On Computing Resilient Virtual Backbone in Cognitive Radio Networks

Yi Hong[1], Donghyun Kim[2], Deying Li[1], Jiaofei Zhong[3], and Alade O. Tokuta[2]

[1]Renmin University of China, Beijing, China
[2]North Carolina Central University, Durham, USA
[3]University of Central Missouri, Warrensburg, USA

Presented by Donghyun Kim on May 5, 2014 at International Conference on Information Science & Applications (ICISA 2014), Seoul, South Korea.

NORTH CAROLINA CENTRAL UNIVERSITY FOUNDED 1910

UNIVERSITY OF CENTRAL MISSOURI. LEARNING TO A GREATER DEGREE

Background

- licensed spectrum bands, e.g. cell phone network
  - lower utilization, especially rural area

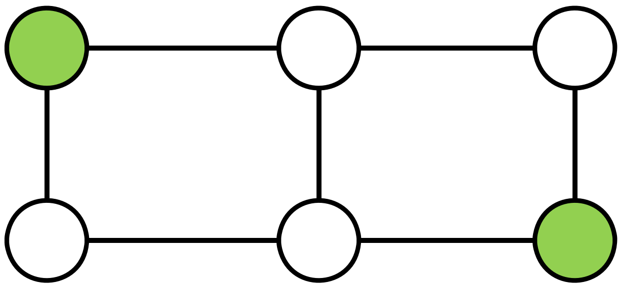
- cognitive radio network (CRN)
  - primary users (PUs)
    - rightful (paid) users of cell phone spectrum bands
  - cognitive users (CUs)
    - opportunistically use idle licensed spectrum bands for communication
    - must release the bands for the primary user once active
  - a supplementary network: a way to provide communication capacity for resource hungry unlicensed spectrum band users

How CRN Works?

- CRN is a temporal network of CUs using licensed bands
- each CU detects any idle licensed spectrum bands
  - if two neighboring CUs share a common idle band, they can use it for the communication
  - once PU of the idle band becomes active, the CUs have to stop using the licensed band for the PU
- energy-efficiency is still an important issue of CRN
  - frequently, each CU is a battery-operated mobile node
- however, routing information is likely to be invalidated by PU activities even in static CRN

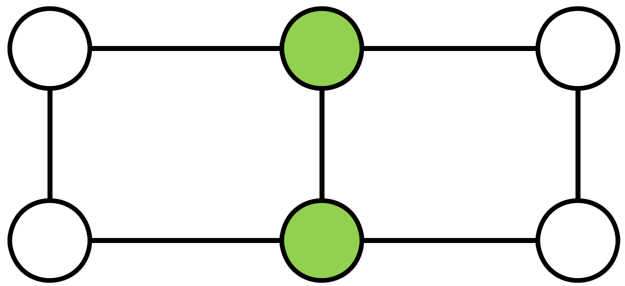
Dominating Set

a **dominating set (DS)** is subset of nodes such that
(a) each node is either in the subset or
(b) neighboring to a node in the subset



Connected Dominating Set

a subset of nodes is a **connected dominating set (CDS)** if
(a) it is a DS and
(b) the sub-graph induced by the subset is connected



Maximal Independent Set and CDS

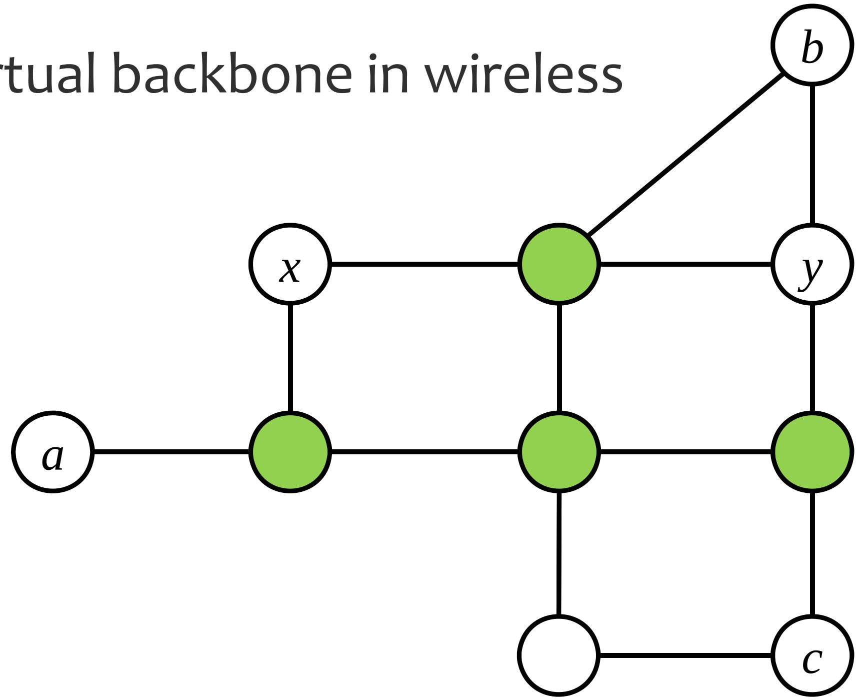
- **independent set (IS)** of G
  - a subset of $V(G)$ such that no two nodes in the subset are adjacent in G

- **maximal independent set (MIS)** of G
  - an independent set $I$of G such that for any $v \in V(G) - I$ $I \cup \{v\}$ is not an independent set

- an MIS is a DS
- computing an MIS and add more nodes to the MIS is a popular to compute a CDS

Virtual Backbone and CDS

- virtual backbone
  - connected subset of nodes responsible for message routing
  - CDS can serve as a virtual backbone in wireless networks



Advantage of Virtual Backbone

regular (flooding-based) routing	routing over virtual backbone
• redundant • heavy collision and interference overhead • energy inefficient	• smaller routing path search space • any routing scheme becomes more efficient

- size is an important metric
- computing a minimum CDS is NP-hard
  - cannot expect to compute an optimal solution in polynomial time
  - polynomial time approximation algorithm (sub-optimal algorithms with performance guarantee) is a popular subject

Motivation

- most virtual backbone construction algorithms for wireless networks focus on
  - minimizing size for efficiency
  - considering battery level of nodes for lifetime maximization
- computing virtual backbone in CRNs [4]
  - given a node-weighted (remaining energy level) network graph find a CDS such that
    - objective 1: maximize minimum node weight in the CDS
    - objective 2: minimize the size of the CDS

Motivation – cont'

- lifetime of a CDS in wireless network is mostly affected by battery-lifetime
- however, CRN is a temporal network
  - depending on PU activities, link availability changes
  - lifetime of CDS in CRN is highly dependent on PU activities
- better formulation for maximum lifetime (fault-tolerant) CDS would be
  - consider an edge-weighted (expected lifetime of link connectivity) graph
  - find a spanning tree of the graph such that
    - objective 1: maximize the minimum edge weight of the tree
    - objective 2: minimize the number of the non-left nodes (CDS nodes) in the tree

Notations

- $G = (V, E)$ is a graph with node set $V = V(G)$ and edge set $E = E(G)$
- Given $V' \subseteq V$, $G[V']$ is a subgraph of G induced by V'
- Given $E' \subseteq E$, $G[E']$ is a subgraph of G induced by E'
- $CU = \{CU_1, \cdots, CU_n\}$: $n \geq 1$ CUs
- $PU = \{PU_1, \cdots, PU_m\}$: $m \geq 2$ PUs
- $C = \{c_1, \cdots, c_l\}$: the set of available spectrum bands
- C_i is the set of spectrum bands licensed to PU_i
  - C_i and C_j are not necessarily disjoint for each i and j pair
  - at any moment, PU_i is either actively using $c_h \in C_i$ or not

Problem Definition

- suppose $A_j \subseteq C$ is the subset of channels to CU_j
- by [5], the activities of PU_j is modeled as a continuous time semi-Markov process
  - $X_{(c,j)}$, the time duration some channel c is available to CU_j, $\lambda_{(c,j)} e^{-\lambda_{(c,j)} X_{(c,j)}}$ for each $j = 1,2,\cdots,m$
  - $E\left(X_{(c,j)}\right) = \lambda_{(c,j)}$ is the expected value of $X_{(c,j)}$: known
- Definition 1. Lifetime of a Communication Link
  - consider C_i and C_j sharing a set of commonly available channels $c_1, \cdots, c_p$
  - the expected lifetime of the communication link is $\rho\left(e_{(i,j)}\right) = \max_{1 \leq q \leq p} \{\min[E(X_{(q,i)}), E(X_{(q,j)})]\}$

Problem Definition – cont'

- Definition 2. Lifetime of a Connected Network
  - given a connected network G, its lifetime is $\rho(G) = \max\{\rho \mid$ deleting all communication links with expected lifetime less than ρ cannot cause G disconnected$\}$
- given a positive integer k, a graph G if k-edge-connected if G is connected after removing any combination of k edges
- Definition 3. kECkDS
  - given a graph $G = (V, E)$, a subset D of V is a k-edge-connected k-dominating set of G if $G[D]$ is
    - k-edge-connected, and
    - for each node u in $V - D$, u is connected to at least k nodes in D in the original graph G

North Carolina Central University Founded 1910
University of Central Missouri
Learning to a Greater Degree

Problem Definition – cont'

- Definition 4. Lifetime of D, a kECkDS
  - internal lifetime: the minimum amount of time that makes kECkDS disconnected
  - external lifetime: the minimum amount of time that any node outside kECkDS is disconnected from kECkDS
  - lifetime of kECkDS: minimum of internal lifetime and external lifetime, i.e. the minimum time kECkDS loses its function

$$\rho(D) = \min[\rho(D_{in}), \rho(D_{out})]$$

- Definition 5. MLSVB
  - given a CRN $G = (V, E)$ of n CUs and a positive constant k, MLSVB is to find a CDS D of G such that
    - D is kECkDS of G
    - lifetime of D is maximized, and
    - size of D is minimized.

North Carolina Central University Founded 1910
University of Central Missouri
Learning to a Greater Degree

MLSVBA: Heuristic Algorithm for MLSVB in CRN

- maximum lifetime sturdier virtual backbone algorithm
  - a 3-stage polynomial time heuristic algorithm

lifetime of edges determined based on the history

Algorithm 1 MLSVBA $(G = (V, E), L = \{\rho(e) | \forall e \in E\})$

1: $G^{(1)} \leftarrow$ TRIMMER (G, L).
2: $G^{(2)} \leftarrow$ LIFETIME-MAXIMIZER $(G^{(1)}, L)$.
3: $S \leftarrow$ SIZE-MINIMIZER $(G^{(1)}, G^{(2)})$.
4: Return S.

NORTH CAROLINA CENTRAL UNIVERSITY FOUNDED 1910
UNIVERSITY OF CENTRAL MISSOURI. LEARNING TO A GREATER DEGREE

MLSVBA: Heuristic Algorithm for MLSVB in CRN – cont’

- TRIMMER
  - gradually remove all edges whose lifetimes are expected to be short such that the residual graph is still k-edge-connected

Algorithm 2 TRIMMER (G, L)

```
Let ρ_1 < ··· < ρ_l be the list of distinct lifetime levels in L.
E^(1) ← E.
for i = 1 to l do
    E_i = {e | ρ(e) = ρ_i and e ∈ E^(1)}.
    if the graph induced by (V, E^(1) \ E_i) is still k-edge connected then
        E^(1) ← E^(1) \ E_i.
    else
        break;
    end if
end for
Return G^(1) = (V, E^(1)).
```

MLSVBA: Heuristic Algorithm for MLSVB in CRN – cont’

- LIFETIME-MAXIMIZER
  - further remove any short-living edges such that the residual graph include a feasible solution

Algorithm 3 LIFETIME-MAXIMIZER $(G^{(1)}, L)$

Let $\rho_1 < \cdots < \rho_l$ be the list of distinct lifetime levels in L.
$E^{(2)} \leftarrow E^{(1)}$.
for $i = 1$ to l **do**
  $E_i = \{e | \rho(e) = \rho_i$ and $e \in E^{(2)}\}$.
  if the graph induced by $(V, E^{(2)} \setminus E_i)$ has a connected component containing a kECkDS of $G^{(1)}$ **then**
    $E^{(2)} \leftarrow E^{(2)} \setminus E_i$.
  else
    break;
  end if
end for
Return $G^{(2)} = (V, E^{(2)})$.

MLSVBA: Heuristic Algorithm for MLSVB in CRN – cont’

- SIZE-MINIMIZER
  - find a feasible solution of kECkDS from the output of the second stage

more than one connected component including a feasible solution is possible

Algorithm 4 SIZE-MINIMIZER $(G^{(1)}, G^{(2)})$

$S \leftarrow V$. Let $G_i = (V_i^{(2)}, E_i^{(2)}), 1 \leq i \leq c$, be c connected components in $G^{(2)}$.
for $i = 1$ to c **do**
  $S_i^1 \leftarrow$ CONSTRUCTMkECkDS (G_i).
  if S_i^1 can k-dominate $V \setminus V_i^{(2)}$ in $G^{(1)}$ **then**
    $S_i \leftarrow S_i^1$.
  else
    $S_i^2 \leftarrow$ FINDMkDSC $(V_i^{(2)} \setminus S_i^1, U)$, where $U \leftarrow \{v | v \in V \setminus V_i^{(2)}$ and v is not k-dominated by $S_i^1\}$.
    $S_i \leftarrow S_i^1 \cup S_i^2$.
  end if
  while S_i is not k-edge connected **do**
    Construct a minimum CDS (MCDS) S_i' of S_i from $V_i^{(2)} \setminus S_i$ using Guha *et al.*'s approach [3] and set $S_i \leftarrow S_i \cup S_i'$.
  end while
  If $|S_i| < |S|$, then $S \leftarrow S_i$.
end for
Return S.

MLSVBA: Heuristic Algorithm for MLSVB in CRN – cont’

- SIZE-MINIMIZER
  - find a feasible solution of kECkDS from the output of the second stage

Algorithm 4 SIZE-MINIMIZER $(G^{(1)}, G^{(2)})$

$S \leftarrow V$. Let $G_i = (V_i^{(2)}, E_i^{(2)}), 1 \leq i \leq c$, be c connected components in $G^{(2)}$.
for $i = 1$ to c **do**
  $S_i^1 \leftarrow$ CONSTRUCTMkECkDS (G_i).
  if S_i^1 can k-dominate $V \setminus V_i^{(2)}$ in $G^{(1)}$ **then**
    $S_i \leftarrow S_i^1$.
  else
    $S_i^2 \leftarrow$ FINDMkDSC $(V_i^{(2)} \setminus S_i^1, U)$, where $U \leftarrow \{v | v \in V \setminus V_i^{(2)}$ and v is not k-dominated by $S_i^1\}$.
    $S_i \leftarrow S_i^1 \cup S_i^2$.
  end if
  while S_i is not k-edge connected **do**
    Construct a minimum CDS (MCDS) S_i' of S_i from $V_i^{(2)} \setminus S_i$ using Guha *et al.*’s approach [3] and set $S_i \leftarrow S_i \cup S_i'$.
  end while
  If $|S_i| < |S|$, then $S \leftarrow S_i$.
end for
Return S.

if S_i^1 cannot k-dominate the all other nodes, all more nodes by solving a minimum weight set-cover problem

MLSVBA: Heuristic Algorithm for MLSVB in CRN – cont'

- SIZE-MINIMIZER
  - find a feasible solution of kECkDS from the output of the second stage

Algorithm 4 SIZE-MINIMIZER $(G^{(1)}, G^{(2)})$

$S \leftarrow V$. Let $G_i = (V_i^{(2)}, E_i^{(2)}), 1 \leq i \leq c$, be c connected components in $G^{(2)}$.
for $i = 1$ to c **do**
  $S_i^1 \leftarrow$ CONSTRUCTMkECkDS (G_i).
  if S_i^1 can k-dominate $V \setminus V_i^{(2)}$ in $G^{(1)}$ **then**
    $S_i \leftarrow S_i^1$.
  else
    $S_i^2 \leftarrow$ FINDMkDSC $(V_i^{(2)} \setminus S_i^1, U)$, where $U \leftarrow \{v | v \in V \setminus V_i^{(2)}$ and v is not k-dominated by $S_i^1\}$.
    $S_i \leftarrow S_i^1 \cup S_i^2$.
  end if
  while S_i is not k-edge connected **do**
    Construct a minimum CDS (MCDS) S_i' of S_i from $V_i^{(2)} \setminus S_i$ using Guha *et al.*'s approach [3] and set $S_i \leftarrow S_i \cup S_i'$.
  end while
  If $|S_i| < |S|$, then $S \leftarrow S_i$.
end for
Return S.

if S_i^1 is not k-edge-connected, add more nodes to make it true

NORTH CAROLINA CENTRAL UNIVERSITY FOUNDED 1910
UNIVERSITY OF CENTRAL MISSOURI
LEARNING TO A GREATER DEGREE

MLSVBA: Heuristic Algorithm for MLSVB in CRN – cont’

- SIZE-MINIMIZER
  - find a feasible solution of kECkDS from the output of the second stage

Algorithm 4 SIZE-MINIMIZER $(G^{(1)}, G^{(2)})$

$S \leftarrow V$. Let $G_i = (V_i^{(2)}, E_i^{(2)}), 1 \le i \le c$, be c connected components in $G^{(2)}$.
for $i = 1$ to c **do**
  $S_i^1 \leftarrow$ CONSTRUCTMkECkDS (G_i).
  if S_i^1 can k-dominate $V \setminus V_i^{(2)}$ in $G^{(1)}$ **then**
    $S_i \leftarrow S_i^1$.
  else
    $S_i^2 \leftarrow$ FINDMkDSC $(V_i^{(2)} \setminus S_i^1, U)$, where $U \leftarrow \{v | v \in V \setminus V_i^{(2)}$ and v is not k-dominated by $S_i^1\}$.
    $S_i \leftarrow S_i^1 \cup S_i^2$.
  end if
  while S_i is not k-edge connected **do**
    Construct a minimum CDS (MCDS) S_i' of S_i from $V_i^{(2)} \setminus S_i$ using Guha *et al.*'s approach [3] and set $S_i \leftarrow S_i \cup S_i'$.
  end while
  If $|S_i| < |S|$, then $S \leftarrow S_i$.
end for
Return S.

try to find the minimum size one

Conclusion

- the first paper studies fault-tolerance issue of virtual backbone construction in CRN
- a 3-staged polynomial time heuristic algorithm is proposed
- plan to
  - analysis the complexity of the problem
  - analysis the performance of the proposed algorithm
  - run the simulation to evaluate the average performance
  - distributed algorithm

Thank you

Question?