Cocircuits of Linear Matroids

John Arellano
Advisor: Dr. Illya Hicks

Rice University Future Faculty Workshop

September 11, 2012
Motivation

Matroids

The Cogirth Problem

The Set Covering Problem

Branch-and-Cut Algorithm

Conclusions and Further Research
M’s represent streams of information or measurements
S’s possible sensor locations
Sensor Network Linear Model

\[ y = Hu + \varepsilon \]

\( y \) is a \( n \times 1 \) vector of measurements

\( H \) is an \( n \times p \) system matrix with \( n > p \)

\( u \) is a \( p \times 1 \) of system states

\( \varepsilon \) is an error term
Given $H$, find the degree of redundancy of the sensor network.

The **degree of redundancy**, $\eta(H)$, can be defined as:

$$\eta(H) = \min \{ d - 1 : \text{there exists } H_{(-d)} \text{ s.t. } r(H_{(-d)}) < p \}$$

- $H_{(-d)}$ is the reduced matrix after deleting $d$ rows of $H$
- $r(H_{(-d)})$ is the **rank** of the reduced matrix
- $p = r(H)$ ($H$ has full column rank)
A **matroid** $M$ is an ordered pair $(S, \mathcal{I})$ consisting of a finite set $S$ and a collection $\mathcal{I}$ of subsets of $S$ satisfying:

- $\emptyset \in \mathcal{I}$
- If $I \in \mathcal{I}$ and $J \subseteq I$, then $J \in \mathcal{I}$
- If $I_1, I_2 \in \mathcal{I}$ and $|I_1| < |I_2|$, then there is an element $e$ of $I_2 - I_1$ such that $I_1 \cup e \in \mathcal{I}$

where the members of $\mathcal{I}$ are the **independent sets** of $M$ and $S$ is the **ground set** of $M$. 
Consider the following matrix

\[ Z^T = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix} \]

\[ M = (S, \mathcal{I}) \]

\[ S = \{1, 2, 3, 4, 5\} \text{ (columns of } Z^T) \]

\[ \mathcal{I} = \{\emptyset, \{1\}, \{2\}, \{4\}, \{5\}, \{1, 2\}, \{1, 5\}, \{2, 4\}, \{2, 5\}, \{4, 5\}\} \]

**Dependent Sets** = \( \mathcal{P}(S) - \mathcal{I} \), where \( \mathcal{P}(S) \) is the set of all subsets of \( S \).
A maximal independent set of $M$ is a **basis**, $B$ of $M$. The collection of bases of $M$ is denoted by $\mathcal{B}(M)$.

$$\mathcal{B}(M) = \{\{1, 2\}, \{1, 5\}, \{2, 4\}, \{2, 5\}, \{4, 5\}\}$$

A minimal dependent set of $M$ is a **circuit**, $C$ of $M$. Equivalently, $C$ is a circuit if $C \not\in \mathcal{I}$ and $C - \{x\} \in \mathcal{I}$ for all $x \in C$. The set of circuits of $M$ is denoted by $\mathcal{C}(M)$.

$$\mathcal{C}(M) = \{\{3\}, \{1, 4\}, \{1, 2, 5\}, \{2, 4, 5\}\}$$

**girth** - cardinality of the smallest circuit of $M$
Dual matroid $M^*$ is the matroid with ground set $S(M)$ and bases $\mathcal{B}(M^*) = \{S(M) - B : B \in \mathcal{B}(M)\}$. We call the bases of $M^*$ the cobases of $M$.

Equivalently, $M^* = (S, \mathcal{I}^*)$, $\mathcal{I}^* = \{J \subseteq S : r(S - J) = r(S)\}$

Referring back to the example:

$$Z^T = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathcal{B}(M) = \{\{1, 2\}, \{1, 5\}, \{2, 4\}, \{2, 5\}, \{4, 5\}\}$$

$\Rightarrow$

$$\mathcal{B}(M^*) = \{\{3, 4, 5\}, \{2, 3, 4\}, \{1, 3, 5\}, \{1, 3, 4\}, \{1, 2, 3\}\}$$
The circuits of $M^*$ are the minimal dependent sets of $M^*$. We call the circuits of $M^*$ the **cocircuits** of $M$.

Referring back to the example:

$$Z^T = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathcal{C}(M^*) = \{\{1, 2, 4\}, \{1, 2, 5\}, \{2, 4, 5\}, \{1, 4, 5\}\}$$

$$\mathcal{B}(M) = \{\{1, 2\}, \{1, 5\}, \{2, 4\}, \{2, 5\}, \{4, 5\}\}$$

**cogirth** - cardinality of the smallest cocircuit of $M$
Relating Redundancy to Cogirth

The **degree of redundancy**, \( \eta(H) \), can also be defined as:

\[
\eta(H) = \min \left\{ d - 1 : \text{there exists } H_{(-d)} \text{ s.t. } r(H_{(-d)}) < p \right\}
\]

- The set of deleted rows intersects every basis of the row space of \( H \)
- Any \( C \in \mathcal{C}(M^*) \) intersects every \( B \in \mathcal{B}(M) \) for a given matroid \( M \)
- \( \eta(H) = \text{cogirth} - 1 \)
Pros and Cons

- **Exhaustive Rank Testing**
  + Good if matrix and cogirth are small
  - Practical matrices are not small

- **Circuit Enumeration (Boros et al. 2003)**
  + Will Find optimal solution
  - Unnecessary generation of all circuits

- **Branch-and-Decompose (Cho et al. 2007)**
  + Divide and Conquer approach
  - Exhaustive rank testing

- **$\ell^1$-norm minimization approach (Govindaraj 2010)**
  + Efficient and good approximation
  - Approximation may not equal optimal value

- **0-1 MIP Formulation (Kianfar et al. 2011)**
  + Provides upper and lower bounds if necessary
  - Does not exploit structure of matrix
The Set Covering Problem

\[ \min c^T x \]
\[ \text{s.t. } Ax \geq 1 \]
\[ x \text{ binary} \]
Cogirth Problem ⇒ Set Covering Problem

\[
Z^T = \begin{pmatrix}
1 & 2 & 3 & 4 & 5 \\
1 & 0 & 0 & 1 & 1 \\
0 & 1 & 0 & 0 & 1
\end{pmatrix}
\]

\[\mathcal{B}^*(M) = \{\{3, 4, 5\}, \{2, 3, 4\}, \{1, 3, 5\}, \{1, 3, 4\}, \{1, 2, 3\}\} \]

\[\mathcal{C}(M) = \{\{3\}, \{1, 4\}, \{1, 2, 5\}, \{2, 4, 5\}\} \]
Cogirth Problem ⇒ Set Covering Problem

\[ C(M) = \{\{3\}, \{1, 4\}, \{1, 2, 5\}, \{2, 4, 5\}\} \]

\[ \text{min } 1^T x \]
\[ \text{s.t. } Ax \geq 1 \]
\[ x \text{ binary} \]
Set covering problem (SCP) solution

\[ \downarrow \]

Cogirth of a linear matroid

\[ \downarrow \]

Degree of redundancy of a sensor network
Addressing the Set Covering Problem

- Greedy Heuristics (Johnson 1974 and Chvátal 1979)
- Subgradient Optimization Algorithm (Balas and Ho 1980)
- Set Covering Algorithm (Beasley 1987)
- Genetic Algorithm - “Survival of the Fittest” (Beasley 1996)
Major Obstacle

- Need to know entire system $Ax \geq 1$
  - Cannot form linear relaxation or dual program
  - Difficult to find good initial population for genetic algorithm
  - Difficult to find good initial solution for greedy heuristics

Silver Lining

- Incorporate bordered block diagonal form (BBDF) (Cho 2007)
- Incorporate greedy heuristics and cutting plane generation techniques
Bordered Block Diagonal Form

\[
\begin{pmatrix}
A_1 & & \\
& A_2 & \\
& & \ddots \\
P_1 & P_2 & \cdots & P_{nb}
\end{pmatrix}
\]

Submatrices, \([A_1, P_1], [A_2, P_2], \ldots, [A_{nb}, P_{nb}]\)
Branch-and-Cut Algorithm

INITIALIZATION

NODE

RESTORE

LP RELAXATION

CUT

PRUNE

BRANCHING

EXIT
## Comparison of Algorithms

| No. | Size ($n \times p$) | $|P|$ | $d^*$ | Computation Time |
|-----|---------------------|------|------|------------------|
|     |                     |      |      | Algorithm 1      | Algorithm 2      | Algorithm 3      |
| 1   | $34 \times 12$      | 2    | 6    | 56 secs          | $\approx 0$ secs| 0.08 secs        |
| 2   | $66 \times 27$      | 3    | 7    | $> 36000$ secs  | $\approx 0$ secs| 11.59 secs       |
| 3   | $154 \times 72$     | 2    | 4    | $> 36000$ secs  | 561 secs         | 24.54 secs       |
| 4   | $221 \times 55$     | 1    | 14   | $> 36000$ secs  | 798 secs         | 10560 secs       |
| 5   | $318 \times 144$    | 2    | 4    | $> 36000$ secs  | 918 secs         | 54.42 secs       |
| 6   | $1009 \times 252$   | 1    | 15#  | $> 36000$ secs  | $> 36000$ secs  | $> 36000$ secs  |

- $d^*$ is the degree of redundancy
- Algorithm 1 is the Branch-and-Cut algorithm tested on the entire matrix
- Algorithm 2 is the BBDF incorporated with the Branch-and-Cut algorithm
- Algorithm 3 is the 0-1 MIP Formulation
- # – reported cogirth. None of the algorithms achieved optimality
Breakdown of Algorithm 1

<table>
<thead>
<tr>
<th>No.</th>
<th>Size ($n \times p$)</th>
<th>$d^*$</th>
<th>BFS</th>
<th>Time</th>
<th>Total Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$34 \times 12$</td>
<td>6</td>
<td>6</td>
<td>3 secs</td>
<td>56 secs</td>
</tr>
<tr>
<td>2</td>
<td>$66 \times 27$</td>
<td>7</td>
<td>7</td>
<td>14 secs</td>
<td>$&gt; 36000$ secs</td>
</tr>
<tr>
<td>3</td>
<td>$154 \times 72$</td>
<td>4</td>
<td>5</td>
<td>248 secs</td>
<td>$&gt; 36000$ secs</td>
</tr>
<tr>
<td>4</td>
<td>$221 \times 55$</td>
<td>14</td>
<td>16</td>
<td>$\approx 0$ secs</td>
<td>$&gt; 36000$ secs</td>
</tr>
<tr>
<td>5</td>
<td>$318 \times 144$</td>
<td>4</td>
<td>4</td>
<td>1 secs</td>
<td>$&gt; 36000$ secs</td>
</tr>
<tr>
<td>6</td>
<td>$1009 \times 252$</td>
<td>15#</td>
<td>20</td>
<td>13 secs</td>
<td>$&gt; 36000$ secs</td>
</tr>
</tbody>
</table>

- BFS - The value of the best feasible solution found
- The BFS is optimal or provides an approximation to the optimal value
### Breakdown of Algorithm 2

<table>
<thead>
<tr>
<th>No.</th>
<th>(P)</th>
<th># blocks</th>
<th>(d^*)</th>
<th>BFS</th>
<th>BBDF Time</th>
<th>BFS Time</th>
<th>Total Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>(\approx 0) secs</td>
<td>(\approx 0) secs</td>
<td>(\approx 0) secs</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>12</td>
<td>7</td>
<td>7</td>
<td>(\approx 0) secs</td>
<td>(\approx 0) secs</td>
<td>(\approx 0) secs</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>20</td>
<td>4</td>
<td>4</td>
<td>14 secs</td>
<td>1 secs</td>
<td>561 secs</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>12</td>
<td>14</td>
<td>14</td>
<td>(\approx 0) secs</td>
<td>5 secs</td>
<td>798 secs</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>38</td>
<td>4</td>
<td>4</td>
<td>151 secs</td>
<td>1 secs</td>
<td>918 secs</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>43</td>
<td>15#</td>
<td>17</td>
<td>(\approx 0) secs</td>
<td>224 secs</td>
<td>(&gt; 36000) secs</td>
</tr>
</tbody>
</table>

- The BFS is optimal for each instance except for the last one.
Conclusions

- Introduction to matroids
- Described how matroids can be incorporated into sensor network design
- Brief discussion of the set covering problem and methods available to address it
- Proposed a branch-and-cut algorithm to address the cogirth problem for linear matroids via the set covering problem
Examine applications to compressive sensing.

Investigate implicit hitting set method (Moreno Centeno 2006)
## Computational Results

Matrix $H$

<table>
<thead>
<tr>
<th>No.</th>
<th>Size $(n \times p)$</th>
<th>$d^*$</th>
<th>$d^{**}$</th>
<th>Comp. Time</th>
<th>$d^{**}$</th>
<th>Comp. Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$34 \times 12$</td>
<td>6</td>
<td>6</td>
<td>1 sec</td>
<td>6</td>
<td>1 sec</td>
</tr>
<tr>
<td>2</td>
<td>$66 \times 27$</td>
<td>7</td>
<td>7</td>
<td>1 sec</td>
<td>7</td>
<td>2 sec</td>
</tr>
<tr>
<td>3</td>
<td>$154 \times 72$</td>
<td>4</td>
<td>4</td>
<td>3 sec</td>
<td>4</td>
<td>8 sec</td>
</tr>
<tr>
<td>4</td>
<td>$221 \times 55$</td>
<td>14</td>
<td>13</td>
<td>9 sec</td>
<td>13</td>
<td>11 sec</td>
</tr>
<tr>
<td>5</td>
<td>$318 \times 144$</td>
<td>4</td>
<td>4</td>
<td>12 sec</td>
<td>4</td>
<td>28 sec</td>
</tr>
<tr>
<td>6</td>
<td>$1009 \times 252$</td>
<td>15#$^*$</td>
<td>17</td>
<td>298 sec</td>
<td>17</td>
<td>572 sec</td>
</tr>
</tbody>
</table>
Finding the cogirth of a matroid

- Exhaustive rank testing

- Circuit Enumeration (Boros et al. 2003)
  - Begin with initial set of circuits.
  - Find another element of $\mathcal{C}(M)$, or conclude enumeration complete.
    - If $C_1, C_2$ are distinct members of $\mathcal{C}(M)$ and $e \in C_1 \cap C_2$, then
      $\exists C_3 \in \mathcal{C}(M)$ s.t. $C_3 \subseteq (C_1 \cup C_2) - e$.
  - Optimal solution is circuit of small cardinality
Bound-and-Decompose (Cho et al. 2007)

- Do exhaustive rank testing on entire matrix while
  \[ d < \left( \frac{n_b}{n_b-1} \right) |P| - 1 \]

- Transform matrix into bordered block diagonal form (BBDF).

\[
\begin{pmatrix}
A_1 & & \\
& A_2 & \\
& & \ddots \\
& & & A_{n_b} \\
P_1 & P_2 & \cdots & P_{n_b}
\end{pmatrix}
\]

- Exhaustive rank testing on submatrices,
  \([A_1, P_1], [A_2, P_2], \ldots, [A_{n_b}, P_{n_b}]\)
Need to consider the dual matroid to find cocircuits

Given $H^T$, find matrix representation of the dual matroid (Oxley 1992)

- Reduce $H^T$ to the form $[I_r | D]$
- $M^*$ is the linear matroid of $[-D^T | I_{n-r}] = \mathcal{H}$

For each column $h_i$ of $\mathcal{H}$, let $\mathcal{H}(\_i)$ denote $\mathcal{H}$ excluding $h_i$

$$\begin{align*}
\min & \|x\|_1 \\
\text{s.t.} & \mathcal{H}(\_i)x = h_i
\end{align*}$$

- Nonzero elements of $x$ correspond to columns that are in a cocircuit containing $h_i$

- Choose best solution
0-1 MIP formulation based on the null space of $H$ (Kianfar et al. 2011)

Assume row vectors of $H$ are scaled such that $\|h_i\|_1 = 1$ for each row $i$

Find the minimum number of vectors, $d$, that if eliminated from $H$, $H_{(-d)}$ has a nonzero null space.

$$
\begin{align*}
\min & \sum_{i=1}^{n} q_i \\
\text{s.t.} & -q_i \leq \sum_{i=1}^{p} h_{ij} x_j \leq q_i, & i = 1, \ldots, n \\
& -1 + 2z_j \leq x_j \leq 1, & j = 1, \ldots, p \\
& \sum_{i=1}^{p} z_i = 1 \\
x & \in \mathbb{R}^p, \ z \in \{0, 1\}^p, \ q \in \{0, 1\}^n
\end{align*}
$$
Greedy Heuristics (Johnson 1974 and Chvátal 1979)

Set $\text{Cov} = \emptyset$, $\text{Uncov} = \{1, \ldots, m\}$, $\text{Sol} = \emptyset$

while $\text{Uncov} \neq \emptyset$

Choose $j$th column of $A$ and $\text{Sol} = \text{Sol} \cup j$ based on weights given to the columns of $A$

for $i = 1 \rightarrow m$

if $a_{ij} = 1$

$\text{Cov} = \text{Cov} \cup i$, $\text{Uncov} = \text{Uncov} - i$

end if

end for

end while

$\text{Sol}$ is a solution to the SCP
Subgradient Optimization Algorithm (Balas and Ho 1980)
- Formulate linear relaxation and its dual program
- Primal and Dual Heuristics
- Cutting Planes from conditional bounds (Balas 1980)

Set Covering Algorithm (Beasley 1987)
- Subgradient optimization
- Problem reduction
- Tree search procedures
Solving the Set Covering Problem

- **Genetic Algorithm - “Survival of the Fittest”** (Beasley 1996)
  - Create initial population of solutions
  - Create child from two parents and replace solution
  - Find a specified number of distinct solutions

- **Modified Greedy Heuristics** (Marchiori and Steenbeek 1998, Musliu 2006)
  - Generate initial solution
  - Take partial cover of best solution
  - Generate new sets of solutions and replace best solution (if necessary)
Cutting Plane Generation techniques

- Find minimum weight basis (greedy algorithm)
  - Use a solution, \( x \), to the LP relaxation as weights for columns
  - Add columns one by one if they increase the rank of the submatrix
  - Stop when basis is found

- Form inequalities of the form \( a^S x \geq 2 \), \( S \) is a subset of columns of \( A \) (Balas and Ng 1989)
  - Given a subset \( S \) of the rows of \( A \) from \( Ax \geq 1 \)
  - (i) add the inequalities \( a^i x \geq 1 \), \( i \in S \)
  - (ii) divide the resulting inequality by \( |S| - \varepsilon \), \( 0.5 < \varepsilon < 1 \)
  - (iii) round up all coefficients to the nearest integer

- Feasible Solution Exclusion Constraints (Beasley and Jornsten 1992)
  \[
  \sum_{j \in T_c} x_j \leq |T_c| - 1 \\
  \sum_{j \notin T_c} x_j \geq 1
  \]
Finding Feasible Solutions

- Given solution, \( x \) to the linear relaxation
- Greedy Algorithm to find feasible solution - use \( x \) as weights

BBDF

- Create bipartite graph using rows and columns of \( H \)
- Use Menger’s Thm. to find separating set of rows
- Find submatrices \([A_1, P_1], [A_2, P_2], ..., [A_{n_b}, P_{n_b}]\)


Set covering algorithms using cutting planes, heuristics, and subgradient optimization: A computational study.
10.1007/BFb0120886.

On the set covering polytope: I. all the facets with coefficients in 0, 1, 2.
10.1007/BF01582278.

An algorithm for set covering problem.

A genetic algorithm for the set covering problem.

Enhancing an algorithm for set covering problems.
References III

*Introduction to Linear Optimization.*  

Algorithms for enumerating circuits in matroids.  
In *Algorithms and Computation, 14th International Symposium, ISAAC 2003*,  

On the (co)girth of a connected matroid.  

*On the robustness of clustered sensor networks.*  

A Greedy Heuristic for the Set-Covering Problem.  

Calculation of sensor redundancy degree for linear sensor systems.  
Computational experience with approximation algorithms for the set covering problem. 


Approximation algorithms for combinatorial problems. 

Reducibility among combinatorial problems. 

On the complexity of some enumeration problems for matroids. 
Observability, redundancy, reliability and integrated design of measurement systems.

[22] E. Marchiori and A. Steenbeek.

[23] E. Moreno Centeno.
Implicit hitting set problems.

Local search algorithm for unicost set covering problem.

Integer and combinatorial optimization.
*Matroid theory.*  

[27] W. J. Rugh.  
*Linear System Theory.*  

Selection of measurements necessary to achieve multicomponent mass balances in chemical plant.  

[29] H. Whitney.  
On the abstract properties of linear dependence.  

*Integer programming.*  
A Wiley-Interscience Publication.