Testing $k$-colorability

Noga Alon and Michael Krivelevich:

Speaker: Joseph, Chuang-Chieh Lin

Advisor: Professor Maw-Shang Chang

Computation Theory Laboratory
Dept. Computer Science and Information Engineering
National Chung Cheng University, Taiwan

October 22, 2008
Outline

1. Introduction
2. The algorithm
3. Preliminaries
   - Some notations
   - Main idea of the proof
4. Detailed analysis
Graph model: **dense graph** (adjacency matrix) for $G(V,E)$.
- undirected, no self-loops, $\leq 1$ edge between any $u, v \in V$
- $|V| = n$ vertices and $|E| = \Omega(n^2)$ edges.

A graph property:
- A set of graphs closed under isomorphisms.

Let $\mathbb{P}$ be a graph property.
- $\epsilon$-far from satisfying $\mathbb{P}$:
  - $\geq \epsilon n^2$ edges should be deleted or added to let the graph satisfy $\mathbb{P}$
Graph model: dense graph (adjacency matrix) for $G(V, E)$.
- undirected, no self-loops, $\leq 1$ edge between any $u, v \in V$
- $|V| = n$ vertices and $|E| = \Omega(n^2)$ edges.

A graph property:
- A set of graphs closed under isomorphisms.

Let $\mathbb{P}$ be a graph property.
- $\epsilon$-far from satisfying $\mathbb{P}$:
  - $\geq \epsilon n^2$ edges should be deleted or added to let the graph satisfy $\mathbb{P}$.
Graph model: dense graph (adjacency matrix) for $G(V, E)$.
- undirected, no self-loops, $\leq 1$ edge between any $u, v \in V$
- $|V| = n$ vertices and $|E| = \Omega(n^2)$ edges.

A graph property:
- A set of graphs closed under isomorphisms.

Let $\mathbb{P}$ be a graph property.
- $\epsilon$-far from satisfying $\mathbb{P}$:
  - $\geq \epsilon n^2$ edges should be deleted or added to let the graph satisfy $\mathbb{P}$
**Property testing:**
- it does NOT precisely determine YES or NO for a decision problem;
- requires sublinear running time

**A property tester for \( \mathbb{P} \):**
- A randomized algorithm such that
  - it answers “YES” with probability of \( \geq 2/3 \) if \( G \) satisfies \( \mathbb{P} \), and
  - it answers “NO” with probability of \( \geq 2/3 \) if \( G \) is \( \epsilon \)-far from satisfying \( \mathbb{P} \)

**\( \mathbb{P} \) is testable if**
- \( \exists \) a property tester for \( \mathbb{P} \) such that its running time complexity is independent of \( n \).
Introduction (property testing)

- **Property testing:**
  - it does NOT precisely determine YES or NO for a decision problem;
  - requires sublinear running time

- **A property tester** for $P$:
  - A randomized algorithm such that
    - it answers “YES” with probability of $\geq 2/3$ if $G$ satisfies $P$, and
    - it answers “NO” with probability of $\geq 2/3$ if $G$ is $\epsilon$-far from satisfying $P$

- $P$ is testable if
  - $\exists$ a property tester for $P$ such that its running time complexity is independent of $n$. 

Computation Theory Lab, CSIE, CCU, Taiwan  Testing $k$-colorability
Property testing:
- it does NOT precisely determine YES or NO for a decision problem;
- requires sublinear running time

A property tester for \( \mathcal{P} \):
- A randomized algorithm such that
  - it answers “YES” with probability of \( \geq 2/3 \) if \( G \) satisfies \( \mathcal{P} \), and
  - it answers “NO” with probability of \( \geq 2/3 \) if \( G \) is \( \epsilon \)-far from satisfying \( \mathcal{P} \)

\( \mathcal{P} \) is testable if
- \( \exists \) a property tester for \( \mathcal{P} \) such that its running time complexity is independent of \( n \).
Testing emptiness of a graph

- Testing $H$-freeness, where $H$ is an edge.
- Query complexity and time complexity: $O(1/\epsilon)$
- How can it be done?

Testing connectivity is trivial (for dense graphs).
- Why?
Examples

- Testing emptiness of a graph
  - Testing $H$-freeness, where $H$ is an edge.
    - Query complexity and time complexity: $O(1/\epsilon)$
    - How can it be done?

- Testing connectivity is trivial (for dense graphs).
  - Why?
Examples

- Testing emptiness of a graph
  - Testing $H$-freeness, where $H$ is an edge.
  - Query complexity and time complexity: $O(1/\epsilon)$
  - How can it be done?

- Testing connectivity is trivial (for dense graphs).
  - Why?
Examples

- Testing emptiness of a graph
  - Testing $H$-freeness, where $H$ is an edge.
  - Query complexity and time complexity: $O(1/\epsilon)$
  - How can it be done?

- Testing connectivity is trivial (for dense graphs).
  - Why?
Examples

- Testing emptiness of a graph
  - Testing $H$-freeness, where $H$ is an edge.
  - Query complexity and time complexity: $O(1/\epsilon)$
  - How can it be done?

- Testing connectivity is trivial (for dense graphs).
  - Why?
Testing emptiness of a graph
- Testing $H$-freeness, where $H$ is an edge.
- Query complexity and time complexity: $O(1/\epsilon)$
- How can it be done?

Testing connectivity is trivial (for dense graphs).
- Why?
Introduction (k-colorability)

- a (proper) k-coloring: a function $f : V \rightarrow \{1, 2, \ldots, k\}$ such that
  
  $f(u) \neq f(v)$ if $(u, v) \in E$.

- Equivalent to a $k$-partition $(V_1, V_2, \ldots, V_k)$ of $V$ such that for each $i$, $(u, v) \notin E$ for every $u, v \in V_i$.

- For convenience, we denote $\{1, 2, \ldots, k\}$ by $[k]$. 

Computation Theory Lab, CSIE, CCU, Taiwan

Testing $k$-colorability
Introduction ($k$-colorability)

- a (proper) $k$-coloring: a function $f : V \rightarrow \{1, 2, \ldots, k\}$ such that
  - $f(u) \neq f(v)$ if $(u, v) \in E$.

- Equivalent to a $k$-partition $(V_1, V_2, \ldots, V_k)$ of $V$ such that for each $i$, $(u, v) \notin E$ for every $u, v \in V_i$.

- For convenience, we denote $\{1, 2, \ldots, k\}$ by $[k]$. 

Introduction ($k$-colorability)

- a (proper) $k$-coloring: a function $f : V \rightarrow \{1, 2, \ldots, k\}$ such that
  
  - $f(u) \neq f(v)$ if $(u, v) \in E$.

- Equivalent to a $k$-partition $(V_1, V_2, \ldots, V_k)$ of $V$ such that for each $i$, $(u, v) \notin E$ for every $u, v \in V_i$.

- For convenience, we denote $\{1, 2, \ldots, k\}$ by $[k]$. 

Testing $k$-colorability
a (proper) $k$-coloring: a function $f : V \rightarrow \{1, 2, \ldots, k\}$ such that

- $f(u) \neq f(v)$ if $(u, v) \in E$.

Equivalent to a $k$-partition $(V_1, V_2, \ldots, V_k)$ of $V$ such that for each $i$, $(u, v) \notin E$ for every $u, v \in V_i$.

For convenience, we denote $\{1, 2, \ldots, k\}$ by $[k]$. 

Introduction (k-colorability)
**Introduction (k-colorability)**

- **NP-complete for** \( k \geq 3 \)

- *k*-colorability is testable.
  - Hereditary graph property is testable [Alon and Shapira 2008] (by Szemerédi’s regularity Lemma)
  - Dependency of tower of 2’s of height polynomial in \( 1/\epsilon \).

- Query complexity: \( O(k^2 \ln^2 k/\epsilon^4) \);
- Time complexity: \( \exp(k \ln k/\epsilon^2) \); [Alon and Krivelevich 2002; this paper]
NP-complete for $k \geq 3$

- $k$-colorability is testable.
  - Hereditary graph property is testable [Alon and Shapira 2008] (by Szemerédi’s regularity Lemma)
    - Dependency of tower of 2’s of height polynomial in $1/\epsilon$.
  - Query complexity: $O(k^2 \ln^2 k/\epsilon^4)$;
    Time complexity: $\exp(k \ln k/\epsilon^2)$; [Alon and Krivelevich 2002; this paper]
Introduction (\(k\)-colorability)

- **NP**-complete for \(k \geq 3\)

- \(k\)-colorability is testable.
  - Hereditary graph property is testable [Alon and Shapira 2008] (by Szemerédi’s regularity Lemma)
    - Dependency of tower of 2’s of height polynomial in \(1/\epsilon\).
  - Query complexity: \(O(k^2 \ln^2 k/\epsilon^4)\);
    Time complexity: \(\exp(k \ln k/\epsilon^2)\); [Alon and Krivelevich 2002; this paper]
Introduction (\(k\)-colorability)

- **NP-complete** for \(k \geq 3\)

- **\(k\)-colorability is testable.**
  
  - Hereditary graph property is testable [Alon and Shapira 2008] (by Szemerédi’s regularity Lemma)
    - Dependency of tower of 2’s of height polynomial in \(1/\epsilon\).
  
  - Query complexity: \(O(k^2 \ln^2 k/\epsilon^4)\);
    Time complexity: \(\exp(k \ln k/\epsilon^2)\); [Alon and Krivelevich 2002; this paper]
Outline

1. Introduction

2. The algorithm

3. Preliminaries
   - Some notations
   - Main idea of the proof

4. Detailed analysis
The property tester for $k$-colorability is very simple.

<table>
<thead>
<tr>
<th>$k$-coloring-tester ($G, s$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generate a random subset $R \subset V$ of size $s = 36k \ln k/\epsilon^2$</td>
</tr>
<tr>
<td>Exhaustively color $R$ by $k$ colors.</td>
</tr>
<tr>
<td>Return YES if $G[R]$ is $k$-colorable, and return NO otherwise.</td>
</tr>
</tbody>
</table>
Testing $k$-colorability
Introduction
The algorithm
Preliminaries
Detailed analysis

Computation Theory Lab, CSIE, CCU, Taiwan

Testing $k$-colorability

Graph with nodes A, B, C, D, E, F, G, H, I, each node colored with integers from 1 to 5.

- Node A is colored red.
- Node B is colored with 2, 3, 4, 5.
- Node C is colored with 2, 3, 4, 5.
- Node D is colored with 1, 2, 3, 4, 5.
- Node E is colored with 2, 3, 4, 5.
- Node F is colored with 2, 3, 4, 5.
- Node G is colored with 2, 3, 4, 5.
- Node H is colored with 1, 2, 3, 4, 5.
- Node I is colored with 1, 2, 3, 4, 5.
Testing $k$-colorability
Introduction
The algorithm
Preliminaries
Detailed analysis

Computation Theory Lab, CSIE, CCU, Taiwan

Testing \( k \)-colorability
Computation Theory Lab, CSIE, CCU, Taiwan

Testing $k$-colorability
Testing $k$-colorability
Testing $k$-colorability
Testing $k$-colorability
Testing $k$-colorability
The property tester for $k$-colorability

- If $G$ is $k$-colorable, then the algorithm always returns YES.

- What if $G$ is $\epsilon$-far from being $k$-colorable?
The property tester for $k$-colorability

- If $G$ is $k$-colorable, then the algorithm always returns YES.
- What if $G$ is $\epsilon$-far from being $k$-colorable?
Outline

1. Introduction
2. The algorithm
3. Preliminaries
   - Some notations
   - Main idea of the proof
4. Detailed analysis
Given $S \subseteq V$ and its $k$-partition $\phi : S \rightarrow [k]$.

The list of feasible labels of a vertex $v \in V \setminus S$

$L_\phi(v) = [k] \setminus \{1 \leq i \leq k : \exists u \in S \cap N(v), \phi(u) = i\}$.

$v \in V \setminus S$ is called colorless if $L_\phi(v) = 0$. 
Some notations

- Given $S \subseteq V$ and its $k$-partition $\phi : S \rightarrow [k]$.

**The list of feasible labels of a vertex $v \in V \setminus S$**

$$L_{\phi}(v) = [k] \setminus \{1 \leq i \leq k : \exists u \in S \cap N(v), \phi(u) = i\}.$$  

- $v \in V \setminus S$ is called colorless if $L_{\phi}(v) = 0$.  

Some notations

- Given $S \subseteq V$ and its $k$-partition $\phi : S \to [k]$.

The list of feasible labels of a vertex $v \in V \setminus S$

$L_\phi(v) = [k] \setminus \{1 \leq i \leq k : \exists u \in S \cap N(v), \phi(u) = i\}$.

- $v \in V \setminus S$ is called **colorless** if $L_\phi(v) = 0$. 
Some notations (contd.)

- \( S = \{A, B, E, H, I\} \).
- \( \phi(A) = 1, \phi(B) = 3, \phi(E) = 2, \phi(H) = 1, \phi(I) = 1. \)
- No colorless vertices w.r.t. \((S, \phi)\).
Some notations (contd.)

- \( S = \{A, B, E, H, I\} \).
- \( \phi(A) = 1, \phi(B) = 3, \phi(E) = 2, \phi(H) = 1, \phi(I) = 1 \).
- No colorless vertices w.r.t. \((S, \phi)\).
Some notations (contd.)

- \( S = \{A, B, E, H, I\} \).
- \( \phi(A) = 1, \phi(B) = 3, \phi(E) = 2, \phi(H) = 1, \phi(I) = 1 \).
- No colorless vertices w.r.t. \((S, \phi)\).
Main idea of the proof

- Assume that $G$ is $\epsilon$-far from being $k$-colorable.

- Suppose we are given a subset $S \subset R \subset V(G)$ and its $k$ partition $\phi : S \rightarrow [k]$.

- Our aim is to find w.h.p. that:
  
  - a succinct (i.e., short & concise) witness in $R \setminus S$ to the fact that $\phi$ can NOT be extended to a (proper) $k$-coloring.

  - **Witness**: a set of vertices which can be used to find out non-$k$-colorability. (colorless or restricting vertices)
  
  - **Extending $\phi$**: giving other vertices colors based on $(S, \phi)$. 

Computation Theory Lab, CSIE, CCU, Taiwan  
Testing $k$-colorability
Main idea of the proof

- Assume that $G$ is $\epsilon$-far from being $k$-colorable.

- Suppose we are given a subset $S \subset R \subset V(G)$ and its $k$ partition $\phi : S \rightarrow [k]$.

- Our aim is to find w.h.p. that:

  - a succinct (i.e., short & concise) witness in $R \setminus S$ to the fact that $\phi$ can NOT be extended to a (proper) $k$-coloring.

    - **Witness**: a set of vertices which can be used to find out non-$k$-colorability. (colorless or restricting vertices)
    - **Extending $\phi$**: giving other vertices colors based on $(S, \phi)$. 
Main idea of the proof

- Assume that $G$ is $\epsilon$-far from being $k$-colorable.

- Suppose we are given a subset $S \subset R \subset V(G)$ and its $k$ partition $\phi : S \to [k]$.

- Our aim is to find w.h.p. that:
  - a succinct (i.e., short & concise) witness in $R \setminus S$ to the fact that $\phi$ can NOT be extended to a (proper) $k$-coloring.

  - **Witness**: a set of vertices which can be used to find out non-$k$-colorability. (colorless or restricting vertices)

  - **Extending $\phi$**: giving other vertices colors based on $(S, \phi)$. 
Main idea of the proof

- Assume that $G$ is $\epsilon$-far from being $k$-colorable.
- Suppose we are given a subset $S \subset R \subset V(G)$ and its $k$ partition $\phi : S \rightarrow [k]$.
- Our aim is to find w.h.p. that:
  - a succinct (i.e., short & concise) witness in $R \setminus S$ to the fact that $\phi$ can NOT be extended to a (proper) $k$-coloring.

- **Witness**: a set of vertices which can be used to find out non-$k$-colorability. (colorless or restricting vertices)
- **Extending $\phi$**: giving other vertices colors based on $(S, \phi)$. 

Testing $k$-colorability
Main idea of the proof

- Assume that $G$ is $\epsilon$-far from being $k$-colorable.

- Suppose we are given a subset $S \subset R \subset V(G)$ and its $k$ partition $\phi : S \rightarrow [k]$.

- Our aim is to find w.h.p. that:

  ▶ a succinct (i.e., short & concise) witness in $R \setminus S$ to the fact that $\phi$ can NOT be extended to a (proper) $k$-coloring.

  - **Witness**: a set of vertices which can be used to find out non-$k$-colorability. (colorless or restricting vertices)
  
  - **Extending $\phi$**: giving other vertices colors based on $(S, \phi)$. 
Main idea of the proof

- Assume that $G$ is $\epsilon$-far from being $k$-colorable.
- Suppose we are given a subset $S \subset R \subset V(G)$ and its $k$-partition $\phi : S \rightarrow [k]$.
- Our aim is to find w.h.p. that:
  - a succinct (i.e., short & concise) witness in $R \setminus S$ to the fact that $\phi$ can NOT be extended to a (proper) $k$-coloring.
    - **Witness**: a set of vertices which can be used to find out non-$k$-colorability. (colorless or restricting vertices)
    - **Extending $\phi$**: giving other vertices colors based on $(S, \phi)$. 
Main idea of the proof (contd.)

- If there are a lot of colorless vertices w.r.t. \((S, \phi)\) ...
  - It is easy to obtain a witness for nonextendability of \(\phi\).

- What if the number of colorless vertices is small?
  - As \(G\) is \(\epsilon\)-far from being \(k\)-colorable, one can show that:
    - \(\exists W \subseteq V\) (\(|W|\) is large) s.t. coloring every vertex \(v \in W\) by any feasible color w.r.t. \(\phi\) reduces the number of feasible colors of at least \(\Omega(\epsilon) n\) neighbors of \(v\).
  - It helps approach the previous case.
If there are a lot of colorless vertices w.r.t. \((S, \phi)\) ...

- It is easy to obtain a witness for nonextendability of \(\phi\).

What if the number of colorless vertices is small?

- As \(G\) is \(\epsilon\)-far from being \(k\)-colorable, one can show that:
  - \(\exists W \subseteq V\) (\(|W|\) is large) s.t. coloring every vertex \(v \in W\) by any feasible color w.r.t. \(\phi\) reduces the number of feasible colors of at least \(\Omega(\epsilon)n\) neighbors of \(v\).

- It helps approach the previous case.
Main idea of the proof (contd.)

- If there are a lot of colorless vertices w.r.t. $(S, \phi)$ ...
  - It is easy to obtain a witness for nonextendability of $\phi$.

- What if the number of colorless vertices is small?
  - As $G$ is $\epsilon$-far from being $k$-colorable, one can show that:
    - $\exists W \subseteq V$ ($|W|$ is large) s.t. coloring every vertex $v \in W$ by any feasible color w.r.t. $\phi$ reduces the number of feasible colors of at least $\Omega(\epsilon)n$ neighbors of $v$.
  - It helps approach the previous case.
If there are a lot of colorless vertices w.r.t. \((S, \phi)\) ... It is easy to obtain a witness for nonextendability of \(\phi\).

What if the number of colorless vertices is small?
As \(G\) is \(\epsilon\)-far from being \(k\)-colorable, one can show that:

\[
\exists W \subset V \ (|W| \text{ is large}) \text{ s.t. coloring every vertex } v \in W \text{ by any feasible color w.r.t. } \phi \text{ reduces the number of feasible colors of at least } \Omega(\epsilon)n \text{ neighbors of } v.
\]

It helps approach the previous case.
Main idea of the proof (contd.)

- If there are a lot of colorless vertices w.r.t. \((S, \phi)\) ...
  - It is easy to obtain a witness for nonextendability of \(\phi\).

- What if the number of colorless vertices is small?
  - As \(G\) is \(\epsilon\)-far from being \(k\)-colorable, one can show that:
    - \(\exists \ W \subset V\ (|W| \text{ is large})\ s.t.\ \text{coloring every vertex } v \in W \text{ by any feasible color w.r.t. } \phi \text{ reduces the number of feasible colors of at least } \Omega(\epsilon)n \text{ neighbors of } v.\)

  - It helps approach the previous case.
The above process can be represented by an auxiliary tree $T$.

- Every node of $T$ corresponds to a colorless or a restricting vertex $v$.
  - Each node is labeled by a vertex of $G$ or by the symbol $\#$ (terminal node).

- Every edge of $T$ corresponds to a feasible color for $v$. 

Testing $k$-colorability
The above process can be represented by an auxiliary tree $T$.

Every node of $T$ corresponds to a colorless or a restricting vertex $v$.

- Each node is labeled by a vertex of $G$ or by the symbol $\#$ (terminal node).

Every edge of $T$ corresponds to a feasible color for $v$. 
Main idea of the proof (contd.)

- The above process can be represented by an auxiliary tree $T$.
- Every node of $T$ corresponds to a colorless or a restricting vertex $v$.
  - Each node is labeled by a vertex of $G$ or by the symbol $\#$ (terminal node).
- Every edge of $T$ corresponds to a feasible color for $v$. 
Main idea of the proof (contd.)

Testing $k$-colorability
Let $t$ be a node of $T$.

The path from the root of $T$ to $t$ not including $t$ itself defines a $k$-partition (we call it $\phi(t)$) of the labels (i.e., vertices of $G$; we call it $S(t)$) along this path.

If $t$ is labeled by $v$ and $v$ has a neighbor in $S(t)$ whose color in $\phi(t)$ is also $i$, the son of $v$ along the edge labeled by $i$ is labeled by $\#$. 

Computation Theory Lab, CSIE, CCU, Taiwan

Testing $k$-colorability
Let $t$ be a node of $T$. 

The path from the root of $T$ to $t$ not including $t$ itself defines a $k$-partition (we call it $\phi(t)$) of the labels (i.e., vertices of $G$; we call it $S(t)$) along this path.

If $t$ is labeled by $v$ and $v$ has a neighbor in $S(t)$ whose color in $\phi(t)$ is also $i$, then the son of $v$ along the edge labeled by $i$ is labeled by $\#$. 
Let $t$ be a node of $T$.

The path from the root of $T$ to $t$ not including $t$ itself defines a $k$-partition (we call it $\phi(t)$) of the labels (i.e., vertices of $G$; we call it $S(t)$) along this path.

If $t$ is labeled by $v$ and $v$ has a neighbor in $S(t)$ whose color in $\phi(t)$ is also $i$, the son of $v$ along the edge labeled by $i$ is labeled by $\#$. 
Since the degree of each node of $T$ can be as large as $k$, the size of $T$ grows exponentially.

We therefore need the probability of choosing colorless or restricting vertices to be exponentially close to 1.
Since the degree of each node of $T$ can be as large as $k$, the size of $T$ grows exponentially.

We therefore need the probability of choosing colorless or restricting vertices to be exponentially close to 1.
Outline

1. Introduction
2. The algorithm
3. Preliminaries
   - Some notations
   - Main idea of the proof
4. Detailed analysis
Reducing feasible colors

- For every \( v \in V \setminus (S \cup U) \):

Estimation of \# excluded feasible colors of \( N(v) \) outside \( S \cup U \)

\[
\delta_{\phi}(v) = \min_{i \in L_{\phi}(v)} |\{ u \in N(v) \setminus (S \cup U) : i \in L_{\phi}(u) \}|.
\]

- \( U \) is the set of colorless vertices w.r.t. \( (S, \phi) \).
\[ \delta_\phi(B) = \min_{i \in \{3, 4, 5\}} \{4, 4, 4\} = 4. \]
\[ \delta_\phi(C) = \min_{i \in \{2, 3, 4, 5\}} \{0, 1, 1, 1\} = 0. \]
\[ \delta_\phi(D) = \min_{i \in \{2, 3, 4, 5\}} \{0, 2, 2, 2\} = 0. \]
\[ \delta_\phi(F) = \min_{i \in \{2, 3, 4, 5\}} \{0, 2, 2, 2\} = 0. \]
\[ \delta_\phi(G) = \min_{i \in \{3, 4, 5\}} \{4, 4, 4\} = 4. \]
\[ \delta_\phi(H) = \min_{i \in \{1, 3, 4, 5\}} \{0, 4, 4, 4\} = 0. \]
Restricting vertices

Given a pair \((S, \phi)\), a vertex is called **restricting** if \(\delta_\phi(v) \geq \epsilon n/2\).

\[
W := \{ v \in V \setminus (S \cup U) \mid \delta_\phi(v) \geq \epsilon n/2 \}.
\]
Claim 1

For every subset $S \subset V$ and every $k$-partition $\phi$ of $S$, to make the graph be $k$-colorable requires deleting at most 

$$(n - 1)(|S| + |U|) + \sum_{v \in V \setminus (S \cup U)} \delta_\phi(v)$$

edges.

- “$\epsilon$-far from being $k$-colorable” makes sense only if $\epsilon n^2 < (n - 1)(|S| + |U|) + \sum_{v \in V \setminus (S \cup U)} \delta_\phi(v)$.
- Thus we have the following corollary.
Claim 1

For every subset $S \subset V$ and every $k$-partition $\phi$ of $S$, to make the graph be $k$-colorable requires deleting at most 

$$ (n - 1)(|S| + |U|) + \sum_{v \in V \setminus (S \cup U)} \delta_\phi(v) $$

edges.

- “$\epsilon$-far from being $k$-colorable” makes sense only if 
  $$ \epsilon n^2 < (n - 1)(|S| + |U|) + \sum_{v \in V \setminus (S \cup U)} \delta_\phi(v). $$

- Thus we have the following corollary.
Clai 1

For every subset $S \subset V$ and every $k$-partition $\phi$ of $S$, to make the graph be $k$-colorable requires deleting at most

$$(n - 1)(|S| + |U|) + \sum_{v \in V \setminus (S \cup U)} \delta_\phi(v)$$

\begin{itemize}
  \item “$\epsilon$-far from being $k$-colorable” makes sense only if $\epsilon n^2 < (n - 1)(|S| + |U|) + \sum_{v \in V \setminus (S \cup U)} \delta_\phi(v)$.
  \item Thus we have the following corollary.
\end{itemize}
Corollary 4.1

If $G$ is $\epsilon$-far from being $k$-colorable, then for any pair $(S, \phi)$, where $S \subset V(G)$, $\phi : S \rightarrow [k]$, one has

$$\sum_{v \in V \setminus (S \cup U)} \delta_\phi(v) > \epsilon n^2 - n(|S| + |U|),$$

where $U$ is the set of colorless vertices w.r.t. $(S, \phi)$. 
The number of restricting vertices must be large

Claim 2

If $G$ is $\epsilon$-far from being $k$-colorable, then for any pair $(S, \phi)$, where $S \subset V(G)$, $\phi : S \rightarrow [k]$, one has

$$|U| + |W| > \frac{\epsilon n}{2} - |S|.$$ 

Proof.

$$\epsilon n^2 - n(|S| + |U|) < \sum_{v \in V \setminus (S \cup U)} \delta_\phi(v) \leq |W|(n - 1) + \sum_{v \in V \setminus (S \cup U \cup W)} \delta_\phi(v) < |W|n + \frac{\epsilon n^2}{2}.$$
The number of restricting vertices must be large

Claim 2

If $G$ is $\epsilon$-far from being $k$-colorable, then for any pair $(S, \phi)$, where $S \subseteq V(G)$, $\phi : S \rightarrow [k]$, one has

$$|U| + |W| > \frac{\epsilon n}{2} - |S|.$$  

Proof.

$$\epsilon n^2 - n(|S| + |U|) < \sum_{v \in V \setminus (S \cup U)} \delta_{\phi}(v) \leq |W|(n - 1) + \sum_{V \setminus (S \cup U \cup W)} \delta_{\phi}(v)$$

$$< |W|n + \frac{\epsilon n^2}{2}.$$
Recall the auxiliary tree $T$ for the coloring process

- Consider a leaf $t$ of $T$.
  - $U(t)$: the set of colorless vertices w.r.t. $(S(t), \phi(t))$.
  - $W(t)$: the set of restricting vertices w.r.t. $(S(t), \phi(t))$.

A nonterminal node of $T$ is labeled only when a vertex in $U(t) \cup W(t)$ is chosen.
Recall the auxiliary tree $T$ for the coloring process

- Consider a leaf $t$ of $T$.
- $U(t)$: the set of colorless vertices w.r.t. $(S(t), \phi(t))$.
- $W(t)$: the set of restricting vertices w.r.t. $(S(t), \phi(t))$.
- A nonterminal node of $T$ is labeled only when a vertex in $U(t) \cup W(t)$ is chosen.
Consider a leaf $t$ of $T$.

- $U(t)$: the set of colorless vertices w.r.t. $(S(t), \phi(t))$.
- $W(t)$: the set of restricting vertices w.r.t. $(S(t), \phi(t))$.

A nonterminal node of $T$ is labeled only when a vertex in $U(t) \cup W(t)$ is chosen.
Recall the auxiliary tree $T$ for the coloring process

- Consider a leaf $t$ of $T$.
- $U(t)$: the set of colorless vertices w.r.t. $(S(t), \phi(t))$.
- $W(t)$: the set of restricting vertices w.r.t. $(S(t), \phi(t))$.
- A nonterminal node of $T$ is labeled only when a vertex in $U(t) \cup W(t)$ is chosen.
An upper bound on the depth of $T$

Claim 3

The depth of $T$ is bounded by $\frac{2k}{\epsilon}$.

Proof.

- The depth of $T$ is mainly due to the restricting vertices.
- The total length of the lists of feasible colors initially: $nk$.
- Coloring a vertex $w \in W$: reduces $\geq \epsilon n/2$ colors.
- We cannot make more than $nk/(\epsilon n/2) = 2k/\epsilon$ steps down from the roof of $T$ to a leaf of $T$. 
An upper bound on the depth of $T$

Claim 3

The depth of $T$ is bounded by $\frac{2k}{\epsilon}$.

Proof.

- The depth of $T$ is mainly due to the restricting vertices.
- The total length of the lists of feasible colors initially: $nk$.
- Coloring a vertex $w \in W$: reduces $\geq \frac{\epsilon n}{2}$ colors.
- We cannot make more than $nk/(\epsilon n/2) = 2k/\epsilon$ steps down from the roof of $T$ to a leaf of $T$. 
An upper bound on the depth of $T$

**Claim 3**

The depth of $T$ is bounded by $\frac{2k}{\epsilon}$.

**Proof.**

- The depth of $T$ is mainly due to the restricting vertices.
- The total length of the lists of feasible colors initially: $nk$.
- Coloring a vertex $w \in W$: reduces $\geq \epsilon n/2$ colors.
- We cannot make more than $nk/(\epsilon n/2) = 2k/\epsilon$ steps down from the roof of $T$ to a leaf of $T$. 
An upper bound on the depth of $T$

Claim 3

*The depth of $T$ is bounded by $\frac{2k}{\epsilon}$.*

**Proof.**

- The depth of $T$ is mainly due to the restricting vertices.
- The total length of the lists of feasible colors initially: $nk$.
- Coloring a vertex $w \in W$: reduces $\geq \frac{\epsilon n}{2}$ colors.
- We cannot make more than $nk/(\epsilon n/2) = 2k/\epsilon$ steps down from the roof of $T$ to a leaf of $T$. 

Computation Theory Lab, CSIE, CCU, Taiwan

Testing $k$-colorability
Claim 4

If a leaf $t^*$ of $T$ is labeled by $\#$, then $\phi(t^*)$ is not a proper $k$-coloring of $S(t^*)$.

Claim 5

If all leaves $t^*$'s of $T$ are terminal nodes after $j$ rounds of the algorithm, then the subgraph induced by the labels along the path from the root of $T$ to $t^*$ is not $k$-colorable.
The leaves of $T$ are all leaves w.h.p. before long

Claim 6

*If $G$ is $\epsilon$-far from being $k$-colorable, then after $36k \ln k/\epsilon^2$ rounds, with probability $\geq 2/3$ all leaves of $T$ are terminal nodes.*

Proof.

- $T$ can be embedded into a $k$-ary tree $T_{k, \frac{2k}{\epsilon}}$ of depth $\frac{2k}{\epsilon}$.
- $T_{k, \frac{2k}{\epsilon}}$ has at most $1 + k + \ldots + k \cdot \frac{2k}{\epsilon} \leq k \cdot \frac{2k}{\epsilon} + 1$ vertices.
- A round of the algorithm is called *successful* if a colorless vertex or a restricting vertex is picked.
The leaves of $T$ are all leaves w.h.p. before long

Claim 6

*If $G$ is $\epsilon$-far from being $k$-colorable, then after $36k \ln k/\epsilon^2$ rounds, with probability $\geq 2/3$ all leaves of $T$ are terminal nodes.*

Proof.

* $T$ can be embedded into a $k$-ary tree $T_k, \frac{2k}{\epsilon}$ of depth $\frac{2k}{\epsilon}$.

* $T_k, \frac{2k}{\epsilon}$ has at most $1 + k + \ldots + k^{\frac{2k}{\epsilon}} \leq k^{\frac{2k}{\epsilon}+1}$ vertices.

* A round of the algorithm is called *successful* a colorless vertex or a restricting vertex is picked.
The leaves of $T$ are all leaves w.h.p. before long

Claim 6

*If $G$ is $\epsilon$-far from being $k$-colorable, then after $36k \ln k/\epsilon^2$ rounds, with probability $\geq 2/3$ all leaves of $T$ are terminal nodes.*

Proof.

- $T$ can be embedded into a $k$-ary tree $T_{k, \frac{2k}{\epsilon}}$ of depth $\frac{2k}{\epsilon}$.
- $T_{k, \frac{2k}{\epsilon}}$ has at most $1 + k + \ldots + k \frac{2k}{\epsilon} \leq k \frac{2k}{\epsilon} + 1$ vertices.
- A round of the algorithm is called **successful** if a colorless vertex or a restricting vertex is picked.
Proof of Claim 6 (contd.)

Proof.

- Fix some leaf node $t$ of $T$ after $36k \ln k / \epsilon^2$ rounds of the algorithm.

- The total number of successful rounds for the path from the root of $T$ to $t$ is equal to the depth of $t$.

- Besides, the probability of choosing a colorless or restricting vertex (i.e., $U(t) \cup W(t)$) is at least $\epsilon/2 - S(t)/n = \epsilon/2 - o(1) \geq \epsilon/3$. 
Proof of Claim 6 (contd.)

Proof.

Fix some leaf node $t$ of $T$ after $36k \ln k/\epsilon^2$ rounds of the algorithm.

The total number of successful rounds for the path from the root of $T$ to $t$ is equal to the depth of $t$.

Besides, the probability of choosing a colorless or restricting vertex (i.e., $U(t) \cup W(t)$) is at least $\epsilon/2 - S(t)/n = \epsilon/2 - o(1) \geq \epsilon/3$. 
Proof of Claim 6 (contd.)

Proof.

- Fix some leaf node $t$ of $T$ after $36k \ln k/\epsilon^2$ rounds of the algorithm.

- The total number of successful rounds for the path from the root of $T$ to $t$ is equal to the depth of $t$.

- Besides, the probability of choosing a colorless or restricting vertex (i.e., $U(t) \cup W(t)$) is at least $\epsilon/2 - S(t)/n = \epsilon/2 - o(1) \geq \epsilon/3$. 
Proof of Claim 6 (contd.)

Proof.

- Fix some leaf node $t$ of $T$ after $36k \ln k/\epsilon^2$ rounds of the algorithm.
- The total number of successful rounds for the path from the root of $T$ to $t$ is equal to the depth of $t$.
- Besides, the probability of choosing a colorless or restricting vertex (i.e., $U(t) \cup W(t)$) is at least $\epsilon/2 - S(t)/n = \epsilon/2 - o(1) \geq \epsilon/3$. 
Proof.

- $\Pr[t \text{ is a nonterminal leaf of } T]$ can be bounded by $\Pr[B(36k \ln k/\epsilon^2, \epsilon/3) < 2k/\epsilon]$.
- $B(n, p)$ is the Binomial random variable of $n$ Bernoulli trials with probability $p$ of success.

The Chernoff bound for $B(n, p)$:

$$\Pr[B(m, p) \leq k] \leq \exp \left( -\frac{1}{2p} \frac{(mp - k)^2}{m} \right).$$
Proof of Claim 6 (contd.)

Proof.

- $\Pr[t \text{ is a nonterminal leaf of } T]$ can be bounded by $\Pr[B(36k \ln k/\epsilon^2, \epsilon/3) < 2k/\epsilon]$.
  - $B(n, p)$ is the Binomial random variable of $n$ Bernoulli trials with probability $p$ of success.
  - The Chernoff bound for $B(n, p)$:

$$\Pr[B(m, p) \leq k] \leq \exp \left( -\frac{1}{2p} \frac{(mp - k)^2}{m} \right).$$
Proof of Claim 6 (contd.)

Proof.

- **Pr** \([B(36k \ln k/\epsilon^2, \epsilon/3) < 2k/\epsilon] < k^{-3k/\epsilon}\) by the Chernoff bound.

  Thus by the union bound we conclude that the probability that some node of \(T_k, \frac{2k}{\epsilon}\) is a nonterminal leaf is

  \[\leq |V(T_k, \frac{2k}{\epsilon})| \cdot k^{-3k/\epsilon} < 1/3.\]

- That means, the probability that the algorithm finds a proper \(k\)-coloring is less than 1/3.

- Hence we derive the error probability of the algorithm < 1/3.
Proof of Claim 6 (contd.)

Proof.

- \( \Pr[B(36k \ln k/\epsilon^2, \epsilon/3) < 2k/\epsilon] < k^{-3k/\epsilon} \) by the Chernoff bound.

- Thus by the union bound we conclude that the probability that some node of \( T_k, \frac{2k}{\epsilon} \) is a nonterminal leaf is

  \[ \leq |V(T_k, \frac{2k}{\epsilon})| \cdot k^{-\frac{3k}{\epsilon}} < 1/3. \]

- That means, the probability that the algorithm finds a proper \( k \)-coloring is less than \( 1/3 \).

- Hence we derive the error probability of the algorithm < \( 1/3 \).
Proof.

- \( \Pr[B(36k \ln k/\varepsilon^2, \varepsilon/3) < 2k/\varepsilon] < k^{-3k/\varepsilon} \) by the Chernoff bound.

- Thus by the union bound we conclude that the probability that some node of \( T_k, \frac{2k}{\varepsilon} \) is a nonterminal leaf is
  \[
  \leq |V(T_k, \frac{2k}{\varepsilon})| \cdot k^{-\frac{3k}{\varepsilon}} < 1/3.
  \]

- That means, the probability that the algorithm finds a proper \( k \)-coloring is less than 1/3.

- Hence we derive the error probability of the algorithm < 1/3.
Proof of Claim 6 (contd.)

Proof.

- \( \Pr[B(36k \ln k/\epsilon^2, \epsilon/3) < 2k/\epsilon] < k^{-3k/\epsilon} \) by the Chernoff bound.

- Thus by the union bound we conclude that the probability that some node of \( T_k, \frac{2k}{\epsilon} \) is a nonterminal leaf is

\[
\leq |V(T_k, \frac{2k}{\epsilon})| \cdot k^{-\frac{3k}{\epsilon}} < \frac{1}{3}.
\]

- That means, the probability that the algorithm finds a proper \( k \)-coloring is less than \( 1/3 \).

- Hence we derive the error probability of the algorithm < \( 1/3 \).
Thank you!