On Computing Resilient Virtual Backbone in Cognitive Radio Networks

Yi Hong\textsuperscript{1}, Donghyun Kim\textsuperscript{2}, Deying Li\textsuperscript{1}, Jiaofei Zhong\textsuperscript{3}, and Alade O. Tokuta\textsuperscript{2}

\textsuperscript{1}Renmin University of China, Beijing, China
\textsuperscript{2}North Carolina Central University, Durham, USA
\textsuperscript{3}University of Central Missouri, Warrensburg, USA

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
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
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

<table>
<thead>
<tr>
<th>regular (flooding-based) routing</th>
<th>routing over virtual backbone</th>
</tr>
</thead>
<tbody>
<tr>
<td>• redundant</td>
<td>• smaller routing path search space</td>
</tr>
<tr>
<td>• heavy collision and interference overhead</td>
<td>• any routing scheme becomes more efficient</td>
</tr>
<tr>
<td>• energy inefficient</td>
<td></td>
</tr>
</tbody>
</table>

- 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 $E'$
• $CU = \{CU_1, \ldots, CU_n\}$: $n \geq 1$ CUs
• $PU = \{PU_1, \ldots, PU_m\}$: $m \geq 2$ PUs
• $C = \{c_1, \ldots, c_i\}$: 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 $j = 1, 2, \cdots, m$
  • $E(X_{(c,j)}) = \lambda_{(c,j)}$ 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 $\rho(e_{(i,j)}) = \max_{1 \leq q \leq p} \{\min[E(X_{(q,i)}), E(X_{(q,j)})]\}$
Problem Definition – cont’

- Definition 2. Lifetime of a Connected Network
  - given a connected network \( G \), its lifetime is
  \[
  \rho(G) = \max\{\rho | \text{deleting all communication links with expected lifetime less than } \rho \text{ cannot cause } G \text{ disconnected}\}
  \]

- 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. \( k \)EC\( k \)DS
  - 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 \( G \)
Problem Definition – cont’

• Definition 4. Lifetime of $D$, a $k$ECKDS
  • internal lifetime: the minimum amount of time that makes $k$ECKDS disconnected
  • external lifetime: the minimum amount of time that any node outside $k$ECKDS is disconnected from $k$ECKDS
  • lifetime of $k$ECKDS: minimum of internal lifetime and external lifetime, i.e. the minimum time $k$ECKDS 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 $k$ECKDS of $G$
    • lifetime of $D$ is maximized, and
    • size of $D$ is minimized.
MLSVBA: Heuristic Algorithm for MLSVB in CRN

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

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

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

1: Let $\rho_1 < \cdots < \rho_l$ be the list of distinct lifetime levels in $L$.
2: $E^{(1)} \leftarrow E$.
3: for $i = 1$ to $l$ do
4:   $E_i = \{e | \rho(e) = \rho_i$ and $e \in E^{(1)}\}$.
5:   if the graph induced by $(V, E^{(1)} \setminus E_i)$ is still $k$-edge-connected then
6:      $E^{(1)} \leftarrow E^{(1)} \setminus E_i$.
7:   else
8:      break;
9:   end if
10: end for
11: 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 \((G^{(1)}, L)\)

1: Let \(\rho_1 < \cdots < \rho_i\) be the list of distinct lifetime levels in \(L\).
2: \(E^{(2)} \leftarrow E^{(1)}\).
3: for \(i = 1\) to \(i\) do
4: \(E_i = \{ e | \rho(e) = \rho_i \text{ and } e \in E^{(2)} \} \).
5: if the graph induced by \((V, E^{(2)} \setminus E_i)\) has a connected component containing a \(k\)EC\&DS of \(G^{(1)}\) then
6: \(E^{(2)} \leftarrow E^{(2)} \setminus E_i\).
7: else
8: break;
9: end if
10: end for
11: 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

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

1: $S \leftarrow V$. Let $G_i = (V_i^{(2)}, E_i^{(2)})$, $1 \leq i \leq c$, be $c$ connected components in $G^{(2)}$.
2: for $i = 1$ to $c$ do
3: \hspace{1em} $S_1^i \leftarrow$ CONSTRUCT $kECKDS$ ($G_i$).
4: \hspace{1em} if $S_1^i$ can $k$-dominate $V \setminus V_i^{(2)}$ in $G^{(1)}$ then
5: \hspace{2em} $S_i \leftarrow S_1^i$.
6: \hspace{1em} else
7: \hspace{2em} $S_2^i \leftarrow$ FIND $k$-DSC ($V_i^{(2)} \setminus S_1^i, U$), where $U \leftarrow \{v|v \in V \setminus V_i^{(2)}$ and $v$ is not $k$-dominated by $S_1^i\}$.
8: \hspace{2em} $S_i \leftarrow S_1^i \cup S_2^i$.
9: \hspace{1em} end if
10: \hspace{1em} while $S_i$ is not $k$-edge connected do
11: \hspace{2em} 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 $S_i \leftarrow S_i' \cup S_i'$.
12: \hspace{1em} end while
13: \hspace{1em} If $|S_i'| < |S|$, then $S \leftarrow S_i$.
14: end for
15: Return $S$.
```
MLSVBA: Heuristic Algorithm for MLSVB in CRN – cont’

- **SIZE-MINIMIZER**
  - find a feasible solution of $k$EC$k$DS from the output of the second stage

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

1: $S \leftarrow V$. Let $G_i = (V_i^{(2)}, E_i^{(2)}), 1 \leq i \leq c$, be $c$ connected components in $G^{(2)}$.
2: for $i = 1$ to $c$ do
3:     $S^1_i \leftarrow \text{CONSTRUCT}k\text{EC}k\text{DS} (G_i)$.
4:     if $S^1_i$ can $k$-dominate $V \setminus V_i^{(2)}$ in $G^{(1)}$ then
5:         $S_i \leftarrow S^1_i$.
6:     else
7:         $S^2_i \leftarrow \text{FIND}k\text{ADSC} (V_i^{(2)} \setminus S^1_i, U)$, where $U \leftarrow \{v|v \in V \setminus V_i^{(2)} \text{ and } v \text{ is not } k\text{-dominated by } S^1_i\}$.
8:         $S_i \leftarrow S^1_i \cup S^2_i$.
9:     end if
10:     while $S_i$ is not $k$-edge connected do
11:         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 $S_i \leftarrow S_i \cup S'_i$.
12:     end while
13:     if $|S_i| < |S|$, then $S \leftarrow S_i$.
14: end for
15: Return $S$.
```

if $S^1_i$ 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 $k$ECKD from the output of the second stage

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

1: $S \leftarrow V$. Let $G_i = (V_i^{(2)}, E_i^{(2)}), 1 \leq i \leq c$, be $c$ connected components in $G^{(2)}$.
2: for $i = 1$ to $c$ do
3:     $S_i^1 \leftarrow$ CONSTRUCTKECKD $(G_i)$.
4:     if $S_i^1$ can $k$-dominate $V \setminus V_i^{(2)}$ in $G^{(1)}$ then
5:         $S_i \leftarrow S_i^1$.
6:     else
7:         $S_i^2 \leftarrow$ FINDKMDSC $(V_i^{(2)} \setminus S_i^1, U)$, where $U \leftarrow \{v | v \in V \setminus V_i^{(2)}$ and $v$ is not $k$-dominated by $S_i^1\}$.
8:         $S_i \leftarrow S_i^1 \cup S_i^2$.
9: end if
10: while $S_i$ is not $k$-edge connected do
11:     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 $S_i \leftarrow S_i \cup S'_i$.
12: end while
13: if $|S_i| < |S|$, then $S \leftarrow S_i$.
14: end for
15: Return $S$.
```
MLSVBA: Heuristic Algorithm for MLSVB in CRN – cont’

- SIZE-MINIMIZER
  - find a feasible solution of $k$ECK$DS$ from the output of the second stage

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

1: $S \leftarrow V$. Let $G_i = (V_i^{(2)}, E_i^{(2)}), 1 \leq i \leq c$, be $c$ connected components in $G^{(2)}$.
2: for $i = 1$ to $c$
3: $S_i^1 \leftarrow$ CONSTRUCT$MkECKDS$ $(G_i)$.
4: if $S_i^1$ can $k$-dominate $V \setminus V_i^{(2)}$ in $G^{(1)}$ then
5: $S_i \leftarrow S_i^1$.
6: else
7: $S_i^2 \leftarrow$ FIND$MkCDS$ $(V_i^{(2)} \setminus S_i^1, U)$, where $U \leftarrow \{v \mid v \in V \setminus V_i^{(2)}$ and $v$ is not $k$-dominated by $S_i^1\}$.
8: $S_i \leftarrow S_i^1 \cup S_i^2$.
9: end if
10: while $S_i$ is not $k$-edge connected do
11: 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 $S_i \leftarrow S_i \cup S_i'$.
12: end while
13: If $|S_i| < |S|$, then $S \leftarrow S_i$.
14: end for
15: 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?