# A Performance Perspective on Function Virtualization for Small Cells in 5G Networks

John N. Daigle

wcdaigle@olemiss.edu

Department of Electrical Engineering
University of Mississippi

# Overview

# Objective Statement

To share some thoughts on network functions virtualization

- The current state of the art as it applies to small-cell deployment
- Performance issues in wireless cellular systems
- Some things that can be done to help to realize some of the potential of small cell deployment

# Motivation for Small Cells and Virtualization

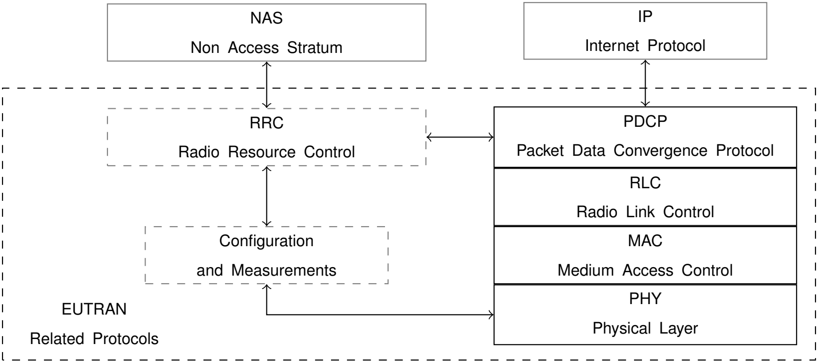
- Lower footprint and energy consumption
- Facilitate remote management and reduce operation complexity
- · · · long list
- Facilitates slicing: multiple virtual networks on single physical infrastructure.

  Each slice
  - is service- and vendor-specific
  - can be defined on-the-fly using SDN as a tool
  - can be provisioned on-the-fly using some kind of orchestrator
  - can be instantiated on-the-fly by some kind of orchestrator through the allocation of infrastructure resources
  - has resources that are isolated from all other virtual networks
  - enjoys standards-based, on-the-fly, self-optimization
- Slicing “potentially offers a viable mechanism for operators to extract the most potential from NFV/SDN, an opportunity of mind-boggling complexity.” (STL Partners, Mar 2017)

# Central Idea

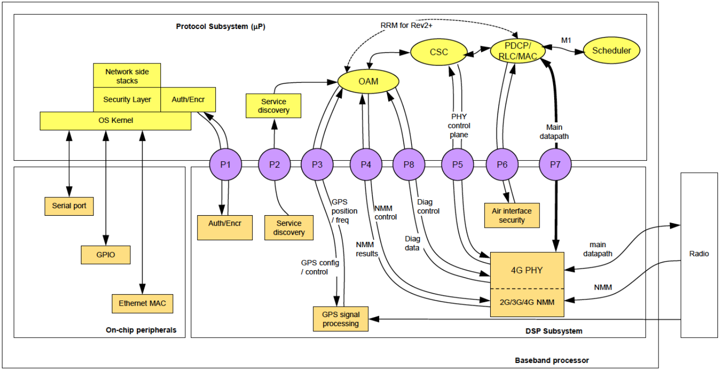
- Current LTE network architecture has only two elements
  - Evolved Packet Core (EPC)
  - E-UTRAN Node B (eNB).
- Function virtualization splits the functionality of eNB into two pieces
  - **PNF**, Functions implemented in a remote radio head that connects directly to user equipment over RF
  - **VNF**, A collection of software, or virtual functions, that implements all other functions needed to manage delivery of services
- Fronthaul: Mechanisms to communicate between the PNF and VNF
- Issues
  - How to organize the processing into separate functional components
  - How to decide which parts to virtualize
  - Decide where to host functionality to assure goals can be achieved
  - Decide how to host functionality to assure goals can be achieved

# EUTRAN–Air Interface Protocol for User Equipments (UE)



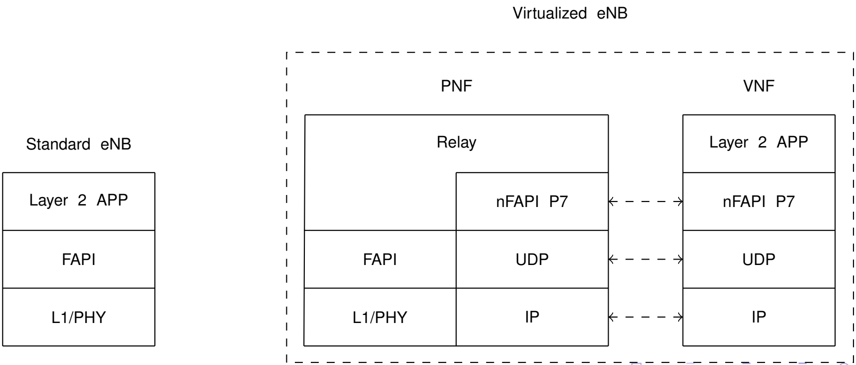

Functionality inside EUTRAN is the target functionality to virtualize!

# Organization of Functionality




SCF 082: SCF Functional Application Platform Interface (FAPI)

# Non-Virtualized versus Virtualized–P7 Example



# Which Services Benefit?

- Venue specific, e.g. sports events
- Enterprise networks
- Mobile telemedicine services
- Content delivery
- Mostly services that have not yet been thought up

What's the difference . . . we have most of this stuff now!

- Slicing
- Install networks on-the-fly when and where needed

# Environment

- State is a area of, say, 300 by 500 km
- A small number of major medical facilities accessible in the state
- Numerous small hospitals distributed around the state
- Generally sparse population, with hospitals relatively small
- Only limited special equipment and skills at small hospitals
- Some patients must be transported periodically to major medical facilities for specialized treatment
- Some patients need transportation urgently

# Telemedicine Application (continued)

Medical vehicle transporting multiple patients to distant medical center

- Route is through a sparsely populated rural area
  - Small communities along the route
    - Generally served by macro cells
    - Highly varying coverage due to large tree canopy
  - Some residences along the route have
    - Internet service
    - Small cells attached to Internet service to improve access

Communications between mobile and medical center

- Real-time two-way voice for coordination of services
- Video streaming of patient conditions in uplink
- Streamed sensor data of patient vital signs in uplink
- Bi-directional file transfer

# Service Delivery Protocol Needs

Need ability to

- Capitalize on all available communication resources
  - Maximal possible content transfer given resources
  - Best integrity of received content
- Survive random periods of communication drop-outs
- Continue interrupted activities seamlessly
- Cope with highly variable one-way and round-trip delays

Fountain codes, specifically RaptorQ codes, may be helpful.

# Fountain Coding and University of Mississippi

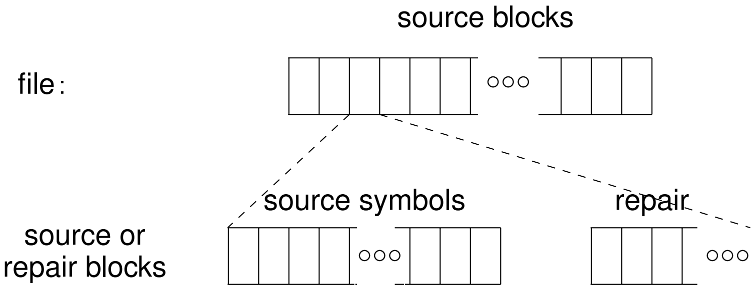
We are passionate about understanding

- fountain codes and their derivatives, specifically RaptorQ
- the potential role of fountain codes for all types of content delivery in communications networks of all types
- development of prototypes of content delivery protocols to address specific paradigms

We have interesting results in the areas of

- Peer-to-peer content delivery
- Content transfer over mobile cellular phone hotspots (plural)
- Content transfer over simultaneous DSL and cell-phone hot spot
- Deep space communications, e.g. Mars to Earth

# High-level View of Raptor Coding



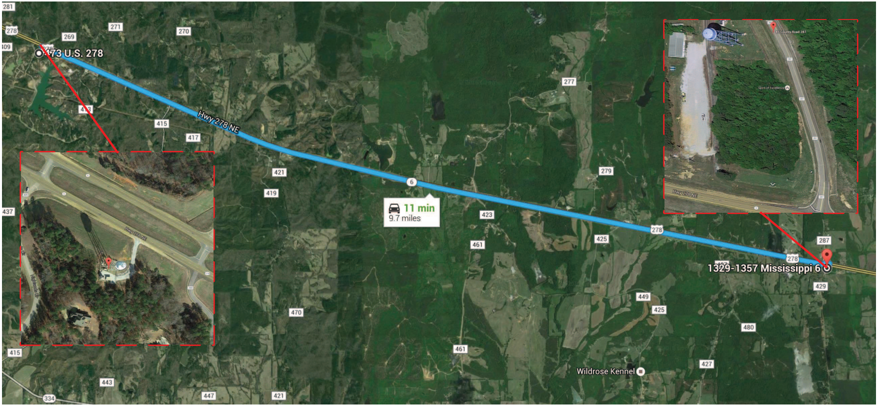
# High-level View of Raptor Coding (concluded)

- Content into $K$ chunks called *source symbols* (Ss)
- *Encoded symbols* (Es) are pseudorandom linear combinations of Ss
  - $\rightarrow$ An encoding symbol is a linear equation!
  - $\rightarrow$ Source can generate $\gg K$ encoded symbols
  - $\rightarrow$ Coding routine yields linear independence of $\approx$ any $K$ Es
- Receiver collects $K + \epsilon$ linear equations (Es)
- Receiver throttles sender
- Symbol identifiers (ESIs) allow formulation of linear system
- Receiver solves linear system to recover source symbols

# Stationary Measurements Tethered over LTE Hotspot

| File Transfer # | SFTP Time(s) | RaptorQ Time (s) |
|---|---|---|
| 1 | 22.966 | 23.069 |
| 2 | 26.007 | 22.945 |
| 3 | 25.922 | 22.724 |
| 4 | 23.782 | 22.921 |
| 5 | 25.369 | 22.805 |
| 6 | 26.291 | 22.964 |
| 7 | 24.501 | 23.153 |
| 8 | 24.139 | 22.891 |
| 9 | 23.448 | 23.060 |
| 10 | 24.736 | 23.468 |
| Mean Time (s) | 24.716 | 23.000 |
| Mean Rate (Mb/s) | 7.625 | 8.194 |

# A Segment of Mississippi Highway



# Measured data transfer rates for consecutive runs

| Data rates in Mb/s | | | | |
|---|---|---|---|---|
| Transfer # | SFTP (W to E) | SFTP (E to W) | RaptorQ (W to E) | RaptorQ (E to W) |
| 1 | 6.731 | 13.461 | 19.977 | 4.770 |
| 2 | 12.564 | 11.778 | 4.967 | 15.640 |
| 3 | 5.093 | 6.980 | 4.094 | 13.872 |
| 4 | 6.282 | 5.543 | 5.442 | 9.936 |
| 5 | 6.731 | 1.327 | 6.378 | 2.492 |
| 6 | 7.248 | 3.846 | 6.503 | 7.363 |
| 7 | 1.545 | 6.079 | 5.187 | 5.893 |
| 8 | 4.959 | 6.980 | 5.068 | 4.127 |
| 9 | 4.383 | 6.282 | 9.997 | 4.678 |
| 10 | 10.470 | 6.282 | 9.101 | 2.392 |
| 11 | 8.974 | 2.771 | 17.610 | 6.731 |
| 12 | 15.705 | | 17.013 | |
| | | | 6.929 | |
| Average | 5.410 | 4.430 | 7.027 | 5.056 |

# Conclusions About Present Mobile Reality

- Provider maps show 100% LTE coverage
- Average transfer rates for the exact same file over route
  - Range from about 3 to 20 Mb/s
  - Depend upon the protocol
  - Vary widely from location to location
- US 278 is a major road; what about Rte 334?
  - Can get steady but slow file transfer using RaptorQ
  - SFTP doesn't work at all; just freezes
- Difficult to see how performance guarantees can be made with macro cell infrastructure

# Can Virtualization and Small Cells Help?

Small Cells? Probably depends on

- available or achievable landline coverage
- achievable density based on cost
- ability to respond quickly to coverage holes

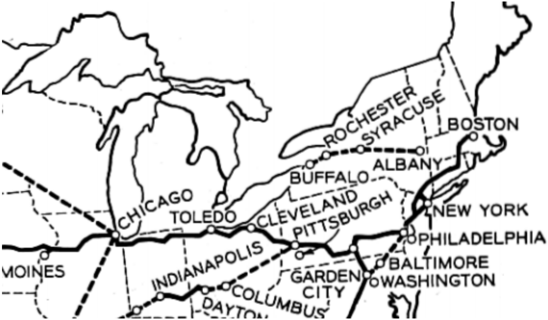
Virtualization? Probably–depends upon

- economically achievable fronthaul bit rates and latencies
- controlling costs of remote radio heads

Remember

- Coverage is supposed to be ubiquitous
- There are many roads
- Population is sparse; 3 million people over 150,000 km$^2$

# The Importance of Hands-on



# Develop Expertise with Functional Virtualization

- Get a kubernetes cluster up and running
  - This can be done cheaply using single board computers, e.g. Raspberry Pis
  - Contains functions to orchestrate network functional virtualization using containers
  - Processing nodes run Docker to process containers
- Define some realistic processing scenarios
- Deploy some parallel processing chains connected
  - Locally
  - Through Internet and openly available platforms
- Develop models to evaluate latencies
- Perform measurements to gain some insight

# Serious Effort on Problem Development

- Definition of real scenarios need continuing effort
- Develop simple scenarios that mimic real virtualized processing and communication needs
- Gain some insight on ways to achieve isolation of services within a shared resource environment
- Develop performance models that capture at least some important properties for shared resouce management
- Test performance model results against measurement results from implementations

# Conclusions

- Remember slicing "potentially offers a viable mechanism for operators to extract the most potential from NFV/SDN, an opportunity of mind-boggling complexity."
  $\Rightarrow$ Don't forget the"mind-boggling complexity" part!
- There appears to be little chance that ubiquitous performance guarantees
  - Have any hope in absence of small cell deployment
  - Can be achieved without some level of functional virtualization
- Ubiquitous service guarantees are likely to be expensive
- Enormous opportunity for performance evaluation towards real-time management
- Little chance of significant contribution without serious problem definition effort