The Fifth Clemson mini-Conference

ON[R]

Discrete Mathematics

Clemson, South Carolina
October 11-12, 1990

DEPARTMENT
OF
MATHEMATICAL
SCIENCES
CLEMSON UNIVERSITY
Clemson, South Carolina

DISTRIBUTION STATEMENT A
Approved for public release; Distribution Unlimited
Best Available Copy
The Fifth Clemson mini-Conference ON[R] Discrete Mathematics

Proceedings

FROM 10-11-90 TO 9-30-91

1991 July 10

262

Discrete Mathematics

This contract provided funds to partially support the Clemson Mini-Conference ON[R] Discrete Mathematics (5th annual). This two day conference featured twelve speakers from the following colleges and universities: the Georgia Institute of Technology, Northeastern University, the College of William and Mary, Memphis State University, the University of Illinois (2), Ohio State University, the University of Tennessee, Wright State University, Vanderbilt University (2) and Old Dominion University. There were approximately 80 attendees. The conference has been sponsored by the Office of Naval Research for five years and in that time the conference has attracted most of the leading researchers in graph theory and discrete mathematics in the United States and some international visitors. The funds were used to pay part of the expenses of the speakers, for publication of the proceedings of the conference and for a small reception given during the conference.
The Fifth Clemson mini-Conference

ON[R]

Discrete Mathematics

Clemson, South Carolina
October 11-12, 1990

Organizers: S. T. Hedetniemi
R. Laskar

This mini-Conference was supported in part by the Office of Naval Research for the University Research Initiative Program, Contract NO. N00014-91-K-1228.

91-05672
The Fifth Clemson mini-Conference

ON[R]

Discrete Mathematics

Clemson, South Carolina
October 11-12, 1990

Schedule of talks
(All talks given in Student Senate Chambers)

Thursday, October 11

11:00 - 12 noon  
Registration

1:00 - 1:10  
Welcoming Remarks by Dr. Bobby Wixson, Dean of College of Sciences

1:10 - 1:50  
 Prof. Richard A. Duke, Department of Mathematics  
Georgia Tech.  
"The Erdős-Ko-Rado Theorem for Small Families"

Let X be a set of size n, \( \mathcal{F} \) a family of m k-element subsets of X, \( k < n/2 \), and \( \mathcal{F}' \) a subfamily of \( \mathcal{F} \) with the property that \( |F_1 \cap F_2| \geq t \) for each choice of \( F_1 \) and \( F_2 \) in \( \mathcal{F}' \). It follows immediately from the well-known Erdős-Ko-Rado Theorem that for \( m \approx \binom{n}{k} \) and \( n \) sufficiently large the maximum size of \( \mathcal{F}' \) in this case is of order \( (k/n)^m \).

In general let \( f(n,k,m) \) be the minimum of \( \max(|\mathcal{F}'|: \mathcal{F}' \subseteq \mathcal{F}, |F_1 \cap F_2| \geq t \) for each \( F_1 \) and \( F_2 \) in \( \mathcal{F}' \) ), over all families \( \mathcal{F} \) of k-element subsets of X, \( \mathcal{F}' \) of m size, \( |X| = n \). Then for \( n \) large and \( m \approx \binom{n}{k} \) we have \( f(n,k,m) \sim (k/n)^m \). In joint work with V. Rödl we investigate this function for small \( m \). We show, for example, that if \( k = cn \), \( 0 < c < \frac{1}{2} \), and \( m = n \), then \( f_0(n,k,m) = cn = (k/n)^m \) (instead of \( (k/n)^2m \) as might be expected). Our proof makes use of the Regularity Lemma of Szemerédi. Taking \( \mathcal{F} \) to be the collection of lines of a finite protective plane shows that \( k = cn \) cannot be replaced by \( k = \sqrt{n} \) in this result. In fact, we show that \( cn \) cannot be replaced by \( \sqrt{n} \ln(n) \). The case of \( m = n \) and \( k = n^{1-e}, 0 < e < \frac{1}{2} \), remains open.
2:10 - 2:50

Prof. Margaret B. Cozzens, Department of Mathematics
Northeastern University

"Critical m-neighbor-connected Graphs"

Let $G$ be a graph and $u$ be a vertex in $G$. The closed neighborhood of $u$ is $N[u] = \{u\} \cup N(u)$. A vertex $u$ is subverted when $N[u]$ is deleted from $G$. If $S$ is a subset of vertices of $G$, then $G/S$ denotes $G - \{N[u]: u \in S\}$. $S$ is called a cut-strategy of $G$ if $G/S$ is disconnected, or a clique, or the empty set. We define the neighborhood-connectivity, $K(G)$, to be the minimum size of all cut-strategies $S$ of $G$. A graph $G$ is said to be $m$-neighbor-connected if $K(G) = m$, and critically $m$-neighbor connected if $K(G) = m$ and for any vertex $v$, $K(G/(v)) = m - 1$.

Gunther in 1975 and 1985 modeled the reliability of a spy network using the neighbor-connectivity of a graph.

A graph $G$ is a minimum critically $m$-neighbor-connected graph if no critically $m$-neighbor-connected graph with the same number of vertices has fewer edges than $G$. Cozzens and Wu give upper bounds on the minimum size of the critically $m$-neighbor-connected graphs of fixed order $v$ and show that the number of edges in a minimum critically $m$-neighbor-connected graph with order $v$, where $v$ is a multiple of $m$, is $\left\lfloor \frac{mv}{2} \right\rfloor$, hence such a graph is always $m$-regular.

Examples of $m$-neighbor connected graphs and methods of constructing $m$-neighbor-connected graphs will be given in this talk. Insight into the structure of this class of graphs will be provided. There are many open problems relating the parameter $K$ to other parameters of connectedness, and domination. These will be discussed.

3:10 - 3:50

Prof. Douglas R. Shier, Department of Mathematics
College of William and Mary

"Cancellation and Consecutive Sets"

The principle of inclusion and exclusion has been applied to numerous areas of discrete mathematics. One manifestation of this principle occurs in expressing the probability of the union of events in terms of the alternating sum of probabilities of intersections of events. If the events themselves are sufficiently well structured, then predictable cancellation occurs in this expansion. This talk discusses the special case of "consecutive sets," in which elements occur consecutively in every set. For such sets the inclusion-exclusion expansion assumes a particularly nice form, with all reduced coefficients being $\pm 1$. In fact the appropriate sign is determined by the length of a certain path in a graph derived from the incidence structure of the given sets.
4:10 - 4:50  
**Prof. Richard H. Schelp, Department of Mathematical Sciences**  
**Memphis State University**  

**Andrew Sobczyk Memorial Lecture**  

"The Local Ramsey Number and Local Colorings"  

A local k-coloring of a graph $H$ is a coloring of the edges of $H$ (by any number of colors) in such a way that the edges incident to each vertex of $H$ are colored with at most $k$ different colors. The local Ramsey number $r_{loc}^k(G)$ is defined as the smallest positive integer $m$ such that $K_m$ contains a monochromatic copy of $G$ for every local k-coloring of $K_m$. This Ramsey number exists and is at least as large as the usual Ramsey number $r^k(G)$ of $G$ for $k$ colors. Results and open questions will be presented for the local Ramsey number as well as for a generalization of local k-colorings.

7:30  

**Social, Jordan Room**

---

**Friday, October 12**

8:00  

Coffee and Doughnuts, Student Senate Chambers

8:10 - 8:50  

**Prof. Pravin Vaidya, Department of Computer Science**  
**University of Illinois**

"New algorithms for minimizing convex functions over convex sets"  

Let $S \subseteq \mathbb{R}^n$ be a convex set for which there is an oracle with the following property. Given any point $z \in \mathbb{R}^n$ the oracle returns a "Yes" if $z \in S$; whereas if $z \notin S$ then the oracle returns a "No" together with a hyperplane that separates $z$ from $S$. The feasibility problem is the problem of finding a point in $S$; the convex optimization problem is the problem of minimizing a convex function over $S$. We present a new class of algorithms for the feasibility problem based on enclosing the target set $S$ in a polytope whose volume shrinks geometrically at each step. A suitable center of the current polytope is used as a test point at each step. The new algorithms are faster than the previously best known algorithms by a factor proportional to $n$. The algorithms for the feasibility problem easily adapt to the convex optimization problem.
9:10 - 9:50  
Prof. Dijen K. Ray-Chaudhuri, Department of Mathematics  
Ohio State University  
"Size of an s-intersection family in a semilattice and construction of vector space designs by quadratic forms"

Let $v, k$ and $s$ be positive integers, $v \geq k + s$. Let $X$ be a $v$-set and $\mathcal{Q}$ be a set of subsets of $X$, each subset containing $k$ elements. $\mathcal{Q}$ is called a $k$-uniform $s$-intersection family if and only if $|\{r \cup b : a, b \in \mathcal{Q}, a \neq b\}| = s$. Ray-Chaudhuri and Wilson in their 1975 paper proved that if $\mathcal{Q}$ is a $k$-uniform $s$-intersection family of $v$-set $X$, then $|\mathcal{Q}| \leq \left(\frac{v}{s}\right)$. This theorem is generalized to a class of semilattices called polynomial semilattices which include many important combinatorial structures. Let $V$ be a $v$-dimensional vector space over a finite field of order $q$ and $\mathcal{B}$ be a family of $k$-dimensional subspaces of $\mathcal{B}$. The pair $(V, \mathcal{B})$ is called a $t$-$(v, k, \lambda, q, \ell)$ design iff every $t$-dimensional subspace $T$ of $V$ is contained in exactly $\lambda$ elements $B$ of $\mathcal{B}$. We construct several families of vector space designs for $t = 2$ and $3$ by using quadratic forms.

10:10 - 10:50  
Prof. Douglas B. West, Department of Mathematics  
University of Illinois-Urbana  
"A Graph-theoretic Game and its Application to the k-Server Problem"

We consider a zero-sum game played on the graph between a tree player and an edge player. The tree player chooses a spanning tree $T$ and the edge player chooses an edge $e$. If $e$ lies in $T$ then the payoff to the edge player is zero; otherwise, the payoff is the length of the unique cycle created when $e$ is added to $T$. We determine the value of the game for specific classes of graphs and derive an upper bound on the value for any $n$-vertex graph. These results yield new competitive randomized algorithms for the $k$-server problem on a wide class of metric spaces. For example, we obtain a $2k$-competitive algorithm (against oblivious adversaries) for the $k$-server problem on a circle. This is joint work with Noga Alon and Richard Karp.

11:10 - 11:50  
Prof. Michael Langston, Department of Computer Science  
University of Tennessee  
"Polynomial-Time Algorithms from Finite Basis Theorems - A Survey"

Traditionally, problems have been roughly classified as either "easy" or "hard", dependent on whether low-degree, polynomial-time, decision algorithms exist to solve them. Until recently, one could expect any proof of easiness to be constructive. That is, the proof itself should provide positive evidence in the form of the promised polynomial-time algorithm.

This appealing picture is dramatically altered, however, by recent "nonconstructive" developments in the theory of well-partially-ordered sets. New algorithmic characterizations are now possible that rely on finite but unknown bases of forbidden subgraphs.

In this talk we will survey some of the main results and open questions related to this general topic.

LUNCH
1:30 - 2:10  

**Prof. Gerd H. Fricke, Department of Mathematics and Statistics**  
Wright State University

"On the Product of the Independent Domination Numbers of a Graph and Its Complement"

Let $i(G)$ denote the smallest cardinality of an independent dominating set (equivalently maximal independent set) of vertices of a graph $G$. We will study $m_{ii}(p) = \max_{|G| = p} i(G)i(\overline{G})$, the maximum value over $p$ vertex graphs of the product of the independent domination numbers of a graph and its complement.

Recently Cockayne, Favoron, Li, and MacGillivray have shown that $i(G)i(\overline{G}) < \frac{p-8}{8}$. We will show that $m_{ii}(p)$ behaves like $\frac{p^2}{16}$ asymptotically by proving the following:

**Theorem:** Let $0 < k < 16$. Then there exists an integer $p_0$ such that

$$i(G)i(\overline{G}) \leq \frac{p^2}{k}$$

for any graph $G$ with $|G| = p \geq p_0$.

2:20 - 3:00

**Prof. Jeremy Spinrad, Department of Computer Science**  
Vanderbilt University

"Containment of Circular-Arcs"

The neighborhood containment matrix of an $n$ vertex graph is an $n$ by $n$ matrix $M$ such that $M[x, y] = 1$ exactly when $N(x) \cap \{y\}$ contains $N(y) \cap \{x\}$. This talk presents a method for determining the neighborhood containment matrix of a circular-arc graph in $O(n^2)$ time. Computing the neighborhood containment matrix was a bottleneck step, and possibly the only bottleneck step, of Tucker's recognition algorithm for circular-arc graphs. The techniques for computing this matrix involve reduction of the problem to containment problems on chordal bipartite graphs, and using special properties of chordal bipartite graphs. We also pose several open problems on chordal bipartite graphs.
3:10 - 3:50

Prof. Stephan Olariu, Department of Computer Science
Old Dominion University

"A Fast Parallel Recognition Algorithm for a Class of Tree-representable Graphs"

A number of problems in computational semantics, group-based cooperation, networking, examination scheduling, to name just a few, suggested the study of graphs featuring certain "local density" characteristics. Typically, the notion of local density is equated with the absence of chordless paths of length three or more. Recently, a new metric for local density has been proposed, allowing a number of such induced paths to occur. More precisely, a graph \( G \) is \( P_4 \)-sparse if no set of five vertices in \( G \) induces more than one chordless path of length three. \( P_4 \)-sparse graphs generalize the well-known class of cographs corresponding to a more stringent local density metric. One remarkable feature of \( P_4 \)-sparse graphs is that they admit a tree representation unique up to isomorphism. In this work we present a parallel algorithm to recognize \( P_4 \)-sparse graphs and show how the data structures returned by the recognition algorithm can be used to construct the corresponding tree representation. With a graph \( G = (V,E) \) with \( |V| = n \) and \( |E| = m \) as input, our algorithms run in \( O(\log n) \) time using \( O(\frac{n^2 + mn}{\log n}) \) processors in the EREW-PRAM model.

4:00 - 4:40

Prof. Mark Ellingham, Department of Mathematics
Vanderbilt University

"Vertex-switching reconstruction and pseudosimilarity"

A vertex-switching \( G_v \) of a graph \( G \) at a vertex \( v \) is obtained by deleting all edges incident with \( v \), and then adding all possible edges incident with \( v \) which were not in \( G \). A graph is vertex-switching reconstructible if it is determined by its collection of vertex-switchings. Two vertices \( u \) and \( v \) of \( G \) are vertex-switching pseudosimilar if they are not similar but \( G_u \) and \( G_v \) are isomorphic. We talk about some recent advances in the theory of vertex-switching reconstruction, including results on vertex-switching reconstruction of classes of graphs and a characterization of vertex-switching pseudosimilar vertices.
THE ERDÖS-KO-RADO THEOREM FOR SMALL FAMILIES

Prof. Richard A. Duke
Department of Mathematics
Georgia Tech. University
The Erdős-Ko-Rado Theorem for Small Families

R. Duke, V. Rödl

Let $\mathcal{N} = \{1,2,\ldots,n\}$ and

$$\mathcal{N}^k = \{A : A \subseteq \mathcal{N}, |A| = k\} \quad (k < \frac{n}{2})$$

Call $\mathcal{A}^t \subseteq \mathcal{N}^k$ $t$-intersecting if

$|A_i \cap A_j| \geq t$ for each $A_i, A_j \in \mathcal{A}^t$.

How large can such a $t$-intersecting family be?

Choosing $\mathcal{A}^t = \{A : A \subseteq \mathcal{N}^k, \{1,2,\ldots,t\} \subseteq A\}$ shows that

$$\max |\mathcal{A}^t| \geq \binom{n-t}{k-t}.$$ 

**Theorem (Erdős-Ko-Rado)** Given $k > t > 0$

There exists $n_0 = n_0(k,t)$ such that for $n > n_0$

If $\mathcal{A}^t$ is a $t$-intersecting family, $\mathcal{A}^t \subseteq \mathcal{N}^k$,

then

$$|\mathcal{A}^t| \leq \binom{n-t}{k-t}.$$ 

($n_0(k,t) = (k-t+1)(t+1)$, Frankl, 1978; Wilson, 1984)
Since $\left(\binom{n-k}{k-1}\right) \sim \left(\frac{k}{n}\right)^t \binom{n}{k}$, if $Q \leq \lceil n/2 \rceil$, $|a_i| \sim m \sim \left(\frac{k}{n}\right)^t$

and at $a_i$ is $t$-intersecting,

$$\max \frac{|a_i|}{|a|} \sim \left(\frac{k}{n}\right)^t.$$

Let $\Omega = \left\{A_i; i = 1, \ldots, m\right\}$ be a subfamily of $\binom{n}{k}$.

Consider the bipartite graph $G$ with vertex classes $X = \binom{n}{k}$, $Y = \left\{y_1, y_2, \ldots, y_m\right\}$ in which $\left\{l, y_i\right\}$ is an edge iff $l \in A_i$.

Simple averaging shows that there are $t$ vertices in $X$ all joined to the same $q$ vertices in $Y$, where

$$q \geq \frac{\binom{k}{t}}{\binom{n}{t}} \cdot m \sim \left(\frac{k}{n}\right)^t m.$$

Since these vertices of $Y$ correspond to a $t$-intersecting family

$$\max \frac{|a_i|}{|a|} \geq \frac{q}{m} \sim \left(\frac{k}{n}\right)^t.$$
Let $p_t(n,h,m)$ be the minimum of $\max \frac{|a^t|}{|a|}$ for $a^t \in A \subseteq \mathbb{F}_{m,n}^k$, $|a|=m$, $a^t$ $t$-intersecting.

By the E-K-R Theorem, for $m \sim \frac{n}{\log n}$ and $n$ large, we have $p_t(n,h,m) \sim (\frac{h}{n})^t$.

For all $m$ we have $p_t(n,h,m) \geq \frac{9}{m} \sim (\frac{h}{n})^t$.

Here we consider this function for small $m$.

In particular we have

**Theorem** For each $t$, if $m=n$ and $h=cn$, $0 < c < \frac{1}{2}$, then for $n$ sufficiently large $p_t(n,h,m) \sim c = (\frac{h}{n})$.

We show that for $A \subseteq \mathbb{F}_{m,n}^k$, $|a|=n$, $h=cn$, there exists a $t$-intersecting family $a^t \in A$ with $|a^t| = cn \left(1 - o(1)\right)$.
Sketch of the proof for $t=2$.

Suppose $\mathcal{A} = \{A_i; i = 1, \ldots, n\} \subseteq 2^{[n]}$, $h = cn$.

Consider the bipartite graph $G$ again. $G$ has $cn^2$ edges.

**Claim.** For $n$ sufficiently large we can delete $g(n)$ edges from $G$, $g(n) = o(n^2)$, so that if edges $\{l, y_i\}$ and $\{l, y_j\}$ remain, then $|A_i \cap A_j| > 2$.

![Diagram](https://via.placeholder.png)

If so, we are done!

Since then some vertex $l \in \mathcal{X}$ still has degree $\geq \frac{1}{n}(cn^2 - g(n)) = cn(1-o(1))$.

Its neighbors correspond to a $2$-intersecting family.
Proof of the claim:

For each $l \in [n]$ form a graph $G_l$ with vertex set $Y = \{y_1, y_2, \ldots, y_n\}$ in which $\{y_i, y_j\}$ is an edge iff $A_i \cap A_j = \emptyset$.

In each $G_l$ choose a maximal matching, $M_l$.

(Each edge of $G_l$ has an endpoint incident with an edge of $M_l$.)

For each $l$ if $\{y_i, y_j\}$ is in $M_l$ delete the edges $\{l, y_i\}$ and $\{l, y_j\}$ from $G$.

If $\{l, y_i\}$ and $\{l, y_j\}$ remain in $G_l$, then $|A_l \cap A_i \cap A_j| \geq 2$.

For otherwise, $\{y_i, y_j\}$ is in $G_l$ and at least one of $\{l, y_i\}, \{l, y_j\}$ would be missing.

It remains to show that $\left| \bigcup_{l=1}^{n} M_l \right| = o(n^2)$. 
Note that in $\bigcup_{l=1}^{n} M_l$ we cannot have
\[ e_1, e_2, e_3 \] with $e_1, e_3 \in M_l$, $e_2 \in M_j$.

This would require

\[ \begin{array}{cccc}
  y_1 & e_1 & y_2 & e_2 \\
  i & \Downarrow & \Downarrow & \Downarrow \\
 &  & j & \Downarrow \\
  y_3 & e_3 & y_4 & \Downarrow
\end{array} \]

But then $i \in A_2 \cap A_3$, so $A_2 \cap A_3 \neq \{j\}$.

**Theorem (Ruzsa, Szemerédi, 1978)** For $m$ sufficiently large if $G$ is a bipartite graph with $2m$ vertices and $cm^2$ edges which are the union of $\leq m$ matchings, then there exist matchings $M_i$ and $M_j$ and edges $e_1, e_2 \in M_i$, $e_2 \in M_j$, with $e_2$ incident with both $e_1$ and $e_3$.

Since $\bigcup_{l=1}^{n} M_l$ does not have such edges,

$|\bigcup_{l=1}^{n} M_l| = o(n^2)$. 

This result follows from Szemerédi's Regularity Lemma and was used to show the following:

Let \( U_3(n) = \max \{|S| : S \subseteq [n], S \text{ does not contain a 3-term arithmetic progression}\} \). Then \( U_3(n) = o(n) \).

Suppose \( S \subseteq [n] \) with \(|S| = cn \), \( c < c' \).

For each \( j \in [n] \) and each \( s \in S \) join \( j + r \) to \( j + 2r \).

This yields a matching. For each \( j \in [n] \) in a bipartite graph with \( n|S| = cn^2 \) edges.

\[ j+2r = k+2t \]
\[ j+s = k+t \]
\[ 2r - s = t \]
\[ r = \frac{s+t}{2} \]

This, \( S \subseteq [n], \) result insures that \( S, r = \frac{s+t}{2}, t \) form a 3-term arithmetic progression in \( S \).
Szemerédi's Regularity Lemma (Bipartite Version)

For a bipartite graph $H$ with vertex classes $\mathcal{X}$ and $\mathcal{Y}$ and $u \in \mathcal{X}$, $v \in \mathcal{Y}$:

$$ d(u,v) = \frac{e(u,v)}{|u||v|} $$

E-regular pair $(u,v)$

For each $T \subseteq U$, $S \subseteq V$ with $|T| \geq \varepsilon |U|$, $|S| \geq \varepsilon |V|$

$$ |d(T,S) - d(u,v)| < \varepsilon $$

Suppose $|\mathcal{X}| = |\mathcal{Y}| = n$.

Theorem (Szemerédi) For each $\varepsilon > 0$, there exist integers $N(\varepsilon)$, $K(\varepsilon)$ such that for $n > N(\varepsilon)$ there are partitions $\mathcal{X} = U_0 \cup U_1 \cup \ldots \cup U_k$ and $\mathcal{Y} = V_0 \cup V_1 \cup \ldots \cup V_k$, $k \leq K(\varepsilon)$, where $|U_0|, |V_0| < \varepsilon n$, $|U_1| = \ldots = |U_k|$, $|V_1| = \ldots = |V_k|$, and all but $\varepsilon k^2$ of the pairs $(U_i, V_j)$, $1 \leq i, j \leq k$, are E-regular.
Szemerédi's Regularity Lemma implies the result on matchings (in a bipartite graph).

Suppose G is a bipartite graph with n vertices in each class, cn² edges in at most n matchings.

Apply the Regularity Lemma (with suitable ε).

Delete all edges between irregular pairs.

Delete edges between pairs of low density. (e.g., ∆)

Many edges remain. (≥ ε²n²)

Some matching M still has many edges (≥ εn).

M must meet some U_i and some V_j in large subsets T and S, respectively. (≥ ε²n³)

Since edges join U_i and V_j, this is a dense and regular pair. Then d(S,T) is large. Edges from other matchings must also join T and S.
For the proof that $\alpha \leq cn^k$, $|A| = n$, $k = cn$ contains a $t$-intersecting subfamily of size $cn(1-o(1))$ when $t \geq 2$.

Again construct the $G_{i,j}$, $i \leq [n]$, now with $\{y_i, y_j\}$ an edge if $i \in A_i \cap A_j$ and $|A_i \cap A_j| < t$.

Choose maximal matchings $M_2$ and if $\{y_i, y_j\}$ is in one of them delete $\{y_i, y_j\}$ and $\{y_i, y_j\}$ from $G$ for each $p \in A_i \cap A_j$.

Show that if $y_i$ and $y_j$ still have a common neighbor, then $|A_i \cap A_j| \geq t$.

Szemerédi's Lemma can be used to show that with $cn^2$ edges in $\leq n$ matchings there exist matchings $M_0, M_1, \ldots, M_{t-1}$ and edges $e_0, e_1, \ldots, e_{t-1}$, $e^*_1, \ldots, e^*_{t-1}$, with $e_0 = \{y_i, y_j\}$ and for each $\mu$, $1 \leq \mu \leq t-1$, $e_{\mu}, e^*_{\mu}$ $e M_\mu$, $e_{\mu}$ incident with $y_i$, $e^*_{\mu}$ incident with $y_j$.

It is not hard to see that this configuration does not exist in $\bigcup_{\mu=1} M_{\mu}$.

So $\left| \bigcup_{\mu=1} M_{\mu} \right| = o(n^2)$. 
The E-K-R Theorem for Small Families

Let $C^m = E_3, \ldots, E_n$ and $C^{n^k} = \{ A : \text{A contains} \ (A \Delta k) \}$ (1 < k).

Call $A \times C^{n^k}$ $\epsilon$-INTERSECTING if

$|A_i \cap A_j| = \epsilon^i$ for each $A_i, A_j \in C^k$.

How large can such an $\epsilon$-intersecting family be?

Consider $A^k = \{ A : \text{A contains} \ (E_3, \ldots, E_k) \}$ which shows that

$\max |A^k| \propto (\epsilon^{-k})$.

Theorem (E-K-R): Given $k > 0$

There exists $r_k^m = n_k^m(k, \epsilon)$ such that for all $A^k$ if $A^k$ is an $\epsilon$-intersecting family, $\max |A^k| \leq (\epsilon^{-k})$.

$(n_k^m(k, \epsilon)) \propto (k^{-m})$, see full, 1974, Wass, 1974.

Let $\rho_k(m, k, m)$ be the minimum of $\max |A^k| / |A|$ for $A \in C^{n^k}, |A| = m, A^k$ $\epsilon$-intersecting.

By the E-K-R Theorem, for $m \geq 2$ and $m$ large, we have

$\rho_k(m, k, m) \approx (\epsilon^{-k})$.

For all $m$ we have

$\rho_k(m, k, m) \geq \sqrt{\frac{m}{2}} \approx (\epsilon^{-k})$.

Here we consider this function for small $m$.

In particular, we have

Theorem for small $\epsilon$, if $m = n$ and $k = cn$, $\epsilon < \frac{1}{c}$, then for $n$ sufficiently large

$\rho_k(n, k, n) \approx c = (\epsilon^{-k})$.

We show that for $A \in C^{n^k}, |A| = n, k = cn$, there exists a $\epsilon$-intersecting family $A^k$ with $|A^k| = c n (1 - o(1))$.

Since $|C^{n^k}| \geq (\epsilon^{-k})$ and in general $\epsilon$-$\Delta$ is $\epsilon$-INTERSECTING,

$\max |A^k| \propto (\epsilon^{-k})$.

Let $(n^k)_{1 \leq i \leq m}$ be a subset of $\{ E_3, \ldots, E_k \}$.

Consider the bipartite graph $G$ with vertex classes $E = \{ E_3, \ldots, E_k \}$.

In which $(x, y)$ is an edge if $x \in A_i$.

Simple averaging shows that there are $\epsilon$ vertices in $E$ all joined to the same $\epsilon$ vertices in $E$, where

$\epsilon \leq \frac{1}{k}, m \approx (\epsilon^{-k})$.

Since these vertices of $E$ correspond to an $\epsilon$-intersecting family

$\max |A^k| \geq \frac{1}{m} \approx (\epsilon^{-k})$.

Sketch of the proof for $k = 2$.

Suppose $G = C_n^2$, $E = C_n^2$, $k = c n$.

Consider the bipartite graph $G$ again. $G$ has $c n^2$ edges.

Claim: For $n$ sufficiently large we can delete $o(n)$ edges from $G$, $q(n) = o(n)$, so that if edges $\{ x, y \}$ and $\{ x', y' \}$ remain, then $\{ A_i \} \Delta A_j > 2$.

If so, we are done.

Since then some vertex $x \in C_n$ still has degree $\geq \frac{1}{n} (c n^2 - o(n)) = n \Omega (1 - o(1))$.

Its neighbors correspond to an $\Delta$-intersecting family.
Proof of the claim:

For each \( i \in \mathbb{N} \) form a graph \( G_i \) with vertex set \( V = \{e_1, e_2, \ldots, e_i\} \) in which \( \{e_i, e_j\} \) is an edge iff \( A_i \cap A_j = \emptyset \).

In each \( G_i \) choose a maximal matching, \( M_i \).

(Each edge of \( G_i \) has an endpoint incident with an edge of \( M_i \).)

For each \( i \) if \( \{e_i, e_j\} \) is in \( M_i \), delete the edges \( \{e_i, e_j\} \) and \( \{e_j, e_i\} \) from \( G_i \).

If \( \{e_i, e_j\} \) and \( \{e_j, e_k\} \) remain in \( G_i \), then \( \{e_i, e_j, e_k\} \) is a 3-term arithmetic progression.

For otherwise, \( \{e_i, e_j\} \) is in \( G_i \) and at least one of \( \{e_i, e_j\}, \{e_j, e_k\} \) would be missing.

It remains to show that \( \sum_{i=1}^{n} M_i = o(n^3) \).

This result follows from Szemerédi's Regularity Lemma which was used to show the following:

Let \( \mu(n) = \max\{\mu \in \mathbb{N} : \mathbb{E}(G, \mu) \text{ is a 3-term arithmetic progression}\} \). Then \( \nu(\mu) = o(n) \).

Suppose \( \{e_i, e_j\} \), with \( i < j \), occurs.

For each \( i \leq k \) and each \( k < j \) join \( e_i, e_k, e_j \). This yields a matching which in \( G_k \) a bipartite graph with \( n^3/\mu(n) \) edges.

The Szemerédi Regularity Lemma was used to show the following:

Let \( \mu(n) = \max\{\mu \in \mathbb{N} : \mathbb{E}(G, \mu) \text{ is a 3-term arithmetic progression}\} \). Then \( \nu(\mu) = o(n) \).

The Szemerédi Regularity Lemma was used to show the following:

Let \( \mu(n) = \max\{\mu \in \mathbb{N} : \mathbb{E}(G, \mu) \text{ is a 3-term arithmetic progression}\} \). Then \( \nu(\mu) = o(n) \).

The Szemerédi Regularity Lemma was used to show the following:

Let \( \mu(n) = \max\{\mu \in \mathbb{N} : \mathbb{E}(G, \mu) \text{ is a 3-term arithmetic progression}\} \). Then \( \nu(\mu) = o(n) \).

The Szemerédi Regularity Lemma was used to show the following:

Let \( \mu(n) = \max\{\mu \in \mathbb{N} : \mathbb{E}(G, \mu) \text{ is a 3-term arithmetic progression}\} \). Then \( \nu(\mu) = o(n) \).
Sierpinski's Regularity Lemma implies the result on matchings (in a bipartite graph).

Suppose $G$ is a bipartite graph with $n$ vertices in each class, $cn^2$ edges in at most $n$ matchings.

* Apply the Regularity Lemma (with $\epsilon n^2$).

Delete all edges between isomorphic pairs.

Delete edges between pairs of low density. ($\epsilon n$)

Many edges remain. ($\approx n^2$)

Some matching $M$ still has many edges ($\approx n$).

$M$ must meet some $U_i$ and some $V_j$ in large subsets $T$ and $S$, respectively. ($\approx 2n$)

Since edges join $U_i$ and $V_j$ this is a dense and regular pair. Then $d(T) > 1$.

Edges from other matchings must also join $T$ and $S$.

For the proof that $d(T) > 1$, $|T| = n$, $k < cn$ contains a $\epsilon$-intersecting subfamily of size $\epsilon n (1 - o(1))$ when $\epsilon > 2$.

Again construct the $G_j, j < n$, now with $\{y_j, z_j\}$ an edge in $2j A_j, A_j$, and $1j A_j, A_j$.

Choose maximal matchings $M_j$ and if $\{y_j, z_j\}$ is in one of them delete $\{y_j, z_j\}$ and $\{y_j, z_j\}$. From $G_j$ for each $p \in A_j$.

Show that if $y_j$ and $y_j$ still have a common neighbor, then $1j A_j, A_j$.

Sierpinski's Lemma can be used to show that with $cn^2$ edges in $n$ matchings there exist matchings $M_1, M_2, \ldots, M_n$, and edges $e_1, e_2, \ldots, e_n$, with $e_1 \in \{y_1, z_1\}$ and for each $M_j$, $1j A_j, e_j \in M_j$. For each $i$, $e_i$ incident with $y_j$, $e_i$ incident with $z_j$.

It is not hard to see that this configuration does not exist in $\cup M_j$.

So $|\cup M_j| = o(n^2)$. 
Critical m-neighbor-connected Graphs

Prof. Margaret B. Cozzens
Department of Mathematics
Northeastern University
APPLICATIONS AND BACKGROUND

Gunther and Hartnell in 1978 introduced the idea of neighbor connected graphs to model a spy network. The vertices of a graph $G$ are stations or people, the edges of $G$ represent lines of communication.

If a station is destroyed, the adjacent stations will be betrayed so that the betrayed stations become useless to the network as a whole.

Therefore, we want to see what happens to a network when not only vertices are removed, but when neighborhoods of vertices are removed.

The ultimate goal is to design networks with high neighbor connectivity at least cost, so that the network communications are compromised the least in attack scenarios.
DEFINITIONS

Let $G$ be a graph with $v$ vertices and $e$ edges.

**closed neighborhood of $u$:** $N[u] = \{u\} \cup N(u)$

**a subverted vertex $u$:** $N[u]$ is deleted from $G$

**$G/S$:** $G - N[S]$ where $S$ is a set of vertices of $G$

$S$ is a cut-strategy if $G/S$ is empty, complete or disconnected

$G$ is **$m$-neighbor connected** if

$m = \min\{|S|: S \text{ is a cut-strategy for } G\}$

$K(G)$ denotes the neighbor connectivity of $G$

$G$ is **critically $m$-neighbor connected** if $K(G) = m$, but $K(G/\{u\}) = m-1$ for all $u \in V(G)$

$G$ is **minimum critically $m$-neighbor connected** if no critically $m$-neighbor connected graph with the same number of vertices has fewer edges than $G$
CONSTRUCTION OF NEW GRAPHS

Given a graph $G$, create the collection $\mathcal{G}_G$:

(i) Each vertex $u$ of $G$ is replaced by a clique $C_u$ of order $\geq \text{degree}(u)$

(ii) $C_{u_1}$ and $C_{u_2}$ are joined by one edge if and only if $u_1$ and $u_2$ are adjacent in $G$

(iii) Each vertex of $C_u$ is adjacent to at most one vertex not in $C_u$

EXAMPLE
Given a graph $G$, create the collection $\mathcal{H}_G$ as follows:

(i) Each vertex $u$ of $G$ is replaced by a clique of order $\geq \text{degree}(u)$

(ii) Each clique is connected to another clique through a vertex called the courier if and only if the corresponding two vertices are connected in $G$.

(iii) Each vertex of a clique is connected to at most one courier.

**EXAMPLE**

![Diagram](image)
THEOREM 1: If $G$ is an $m$-connected graph then each member of $\mathcal{G}$ is an $m$-neighbor connected graph.

THEOREM 2: For any positive integers $m$ and $n$ such that $m > 1$ and $n \geq m+1$, there is a class of critically $m$-neighbor connected graphs, each of which has $n$ cliques.
THEOREM 3: Let $m$ be a positive integer. If $G$ is minimum critically $m$-neighbor connected with order $v$ and $\varepsilon$ edges then

$$\lceil \frac{1}{2}mv \rceil \leq \varepsilon \leq \lceil \frac{1}{2}mv + \frac{1}{2}mr \rceil$$

where $r$ is the remainder of $v/m$.

COROLLARY: If the order of $G$, $v$, is a multiple of $m$ and $G$ is a minimum critically $m$-neighbor connected graph then $\varepsilon = \lceil \frac{1}{2}mv \rceil$. 
RELATIONSHIP WITH OTHER PARAMETERS

The neighbor-connectivity number is less than or equal to the domination number.

\[ K(G) \leq \beta(G) \]

Therefore:

1. If a connected graph \( G \) does not contain \( P_4 \) or \( C_4 \) as induced subgraphs then \( K(G) = 1 \).
2. If a connected graph \( G \) does not contain \( P_5 \) or \( C_5 \) or \( K_{3+p} \) as induced subgraphs then \( K(G) \leq 2 \).

The neighbor-connectivity number is less than or equal to the connectivity number.

\[ K(G) \leq \kappa(G) \]

QUESTIONS:

1. When are they the same?
2. What graphs on \( v \) vertices maximize both the connectivity and the neighbor connectivity simultaneously?
Define the **vertex-neighbor integrity** of a graph $G$ to be:

$$NI(G) = \min \{|S| + w(G/S)|$$

where $w(G/S)$ is the size of the largest component in $G/S$ and the minimum is taken over all cut strategies $S$.

3. For fixed $v$, what graphs on $v$ vertices maximize the vertex-neighbor integrity?

4. For fixed $v$, what graphs on $v$ vertices maximize the vertex-neighbor integrity and the neighbor connectivity simultaneously?
APPLICATIONS AND BACKGROUND

Gunther and Hartnell in 1978 introduced the idea of neighbor connected graphs to model a spy network. The vertices of a graph G are stations or people, the edges of G represent lines of communication.

If a station is destroyed, the adjacent stations will be betrayed so that the betrayed stations become useless to the network as a whole.

Therefore, we want to see what happens to a network when not only vertices are removed, but when neighborhoods of vertices are removed.

The ultimate goal is to design networks with high neighbor connectivity at least cost, so that the network communications are compromised the least in attack scenarios.

DEFINITIONS

Let G be a graph with v vertices and e edges.

closed neighborhood of u: N(u) = {u} U N(u)

a subverted vertex u: N[u] is deleted from G

G/S: G - N[S] where S is a set of vertices of G

S is a cut-strategy if G/S is empty, complete or disconnected

G is m-neighbor connected if

m = min(|S|: S is a cut-strategy for G)

K(G) denotes the neighbor connectivity of G

G is critically m-neighbor connected if K(G) = m, but K(G/{u}) = m-1 for all u ∈ V(G)

G is minimum critically m-neighbor connected if no critically m-neighbor connected graph with the same number of vertices has fewer edges than G

CONSTRUCTION OF NEW GRAPHS

Given a graph G, create the collection $\mathcal{K}_G$ as follows:

(i) Each vertex u of G is replaced by a clique $C_u$ of order $\geq$ degree(u)

(ii) $C_{u_1}$ and $C_{u_2}$ are joined by one edge if and only if $u_1$ and $u_2$ are adjacent in G.

(iii) Each vertex of $C_u$ is adjacent to at most one vertex not in $C_u$

EXAMPLE

![Diagram of G, $\mathcal{K}_G$, and $\mathcal{K}_G$]

Given a graph G, create the collection $\mathcal{K}_G$ as follows:

(i) Each vertex u of G is replaced by a clique of order $\geq$ degree(u)

(ii) Each clique is connected to another clique through a vertex called the courier if and only if the corresponding two vertices are connected in G.

(iii) Each vertex of a clique is connected to at most one courier.

EXAMPLE

![Diagram of G, $\mathcal{K}_G$, and $\mathcal{K}_G$]
THEOREM 1: If $G$ is an $m$-connected graph then each member of $\mathcal{G}$ is an $m$-neighbor connected graph.

THEOREM 2: For any positive integers $m$ and $n$ such that $m > 1$ and $n \geq m + 1$, there is a class of critically $m$-neighbor connected graphs, each of which has $n$ cliques.

Theorem 3: Let $m$ be a positive integer. If $G$ is minimum critically $m$-neighbor connected with order $v$ and $\varepsilon$ edges then
$$\left\lfloor \frac{manu}{m} \right\rfloor \leq \varepsilon \leq \left\lfloor \frac{manu + manv}{m} \right\rfloor$$
where $r$ is the remainder of $v/m$.

COROLLARY: If the order of $G$, $v$, is a multiple of $m$ and $G$ is a minimum critically $m$-neighbor connected graph then $\varepsilon = \left\lfloor \frac{manu}{m} \right\rfloor$.

**RELATIONSHIP WITH OTHER PARAMETERS**

The neighbor-connectivity number is less than or equal to the domination number,
$$K(G) \leq \beta(G)$$
Therefore:
1. If a connected graph $G$ does not contain $P_4$ or $C_4$ as induced subgraphs then $K(G) = 1$.
2. If a connected graph $G$ does not contain $P_5$ or $C_5$ or $K_3 + p$ as induced subgraphs then $K(G) = 2$.

The neighbor-connectivity number is less than or equal to the connectivity number.
$$K(G) \leq \kappa(G)$$

**QUESTIONS:**
1. When are they the same?
2. What graphs on $v$ vertices maximize both the connectivity and the neighbor connectivity simultaneously?

Define the vertex-neighbor integrity of a graph $G$ to be:
$$NI(G) = \min (|S| + w(G/S))$$
where $w(G/S)$ is the size of the largest component in $G/S$ and the minimum is taken over all cut strategies $S$.

3. For fixed $v$, what graphs on $v$ vertices maximize the vertex-neighbor integrity?

4. For fixed $v$, what graphs on $v$ vertices maximize the vertex-neighbor integrity and the neighbor connectivity simultaneously?
CANCELLATION

AND

CONSECUTIVE SETS

D. R. Shier

M. H. McIlwain*

* Supported by NSF-REU at College of William & Mary, Summer 1990.
COHERENT SYSTEMS

In general, have a set of components

\[ E = \{1, 2, \ldots, n\} \]

Component \(i\): fails with probability \( q_i = 1 - p_i \)

(independently)

For subsets \( X \subseteq E \)

\[ \Phi(X) = \begin{cases} 
1 & \text{if system operates when components in } X \text{ operate, } E - X \text{ fail} \\
0 & \text{otherwise} 
\end{cases} \]

Coherent system: \( X \subseteq Y \Rightarrow \Phi(X) \leq \Phi(Y) \)

pathset: minimal \( S \subseteq E \) such that \( \Phi(S) = 1 \).

A coherent system completely described by \( E \) and \( \Phi \), collection of pathsets.

PROBLEM: Calculate \( R = \Pr[\Phi(X) = 1] \).

If \( F_i \) all components in pathset \( S_i \) operate \( \Rightarrow \)

\[ R = \Pr[F_1 \cup F_2 \cup \ldots \cup F_k] \]
INCLUSION-EXCLUSION APPROACH

Coherent system \((E, \mathcal{J})\)

- Components \(E = \{1, 2, \ldots, n\}\)
- Path sets \(\mathcal{J} = \{S_1, S_2, \ldots, S_k\}\)

WHEN is there interesting cancellation in I/E formula?

\[
R = \Pr [E_1 \cup E_2 \cup \ldots \cup E_k] = \sum_{i} \Pr [E_i] - \sum_{i<j} \Pr [E_i E_j] + \ldots
\]

Ex. 1: \(S_1 = \{1, 2, 4\}, \ S_2 = \{2, 3\}, \ S_3 = \{1, 3, 4\}\)

\[
R = p_1 p_2 p_4 + p_2 p_3 + p_1 p_3 p_4 - 2 p_1 p_2 p_3 p_4
\]

Ex. 2: \(S_1 = \{1, 2\}, \ S_2 = \{2, 3, 4\}, \ S_3 = \{3, 4, 5\}, \ S_4 = \{5, 6\}\)

\[
R = p_1 p_2 + p_2 p_3 p_4 + p_3 p_4 p_5 + p_5 p_6 - p_1 p_2 p_3 p_4 - p_1 p_2 p_5 p_6 - p_2 p_3 p_4 p_5 - p_3 p_4 p_5 p_6 - p_1 p_2 p_3 p_4 p_5 p_6
\]

\(9\) terms \((\pm 1)\) versus \(15\) possible.
Consecutive Systems

\[ S = \{ S_1, S_2, \ldots, S_k \} \] is a consecutive system on
\[ E = \{ 1, 2, \ldots, n \} \] if each \( S_j \) contains consecutive elements of \( E \):
\[ S_j = [l_j, r_j] \]

**Main Result.** The \( \pm 1 \) property holds for consecutive systems. Moreover, there is
a nice interpretation of \( 0, +1, -1 \) coeffs.

It suffices to study the coefficients
\[ d(i, i+1, \ldots, n) \]
of \( P_1 P_{i+1} \ldots P_n \) in the I/E expansion for \( S \).

**Fundamental Tool.**

\[
Pr [\Phi = 1] = (1 - p_e) Pr [\Phi = 1 | \overline{e}] + p_e Pr [\Phi = 1 | e]
\]

Fails

Works

Can be repeatedly applied to find \( d(i, i+1, \ldots, n) \).
EXAMPLE

\[ S_6 : \{1, 2, 3\} \]
\[ S_5 : \{3, 4, 5, 6\} \]
\[ S_4 : \{4, 5, 6, 7\} \]
\[ S_3 : \{6, 7, 8\} \]
\[ S_2 : \{7, 8, 9\} \]
\[ S_1 : \{9, 10, 11\} \]

\[ \text{In } \{ S_1 \} : \quad d(9, \ldots, 11) = +1 \]
\[ \text{In } \{ S_1, S_2 \} : \quad d(7, \ldots, 11) = -1 \]
\[ \text{In } \{ S_1, S_2, S_3 \} : \quad d(6, \ldots, 11) = ? \]

\[
Pr[\Phi = 1] = (1 - p_6) Pr[\Phi = 1 | 6] + p_6 \underbrace{Pr[\Phi = 1 | 6]}_{(1 - p_7) Pr[\Phi = 1 | 67] + p_7 \underbrace{Pr[\Phi = 1 | 67]}_{(1 - p_8) Pr[\Phi = 1 | 678] + p_8 \underbrace{Pr[\Phi = 1 | 678]}_{= 1}}} \]

Now equate coeffs of \( p_6, p_7, \ldots, p_{11} \):

\[
d(6, \ldots, 11) = -\left\{ d(7, \ldots, 11 | 6) + d(8, \ldots, 11 | 67) + d(9, \ldots, 11 | 678) \right\} - 1 \quad 0 \quad + 1 
= 0
\]
RECURSION

If \( S_j = [l_j, r_j] \) then

\[
d(l_j, ..., n) = - \left[ d(l_{j+1}, ..., n | \bar{l}_j) + d(l_{j+2}, ..., n | l_j, \bar{l}_{j+1}) + \ldots + d(r_{j+1}, ..., n | l_j, ..., \bar{r}_j) \right]
\]

Certain of the terms are automatically 0; others are \( d(r, ..., n) \) in the subsystem \( \{ S_1, ..., S_m \} \).

GIVEN sets \( S_k, S_{k-1}, ..., S_1 \) ordered by increasing \( l_j \), DEFINE consecutive union graph, with vertex \( v \) for each set \( S_v \) and directed edges \((v, w)\), \( v > w \), if \( S_v \cup S_w \) is consecutive: \( r_v + 1 > l_w \).

\[
S_6 : \{1, 2, 3, 4\}
S_5 : \{3, 4, 5, 6\}
S_4 : \{4, 5, 6, 7\}
S_3 : \{6, 7, 8\}
S_2 : \{7, 8, 9\}
S_1 : \{9, 10, 11\}
\]

\[
x_i = d(d_i, l_i, ..., n)
\]

\[
d_i^+ = \text{outdegree of } i
\]

Recursion on sets, not components.

Induction shows \( x_i \in \{-1, 0, 1\} \).
C.U. Graphs

Which graphs can arise as C.U. graphs G?

Note: $(i, j) \in G \Rightarrow (i, r) \in G, \ i < r < j$

What conditions on $d_i^+$?
Always have $d_i^+ = 0$ and for $i > 1$

1. $d_i^+ \leq i - 1$
2. $d_{i+1}^+ \leq d_i^+ + 1$

These conditions on (consecutive) outdegrees characterize C.U. graphs; e.g.

\[
\{d_4, d_3, d_2\} = \begin{cases} 1 & 1 & 1 \\ 2 & 1 & 1 \\ 1 & 2 & 1 \\ 2 & 2 & 1 \\ 3 & 2 & 1 \end{cases}
\]

5 such graphs for $k = 4$

Can show

\[
\text{# C.U. graphs on } k \text{ vertices} = \binom{k}{k-1} = \frac{1}{k} \binom{2k-2}{k-1}
\]
ANOTHER VIEWPOINT

Characterize when \( x_r = d(r, \ldots, n) \) is 0, -1, +1?

Recall example:

![Diagram of a graph with vertices labeled 1 to 6 and edges connecting them.]

The \( \{x_r\} \) satisfy:

\[
\begin{align*}
  x_1 &= 1 \\
  x_2 &= -x_1 \\
  x_3 &= -x_2 - x_1 \\
  x_4 &= -x_3 - x_2 \\
  x_5 &= -x_4 - x_3 - x_2 \\
  x_6 &= -x_5 - x_4
\end{align*}
\]

or

\[
\begin{align*}
  x_1 &= 1 \\
  x_1 + x_2 &= 0 \\
  x_1 + x_2 + x_3 &= 0 \\
  x_2 + x_3 + x_4 &= 0 \\
  x_2 + x_3 + x_4 + x_5 &= 0 \\
  x_4 + x_5 + x_6 &= 0
\end{align*}
\]

Solve \( Ax = e_1 \), where \( A = (a_{ij}) \) is unit lower triangular with \( a_{ij} = 1 \) for \( i - d_i^+ \leq j \leq i \).

\( e_1 \) is unit vector \((1, 0, \ldots, 0)^t\)
LINEAR SYSTEM

See that $A$ has consecutive 1's in rows & in columns: particularly easy to solve $Ax = e_1$.

LARGER EXAMPLE:

\[
\begin{array}{cccccccccc}
+ & - & + & - & + & - & + & - & + & - \\
x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} x_{11}
\end{array}
\]

\[
\begin{array}{c}
S_1 \\
S_2 \\
S_3 \\
S_4 \\
S_5 \\
S_6 \\
S_7 \\
S_8 \\
S_9 \\
S_{10} \\
S_{11}
\end{array}
\]

\[
\Rightarrow x = (1, -1, 1, -1, 0, 0, 1, -1, 1, -1)
\]

a path $1 \rightarrow 2$ using edge $(10, 11)$
Since in general $A$ is totally unimodular, the system $Ax = e_1$ has solution $x$ with entries in $\{-1, 0, 1\}$.

**NOTE:** +1 entries produce $(1, 1, \ldots, 1, 0, \ldots, 0)^t$

-1 entries produce $(0, -1, \ldots, -1, 0, \ldots, 0)^t$

Define another graph $T(A)$ to indicate how these positive (negative) columns fit together:

$$d_i \begin{cases} \begin{array}{c} + \end{array} \end{cases} \text{ vertex } i \sim \text{ column } i$$

$$\begin{array}{c} + \end{array} \begin{cases} \begin{array}{c} + \end{array} \end{cases} \text{ edge } (i, j) \text{ for } j = i + d_i + 1$$

Convenient to append a new row & column to $A$ with $a_{k+1, k+1} = 1$, other entries 0.

Then each $i \neq k+1$ has a unique successor $j \Rightarrow$ $T(A)$ is a tree rooted at vertex $k+1$. 
RESULT

Theorem. Let $P$ be the path joining 1 and 2 in $T(A)$. Then

$$P \text{ contains } (k,k+1) \iff x_k \neq 0.$$  

Moreover, in this case, $x_k = (-1)^{|P|+1}$

Comments:

1. $T(A) - T(S)$ can be directly constructed from

$$S = \{S_k, ..., S_i\} : (i,j) \in T(A) \iff j = i + d_i + 1.$$  

2. Once $T(A)$ is constructed, the path joining $j$ and $j+1$ determines the coefficient $d(l_k, ..., r_j)$ in

$$S_k = [l_k, r_k]$$

$$\vdots$$

$$S_j = [l_j, r_j]$$

3. By coalassing vertices $k, k+1 \rightarrow k$, get the appropriate tree for system with $S_k$ removed.
Example

\[ S_4 : \{1, 2\} \]
\[ S_3 : \{2, 3, 4\} \]
\[ S_2 : \{3, 4, 5\} \]
\[ S_1 : \{5, 6\} \]

Nonzero Coeffs.
\[ \{S_4, \ldots, S_i\} \]

1-2:  \[ + P_1 P_2 P_3 P_4 P_5 P_6 \]
3-4:  \[ - P_1 P_2 P_3 P_4 \]
4-5:  \[ + P_1 P_2 \]

\[ \{S_3, \ldots, S_i\} \]

2-3:  \[ - P_2 P_3 P_4 P_5 \]
3-4:  \[ + P_2 P_3 P_4 \]

\[ \{S_2, \ldots, S_i\} \]

1-2:  \[ - P_3 P_4 P_5 P_6 \]
2-3:  \[ + P_3 P_4 P_5 \]

\[ \{S_1, \ldots, S_i\} \]

1-2:  \[ + P_5 P_6 \]
Nonconsecutive Terms

Construction of $T(f)$ enables determination of coefficient $d(v, v+1, ..., w)$ for $p_v p_{v+1} ... p_w$ in the $I/E$ expansion of $f$.

There can be other terms $A_1, A_2, ..., A_r$ each corresponding to (maximal) sets of consecutive elements: e.g. $A_1 = \{1, 2, 3\}$, $A_2 = \{5, 6, 7, 8\}$

**Theorem.** $d(A_1 A_2 ... A_r) = (-1)^{r+1} d(A_1) d(A_2) ... d(A_r)$

**Previous example:**

\[ d(1, 2, 5, 6) = -d(1, 2) d(5, 6) = -1 \]
SUMMARY

Inclusion-exclusion expansion

\[ \Pr [ E_1 \cup \ldots \cup E_k ] \]

predictable cancellation?

Consecutive sets \( S_1, S_2, \ldots, S_k \)

Recursion

consecutive union graph

uses outdegrees

Linear system

based on indegrees

\( T(\mathcal{G}) \)

character of \( j, j+1 \) path in \( T(\mathcal{G}) \)

Extension

column consecutive systems

\( S_3 : \{1, 2, 3\} \)

\( S_2 : \{2, 3, 4, 5\} \)

\( S_1 : \{3, 4, 6\} \)
CANCELLATION
AND
CONSECUTIVE SETS

D. R. Shier
M. H. McIlwain

* Supported by NSF-REU at College of
William & Mary, Summer 1990.

INCLUSION-EXCLUSION APPROACH

Coherent system \((E, S)\)

- components \(E = \{1, 2, \ldots, n\}\)
- pathsets \(S = \{S_1, S_2, \ldots, S_k\}\)

WHEN is there interesting cancellation in I/E formula?

\[ R = \Pr[\bigcap_{j=1}^{k} E_j] = \sum_{\phi(S) = \emptyset} \Pr[E_1] \cdot \Pr[E_2] \cdot \ldots \cdot \Pr[E_k] + \ldots \]

Example 1: \(S_1 = \{1, 2, 4\}, S_2 = \{2, 3\}, S_3 = \{2, 3, 4\}\)

\[ R = p_1 p_2 p_4 + p_1 p_3 + p_2 p_3 p_4 - 2 \cdot p_1 p_2 p_3 p_4 \]

Example 2: \(S_1 = \{1, 2\}, S_2 = \{2, 3, 4\}, S_3 = \{3, 4, 5\}, S_4 = \{5, 6\}\)

\[ R = p_1 p_2 + p_1 p_3 + p_3 p_4 + p_2 p_3 - p_2 p_3 p_4 - p_1 p_2 p_4 + p_1 p_2 p_3 p_4 + p_1 p_2 p_3 - p_1 p_2 p_3 p_4 + p_1 p_2 p_3 p_4 p_5 \]

\(G^2\) terms \((\geq 1)\) versus \(1\) possible.

COHERENT SYSTEMS

In general, have a set of components

\(E = \{1, 2, \ldots, n\}\)

Component \(i\): fails with probability \(q_i = 1 - p_i\)

(Independently)

For subsets \(X \subseteq E\)

\[ \Phi(X) = \begin{cases} 1 & \text{if system operates when components in } X \text{ operate, } E \setminus X \text{ fail} \\ 0 & \text{otherwise} \end{cases} \]

Coherent system: \(X \subseteq Y \Rightarrow \Phi(X) \leq \Phi(Y)\)

Pathset: minimal \(X \subseteq E\) such that \(\Phi(X) = 1\)

A coherent system completely described by \(E\) and \(S\), collection of pathsets.

PROBLEM: Calculate \(R = \Pr[\Phi(X) = 1]\).

If \(E\)-all components in pathset \(S_j\) operate \(\Rightarrow\)

\[ R = \Pr[\bigcap_{j=1}^{k} E_j] = \sum_{\phi(S) = \emptyset} \Pr[E_1] \cdot \Pr[E_2] \cdot \ldots \cdot \Pr[E_k] + \ldots \]

CONSECUTIVE SYSTEMS

\(S = \{S_1, S_2, \ldots, S_k\}\) is a consecutive system on \(E = \{1, 2, \ldots, n\}\) if each \(S_j\) contains consecutive elements of \(E\):

\(S_j = \{\delta_j, \gamma_j\}\)

MAIN RESULT: The \(11\) property holds for consecutive systems. Moreover, there is a nice interpretation of \(C_i + i, -1\) coeffs.

It suffices to study the coefficients \(d(i, i+1, \ldots, n)\) of \(p_i p_{i+1} \ldots p_n\) in the I/E expansion for \(S\).

FUNDAMENTAL TOOL

\[ \Pr[\Phi = 1] = (-p_0) \Pr[\Phi = 1 | \emptyset] + p_0 \Pr[\Phi = 1 | \emptyset] \]

\(\text{fails}\)

\(\text{works}\)

Can be repeatedly applied to find \(d(i, i+1, \ldots, n)\).
EXAMPLE

\[ S_0 : \{1, 2, 3\} \]
\[ S_1 : \{3, 4, 5, 6\} \]
\[ S_2 : \{4, 5, 6, 7\} \]
\[ S_3 : \{3, 4, 7\} \]
\[ S_4 : \{1, 10, 11\} \]

In \( S_1 \): \( d(9, 11) = +1 \)
In \( S_2 \): \( d(7, 11) = -1 \)
In \( S_3 \): \( d(6, 11) = ? \)

\[ d(9, 11) = \frac{(1-p_3) \cdot (1-p_2) \cdot (1-p_1)}{1-p_3 \cdot (1-p_2) \cdot (1-p_1)} \]

Now equate coeffs of \( p_1 p_2 \): \( p_2 = 1 \)
\[ d(6, 11) = \left( d(7, 11) + d(8, 11) - d(9, 11) \right) \]
\[ = 0 \]

RECURSION

If \( S_j \in [a_j, c_j] \) then

\[ d(l_j, n) = d(l_j, n) + d(l_j, n) d(l_j, n) \]

Certain of the terms \( ^* \) are automatically \( 0 \); others are \( d(l_j, n) \) in the sub-system \( [S_l, \ldots, S_m] \).

GIVEN sets \( S_0, S_1, \ldots, S_k \) ordered by increasing \( l_j \).

DEFINe consecutive union graph, with vertex \( \gamma_i \) for each set \( S_i \) and directed edges \( (v, w), v > w \), if \( S_v U S_w \) is consecutive: \( \gamma_i = \gamma_j \).

ANOTHER VIEWPOINT

Characterize when \( x_i d(l_j, n) \) is \( 0, 1, 1 \)?

Recall example:

\[ \begin{array}{ccccccc}
    & x_6 & x_5 & x_4 & x_3 & x_2 & x_1 \\
    \hline
    x_i & 0 & 1 & 0 & 1 & 0 & 1 \\
\end{array} \]

The \( \{x_i\} \) satisfy:

\[ \begin{array}{ccccccc}
    x_1 = 1 \\
    x_2 = x_1 \\
    x_3 = 1 - x_1 - x_3 \\
    x_4 = -x_3 - x_4 \\
    x_5 = -x_4 - x_5 - x_6 \\
    x_6 = -x_5 - x_6 \\
\end{array} \]

or

\[ \begin{array}{ccccccc}
    x_1 & 0 & 1 & 0 & 1 & 0 & 1 \\
    x_2 & 1 & 0 & 1 & 0 & 1 & 0 \\
    x_3 & 1 & 0 & 1 & 0 & 1 & 0 \\
    x_4 & 1 & 0 & 1 & 0 & 1 & 0 \\
    x_5 & 1 & 0 & 1 & 0 & 1 & 0 \\
    x_6 & 1 & 0 & 1 & 0 & 1 & 0 \\
\end{array} \]

Solve \( Ax = e_i \), where \( A = (a_{ij}) \) is unit lower triangular

\( a_{ij} = 1 \) for \( i = l_j \).

\( e_i \) is unit vector \( (1, 0, \ldots, 0)^t \).
LINEAR SYSTEM

See that $A$ has consecutive $1$'s in rows $1$ in columns, particularly easy to solve $Ax = e_1$.

Large Example:

\[
\begin{array}{cccccccc}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
\end{array}
\]

\[
x = \begin{pmatrix}
1 \\
2 \\
3 \\
4 \\
5 \\
6 \\
7 \\
8
\end{pmatrix}
\]

Since in general $A$ is totally unimodular, the system $Ax = e_1$ has solution $x$ with entries in $\{1, -1, 0\}$.

Note: $+1$ entries produce $(1, 3, 5, 7, 9, 11, 13, 15)$

Define another graph $T(A)$ to indicate how these positive (negative) columns fit together

- vertex $i = \text{column } i$
- edge $(i, j)$ for $j = i + d_i^e$

Convenient to append a new row $K+1$, columns to $A$ with $a_{k+1, k+1} = 1$, other entries $0$.

Then each $i \neq k+1$ has a unique successor $j \Rightarrow T(A)$ is a tree rooted at vertex $k+1$.

RESULT

Theorem. Let $P$ be the path joining $1$ and $2$ in $T(A)$.

$P$ contains $(k, k+1)$ for $k \neq 0$.

Moreover, in this case, $x_k = (-1)^{k+1}$

Comments:

1. $T(A) = T(\delta)$ can be directly constructed from

   \[ S^0 = \{ S_1, \ldots, S_8 \}; \ (i, j) \in T(A) \Rightarrow j = i + d^e i + 1 \]

2. Once $T(A)$ is constructed, the path joining $j$ and $j+1$ determines the coefficient $d_i j, r_j$ in

   \[ S_k = [d_k, r_k] \]

   \[ S_j = [d_j, r_j] \]

3. By contracting vertices $k, k+1 \rightarrow k$, get the appropriate tree for system with $S_k$ removed.

EXAMPLE

\[
\begin{array}{c}
1 \\
2 \\
3 \\
4 \\
5 \\
6 \\
7 \\
8 \\
\end{array}
\]

\[
\begin{array}{c}
S_1: \{1, 2\} \\
S_2: \{2, 3, 4\} \\
S_3: \{3, 4, 5\} \\
S_4: \{5, 6\} \\
\end{array}
\]

Monge's Graphs.

\[
\begin{array}{c}
S_1, \ldots, S_4 \\
\end{array}
\]

1-2: $+P, PA, PA, PA, PA, PA$

2-3: $-P, PA, PA, PA, PA$

3-4: $+P, PA, PA, PA, PA$

4-5: $+P, PA, PA, PA, PA$

5-6: $+P, PA, PA, PA, PA$

1-2: $-PA, PA, PA, PA, PA$

2-3: $+PA, PA, PA, PA, PA$

3-4: $+PA, PA, PA, PA, PA$

4-5: $+PA, PA, PA, PA, PA$

5-6: $+PA, PA, PA, PA, PA$

\[
\begin{array}{c}
1 \\
2 \\
3 \\
4 \\
5 \\
6 \\
7 \\
8 \\
\end{array}
\]

\[
\begin{array}{c}
S_1, \ldots, S_8 \\
\end{array}
\]

1-2: $+PA, PA, PA, PA, PA, PA$

2-3: $-PA, PA, PA, PA, PA, PA$

3-4: $+PA, PA, PA, PA, PA, PA$

4-5: $+PA, PA, PA, PA, PA, PA$

5-6: $+PA, PA, PA, PA, PA, PA$

6-7: $+PA, PA, PA, PA, PA, PA$

7-8: $+PA, PA, PA, PA, PA, PA$
**Nonconsecutive Terms**

Construction of $T(d)$ enables determination of coefficient $d(v_1, v_2, ..., v_w)$ for $P_v P_{v_1}...P_w$ in the $I/E$ expansion of $T$.

There can be other terms $A_1, A_2, ..., A_r$ each corresponding to (maximal) sets of consecutive elements: e.g. $A_1 = \{1, a\}$, $A_2 = \{5, b, 3\}

**Theorem:** $d(A_1, A_2, ..., A_r) = (-1)^{m-1} d(A_1) d(A_2) ... d(A_r)$

Previous example:

$d(1, 2, 5, 6) = -d(1, 2) d(5, 6) = -1$

**Summary**

Inclusion-exclusion expansion

$Pr \left[ E \cup ... \cup E_k \right]$ predictable cancellation

Consecutive sets $S_1, S_2, ..., S_k$

Recursion

consecutive union graph uses outdegrees

Linear system

based on in-degrees $T(d)$ character of $i, j+1$ path in $T(d)$

Extension

column consecutive systems

$S_3: \{1, 2, 3\}$

$S_4: \{2, 3, 4, 5\}$

$S_5: \{3, 4, 5\}$
Andrew Sobczyk Memorial Lecture

The Local Ramsey Number
and Local Colorings

Prof. Richard H. Schelp
Department of Mathematical Sciences
Memphis State University
The Local Ramsey Number
and
Local Colorings
by
R. H. Schelp

Def. A local $k$-coloring of a graph $H$ is a coloring of its edges in such a way that the edges incident to each vertex of $H$ are colored by at most $k$ different colors. The local Ramsey number $R_{loc}(G)$ of a graph $G$ is the smallest positive integer $n$ such that $K_n$ contains a monochromatic copy of $G$ under each local $k$-coloring.

References


3) M. Truszczynski, Generalized Local Colorings of Graphs, To appear in JCTB.

4) R.H. Clapham and R.H. Schelp, Local Edge Colorings that are Global, in preparation.

Observations:

1) For $k$ fixed and $m$ large a local $k$-coloring of $K_m$ can use a large number of colors independent of $k$.

$$K_m = \begin{array}{cccc}
& & & \vdots \\
& c_1 & & \\
& & & \vdots \\
& & & c_m
\end{array}$$

Here all edges of $K_m$ are colored with color $c_i$ except for the shown $\left\lceil \frac{m}{3} \right\rceil$ matching edges.

2) If $R(k)\{G\}$ denote the usual Ramsey number, then $R_{loc}(G) \leq R(k)\{G\}$ since each coloring of a $K_m$ by at most $k$ colors is a local $k$-coloring.
Theorem 1 (Existence) For each $k \geq 3$, $P_{k^m}(G) \leq \binom{k^m(n-2)+1}{k-1}$.

Proof. Let $G$ be a local $k$-colored complete graph with $\geq \binom{k^m(n-2)+1}{k-1}$ vertices. We show $G$ contains a spanning tree $T$, rooted at a fixed vertex $x_0$, such that the following holds:

(i) Each $x \in V(T)$ has at most $k$ successors $x_1, x_2, \ldots, x_a$ ($a \leq k$) with different colors assigned to each edge $xx_i$, $1 \leq i \leq a$.

Select a fixed vertex $x_{i_0}$ in each $I_{i_0}$ and partition $I_{i_0}$ as was just done for $V(G)$ replacing $x_0$ by $x_{i_0}$.

Condition (i) there exists a path $P$ from the root $x_0$ to a vertex $y$ with at least $k^m(n-2)+1$ edges.

By condition (ii) $x_0$ is incident to edges colored $c_1, c_2, \ldots, c_k$. Therefore, the edges of $P$ get at most $k$ colors.

Hence there exist $m-1$ edges, say $x_0, x_1, x_2, \ldots, x_{m-1}$ with the same color. Then by (ii) $\{x_0, x_1, \ldots, x_{m-1}, x_m\}$ again a $P_m$ in color $c$. 

(iii) Edges $x_0$ and $x_2$ have the same color for $x_0, x_2 \in V(G)$ when $x < x_2$. Where the ordering corresponds to a partial order with $x_0$ as minimum element.

How $T$ is found

A partition of $V(G)$. 
Theorem 2.

(i) \( n_{\text{loc}}^2(K_n) = n_2(K_n) \) for all \( n \).

(ii) \( n_{\text{loc}}^2(K_{m,n} + K_m) = n_2(K_{m,n} + K_m) \),

\((m,m) \neq (3,3), m > 2m-1.\)

(iii) \( n_{\text{loc}}^2(K_m) = n_2(K_m) \) for all \( m \geq 3 \).

(iv) \( n_{\text{loc}}^2(P_{2m}) = n_2(P_{2m}) = 3m-1, m > 1.\)

(v) \( n_{\text{loc}}^2(P_{2m+1}) = n_2(P_{2m+1}) + 3m, m > 1.\)

(vi) \( n_{\text{loc}}^2(mK_3) = 7m-2, m > 2.\)

\( n_{\text{loc}}^2(mK_3) = 5m, m > 2.\)

(vii) For a connected graph \( G \),

\( n_{\text{loc}}^2(G) \geq 3\chi(G)/2 \) and there are

then \( T \) such that \( n_2(T) \leq \lceil \sqrt{3\chi(G)/2} \rceil.\)

Theorem 3.

(i) \( n_{\text{loc}}^3(S_m) = k(m-1) + 2, k,m \geq 1.\)

\( S_m \) is an \( m \)-star on \( m \) vertices.

(ii) \( n_{\text{loc}}^2(P_3) = \begin{cases} 2k+2 & \text{if } k \equiv 0,1 \pmod{3} \\ 2k+1 & \text{if } k \equiv 2 \pmod{3} \end{cases} \)

(iii) \( n_{\text{loc}}^3(K_3) = 17.\)

(iv) (Turszczynski-Tuza)

For a connected graph \( G \),

\( n_{\text{loc}}^k(G) / n_2(G) \leq C_k \) (\( C_k \)-constant depending on \( k \)).

(v) For \( m > 1, t > 1 \)

\( n_{\text{loc}}^3(mK_t) \geq m(t-1) - t + 1 \) and

\( n_{\text{loc}}^2(mK_t) \leq (t-1)m + C_t.\)

Nature of local \( k \)-coloring

Theorem 4. If \( G \) is a locally \( k \)-colored graph, then for some monochromatic subgraph \( G' \), the average degree \( \overline{d}^*(G) \)

\( \geq d^*(G) / k.\)

Corollary. If \( G \) is locally \( k \)-colored, then it contains a monochromatic subgraph of minimum degree \( \geq d^*(G) / k.\)

It is easy to see that if the edges of \( G \) are colored by at most \( k \) colors and \( \chi(G) = m^2 + 1 \), then \( G \) contains a monochromatic subgraph \( G' \) with \( \chi(G') \geq m^3 + 1.\)

This no longer holds for \( k \)-colorings.
Theorem 5. There exist graphs with arbitrary large chromatic number with local 2-colorings such that each monochromatic graph in bipartite.

Def. Let $K^n_m$ denote the complete $n$-uniform hypergraph. A local $k$-coloring of $K^n_m$ is a coloring of its edges such that the set of edges containing any $(n-1)$-clique subset of vertices are colored by at most $k$ different colors.

Theorem 6. (Existence - Ramsey Number for Hypoedge)

Let $k, n$ and $m$ be positive integers, $n \leq m$. Then there exists an $N = N(k, n, m)$ such that every local $k$-coloring of $K^n_m$ contains a monochromatic $K^n_m$.

As an application of Theorem 6, we prove the following theorem:

Theorem 7. For all bipartite graphs $G$ and for all $k$, there exists a bipartite graph $G'$ such that when $G'$ is locally $k$-colorable, then it contains a monochromatic copy of $G$ as an induced subgraph of $G'$.

Theorem 8. Let $G$ be a graph on $m$ vertices with $\lambda(G) \leq d$. Then for each $k$ there exists a function $c = c(k, d)$ such that $r_k^n(G) \leq c$.

A Generalization by M. Traczykowski

Def. Let $k$ be a fixed positive integer and let $H$ be a fixed graph with at least $k+1$ edges. We say a graph $G$ has been given a local $(H, k)$-coloring (or simply an $(H, k)$-coloring) if each subgraph of $G$ isomorphic to $H$ has its edges colored by at most $k$ different colors.

Note that a local $k$-coloring of $K^n_m$ is a $(K^n_m, n, k)$-coloring.

The $(H, k)$ local Ramsey number $R^{(H, k)}(G)$ is the smallest positive integer $m$ such that each local $(H, k)$-coloring of $K^n_m$ contains a monochromatic subgraph isomorphic to $G$.

Theorem 9. Let $H$ be a graph with at least $k+1$ edges. The Ramsey number $R^{(H, k)}(G)$ is well defined for every graph $G$ if and only if $H$ contains a forest with $k+1$ edges. In such a case, $R^{(H, k)}(G) \leq (2 + k^2)(2m+1)^{k-1}$.
Let $E(K_n)$ be colored such that at least three edges have different colors. Then $K_n$ contains a $K_4$, with three of its edges of different colors. Reason: $\sqrt[3]{n} \geq 3$.

Thus it follows that each $(K_{2k}, k)$-coloring of $K_n$ is a $k$-coloring. Hence for $k \geq 2$ and for each graph $G$, $\chi(K_{2k}, k)(G) = 2 \cdot \chi(G)$.

Theorem 10. Let $k \geq 1$ and $m = \lfloor 2\sqrt{k} \rfloor + 1$.

Then for each connected graph $G$, $\chi(K_{2k}, k)(G) = \chi(G)$.

Proof. We need show $\chi(K_{2k}, k)(G) \leq \chi(G)$. Suppose the not case. Let $m = \chi(G)$ and let $\phi$ be a $(K_{2k}, k)$-coloring of $K_n$ with no monochromatic $K_4$. If $c_1$ and $c_2$ are two colors of $\phi$ such that no pair of edges with these two colors are adjacent, then recolor $K_n$ changing each edge colored $c_2$ to color $c_1$. Clearly, the recolored $K_n$ is an $(K_{2k}, k)$-coloring with no monochromatic $G$ (since $G$ is connected).

Repeat this recoloring procedure until an $(K_{2k}, k)$-coloring $\psi$ of $K_n$ is obtained in which each pair of colors appear at least once as adjacent edges.

By assumption, $\psi$ uses at least $m = \chi(G)$ colors.

$\chi(G)$ colors. If $k+1$ is even, select any $k+1$ colors $c_1, c_2, \ldots, c_{k+1}$ used by $\psi$ and if $k+1$ is odd, select the three colors $a, b, c$ appearing as edges of some $K_4$ and select the remaining $k-2$ colors $c_1, c_2, \ldots, c_{k+1}$ arbitrarily.

Next choose vertices $x_i, y_i, z_i$ such that $c_i, c_j$ for $1 \leq i \leq \frac{m}{2} (k+1 \text{ even})$

Then the set of chosen vertices $X$ is such that $|X| \leq \frac{m}{2} + 1 = m$.

Hence $\psi$ colors $K_n$ with at least $k+1$ colors, a contradiction.

Question: What is the smallest value of $m > k+2$ such that for connected graphs $G$, $\chi(K_{2k+1}, k)(G) = \chi(G)$? More generally, what are the minimal graphs $G$ containing a forest on $k+1$ edges such that for every graph $G$, $\chi(K_{2k+1}, k)(G) = \chi(G)$?

Theorem 11. Let $F$ be a forest with $k+1$ edges. For every graph $G$, there exists a graph $H$ such that the maximum clique size of $H$ and $G$ are the same and every $(F, k)$-coloring of $H$ contains an induced monochromatic subgraph isomorphic to $G$. 
Special \((H,k)\) - colorings

Let \(H\) be a graph containing at least \(k+1\) edges. We are interested in those \((H,k)\)-colorings of \(K_n\) \((n \geq m)\) such that each \((H,k)\)-coloring is a \(k\)-coloring. It was observed earlier that \((K_k,k)\)-colorings were such colorings. Whenever each \((H,k)\)-coloring of \(K_n\) is a \(k\)-coloring we will call \(H\) a \(k\)-good graph.

**Problem.** Find necessary and sufficient conditions for a graph \(H\) to be \(k\)-good.

**Theorem 12.** If \(H\) is a \(k\)-good graph, then \(H\) contains each \(k\) edge graph as a subgraph.

*Proof.* Suppose not and assume \(H\) fails to contain some \(k\) edge graph as a subgraph. Consider a fixed copy of \(L\) contained in \(K_n\). Color each edge of this copy of \(L\) with a different color \((\text{colors } 1, 2, \ldots, k)\). Color all remaining edges of \(K_n\) with a \((k+1)\)-at-color. Clearly each copy of \(L\) in \(K_n\) is colored with at most \(k\) different colors \((\text{it must fail to contain some edge of } L)\). Thus \(K_n\) has been given an \((H,k)\)-coloring with \(k+1\) colors, a contradiction.

---

**Theorem 13.** The conjecture holds when

(i) \(k = 2, 3, \text{ and } 4\),

(ii) \(H\) is the vertex disjoint union of all connected graphs on \(k\) edges \((k > 3)\),

(iii) \(H = \bigwedge_{k} K_{k}\), i.e. \(H\) is obtained from \(K_k\) by attaching a pendent edge to each of its \(k\) vertices.

---

**Theorem 14.** The only edge minimal \(2\)-good graphs are \(P_4 = \quad \text{ and } P_2 \cup P_3 = \quad \text{—}

(ii) The only edge minimal \(3\)-good graphs are

\[ \bigtriangleup, \quad \bigcirc, \quad \bigtriangledown, \quad \bigDelta, \quad \bigtriangleup V \quad \text{—} \]

**Problem.** Which \(k\) edge graphs must \(H\) contain in order that under all \((H,k)\)-colorings of \(K_n\) at most a bounded number of edges of \(K_m\) are colored differently?
We show $H \geq K_{1, k} \cup K_2$.

**Reason:**

1. Color all edges of a fixed $K_{n-1}$ in $K_n$ differently, and all the remaining edges in $K_n$ with a fixed color. Let $H = \cup (V, E)$ be a disjoint union of all connected $k$-edge graphs $G_i$ (except for the star $K_{1, k}$). Clearly, this is an $(H, k)$-coloring of $K_n$ which uses $m-1$ colors.

2. Next color $\left[ \frac{m}{2} \right]$ independent edges of $K_n$ differently and the other edges of $K_n$ with a single color. In this case, let $H = K_{2n}$. Again, this is an $(H, k)$-coloring of $K_n$ and contains $\left[ \frac{m}{2} \right]$ colors. Thus, $H \geq K_{1, k} \cup K_2$.

**Theorem 15.** Let $H$ be a graph with at least $k+1$ edges such that $H \geq K_{1, k} \cup K_2$ (as subgraphs). If $\phi$ is an $(H, k)$-coloring of $K_n$, then $K_n$ is colored by at most $3k^2$ colors.

Note $k^2$ is the correct order of magnitude. This can be seen by coloring each edge of a fixed copy of $K_k$ in $K_n$ differently and the remaining edges of $K_n$ with a single color.

Let $H = K_{1, k} \cup K_2$ and observe that this is an $(H, k)$-coloring of $K_n$ with $(\frac{m}{2})+1$ colors.

---

**Questions**

1. **Local to Global Colorings**

   1. If $H$ contains all $k$-edge graphs as subgraphs, then is $H$ $k$-good?

   2. Can one prove questions raised above (2)-(3) for special families of graphs $H$ which contain all $k$-edge graphs?

   3. What happens to the bound $c = 2$ of Theorem 14 when we assume $H$ contains both $K_{1, k}$ and $k K_2$ and some of the other $k$-edge graphs as subgraphs? How many of the $k$-edge graphs must $H$ contain for the bound to be linear, i.e., for $c = k$?

---

**Ramsey Questions**

1. For which graphs $H$ does $\chi^+(H, k) = \chi^+(G)$ for all graphs $G$?

2. If $\chi^+(H, k, \ell) = \chi^+(G)$, $G$ connected, but $\chi^+(K_{n-1, k} \cup k K_2) > \chi^+(G)$, how large is this difference?

3. The size Ramsey number \( \overline{\chi} \) is defined as

\[
\overline{\chi}(G) = \min \{|E(H)| : H \rightarrow (G, C) \text{ minimally} \}
\]

Investigate the local size Ramsey number \( \overline{\chi}_{\text{loc}}(G) \) and more generally \( \overline{\chi}^+(H, G) \).
The Local Ramsey Number
And
Local Colorings
by
R. H. Schelp
REFERENCES


3) M. Truszczyński, Generalized Local Colorings of Graphs, to appear in JCTB.

4) R.A. Clapsaddle and R.H. Schelp, Local Edge Colorings that are Global, in preparation.
Def. A local $k$-coloring of a graph $H$ is a coloring of its edges in such a way that the edges incident to each vertex of $H$ are colored by at most $k$ different colors. The local Ramsey number $R_{loc}^k(G)$ of a graph $G$ is the smallest positive integer $m$ such that $K_m$ contains a monochromatic copy of $G$ under each local $k$-coloring.
Observations:

1) For $k$ fixed and $m$ large a local $k$-coloring of $K_m$ can use a large number of colors independent of $k$.

$$K_m = \begin{array}{cccc}
    & & c_1 \\
  s_1 & s_2 & \ldots & s_m
\end{array}$$

Here all edges of $K_m$ are colored with color $c_1$ except for the shown $\left\lfloor \frac{m}{5} \right\rfloor$ matching edges.

2) If $r^k(G)$ denotes the usual Ramsey number, then $r^k(G) \leq r^{k}_{loc}(G)$ since each coloring of a $K_m$ by at most $k$ colors is a local $k$-coloring.
Theorem 1 (Existence) \[ k \geq 2 \ \text{and} \ \Delta_{\text{loc}}(G) \leq \left\lceil \frac{k^k(n-2)+1}{k-1} \right\rceil \]

Proof. Let $G$ be a local $k$-colored complete graph with $\geq \left\lceil \frac{k^k(n-2)+1}{k-1} \right\rceil$ vertices. We show $G$ contains a spanning tree $T$, rooted at a fixed vertex $x_0$, such that the following holds.

(1) Each $x \in V(T)$ has at most $k$ successors $x_1, x_2, \ldots, x_s$ ($s \leq k$) with different colors assigned to each edge $xx_i$, $1 \leq i \leq s$. 
(ii) Edges $x,y$ and $x,z$ have the same color for $x,y,z \in V(T)$ when $x < y < z$, where the ordering corresponds to a partial order with $x_0$ as minimum element.

**How I is found**

A partition of $V(G)$.

\[ \xymatrix{ & & X_{11} \ar[lddd]^{	ext{Color } C_{11}} \ar[dd]^{	ext{Color } C_{12}} & \ar[d]^{	ext{Color } C_{1k}} & & \\
X_0 & & & & X_{12} & \ldots \ar[uuuu]^{X_{1k}} } \]
Select a fixed vertex $x_{1i}$ in each $X_{1i}$ and partition $X_{1i}$ as was just done for $V(G)$ replacing $x_0$ by $x_{1i}$.

Continue this process until a spanning tree is obtained.
By condition (i) there exists a path $P$ from the root $x_0$ to a vertex $y$ with at least $k(n-2)+1$ edges.

$P: x_0 \rightarrow x_1 \rightarrow x_2 \rightarrow \cdots \rightarrow x_{k(n-2)} \rightarrow x_{k(n-2)+1} \rightarrow \cdots \rightarrow y$

By condition (ii) $x_2$ is incident to edges colored $c_{i_1}, c_{i_2}, \ldots, c_{i_k}$. Therefore the edges of $P$ get at most $k$ colors.

Hence there exist $n-1$ edges, say $x_1 y_1, x_2 y_2, \ldots, x_{n-1} y_{n-1}$, with the same color. $x_1 y_1 \xrightarrow{c} x_2 y_2 \xrightarrow{c} \cdots \xrightarrow{c} x_{n-1} y_{n-1}$

Then by (ii) $\{x_1, x_2, \ldots, x_{n-1}, y_{n-1}\}$ spans $\mathcal{H}_n$ in color $c$. 
Theorem 2

(i) \( n_{\text{loc}}^2(K_n) = n^2(K_n) \) for all \( n \).

(ii) \( n_{\text{loc}}^2(K_{m-m} + K_m) = n^2(K_{m-m} + K_m) \)

\((m, m) \neq (3, 2), m \geq 3.

(iii) \( n_{\text{loc}}^2(C_n) = n^2(C_n) \) for all \( n \geq 3 \).

(iv) \( n_{\text{loc}}^2(P_{2m}) = n^2(P_{2m}) = 3m-1, m \geq 1 \)

(v) \( n_{\text{loc}}^2(P_{2m+1}) = n^2(P_{2m+1}) + 1 = 3m+1, m \geq 1 \)

(vi) \( n_{\text{loc}}^2(mK_3) = 7m-2, m \geq 2, \)

\( n_{\text{loc}}^2(mK_3) = 5m, m \geq 2, \)

(vii) For a connected graph \( G \)

\( n_{\text{loc}}^2(G) \geq 3\sqrt{V(G)} \), and there are trees \( T \) such that \( n^2(T) \leq \left\lfloor \frac{2}{3} \sqrt{V(T)} + 1 \right\rfloor \)
Theorem 3.

(i) \( \nabla^{k}_{\text{loc}}(S_n) = k(n-1)+2 \), \( k, n \geq 1 \), where \( S_n \) is star on \( n \) edges.

(ii) \( \nabla^{k}_{\text{loc}}(P_4) = \begin{cases} 
2k+2 & \text{if } k \equiv 0, 1 \pmod{3} \\
2k+1 & \text{if } k \equiv 2 \pmod{3} 
\end{cases} \)

(iii) \( \nabla^{3}_{\text{loc}}(K_3) = 17 \)

(iv) (Truszczynski-Tuza)

For \( G \) a connected graph

\[ \nabla^{k}_{\text{loc}}(G)/\nabla^{k}(G) \leq c_k \quad (c_k \text{-constant depending on } k) \]

(v) For \( m \geq 1, t \geq 2 \)

\[ \nabla^{2}_{\text{loc}}(mK_t) \geq m(t^2-t+1)-t+1 \] and for \( m \) large WRT \( t \)

\[ \nabla^{2}(mK_t) \leq (2t-1)m + c_t \]
Reason local 2-coloring problem is easier than for arbitrary k: Any local 2-coloring of $K_n$ looks like

All edges in $A_{ij}$ colored with either color $i$ or color $j$.

```
A_{12} 1 A_{13} 1 ... 1 A_{nn}
```
Nature of local \( k \)-colorings

**Theorem 4.** If \( G \) is a locally \( k \)-colored graph, then for some monochromatic subgraph \( G_i \), the average degree \( d^*(G_i) \geq d^*(G)/k \).

**Corollary.** If \( G \) is locally \( k \)-colored, then it contains a monochromatic subgraph of minimum degree \( \geq d^*(G)/2k \).

It is easy to see that if the edges of \( G \) are colored by at most \( k \) colors and \( \chi(G) \geq m^{k+1} \), then \( G \) contains a monochromatic subgraph \( G' \) with \( \chi(G') \geq m+1 \).

This no longer holds for \( k \)-colorings.
Theorem 5. There exist graphs with arbitrary large chromatic number with local 2-colorings such that each monochromatic graph is bipartite.

Def. Let $K^r_m$ denote the complete $r$-uniform hypergraph. A local $k$-coloring of $K^r_m$ is a coloring of its edges such that the set of edges containing any $(r-1)$-element subset of vertices are colored by at most $k$ different colors.
Theorem 6. (Existence - Ramsey Number for Hypergraphs)

Let $k$, $r$, and $n$ be positive integers, $n \leq m$. Then there exists an $N = N(k, r, m)$ such that every local $k$-coloring of $K^m_n$ contains a monochromatic $K^r_k$.

As an application of Theorem 6 we can prove the following theorem.

Theorem 7. For all bipartite graphs $B$ and for all $k$ there exists a bipartite graph $B'$ such that when $B'$ is locally $k$-colored, then it contains a monochromatic copy of $B$ as an induced subgraph of $B'$.

Theorem 8. Let $G$ be a graph on $n$ vertices with $\Delta(G) \leq d$. Then for each $k$ there exists a function $c = c(k, d)$ such that $n^{k/2} \leq c(n)$. 
A Generalization by M. Truszczynski

Def. Let \( k \) be a fixed positive integer and let \( H \) be a fixed graph with at least \( k+1 \) edges. We say a graph \( G \) has been given a local \((H,k)\)-coloring (or simply an \((H,k)\)-coloring) if each subgraph of \( G \) isomorphic to \( H \) has its edges colored by at most \( k \) different colors.

Note that a local \( k \)-coloring of \( K_n \) is a \((K_1,m-1,k)\)-coloring.
The \((H,k)\) local Ramsey number \(r^{(H,k)}(G)\) is the smallest positive integer \(m\) such that each local \((H,k)\)-coloring of \(K_m^m\) contains a monochromatic subgraph isomorphic to \(G\).

**Theorem 9.** Let \(H\) be a graph with at least \(2k+1\) edges. The Ramsey number \(r^{(H,k)}(G)\) is well defined for every graph \(G\) if and only if \(H\) contains a forest with \(2k+1\) edges. In such a case \(r^{(H,k)}(G) \leq (2+6k^2)(2k+1)^2 r^{k}_{loc}(G)\).
Let $E(K_n)$ be colored such that at least three edges have different colors. Then $K_n$ contains a $K_4$ with three of its edges of different colors. Reason: $1/\sqrt{2} < \sqrt{3}$.

Thus it follows that each $(K_{2k}, k)$-coloring of $K_n$ is a $k$-coloring.

Hence for $k \geq 2$ and for each graph $G$

$\chi((K_{2k}, k))(G) = \chi^k(G)$.

**Theorem 10.** Let $k \geq 1$ and $m = \lceil 3k^2 \rceil + 1$.

Then for each connected graph $G$

$\chi((K_m, k))(G) = \chi^k(G)$.

Proof. We need show $\chi((K_m, k))(G) \leq \chi^k(G)$.

Suppose this not case. Set
$m = n^k(G)$ and let $\phi$ be a $(K_m, k)$-coloring of $K_m$ with no mono. $G$. If $c_1$ and $c_2$ are two colors of $\phi$ such that no pair of edges with these two colors are adjacent, then recolor $K_m$ changing each edge colored $c_2$ to color $c_1$. Clearly the recolored $K_m$ is an $(K_m, k)$-coloring with no mono. $G$ (since $G$ is connected).

Repeat this recoloring procedure until an $(K_m, k)$-coloring $\psi$ of $K_m$ is obtained in which each pair of colors appear at least once as adjacent edges.

By assumption $\psi$ uses at least
$k+1$ colors. If $k+1$ is even select any $k+1$ colors $c_1, c_2, \ldots, c_{k+1}$ used by $U$.

And if $k+1$ is odd select three colors $a, b, c$ appearing on edges of some $K_4$ and select the remaining $k-2$ colors $c_1, c_2, \ldots, c_{k-2}$ arbitrarily.

Next choose vertices $x_i, y_i, z_i$ such that

$$
\begin{align*}
  c_{2i-1} & \sim c_{2i} \\
  x_i & \sim z_i
\end{align*}
$$

for $1 \leq i \leq \frac{k+1}{2}$ ($k+1$ even),

$$
1 \leq i \leq \frac{k-2}{2}$ ($k+1$ odd)

Then the set of chosen vertices $X$ is such that $|X| \leq \lceil \frac{3k}{2} \rceil + 1 = m$.

Hence $U$ colors $K_X$ with at least $k+1$ colors, a contradiction.
Question: What is the smallest value of \( m \geq k+2 \) such that for connected graph \( G \), \( n^{(k,m,k)}(G) = n^k(G) \)? More generally, what are the minimal graphs \( H \) containing a forest on \( k+1 \) edges such that for every graph \( G \), \( n^{(H,k)}(G) = n^k(G) \)?

Theorem 11: Let \( F \) be a forest with \( k+1 \) edges. For every graph \( G \), there exists a graph \( H \) such that the maximum clique size of \( H \) and \( G \) are the same and every \((F,k)\)-coloring of \( H \) contains an induced monochromatic subgraph isomorphic to \( G \).
Special \((H, k)\) - colorings

Let \(H\) be a graph containing at least \(k+1\) edges. We are interested in those \((H, k)\) - colorings of \(K_n\) \((n > n_0)\) such that each \((H, k)\) - coloring is a \(k\) - coloring. It was observed earlier that \((K_{2k}, k)\) - colorings were such colorings. Whenever each \((H, k)\) - coloring of \(K_n\) is a \(k\) - coloring we will call \(H\) a \(k\) - good graph.

Problem. Find necessary and sufficient conditions for a graph \(H\) to be \(k\) - good.
Theorem 12. If $H$ is a $k$-good graph, then $H$ contains each $k$ edge-graph as a subgraph.

Proof. Suppose not and assume $H$ fails to contain some $k$ edge-graph $K'$ as a subgraph. Consider a fixed copy of $K'$ contained in $K_n$. Color each edge of this copy of $K'$ with a different color (colors $1, 2, \ldots, k$). Color all remaining edges of $K_n$ with a $(k+1)$st color. Clearly each copy of $H$ in $K_n$ is colored with at most $k$ different colors (it must fail to contain some edge of $K'$). Thus $K_n$ has been given an $(H, k)$-coloring with $k+1$ colors, a contradiction.
Conjecture. The graph $H$ containing at least $k+1$ edges is $k$-good if and only if $H$ contains each $k$ edge graph as a subgraph.

**Theorem 13.** The conjecture holds when

(i) $k = 2, 3, \text{ and } 4$, 

(ii) $H$ is the vertex disjoint union of all connected graphs on $k$ edges ($k > 3$). 

(iii) $H = \bigcup_{i=1}^{k} K_{k-1}$, i.e., $H$ is obtained from $K_{k-1}$ by attaching a pendant edge to each of its $k$ vertices.
Theorem 14. The only edge minimal
\( (i) \) 2-good graphs are \( P_4 = \quad \)
and \( P_3 \cup P_2 = \quad \).

\( (ii) \) The only edge minimal 3-good graphs
are \( \Delta, \Delta \quad \),
\( \Delta \quad \), \( \Delta \quad \).

Problem. Which \( k \) edge graphs must \( H \)
contain in order that under all
\( \{H, k\} \)-colorings of \( K_n \) at most a
bounded number of edges of \( K_n \)
are colored differently?
We show \( H \cong K_1, k, k K_2 \)

Reason:

1. Color all edges of a fixed \( K_1, m-1 \) in \( K_m \) differently and all the remaining edges in \( K_m \) with a fixed color. Let \( H = \cup (H, k) \) (vertex disjoint) of all connected \( k \) edge graphs (except for the star \( K_{1,k} \)). Clearly this an \((H,k)\)-coloring of \( K_m \) which uses \( m-1 \) colors.

2. Next color \( \lceil \frac{m}{2} \rceil \) independent edges of \( K_m \) differently and the other edges of \( K_m \) with a single color. In this case let \( H = K_{2k-1} \). Again this is an \((H,k)\)-coloring of \( K_m \) and contains \( \lceil \frac{m}{2} \rceil \) colors.

\( \therefore H \cong K_1, k, k K_2 \)
Theorem 15. Let $H$ be a graph with at least $k+1$ edges such that $H \supseteq K_{1,k} \cup kK_2$ (as subgraphs). If $\phi$ is an $(H,k)$-coloring of $K_n$, then $K_n$ is colored by at most $3k^2$ colors.

Note $k^2$ is the correct order of magnitude. This seen by coloring each edge of a fixed copy of $K_k$ in $K_n$ differently and the remaining edges of $K_n$ with a single color.

Let $H = K_{1,k} \cup kK_2$ and observe that this is an $(H,k)$-coloring of $K_n$ with $(\frac{k^2}{2}) + 1$ colors.
Questions

Local to Global Coloring

1) If $H$ contains all $k$-edge graphs as subgraphs, then is $H$ $k$-good?

2) Can one prove question raised above (in (1)) for special families of graphs $H$ which contain all $k$-edge graphs?

3) What happens to the bound $c_k^2$ of Theorem 14 when we assume $H$ contains both $K_{1,k}$ and $kK_2$ and some of the other $k$-edge graphs as subgraphs? How many of the $k$-edge graphs must $H$ contain for the bound to be linear in $k$?
Ramsey Questions

1) For which graphs $H$ does
   \[ n^{(H,k)}(G) = n^k(G) \text{ for all graphs } G ? \]

2) If $n^{(K_m,k)}(G) = n^k(G)$, $G$ connected,
   but $n^{(K_{m-1},k)}(G) > n^k(G)$, how large is this difference?

3) The size Ramsey number is defined as
   \[ \hat{r}(G) = \min \{ |E(H)| : H \to (G,G) \text{ minimally} \} \]
   Investigate the local size Ramsey number $\hat{r}^{loc}(G)$ and more generally $\hat{r}^{(H,k)}(G)$. 
New algorithms for minimizing convex functions over convex sets

Prof. Pravin Vaidya
Department of Computer Science
University of Illinois
New algorithms for minimizing convex functions over convex sets

Pravin M. Vaidya
Dept. of Computer Science,
Univ. of Illinois at-Urbana-Champaign

* Part of this work was done when the author was at AT&T Bell Laboratories, Murray Hill, NJ
Let $S \subseteq \mathbb{R}^n$ such that there is an oracle for $S$ with the following property.

Let $z \in \mathbb{R}^n$ be a test point:

i) Oracle answers "Yes" if $z \in S$.

ii) Oracle returns a vector $c$ such that $S = \{x : c^T x \geq c^T z\}$ if $z \notin S$.

Feasibility Problem
Find a point in $S$

Optimization Problem
Minimize a convex function over $S$. 
Applications

1. Econometric, statistical modelling
2. Structural Optimization
3. Relaxations of NP-hard problems
4. Certain non-linear PDE's
5. VLSI Design
6. Combinatorial Optimization
Feasibility Problem

i) Maintain a region $R$ s.t. $S \subseteq R$.

ii) At each step select a test point $z \in R$ & call the oracle with $z$ as input.

$z \notin S \implies S \subseteq R \cap \{x : c^T x \geq c^T z\}$
Feasibility Problem

i) Maintain a region $R$ s.t. $S \subseteq R$
ii) At each step select a test point $z \in R$ & call the oracle with $z$ as input.

$z \notin S \implies S \subseteq R \cap \{x : c^T x \geq c^T z\}$
Ellipsoid algorithm

(i) Region $R$ is an ellipsoid

(ii) $Z$: center of ellipsoid $R$

Description of $R$ is simplified at each step by redrawing an ellipsoid of smaller volume around half ellipsoid $E_H$
A class of algorithms based on polytopes

(i) Region $R$ is a bounded full dimensional polytope

$$P = \{ x : Ax \geq b \}$$

$A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$

(ii) $Z$ is a suitable "center" (or a balanced point) in the polytope $P$
Possible choices for centers

(i) **Analytic center:**

Logarithmic barrier \( \phi(x) \)

\[
\phi(x) = - \sum_{i=1}^{m} \ln (a_i^T x - b_i)
\]

Analytic center is the minimizer of \( \phi(x) \) over \( P \).

(ii) **Volumetric center:**

Determinant Barrier \( F(x) \)

\[
F(x) = \frac{1}{2} \ln \left( \det \left( \nabla^2 \phi(x) \right) \right)
\]
(iii) Center that maximizes the volume of an ellipsoid inscribable in the polytope

(iv) Weighted analytic center

\[ \log \bar{x}(W, x) = - \frac{1}{\sum_{i=1}^{M} w_i} \prod_{i=1}^{M} w_i \ln (a_i x - b_i) \]

This center minimizes \( \log \bar{x}(W, x) \)

(v) Center of gravity
Algorithm for the feasibility problem

1) The region \( R \) is a polytope \( P \),
\[
P = \{ x : A x \geq b \}
\]
P - full dimensional, bounded

2) Test point \( z \) is the volumetric center of \( P \)

[Diagram of a polytope with a volumetric center and cutting planes]
\[ \text{Volumetric center of } P \]
\[ P = \{ x : Ax \geq b \} , \quad A \in \mathbb{R}^{m \times n}, \quad b \in \mathbb{R}^m \]
\[ \log \text{barrier: } \phi(x) = -\sum_{i=1}^{m} \ln (a_i^T x - b_i) \]
\[ H(x) = \sum_{i=1}^{m} \frac{a_ia_i^T}{(a_i^T x - b_i)^2} \]

Volumetric center minimizes \( \det(H(x)) \) over \( P \).
Geometric Interpretation of the volumetric center $\omega$

$E(x) = \{ y : (y-x)^TH(x)(y-x) \leq 1 \}$

(i) $E(x) \subseteq P$

(ii) $E(\omega)$ has the largest volume among all ellipsoids $E(x)$

$E(\omega)$ is a maximum volume quadratic approximation to polytope $P$. 
Computing the volumetric center

A step (Newton's Method)

Current point \( z \)

\[
z \leftarrow z - t \, G(z)^T \nabla F(z)
\]

\( t \) a suitable scalar

1) \( z \in \Sigma \): \( F(z) - F(w) \) decreases by a constant factor

2) \( z \notin \Sigma \): \( F(z) \) decreases by about \( \frac{1}{\sqrt{m}} \)
Finding the volumetric center \( \omega \)

\[
H(x) = \sum_{i=1}^{m} \frac{a_i a_i^T}{(a_i^T x - b_i)^2}
\]

\[
F(x) = \frac{1}{2} \ln \left( \det \left( H(x) \right) \right)
\]

\[
\delta_i(x) = \frac{a_i^T H(x)^{-1} a_i}{(a_i^T x - b_i)^2}, \quad 1 \leq i \leq m
\]

\[
\nabla F(x) = -\sum_{i=1}^{m} \delta_i(x) \frac{a_i}{a_i^T x - b_i}
\]

\[
Q(x) = \sum_{i=1}^{m} \delta_i(x) \frac{a_i a_i^T}{(a_i^T x - b_i)^2}
\]

\( Q(x) \) approximates the Hessian \( \nabla^2 F(x) \)
Interpretation of weights $\xi_i(x)$

\[
E(x) = \{ y : (y-x)^T H(x) (y-x) \leq 1 \}
\]

$z$ minimizes $\alpha_i^T x$ over $E(x)$

$\xi_i(x) = \left( \frac{d_1}{d_2} \right)^2$
Pruning the Polytope $P$

$P = \{ x : A x \geq b \}$

$A \in \mathbb{R}^{m \times n}$; $m$ constraining planes

As $m$ increases, the "centres" get unbalanced, convergence can slow down & computational work/step increases.

Polytope $P$ may be pruned i.e. some of the planes defining $P$ are dropped

$6i(x)$ small $\Rightarrow$ $i^{th}$ constraint $a_i^T x = b_i$ may be dropped
Cutting the polytope near the volumetric center

\[ \tilde{P} = P \cap \{ x : c^T x \geq \beta \} \]

\[ \frac{c^T H(\omega)^{-1} c}{(c^T \omega - \beta)^2} = \frac{\alpha}{\sqrt{m}} \]

\[ F(\tilde{\omega}) - F(\omega) \sim \frac{\alpha}{2 \sqrt{m}} \]
Algorithm with best complexity

1) Maintain a polytope P such that S ⊆ P.
2) Use a good approximation to volumetric center as the test point.
3) Also prune the polytope P i.e. drop some of the planes from time to time so m = O(n).

F(w) increases by a fixed constant S at each step & after K steps

\[
\text{volume}(P) \leq \left(\frac{n}{S}\right)^n e^{-KS}.
\]
Variants of the algorithm

Desirable properties

(a) Computation at a step as simple as possible
   Preferably a single linear system solve

(b) Exploit underlying structure of constraints defining $S$
   E.g., constraints defining $S$ may be explicitly given & each constraint depends only on a few variables.

(c) Polynomial convergence still maintained in the worst case
Possible directions for variants

1) Interpreting the volumetric center as a weighted analytic center and dynamically weighting the planes

2) Combination of determinant barrier & logarithmic barrier

3) Combination of determinant barriers

4) Several mildly non-linear functions together with a few highly non-linear functions
Several mildly non-linear fun. together with a few highly non-linear ones

\[ \max p^T x \]

s.t. \( g_i(x) \geq 0, 1 \leq i \leq m \)

Most of \( g_i \)'s are only mildly non-linear, \( g_i \)'s are concave:

\[ \phi(\beta, x) = m \ln (p^T x - \beta) + \sum_{i=1}^{m} \ln (g_i(x)) \]

Related centering problem

Compute maximizer of \( \phi(\beta, x) \)

"Lazy use of separating tangent planes"
Centering problem

maximize $\phi(\beta, x)$ where

$$\phi(\beta, x) = m \ln(p^T x - \beta) + \sum_{i=1}^{M} \ln(g_i(x))$$

Alternate between Newton’s method & a method that is based on separating (tangent) planes; the subroutine based on separating planes is called only when Newton’s method fails to make progress in a consecutive number of steps.
Applications to linear programming

1) The basic algorithm or any suitable variant can solve a linear program with exponentially many constraints as long as there is a good subroutine to generate violated constraints.

   Examples - LP relaxations of TSP & maximum independent set.
   Weighted matching

2) Possible dynamic weighting of planes in ordinary linear programming.
New algorithms for minimizing convex functions over convex sets

Pravin M. Vaidya
Dept. of Computer Science,
Univ. of Illinois at Urbana-Champaign

+ Part of this work was done when the author was at AT&T Bell Laboratories, Murray Hill, NJ

Applications
1. Econometric, statistical modelling
2. Structural Optimization
3. Relaxations of NP-hard problems
4. Certain non-linear PDE's
5. VLSI Design
6. Combinatorial Optimization

Let \( S \subseteq \mathbb{R}^n \) such that there is an oracle for \( S \) with the following property:

Let \( z \in \mathbb{R}^n \) be a test point.

i) Oracle answers "yes" if \( z \in S \).

ii) Oracle returns a vector \( c \) such that \( S = \{ x : c^T x \geq c^T z \} \) if \( z \notin S \).

Feasibility Problem
Find a point in \( S \)

Optimization Problem
Minimize a convex function over \( S \).

Feasibility Problem

i) Maintain a region \( R \) s.t. \( S \subseteq R \).

ii) At each step select a test point \( z \in R \) & call the oracle with \( z \) as input.

\[ z \notin S \Rightarrow S = R \cap \{ x : c^T x \geq c^T z \} \]
Feasibility Problem

(i) Maintain a region \( R \) s.t. \( S \subseteq R \)
(ii) At each step, select a test point \( z \in R \) & call the oracle with \( z \) as input.

\[
z \in S \implies S \subseteq R \cap \{ x : c^T x = c^T z \}
\]

Ellipsoid algorithm

(i) Region \( R \) is an ellipsoid
(ii) \( z \) : center of ellipsoid \( R \)

Description of \( R \) is simplified at each step by redrawing an ellipsoid of smaller volume around half ellipsoid \( E_H \)

A class of algorithms based on polytopes

(i) Region \( R \) is a bounded full dimensional polytope

\[
P = \{ x : A x \geq b \}
\]

\( A \in \mathbb{R}^{m \times n} \) \( b \in \mathbb{R}^m \)

(ii) \( z \) is a suitable "center" (or a balanced point) in the polytope \( P \)

Possible choices for centers

(i) Analytic center:

Logarithmic barrier function \( \phi(x) \)

\[
\phi(x) = -\sum_{i=1}^{m} \ln(a_i^T x - b_i)
\]

Analytic center is the minimizer of \( \phi(x) \) over \( P \).

(ii) Volumetric center:

Determinant Barrier \( F(x) \)

\[
F(x) = \frac{1}{2} \ln(\det(\nabla^2 \phi(x)))
\]
(iii) Center that maximizes the volume of an ellipsoid inscribable in the polytope.

(iv) Weighted analytic center

\[ \log \text{bar}(W;x) = - \sum_{i=1}^{m} W_i \ln(a_i^Tx - b_i) \]

This center minimizes \( \log \text{bar}(W;x) \).

(v) Center of gravity

Algorithm for the feasibility problem:

1) The region \( R \) is a polytope \( P \):

\[ P = \{ x : A x \geq b \} \]

\( P \) - full dimensional, bounded.

2) Test point \( z \) is the

volumetric center of \( P \).

Geometric Interpretation of the Volumetric Center \( \omega \):

Volumetric center of \( P \):

\[ P = \{ x : A x \geq b \} , A \in \mathbb{R}^{m \times n} , b \in \mathbb{R}^m \]

\[ \log \text{bar}z(x) = - \sum_{i=1}^{m} \ln(a_i^Tx - b_i) \]

\[ H(x) = \frac{\sum_{i=1}^{m} a_i^T a_i}{\sum_{i=1}^{m} (a_i^Tx - b_i)^2} \]

Volumetric center minimizes \( \det(H(x)) \) over \( P \).

\( \rightarrow \) determinant of \( H(x) \).

\[ E(x) = \{ y : (y-x)^T \text{H}(x)(y-x) \leq 1 \} \]

(i) \( E(x) \subseteq P \)

(ii) \( E(\omega) \) has the largest volume among all ellipsoids \( E(x) \).

\( E(\omega) \) is a maximum volume quadratic approximation to polytope \( P \).
Computing the volumetric center

A step (Newton's Method)

Current point \( z \)

\( z \leftarrow z - t \cdot \nabla F(z) \)

t a suitable scalar

1) \( z \in \Sigma \): \( F(z) - F(w) \) decreases by a constant factor

2) \( z \notin \Sigma \): \( F(z) \) decreases by about \( \frac{1}{\sqrt{m}} \)

Finding the volumetric center \( w \)

\[
H(x) = \sum_{i=1}^{m} \frac{a_i a_i^T}{(a_i^T x - b_i)^2}
\]

\[
F(x) = \frac{1}{2} \ln \left( \det (H(x)) \right)
\]

\[
\delta_i(x) = \frac{a_i^T H(x) a_i}{(a_i^T x - b_i)^2}, \quad 1 \leq i \leq m
\]

\[
\nabla F(x) = -\sum_{i=1}^{m} \delta_i(x) \frac{a_i}{a_i^T x - b_i}
\]

\[
Q(x) = \sum_{i=1}^{m} \delta_i(x) \frac{a_i a_i^T}{(a_i^T x - b_i)^2}
\]

\( Q(x) \) approximates the Hessian \( \nabla^2 F(x) \)

Interpretation of weights \( \delta_i(x) \)

\[
E(x) = \{ y : (y-x)^T H(x) (y-x) \leq 1 \}
\]

\( z \) minimizes \( a_i^T x \) over \( E(x) \)

\[
\delta_i(x) = \left( \frac{d_1}{d_2} \right)^2
\]

Pruning the Polytope \( P \)

\[
P = \{ x : A x \geq b \}
\]

\( A \in \mathbb{R}^{m \times n} \), \( m \) constraining planes

As \( m \) increases, the "centers" get unbalanced, convergence can slow down & computational work/step increases.

Polytope \( P \) may be pruned
i.e. some of the planes defining \( P \) are dropped

\( \delta_i(x) \) small \( \Rightarrow \) \( i^{th} \) constraint \( a_i^T x = b_i \) may be dropped
Algorithm with best complexity:

1) Maintain a polytope \( P \) such that \( S \subseteq P \).
2) Use a good approximation to volumetric center as the test point.
3) Also prune the polytope \( P \) i.e., drop some of the planes from time to time so \( m = O(n) \).

\[ F(w) \text{ increases by a fixed constant } S \text{ at each step } \& \text{ after } k \text{ steps} \]

\[ \text{volume}(P) \leq \left( \frac{n}{S} \right)^{m} e^{-kS} \]

Possible directions for variants:

4) Interpreting the volumetric center as a weighted analytic center and dynamically weighting the planes.

3) Combination of determinant barriers and logarithmic barriers.

2) Combination of determinant barriers.

1) Several mildly non-linear functions together with a few highly non-linear functions.

Variant of the algorithm

Desirable properties

4) Computation at a step as simple as possible. Preferably a single linear system solve.

5) Exploit underlying structure of constraints defining \( S \). E.g., constraints defining \( S \) may be explicitly given & each constraint depends only on a few variables.

6) Polynomial convergence still maintained in the worst case.
Several mildly non-linear \( n \) together with a few highly non-linear ones.

\[
\text{max} \quad p^T x \\
\text{st.} \quad g_i(x) \geq 0, \ 1 \leq i \leq m
\]

Most of \( g_i \)'s are only mildly non-linear, \( g_i \)'s are concave.

\[
\phi(p, x) = m \ln(p^T x - \beta) + \sum_{i=1}^{m} \ln(g_i(x))
\]

Related centering problem

Compute maximizer of \( \phi(p, x) \)

"Lazy use of separating tangent planes"

---

Centering Problem

maximize \( \phi(p, x) \) where

\[
\phi(p, x) = m \ln(p^T x - \beta) + \sum_{i=1}^{m} \ln(g_i(x))
\]

Alternate between Newton's method & a method that is based on separating (tangent) planes; the subroutine based on separating planes is called only when Newton's method fails to make progress in a consecutive number of steps.

---

Applications to linear programming

1) The basic algorithm or any suitable variant can solve a linear program with exponentially many constraints as long as there is a good subroutine to generate violated constraints.

Examples - LP relaxations of TSP & maximum independent set, weighted matching

2) Possible dynamic weighting of planes in ordinary linear programming.
Size of an $s$-intersection family in a semilattice and construction of vector space designs by quadratic forms

Prof. Dijen K. Ray-Chaudhuri
Department of Mathematics
Ohio State University
Size of an $s$-intersection family in a polynomial semilattice and construction of vector-space designs by quadratic forms.

by

D.K. RAY-CHAUDHURI
Ohio State University.

1975 R.M. Wilson and D.K. R.-C proved the following: $\forall x, k, \lambda, \lambda \geq k+3$

$|X| = \lambda \quad P_k(X) = \{A : \lambda \leq \lambda, |A| = \lambda^2\}$

let $\Omega \subseteq P_k(X)$ such that $|\{A \cap B : A \cap B < \langle\rangle_3\}| = 2$

Then $|\Omega| \leq \sqrt{3}$

T. Zhu and D.K.R.-C generalized this result to polynomial semilattices

Defn. Let $(X, \leq)$ be a partially ordered set. Semi lattice iff for all $x, y \in X$, any exists. Assume that the Poset has a length function $L$.

Let $X_i = \{x : L(x) = i\}$, $X_0 = \{0\}$, $X = \bigcup_{i=0}^{n} X_i$

Polynomial semilattice iff there exist
Integers \( m_0, m_1, \ldots, m_n \) and polynomials \( f_0, f_1, \ldots, f_n \) satisfying

(a) \( m_0 < m_1 < \cdots < m_n \)

(b) \( \deg f_i = i \) and for \( i < j \), \( f_i \mid f_j \)

(c) for all \( i, j, k = 0, 1, \ldots, n \), \( k \leq i, k \leq j \)

\[
|\{ z : x \leq z \leq y \} | = f_{i-k}(c_i)
\]

\[
\text{Example 1. Lattice of subsets, } M = V, X = P(V)
\]

\[
X_i = \{ A : A \subseteq V, M = c_f, c = 0, 1, \ldots, n \}
\]

2. Lattice of subspaces, \( V \) a vector space over a finite field of order \( q \)

\[
X_i = \text{subspaces of dim } i
\]

\[
m_i = q^i - 1, \quad f_i(z) = \frac{(x - q^i)(x - q^{2i}) \cdots (x - q^{(i-1)i})}{(q^i - 1)(q^{i-1} - 1) \cdots (q - 1)}
\]
3. Hamming Scheme (Orthogonal Array)
\[ \{W\mid \text{w a positive integer} \}
\]
\[ X_i = \{ (L, f) : L \subseteq \{1, 2, \ldots, n\}, f : L \rightarrow W, \text{ |L| = i} \} \]
\[ i = 0, 1, \ldots, n \]
\[ X = \bigcup X_i \quad (L_1, f_1) \preceq (L_2, f_2) \text{ if } \]
\[ L_1 \subseteq L_2 \quad \text{and} \quad f_2 \mid L_1 = f_1 . \]
\[ m_i = i \quad f_i(x) \equiv (x_i) \]

4. Ordered design, same as 3

with the condition that \( f \) is injective.

5. \( q \)-analogue of Hamming Scheme.

\( V \) an \( n \)-dim vector space over \( GF(q) \)

\( W \) a \( w \)-dim

\[ X_i = \{ (U, f) : U \text{ idim subspace of } V \]
\[ f : U \rightarrow W \text{ linear map} \}
\[ (U, f) \preceq (U', f') \text{ if } U \subseteq U', f' \mid U = f \]

6. \( q \)-analogue of ordered design.
let \((X, \leq)\) be a semilattice with length function \(l\). Let \(|x| = l(x)\).

\(\alpha\) be an integer.

\(Y \subseteq X\) is called an \(\alpha\)-intersection family if \(\forall y_1, y_2, y_3 \in Y\), \(y_1 \neq y_2 \neq y_3\).

**Thm 1** let \((X, \leq)\) be a poly-semilattice, \(\alpha\) an integer, \(Y \subseteq X\) an \(\alpha\)-intersection family then \(|Y| \leq |X_0| + |X_1| + \ldots + |X_\alpha|\)

**Thm 2** Assume conditions of Thm 1 let \(Y \subseteq X_\alpha\), \(\alpha\) an integer then \(|Y| \leq |X_\alpha|\)

**Thm 3** poly semilattice \((X, \leq)\)

\(Y \subseteq X_0, U_{i=1}^{\infty} X_{n_i} = X_\infty, n_i \geq \omega(i+1)

\|Y\| \leq |X_0| + |X_{n_0}| + \ldots + |X_{n_{\infty}}|\)
Sketch of the proof.

For any poly $g$, define a matrix $A(Y, g) \in Y \times Y$ the entry $g(y)$

$I(Y, x_i) \in Y \times \{x\}$ matrix

$(y, x)_i$th entry $= 1$ if $y \geq x$

0, otherwise

Then $I(Y, x_i) : I(Y, x_i)^T = A(Y, f_i)$

$\{y, y\}'$th entry $= \exists x : x = \exists \forall y' f(x)$ (my)

Columns of $A(Y, f_i)$ are in comb. of columns of $I[Y, x_i]$.

For a poly $g$ of deg. $\geq 1$, col. of $A(Y, g)$

are in comb. of cols $I[Y, X, U, \ldots, UX_s]$.

Then we find a poly $g$ for which

$A[Y, g]$ has rank $\sum_{j=1}^Y |Y_j|$

$|Y_j| \leq (x_1 + 1|x_1| + \ldots + |x_s|)$. 
To prove thm 3, we need to show that

\( \text{rank } I \left[ y, x_0 u, x_1 u, \ldots, x_p u \right] = \text{rank } I \left[ y, x, x_0 u, \ldots, x_p u \right] \)

i.e. columns \( x_0, x_1, \ldots, x_{\lambda + t} \)
are redundant.

Vectorspace Designs

\( V \) a \( \mathbb{F} \)-dim vector space over \( \mathbb{F} \)
\( X_i \) be the set of \( i \)-dim subspaces.

Let \( t, k, \lambda \) be integers

\( B \subseteq X_k \) is called a \( t-(\mathbb{F}^{\lambda + t}; \mathbb{F}) \)

design if for all \( T \in X_k \)

\[ | \Sigma B : B \in B, B \not\subseteq T \} | = \lambda \]

If repeated blocks are allowed
then we take \( B \) to be a family
\[ B = \{ B_i : i \in I \} \text{ where each } B_i \in X_k \text{ and } (B_i = B_i') \text{ is possible.} \]

**Standard Results**

\[ |B| - b = \frac{\alpha L^2}{k[t]} \quad \text{where } [m] \]

is the number of \( k \)-dim subspaces of an \( m \)-dim vector space over \( \mathbb{F}_q \).

Let \( I \) be a fixed \( i \)-dim subspace. Then let \( B^*_I = \{ B : B \in B, B \supset I \} \),

then \[ b_i = |B^*_I| = \alpha \frac{\binom{v-i}{t-i}}{\binom{k-i}{t-i}} \]

so we get some nec. condition.

Fisher's Inequality \( \mu_n \geq 2\alpha - [v, k, \lambda, \beta] \)

design \( b \geq [\frac{v}{\lambda}] \quad (v > 0, \lambda) \)
> ineq. holds by a result of L. Chihara.

**Analogy of Kramer's Result**

Let $(V, B)$ be a $t$-$(v, k, \lambda, \beta)$ design. $G \in GL(v, \mathbb{F})$ be an automorphism group of the design.

Then $|B \cap G| \geq \left(\frac{Xv}{d}\right)$

where $d$ blocks orbits
Some constructions by quadratic forms.

Simon Thomas (1987 Geo. Ded.)

2. \( [v, 3, 7, 2] \) for \((v, c) = 1\)

Let \( F \) be a field of order \( 2^v \)

\( F^* = \text{nonzero elements} \)

\( F \) a \( v \)-dim vector space over \( GF(2) \)

\( \sigma \in F^*, \quad x \mapsto \sigma x \) a lin. trans

A triple \( \{c_1, c_2, c_3\} \) of elements of \( F^* \)

is called a special triangle

of the pairs \( \{c_1, c_2, c_3\}, \{c_2, c_3, c_1\}, \{c_3, c_1, c_2\} \)

belong to the same orbit under \( F^* \)

let \( B = \{ \langle c_1, c_2, c_3 \rangle \mid \{c_1, c_2, c_3\} \text{ is a special triangle} \} \)

\( B \) is a

2-(v^3, 3, v^3) design which is simple
and non-trivial if \((\mathfrak{v}, \mathfrak{c}) = 1\) to 10

E. Schramm and myself gen. the let \(F\) be a field endore, \(V\) an extension f deg \(v\) over \(F\), quadratic form

\[ Q : V^k \to V \]

\[ Q(x_1, \ldots, x_k) = \sum_{i,j} \lambda_{ij} x_i x_j, \lambda_{ij} \in F \]

\(Q\) be the set of non-deg. quad. forms. For integers \(k, n, j\)

let \(D^j_k = \{ (Q, \alpha) : \alpha \in \mathbb{R}^k, \lambda (Q) = 0 \text{ and dist} \alpha, \lambda \}

\(Q(\alpha) = 0\) and dist \(\alpha, \lambda \) finite \(F = \mathbb{T}^3\)

but \(\mathfrak{F}_t = (\mathfrak{D}^j_k \times \mathbb{R}^n) \cup (\mathfrak{D}^j_k \times \mathbb{I}_F)\)
\( f_t \) is the indexing set of blocks for \((a_1, a_2, i) \in F_t\), define

\[
B((a_1, a_2, i)) = \langle a_1, a_2, a_1 > f_t
\]

Let \( B_t = \left\{ (a_1, a_2, i) \mid (a_1, a_2, i) \in F_t \right\} \)

Then, let \( \nu \) be odd. Then

\((V, B_t) \cong t - [V, K_3 \times t, 6 + 3, \lambda, \delta]\)

design. Here \( \lambda \) can be computed.

Define \((Q, a_1, i) \sim (Q', a_1, i)\) if \( i \neq i', \) \( i \in F_t \) and \( R \in \Delta (Q, i)\)

such that \( Q' = Q R^{-1} a_1 = R a_2.\)

Pick one representative from each class. Let \( B_t \) be the set of
equiv. class. Then

\[ \Theta_2 = (V, \overline{\mathbf{B}}_2) \] is a \( b - [v, k, \lambda_b, \varphi] \) design. Then

\[ \lambda_b = \frac{1}{9} \frac{1}{(q^3 - q)(q^3 - 1)} \]

\[ \lambda_2 = q^6 + q^4 + 1 \]

\[ \lambda_3 = q^3(q^2 + 2q + 1) \]

\[ \lambda_4 = q^2(q^2 + 2q + 1)(q - 1) \]

For \( t = 2 \), we get lucky.

\( \Delta_3 \) is empty so we get

\( \Theta_3 = (V, \overline{\mathbf{B}}_2) \) is a \( 2 - [v, 3, q^2 + q + 2] \) design.

and if \( 3 \nmid v \), this design is also simple. For \( q = 2 \), we get back Swan Thomas's result.
Then let $v$ be odd.

$(1)$ $\beta(D_k^v)$ is a $2-[v, k, \frac{(v^k-1)(v^k-v)}{(v^k-1)(v^k-v^2)}]$ design with $\lambda_{\alpha} = \frac{(v^k-1)(v^k-v)}{(v^k-1)(v^k-v^2)}$, $\kappa = \{0\}$.

$(2)$ If we take one representative from each equiv. class we get $\beta(D_k^v)$ is a $2-[v, k, \frac{(v^k-1)(v^k-v)}{(v^k-1)(v^k-v^2)}]$ design and $\overline{\lambda_{\alpha}} = \frac{\overline{v^k}}{(v^k-1)(\Theta_k)}$.
3-design

let \( B \in X_k \) be defining \( B \), and let \( I(\mathcal{B}) = \{ \mathcal{B} \} \) be the subfamily containing \( B \).

For a family \( \mathcal{B} = (B_i : i \in I) \)

\[ I(\mathcal{B}) = \text{the multiset } u I(\mathcal{B}_j) \]

Thus, \( I(\mathcal{B}) = \{ \mathcal{B} \} \) if \( (V, \mathcal{B}) \) is a \( t \)-\( [v, k, \lambda] \) design then \( I(\mathcal{B}) \) is a \( t \)-\( [v, k, \lambda] \) design.

Thus, \[ v \text{ odd, } k \text{ even, } k \geq 4 \]

\[ B = B(\overline{B_k} \times I_{\frac{v-k-2}{2}}) \cup I(\overline{B_{2k-2}}) \]

is a 3-\( [v, k, 3] \) design. Thus a comb.
Sign of an $n$-intersection family in a polynomial semilattice and construction of vector space designs by quadratic forms.

D.R. Ray-Chaudhuri, Ohio State University.

1973

R.M. Wilson and D.R.C. proved the following:

$V_{x,y} = \{ A : A \in X, 1 \leq x \leq \}$

Let $\mathcal{O} \leq P_\alpha(X)$ and let $\{ \{ \{ A \} : A \in \mathcal{O} \} \}$.

Then $|\mathcal{O}| \leq (V)$.

T. Zhu and D.R.C. generalized this result to polynomial semilattices.

Defn. Let $(X, \leq)$ be a partially ordered set.

Semi lattice iff for all $x, y, z \in X$ there exists $\exists x$, such that $x \leq y$.

Let $X_i = \{ x : \exists (x) \in \mathcal{O} \} = \mathcal{O}$. Polynomial semilattice iff there exist

Integers $m_0, m_1, \ldots, m_n$ and polynomial $f_0, f_1, \ldots, f_m$, satisfying

(a) $m_0 < m_1 < \ldots < m_n$

(b) $\deg f_i = i$ and for $i < j$, $f_i | f_j$

(c) for all $x, y, z \in \mathcal{O}$, $x \leq y, x \leq z$ if $f_i(x) = f_i(y)$ and $f_i(z) = f_i(y)$

Then $f_i(x) = f_i(y)$.

Ex. 1. Lattice of subsets $\mathcal{O} = P(V)$

$x_i = \{ A : A \in \mathcal{O}, m_i \leq A \}$

2. Lattice of subspaces, $V$ a vector space over a finite field of order $q$.

$L_i = \{ A : A \in \mathcal{O}, \dim A = m_i \}$

$V = \{ f : A \in \mathcal{O}, f(A) = \prod (x_{i,j}^q - x_{i,j}) \}

3. Hamming Scheme (Orthogonal Array)

$\{U \leq V : \dim U = \}$

$X = \{ (x, f) : L \in \mathcal{O}, f : L \rightarrow V \}$

$X \leq \mathcal{O}$ iff $L_1 \leq L_2$ and $f_1 \mid f_2$.

$m_i = \dim f_i(X)$

1. Ordered design, same as 3.

with the condition that $f_i$ is bijective.

5. $q$-analog of Hamming Scheme.

$V$ an $n$-dim vector space over $GF(q)$

$W = \mathcal{O}$

$X = \{ (U, f) : U \in \mathcal{O}, f : U \rightarrow \mathcal{O} $ linear $\}$

$(U, f) \leq (U', f')$ if $U \leq U'$, $f' \mid f$

6. $q$-analog of ordered design.

Let $(X, \leq)$ be a semilattice with a length function $L$. Let $L_0 = L(x)$.

Then $Y \in X$ is called an $n$-intersection family.

Then $|Y| \leq |X_1| + |X_2| + \ldots + |X_n|$

Thm 1. Let $(X, \leq)$ be a poly. semilattice, $n$ an integer, $Y \in X$ an $n$-intersection family.

Then $|Y| \leq |X_1| + |X_2| + \ldots + |X_n|$

Thm 2. Assume conditions of Thm 1.

Let $Y \subseteq X_k$, $k$ an integer.

Then $|Y| \leq |X_k|$

Thm 3. Poly semilattice $(X, \leq)$

$Y \subseteq X_1, U, X_2, \ldots, X_n, X_{n+1}, \ldots, X_{n+t}$

Then $|Y| \leq |X_1| + |X_2| + \ldots + |X_{n+t}|$
Fishers' inequality [57, 132] shows that if the design is balanced, then:
\[
\sum_{i=1}^{B} e_i = 0
\]
\[
\sum_{i=1}^{B} b_i = 0
\]

Let 
\[
\begin{bmatrix}
\frac{1}{b_1} & \cdots & \frac{1}{b_B}
\end{bmatrix}
\]
be a fixed vector of order \(B\) such that:
\[
\frac{1}{b_1} + \cdots + \frac{1}{b_B} = 1
\]

Then, the \(b_i\)'s are the number of \(a_i\)'s in each level. Let 
\[
[1 \times B]
\]
be the design matrix of the \(B\) levels and 
\[
[\text{vec}(a_i)]
\]
be the design matrix of the \(a_i\)'s in each level. Then:
\[
\begin{bmatrix}
[1 \times B]
\end{bmatrix}
\begin{bmatrix}
[\text{vec}(a_i)]
\end{bmatrix}
= [1 \times B]
\]

For each \(i\), there is a \(b_i\) such that:
\[
[1 \times B]
\begin{bmatrix}
\text{vec}(a_i)
\end{bmatrix}
= b_i
\]

Then, the proof relies on the fact that:
\[
[1 \times B]
\begin{bmatrix}
[\text{vec}(a_i)]
\end{bmatrix}
= b_i
\]

The design matrix is:
\[
[1 \times B]
\begin{bmatrix}
[\text{vec}(a_i)]
\end{bmatrix}
= b_i
\]

For all \(i\), there is a \(b_i\) such that:
\[
[1 \times B]
\begin{bmatrix}
[\text{vec}(a_i)]
\end{bmatrix}
= b_i
\]

Then, the proof relies on the fact that:
\[
[1 \times B]
\begin{bmatrix}
[\text{vec}(a_i)]
\end{bmatrix}
= b_i
\]

The design matrix is:
\[
[1 \times B]
\begin{bmatrix}
[\text{vec}(a_i)]
\end{bmatrix}
= b_i
\]
Some constructions by

Quadratic forms.

Simon Thomas (1987, Def. 11).

2. Let $F$ be a field of order $2^v$.

$F^* = \text{nonzero elements}$

$F$ a $v$-dim vector space over $GF(2)$

For $Q \in F^*$, $x \mapsto -x$ a lin. trans.

A triple $\{c, c', c''\}$ of elements $F^*$ is called a special triangle.

The pairs $\{c, c', c''\}$ belong to the same orbit under $F^*$

let $\beta = \{c, c', c''\}$ special triangle.

$\mathcal{T}_2$ is a 2-$(v^3, v^3, 1)$ design which is unique.

Some non-trivial $\delta(v,c) = 1$

Let $F$ be a field of order $2^v$.

$\mathcal{Q}(\mathcal{T}_2)$ be a set of non deg. quad.

forms. For integers $k$ and $s$.

let $\mathcal{A}_k = \mathcal{Q}(\mathcal{T}_2)$.

The $\mathcal{A}_k$ of $\mathcal{T}_2$ over $F = GF(2)$

let $\mathcal{F}_k = (\mathcal{A}_k^t \times I, v(\mathcal{A}_k^t \times I))$.
\[ (B(x) \oplus c) \oplus \{y - x \mid y \in \mathbb{Z} \} \]

Let \( B \in \mathbb{D}_2 \) denote a 3-design. Let \( \mathcal{B}_2 \) be a 2-design. If \( \mathcal{B}_2 \) is a 2-design, then \( \mathcal{B}_2 \) is a 2-design. For any 2-design \( \mathcal{B}_2 \), the support of \( \mathcal{B}_2 \) contains \( \mathcal{B}_2 \). For any 2-design \( \mathcal{B}_2 \), the support of \( \mathcal{B}_2 \) contains \( \mathcal{B}_2 \).
A Graph-theoretic Game
and its Application to the
k-Server Problem

Prof. Douglas B. West
Department of Mathematics
University of Illinois-Urbana
GRAPH-THEORETIC GAME AND ITS APPLICATION TO THE K-SERVER PROBLEM

Noga Alon
Richard Karp
Douglas West
The Game

Given \((G,w)\), a connected multigraph \(G\) with positive edge weights \(w(e)\)

define a matrix game: tree player picks \(T\)
edge player picks \(e\)

payoff to edge player is 
\[
c(T, e) = \begin{cases} 
0 & \text{if } e \in T \\
\frac{\text{cycle}(e)}{w(e)} & \text{if } e \notin T
\end{cases}
\]

Mixed strategies:
- \(p = \text{prob. dist on trees}\) (to limit expected loss)
- \(q = \text{prob. dist on edges}\) (to guarantee expected gain)

Minimax Theorem of Game Theory:
\[
\min_p \max_q \sum_e \sum_{T \in p} q_e c(T, e) = \max_q \min_p \sum_e \sum_{T \in p} q_e c(T, e)
\]

The common value is 
\[
\text{Val}(G,w) = \max_q \min_p \sum_e \sum_{T \in p} q_e c(T, e)
\]
\[
\text{Val}(G) \text{ if } w = 1 \text{ (unweighted)}
\]

Note: \(\text{Val}(G,w) \leq n\) by using pure strategy MST
complete graph $K_n$, $w=1$.

uniform edge strategy guarantees at least
\[ \frac{n-1}{(n-2)} \cdot 0 + \left[ 1 - \frac{n-1}{(n-2)} \right] \cdot 3 = 3 - 6/n \] against any tree, equality only for stars.

uniform star tree strategy guarantees at most
\[ \frac{2}{n} \cdot 0 + \left[ 1 - \frac{2}{n} \right] \cdot 3 = 3 - 6/n \] against any edge.

$\therefore \text{Val}(K_n) = 3 - 6/n$

small diameter

Weighted cycles

\[ \sum w_i \]
\[ T_i = C_n - e_i \]
\[ W = \sum w_i \]

or $p_i = \frac{w_i}{W}$, every edge has expected payoff $(1 - \frac{w_i}{W}) \cdot 0 + \frac{w_i}{W} \cdot \frac{W}{w_i} = 1$

or $q_i = \frac{w_i}{W}$, "tree"

$\therefore \text{Val}(C_n, w) = 1$

Cages (unweighted)

4-regular graphs with girth $c \log n$

$n+1$ edges of cost $\geq c \log n$ \[ \text{Val} \in \Omega(\log n) \]

Explicit family: GRIDS!

Conjecture: $\text{Val}(G) \in O(\log n)$?
The k-Server Problem

Given metric space $M$, service requests processed by $k$ servers.

Process by moving server to request location.

Cost = distance moved by servers.

Initial positions $\pi$, request sequence $\rho$.

$OPT(\pi, \rho) =$ optimal off-line service cost.

$A(\pi, \rho) =$ service cost by (deterministic) on-line algorithm $A$.

An on-line algorithm $A$ is $c$-competitive if

$A(\pi, \rho) \leq c \cdot OPT(\pi, \rho) + a$ for all $(\pi, \rho)$.

- If $|M| > k$ and $c < k$, $\exists$ $c$-competitive deterministic on-line alg.

- Bounded competitiveness always achievable

  Note: greedy doesn't work

Road network: $d(x,y) =$ shortest journey

Model by $(G,w)$

Chrobak-Larmore: For a tree-like road network, there is $k$-competitive deterministic on-line alg.
randomized (on-line) algorithm
Algorithm uses outcome of an experiment, so \( A(\pi, \rho) \) is a random variable
Adversary: may specify entire \( \rho \) in advance = oblivious
may specify next request based on service choices = adaptive

A is \( c \)-competitive, if
\[
E(A(\pi, \rho)) \leq c \cdot \text{OPT}(\pi, \rho) + \alpha \quad \text{for all } (\pi, \rho)
\]
against oblivious adv.

Theorem 1: If \((G, w)\) models a road network \( M \), then
\[
\exists k(1 + \text{Val}(G, w))- \text{competitive randomized on-line algorithm for the k-server problem on } M \text{ against an oblivious adv.}
\]

Proof: Algorithm:
\[
\rightarrow \text{Use optimal tree strategy on } (G, w) \text{ to select tree } T.
\rightarrow \text{Along each } e \in T, \text{ pick a random point } x_e \text{ to cut at.}
\rightarrow \text{Process } \rho \text{ along resulting } G' \text{ using C-L algorithm}
\]

C-L implies
\[
A(\pi, \rho) \leq k \cdot \text{OPT}'(\pi, \rho) \quad \text{for all } (\pi, \rho) \text{ and experimental } G'
\]
suffices to show
\[
E(\text{OPT}'(\pi, \rho)) \leq (1 + \text{Val}(G, w)) \cdot \text{OPT}(\pi, \rho)
\]
for online algorithm is valid.
Proof of \( E(\text{OPT}'(\pi, \rho)) \leq [1 + \text{Val}(G, w)] \text{OPT}(\pi, \rho) \)

Simulate the moves for \( \text{OPT}(\pi, \rho) \) on \( G' \).

I.e., when asked to cross roadblock on \( c \) traverse cycle(\( c \)) to get to the other side.

Cost of detour is cycle(\( c \)) when cross \( x_e \) given \( T \) chosen with probability \( p(T) \) ... 

Let \( d(e) = \text{total distance traveled on } c \text{ by } \text{OPT}(\pi, \rho) \)

If \( c \notin T \), expected \# times cross random point \( x_e \) is \( \frac{d(e)}{w(e)} \)

Expected cost of detours for \( c \) is \( \frac{d(e)}{w(e)} \left\{ \text{cycle}(c) \text{ if } c \notin T \right\} = d(e) c(T, e) \)

Expected total cost of detours is

\[
\sum_T p(T) \sum_e d(e) c(T, e) = \text{OPT}(\pi, \rho) \sum_T \sum_e p(T) \left( \frac{d(e)}{w(e)} \right) c(T, e) 
\leq \text{OPT}(\pi, \rho) \text{ Val}(G, w) \]

\( E(\text{OPT}'(\pi, \rho)) \leq E(\text{this simulation procedure}) \leq [1 + \text{Val}(G, w)] \text{OPT}(\pi, \rho) \)
Optimization Problem

What is the best tree against the uniform edge strategy?

Let \( F_{G,w}(T) = \frac{1}{|E|} \sum_{c} c(T,c) \) (minimizes the average cost)

Let \( v(G,w) = \min_{T} F_{G,w}(T) \) (use \( v(G) \) if \( w \equiv 1 \))

Theorem 2. \( \text{Val}(G,w) = \sup_{(G,w')} v(G,w') \), where \((G,w')\) ranges over all weighted multigraphs obtained from \((G,w)\) by replication.

Proof: Given \((G,w)\), let \( d(e) = \# \) copies of \( e \).

Then \( v(G,w') = \min_{T} \frac{1}{\sum d(e)} \sum_{c} d(e) c(T,c) \)

\[ = \min_{T} \sum_{c} q(e) c(T,c) \quad \text{where} \quad q(e) = \frac{d(e)}{\sum d(e)} \]

\[ \leq \max_{q} \min_{T} \sum_{c} q(e) c(T,c) = \max_{q} \min_{T} \sum_{c} p_{T} q (c(T,c)) = \text{Val}(G,w) \]

Even optimal \( q \), \((G,w')\) can approximate it with \( d(e) = 1 + \frac{1}{M} \) as \( M \to \infty \)

Theorem 3. If \( G \) is edge-transitive, then \( \text{Val}(G) = v(G) \).

Proof: Take \( T \) with \( F_{G}(T) = v(G) \) and images \( \bar{T} \) under \( \Pi(G) \).

"Play \( T \) with probability \( \frac{1}{|\Pi(G)|} \) \# \{ \sigma \in \Pi(G) : \sigma(T) = T' \} \)

Then expected payoff for any edge \( e \) is

\[ \frac{1}{|\Pi|} \sum_{\sigma} c(\sigma(T),c) = \frac{1}{|\Pi|} \sum_{\sigma} c(T,\sigma'(c)) \]

LaGrange
Maximum Val for n-vertex multigraphs - UNWEIGHTED

\[ \Omega(\log n) \leq \max_{n(G) \leq f(n)} \text{Val}(G). \leq e^{\frac{\log \log \log n}{f(n)}} \]  

Proof:

\[ \text{Suffices to prove that } v(G') \leq e^{\frac{\log \log \log n}{f(n)}} \]

\[ \text{Begin by reducing attention to multigraphs with } \leq n(n+1) \text{ edges.} \]

Replace \( G' \) by \( H \) such that \( v(G') \leq 2v(H) \) and \( |E(H)| \leq n(n+1) \)

\( H \) has same underlying graph with \( D \) distinct edges as \( G' \).

Multiplicities \( h(e) = 1 + \left\lfloor \frac{g(e)}{2g(e)} \right\rfloor \)

\[ P_n(T) = \frac{1}{Zg(e)} e^{h(e)c(T)} \geq \frac{1}{2D} \frac{Zg(e)Dc(T)}{Zg(e)} = \frac{1}{2} P_G(T) \]

\[ \text{Recursive construction of tree} \]

Seek large clumps with small diameter and few edges between

Given an integer \( x = x(n) > 1 \), partition \( V(G') \) into parts such that

A) each part has \( > x \ln n \) vertices.

B) each part has spanning tree of diameter \( \leq 8x \ln n \).

C) fraction of the edges joining vertices in distinct parts \( \leq \frac{1}{x} \).

Use these trees within these parts, contract parts, and build tree recursively on edges between parts.
build partition: Build parts one by one
Components of remaining graph have \( \geq x \ln n \) vertices.
Take a vertex in a remaining component \( K \), stratify by levels.

\[ V_i = \text{vertices at distance } i \text{ in } K \text{ (from start)} \]
\[ E_i = \text{edges within } V_i \text{ or } \text{to } V_{i-1} \]

Let \( i^* \) least \# shells such that \( |V_0 v \cdots V_{i^*}| > x \ln n \)
and \( |E_{i^*}| \leq \frac{1}{x} |E_1 v \cdots E_{i^*}| \).

Now, partition \( V_0 v \cdots V_{i^*} \) and vertices of \( K - V_0 v \cdots V_{i^*} \)
in components of size at most \( x \ln n \).

\( \sqrt{C} \) hold by construction. To show diameter \( < 8x \ln n \):

Let \( i' = \text{least level so } |V_0 v \cdots V_{i'}| > x \ln n \)
Note \( i' \geq x \ln n \) and \( |E_1 v \cdots E_{i'}| \geq x \ln n \)

Claim: \( i^* < 3x \ln n \).

Else \( |E_1 v \cdots E_{i^*}| \geq x \ln n (1 + \frac{1}{x})^{i^* - 1} |V_0 v \cdots V_{i^*}| = x \ln n \frac{x}{(1 + \frac{1}{x})}^{i^*} > x \ln n n^{2x} > n (\text{rel}) \).

Recurrence: Let \( z = 8x(n) \ln n \).

\[
f(n) \leq 2 \left[ z + \frac{1}{x} f \left( \frac{8n}{z} \right)(1 + z) \right]
\]

1 - \( H \) instead of \( x \)
2 - from diameter bound on parts
3 - fraction of edges between parts
4 - bound on \# parts
5 - dilation for passing through parts

With \( M = 17 \ln n \), have \( f(n) \leq M \left[ x(n) + f \left( \frac{n}{x(n)} \right) \right] \)

Iterate recurrence, choosing \( n_0 = n \quad n_i = n_{i-1}/x(n_{i-1}) \).

With \( x = \text{e}^{\sqrt{x(n) \ln n}} \), obtain \( f(n) \leq \text{e}^{C' \sqrt{\ln n \ln n}} \)
Theorem 5: For grid $G$ with $N = n^2$ vertices, \( v(G) \in \Theta(\lg N) \), and hence $\text{Val}(G) \in \Omega(\lg N)$.

Upper bound: Let $n = 2^k$.

Define tree $T_k$ by four copies of $T_{k-1}$, plus center.

Diameter $d_k = 3 + 2d_{k-1}$, solution $d_k = 3(2^k-1) = 3(n-1)$

Average cost:

\[
F(T_k) \leq 4 \left( \frac{2 \frac{d_k}{2}(\frac{d_k}{2}-1)}{d_k} \right) F(T_{k-1}) + (2n-3) d_k < F(T_{k-1}) + 3 = F(T_0) + 3k = 3\lg n
\]

Lower bound: Main idea—show that for an arbitrary tree, some edges yield long cycles, somewhat more yield cycles with a smaller lower bound on length, etc.

Count up lower bounds on *edges* with given lower bound on cycle(s) and proxy!

Lemma: If $A$ is vertex subset with $|A| = \alpha^2 \leq n^2/2$, then $\exists$ at least $\alpha$ rows or at least $\alpha$ columns that $A$ meets but doesn't fill.

Proof: Suppose $A$ hits $r$ rows, $s$ cols, $r \geq s$. Then $rs \geq \alpha^2 \Rightarrow r \geq \alpha$.

Done unless $A$ fills a row, but then $s = n - r$. If $A$ fills more than $n - \alpha$ rows and $n - \alpha$ columns, then $A$ has more than $n^2 - \alpha^2$ vertices.
Lemma 2 If $|A| = \alpha^2 < \frac{n^2}{2}$ and $|B| \leq 4$, then at least $\alpha/2$ vertices of $A$ have neighbors outside $A$ and distance $\geq \alpha/16$ from all of $B$.

Proof: Pick $\alpha$ vertices from distinct rows with outside nrs; $B$ eliminates $\leq \alpha/2$

Lemma 3 For any sp. tree $T$ and $\alpha \leq n/4$, at least $\frac{n^2}{32} \alpha$ edges e have cycle $(e) > \alpha/16$.

Proof: Max degree 4 guarantees bifurcation as balanced as $\frac{1}{4}, \frac{3}{4}$. Iteratively cut biggest till get $m = \lfloor \frac{n^2}{4} \alpha^2 \rfloor$ pieces.

Claim: smallest piece has $\geq \alpha^2$ vertices. Minimizing $x_i$ st. $x_1 + \cdots + x_m = \frac{3}{4} x_i$ and $\sum x_i = M$ set $x_i = M/(4m-1)$

Average # deleted edges incident with a piece is $2$.

$\therefore$ At least half the pieces incident to at most 4 deleted edges.

Lemma 2 guarantees $\frac{\alpha}{2}$ verts w distance $\geq \alpha/16$ to exit.

$\left(\frac{1}{2} \text{ edges} \right) \times \left(\frac{1}{2} \text{ endpoints} \right) \times \left(\frac{n^2}{4} \alpha^2 \text{ pieces} \right) = \left(\frac{n^3}{32} \alpha \text{ edges} \right)$

Proof of Theorem: Given $T$

Choose edge $c$ at random, set $X = c(T, e)$

Then $F(T) = E(X) = \sum_{k \geq 1} \text{Prob}(X \geq k)$.

If $k \leq n/64$, set $\alpha = 16k$.

Then $\text{Prob}(X \geq k) \geq \frac{n^2/512k}{2n(n-1)} > \frac{1}{1024k}$.

Theorem $F(T) \geq \sum_{k=1}^{n/4} \frac{1}{1024k} \sim \frac{\ln n}{1024}$.
A GRAPH-THEORETIC GAME
AND ITS APPLICATION
TO THE K-SERVER PROBLEM

Noga Alon
Richard Karp
Douglas West

The Game

Given \((G, w)\), a connected multigraph \(G\) with positive edge weights \(w(e)\).

Define a matrix game: tree player picks \(T\) edge player picks \(e\)

payoff to edge player is \(c(T,e) = \begin{cases} 0 & \text{if } e \notin T \\ \infty & \text{if } e \in T \end{cases}\)

Mixed strategies:

\[
p = \text{prob dist on trees (no limit expected gain)}
q = \text{prob dist on edges (no guarantee expected gain)}
\]

Minimax Theorem of Game Theory:

\[
\min_{p} \max_{q} \sum_{e} p(e) c(T,e) = \max_{q} \min_{p} \sum_{e} p(e) c(T,e)
\]

optimal payoff for trees = expected payoff for edges

The common value is \(\text{Val}(G, w)\).

\(\text{Val}(G)\) if \(w\) is (un)weighted.

Note: \(\text{Val}(G, w) \leq n\) by using pure strategy MST

Examples

1) Complete graph \(K_n, w = 1\).

uniform edge strategy guarantees at least
\[
\frac{\log n}{n} \cdot 0 + \left(1 - \frac{\log n}{n}\right) \cdot 0 = 1 - \frac{\log n}{n}
\]
against any tree.

uniform star tree strategy guarantees at most
\[
\frac{\log n}{n} \cdot 0 + \left(1 - \frac{\log n}{n}\right) \cdot 3 = 3 - \frac{3\log n}{n}
\]
against any edge.

\:: \text{Val}(K_n) = 3 - \frac{3\log n}{n}

2) Weighted cycles

\[
\bigcirc \quad T = C_n - e; \quad W = \Sigma w_e
\]

For \(n = \frac{w}{W}\), every tree has expected payoff \((\frac{w}{W}) \cdot 0 + \frac{W - w}{W} \cdot 1 = \frac{1}{n}\)

\:: \text{Val}(C_n, w) = \frac{1}{n}

3) Cubes (unweighted)

I-regular graphs with girth \(\geq 2\) and \(e = 2\)

\:: \text{Val}(G, w) = 0(1/m)

Explicit family: GRIDS

Conjecture: \(\text{Val}(G) \in O(1/m)\)?

The k-Server Problem

Given metric space \(M\), service requests processed by \(k\) servers.

\[
\text{Cost} = \text{distance moved by servers}
\]

Initial positions \(\pi\), request sequence \(\rho\)

Let \(\text{OPT}(\pi, \rho)\) = optimal offline service cost

Let \(A(\pi, \rho)\) = service cost by (deterministic) online algorithm \(A\)

An online algorithm \(A\) is \(c\)-competitive if

\[
A(\pi, \rho) \leq c \cdot \text{OPT}(\pi, \rho) \text{ for all } (\pi, \rho)\)

\[
\text{If } |M| > k \text{ and } c > k, \text{ no } c\text{-competitive deterministic online alg}
\]

Bounded competitiveness always achievable

Note: greedy doesn't work

Road network:

Model by \((G, w)\)

- Lighthart-Lamore: For a tree-like road network

there is \(k\)-competitive deterministic online alg
Randomized (online) algorithm
Algorithm uses outcome of an experiment,
\( p(A, \pi, \rho) \) is a random variable
Adversary: may specify entire \( \pi \) in advance; oblivious
may specify next request based on service choices; adaptive
A is c-competitive, \( \sum E(A(\pi, \rho)) \leq c \cdot OPT(\pi, \rho) \) for all \( \pi, \rho \)
Adversary: may specify entire \( \rho \) in advance; oblivious

Theorem 1: If \( (G_w) \) models a road network \( M \), then
there exists a \( k \)-competitive randomized online algorithm for the \( k \)-server problem on \( M \) against an oblivious adversary.

Proof: Algorithm:
- Use optimal tree strategy on \( (G_w) \) to select tree \( T \).
- Along each edge \( e \in T \), pick a random point \( x_e \) to cut at.
- Process \( \rho \) along resulting \( G' \) using C-L algorithm.

C-L implies \( A(\pi, \rho) \leq k \cdot OPT(\pi, \rho) \) for all \( \pi \) and \( \rho \).

\[ C \cdot L \implies E(\pi, \rho) \leq k \cdot OPT(\pi, \rho) \]

An Optimization Problem
What is the best tree against the uniform edge strategy?
Let \( F_{\pi, T} \) be the average cost
\( \min_{\pi} \sum_{e \in E} \pi(e) \cdot v(e) \)

Theorem 2: \( \text{Val}(G_w) = \max_{\pi} \sum_{e \in E} \pi(e) \cdot v(e) \)

Proof: Given \( (G_w) \), let \( v(a) = v(a) \cdot \pi(a) \).
Then \( v(G_w) = \min_{\pi} \sum_{e \in E} \pi(e) \cdot v(e) \cdot \pi(e) \).

Maximum Val for \( n \)-vertex multigraphs - UNWEIGHTED

\[ \text{Val}(G) = \max_{\pi} \sum_{e \in E} \pi(e) \cdot v(e) \]

Proof:
Suffices to prove that \( v(G') = \max_{\pi} \sum_{e \in E} \pi(e) \cdot v(e) \)
- Begin by reducing attention to multigraphs with unique edges.
- Replace \( G' \) by \( H' \) such that \( v(G') = v(H') \) and \( |H'| = |E'| \).
- \( H' \) has same underlying graph with \( 0 \)-distinct edges as \( G' \).

Given optimal \( \pi \), \( (G_w) \) can approximate it with \( v(G') = v(G') \) for \( \pi \).

Theorem 3: If \( G \) is edge-transitive, then \( \text{Val}(G) = v(G) \).
Proof: Take \( T \) with \( F_{\pi, T} = v(G) \) and images \( T' \) under \( (G) \).
Play \( T' \) on \( G' \) with probability \( \frac{1}{|G'|} \cdot \frac{v(T')}{v(T')} \).

Then expected payoff for any edge \( e \) is
\[ \frac{1}{|G'|} \sum_{T' \in \pi} v(T') \cdot \frac{1}{|G'|} \cdot \frac{v(T')}{v(T')} = \frac{1}{|G'|} \frac{1}{|G'|} v(T, e) = v(G) \]

Random Play:
In a random play, let \( \pi \) be uniform.

Proof: \( E(\pi, \rho) \leq [1 + \text{Val}(G) \cdot \text{OPT}(\pi, \rho)] \)

Given \( T \) chosen with probability \( p(T) \).
Let \( d(e) \) be total distance traveled on \( e \) by \( \text{OPT}(\pi, \rho) \).

<table>
<thead>
<tr>
<th>Total Cost of Movement</th>
</tr>
</thead>
<tbody>
<tr>
<td>( E(\pi, \rho) = E(\pi, \rho) \cdot \text{Val}(G_w) \cdot \text{OPT}(\pi, \rho) )</td>
</tr>
</tbody>
</table>

Maximum Val for \( n \)-vertex multigraphs - WEIGHTED

\[ \text{Val}(G) = \max_{\pi} \sum_{e \in E} \pi(e) \cdot v(e) \]

Proof:
Suffices to prove that \( v(G') = \max_{\pi} \sum_{e \in E} \pi(e) \cdot v(e) \)
- Begin by reducing attention to multigraphs with unique edges.
- Replace \( G' \) by \( H' \) such that \( v(G') = v(H') \) and \( |H'| = |E'| \).
- \( H' \) has same underlying graph with \( 0 \)-distinct edges as \( G' \).

Given optimal \( \pi \), \( (G_w) \) can approximate it with \( v(G') = v(G') \) for \( \pi \).

Theorem 3: If \( G \) is edge-transitive, then \( \text{Val}(G) = v(G) \).
Proof: Take \( T \) with \( F_{\pi, T} = v(G) \) and images \( T' \) under \( (G) \).
Play \( T' \) on \( G' \) with probability \( \frac{1}{|G'|} \cdot \frac{v(T')}{v(T')} \).

Then expected payoff for any edge \( e \) is
\[ \frac{1}{|G'|} \sum_{T' \in \pi} v(T') \cdot \frac{1}{|G'|} \cdot \frac{v(T')}{v(T')} = \frac{1}{|G'|} \frac{1}{|G'|} v(T, e) = v(G) \]

Random Play:
In a random play, let \( \pi \) be uniform.
To

parfi~iow:

w&iict

V-

Vi

CyQ,
ri cs

pai'+3 aw

vart~ats &4 dals4~c

~.
a~ ede

{'

0

Ba.L.

Z t. K

V

0'

-

Cc
41 oL lpercw

Uppe

boOnd

Lorg

b1ond

-,

".t

&o= I@M%+4stl

SC

k~

101Ist Plav. prt.ke 14...

A

IV*

41"^

-4

u-j
Ad vart-cos of

La j j'

les

4

.VJ

z-

Z0M

6oi

T^41m

.

,

r-AlW

A/

17.... Ne.

f(-) f M .~f(j)

~'

f

I

G
Cis+ace~

x all(

For an

sp. tree

n1i

rf
Ct'e'":

~

of

ctiI

fec
l.s i rows
B*~ii-a*.3
el
gp/

1

W14, cait

Loenitc3

+row,

*o~ie
A~

Proo: ic

r tnc

A

&C19A&A~cd.
ka~l

aJ,(
&4 121"Oed fil( got

PMaIld

Pbiece

hAS 1?i'

A~'e~. leltauCo~i*

A4 leasf halif 460
LCm.... 2. qUr~V'

?ro4Thmere:

I-koaso~..
C___-JM

7

et)tag

en

,.

jAijei.if W~

04VrS-

F(T)

n/q

cycle)~
;Wx/(

V.O*CU.

pieces iACil+

Giveni

Le.ttria. IF

A

n915
t

cs vorerfg

sulWsef

rmevts bM+
4

n+

~p~ ~~
3(2

o.,.,d

3

.FT,'

FT.

wil

~Vijfj

given~ lew

w.

lAI x (,:

?'1*5 Sitpose

A K;+

fill,

r rows, I cols, r es.

P on* 14eistis A Clib . row

ver+jcfl of A have neigh~ovs otA4cide A

'"

3 24

of

then 1 Qa+ltia+ o..rows or of levz+ A colum.ns ikA+ A

~ ~~ ~
tV"Ik M

'y' fo.,r~e

Low", tft
Meim .ct's. 5"W-+ 4~An
fo
xr6;fdry. 4ree,
Same..
t
oile Yil
y
OA7 cyls
>'
SOMCIf Pec
i.
cw&Ia ifi a Sstitev low., bone on
Icny*j6, &it.

£ Afd

;pKI )
Lek

du

CT-Ks"2 V 4JF(T..J. z-)

Cla im 10 r 2xifn A.'-L'sI'"'1

Re-rtftnav

ver T.

D....r

K- V.-*.

So IV.

teVu

Neig ! :!wi..

2

Lin
Odin.q

Iln qa
4
M~j

Pieces-

7

A, .6

sA.,

q&,4P

a piece fS _2.
fo a4 ones+ 4.

d~5IU1Sco

if

a.A.*.d

To exct.

.04-A.

bu ."

s - n.

he rs $(

cl

ee


Polynomial-Time Algorithms from
Finite Basis Theorems - A Survey

Prof. Michael Langston
Department of Computer Science
University of Tennessee
POLYNOMIAL-TIME ALGORITHMS 
FROM FINITE BASIS THEOREMS 
— A SURVEY—

MIKE LANGSTON

• RELEVANT GRAPH METRICS

• DECISION ALGORITHMS

• SEARCH ALGORITHMS

• CONSTRUCTIVIZATIONS

• APPROXIMATE BASES

• FUTURE WORK
SOME AMENABLE METRICS

CUTWIDTH, MOD. CUTWIDTH

PATHWIDTH, TREENWIDTH

SEARCH NUMBER, NODE SEARCH NUMBER

NP-COMPLETE IN GENERAL

FOR FIXED WIDTH (k), BEST
PREVIOUS BOUNDS: OPEN

O(\exp(k))

NEW BOUNDS: \(\Theta(n^2)\)
- **IMMERSION ORDER** \( (\leq i) \)
  
  - Take subgraph
  
  - Lift pair of edges

- **EXAMPLE:** \( C_4 \leq i K_1 + 2K_2 \ (\nmid_t, \nmid_m) \)
THM [RS] (NASH-WILLIAMS' CONJ)  
$\leq_\iota$ IS WPO

THM [FL] $\exists$ POLY-TIME TEST  
FOR $H \leq_\iota G$ FOR EVERY 
FIXED $H$.

- **CLOSURE**:  $G \in F$  
  $\Rightarrow H \in F$  
  $H \leq_\iota G$

- **OBSTRUCTION SET**

- **IMMERSION-CLOSED $\Rightarrow$ POLY-TIME  
  DECISION ALG**
• Non constructive at two levels

• Horrible constants

• Time complexity:

\[ O(1 \cdot 1^h) \text{ where } h \]

denotes order of largest obstruction
SAMPLE APPLICATION:

- $k$-MIN CUT
  
  $O(|V|^k)$ [MS]

- NEITHER "YES" NOR "NO"
  FAMILIES CLOSED $\leq_m$

- "YES" FAMILY CLOSED $\leq_c$

  $\therefore \in P$

SELF-REDUCIBILITY $\mathcal{C}(|V|^{h+6+3})$
Better bounds for $\leq_i$

$k$-min cut $[O(\sqrt{h+6})]$

- "YES" family excludes a (large) binary tree, $T$

  $T \leq_m G \Rightarrow T \leq_i G \Rightarrow G \in "NO"$ family

- "YES" family has bounded tree-width

- Tree decomposition $\leq_i$ tests, etc are $O(\sqrt{h+6})$
CONSTRUCTIVITY

\((R, \leq)\) is uniformly enumerable if there is a recursive enumeration \((r_0, r_1, \ldots)\) of \(R\) such that \(r_i \leq r_j \implies i \leq j\).

Under such an enumeration, a self-reducible algorithm is uniform if, on input \(r_j\), it consults its oracle only for \(r_i\) with \(i \leq j\).

Example: \(R = \text{finite graphs}\)

\(\leq_m \quad (\text{or } \leq_i)\)

Self-reducible by edge deletion
HM. IF $F$ IS CLOSED IN A KNOW UNIF. ENUM. WPO $\exists$ WE/ALGS TO:

1) CHECK CANDIDATE SEARCH SOLUTION IN $O(T_1(n))$ TIME

2) PERFORM ANY OBSTRUCTION TEST IN $O(T_2(n))$ TIME

3) DO UNIFORM SELF-RED

4) DO (SOME FORM OF) SELF-RED WITH $O(T_3(n))$ CALLS & OVERHEAD, THEN WE CAN CONSTRUCT AN

$O(\max \{T_1(n), T_2(n) \cdot T_3(n)\})$ TIME ALG TO COMPUTE A SEARCH FCN FOR $F$ (& HENCE DECIDE MEMBERSHIP IN $F$).
EXAMPLE: $k$ MIN CUT

CHECKING \[ T_1(n) = O(n) \]

OBST TESTS \[ T_2(n) = O(n^2) \]

UNIF. SELF-RED. (SLOW, USES GADGET)

SELF-RED. \[ T_3(n) = \Omega(n) \] (WE USE SCAFFOLDING & AMORTIZED ANALYSIS)

COROLLARY: $k$ MIN CUT CAN BE SOLVED IN $O(n^3)$ TIME WITH A KNOWN ALGORITHM.

OBSERVATION: $\exists$ A LACK OF SYM. WORKS ONLY FOR "YES" CLOSURE
PRACTICAL POSSIBILITIES:
- LEARNING SYSTEM
- SMALL # OF REQ'D OBSTS
- FAST OBST TESTS

THEORETICAL EXTENSION:
- RETAIN OBSTRUCTION TESTING & GENERATION (IN CASE "NO")
- REPLACE SELF-RED. WITH TM DIAGONALIZATION (IN CASE "YES")
- INTERLEAVE
SUPPOSE AN NP-COMPLETE PROBLEM IS FOUND TO BE MINOR- OR IMMERSION-CLOSED?

THEN P = NP NONCONSTRUCTIVELY

NOT ONLY A SURPRISE BUT, WHAT'S WORSE, NP-COMPLETENESS ROOFS DO NOT (AS GENERALLY CLAIMED) PROVIDE POLY-TIME ALGS!
But, as long as the poly-time reduction is known, we can exploit:

1) SAT ≤ T and T closed ⇒ image of SAT closed

2) Checking solutions is uniformly self-reducing SAT are easy

yielding a constructive P=NP proof (without any means for isolating the relevant obst. set).
**APPROXIMATE OBSTRUCTION SETS**

- **USE:** \( \{\text{HANDY OBSTS}\} \cup \{\text{HANDY NON-OBSTS}\} \)

<table>
<thead>
<tr>
<th></th>
<th>≠ HANDY NON-OBST</th>
<th>← FALSE NEGATIVES</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;YES&quot;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;NO&quot;</td>
<td>≥ HANDY OBST</td>
<td>← FALSE POSITIVES</td>
</tr>
</tbody>
</table>

- **EXAMPLE:** 3-GML (2-pw) AND \( \{K_4\} U \{K_4\} \)
  
  [FAST \( K_4 \) TOPO TEST YIELDS FEW FP's]

4-GML (3-pw) AND \( \{K_5\} U \{K_{3,3}\} \)

[FAST PLANARITY TEST YIELDS FEW FP's, FN's]

- **MAKES SELF-REDUCTION INTERESTING:**
  - "YES" Instance → "YES" Evidence without FP.
QUESTIONS

- PRACTICALITY: NONCONSTRUCTIVE ✓
  - DECIDE ONLY ✓
  - HUGE CONSTANTS ~

- BANDWIDTH: TOPO, Δ≤3 ✓
  - GENERAL, FOR FIXED k?

- TREE-DECOMP BOTTLENECK: $O(n^2)$ ✓
  - FASTER? ($O(n \log^2 n)$ recently claimed)

- PARALLEL ALGS: TREE-DECOMP ✓
  - ORDER TESTS ✓
  - DISJOINT PATHS?

- SELF-REDUCTION: ALWAYS POSSIBLE?
  - ALWAYS UNIFORM?
POLYNOMIAL-TIME ALGORITHMS
FROMFINITE BASIS THEOREMS
-A SURVEY-
MIKE LANGSTON

-RELEVANT GRAPH METRICS
-DECISION ALGORITHMS
-SEARCH ALGORITHMS
-CONSTRUCTIVIZATIONS
-APPROXIMATE BASES
-FUTURE WORK

SOME AMENABLE METRICS

- CUTWIDTH, MOD. CUTWIDTH
- PATHWIDTH, TREEWIDTH
- SEARCH NUMBER, NODE SEARCH NUMBER

NP-COMPLETE IN GENERAL
FOR FIXED WIDTH (\(\mathcal{W}\)), BEST
PREVIOUS BOUNDS: OPEN
\(O(\exp(k))\)
NEW BOUNDS: \(O(n^2)\)

THM [RS] (NASH-WILLIAMS' CONJ)
\(\preceq\) IS WPO

THM [FL] \(\exists\) POLY-TIME TEST
FOR \(H \preceq G\) FOR EVERY FIXED \(H\)

- CLOSURE: \(G \in F \Rightarrow H \in F\)
\(H \preceq G\)

- OBSTRUCTION SET

- IMMERSION CLOSED \(\Rightarrow\) POLY-TIME
DECISION ALG

\(C_4 \preceq K_{1+2K_2} (\mathcal{W}_2, \mathcal{W}_1)\)

\(K_{1+2K_2}\)

\(C_4 \subseteq \mathcal{H}\)
NON CONSTRUCTIVE AT TWO LEVELS

HORRIBLE CONSTANTS

TIME COMPLEXITY:
$O(IV^{h+6})$ WHERE $h$
DENOTES ORDER OF LARGEST
OBSTRUCTION

SAMPLE APPLICATION:

- $\mathcal{K}$ - MIN CUT
  $\mathcal{O}(IV^{h-1})$ [MS]

- NEITHER "YES" NOR "NO"
  FAMILIES CLOSED $\leq_m$

- "YES" FAMILY CLOSED $\leq_c$
  $\therefore \in P$

- SELF-REDUCIBILITY $\mathcal{O}(IV^{h+6+2})$

BETTER BOUNDS FOR $\leq_c$

$\mathcal{K}$ - MIN CUT $[\mathcal{O}(IV^{h+6})]$

- "YES" FAMILY EXCLUDES A
  (LARGE) BINARY TREE, $T$
  $T \leq_m G \Rightarrow T \leq_c G$
  $\Rightarrow G \in "NO"$ FAMILY

- "YES" FAMILY HAS BOUNDED
  TREE-WIDTH

- TREE DECOMPOSITION, $\leq_c$ TESTS,
  ETC ARE $\mathcal{O}(IV^2)$

CONSTRUCTIVITY

- $(R, \leq)$ IS UNIFORMLY ENUMERABLE
  IF $\exists$ RECURSIVE ENUMERATION $(r_0, r_1, \ldots)$
  OF $R \ni r_i \leq r_j \Rightarrow i \neq j$.

- UNDER SUCH AN ENUMERATION, A SELF-
  RED. ALG IS UNIFORM IF, ON INPUT $r_i$
  IT CONSULTS ITS ORACLE ONLY FOR
  $r_j$ WITH $i \leq j$.

EXAMPLE: $R =$ FINITE GRAPHS
$\leq_m (\text{OR } \leq_c)$

SELF-RED. BY EDGE DELETION
THM. If $F$ is closed in a known unif. enum. WPO $\equiv$ $\text{WE}/\text{ALGS}$ to:

1) Check candidate search solution in $O(T_1(n))$ time
2) Perform any obstruction test in $O(T_2(n))$ time
3) Do uniform self-red
4) Do (some form of) self-red with $O(T_3(n))$ calls & overhead, then we can construct an $O(\max\{T_1(n), T_2(n), T_3(n)\})$ time alg to compute a search fn for $F$ (hence decide membership in $F$).

EXAMPLE: $\text{MIN CUT}$

- Checking $T_1(n) = O(n)$
- Obst tests $T_2(n) = O(n^3)$
- Unif. self-red. (slow, uses gadget)
- Self-red. $T_3(n) = O(n)$ (we use scaffolding & amortized analysis)

COROLLARY: $\text{MIN CUT}$ can be solved in $O(n^3)$ time with a known algorithm.

OBSERVATION: A lack of sym. works only for "yes" closure.

PRACTICAL POSSIBILITIES:
- Learning system
- Small # of req'd obsts
- Fast obst tests

THEORETICAL EXTENSION:
- Retain obstruction testing & generation (in case "no")
- Replace self-red with TM diagonalization (in case "yes")
- Interleave

SUPPOSE AN NP-COMPLETE PROBLEM IS FOUND TO BE MINOR- OR IMMERSION-CLOSED?

THEN $P=NP$ nonconstructively.

NOT ONLY A SURPRISE BUT, WHAT'S WORSE, NP-COMPLETENESS PROVES DO NOT (AS GENERALLY CLAIMED) PROVIDE POLY-TIME ALGS!
But, as long as the polytime reduction is known, we can exploit:

1) SAT $\equiv$ $\Pi$ and $\Pi$ closed $\Rightarrow$ image of SAT closed

2) Checking solutions uniformly self-reducing SAT are easy

Yielding a constructive $P=NP$ proof (without any means for isolating the relevant OBST set).

Questions

- Practicality: nonconstructive $\bigvee$
  - Decide only $\bigvee$
    - Huge constants $\sim$

- Bandwidth: $\text{topo} \Delta \leq 3$ $\bigvee$
  - General, for fixed $\Delta$?

- Tree-decomp bottleneck: $O(n^2)$ $\bigvee$
  - Faster? ($O(n \log^2 n)$ recently claimed)

- Parallel algs: tree-decomp $\bigvee$
  - Order tests $\bigvee$
  - Disjoint paths $\bigvee$

- Self-reduction: always possible $\bigvee$
  - Always uniform $\bigvee$

Approximate Obstruction Sets

- Use: $\{\text{Handy OBSTS}\} \cup \{\text{Handy non-OBSTS}\}$

- Example: 3-GML (2-pw) and $\{K_4 \cup \}$
  - Fast $K_4$ topo test yields few FPS
  - 4-GML (3-pw) and $\{K_5 \cup [K_3]\}$
  - Fast planarity test yields few FPS, FN's

- Makes self-reduction interesting:
  - "YES" instance $\Rightarrow$ "YES" evidence without FP.
On the Product of the independent Domination Numbers of a Graph and its Complement

Prof. Gerd H. Fricke
Department of Mathematics and Statistics
Wright State University
On the Product of the Independence Domination Numbers of a Graph and its Complement

Let $p = tm$ and consider the following graph $G$ and $\overline{G}$.

For a graph $G$ let

$$ i(G) = \min \{ |S| : S \text{ is a maximal independent set} \} = \min \{ |S| : S \text{ is a dominating independent set} \}. $$

$i(G)$ is the smallest cardinality of a maximal clique in $G$.

Let $\text{mii}(p) = \max \{ i(G) : i(\overline{G}) \}$. If $|G| = p$, it is easy to show that $\text{mii}(p)$ is nondecreasing.

Also

$$ \left( \frac{(p+3)^2}{16} \right) \leq \text{mii}(p). $$

Recently Cockayne, Favaron, Li, and MacGillivray showed that

$$ \text{mii}(p) \geq \min \left\{ \frac{(p+3)^2}{4}, \frac{(p+4)^2}{4} \right\}. $$

Theorem: Let $0 < k < 16$. Then there exists an integer $p = p_k$ such that

$$ \text{mii}(p) \geq \frac{p}{k} \text{ for all } p \geq p_k. $$

Proof: Let $0 < k < 16$ and let $G$ be a graph of $p$ vertices such that $i(G) : i(\overline{G}) > \frac{p}{k}$. Note that any vertex is contained in an independent set of size $i(G)$ and a clique of size $i(\overline{G})$.

Let $s = i(G)$ and $r = i(\overline{G})$ and assume $s > r$.

$$ \text{mii}(p) \geq \frac{p}{k} \text{ for all } p \geq p_k. $$
Each vertex in $X$ is contained in a complete $K_{p,p}$ and is adjacent to at least $rp-1$ vertices in $V \setminus X$. Then we have at least $sp(rp-1)$ edges to $p - sp$ points in $V \setminus X$. Let \( \frac{sp(rp-1)}{p - sp} = s^*\). Then there exists a point \( V \), such that \( v \in V \setminus X \) and is adjacent to at least $s^*p$ vertices in $X$.

Again any vertex \( v \in X \), is contained in a maximal clique of size $\geq p$ and that clique contains at most one vertex of $X$, (namely \( v \)) and at most one vertex of $X_i$.

Thus every \( v \in X \), has $rp-1$ edges to vertices not in $X$, or $X_i$.

Let \( \frac{sp(rp-2)}{p - 4sp - rp} = s^*p \)

Hence there is a vertex in $X \cup X_i$ that is adjacent to $s^*p$ vertices of $X_i$.

Simplify and we have

\[
\frac{S(r - \frac{3}{2})}{1 - s^* - s} = s^* \quad \text{and repeat the argument.}
\]

\[
X = \frac{1 - s - \sqrt{(1 - s)^2 + 4s + \frac{3}{4}}}{2}
\]

\[
X \geq \frac{\sqrt{\frac{(1 - s)^2}{1 - s} - \frac{2s}{1 - s} - \frac{3}{2}}}{1 - s}
\]

Since $rs = \frac{1}{k} > \frac{1}{2}$ and $r > \frac{k}{2}$ we have $(\frac{3}{2} - \frac{k}{2} - s) > 0$ for $p > \frac{k}{2}$.

Hence $X > rs$. Let $s = x$.

\[
\frac{S}{s^*p}
\]

\[
\frac{\sqrt{\frac{(1 - s)^2}{1 - s} - \frac{2s}{1 - s} - \frac{3}{2}} - \frac{1}{k}}{s^*p}
\]

\[
\left( s\frac{p + s}{p}\right) \left( \frac{r - \frac{3}{2}}{p} \right) = s^*.
\]

Every $x \in U_w$ is in a maximal clique $3p$ and thus an $(s,p + sp)(r,p - 2)$ edge to points not in $U_w$.

\[
\frac{(s,1)(r - \frac{3}{2})}{1 - s - s} = s^*.
\]
Then sepsep vertices of \( UW \) don’t contain a triangle and hence are (sepsep) (p-s) edges to vertices not in \( UW \).

\[
\frac{(S_4 + 5)(r - \frac{3}{4})}{1 - s - s} = S_5
\]

In general

\[
\frac{(S_n + 5)(r - \frac{3n}{4})}{1 - s - s} = S_{n+1}
\]

Thus with an \( h > 0 \) such that

\[
S_{n+1} - S_n > \frac{1}{2} \frac{\Delta}{h} \quad \text{for any } \Delta \text{ and } n, h \leq \frac{1}{2}.
\]

Hence \( S_n > 1 - s \) for some \( n \) with \( h > 0 \), which contradicts \( S_n < 1 - s \).
On the Product of the Independence Domination Numbers of a Graph and its Complement

Gerd H. Fricke
Wright State U.

For a graph $G$ let

$$i(G) = \min \{ |S| : S \text{ is a maximal independent set} \}$$

$$= \min \{ |S| : S \text{ is a dominating independent set} \}.$$ 

$i(G)$ is the smallest cardinality of a maximal clique in $G$.

Let $m_{ii}(p) = \max \{ i(G), i(\overline{G}) \}$.

It is easy to show that $m_{ii}(p)$ is nondecreasing.

Also

$$\left\lfloor \frac{(p+3)^2}{16} \right\rfloor \leq m_{ii}(p)$$
Let $p = 4m$ and consider the following graph $G$.

\[ G \]

$\uparrow$ complete $\quad \uparrow$ independent

\[ \overline{G} \]

\[ i(G) = m + 1 \quad i(\overline{G}) = m + 1 \]

\[ mii(p) \geq \frac{(p+4)^2}{16} \]

Now, \quad $i(G) + i(\overline{G}) \leq p + 1$ and thus

\[ \left\lfloor \frac{(p+3)^2}{16} \right\rfloor \leq mii(p) \leq \frac{(p+1)^2}{4} \]
Recently Cochayne, Favaron, et al., and Mac Gillivray showed that

\[ mii(p) = \min \left\{ \frac{(p+3)^2}{3}, \frac{(p+8)^2}{10.8} \right\} \]

**Theorem:** Let \( 0 < K < 16 \) then there exists an integer \( p_0 \) such that

\[ mii(p) = \frac{p^2}{K} \quad \text{for all } p \geq p_0. \]

\[ \lim_{{p \to \infty}} \frac{mii(p)}{p^2} = \frac{1}{16} \]
Proof: Let $0 < \kappa < 16$ and let $G$ be a graph of $p$ vertices such that $\delta(G) \delta(\overline{G}) > \frac{p^2}{K}$.

Note that any vertex is contained in an independent set of size $\geq \delta(G)$ and a clique of size $\delta(\overline{G})$.

Let $s_p = \delta(G)$ and $t_p = \delta(\overline{G})$ and assume $r \leq s_p$.

\[ X \]
\[ \geq s_p \]
\[ \subseteq Y \]
\[ \geq t_p \]

$X$ independent

$Y$ complete
Each vertex in $X$ is contained in a complete $K_{rp}$ and is adjacent to at least $rp-1$ vertices in $V - X$. Thus we have at least $sp(rp-1)$ edges to $p - sp$ points in $V - X$.

Let $\frac{sp(rp-1)}{p - sp} = s^* + p$, then there exists a point $V_i$ such that $V_i \in V - X$ and is adjacent to at least $s^* + p$ vertices in $X$. 

\[ X \]

\[ X^* \]

\[ \geq sp \]

\[ \geq s^* + p \]
Again any vertex $v \in X_1$ is contained in a maximal clique of size $\geq r_1 p$ and that clique contains at most one vertex of $X_1$ (namely $v$) and at most one vertex of $X_1^*$.

Thus every $v \in X_1$ has $r_1 p - 2$ edges to vertices not in $X_1$ or $X_1^*$.

Let $\frac{3p(r_1 p - 2)}{\rho - s_1^* - r_1 p - s} = s_2^*$

Hence there exists a $v_2 \in X_1 \cup X_1^*$ that is adjacent to $s_2^* \rho$ vertices of $X_1$.

Simplify and we have:

$$\frac{s \left( r - \frac{2}{\rho} \right)}{1 - s_1^* - s} = s_2^*$$

and repeat the argument.
Let
\[ \frac{S(r - \frac{2}{p})}{1 - S_{2^*} - s} = S_3^* \]
and in general
\[ \frac{S(r - \frac{2}{p})}{1 - S_{n^*} - s} = S_{n+1}^*. \]

\[ \{S_n^*\} \] is increasing and let \( \lim_{n \to \infty} S_n^* = X. \)

Then
\[ \frac{S(r - \frac{2}{p})}{1 - X - s} = X \]
and
\[ X^2 - (1 - s)X + sr - \frac{2s}{p} = 0. \]
\[ X = \frac{1-s - \sqrt{(1-s)^2 - 4s \alpha + \frac{\alpha^2}{p}}}{2} \]

\[ X \geq \frac{1-s}{2} - \frac{1}{2} \sqrt{\left[ 1 - \left(1+r\right)s \right]^2 + 4s \left( \frac{3}{p} - \frac{1}{p} - \frac{r}{16} \right)} \]

Since \( rs \geq \frac{1}{K} > \frac{1}{16} \) and \( r > \frac{1}{16} \) we have \( \left( \frac{2}{p} - \frac{1}{16} - \frac{r}{16} \right) > 0 \) for \( p > 30 \).

Hence \( X > rs \). Let \( s_1 = X \).

Every \( x \in U W_j \) is in a maximal clique \( \geq r p \) and there are \(( s_1 p + s p)( r p - 2)\) edges to points not in \( U W_j \): \[
\frac{(s_1 + s)( r - \frac{2}{p})}{1 - s_1 - s} = s_2.
\]
Thus $s_2 + s_p$ vertices of $W_6 W_1 W_2 W_3$ don't contain a triangle and there are $(s_2 + s_p)(p-3)$ edges to vertices not in $U W_j$.

$$\frac{(s_2 + s)(r - \frac{3p}{r})}{1 - s_2 - s} = S_3$$

In general

$$\frac{(S_n + s)(r - \frac{n+1}{p})}{1 - s_n - s} = S_{n+1}.$$
\[
S_{n+1} - S_n = \frac{(r+s)S_n + rs + s^2 - S_n - (s_n + s) \frac{n+1}{p}}{1 - s - s_n}
\]

Let \( S_n = rs + d_n \) and note that \( S_n > S_i \geq rs \) and thus \( d > 0 \).

Thus \( A = (r+s)S_n + rs + s^2 - S_n \)

\[
= (r+s+rs)rs + d_n^2 + (r+s+2rs-1)d_n
\]

Now \( 0 < d = rs - \frac{1}{16} < \frac{1}{4} \) and thus

\[
\sqrt{r_s} = \sqrt{\frac{1}{16} + d} \geq \frac{1}{4} + \frac{3}{2}d
\]

Also \( r+s \geq 2 Fr_s \geq \frac{1}{2} + 3d \)

Thus \( A \geq \left(\frac{1}{2} + \frac{1}{16} + 4d\right)\left(\frac{1}{16} + d\right) + d_n^2 + \left(\frac{1}{2} + \frac{3}{2} + s(2d-1)\right)d_n
\]

\[
= \frac{9}{16}d + \frac{13}{16}d + 4d^2 + (s(2d-\frac{7}{4}) + d)dn + d_n^2
\]

\[
= \left(\frac{3}{16} - d_n\right)^2 + \frac{13}{16}d + 4d^2 + s(2d-\frac{7}{4})dn
\]

\[
\geq 2h \text{ for some } h > 0.
\]

Choose \( p \geq h^2 \) then for \( n+1 \leq \frac{1}{h} \) we have

\[
S_{n+1} - S_n = \frac{A - (s_n + s) \frac{n+1}{p}}{1 - s - s_n} \geq 2h - \frac{n+1}{p} \geq \frac{h}{1 - s - s_n} > \frac{4}{3}h.
\]

Now \( S_n > S_i + (n-1) \frac{4}{3}h \) and \( S_n \geq 1-s \) for some \( n \) with \( n+1 < \frac{1}{h} \), which contradicts \( S_n < 1-s \).
There exists an $h > 0$ such that

$$S_{n_h} - S_n \geq \frac{4}{3} h$$

for $p \geq h^2$ and $n+1 \leq \frac{1}{n}$.

Hence $S_k \geq 1-s$ for some $k$ with $k+1 \leq \frac{1}{n}$ which contradicts that $S_n < 1-s$. 
Containment of Circular-Arcs

Prof. Jeremy Spinrad
Department of Computer Science
Vanderbilt University
Containment of Circular-Arcs

circular-arc graph

vertices $\implies$ arcs on circle

$x - y$ iff arcs intersect

Diagram:

```
  a -- b
  |
  d -- c
```

```
  a
  |
  b
```

```
  d
  |
  c
```
Application?

Hypothesis: genes arranged in circular pattern

\[ g_k \circ \begin{array}{c}
g_1 \\ \\
g_2 \\ \\
g_3 \\
\end{array} \]

Mutations damage consecutive portion of gene

\[ \downarrow \]

Mutation 1 \( g_1 \) cleft palate

Mutation 2 \( g_2 \) harelip

Mutation 3 \( g_3 \)

Must be a circular-arc graph
Recognition

posed: Klee, 1969

can be used to test genetic hypothesis

solved: Tucker, 1982

difficult algorithm

also Hsu, 1990

recognition and isomorphism
Containment

\( \forall x, y \in N(x) \subseteq N(y) \)?

easy to stretch are so

\[ \frac{x}{y} \]

\( \text{iff } N(y) \subseteq N(x) \)

\( a / \text{the? bottleneck step of Tucker's recognition algorithm} \)

naive: \( O(n^3) \)

\( O(MM) \)

this talk: \( O(n^2) \)
General Approach

Transform to a set of bipartite problems

$G$ circular-arc $\Rightarrow G'$ chordal bipartite

use special properties of chordal bipartite graphs to get good algorithms
Chordal Bipartite

bipartite, any cycle of length \( \geq 6 \) has a chord

close correspondences

\( \beta \)-acyclic hypergraphs
totally balanced matrices
strongly chordal graphs
Key Characterization [CHKS]

\[ \Gamma = \begin{array}{cccccc}
& & & & & \\
& & & & & \\
& & & & & \\
& & & & & \\
& & & & & \\
& & & & & \\
& & & & & \\
\end{array} \]

\[ \begin{array}{cccccc}
& & & & & \\
& & & & & \\
& & & & & 0 \\
& & & & & \\
& & & & & \\
& & & & & \\
& & & & & \\
\end{array} \]

\[ G \text{ chordal bipartite } \iff \]
\[ M_b(G) \text{ can be } \Gamma \text{-free ordered } \iff \]
\[ \text{doubly lexical order } M_b(G) \text{ } \Gamma \text{-Free} \]

\[ \text{can verify that matrix is } \Gamma \text{-Free in linear time Clubini} \]
Doubly Lexical Ordering

Input: Matrix M

<table>
<thead>
<tr>
<th>3</th>
<th>1</th>
<th>5</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>4</td>
<td>9</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>

Permute so it read down/up, right/left, rows and columns ↑

<table>
<thead>
<tr>
<th>3</th>
<th>5</th>
<th>7</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>9</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>4</td>
<td>9</td>
</tr>
</tbody>
</table>

arbitrary matrices: \( O(m \log n) \)

[Cubitt], [Paige + Tarjan]

Graphs or 0/1 matrices: \( O(n^2) \)
Chordal Bipartite Containment
doubly lexical order $M_{rb}(G)$

$N(x) \leq N(y) \ ?$

$\begin{array}{cccc}
1 & 1 & \text{1st row}\n\end{array}$

$\begin{array}{ccc}
\text{otherwise} & y & - x
\end{array}$

$\begin{array}{cccc}
\text{x} & \text{y} & \text{y} & \text{y} \ \n\end{array}$

$\begin{array}{cccc}
\text{y} & \text{y} & \text{o} & \text{y} \ \n\end{array}$
Side Issue

Nonredundant Is represent chordal bipartite by only those 1 values which cannot be implied by $\Gamma$-freeness.

```
1 0 1 0
1 1 * 0
0 1 * 1
```

how many nonredundant Is can there be?
Open Problems

Known: $\Omega(n \log n)$ nonredundant:

$O(n^{3/4+\epsilon})$

Conjecture $\Theta(n \log n)$

if true, optimal representation

How many chordal bipartite graphs?

$\Omega(2^{cn \log^2 n})$

$O(2^{n^{3/4+\epsilon} \log n})$
Relation to Circular-Arc Graph:

Select minimal arc \( v \)

\[ N(v) \]

I is an interval graph

\( N(v) \) covered by 2 cliques
Step 1

If $G$ is circular-arc, then

$N(v)$ is chordal bipartite

*Example:*

```
  v
 /\   /
|   |  \\
| n1 | n2 |
 \   /
  |   |
  | n3 |
```
"Proof"

Let C be a cycle.

No arcs in "bottom" $N(v)$ contain others.

Top member of cycle misses exactly 2 bottom members, must be consecutive.
$x$ misses $n_i, n_j$

$x \in I \implies x$ misses $n_{i+1}$

$x \in N(v) \implies$

$x$ misses $n_{i+1}$
could lay out cycle so neighbors of any top vertex adjacent

convex \subseteq \text{chordal bipartite}

contrastion

N(v) = \overline{N(v)}\quad \text{complement is chordal bipartite}

can compute all containment relation with respect to edges to \text{N}(v) in \mathcal{O}(n^2)
Containment wrt Edges to $I$

1) $n_1, n_2 \in N(v) \subseteq n_2 \in N(v)$

$N(v)$ complement is chordal bipartite

2) $i_1, i_2 \in I \subseteq i_2 \in I$

easy from "standard representation" of interval graph

3) $i_1, i_2 \in I \subseteq n_1, n_2 \in N(v)$

next slide
Lay out interval graph

"Standard": start/end in same maximal clique ⇒ same endpoint

\[ x \in N(v) \]

'walk through' I. i start+point, i \( \rightarrow \) x, i \( \leq \) x's next nonneighbor after endpoint of

\[ O(nm) \]?

Store only endpoints of I. can 'walk through' I in

\[ O(n^2) \] time.
What's Next?

1) Does Tucker's algorithm run in $O(n^2)$ time?

2) Simpler $O(n^2)$ recognition

3) What else on circular arc graphs is easier than constructing representation?

   independent set

4) other uses, chordal bipartite graphs trapezoid graphs
Containment of Circular - Arcs

Circular-arc graph
vertices \( \Rightarrow \) arcs on circle
\( x - y \) iff arcs intersect

Recall

Recognition
posed: Klee, 1969

Application?

hypothesis: genes arranged in circular pattern

\[ g_1, g_2, g_3 \]

mutations damage consecutive portion of gene

\[ \downarrow \]

mutation 1

\[ g_1 \text{ cleft palate} \]

mutation 2

\[ g_2 \text{ hare lip} \]

mutation 3

\[ g_3 \]

must be a circular-arc graph

Containment

\[ \forall x, y \text{ is } N(x) \subseteq N(y) ? \]

easy to stretch arc so

\[ \overline{x} \]

iff

\[ N(y) \subseteq N(x) \]

\( a \) the bottleneck step of Tucker's recognition algorithm

naive: \( O(n^3) \)

\( O(MM) \)

this talk: \( O(n^2) \)
General Approach

Transform to a set of bipartite problems

G circular-arc $\Rightarrow$

G' chordal bipartite

use special properties of chordal bipartite graphs
to get good algorithms

Chordal Bipartite

bipartite, any cycle of length $\geq 6$ has a chord

close correspondences

$\beta$-acyclic hypergraphs
totally balanced matrices

Strongly chordal graphs

Key Characterization [HKSJ]


G chordal bipartite $\Leftrightarrow$

$M_{\beta}(G)$ can be $\Gamma$-free ordered $\Leftrightarrow$

doubly lexical order $M_{\beta}(G)$ $\Gamma$-free

can verify that matrix is $\Gamma$-Free in linear time [Lubiw]

Doubly Lexical Ordering

Input: Matrix M

\begin{bmatrix} 3 & 1 & 5 & 7 \\ 2 & 4 & 6 & 3 \\ 4 & 1 & 2 & 4 \end{bmatrix}

Permute so if read down/up,
right/left, rows and columns $\uparrow$

\begin{bmatrix} 3 & 5 & 7 & 1 \\ 2 & 4 & 6 & 3 \\ 4 & 1 & 2 & 4 \end{bmatrix}

arbitrary matrices: $O(m \log n)$

[Lubiw], [Paige + Tarjan]

Graphs or 0/1 matrices: $O(n^2)$
Chordal Bipartite Containment

doubly lexical order $M_B(G)$

$$N(x) \leq N(y) \iff \begin{cases} 1 \text{st } 1 \text{ in row } x \\
N(x) \leq N(y) \text{ iff } y - u \\
\text{otherwise} \\
x < y \\
y \leq 0 \\
\Gamma \\
\end{cases}$$

Open Problems

Known $\Omega(n \log n)$ nonredundant $O(n^{3/4+c})$

Conjecture $\Theta(n \log n)$
if true, optimal representation

How many chordal bipartite graphs?

$\Omega(2^{cn \log^2 n})$

$O(2^{n^{3/4+c} \log n})$

Side Issue

Nonredundant Is represent chordal bipartite by only those 1 values which cannot be implied by $\Gamma$-free sets

1 0 1 0
1 1 * 0
0 1 * 1

how many nonredundant Is can there be?

Lower Bound

\[
\begin{array}{c|c|c}
\text{nonredundant}(n) & \geq 2 \eta(n) + \frac{n}{2} \\
\Omega(cn \log n) & \text{any perfect matching in upper left} \\
\Omega(2^{cn \log^2 n}) \text{ graphs} & \text{if } \Theta(2^{cn \log^2 n}), \text{ optimal}
\end{array}
\]
Relation to Circular-Arc Graph.

Select minimal arc \( v \)

\[ N(v) \]

\[ I \]

\( I \) is an interval graph

\( N(v) \) covered by 2 cliques

"Proof"

Let \( C \) be a cycle

\( \text{no arcs in "bottom" } N(v) \text{ contain others} \)

\[ n_1 \quad n_2 \quad n_3 \quad n_4 \]

Top member of cycle misses exactly 2 bottom members must be consecutive

Step 1

if \( G \) is circular-arc, then

\[ N(v) \Rightarrow I \]

\( N(v) \)

Bipartite complement is chordal bipartite

e.g.

\[ n_1 \quad n_2 \quad n_3 \quad i_1 \quad i_2 \quad i_3 \quad i_4 \]

\[ n_1 \quad n_2 \quad n_3 \]

\[ x \text{ misses } n_1, n_3 \]

\[ x \in I \Rightarrow x \text{ misses } n_{i+1} \]

\[ x \in N(v) \Rightarrow \]

\[ x \text{ misses } n_{i+1} \]
Could lay out cycle so neighbors of any top vertex adjacent

Convex ≤ chordal bipartite

Contradiction

\[ N(v) \cap I \] complement is chordal bipartite

\[ N(v) \] can compute all containment relation with respect to edges to \( N(v) \) in \( O(n^2) \)

### Containment wrt Edges to \( I \)
1) \( n_i \in N(v) \subseteq n_2 \in N(v) \)
2) \( n_i \in I \leq n_2 \in I \)
   easy from "standard representation" of interval graph
3) \( i, i \in I \leq n_i \in N(v) \)
next slide

---

**What's Next?**

1) Does Tucker's algorithm run in \( O(n^2) \) time?
2) Simpler \( O(n^2) \) recognition
3) What else on circular arc graphs is easier than constructing representation? independent set
4) other uses, chordal bipartite graphs
   trapezoid graphs

---

**Lay out interval graph**

"Standard": start/end in same maximal clique ⇒ same endpoint

\[ x \in N(v) \]

'walk through' \( I \); i start/point, \( i \rightarrow x \); is \( x \)'s next
nonneighbor after endpoint of \( O(nm) \)?

Store only endpoints of \( I \).
can 'walk through' \( I \) in \( O(n) \) time.

\( O(n^2) \)
A Fast Parallel Recognition Algorithm
for
a Class of Tree–representable Graphs

Stephan Olariu
Department of Computer Science
Old Dominion University
Common metrics for "local density"

- Complete graph (clique)
- Cliques with a "few" edges missing
- No "long" paths allowed
- A "few" long paths allowed

"long" path
Definition: A graph $G$ is P4-sparse if no set of five vertices of $G$ induces more than one P4.
Cographs: a class of graphs containing no P4s

P4-reducible graphs: a class of graphs such that every vertex belongs to at most one P4

Applications: scheduling, computational semantics, pattern recognition etc.
Definition  For every graph $G$ consider the graph $C(G)$ returned by the following procedure:

Procedure Greedy($G$);
{Input: an arbitrary graph $G$;
Output: a graph $C(G)$}
begin
  $C(G) = G$;
  while there exists a P4 in $C(G)$ do
    pick an arbitrary P4 $uvxy$;
    pick $z$ at random in $\{u,y\}$;
    $C(G) = C(G) - \{z\}$;
  return($C(G)$)
end; {Greedy}

Theorem  For a graph $G$ with no induced $C_5$ the following statements are equivalent:
(i) $G$ is P4-sparse;
(ii) for every induced subgraph $H$ of $G$, $C(H)$ is unique up to isomorphism
Consider $G_1=(V_1,\emptyset)$ and $G_2=(V_2,E_2)$ ($V_1 \cap V_2 = \emptyset$) with $V_2 = \{v\} \cup K \cup R$ such that

- $|K|=|V_1|+1 \geq 2$
- $K$ is a clique.
- Every vertex in $R$ is adjacent to all the vertices in $K$ and non-adjacent to $v$.
- There exists a vertex $v'$ in $K$ such that $N_{G_2}(v) = \{v'\}$ or $N_{G_2}(v) = K - \{v'\}$.

Choose a bijection $f: V_1 \to K - \{v'\}$ and define

$$G_1 \circ G_2 = (V_1 \cup V_2, E_2 \cup E')$$

with

$$E' = \begin{cases} 
\{xf(x) \mid x \in V_1\} & \text{whenever } N_{G_2}(v) = \{v'\} \\
\{xz \mid x \in V_1, z \in K - \{f(x)\}\} & \text{whenever } N_{G_2}(v) = K - \{v'\}
\end{cases}$$

**Theorem** $G$ is a $P_4$-sparse graph if, and only if, $G$ is obtained from single-vertex graphs by a finite sequence of operations $\circ, \circ, \circ$. $\square$
Procedure Build_tree(G);
{Input: a $P_4$-sparse graph $G=(V,E)$;
Output: the ps-tree $T(G)$ corresponding to $G$.}
begin
if $M = 1$ then
    return the tree $T(G)$ consisting of the unique vertex of $G$;
if $G$ ($\overline{G}$) is disconnected then begin
    let $G_1, G_2, ..., G_p$ (p$\geq$2) be the components of $G$ ($\overline{G}$);
    let $T_1, T_2, ..., T_p$ be the corresponding ps-trees rooted at $r_1, r_2, ..., r_p$;
    return the tree $T(G)$ obtained by adding $r_1, r_2, ..., r_p$ as children of a node
    labelled 0 (1);
end
else begin {now both $G$ and $\overline{G}$ are connected}
    write $G = G_1 \otimes G_2$
    let $T_1, T_2$ be the corresponding ps-trees rooted at $r_1$ and $r_2$;
    return the tree $T(G)$ obtained by adding $r_1, r_2$ as children of a node labelled 2
end
end; {Build_tree}
An example...

\[ G \]

\[ G_1 (\{b,c\}, \Phi) \]

\[ G_2 (\{a\} \cup \{a',b',c'\} \cup \{d\}, \{aa', a'b', a'c', b'c', a'd, b'd, c'd\}) \]

\[ K \ R \]
An example...

a P4-sparse graph

the corresponding tree
A set $C$ of vertices of $G$ is termed \textit{regular} if it admits a partition into non-empty, disjoint sets $K$ and $S$ satisfying the following conditions:

(r1) $|K| \geq |S| \geq 2$, $S$ stable, $K$ a clique;

(r2) every vertex in $V-C$ belongs to precisely one of the sets:

$\text{T}(C) = \{x \in V-C \mid x \text{ adjacent to all the vertices in } C\}$;

$\text{I}(C) = \{x \in V-C \mid x \text{ non-adjacent to all the vertices in } C\}$;

$\text{P}(C) = \{x \in V-C \mid x \text{ adjacent to all the vertices in } K \text{ and non-adjacent to all the vertices in } S\}$.

(r3) there exists a bijection $f: S \rightarrow K$ such that either $\text{N}(x) \cap K = \{f(x)\}$ for every $x$ in $S$,

or else $\text{N}(x) \cap K = K - \{f(x)\}$ for every $x$ in $S$.
Let $G = (V,E)$ be an arbitrary graph.

Fact (Regularity is hereditary)

Let $C = (K,S,f)$ be a regular set in $G$ and let $Z$ be a subset of $S$ with $|Z| < |S| - 2$. Then $C' = C - \{x, f(x) \mid x \in Z\}$ is regular.

Fact (Containment)

Let $C = (K,S,f)$ be regular. For every pair of distinct $u, v$ in $C$ with $u \neq f(v)$ and $v \neq f(u)$, the unique $P_4$ containing $u$ and $v$ belongs to $C$.

Fact (Black hole property)

A regular set is maximal if, and only if, every regular $P_4$ containing a vertex in $C$ is included in $C$.

Fact (Separation property)

Two maximal regular sets coincide whenever they intersect.
The "world" of regular sets
Given an arbitrary graph $G$ construct a graph $G^*$ as follows:

remove in every maximal regular set $C = (K, S, f)$ all the vertices in $S$ except for an arbitrary one

Theorem For every graph $G$, the graph $G^*$ is unique up to isomorphism

Theorem For an arbitrary graph $G$ the following statements are equivalent:
(i) $G$ is $P_4$-sparse;
(ii) $G^*$ is a cograph
Algorithm Recognize(G);
{Input: an arbitrary graph G;
Output: "yes" or "no" depending on whether or not G is P4-sparse}

Step 1. Find all maximal regular sets in G;

Step 2. Compute $G^*$;

Step 3. if G is a cograph then
    return("yes")
else
    return("no")

Step 4. Stop.
Our algorithm:

O(\log n) EREW time using $O\left(\frac{n^2+mn}{\log n}\right)$ processors

What we do:

- recognize P4-sparse graphs;
- construct the corresponding tree
The Algorithm

- The EREW model of computation is assumed;
- G is an arbitrary graph represented by adjacency lists;
- for every vertex x, assign one processor to every entry on the adjacency list of x;
- the vertices are enumerated as \( v_1, v_2, \ldots, v_n \) in a way that will be explained later;
- sets will be represented by their characteristic vector;
- computing the cardinality of a set takes \( O(\log n) \) time using \( O(n/\log n) \) processors;
- given sets \( S, S' \) of vertices of G, computing \( S - S', S \cup S', S \cap S' \), as well as testing \( S = O, S \subseteq S' \) takes \( O(\log n) \) time using \( O(n/\log n) \) processors;
- to compute \( N[x] \) we need \( O(\log n) \) time and \( O(n/\log n) \) processors.
Processor assignment:

- for every x of G, every entry on the adjacency list of x receives one processor;
- every edge $e_i$ ($i=1,2,...,m$) receives $1+\left\lceil \frac{n}{\log n} \right\rceil$ processors

$$P(e_i, 0), P(e_i, 1), ... P(e_i, m)$$

Note: the total number of processors is bounded by

$$O\left( \frac{mn}{\log n} \right)$$

How do we find a regular set?

$$N_{uw} = N[u] - N[w] \quad N_{wu} = N[w] - N[u]$$
\[ N_{vw} = N[v] - N[w] \quad \text{and} \quad N_{wv} = N[w] - N[v] \]

**Fact** The edge \( vw \) is the *midedge* of a regular P4 in \( G \) only if \( |N_{vw}| = |N_{wv}| = 1 \), and \( u,z \notin E \) with \( u, z \) standing for the unique vertex in \( N_{vw} \) and \( N_{wv} \), respectively.

- For every edge \( e = vw \) the sets \( N_{vw} \) and \( N_{wv} \) can be computed in \( O(\log n) \) time as follows:
  - \( N[v] \) will be broadcast to all \( d_G(v) \) edges incident with \( v \).
  - \( N[w] \) will be broadcast to all \( d_G(w) \) edges incident with \( w \).

Note: Total number of processors \( O\left(\frac{n}{\log n} \sum d_G(v)\right) = O\left(\frac{mn}{\log n}\right) \)
Procedure Find-Regular_P4s(G);
0. begin
1. for every edge e_i=(v,w) of G do in parallel begin
2. \( N_{vw} \leftarrow N[v] - N[w]; \)
3. \( N_{wv} \leftarrow N[w] - N[v]; \)
4. if \( N_{vw} \cap N_{wv} = \emptyset \) then (let \( N_v = \{u\}, N_w = \{z\}, U = \{u,v,w,z\} \})
5. if \( u \in E \) then begin
6. for all the vertices \( x \) in \( V - \{u,v,w,z\} \) do in parallel
7. if \( x \notin T(U) \cup P(U) \cup I(U) \) then
8. some processor \( P(e_i, t(\neq 0)) \) writes a "1" in its own memory;
9. if no "1" was written then \( P(e_i, 0) \) does the following
10. - remembers \( \{u,v,w,z\} \);
11. - flags itself
12. end (if)
13. end (for)
14. end; \{Find-Regular_P4s\}

Fact Procedure Find-Regular_P4s correctly computes the set of all the regular \( P_4 \)s in \( G \) in \( O(\log n) \) EREW time using \( O(\frac{n^2+mn}{\log n}) \) processors.
More terminology...

Assume \( u = v_j \) and \( z = v_k \) with \( j < k \)

- \( u \) is local "loser"
- \( v \) is local "winner"

- Each flagged processor \( P(i) \) writes the identity of the local loser and winner into \( A[i] \) and \( B[i] \), respectively (\( A \) and \( B \) are one-dimensional arrays of \( m \) elements initialized to 0)

- Sort all non-zero entries of \( A \) and \( B \) and remove duplicates: this takes \( O(\log n) \) time using \( O(m) \) processors

- Construct a bitvector \( L \): bit \( i \) of \( L \) is set to 1 iff \( v_i \) is a local loser (This takes \( O(\log n) \) time and \( O(n/\log n) \) processors)
More terminology...

An endpoint $u$ of a regular P4 is called a *global winner* if the bit corresponding to $u$ in $L$ is 0.

To record all the global winners we construct a bitvector $W$ using the information in the array $B$;
(This takes $O(\log n)$ time and $O(n/\log n)$ processors)

$W = W - L$ (this is the set of all global winners)

Every flagged processor corresponding to a global winner is referred to as *essential*.
Procedure Find_Winners_and_Losers(G);
0. begin
2. L←W←0;
3. for every flagged processor P(i) in parallel begin
   4. A[i]← local loser corresponding to e_i;
   5. B[i]← local winner corresponding to e_i;
   end; {for}
       in sorted order with all duplicates removed;
       in sorted order with all duplicates removed;
9. for all i←1 to k do in parallel
   10. set the A[i]-th bit of L to 1;
   11. for all i←1 to l do in parallel
   12. set the B[i]-th bit of W to 1.
13. W←W−L; {find global winners}
14. broadcast W to all the processors P(i);
15. for every flagged processor P(i) in parallel
16. if the local winner of e_i is in W then
17. P(i) does the following:
   18. - remembers that its local winner is a global winner;
   19. - marks itself as "essential"
20. return(L,W)
21. end; {Find_Winners_and_Losers}

Fact Procedure Find_Winners_and_Losers correctly computes the set of all the
global winners and losers in $O(\log n)$ EREW time using $O(\frac{mn}{\log n})$ processors.
Procedure Construct\_SK(G);

0. begin
1. let w_1, w_2, ..., w_p stand for the global winners;
2. for i←1 to p do in parallel
3. if processor P(i) is essential then begin
4. processor P(i) sets to 1 the bit of S_i corresponding to w_i;
5. let P(i1), P(i2), ..., P(i_t) (1≤i≤p) be the essential processors whose local winner is w_i;
6. for j←1 to t_i do in parallel
7. processor P(ij) sets the k-th bit of S_i with v^k standing for its local loser;
8. processor P(i1) broadcasts to P(i2), ..., P(i_t) the identity of the two midpoints it stores;
9. for j←2 to t_i do in parallel
10. processor P(ij) marks the midpoint it stores coinciding with one of the midpoints received;
11. for j←1 to t_i do in parallel
12. processor P(ij) sets to 1 the bit of K_i corresponding to its unmarked midpoint;
13. r_i←K_i\|S_i;
14. if \{N(w_i)\∩K_i=1 then
15. f_i(w_i)← the unique vertex in N(w_i)\∩K_i
16. else
17. f_i(w_i)← the unique vertex in K_i\-N(w_i)
18. end; {if}
19. return(SK(G))
20. end; {Construct\_SK}

To summarize our previous discussion, we state the following result.

Fact Procedure Construct\_SK correctly computes the information in every SK[i] (1≤i≤p) in O(\log n) time using O(\frac{n^2}{\log n}) processors in the EREW-PRAM model. □
Procedure Recognize_P4sparse(G);
{Input: an arbitrary graph G,E) with V=n and E=m;
Output: "yes" or "no" depending on whether or not G is a P₄-sparse graph;}
0. begin
1. Find-Regular_P4s(G);
2. Find-Winners-and-Losers(G);
3. using the information contained in L construct the graph G*;
4. if Cograph(G*) then
   5. return("yes");
6. return("no")
7. end; {Recognize_P4sparse}

Theorem  Procedure Recognize_P4sparse correctly determines whether an
arbitrary graph G=(V,E) with V=n and E=m is a P₄-sparse graph in O(log n)
time using O\left(\frac{n^2+mn}{\log n}\right) processors in the EREW-PRAM model.
Constructing the tree representation of P4-sparse graphs

- $T(G)$, the cotree of the reduced graph $G^*$ is available as a byproduct of $\text{Cograph}(G^*)$

- for convenience we enumerate the maximal regular sets as

$$C_1 = (K_1, S_1, f_1), \ C_2 = (K_2, S_2, f_2), \ \ldots, \ C_p = (K_p, S_p, f_p),$$

- at the end of the successful recognition of a P4-sparse graph $G$, the relevant information about $G$ is stored in the tuple $(T(G), SK(G))$

What is $SK(G)$??
We can think of $SK(G)$ as a 1-dimensional array such that $S[i]$ contains the following information:

- characteristic vectors of $K_i$ and $S_i$
- the identity of the unique vertex $w_i$ in $S_i$ that belongs to $G^*$
- the identity of $f_i(w_i)$
- $r_i = |K_i| = |S_i|$

Let $w, w, ..., w$ be the global winners as recorded in $W$.

To compute $S_i$, every essential processor whose local winner is $w_i$ sets the $j$-th bit of $S_i$, with $j$ standing for its local loser.
To compute $K_i$, we do the following:

- In $O(\log n)$ time identify the subset $P(i_1), P(i_2), ..., P(i_t)$ of essential processors whose local winner is $w[i]$

- Processor $P(i_1)$ broadcasts to $P(i_2), ..., P(i_t)$ the identity of the midpoint it has remembered.

- Every processor $P(i_j)$ marks its own midpoint coinciding with the one received by broadcasting.

- Every processor $P(i_j)$ sets to 1 the bit of $K_i$ corresponding to the unmarked midpoint it stores.
Procedure Parallel_Build_ps_Tree(G);
(Input: a $P_4$-sparse graph represented as $(T(G), SK(G))$
Output: the corresponding ps-tree $T(G)$, rooted at $R$;)
1. begin
2. for every essential processor $P(i)$ do in parallel begin
3. create a 2-node $\beta$;
4. create a 1-node $\gamma$;
5. add $\gamma$ as a child of $\beta$;
6. add $\lambda$ as a child of $\gamma$;
7. if $r_i=2$ then begin
8. add the unique vertex in $S_i-\{w_i\}$ as a child of $\beta$;
9. add $f_i(w_i)$ as a child of $\gamma$
10. end
11. else begin
12. create a 0-node $\alpha$;
13. add $\alpha$ as a child of $\beta$;
14. add all vertices in $S_i-\{w_i\}$ as children of $\alpha$;
15. if $w_i$ is adjacent to $f_i(w_i)$ then
16. add $f_i(w_i)$ as a child of $\gamma$
17. else
18. add all vertices in $K_i-f(\{w_i\})$ as children of $\gamma$
19. end; {if}
20. if $d(\lambda') \neq N(\lambda') \cap K_i + 1$ then
21. add $\beta$ as a child of $\lambda'$
22. else begin
23. add $\beta$ as a child of $p(\lambda')$;
24. delete $\lambda'$
25. end {if}
26. if $d(R)=1$ then $R \leftarrow$ unique child of $R$;
27. return($T(G)$)
28. end; {Build_ps_Tree}

Theorem  Procedure Parallel_Build_ps_Tree correctly constructs the ps-tree of
a $P_4$-sparse graph $G=(V,E)$ with $M=n$ and $E=m$ in $O(\log n)$ EREW time using
$O(\frac{n}{\log n})$ processors.
A Fast Parallel Recognition Algorithm

for

a Class of Tree-representable Graphs

Stephan Olariu

Department of Computer Science
Old Dominion University

Definition
A graph $G$ is $P_4$-sparse if no set of five vertices of $G$ induces more than one $P_4$.

Common metrics for "local density"

- Complete graph (clique)
- Cliques with a "few" edges missing
- No "long" paths allowed
- A "few" long paths allowed

"Long" path

Cographs: a class of graphs containing no $P_4$s

$P_4$-reducible graphs: a class of graphs such that every vertex belongs to at most one $P_4$

Applications: scheduling, computational semantics, pattern recognition etc.

Cographs

Clique

$P_4$-reducible graphs

$P_4$-sparse graphs
Definition
For every graph $G$ consider the graph $C(G)$ returned by the following procedure:

Procedure Greedy($G$):
(Input: an arbitrary graph $G$; Output: a graph $C(G)$)
begin
$C(G) = G$;
while there exists a $P_4$ in $C(G)$ do
pick an arbitrary $P_4$ $uvxy$;
pick $z$ at random in $\{u,x,y\}$;
$C(G) = C(G) - \{z\}$;
return $C(G)$
end; (Greedy)

Theorem
For a graph $G$ with no induced $CS$ the following statements are equivalent:
(i) $G$ is $P_4$-sparse;
(ii) for every induced subgraph $H$ of $G$, $C(H)$ is unique up to isomorphism.

Procedure Build-tree($G$):
(Input: a $P_4$-sparse graph $G=(V,E)$; Output: the $PS$-tree $T(G)$ corresponding to $G$.)
begin
if $|V|=1$ then
return the $T(G)$ consisting of the unique vertex of $G$;
if $G$ ($G$) is disconnected then begin
let $G_1, G_2, \ldots, G_n$ be the components of $G$ ($G$);
let $T_1, T_2, \ldots, T_n$ be the corresponding $PS$-trees rooted at $r_1, r_2, \ldots, r_n$;
return the tree $T(G)$ obtained by adding $r_1, r_2, \ldots, r_n$ as children of a node labelled $0$ ($1$);
end
else begin (now both $G$ and $T$ are connected)
write $G = G_1 \cup G_2$;
let $T_1, T_2$ be the corresponding $PS$-trees rooted at $r_1$ and $r_2$;
return the tree $T(G)$ obtained by adding $r_1, r_2$ as children of a node labelled 2;
end
end (Build-tree)
A set \( C \) of vertices of \( G \) is termed regular if it admits a partition into non-empty, disjoint sets \( K \) and \( S \) satisfying the following conditions:

1. \( K \cup S \), \( S \) stable, \( K \) a clique;
2. every vertex in \( V-C \) belongs to precisely one of the sets:
   - \( T(C) = \{ x \in V-C | x \text{ adjacent to all the vertices in } C \} \);
   - \( I(C) = \{ x \in V-C | x \text{ non-adjacent to all the vertices in } S \} \);
   - \( P(C) = \{ x \in V-C | x \text{ non-adjacent to all the vertices in } K \} \).
3. there exist a bijection \( f:S \rightarrow K \) such that
   - either \( N(x) \cap K = \{ f(x) \} \) for every \( x \in S \),
   - or else \( N(x) \cap K = \{ f(x) \} \) for every \( x \in S \).

\[ \text{Let } G = (V,E) \text{ be an arbitrary graph.} \]

**Fact (Regularity is hereditary)**

Let \( C = (K,S,f) \) be a regular set in \( G \) and let \( Z \) be a subset of \( S \) with \( |Z| < |S| - 2 \). Then \( C' = C - \{ x, f(x) | x \in Z \} \) is regular.

**Fact (Containment)**

Let \( C = (K,S,f) \) be regular. For every pair of distinct \( u, v \) in \( C \) with \( u \neq f(v) \) and \( v \neq f(u) \), the unique \( P_4 \) containing \( u \) and \( v \) belongs to \( C \).

**Fact (Black hole property)**

A regular set is maximal if, and only if, every regular \( P_4 \) containing a vertex in \( C \) is included in \( C \).

**Fact (Separation property)**

Two maximal regular sets coincide whenever they intersect.
Given an arbitrary graph $G$ construct a graph $G'$ as follows:

- remove in every maximal regular set $C = (K,S,f)$ all the vertices in $S$ except for an arbitrary one.

Theorem: For every graph $G$, the graph $G'$ is unique up to isomorphism.

Theorem: For an arbitrary graph $G$ the following statements are equivalent:
(i) $G$ is P4-sparse;
(ii) $G'$ is a cograph.

Algorithm Recognize($G$):

{Input: an arbitrary graph $G$.
Output: "yes" or "no" depending on whether or not $G$ is P4-sparse}

Step 1. Find all maximal regular sets in $G$.
Step 2. Compute $G'$.
Step 3. if $G'$ is a cograph then
    return("yes")
else
    return("no")
Step 4. Stop.

The Algorithm

- The EREW model of computation is assumed;
- $G$ is an arbitrary graph represented by adjacency lists;
- for every vertex $x$, assign one processor to every entry on the adjacency list of $x$;
- the vertices are enumerated as $v_1, v_2, ..., v_n$ in a way that will be explained later;
- sets will be represented by their characteristic vector;
- computing the cardinality of a set takes $O(\log n)$ time using $O(n/\log n)$ processors;
- given sets $S, S'$ of vertices of $G$, computing $S - S'$, $S \cup S'$, $S \cap S'$, as well as testing $S = O$, $S \subseteq S'$ takes $O(\log n)$ time using $O(n/\log n)$ processors;
- to compute $N(x)$ we need $O(\log n)$ time and $O(n/\log n)$ processors.
Processor assignment:

- for every \( x \) of \( G \), every entry on the adjacency list of \( x \) receives one processor;
- every edge \( e_i \) (\( i \in \{1, 2, \ldots, m\} \)) receives \( 1 + \left\lceil \frac{n}{\log n} \right\rceil \) processors

\[ P(e_0, 0), P(e_1, 1), \ldots, P(e_m, m) \]

Note: the total number of processors is bounded by \( O\left(\frac{mn}{\log n}\right) \)

How do we find a regular set?

\[ N_{uv} = N[u] - N[w] \quad N_{uw} = N[w] - N[u] \]

Fact

The edge \( vw \) is the midedge of a regular \( P_4 \) in \( G \) only if \( |N_{vw}| = |N_{wu}| = 1 \) and \( uv \in E \) with \( u, z \) standing for the unique vertex in \( N_{vw} \) and \( N_{uw} \), respectively.

- For every edge \( e = uvw \) the sets \( N_{vw} \) and \( N_{uw} \) can be computed in \( O(\log n) \) time as follows:
  - \( N[v] \) will be broadcast to all \( d_G(v) \) edges incident with \( v \).
  - \( N[w] \) will be broadcast to all \( d_G(w) \) edges incident with \( w \).

More terminology...

Procedure `Find_Reg_P4(G)`:

1. begin
2. for every edge \( e = uvw \) of \( G \) do in parallel begin
3. \( N_{uv} = N[u] - N[w] \)
4. \( N_{uw} = N[w] - N[u] \)
5. if \( |N_{uv}| = 1 \) or \( |N_{uw}| = 1 \) then begin
6. for all \( x \) in \( V \) \( \{u, v, w, x\} \) do in parallel
7. if \( x \notin (U) \cup (V) \cup (U) \cup (V) \) then
8. some processor \( P(x_i, (x_i)) \) writes a "1" in its own memory;
9. if no "1" was written then \( P(x_i, (x_i)) \) does the following:
10. - remembers \( (x_i, x_j, (x_k, (x_l)) \)
11. - flags itself
12. end (if)
13. end (for)
14. end
15. \( \text{Find_Reg_P4s} \)

Fact

Procedure `Find_Reg_P4s` correctly computes the set of all the regular \( P_4 \) in \( G \) in \( O(\log n) \) EREW time using \( O\left(\frac{2^m n}{\log n}\right) \) processors.

- Each flagged processor \( P(i) \) writes the identity of the local loser and winner into \( A[i] \) and \( B[i] \), respectively (\( A \) and \( B \) are one-dimensional arrays of \( n \) elements initialized to 0).
- Sort all non-zero entries of \( A \) and \( B \) and remove duplicates: this takes \( O(\log n) \) time using \( O(\log n) \) processors.
- Construct a bitvector \( L \): bit \( i \) of \( L \) is set to 1 if \( i \) is a local loser (This takes \( O(\log n) \) time and \( O(\log n) \) processors).
More terminology.

- An endpoint \( u \) of a regular P4 is called a global winner if the bit corresponding to \( u \) in \( L \) is 0.

- To record all the global winners we construct a bitvector \( W \) using the information in the array \( B \);
  (This takes \( O(\log n) \) time and \( O(n/\log n) \) processors)
  \( W = W - L \) (this is the set of all global winners).

- Every flagged processor corresponding to a global winner is referred to as essential.

---

Procedure `Construct_SK(G);`

0. begin
1. let \( w_1, w_2, ..., w_{\ell} \) stand for the global winners;
2. for \( i = 1 \) to \( \ell \) in parallel begin
3. if processor \( P(i) \) is essential then begin
4. processor \( P(i) \) sets to 1 the bit of \( S \) corresponding to \( w_i \);
5. let \( P(1), P(2), ..., P(\ell) \) (15+5p) be the essential processors whose local winner is \( w_i \);
6. for \( j = 1 \) to \( k \) do in parallel begin
7. processor \( P(j) \) sets the \( k \)-th bit of \( S \) with \( w_i \) concluding for its local loser;
8. processor \( P(i) \) broadcasts to \( P(2), ..., P(\ell) \)
the identity of the two endpoints it stores;
9. for \( j = 2 \) to \( k \) do in parallel begin
10. processor \( P(j) \) marks the midpoint it stores coinciding with one of the endpoints received;
11. for \( j = 1 \) to \( k \) do in parallel begin
12. processor \( P(j) \) sets to 1 the bit of \( K \) corresponding to its unmarked midpoint;
13. \( r = P(K) \);
14. if \( P(w_i \land K) = 1 \) then
15. \( (w_i) = \) the unique vertex in \( N(w_i \land K) \)
16. else
17. \( (w) = \) the unique vertex in \( K \land N(w_i) \)
18. end; \( (\ell) \);
19. return(SK(G));
20. end; \( (\text{Construct\_SK}) \).

To summarize our previous discussion, we state the following result.

Fact Procedure `Construct_SK` correctly computes the information in every

\( SK(i) \) (15+5p) in \( O(\log n) \) time using \( O(n/\log n) \) processors in the EREW-PRAM model.

---

Procedure `Find_Winners_and_Losers(G);`

0. begin
1. for every flagged processor \( P(i) \) in parallel begin
2. \( \text{let } A(i) = \text{ local winner corresponding to } i; \)
3. \( \text{let } A(i) = \text{ local winner corresponding to } i; \)
4. set \( S(P(i)) = \text{ the non-zero entries of } \Lambda \)
5. in sorted order with all duplicates removed;
6. set \( B(1), B(2), ..., B(\ell) \) be the non-zero entries of \( \Lambda \)
7. in sorted order with all duplicates removed;
8. for all \( i = 1 \) to \( \ell \) in parallel begin
9. set the \( i \)-th bit of \( A(i) \) to 1;
10. set the \( \ell \)-th bit of \( C \) to 1;
11. broadcast \( W \) to all processors \( P(i) \);
12. if the local winner of \( i \) is in \( W \) then
13. \( \text{let } \text{ the processor } P(i) \text{ sets to } \text{ parallel } \text{ set the bit } \text{ corresponding to } \text{ local winner is a global winner;} \)
14. mark itself as "essential";
15. return \( (L, W) \);
16. end; \( (\text{Find\_Winners\_and\_Losers}) \).

---

Procedure `Recognize_P4Sparse(G);`

0. begin
1. \( \text{if } G \text{ is an arbitrary graph } G = (V, E) \text{ with } m_{\text{in}} \text{ and } \text{in} E \text{ is a } P_4 \text{-sparse graph; then} \)
2. return "yes" or "no" depending on whether \( \text{or not } G \text{ is a } P_4 \text{-sparse graph;} \)
3. return \( (\text{yes}); \)
4. return \( (\text{no}); \)
5. end; \( (\text{Recognize\_P4Sparse}) \).

---

Theorem Procedure `Recognize_P4Sparse` correctly determines whether an arbitrary graph \( G = (V, E) \) with \( m_{\text{in}} \text{ and } \text{in} E \text{ is a } P_4 \text{-sparse graph in } O(\log n) \)

\( \text{time using } O(n/\log n) \) processors in the EREW-PRAM model.
Constructing the tree representation of P4-sparse graphs

- \( T(G) \), the cotree of the reduced graph \( G^* \) is available at a byproduct of \( \text{Cograph}(G^*) \)
- for convenience we enumerate the maximal regular sets as
  \[ C_1 = (K_2 S_2 f_p), \quad C_2 = (K_2 S_2 f_p), \quad \ldots \quad C_p = (K_p S_p f_p), \]
- at the end of the successful recognition of a P4-sparse graph \( G \), the relevant information about \( G \) is stored in the tuple \( (T(G),SK(G)) \)

What is \( SK(G) \)?

To compute \( K_1 \) we do the following:

- In \( O(\log n) \) time identify the subset \( P(i_j), P(i_j), \ldots, P(i_j) \) of essential processors whose local winner is \( W[i_j] \)
- Processor \( P(i_j) \) broadcasts to \( P(i_j), \ldots, P(i_j) \) the identity of the midpoint it has remembered
- Every processor \( P(i_j) \) marks its own midpoint coinciding with the one received by broadcasting
- Every processor \( P(i_j) \) sets to 1 the bit of \( K_1 \) corresponding to the unmarked midpoint it stores

We can think of \( SK(G) \) as a 1-dimensional array such that \( S[i] \) contains the following information:

- characteristic vectors of \( K \) and \( S \)
- the identity of the unique vertex \( w_i \) in \( S \) that belongs to \( G^* \)
- the identity of \( f_i(w) \)
- \( r_i = |K_i| = |S_i| \)

Let \( w, w, \ldots, w \) be the global winners as recorded in \( W \)

To compute \( S \), every essential processor whose local winner is \( w_i \) sets the \( j \)-th bit of \( S_j \), with \( j \) standing for its local loser

Procedure Parallel_Build_p4_Tree(G);
(\( G \) is a P4-sparse graph represented as \( (T(G),SK(G)) \))
Output: the corresponding p4-tree \( T(G) \), rooted at \( n \)

begin
1. for every essential processor \( P(i) \) do in parallel begin
2. create a 2-node \( z \);
3. create a 2-node \( \gamma \);
4. add \( \gamma \) as a child of \( z \);
5. add \( z \) as a child of \( \gamma \);
6. if \( r_i \) is then begin
7. add the unique vertex in \( S_i-\{w_i\} \) as a child of \( z \);
8. \( f_i\) as a child of \( \gamma \);
9. end
10. else begin
11. create a 3-node \( a \);
12. add \( a \) as a child of \( z \);
13. add all vertices in \( S_i-\{w_i\} \) as children of \( a \);
14. if \( w \) is adjacent to \( f_i(w) \) then
15. add \( f_i\) as a child of \( \gamma \);
16. else
17. add all vertices in \( K_i-\{w_i\} \) as children of \( \gamma \);
18. end; end (if)
19. add \( f_i\) as a child of \( K_i \);
20. remove \( \gamma \);
21. end;
22. return \( T(G) \);
end

Theorem Procedure Parallel_Build_p4_Tree correctly constructs the p4-tree of a P4-sparse graph \( G=(V,E) \) with \( \log n \) processors.
Vertex-switching reconstruction
and pseudosimilarity

Prof. Mark Ellingham
Department of Mathematics
Vanderbilt University
Recent Results on Vertex-Switching Reconstruction

Mark Ellingham
Vanderbilt University
Defn: Let \( S \) be a set of vertices in graph \( G \). Then \( \overline{S} \) denotes \( V(G) - S \) (vertices not in \( S \)). The \( S \)-switching \( G_S \) of \( G \) is obtained by
\( (1) \) deleting all present edges between \( S \) and \( \overline{S} \)
\( (2) \) adding all absent edges between \( S \) and \( \overline{S} \)

**Ex:**

\[
\begin{array}{c}
\text{G} \\
\text{\overline{S}}
\end{array}
\xrightarrow{}
\begin{array}{c}
\text{G}_S
\end{array}
\]

**Notation:**
\( G_{uv} = G_{v\overline{v}} \), a vertex-switching
\( G_{uv} = (G_{u\overline{v}})_{uv} \)

**Notes:**
\( G_{\overline{S}} = G_S \), \( G_{uv} = G \), \( G_{uv} = G_{vu} \).

Defn: The vertex-switching deck \( D_{VS}(G) \) is the collection of vertex-switchings of \( G \).

**Ex:**

\[
\begin{array}{c}
\text{1} \\
\text{2}
\end{array}
\xrightarrow{}
\begin{array}{c}
\text{G}_1 \\
\text{G}_2 \\
\text{G}_3 \\
\text{G}_4
\end{array}
\]

**VS deck of \( G \)**
(no labels on vertices)
**Defn:** H is a VS-reconstruction of G if \( \text{Dys}(H) = \text{Dys}(G) \).

G is VS reconstructible if every VS reconstruction of G is isomorphic to G.

**Ex:** A non-VS reconstructible pair:

\[
\begin{array}{c}
\text{G} \\
\end{array} \rightarrow \begin{array}{c}
\text{H}
\end{array}
\]

VS deck of both

Several other such pairs on 4 vertices exist.

**VS Reconstruction Conjecture** (Stanley 1985):

Any graph with \( n \neq 4 \) vertices is VS reconstructible.

**Theorem** (Stanley 1985): An \( n \)-vertex graph with \( n \) not divisible by 4 is VS reconstructible.

**Open Question:** What about graphs with 12, 16, 20, ... vertices? (For 8 vertices, conjecture true by computer testing.)
Alternative proof (Krasikov & Roditty)
- uses counting methods.

Notation: \( X_k(G \to F) = \) number of \( k \)-switchings of \( G \)
(i.e. \( G/s, |s| = k \)) which are isomorphic to \( F \).

Examine \( k \)-switchings of cards in VS deck, i.e.
\( k \)-switchings of 1-switchings of \( G \). Two cases:

1. \[
\begin{array}{c}
\text{switch} \\
\downarrow \ 
\end{array}
\]
   \[
\begin{array}{c}
|s| = k \\
\uparrow \\
\text{switch}
\end{array}
\]
   \[
\begin{array}{c}
S \\
\uparrow \ 
\end{array}
\]
   \[
\begin{array}{c}
S_{\text{inv}}
\end{array}
\]

\((k+1)\)-switching

2. \[
\begin{array}{c}
\text{switch} \\
\downarrow \ 
\end{array}
\]
   \[
\begin{array}{c}
|s| = k \\
\uparrow \\
\text{switch}
\end{array}
\]
   \[
\begin{array}{c}
S \\
\uparrow \ 
\end{array}
\]
   \[
\begin{array}{c}
S_{\text{linv}}
\end{array}
\]

\((k-1)\)-switching

In fact
\[
\sum_{J \in \mathcal{D}_V(G)} X_k(J \to F) = (k+1)X_{k+1}(G \to F) + (n-k+1)X_{k-1}(G \to F)
\]

But if \( \mathcal{D}_V(G) = \mathcal{D}_V(H) \) same equation holds
with \( G \) replaced by \( H \), so if, for given \( F \), we
define
\[
\delta_k = X_k(G \to F) - X_k(H \to F)
\]
we get
\[
(k+1)\delta_{k+1} + (n-k+1)\delta_{k-1} = 0
\]
(subtract \( \Delta \) for \( H \) from \( \Delta \) for \( G \)).
In particular, if \( F = G \) we have 
\[
\delta_k = X_k(G \rightarrow G) - X_k(H \rightarrow G)
\]
and we get
\[
(k+1)\delta_{k+1} + (n-k+1)\delta_k = 0
\]
where, if \( C \neq H \),
\[
\delta_0 = 1 - 0 = 1
\]
\[
\delta_1 = 0 \quad \text{since } G, H \text{ have same}
\]
\[
\text{VS deck, i.e. same l-switchings.}
\]
Solving (8) with initial conditions (2) gives
\[
\delta_{2i} = (-1)^i \left( \frac{n}{2} \right)^i
\]
\[
\delta_{2i+1} = 0
\]
But also, since \( G_S = G \bar{S} \), must have \( \delta_k = \delta_{n-k} \).
However,
- if \( n \) odd \( \delta_0 = 1 \neq \delta_n = 0 \)
- if \( n \equiv 2 \pmod{4} \) \( \delta_0 = 1 \neq \delta_n = -1 \)
So we can only have \( \text{DvS } (G) = \text{DvS } (H) \) but 
\( G \neq H \) if \( n \equiv 0 \pmod{4} \).

Important: For non-VS reconstructible \( G \), (3) says that \( \delta_k > 0 \), implying that \( X_k(G \rightarrow G) > 0 \), if \( k \) is divisible by 4. Thus for any \( k \) divisible by 4, \( 0 \leq k \leq n \), \( G \) has \( k \)-switching \( G_S \equiv G \).
Reconstructing subgraph numbers from VS deck:
(Stanley, for edges. ME & Royle / Krasikov & Roditty in general)

Defn: An induced subgraph of graph G contains all edges incident with a given vertex set.

Ex: $\square$ induced $\not\square$ not induced

Let $i(F,G)$ be number of induced subgraphs of G which are isomorphic to F. Want to find $i(F,G)$ from VS deck.

How can F occur in VS deck? Two cases:

1. \[ F \quad \text{switch} \quad G \quad \rightarrow \quad F \quad G_v \]

2. \[ J \quad \text{switch} \quad G \quad \rightarrow \quad F=J_v \quad G_v \]
So for \( p \)-vertex \( F \) get equation

\[
\Sigma \hat{i}(F,G) = (n-p) \hat{i}(F,G) + \Sigma \ x_i (J\rightarrow F) \hat{i}(J,G) \\
\text{C-Edges}(G)
\]

By taking all such equations for switching class of graphs, get system of linear equations, try to solve for \( \hat{i}(F,G) \)'s.

Ex: Switching class \{ \( \triangle \) \( C_3 \), \( \circ \) \( \overline{P}_3 \) \}

Get equations

\( C_3 \)'s in deck = \( (n-3) \hat{i}(C_3,G) + \hat{i}(\overline{P}_3,G) \)

\( \overline{P}_3 \)'s in deck = \( (n-3) \hat{i}(\overline{P}_3,G) + 3\hat{i}(C_3,G) + 2\hat{i}(\overline{P}_3,G) \)

i.e.

\[
\begin{pmatrix}
(n-3) & 1 \\
3 & n-1
\end{pmatrix}
\begin{pmatrix}
\hat{i}(C_3,G) \\
\hat{i}(\overline{P}_3,G)
\end{pmatrix}
= \begin{pmatrix}
C_3 \text{'}s \text{ in deck} \\
\overline{P}_3 \text{'}s \text{ in deck}
\end{pmatrix}
\]

Can solve provided \((n-3)(n-1)-3 \neq 0\), i.e. \( n \neq 0, 4 \).

Theorem: For \( n \)-vertex \( G \) with \( n \equiv 0 \mod 4 \) and \( p \)-vertex \( F \), can VS reconstruct \( \hat{i}(F,G) \)

if \( n > 2p \).

Corollary: For \( n \neq 4 \) can VS reconstruct number of edges and vertex degrees.
VS. Reconstruction by structural means

Assume $Dvs(G) = Dvs(H)$, $G \neq H$.

Lemma (Krasikov & Roditty): For every vertex $v$ of $G$
there exists $u$ such that
(i) $G vu \cong H$;
(ii) $\{v, u\}$ joined by exactly $n-2$ edges to $\{v, u\}$;
(iii) $v$ and $u$ have a common neighbour in $G$.

Proof: (i)

Now $G vu = (G v) u \cong C u' \cong (H u'') u'' = H$.

(ii) $G vu \cong H$ has exactly same number of
edges as $G$. So $G$ has exactly half of
possible $2(n-2)$ edges from $\{v, u\}$ to $\{v, u\}$.

(iii) If not:

But then $G vu \cong G$, so $H \cong G$, contradiction.
VS Reconstruction Results for \( n \in \mathbb{C} \) (mod 1)

1. disconnected graphs (Krasikov)
   Used structural lemma.

2. graphs with \( n \left( \frac{n-1}{\Delta} \right) < 2^{n/2-2} \) (\( \Delta = \) maximum degree) (Krasikov)
   Used \( \delta_{2j} > 0 \), counting arguments.

3. regular graphs (ME & Royle)
   Used structural arguments. Simple, but harder than for vertex deletion reconstruction.

4. triangle-free graphs (ME & Royle)
   Recognition: used fact that \( \varepsilon(C_3, G) \) is reconstructible.
   Reconstruction: used structural lemma, regular graphs result.
Vertex switching pseudosimilarity

Defn: $u, v$ similar if some automorphism maps $u$ to $v$

quasisimilar if $G-u \cong G-v$

pseudosimilar if quasisimilar, not similar

Theorem (Godsil & Kocay): All pairs of quasi-
similar vertices in finite graphs arise from
following construction:

delete to \[ \rightarrow \]

get $G$

\[ H: \quad u \quad \overset{\text{automorphism, } (\sigma)}{\downarrow} \quad \overset{\text{(green vertices}}{\downarrow} \quad \text{are an orbit}}{\downarrow} \quad \text{of } (\sigma) \]

Ex:

\[ \text{deleted} \]

Harary & Palmer's original example

Defn: $u, v$ VS quasisimilar if $G_u \cong G_v$

VS pseudosimilar if VS quasisimilar, not similar
Theorem (ME): For finite graphs, all occurrences of US quasisimilar vertices arise from two constructions:

(i) analogous to Godsil & Kocay's construction

Ex:

\[ u \rightarrow \text{switch } w \rightarrow u \]

\[ \begin{array}{c}
\text{u, v VS pseudosimilar}\n\end{array} \]

(ii) funny construction involving switching alternate vertices along orbits of automorphism \( \Theta \) of graph \( H \).

Note: Proof of Theorem involved characterising all situations where \( G_S \cong G \) for some set of vertices \( S \). Used this because if \( G_u \cong G_v \) then \( (G_u)_{S \setminus uv} = G_v \cong G_u \).

Characterisation of when \( G_S \cong G \) has other possible implications. Know that for nonrecognizable \( G \), must be sets \( S \) of size divisible by 4 with \( G_S \cong G \).
Recent Results on Vertex-Switching Reconstruction

Mark Ellingham
Vanderbilt University

- Define $H$ is a VS-reconstruction of $G$ if $D_{VS}(H) = D_V(G)$.
- $G$ is VS reconstructible if every VS reconstruction of $G$ is isomorphic to $G$.

Ex: A non-VS reconstructible pair:

Ex: The vertex-switching deck $D_{VS}(G)$ is the collection of vertex-switchings of $G$:

VS deck of $G$ (no labels on vertices)

- Define: Let $S$ be a set of vertices in graph $G$. Then $G^S$ denotes $V(G) - S$ (vertices not in $S$).
- The $S$-switching $G^S$ of $G$ is obtained by:
  1. Deleting all present edges between $S$ and $\overline{S}$
  2. Adding all absent edges between $S$ and $\overline{S}$

Ex:

Notation: $G_{uv} = G_{1,uv}$, a vertex-switching
$G_{uv} = (G_{uv})_{v}$

Notes: $G^S = G_S$, $G_{uv} = G$, $G_{uv} = G_{uv}$.

- Defn: The vertex-switching deck $D_{VS}(G)$ is the collection of vertex-switchings of $G$.

Ex:

Alternative proof (Krukowski & Radzio):
- Use counting methods.

Notation: $X_k(G+F)$ a number of $k$-switchings of $G$
(ie. $G_k$, $G_{kx}$) which are isomorphic to $F$.

Examine $k$-switchings of cards in VS deck, i.e.
$k$-switchings of $k$-switchings of $G$.

Two cases:

In fact:

But if $D_{VS}(H) = D_{VS}(G)$, same equation holds
with $G$ replaced by $H$, so if, for given $F$, we define
$S_{uv} = X_k(G_{uv} + (G_{uv} - G_{uv}))$
we get:

Open Question: What about graphs with 12, 14, 20, ... vertices? (For 8 vertices, conjecture true by computer testing.)
In particular, if \( F \subseteq G \) we have
\[
\Delta_5 = X_5(C \rightarrow C) - X_5(\emptyset \rightarrow C)
\]
and we get
\[
(\text{(1)}) \quad \Delta_{5+1} + (n-4)\Delta_5 = 0
\]
where, if \( C \neq \emptyset \),
\[
\Delta_5 = 1 - O = n \\
\Delta_1 = 0 \\
\Delta_0 = 0 \\
\text{since } C, H \text{ have some } \text{ VS deck, i.e. same 1-swings}
\]
Solving (1) with initial conditions (2) gives
\[
\Delta_{5+1} + (n-4)\Delta_5 = 0
\]
\[
\Delta_{2n-1} = 0
\]
but also, since \( C_G = C_H \), must have \( \Delta_5 = \Delta_1 = 0 \).

However,
- if \( n \) is odd \( \Delta_5 = 1 + \Delta_0 = 0 \)
- if \( n \) is even \( \Delta_5 = 1 + \Delta_0 = 0 \)

So we can only have \( D_{3+1}(C) = D_{3+1}(H) \) but \( C \neq H \) if \( n \) is odd.

Importantly for non-VS reconstructable \( C \), (1) says that \( \Delta_5 = 0 \), implying that \( X_5(C \rightarrow C) = 0 \), if \( k \) is divisible by 4. Thus for any \( k \) divisible by \( 4, o(x) = 4n \), \( C \) has k-swinging \( \Delta_5 = 0 \).

So far our rule \( F \) get equation
\[
\Sigma \mathcal{I}(F, C) = (n-p) \mathcal{I}(F, G) + \Sigma \mathcal{I}(X(F), C) = \Sigma \mathcal{I}(X(F), G)
\]
By taking all such equations for switching class of graphs, get system of linear equations, try to solve for \( \mathcal{I}(F, C) \).

Ex: Switching class \( \{ (G_1, 1), (G_2, 2) \} \)

Each equations
- \( G \)'s in deck : \( (n-3) \mathcal{I}(G_1, C) + \mathcal{I}(G_2, C) \)
- \( F \)'s in deck : \( (n-3) \mathcal{I}(F_1, C) + \mathcal{I}(F_2, C) \)

i.e.
\[
\begin{align*}
(n-3) & \quad (G_1) + (G_2) \\
3 & \quad (F_1) + (F_2)
\end{align*}
\]
Can solve provided \( (n-3)(n-1) - 4 \) is. i.e. \( n \neq 4, 4 \).

Theorem: For \( n \)-vertex \( G \) with \( n \) odd (mod 4) and \( p \)-vertex \( F \), can VS reconstruct \( \mathcal{I}(F, G) \) if \( n > 2p \).

Corollary: For \( n \geq 4 \) can VS reconstruct number of edges and vertex degrees.

VS Reconstruction by structural means
Assume \( D_{3+1}(C) = D_{3+1}(H) \), \( C \neq H \).

Lemma (Kreinov & Raudenbush): For every vertex \( v \) of \( C \), there exists \( u \) such that
- (1) \( C \) and \( H \) have exactly \( n-2 \) edges to \( \{x,v\} \).
- (2) \( v \) and \( u \) have a common neighbour \( K \).

Proof: (1) \( C \)

(2) \( C \) and \( H \) have exactly \( n-2 \) edges to \( \{x,v\} \).

(3) If not \( C \) and \( H \) have exactly \( n-2 \) edges to \( \{x,v\} \).

But then \( C \) and \( H \) have exactly \( n-2 \) common neighbours.
**VS Reconstruction Results for nec. (n-1) k**

- disconnected graphs (Krolikov)
  Used structural lemma.
- graphs with $n \left(\frac{n}{2}\right) < 2^{n-2}$ ($d =$ maximum degree) (Krolikov)
  Used $\delta_2 > 0$, counting arguments.
- regular graphs (ME & Boyle)
  Used structural arguments. Simple, but harder than for vertex deletion reconstruction.
- triangle-free graphs (ME & Boyle)
  Recognition: used fact that $\Delta(G, C)$ is reconstructible.
  Reconstruction: used structural lemma, regular graphs result.

---

**Vertex switching pseudosimilarity**

**Defn:** $u,v$ similar if some automorphism maps $u$ to $v$.

**Quasisimilar if $G_u \cong G_v$**

**Pseudosimilar if $G_u \cong G_v$, not similar**

**Theorem (Cedgil & Kocay):** All pairs of quasisimilar vertices in finite graphs arise from following construction:

- $u,v \in G$
- $\theta \in \text{Aut}(G)$
- $u,v \in \text{orbit of } \theta$

*Ex:*

- Original example

**Defn:** $u,v$ VS quasisimilar if $G_u \cong G_v$

**VS pseudosimilar if $G_u \cong G_v$, not similar**

---

**Theorem (ME):** For k-regular graphs, all occurrences of VS quasisimilar vertices arise from two constructions:

1. Analogous to Cedgil & Kocay's construction

*Ex:*

2. Shiny construction involving switching alternate vertices along orbits of automorphism $\theta$ of graph $H$.

**Note:** Proof of Theorem involves characterizing all situations where $G_1 \cong G_2$ for some set of vertices $S$. Used HS because if $G_u \cong G_v$, then $(G_u)_|S = G_v \cong G_u$, $S$.

Characterization of when $G_1 \cong G_2$ has other possible implications. Know that for non-constructible $G$, must be sets $S$ of size divisible by 4 with $G_u \cong G_v$. 