Modeling NASPInet Data Flows

Rakesh Bobba
University of Illinois, Urbana Champaign

Joint work with Ragib Hasan and Himanshu Khurana

NASPI Working Group Meeting, Charlotte, NC. Oct 16-17, 2008.

Funded by NSF through TCIP and ONR through NCASSR

Motivation and Goals

- **Motivation**: How can we design and implement a scalable PMU data sharing NASPInet?
  - what kind of bandwidth is needed for NASPInet?
  - how do latency constraints affect bandwidth provisioning and security guarantees?
  - will it scale to multiple applications (current/future) using data from thousands of PMUs?

- **Goal**: To build a modeling framework that will analyze and validate network and storage architectures as well as security technologies suitable for PMU data sharing in a scalable manner

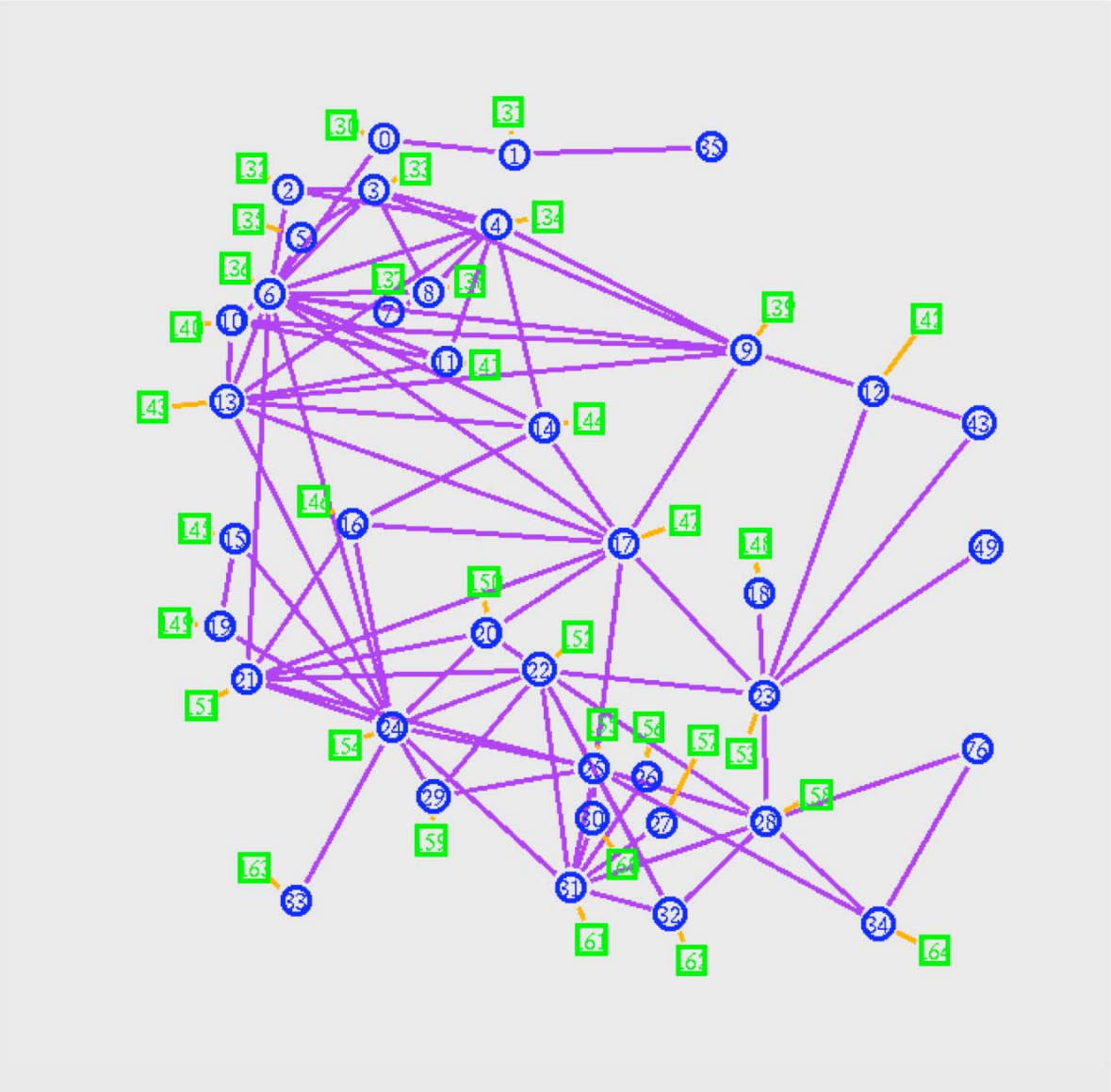
Why a Simulation Based Framework?

- Allows for extensive design space exploration for NASPInet
  - Network Architectures
    - point-to-point vs. IP multi-cast vs. publish/subscribe
  - Storage and Processing
    - distributed vs. centralized
  - Security
    - architecture dependent requirements
    - bandwidth and latency constraint

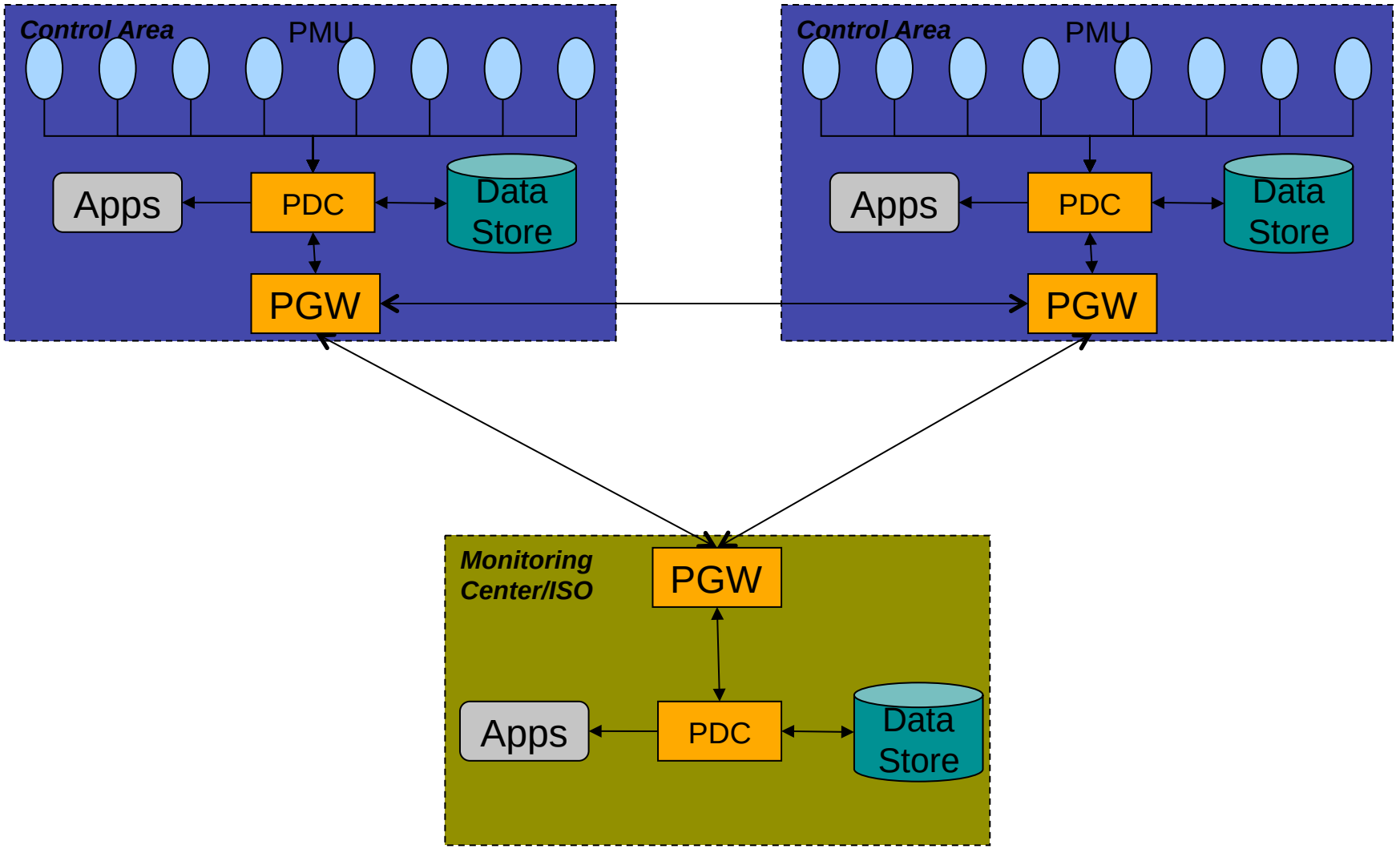
First Study: WECC Point-to-Point

- WECC topology
  - 35 PGWs
  - 1 PDC per PGW
  - 200 PMUs per PDC/PGW on average
  - 7387 PMUs in total
- Point-to-point communication links
  - 56 to 128 Kbps between PMUs and PDC
  - 6Mbps / 12Mbps between PGWs
- Standard security mechanisms
  - hop-by-hop authentications
  - MAC/signatures for authentication
- Distributed storage
  - everybody stores all data

Study Topology



Architecture



End-To-End Latency – Dedicated Links

- Scenario
  - PGW1 – 248 PMUs, PGW2 – 193 PMUs
  - 1000 miles apart, **dedicated** comm. link
  - Authentication Scheme – HMAC-SHA1
- Only 3ms increase in latency – to send extra 20 bytes MAC code

Average End-to-End Latency (Var. ~ 0)

	12 Mbps	12 Mbps with Authentication	7.23 / 5.72 Mbps	7.23 / 5.72 Mbps with Authentication
PGW1 -> PGW2	85ms	88ms	98ms	101ms
PGW2 -> PGW1	76ms	79ms	93ms	96ms

Meets latency requirements of Class A applications!

Bandwidth

- Average B/W out of a node ~ 29 Mbps
- Total B/W in the network ~ 1000Mbps simplex
- PGW1
  - has 15 outgoing links
- PGW2
  - has 7 outgoing links

Inefficient use of bandwidth when using dedicated links

End-To-End Latency – Shared Links

- Scenario
  - PGW1 – 248 PMUs, PGW2 – 193 PMUs
  - 1000 miles apart, **shared** comm. link
  - Pareto on/off traffic source adjusted to use available b/w.
  - Authentication Scheme – HMAC-SHA1

– Small increase in average latency but is variable.

End-to-End Latency

	12 Mbps		**12 Mbps w/ Authentication**	
	(Avg./Std. Dev.)	**(Min./Max)**	**(Avg./Std. Dev.)**	**(Min./Max)**
PGW1 -> PGW2	88ms / 4ms	85ms / 98ms	92ms / 6ms	88ms / 107ms
PGW2 -> PGW1	80ms / 4ms	76ms / 88ms	85ms / 5ms	79ms / 97ms

Meets latency requirements of Class A applications but variable latency

Time Alignment vs. Latency

- Time Alignment at Source vs. Time Alignment at Destination
  - Advantages
    - amortizes security cost over PGW-PGW link
      - signatures come into realm of possibility!!
  - Disadvantages
    - creates more bursty traffic
      - increases latency
- Examples:
  - PGW1 -> PGW2
    - latency with time alignment – 85ms
    - latency without alignment – 66ms
  - PGW3 -> PGW1 (319 PMUs, 1000 mile link)
    - latency with time alignment – 109ms
    - latency without time alignment – 72ms

Storage

- PGW1
  - 1MB generated per second
    - 84GB per day!
    - > 1 TB every 2 weeks!!
  - 12MB total per second (recvd. + generated)
    - 1TB per day!!
- 27MB generated per second in the network!
- Need to think about storage and data management issues
  - especially if Class B, C, D applications are to be served from storage

Future Work

- Analyze bandwidth requirements and feasible latency guarantees
  - using multi-hop network
  - store and forward architectures (for classes B, C & D)
- Analyze feasible security mechanisms for above cases
- Analyze trade-offs between time alignment strategy, latency and feasible security