
set ql(cdma) 20
```

```
-delayInt 'E(0.1)' \
```

```
-delayLen 'E(0.04)'
```

Run Simulation GSM/CDMA

- **For GSM simulation**

```
ns cellular.tcl gsm
nam cdma.nam
xgraph -t GSM -x time -y packets
gsmpplot.xgr
```

- **For CDMA simulation**

```
ns cellular.tcl cdma
nam cdma.nam
xgraph -t CDMA -x time -y packets
cdmaplot.xgr
```

- **Variation of GSM (Good, poor, moderate) params**
 - parameters-for-mtp.txt

Summary

- Understanding radio params
- Define config parameters for GSM
- Define config parameters for CDMA
- Invoke the cellular simulation with required cellular parameters.
 - Analyze the packets sent and ack received
 - Study the animation