Color Coding

Students: Wuwei Zhang *June 11th, 2014*

1.1 Introduction

In computer science, the method of color-coding efficiently find k-vertex simple paths, k-vertex cycles, and other small subgraphs within a given graph using probabilistic algorithms. The results can be derandomized by using families of perfect hash functions.

1.2 Terms Preliminary

Definition 1.1 (Simple Path) *A* **simple path** *in a graph G is that path that does not contain*

Definition 1.2 (Colorful Path) *A path is said to be colorful if each vertex is colored by a distinct color.*

Corollary 1.3 *A colorful path is a simple path.*

Definition 1.4 (Planar Graph) *A planar graph is a graph that can be drawn on the plane in such a way that its edges intersect* **only at their endpoints**.

Theorem 1.5 *A graph G is a planar graph if and only if it does not contain a subgraph that is a subdivision of complete graph on five vertices or a complete bipartite graph on six vertices.*

Definition 1.6 (Graph isomorphism) *In graph theory, an isomorphism of graphs G and H is a bijection(1-1 and onto) between the vertex sets of G and H.*

$$f:\ V(G) \rightarrow V(H)$$

such that any two vertices u and v of G are adjacent in G if and only if f(u) and f(v) are adjacent in H.

Definition 1.7 (Tree Decomposition) *A tree decomposition of graph $G = (V, E)$ is a tree T with nodes $X_1, \ldots, X_n$, where each X_i is a* **subset** *of V satisfying the following properties:*

1. $\bigcup_i X_i = V$
2. $v \in X_i \cap X_j$, *then all nodes X_k of the tree in the path between X_i and X_j contain v as well.*
3. $\forall$ *edge (v, w) in the graph,* $\exists\ X_i$ *contains v and w.*

Definition 1.8 (Width of a tree decomposition) *The width of a tree decomposition is the size of its largest set X_i minus one.*

Definition 1.9 (Treewidth) *The treewidth tw(G) of a graph G is the minimum width among all possible tree decomposition of G.*

1.3 Color-Coding Algorithms

1.3.1 Overview

∀ fixed k, if a graph G = (V, E) contains a simple cycle of exactly length of k, then it can be found either in O(V^{ω}) expected time or O(V^{ω}log V) worst time. Where ω = 2.376 comes from matrix multiplication algorithm.

∀ fixed k, if a **planar** graph G = (V, E) contains a simple cycle of exactly length of k, then it can be found either in O(V) expected time or O(VlogV) worst time.

If a graph G = (V, E) contains a subgraph isomorphic to a bounded tree-width graph H = (V_H, E_H) where $|V_H|$ = O(logV), then such a copy of H can be found in polynomial time.

1.3.2 Method to find a path of length k

Step 1: Find all path that have a length of k

Let G = (V, E) be an undirected graph. Suppose we want to find pairs of vertices connected by simply paths of length exactly k. By raising the adjacency matrix A_G of G to the k th power, or by using other methods, we can easily find all pairs of vertices connected by paths of length k.

Step 2: Eliminating nonsimple path

An easy way to eliminating those path is by choosing a **random acylic orientation of the graph G**

1. Choosing a random permutation π : V → $\{1, ..., |V|\}$.
2. **Undirected version:** Direct an edge(u, v) ∈ E from u to v if and only if $\pi(u) < \pi(v)$.
   Directed version: Delete every edge (u, v) ∈ E if $\pi(u) > \pi(v)$.

Notice that every simple path of length k in G has a $\frac{2}{(k+1)!}$ chance of becoming a directed path in the random acylic orientation.

Theorem 1.10 *A simple directed or undirected path of length k in a graph G that contains such a path can be found in* **O((k+2)!·V)** *expected time in the undirected case and in* **O((k+1)!·E)** *expected time in the directed case.*

1.3.3 Method to find a cycle of length k

Use algorithms in the previous section to find path of k - 1. Check the starting and ending vertices.

1.4 Random Colorings

Let G = (V, E) be a directed or undirected graph. Each simple path of length k - 1 has a chance $\frac{k!}{k^k} > e^{-k}$ to become colorful.

Lemma 1.11 *Let G = (V, E) be a directed or undirected graph and let c: $V \to 1, ..., k$ be a coloring of its vertices with k colors. A colorful path of length k - 1 in G can be found in $2^{O(k)} \cdot E$ worst case time.*

The proof of above lemma using a dynamic programming approach.

Lemma 1.12 *Let G = (V, E) be a directed or undirected graph and let c: $V \to 1, ..., k$ be a coloring of its vertices with k colors. All pairs of vertices connected by colorful paths of length k - 1 in G can be found in either $2^{O(k)} \cdot VE$ or $2^{O(k)} \cdot V^{\omega}$ worst-case time.*

The above lemma use the following approach to obtain the $2^{O(k)} \cdot VE$:

1. Enumerate all partitions of the color set $\{1, 2, ..., k\}$ into two subset C_1 and C_2 of size $\frac{k}{2}$ each.
2. For each partition C_1, C_2, let V_1, V_2 be the set of vertices of G colored colors from C_1 and C_2 respectively.
3. Let G_1 and G_2 be the subgraphs of G induced by V_1 and V_2 respectively.
4. Recursively find all pairs of vertices in G_1 and in G_2 connected by colorful paths of length $\frac{k}{2}$ - 1. Collect the information into Boolean matrices A_1 and A_2.
5. Let B be a Boolean matrix that describes the adjacency relations between the vertices of V_1 and vertices of V_2. Product A_1BA_2 gives all pairs of vertices in V that are connected by colorful paths of length exactly k - 1.