**Split Graphs**

- An undirected graph $G = (V, E)$ is defined to be split if there is a partition $V = S + K$ of its vertex set:
  
  $S$ = stable set and $K$ = complete set.

- In general, the partition $V = S + K$ of a split graph will not be unique.

- $S$ will not necessarily be a maximal stable set.
  $K$ ⇒ $\Rightarrow$ maximal clique.
• Since a stable set of $G$ is a complete set of $\overline{G}$ and vice versa, we have an immediate result.

• **Theorem**: $G$ is a split graph if and only if $\overline{G}$ is a split graph.

• The next theorem follows from the work of Hammer and Simeone [1977].

• **Theorem**: Let $G$ be a split graph and $V= S + k$. Exactly one of the following conditions holds:

  (i) $|S| = \alpha(G)$ and $|k| = \omega(G)$

  (ii) $|S| = \alpha(G)$ and $|k| = \omega(G) - 1$

  (iii) $|S| = \alpha(G) - 1$ and $|k| = \omega(G)$
• Theorem (Földes and Hammer [1977]). Let $G$ be an undirected graph. The following conditions are equivalent:
  (i) $G$ is a split graph;
  (ii) $G$ and $\overline{G}$ are triangulated graphs;
  (iii) $G$ contains no induced subgraph isomorphic to $2k_2$, $C_4$, $C_5$.

• A characterization of when a split graph is also a comparability graph is given by the following theorem:

• Theorem: If $G$ is a split graph, then $G$ is a comparability graph iff $G$ contains no induced subgraph isomorphic to $H_1$, $H_2$, or $H_3$.

$H_1$

$H_2$

$H_3$
Cographs (or Complement reducible graphs)

- Cographs are defined as the class of graphs formed from a single vertex under the closure of the operations of union and complement.

- More precisely, the class of cographs can be defined recursively as follows:
  1. a single-vertex graph is a cograph;
  2. the disjoint union of a cograph is a cograph;
  3. the complement of a cograph is a cograph;

- Example:
Construction of the following cograph:

\[ \begin{align*}
U: \quad & a + b \\
C: \quad & a \quad b \\
U: \quad & a + b \\
C: \quad & a \\
U: \quad & a + b \\
C: \quad & b \quad c \\
U: \quad & d + f \\
C: \quad & d \quad e \\
U: \quad & b + c \\
C: \quad & c \quad d \\
\end{align*} \]
- Cographs were introduced in the early 1970s by Lerchs.

- Lerchs has shown, among other properties, the following two very nice algorithmic properties:
  1. (P1) Cographs are exactly the $P_4$ restricted graphs.
  2. (P2) Cographs have a unique tree representation called cotree.

- A cograph and its cotree:

  ![Diagram of a cograph and its cotree]

- The root $R$ will have only one (0) node child if the represented cograph is disconnected.
Permutation Representation of a Cograph

- Let $G$ be a cograph with cotree $T$.
- We plot the vertices of $G$ on an $nxn$ grid.
- For each node $\alpha \in T$, we denote:
  - $G(\alpha)$ the subgraph $G[L(T(\alpha))]$.
  - $L(T(\alpha))$ the set of the leaf nodes in $T(\alpha)$.
  - $T(\alpha)$ the subtree of $T$ rooted at $\alpha$.
- The vertices of $G(\alpha)$ will be plotted on a $n(\alpha) \times n(\alpha)$ square region $R(\alpha)$ on the grid, where $n(\alpha)$ is the number of vertices in $G(\alpha)$.
- We represent
  \[
  R(\alpha) = [x_1(\alpha), y_1(\alpha), x_2(\alpha), y_2(\alpha)]
  \]
  where
  - $(x_1(\alpha), y_1(\alpha))$ is the lower-left corner, and
  - $(x_2(\alpha), y_2(\alpha))$ is the upper-right corner.
- Our algorithm is a top-down computation on $T$.

- For the root $r$ of $T$, let $R(r) = [1, 1, n, n]$. 

- Consider an internal node $\alpha$ with children $b_2, b_3, \ldots, b_k$. 

- Suppose $R(\alpha) = [x_1(\alpha), y_1(\alpha), x_2(\alpha), y_2(\alpha)]$ has been computed, we describe how to compute 
  
  $R(b_i) = [x_2(b_i), y_2(b_i), x_2(b_i), y_2(b_i)]$

  for each $1 \leq i \leq k$. 

- Case 1: $\alpha$ is a 0-node 

- Case 2: $\alpha$ is a 1-node
Example:

\[
\begin{align*}
\Gamma &= [2, 3, 1, 6, 5, 4]
\end{align*}
\]
Recognition Properties of Cograph

- Theorem: Let $G$ be a graph. The following statements are equivalent:
  (i) $G$ is a cograph;
  (ii) $G$ does not contain $P_4$ as a subgraph;

- Example:

- For every actual edge $(x,y) \in E_A$ we define:
  
  \[
  AV_{(x,y)}^x = \{ z \in V \mid (x,z) \in E \text{ and } (y,z) \notin E \} \\
  AV_{(x,y)}^y = \{ z \in V \mid (x,z) \notin E \text{ and } (y,z) \in E \} \\
  
  \]

- For every pair $x,y \in V$: $(x,y) \notin E$ we define:
  
  \[
  FV_{x,y} = \{ z \in V \mid (x,z) \in E \text{ and } (y,z) \in E \} \\
  \]
Theorem: Let $G=(V,E)$ be an $A$-free graph. Then, $G$ is a cograph.

Theorem: Let $G=(V,E)$ be a graph. The following statements are equivalent:
(i) $G$ is a cograph;
(ii) There exists no actual edge $(x,y) \in AE$ and vertices $v \in AV(x,y)$, $u \in AV(y,x)$ such that either $N(v) \cup N(y) \neq F_{\{v,y\}}$ or $N(u) \cup N(x) \neq F_{\{u,x\}}$.

Example:

1) $(x,y) \in AE$
   \[ N(v) \cup N(y) = \{x,u\} \]
   \[ F_{\{v,y\}} = \{x,u\} \]
   \[ N(u) \cup N(x) = \{v,y\} \]
   \[ F_{\{u,x\}} = \{v,y\} \]

2) $(v,x) \in AE$
   \[ N(u) \cup N(x) = \{v,w,y\} \quad \Rightarrow \quad G \text{ is not a cograph.} \]
   \[ F_{\{u,x\}} = \{v,y\} \]
• Algorithm Recognition - Cographs
  1. For every \((x,y) \in E\) do
     1.1. Compute \(AV^x_{(x,y)}\) and \(AV^y_{(x,y)}\)
  2. For every pair \(x,y \in V: (x,y) \notin E\) do
     2.1. Compute \(FV_{x,z,y}y\)
  3. For every actual edge \((x,y) \in E\) do
     3.1. For every vertex \(v \in AV^x_{(x,y)}\) do
          if \(N(v) \cup N(y) \neq FV_{x,z,y}y\) then \(P_z\); stop;
     3.2. For every vertex \(u \in AV^y_{(x,y)}\) do
          if \(N(u) \cup N(x) \neq FV_{x,z,y}y\) then \(P_y\); stop;
end.

• The algorithm runs in \(O(S.m)\) where \(S\) is the degree of the input graph.

• The algorithm can be implemented in \(O(1)\) time with \(O(S.m)\) processors on a CRCW-PRAM.
Threshold Graphs

- Chvátal and Hammer have proved the following result:

- **Theorem**: Let $G = (V, E)$ be a graph. The following statements are equivalent:
  
  (i) $G$ is a threshold graph; 
  
  (ii) $G$ has no induced subgraphs isomorphic to $P_4, C_4$, or $E_7$.

- There exist an $O(n)$-time sequential algorithm for recognizing threshold graphs.

- The algorithm is based on the degree partition.

- Let $\delta(v)$ be the degree of the vertex $v$ of an undirected graph.
- We define the degree partition of a graph $G=(V,E)$ in which we associate vertices having the same degree.

- Let $0 < \delta_1 < \delta_2 < \ldots < \delta_k < |V|$ be the degrees of the nonisolated vertices.

- $\delta_i$ are distinct; there may be many vertices of degree $\delta_i$.

- Define: $\delta_0 = 0$ and $\delta_{k+1} = |V| - 1$

- The degree partition of $V$ is given by

\[ V = D_0 + D_1 + \ldots + D_k \]

where $D_i =$ set of all vertices of degree $\delta_i$.

- Example:

```
Example: 2 2 5
2 5
5
3 3
\delta_1 = 2
\delta_2 = 3
\delta_3 = 5
\delta_4 = 2
```

\[ 0 < \delta_2 \leq \delta_3 \leq \delta_4 \leq 6 \]
The main idea of the recognition algorithm:

(a) First, it brings together all vertices with the same degree.

(b) The vertex set $V$ is partitioned into $k+1$ disjoint vertex sets

$$V = D_0 + D_1 + \cdots + D_k$$

Satisfying the property

$$u \in D_i \iff \delta(u) = i$$

$0 \leq i \leq k$.

(c) Then, the algorithm uses the following result:

$G$ is a threshold graph iff the recursions below are satisfied:

$$\delta_{i+1} = \delta_i + |D_{k-i}| \quad (i=0,1,\ldots,L_k-1)$$

$$\delta_i = \delta_{i+1} + |D_{k-i}| \quad (i=k,k-1,\ldots,L_k+1)$$
The typical structure of a threshold graph:

- $D_0$ contains all isolated vertices (may be empty)
- $D_{\frac{k+1}{2}}$ only exists if $k$ is odd
- $\delta_0 = 0$ and $\delta_{k+1} = |V| - 1$
A-free Graphs

- A graph $G=(V,E)$ is called an A-free graph if every edge of $G$ is either free or semi-free.

- We define

$$\text{cent}(G) = \{x \in V \mid N[x] = V\}$$

- **Theorem:** Let $G$ be a graph. Then the following statements are equivalent:
  (i) $G$ is a A-free graph;
  (ii) $G$ has no induced subgraph isomorphic to $P_4$ or $C_4$;
  (iii) Every connected induced subgraph $G[S]$, $S \subseteq V$, satisfies $\text{cent}(G[S]) \neq \emptyset$.

- **Lemma:** The following two statements hold:
  (i) $G$ is an A-free if and only if $G$-cent$(G)$ is an A-free;
  (ii) If $G$-cent$(G) \neq \emptyset$, then $G$-cent$(G)$ contains at least two components.
• Let $G$ be an $A$-free graph. Then,
  \[ V_1 = \text{cent}(G). \]

• Put \[ G_1 = G \]
  \[ G = V_1 \cup G_2 \cup G_3 \cup \ldots \cup G_r \]
  where
  \[ G_i \text{ is a component of } G - V_1, \quad r \geq 2. \]

• Then, $G_i$ is an $A$-free graph, and so
  \[ V_i = \text{cent}(G_i) \neq \emptyset. \]

• We finally obtain the following partition of $V$:
  \[ V = V_1 + V_2 + \ldots + V_k \]
  where \[ V_i = \text{cent}(G_i). \]
The typical structure of an A-tree graph:

\[ \begin{align*}
&V_1 \\
&V_2 \\
&\vdots \\
&\vdots \\
&\vdots \\
&V_k
\end{align*} \]

Moreover we can define a partial order \( \leq \) on \( \{V_1, V_2, \ldots, V_k\} \) as follows:

\[ V_i \leq V_j \text{ if } V_i = \text{cent}(G_i) \text{ and } V_j \subseteq V(G_i). \]
Let $G$ be a connected $A$-free graph, and let $V(G) = V_1 + V_2 + \ldots + V_k$ be the partition defined above.

Then this partition and the partially ordered set $(\{V_i\}, \leq)$ have the following properties:

(P1) If $V_i \leq V_j$ then every vertex of $V_i$ and every vertex of $V_j$ are joined by an edge.

(P2) For every $V_i$, $\text{cent}(G[\{UV_i \mid V_i \leq U \leq V_j\}]) = V_i$.

(P3) For every two $V_s$ and $V_t$: $V_s \leq V_t$, $G[\{UV_i \mid V_s \leq V_i \leq V_t\}]$ is a complete graph. Moreover, for every maximal element $V_t$ of $(\{V_i\}, \leq)$, $G[\{UV_i \mid V_t \leq V_i \leq V_t\}]$ is a maximal complete subgraph of $G$.

(P4) Every edge $(x,y): x, y \in V_i$, $(x,y) \in FE$; every edge $(x,y): z \in V_i, y \in V_j, V_i \neq V_j$, $(x,y) \in SE$. 

-19-
- G is a threshold graph if G has no induced subgraphs isomorphic to C_4, P_4 or 2K_2.

- **Theorem:** The threshold graphs are precisely those A-free graphs containing no induced subgraph isomorphic to 2K_2.

- **Theorem:** Let G be an A-free graph. The following statements are equivalent:
  1. G contains no 2K_2;
  2. G is an A-free graph;

**Proof.** (i) ⇒ (ii): Let T_c(G) be the cent-tree of G and let \( V_{i,1}, V_{i,2}, \ldots, V_{i,k_i} \) be the children of the node \( V_{0,1} = \text{cent}(G) \);

Then, \( \exists \) at least one child of \( V_{0,1} : |V_{1,1}| \geq 1 \).
We can only prove that the typical structure of an $A$-free graph which contains no induced subgraph $\cong$ to $2k_2$ is the following:

The cent-tree has the following properties:

(P1) $K = V_{0,1} \cup V_{1,1} \cup \ldots \cup V_{h,1}$ is a clique.

(P2) $S = V - K$ is an independent set.

(P3) For every $x, y \in S$ : $\text{level}(x) \leq \text{level}(y)$, $N(x) \leq N(y)$.

We can only prove that $\overline{G}$ is an $A$-free graph.

(ii) $\Rightarrow$ (i): Suppose $G$ contains $2k_2$. Then, $\overline{G}$ contains $C_4$, a contradiction.
Theorem: Let $G=(V,E)$ be an undirected graph. The following statements are equivalent:
(i) $G$ is a threshold graph;
(ii) $G$ and $\overline{G}$ are $\lambda$-free graphs.

- The free, semi-free and actual vertices of a connected graph $G$ with $n$ vertices and $m$ edges can be computed in $O(S.m)$ time.
- Thus, threshold graphs can be recognized in $O(S.m+n^2)$ time.
- $\lambda$-free, cographs and threshold graphs can be recognized in $O(s)$ time by using $O(n.m)$ processors on a CRCW-PRAM or in $O(cgy.n)$ time by using $O(n.m/lgyn)$ processors.