Anomalies in Optimal Rate-control and Scheduling Protocols for Cognitive Radio Networks

Vinay Kolar [1] V. Munishwar [2] N. Abu-Ghazaleh [1,2]

[1]Department of Computer Science
Carnegie Mellon University, Qatar

[2]Department of Computer Science
State University of New York, Binghamton

CogART, 2010

Introduction

Cognitive Radio Networks are promising

- Large spectrum can meet growing capacity demands
- WiFi, Mesh, Sensor Networks based on CRNs

But, large spectrum $\neq$ large capacity to end-applications

- Primary user activities
- Resource allocation among secondary nodes

Paper focus: How do we design and realize efficient protocols?

How do we design and realize efficient protocols?

Two broad categories:

- Heuristic protocols: Fast design cycle, low-complexity
- Theoretical models: Systematically formulate, derive insights

Well-known theoretical methodology: Generalized Network Utility Maximization (GNUM)

- Formulate optimal network models
- Derive optimal protocols, network layers
- Demonstrated in real-systems (e.g. FAST-TCP)

Contribution

Varieties of GNUM formulations have been studied in CRN

- Power-control, Scheduling, etc
- Joint optimization of source-rate, routing and scheduling

Paper focus:

- Can we realize Joint source-rate, routing and scheduling in systems?
- What are the anomalies when translating theory into systems?

Joint Source-rate, routing and scheduling

Deriving optimal protocols in GNUM: A three step recipe

- Formulate the primal optimization problem
- Decompose into sub-problems
  - Structure of sub-problems $\longrightarrow$ Functions carried out at physical entities
- Identify message-passing between physical entities

Joint Source-rate, routing and scheduling

Deriving optimal protocols in GNUM: A three step recipe

- Formulate the primal optimization problem
- Decompose into sub-problems
  - Structure of sub-problems $\rightarrow$ Functions carried out at physical entities
- Identify message-passing between physical entities

Primal Problem

1. Maximize application utilities (e.g., throughput, fairness, ...) such that

Maximize $\sum_{\forall k \in \mathcal{K}} U(r_k)$
s.t.

2. Packets are routed from src to dest

$$x_a^k \leq \sum_{b:(a,b)\in\mathcal{L}} \sum_{c\in\mathcal{C}} f_{ab,c}^k - \sum_{b:(b,a)\in\mathcal{L}} \sum_{c\in\mathcal{C}} f_{ba,c}^k$$

3. Schedule for links are feasible (an NP-hard problem)

$$\mathbf{f} \in \Pi$$

4. Don't schedule when primary is on

$$\sum_{k\in\mathcal{K}} t_{i,c}^k + P_{i,c} \leq 1$$

Primal Problem

1. Maximize application utilities (e.g., throughput, fairness, ...) such that

Maximize $\sum_{\forall k \in \mathcal{K}} U(r_k)$ s.t.

2. Packets are routed from src to dest

$$x_a^k \leq \sum_{b:(a,b)\in\mathcal{L}} \sum_{c\in\mathcal{C}} f_{ab,c}^k - \sum_{b:(b,a)\in\mathcal{L}} \sum_{c\in\mathcal{C}} f_{ba,c}^k$$

3. Schedule for links are feasible (an NP-hard problem)

$$\mathbf{f} \in \Pi$$

4. Don't schedule when primary is on

$$\sum_{k\in\mathcal{K}} t_{i,c}^k + P_{i,c} \leq 1$$

Dual decomposition

Use standard Lagrangian Dual Decomposition method.

Two subproblems

1. Source-rate maximization problem

   $D_1(\mathbf{q}) = \max_{\mathbf{r} \geq 0} \sum_{k \in \mathcal{K}} U(r_k) - \sum_{k \in \mathcal{K}} r_k q^k_{\mathrm{src}(k)}$

   - Completely distributed

2. Joint routing and scheduling problem

   $D_2(\mathbf{q}) = \max_{\mathbf{f} \geq 0} \sum_{(a,b) \in \mathcal{L}} \sum_{c \in \mathcal{C}} f^k_{ab,c} \max_{k \in \mathcal{K}} (q^k_a - q^k_b)$

   - Weight for a flow = *Congestion price differential* at link end-points
   - Needs message passing at each time-slot

Dual decomposition

Use standard Lagrangian Dual Decomposition method.

Two subproblems

1. Source-rate maximization problem

   $D_1(\mathbf{q}) = \max_{\mathbf{r} \geq 0} \sum_{k \in \mathcal{K}} U(r_k) - \sum_{k \in \mathcal{K}} r_k q^k_{\mathrm{src}(k)}$

   - Completely distributed

2. Joint routing and scheduling problem

   $D_2(\mathbf{q}) = \max_{\mathbf{f} \geq 0} \sum_{(a,b) \in \mathcal{L}} \sum_{c \in \mathcal{C}} f^k_{ab,c} \max_{k \in \mathcal{K}} (q^k_a - q^k_b)$

   - Weight for a flow = *Congestion price differential* at link end-points
   - Needs message passing at each time-slot

Dual decomposition

Use standard Lagrangian Dual Decomposition method.

Two subproblems

1. Source-rate maximization problem
   $D_1(\mathbf{q}) = \max_{\mathbf{r}\geq 0} \sum_{k\in\mathcal{K}} U(r_k) - \sum_{k\in\mathcal{K}} r_k q^k_{\mathrm{src}(k)}$
   - Completely distributed

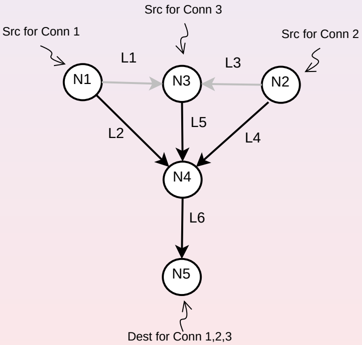
2. Joint routing and scheduling problem
   $D_2(\mathbf{q}) = \max_{\mathbf{f}\geq 0} \sum_{(a,b)\in\mathcal{L}} \sum_{c\in\mathcal{C}} f^k_{ab,c} \max_{k\in\mathcal{K}}(q^k_a - q^k_b)$
   - Weight for a flow = *Congestion price differential* at link end-points
   - Needs message passing at each time-slot

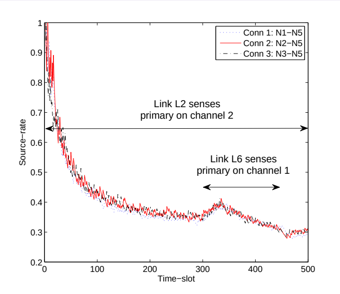
Optimal algorithms

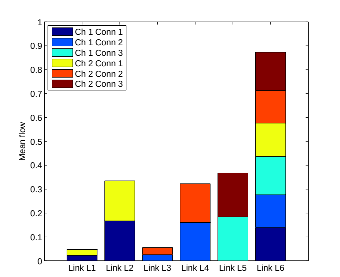
At each time-slot

1. Each node updates *congestion-prices*
2. Source node locally solves the Source-rate problem.
3. Centralized scheduler computes schedule
   - Each link computes congestion price differential for each connection.
   - Sender and receiver will transmit Primary Usage Map
   - Scheduler computes optimal flow for each link, and disseminates

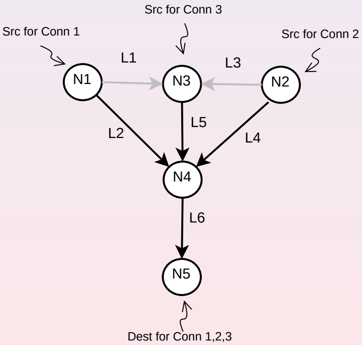
Example scenario

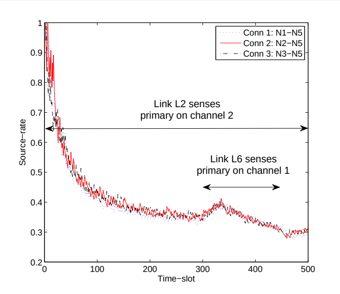


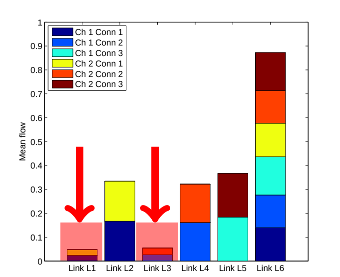




Example scenario







Closer look at scheduling

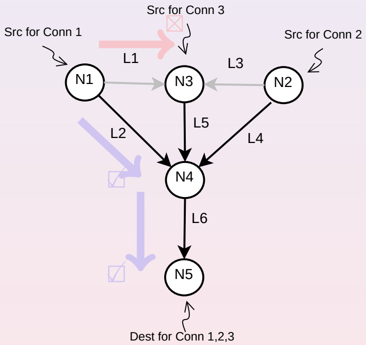
Scheduling works on the basis of back-pressure

Scheduler balances queue-length differences

- Links with larger price differential are given priority

System issue 1: What happens at inactive links of a connection?

- *Spurious pressure* at inactive links is necessary to push packets through actual link



Example scenario

Closer look at scheduling

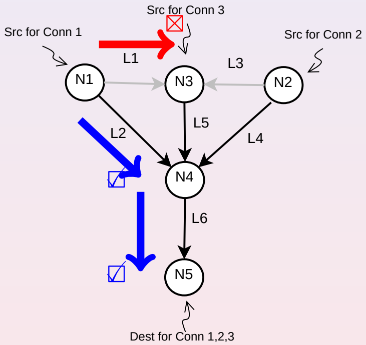
Scheduling works on the basis of back-pressure

Scheduler balances queue-length differences

- Links with larger price differential are given priority

System issue 1: What happens at inactive links of a connection?

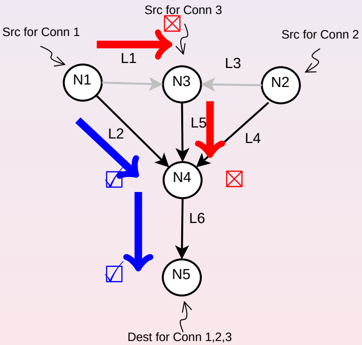
- *Spurious pressure* at inactive links is necessary to push packets through actual link

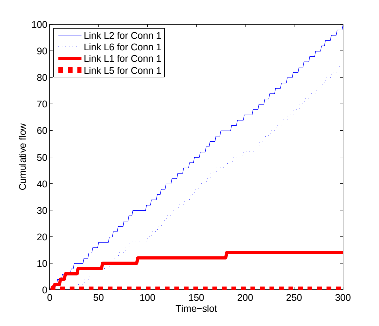


Example scenario

Spurious pressure points

Packets are blocked at wrong nodes!





Example scenario

Choice of time-slot

Scheduling happens in microseconds or milliseconds

Message passing overhead too high

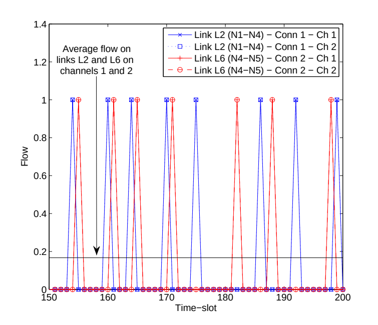
Practical solutions:

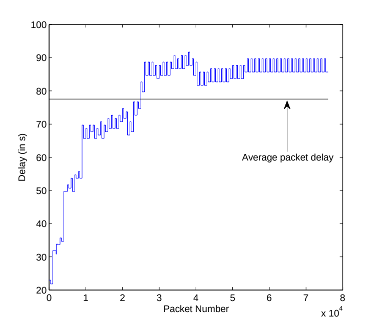
- Exchange messages at coarse time-granularity (say, 1 sec)
- Reuse schedules

System issue 2: Bursty schedules

Links are turned on and off for long times

Bursty schedules





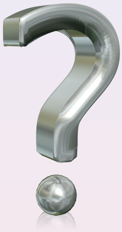
Prohibitively large delays, unfairness, large buffer spaces

Conclusions and future work

- Extended Joint Source-rate, routing and scheduling GNUM model for CRNs
- Analyzed system issues and impact of back-pressure based GNUM models
  - *Spurious pressure points* induce packet losses
  - *Bursty schedules* result in large delays, buffer-spaces and unfairness
  - *Link-pruning* can help in designing fast-schedulers

Future work:

- Evaluation in simulation and SDRs



Thank you.

For further information, please contact:
Vinay Kolar: vkolar@cmu.edu

Misc - Scheduling problem

Problem 1:

Compute list of all links that can be scheduled

- Maximal Independent Set (MIS) problem on multi-channel conflict graph of CRN

Each MIS M_i: (link, channel) pair

Problem 2:

Schedule each MIS

- Total time for all MIS ≤ 1
- Time for a (link, channel) should respect MIS membership

System issue 1: Computing MIS is NP hard

- Number of MISs grow exponentially if number of edges, channels grow

Misc - Scheduling problem

Problem 1:

Compute list of all links that can be scheduled

- Maximal Independent Set (MIS) problem on multi-channel conflict graph of CRN

Each MIS M_i: (link, channel) pair

Problem 2:

Schedule each MIS

- Total time for all MIS ≤ 1
- Time for a (link, channel) should respect MIS membership

System issue 1: Computing MIS is NP hard

- Number of MISs grow exponentially if number of edges, channels grow

Misc - Scheduling problem

Problem 1:

Compute list of all links that can be scheduled

- Maximal Independent Set (MIS) problem on multi-channel conflict graph of CRN

Each MIS M_i: (link, channel) pair

Problem 2:

Schedule each MIS

- Total time for all MIS ≤ 1
- Time for a (link, channel) should respect MIS membership

System issue 1: Computing MIS is NP hard

- Number of MISs grow exponentially if number of edges, channels grow

Misc - Scheduling problem

Problem 1:

Compute list of all links that can be scheduled

- Maximal Independent Set (MIS) problem on multi-channel conflict graph of CRN

Each MIS M_i: (link, channel) pair

Problem 2:

Schedule each MIS

- Total time for all MIS ≤ 1
- Time for a (link, channel) should respect MIS membership

System issue 1: Computing MIS is NP hard

- Number of MISs grow exponentially if number of edges, channels grow