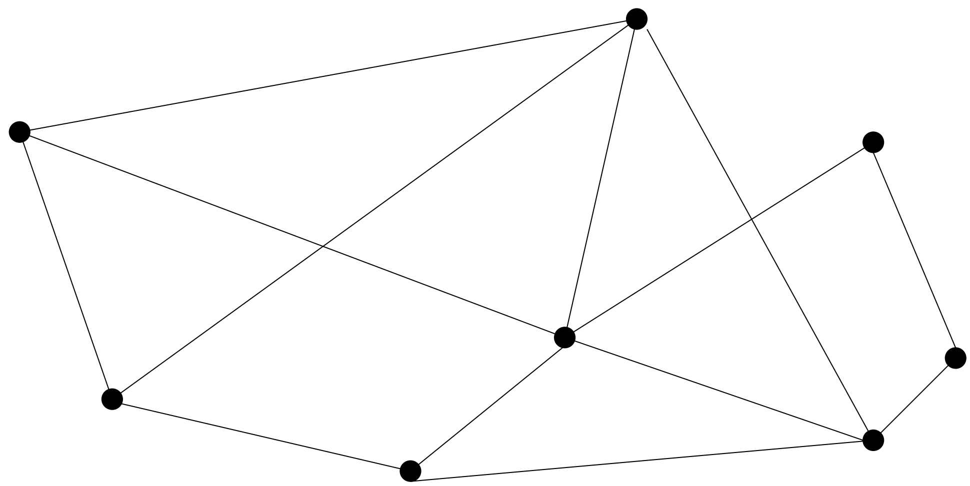
Distributed Deterministic Graph Coloring

Michael Elkin

Ben-Gurion University

The Model



- Unweighted undirected graph $G = (V, E)$.

- Vertices host processors.

- Processors communicate over edges of G.

- Communication is synchronous, i.e., occurs in *discrete* rounds.

- Running time $=$ $\#$ rounds.
- All vertices wake up simultaneously.
- Vertices have unique Ids from $\{1, 2, \ldots, n\} = [n]$.
- Arbitrarily large messages are allowed, though short (of size $O(\log n)$) are preferred.

Coloring

- $\Delta = \Delta(G)$ - maximum degree of a vertex in G.

- $\varphi : V \rightarrow [k]$ is a k-*coloring* if $\forall e = (u,w) \in E, \quad \varphi(u) \neq \varphi(w)$.

- In distributed setting, traditionally $k \geq \Delta + 1$.

- MIS U:
  (1) $\forall v, w \in U, \quad (v,w) \notin E$.
  (2) $\forall v \notin U, \quad \exists u \in U$ s.t. $(u,v) \in E$.

- $(\Delta + 1)$-coloring in $O(n)$ rounds is easy.

  Color vertices one-by-one:
  For each new vertex v there are
  $\leq \Delta$ forbidden colors.

$$\Downarrow$$

  There is always an available color
  for v in $[\Delta + 1]$.

- MIS in $O(n)$ rounds is easy too.

  Initialize $U \leftarrow \emptyset$;
  Treat vertices one-by-one:
  For each new vertex v do:
  if $\Gamma(v) \cap U = \emptyset$ then
      v joins U;

The Relation between Coloring and MIS

Observation: Given α-coloring φ an MIS U can be computed in α rounds.

Initialize $U = \{v \mid \varphi(v) = 1\}$;

On round 2 each vertex v with $\varphi(v) = 2$ in parallel does:
if $\Gamma(v) \cap U = \emptyset$ then
  v joins U;

// $\{v \mid \varphi(v) = 2\}$ is an independent set

On round 3 each vertex v with $\varphi(v) = 3$ does the same, etc.

Vertices are only added to U.

A vertex v that decides not to join U,
is dominated by U at that time,
and stays dominated it.

Hence U is maximal.

THM: [Linial,87] (The opposite direction)
Given an algorithm that computes MIS,
$(\Delta + 1)$-coloring can be computed
within the same time (using larger messages).

Distributed Coloring - Known Randomized Results

- $(\Delta+1)$-coloring and MIS in $O(\log n)$ time. [Luby,86], [Alon,Babai,Itai,86].

  $(\Delta+1)$-coloring in $O(\log\Delta+\sqrt{\log n})$ time. [Schneider,Wattenhofer,PODC'10].

- $O(\Delta)$-coloring in $O(\sqrt{\log n})$ time [Kothapalli,Scheideler,Onus, Schindelhauer,06].

- $O(\Delta+\log n)$-coloring in $O(\log\log n)$ time, and $O(\Delta\log^{(c)} n+\log^{1+1/c} n)$-coloring in $O(f(c))=O(1)$ time. [Schneider,Wattenhofer,PODC'10].

Lower Bounds

- $f(\Delta)$-coloring requires $\frac{1}{2}\log^* n$ time.

  [Linial,87]

- Coloring Δ-regular trees in $o(\sqrt{\Delta})$ colors requires $\omega(\log_\Delta n)$ time.

  [Linial,87]

- $\Omega(\frac{\log \Delta}{\log\log \Delta})$ and $\Omega(\sqrt{\frac{\log n}{\log\log n}})$ time is required for maximal independent set.

  [Kuhn,Moscibroda,Wattenhofer,04]

Known Deterministic Results

- 3-coloring and MIS for an n-path (or an n-cycle) in $\log^* n$ time. [Cole,Vishkin,86]

- $(\Delta + 1)$-coloring and MIS in $2^{O(\Delta)} + \log^* n$ time. [Goldberg,Plotkin,86]

- $(\Delta + 1)$-coloring and MIS in $O(\Delta^2 + \log^* n)$ time, and in $O(\Delta \log n)$ time. [Goldberg,Plotkin,Shannon,87]

- $O(\Delta^2)$-coloring in $\log^* n + O(1)$ time. [Linial,87]

  Asked: can one get much fewer than Δ^2 colors in time polylogarithmic in n?

- $(\Delta + 1)$-coloring and MIS
  in $2^{O(\sqrt{\log n \log\log n})}$ time. (Large messages)
  [Awerbuch,Goldberg,Luby,Plotkin,89]
  Improved to $2^{O(\sqrt{\log n})}$ time.
  [Panconesi,Srinivasan,92]

- $(\Delta + 1)$-coloring and MIS
  in $O(\Delta \log \Delta) + \log^* n$ time.
  [Kuhn,Wattenhofer,06]

- $(\Delta + 1)$-coloring and MIS
  in $O(\Delta) + \frac{1}{2}\log^* n$ time.
  And $\forall t \leq \Delta + 1$,
  $(\Delta + 1) \cdot t$-coloring in $O(\Delta/t) + \frac{1}{2}\log^* n$ time.

  [Barenboim,Elkin, ArXiv'08,STOC'09],
  [Kuhn, SPAA'09]

- $\Delta^{1+\eta}$-coloring in $O(\log \Delta \cdot \log n)$ time, for any $\eta > 0$.

  $\Delta^{1+o(1)}$-coloring in $O(f(\Delta) \log \Delta \cdot \log n)$ time, for any $f(\Delta) = \omega(1)$.

  $O(\Delta)$-coloring in $O(\Delta^{\epsilon} \cdot \log n)$ time, for any $\epsilon > 0$.

  [Barenboim,Elkin, PODC'10]

  Answers Linial's question in the affirmative.

  (In polylogarithmic time one can get $\Delta \cdot 2^{O(\log \Delta / \log \log \Delta)}$-coloring.)

Special Families of Graphs

- **Planar graphs:**
  $(\Delta + 1)$-coloring and MIS in $O(\log n)$ time
  [Goldberg,Plotkin,Shannon,87]

  Improved to $O(\frac{\log n}{\log\log n})$ time.
  [Barenboim,Elkin,08]

- **Graphs of bounded growth:**
  In each locality independent sets are small.
  ($\forall v$, largest independent set
  in $\Gamma_2(v)$ has size $O(1)$.)

  This family includes Unit Disk Graphs, and more generally, Unit Ball Graphs, where the underlying metric has fixed *doubling dimension*.

  $(\Delta + 1)$-coloring and MIS in $O(\log^* n)$ time.
  [Schneider,Wattenhofer,08].
  Tight (in terms of n).

Graphs of Bounded Arboricity

aka k-*inductive* or k-*degenerate* graphs

Def: For $G = (V, E)$,

$$a(G) = \max\{\lceil \frac{|E(U)|}{|U| - 1} \rceil : U \subseteq V, |U| \geq 2\}.$$

Nash-Williams Thm: (1961)
$a(G)$ is the minimum number of edge-disjoint forests $F_1, F_2, \ldots, F_a$ such that $E = \bigcup_{i=1}^{a} F_i$, $(F_i \cap F_j = \emptyset \quad \forall i \neq j)$.

Example:
A tree T has $a(T) = 1$.
A cycle C has $a(C) = 2$.
Planar graphs have arboricity ≤ 3.

Very general family of graphs.
It includes graphs of bounded genus,
graphs of bounded treewidth,
graphs that exclude any fixed minor.

Coloring Graphs of Bounded Arboricity

Each graph with arboricity a is $2a$-colorable. Tight.

Distributedly *deterministically* one can get a

- $(2a+1)$-coloring in $O(a^2 \cdot \log n)$ time.
- $(2+\eta)a$-coloring in $O(a \cdot \log n)$ time, $\forall \eta > 0$.
- $O(t \cdot a)$-coloring in $O(a \cdot (\frac{\log n}{t} + 1))$ time, for any parameter t, $1 \leq t \leq a$.
- $O(q \cdot a^2)$-coloring in $O(\frac{\log n}{\log q} + \log^* n)$ time, for any parameter q.

[Barenboim,Elkin,08]

In particular, for $a \le \log^{1/2-\epsilon} n$,
(e.g., bounded genus, bounded tree-width, etc.)
$(\Delta + 1)$-coloring and MIS in $O(\frac{\log n}{\log \log n})$ time.

A lower bound:
$O(q \cdot a^2)$-coloring requires $\Omega(\frac{\log n}{\log a + \log q})$ time.

[BE08], based on [Linial,87]

More recent upper bounds [BE10]:

$a^{1+\eta}$-coloring in $O(\log a \cdot \log n)$ time.

$$\Downarrow$$

$(\Delta + 1)$-coloring in $O(\log \Delta \cdot \log n)$ time,
whenever $a \le \Delta^{1-\eta}$, for some $\eta > 0$.

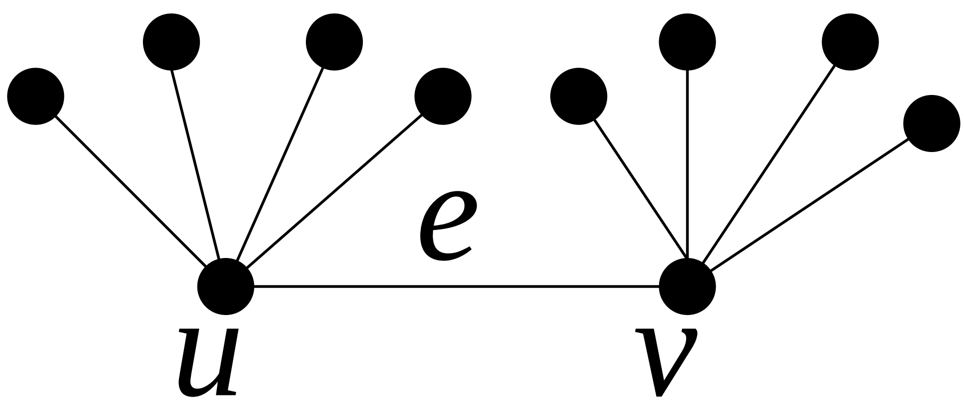
Also, $O(a)$-coloring in $O(a^{\epsilon} \cdot \log n)$ time.

Line Graphs (Edge-Coloring)

- Line graph $L(G)$ of G:
  egdes of G are vertices of $L(G)$,
  and two vertices $v(e_1), v(e_2)$ of $L(G)$ are adjacent if e_1, e_2 are incident in G.

  $\Delta(L(G)) \leq 2(\Delta(G) - 1)$.

  $2(\Delta - 1) + 1 = (2\Delta - 1)$-edge-coloring is "analogous" to $(\Delta + 1)$-coloring.



- $\chi(L(G)) \leq \Delta + 1$
  [Vizing,64]

- Deterministic $(2\Delta - 1)$-edge-coloring in $O(\Delta + \log^* n)$ time. [Panconesi,Rizzi,01]

- Deterministic $O(\Delta \log n)$-edge-coloring in $O(\log^4 n)$ time. [Czygrinow,Hanckowiak,Karonski,01]

- Randomized $(1 + \epsilon)\Delta$-coloring in $O(\log n)$ time, as long as $\Delta = \omega(\log n)$. [Dubhashi,Grable,Panconesi,98], [Panconesi,Srinivasan,97]

- Deterministic maximal matching (i.e., MIS in line graphs) in $O(\log^4 n)$ time. [Hanckowiak,Karonski,Panconesi,99]

Recent Results for Line Graphs

- Deterministic $\Delta^{1+\eta}$-edge-coloring in $O(\log \Delta)$ time, for any $\eta > 0$.

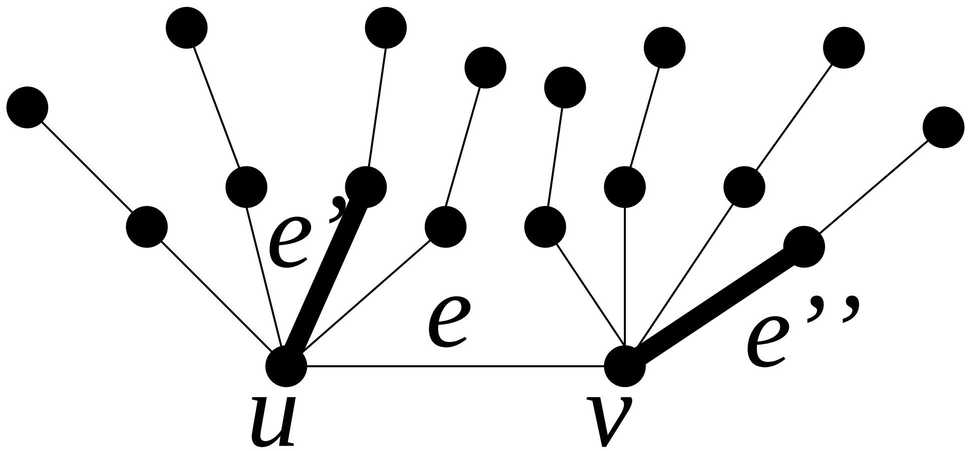
- Deterministic $\Delta^{1+o(1)}$-edge-coloring in $O(f(\Delta) \log \Delta)$ time, for any $f(\Delta) = \omega(1)$.

- Deterministic $O(\Delta)$-edge-coloring in $O(\Delta^{\epsilon})$ time, for any $\epsilon > 0$.

[Barenboim,Elkin,10]

Applies also to vertex-coloring graphs of bounded *neighborhood independence*:

$\forall v$, largest independent set in $\Gamma_1(v)$ has size $O(1)$.



Neighborhood independence of a line graph is ≤ 2.

But in a line graph, $\Gamma_2(e)$ may contain $\Omega(\Delta)$ independent vertices. (I.e., a line graph might not have bounded growth.)

$(\Delta + 1)$-Coloring

A cornerstone:
$O(\Delta^2)$-coloring in $\log^* n$ time.
[Linial,87]

Based on:

THM: $\forall n', r, \quad n' > r$,
$\exists$ family $\mathcal{Q} = \mathcal{Q}(n', r)$ of n' subsets
of $\{1, 2, \ldots, 5r^2 \cdot \log n'\}$ s.t.
$\forall r + 1$ sets $Q_0, Q_1, \ldots, Q_r \in \mathcal{Q}$,

$$Q_0 \not\subseteq \bigcup_{i=1}^{r} Q_i \ .$$

[Erdős,Frankl,Furedi,85]

Linial's Algorithm

- Each vertex v sets
  $\varphi(v) \leftarrow Id(v)$; $n' \leftarrow n$; $r \leftarrow \Delta$.

- Each v computes (locally)
  the set system $\mathcal{Q} = \mathcal{Q}(n, \Delta)$.

- Each v sends $\varphi(v)$ to all its neighbors.

- Given the colors (Ids) $\varphi(u_1), \varphi(u_2), \ldots, \varphi(u_\Delta)$, v finds an element $q \in Q_{\varphi(v)} \setminus \bigcup_{i=1}^{\Delta} Q_{\varphi(u_i)}$.

  (Exists by EFF's thm.)

- Set $\varphi(v) \leftarrow q$.

We get $5\Delta^2 \cdot \log n$-coloring in one single round.

The coloring is legal, because for $e = (u, v)$, $q = \varphi(v) \in Q_{\varphi(v)} \setminus Q_{\varphi(u)}$, and $q' = \varphi(u) \in Q_{\varphi(u)}$.

Iterate for $\log^* n$ rounds to get $O(\Delta^2)$-coloring.

Color Reduction

We have a $(\Delta + 1)^2$-coloring φ,
computed in $\log^* n$ time.

Eliminate one color per round,
for $(\Delta + 1)^2 - (\Delta + 1)$ rounds,
to get a $(\Delta + 1)$-coloring.

Observation:
Each φ-color class is an independent set.

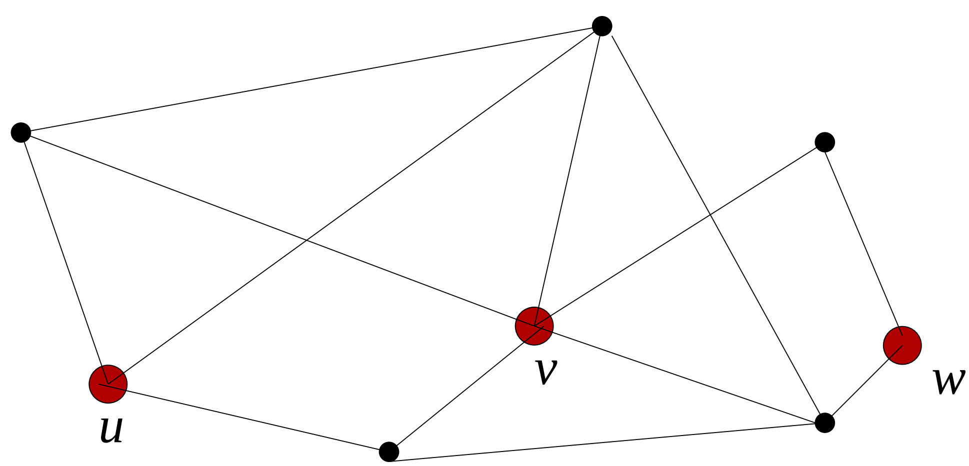
For each vertex v with $\varphi(v) = (\Delta + 1)^2$,
there is an available color in $[\Delta + 1]$.

Color Reduction (Cont.)

All vertices v with $\varphi(v) = (\Delta + 1)^2$
select in parallel
an available color $i \in [\Delta + 1]$.

Then do it for vertices v
with $\varphi(v) = (\Delta + 1)^2 - 1$, etc.,
until $\varphi(v) = \Delta + 2$.

Correct, since each time
we recolor an independent set.



Kuhn-Wattenhofer's Algorithm

$(\Delta + 1)$-coloring in $O(\Delta \log \Delta) + \log^* n$ time.

- Invoke Linial's algorithm to compute $(\Delta + 1)^2$-coloring φ in $\log^* n$ time.

- $\forall i \in [\Delta + 1]$, let

$$U_i = \{v \mid (i-1) \cdot (\Delta + 1) + 1 \leq \varphi(v) \leq i \cdot (\Delta + 1)\}.$$

- Pair subgraphs $G(U_1)$ with $G(U_2)$, $G(U_3)$ with $G(U_4)$,..., $G(U_\Delta)$ with $G(U_{\Delta+1})$.

  Consider $G(U_1 \cup U_2)$.
  It is $2 \cdot (\Delta + 1)$-colored by φ.

- Reduce the $2(\Delta + 1)$-coloring of $G(U_1 \cup U_2)$ to get a $(\Delta + 1)$-coloring of $G(U_1 \cup U_2)$ in $2(\Delta + 1) - (\Delta + 1) = \Delta + 1$ rounds.

  In parallel, reduce the colorings of $G(U_3 \cup U_4), G(U_5 \cup U_6), \ldots$

$$\Downarrow$$

  In $\Delta + 1$ rounds we get $\frac{1}{2}(\Delta + 1)^2$-coloring of G.

- Redefine $U_1, U_2, \ldots, U_{(\Delta+1)/2}$ in terms of the new $\frac{1}{2}(\Delta+1)^2$-coloring. Reduce the number of colors again.

$$\Downarrow$$

Overall, $\log(\Delta+1)$ phases, each requires $(\Delta+1)$ rounds. That is, $(\Delta+1)\log(\Delta+1)$ time.

Number of colors reduces from $(\Delta+1)^2$ to $\frac{(\Delta+1)^2}{2}$, to $\frac{(\Delta+1)^2}{2^2}, \ldots, \frac{(\Delta+1)^2}{2^{\log(\Delta+1)}} = \Delta+1$.

- The KW iterative procedure requires $(\Delta+1)\log(\Delta+1)$ time.

  We also spent $\log^* n$ rounds for Linial's algorithm.

  The total time of the KW algorithm is $O(\Delta \log \Delta + \log^* n)$ (it computes $(\Delta+1)$-coloring).

$(\Delta + 1)$-Coloring in $O(\Delta) + \log^* n$ Time

Def: (Defective Coloring)
[Burr,Jacobson],[Harary,Jones]
[Cowen,Cowen,Woodall] 85-86

Def: The *defect* of a vertex v wrt coloring φ is the number of neighbors $u \in \Gamma(v)$ with $\varphi(u) = \varphi(v)$.

Def: The *defect* d of a k-coloring φ is the maximum defect of a vertex wrt φ.
φ is called a d-*defective* k-*coloring*.

Thm: [Lovasz,66]
$\forall G$, $\forall p$ there exists a $\lfloor \Delta/p \rfloor$-defective p-coloring of G.

Thm: [Barenboim,Elkin, ArXiv'08,STOC'09]
$\forall G,\ \forall p\ \lfloor \Delta/p \rfloor$-defective $O(p^2)$-coloring of G can be computed in $O(\Delta^\epsilon) + \frac{1}{2}\log^* n$ time, $\forall \epsilon > 0$.

Thm: [Kuhn, SPAA'09]
$\forall G,\ \forall p\ \lfloor \Delta/p \rfloor$-defective $O(p^2)$-coloring of G can be computed in $O(\log^* \Delta) + \frac{1}{2}\log^* n$ time.

Open: can one efficiently achieve a linear (in Δ) product of defect and #colors?

Partial answer: for *edge*-coloring it is possible.
Also, for vertex-coloring of graphs with bounded independence.
[Barenboim,Elkin,10]

$(\Delta + 1)$-Coloring Algorithm

- Compute $O\left(\frac{\Delta}{\log \Delta}\right)$-defective $\log^2 \Delta$-coloring of G in $o(\Delta) + O(\log^* n)$ time.
  $(p = \log \Delta)$

- Each color class induces a subgraph with maximum degree $\Delta' = O\left(\frac{\Delta}{\log \Delta}\right)$.

  Subgraphs are vertex-disjoint.

- In parallel, compute $(\Delta' + 1)$-coloring in each of the $\log^2 \Delta$ subgraphs in $O(\Delta' \log \Delta' + \log^* n) = O(\Delta + \log^* n)$ time, using KW algorithm.

- Overall we get
  $O((\Delta' + 1) \log^2 \Delta) = O(\Delta \log \Delta)$-coloring φ of the entire original graph.

  (Using distinct palettes.)

- Invoke KW iterative procedure.

  We have $O(\log \Delta)$ sets

$$U_i = \{v \mid (i-1)(\Delta+1)+1 \le \varphi(v) \le i(\Delta+1)\}.$$

  Recolor $G(U_1 \cup U_2)$ in $(\Delta+1)$ colors.

  In parallel recolor $G(U_3 \cup U_4), \ldots, G(U_{(\log \Delta)-1} \cup U_{\log \Delta})$.

  We get $(\frac{1}{2} \cdot (\Delta+1) \log \Delta)$-coloring within additional $(\Delta+1)$ time.

$$\Downarrow$$

  After $\log(\log \Delta)$ phases we get $\frac{1}{2^{\log \log \Delta}} \cdot (\Delta+1) \cdot \log \Delta = (\Delta+1)$-coloring.

  Overall running time is $O((\Delta+1) \cdot \log \log \Delta + \log^* n) + o(\Delta)$.

This is a self-improving scheme!

Now we have $(\Delta + 1)$-coloring algorithm that runs in $O(\Delta \log\log \Delta + \log^* n)$ time.

- Compute $O\left(\frac{\Delta}{\log\log\Delta}\right)$-defective $(\log\log\Delta)^2$-coloring in $o(\Delta) + O(\log^* n)$ time.

- $\Delta' = \frac{\Delta}{\log\log\Delta}$.

  Compute $(\Delta' + 1)$-coloring of each subgraph in $O(\Delta' \log\log \Delta' + \log^* n) = O(\Delta + \log^* n)$ time.

- Combine these colorings into an $O(\Delta \log\log \Delta)$-coloring of G (in zero time).

- Reduce the $O(\Delta \cdot \log\log \Delta)$-coloring via KW iterative procedure into a $(\Delta + 1)$-coloring within $O(\Delta \cdot \log^{(3)} \Delta + \log^* n)$ additional time.

  Overall we get $(\Delta + 1)$-coloring in $O(\Delta \cdot \log^{(3)} \Delta + \log^* n)$ time.

$$\Downarrow$$

Repeating this argument $\log^* \Delta$ times we get $(\Delta + 1)$-coloring in $O(\Delta + \log^* n)$ time.

A tradeoff (an application)

$\forall t$, $O(\Delta \cdot t)$-coloring in $O(\Delta/t + \log^* n)$ time. (Interpolates between Linial's $O(\Delta^2)$-coloring in $\log^* n$ time, and our $(\Delta+1)$-coloring in $O(\Delta + \log^* n)$ time.)

- Compute (Δ/t)-defective $O(t^2)$-coloring in $O(\log^* n)$ time.

- We get $O(t^2)$ vertex-disjoint subgraphs, each with $\Delta' \leq \Delta/t$.

  Compute $(\Delta'+1)$-coloring of each, in parallel, in $O(\Delta' + \log^* n) = O(\Delta/t + \log^* n)$, using the last result for $(\Delta'+1)$-coloring.

- Combine the colorings in zero time to get $O(t^2 \cdot \Delta') = O(\Delta \cdot t)$-coloring, in total $O(\Delta/t + \log^* n)$ time.

Reminder: Graphs of bounded arboricity

aka k-*inductive* or k-*degenerate* graphs

Def: For $G = (V, E)$,

$$a(G) = \max\{\lceil \frac{|E(U)|}{|U| - 1} \rceil : U \subseteq V, |U| \geq 2\}.$$

Nash-Williams Thm: (1961)
$a(G)$ is the minimum number of edge-disjoint forests $F_1, F_2, \ldots, F_a$ such that $E = \bigcup_{i=1}^{a} F_i$, $(F_i \cap F_j = \emptyset \quad \forall i \neq j)$.

Graphs with Small Arboricity

Observation 1:
In an n-vertex graph $G = (V, E)$
with $a(G) = a$, $\forall \eta > 0$ there exists
a subset H of $\Omega(\eta \cdot n)$ vertices s.t.
$\forall v \in H, \quad deg(v) \leq (2 + \eta) \cdot a$.

It extends the notion of *degeneracy*:
a graph of degeneracy d
must contain at least one vertex v
with $deg(v) \leq d$.

Observation 2: The family of graphs of bounded arboricity is closed under taking induced subgraphs.
(Specifically, $a(G(V \setminus H)) \leq a(G)$)

$$\Downarrow$$

We can extract sets H as above multiple times, and get an H-*partition* of G.

Proof of Observation 1

Let αn be the # vertices
v with $deg(v) \leq (2 + \eta) \cdot a$.

Suppose for contradiction that $\alpha < \eta/3$.

$$U = \{v \mid deg(v) > (2 + \eta) \cdot a\}.$$

Hence $|U| = (1 - \alpha)n > (1 - \eta/3)n$.

$$|E(U)| > \frac{1}{2}(1 - \eta/3) \cdot (2 + \eta) \cdot a \cdot n > a \cdot n.$$

$$\Downarrow$$

$$a(G) \geq \lceil \frac{|E(U)|}{|U| - 1} \rceil > a.$$

$\Rightarrow\Leftarrow$

QED

H-Decomposition

$G = (V, E)$, $a(G) = a$. Fix $\eta > 0$.

$G_0 \leftarrow G$;

$H_1 \leftarrow \{v \in V \mid deg(v, V) \leq (2 + \eta) \cdot a\}$;

$G_1 \leftarrow (V \setminus H_1, E(V \setminus H_1))$;

$a(G_1) \leq a(G_0)$;

$H_2 \leftarrow \{v \in V \setminus H_1 \mid deg(v, V \setminus H_1) \leq (2 + \eta) \cdot a\}$;

$\vdots$

$$H_j \leftarrow \{v \in V \setminus \bigcup_{i=1}^{j-1} H_i \mid deg(v, V \setminus \bigcup_{i=1}^{j-1} H_i) \leq (2 + \eta) \cdot a\}.$$

For some ℓ, all vertices v in $H_\ell = V \setminus \bigcup_{i=1}^{\ell-1} H_i$ have $deg(v, H_\ell) \leq (2 + \eta) \cdot a$.

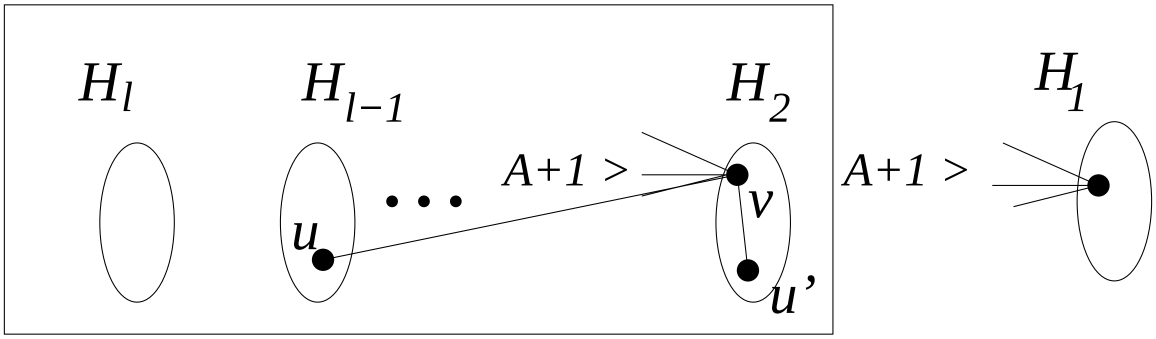
H_ℓ is the last set in the H-decomposition.

ℓ - the number of H-sets.

On each step at least an $\Omega(\eta)$-fraction of all vertices is eliminated.

So $(1 - \Omega(\eta))^\ell < 1/n$, i.e.,

$\ell = O(\frac{1}{\eta} \log n)$.



$A = (2 + \eta) \cdot a, \ \ \eta > 0$

$V = \bigcup_{i=1}^{\ell} H_i, \quad H_i \cap H_j = \emptyset, \ \forall i \neq j$

$\forall i \in [\ell], \ \forall v \in H_i, \ deg(v, \bigcup_{j=i}^{\ell} H_j) \leq A.$

In particular, $deg(v, H_i) \leq deg(v, \bigcup_{j=i}^{\ell} H_j) \leq A.$

The H-decomposition can be computed
in $O(\ell) = O(\log n)$ time.
(One round for each H_i.)

[Zhou,Nishizeki,95],
[Barenboim,Elkin,08]

Coloring Using H-Decomposition

- Compute an H-decomposition $H_1, H_2, \ldots, H_\ell$ in $O(\ell) = O(\log n)$ time.

- In parallel, in each H_i compute $(A+1)$-coloring φ in $O(A + \log^* n)$ time. ($\Delta(H_i) \leq A$)

- Recolor to obtain an $(A+1)$-coloring ψ of the entire original graph G.

  On this step we spend $O(A \cdot \ell) = O(a \cdot \log n)$ time.

Recoloring (Producing ψ)

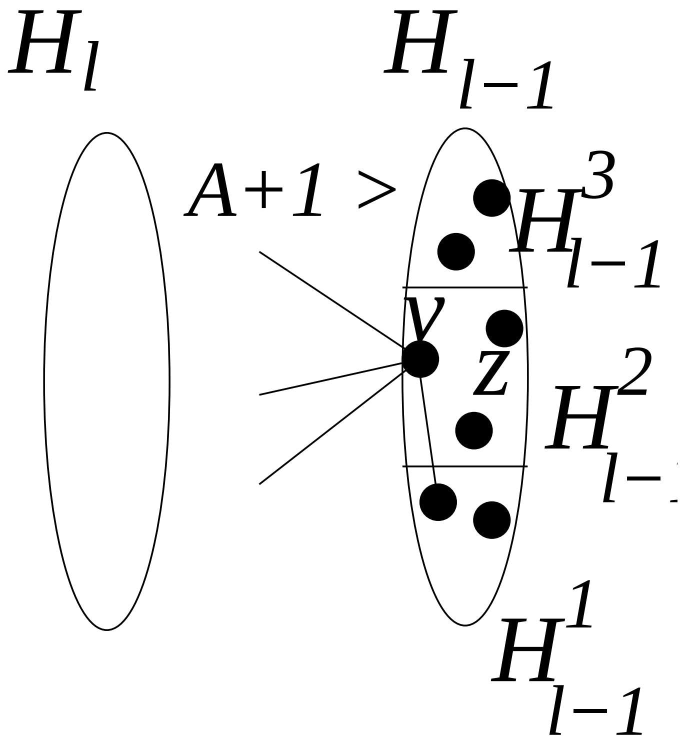
Spend $(A+1)$ rounds on each set H_i.

Start with H_ℓ.
Each $v \in H_\ell$ sets $\psi(v) \leftarrow \varphi(v)$.

Proceed to $H_{\ell-1}$.
$\forall r \in [A+1]$,
$H^r_{\ell-1} = \{v \in H_{\ell-1} \mid \varphi(v) = r\}$.

Recolor one φ-color class at a time.
(Each φ-color class is an independent set.)

Suppose for some $r \in [A]$, that
$H^1_{\ell-1} \cup \ldots \cup H^r_{\ell-1}$ are already recolored.

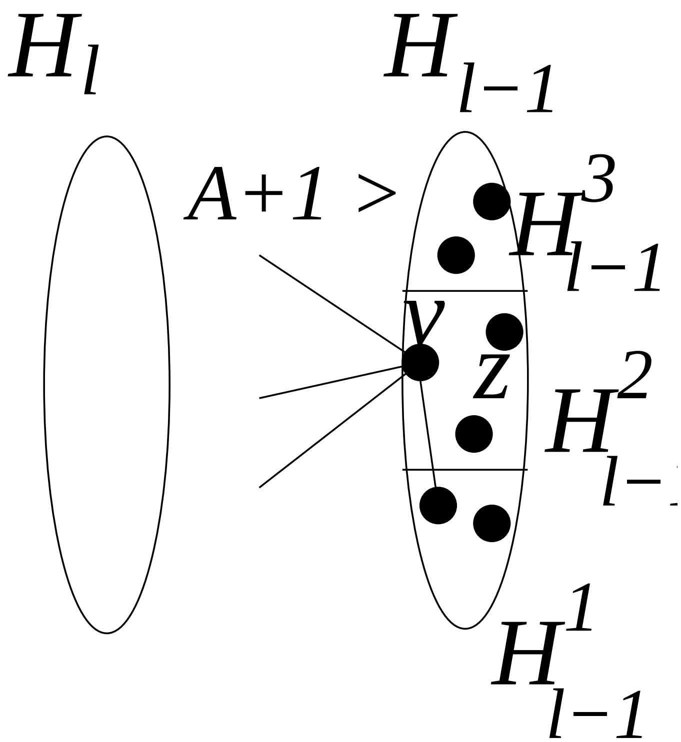


Consider $v \in H_{\ell-1}^{r+1}$.

v has $\leq A$ neighbors in $H_\ell \cup H_{\ell-1}$.

$$\Downarrow$$

v has $\leq A$ recolored neighbors.
(Because those are in $H_\ell \cup \bigcup_{j=1}^{r} H_{\ell-1}^{j}$.)



Hence there is a color $c = c(v) \in [A+1]$
s.t. no recolored neighbor u of v
has $\psi(u) = c$.

All vertices $v \in H_{\ell-1}^{r+1}$
compute in parallel $c(v)$ and
set $\psi(v) \leftarrow c(v)$.

Since $H_{\ell-1}^{r+1}$ is an independent set,
the new coloring ψ is legal.

The algorithm:

Recolor $H_{\ell-1}^{1}$, then $H_{\ell-1}^{2}, \ldots, H_{\ell-1}^{A+1}$;
then recolor $H_{\ell-2}^{1}, H_{\ell-2}^{2}, \ldots, H_{\ell-2}^{A+1}$;

$$\vdots$$

$$H_{1}^{1}, H_{1}^{2}, \ldots, H_{1}^{A+1}.$$

There are $A+1$ color classes in each H_i, and ℓ sets H_i.

One round per color class.
Overall $O((A+1) \cdot \ell) = O(a \cdot \log n)$ time.

Thm: $(2+\eta) \cdot a$-coloring can be computed in $O(a \cdot \log n)$ time.

[Barenboim,Elkin,08]

Further Results on Arboricity Coloring

$\forall i = O(1)$,
$(2+\eta)^{i+1} \cdot a$-coloring in $O(a^{\frac{2}{i+2}} \cdot \log n)$ time.

E.g.,
$(2+\eta) \cdot a$-coloring in $O(a \cdot \log n)$ time.
$(4+\eta) \cdot a$-coloring in $O(a^{2/3} \cdot \log n)$ time.
$(8+\eta) \cdot a$-coloring in $O(a^{1/2} \cdot \log n)$ time.

$\vdots$

Hence,
$O(a)$-coloring in $O(a^{\epsilon} \cdot \log n)$ time.
$a^{1+\eta}$-coloring in $O(\log a \cdot \log n)$ time.
$a^{1+o(1)}$-coloring in $O(f(a) \cdot \log a \cdot \log n)$ time,
for any function $f(a) = \omega(1)$.

All this applies when replacing a by Δ.

$$\Downarrow$$

$(\Delta + 1)$-coloring in $O(\log \Delta \cdot \log n)$ time, whenever $a < \Delta^{1-\eta}$, for some constant $\eta > 0$.

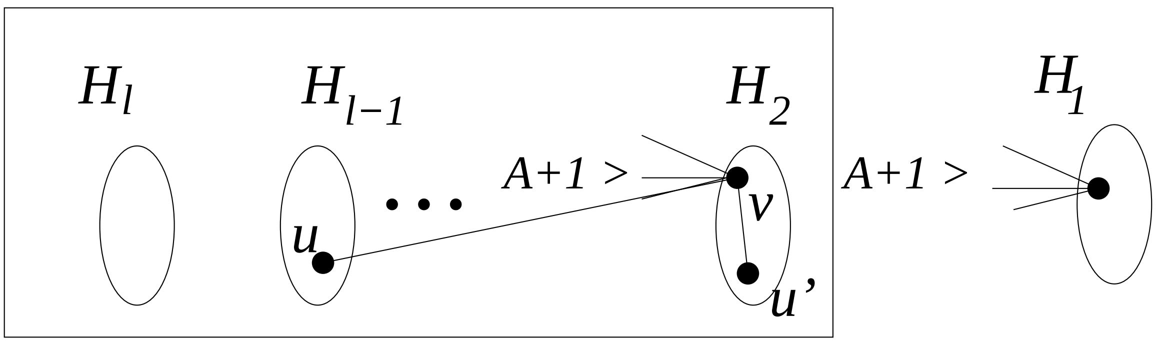
[Barenboim,Elkin PODC'10]

Open: Improve this result.
Achieve sublogarithmic time?
Or, $O(a)$-coloring in polylogarithmic time?

(Rather than $a^{1+o(1)}$-coloring in polylogarithmic time.)

Forest Decomposition

Another view of H-decomposition.



$\forall i \in [\ell],\ \forall v \in H_i$,
$deg(v, \bigcup_{j=i}^{\ell} H_j) \leq A$.

$\forall (u,v)$ with $u \in H_j,\ v \in H_i,\ j > i$,
call u a *parent* of v.

Also, $\forall (u', v)$ with $u', v \in H_i$,
and $Id(u') > Id(v)$,
call u' a *parent* of v.

$\Downarrow$

v has $\leq A$ parents,
and for every edge, exactly one of the end-points is a parent of the other.

Each vertex v labels edges (v, u) s.t.
u is a parent of v with numbers $1, 2, \ldots, A$.

G is decomposed into $\leq A$
edge-disjoint forests.

(Forest of edges labeled by 1, by 2, ..., by A.)

Applications of Forest Decomposition

THM: $\forall n', r, \quad n' > r$,
$\exists$ family $\mathcal{Q} = \mathcal{Q}(n', r)$ of n' subsets of $\{1, 2, \ldots, 5r^2 \cdot \log n'\}$ s.t.
$\forall r+1$ sets $Q_0, Q_1, \ldots, Q_r \in \mathcal{Q}$,

$$Q_0 \not\subseteq \bigcup_{i=1}^{r} Q_i \ .$$

[Erdős,Frankl,Furedi,85]

Algorithm Arb-Linial (BE08)

- Each vertex v sets
  $\varphi(v) \leftarrow Id(v)$; $n' \leftarrow n$; $r \leftarrow A$.

- Each v computes (locally)
  the set system $\mathcal{Q} = \mathcal{Q}(n, A)$.

- Each v sends $\varphi(v)$ to all its *children*.

- Given the colors (Ids) of its *parents*
  $\varphi(u_1), \varphi(u_2), \ldots, \varphi(u_A)$,
  v finds an element
  $q \in Q_{\varphi(v)} \setminus \bigcup_{i=1}^{A} Q_{\varphi(u_i)}$.

- Set $\varphi(v) \leftarrow q$.

The coloring is legal because for every edge (v, u) s.t. u is a parent of v, $\varphi(v) = q \in Q_{\varphi(v)} \setminus Q_{\varphi(u)}$ and $\varphi(u) = q' \in Q_{\varphi(u)}$.

We get $5A^2 \cdot \log n$-coloring in one single round.

Iterate for $\log^* n$ rounds to get $O(A^2)$-coloring.
$+O(\log n)$ time for computing the forest decomposition.

Thm: [BE08]
$O(a^2)$-coloring can be computed in $O(\log n)$ time.

This can be modified to get $O(a^2 \cdot \log^\epsilon n)$-coloring in $O\left(\frac{\log n}{\log \log n}\right)$ time.

For $a = O(1)$ (e.g., planar graphs), this implies $O(\log^\epsilon n)$-coloring in $O\left(\frac{\log n}{\log \log n}\right)$ time.

If $\Delta \leq \log^{\epsilon} n$ then we obtained $(\Delta + 1)$-coloring in sublogarithmic time.

If $\Delta > \log^{\epsilon} n$ then in $O(\log^{\epsilon} n) - (\Delta + 1) = O(\log^{\epsilon} n)$ additional time we get $(\Delta + 1)$-coloring.

Also, in both cases, in $O(\log^{\epsilon} n)$ additional time we get an MIS.

Thm: [BE08]
For graphs with $a = O(\log^{1/2-\epsilon} n)$, $(\Delta + 1)$-coloring and MIS can be computed in $O\left(\frac{\log n}{\log \log n}\right)$ time.

(Improved logarithmic bounds of [Goldberg,Plotkin,Shannon,87].)

Open Questions

1. $(\Delta + 1)$-coloring or MIS in deterministic polylogarithmic time?

   Or at least $O(\Delta)$-coloring.

   Currently we have $\Delta \cdot 2^{O(\frac{\log \Delta}{\log\log \Delta})}$-coloring.

2. $\Delta^{2-\epsilon}$-coloring in sublogarithmic time?

3. $(\Delta + 1)$-coloring in $o(\Delta)$ time?
   Or a lower bound?

   Currently we have $O(\Delta) + \frac{1}{2}\log^* n$ time.

4. Δ/p-defective $O(p)$-coloring in deterministic polylogarithmic time? (Known for edge-coloring, and for vertex-coloring of graphs with bounded neighborhood independence.)

5. $(2a+1)$-coloring faster than in $O(a^2 \log n)$ time?
$(2+\eta)\cdot a$-coloring faster than in $O(a \log n)$ time?

We know
$(2+\eta)^{1/\epsilon}a$-coloring in $O(a^\epsilon \cdot \log n)$ time, and $a^{1+\epsilon}$-coloring in $O(\log a \cdot \log n)$ time.

There is also a lower bound of $\Omega\left(\frac{\log n}{\log a}\right)$ for $O(a^2)$-coloring.

So unlike graphs with bounded degree, for graphs of bounded arboricity one cannot hope for sublogarithmic time.