Implementing Path 3-Coloring and Path 3-Choosing Algorithms on Plane Graphs

August 2, 2016

Abstract

Path coloring a graph partitions its vertices into sets inducing a disjoint union of paths. In this project we consider several algorithms to compute path colorings of graphs embedded in the plane. We first implement an algorithm path 3-color plane graphs from Poh's proof in [a]. Second, we present a linear time implementation of an algorithm to path 3-choose plane graphs from the independant work of Hartman [b] and Skrekovski [c].

1 Introduction

All graphs discussed in this project will be simple, undirected, and finite. A graph is planar if it may be drawn in the plane without edge crossings. A k-coloring of a graph partitions its vertices into k color classes. Such a coloring is called proper if each color class consists of nonadjacent vertices. In 1976 Appel and Haken [d,e] displayed that all planar graphs have a proper 4-coloring. This result is best possible and solved the century old Four Color Conjecture. Generalizations of proper coloring were introduced in [f,g,h] allowing color classes to form forests, or allowing vertices to have some bounded number of same color neighbors. Cowen et al. ([j]) show a best possible result that planar graphs may be 3-colored such that each vertex recieves at most two same color neighbors.

We will be considering the problem of path coloring, producing a k-coloring of a graph such that each color class induces a disjoint union of paths, or equivalently a forest where each component is a path. This coloring was introduced by Harary in [i]. Note that this is similar to the defective coloring of Cowen et al. above, with the added restriction that path coloring forbids cycles. In [l] Poh displayes that all planar graphs have a path 3-coloring. Here we present an implementation of Poh's algorithm to path 3-color plane graphs.

Given a list of k colors for each vertex, a k-list-coloring assigns each vertex a color from its list. If a graph has a proper k-list-coloring it is said to be k-choosable. List-coloring was first introduced by Erdös et al. in [m]. Thomassen in [n] proves that all planar graphs are 5 choosable. Planar graphs that are not 4-choosable are described by Mirzakhani in [o] and Voigt in [p], so Thomassen's result is best possible.

Hull and Eaton in [q] prove planar graphs are 3-choosable such that each vertex recieves at most two same color neighbors, and furthermore show this result is best possible. Hartman in [r] and

Skrekovski in [s] independantly provide similar proofs that planar graphs are path 3-choosable. Hartman claims the proof yields a linear time algorithm for path 3-list-coloring, and thus path 3-coloring, plane graphs. Here we present a linear time implementation of Hartman and Skrekovski's algorithm.

2 Path 3-Coloring Plane Graphs

We first restate the algorithm and proof of Poh [l] for path 3-coloring plane graphs.

> **Theorem 1.** Let G be a 2-connected weakly triangulated plane graph or a complete graph on two vertices and suppose the outer face C has been 2-colored such that each color class induces a non-empty path. This 2-coloring may be extended to a path 3-coloring of G such that no vertex in $V(C)$ recieves a same color neighbor in $V(G) \setminus V(C)$.

Proof. If $|V(G)| \leq 3$ the coloring follows trivially. Let $|V(G)| > 3$ and suppose the theorem holds for all graphs H with $|V(H)| < |V(G)|$. Let $P = p_0 \ldots p_n$ and $Q = q_0 \ldots q_m$ denote the two induced paths from the 2-coloring of C such that the edges p_0q_0 and p_nq_m are in C. Suppose there exist uncolored vertices, that is $V(G) \setminus V(C) \neq \emptyset$.

Let t_0 be the vertex forming a face with p_0 and q_0. If $t_0 \in P$, this face is already colored and we consider the graph bounded by $P - p_0$ and Q. Similarly, if $t_0 \in Q$ then the inductive hypothesis applies to the graph bounded by P and $Q - q_0$. Let t_1 be the vertex forming a face with p_n and q_m and proceed in the same manner until t_1 is not in either path.

Suppose there exists an induced path T from t_0 to t_1. We color T the remaining color not assigned to P or Q and apply the inductive hypothesis to the subgraph bounded by P and T, and the subgraph bounded by T and Q. With only the path T in common between the two subgraphs, the combined 3-coloring forms a path coloring of G.

Suppose no such path exists from t_0 to t_1. Since G is weakly triangulated there must exist an edge $p_iq_m \in E(G)$ with $p_i \in P$ and $q_i \in Q$, distinct from p_0q_0 and p_nq_m. We may then separately apply the inductive hypothesis to the subgraph bounded by $p_0 \ldots p_i$ and $q_0 \ldots q_j$, and the subgraph bounded by $p_i \ldots p_n$ and $q_j \ldots q_m$. The two subgraphs only share the vertices p_i and q_j, thus the combined 3-coloring forms a path coloring of G. $\square$

Implementation

Let the plane graph G be represented as an adjacency list and an ordering of neighbors following a combinatorial embedding. We track the paths P and Q by marking each vertex with its respective path and storing the just path start and end vertices p_0, p_n, q_0, and q_m. To find t_0 we look through the ordered neighbors of p_0 and take the vertex counterclockwise past q_0. We find t_1 similarly by looking at p_n and q_m. These steps are repeated until the graph is colored or both t_0 and t_1 are not in either path.

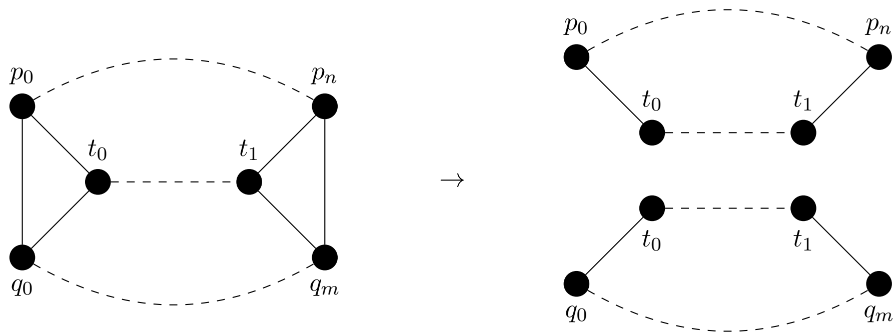

Figure 2.1 The case of a t_0t_1-path.

To check for the path from t_0 to t_1 we perform a breadth first search starting at t_1, and store parents for each vertex visited. If t_0 is reached, an induced path from t_0 to t_1 is produced by backtracking through the search. We then color and mark each vertex on the new path. Otherwise, we find a vertex $u \neq t_0$ with consecutive neighbors in opposite paths. These vertices will be p_i and q_j. We may then immediately recurse on region bounded by the p_ip_n-path and ut_1-path and the region bounded by the ut_1-path and the q_jp_m-path. To handle the remaining graph we also recurse using the p_0p_i-path and q_0p_j-path. Note as each recursive step is independant and marks are shared, this process may also be performed iteratively by instead pushing paths, represented by their start and end vertices, into a stack or queue.

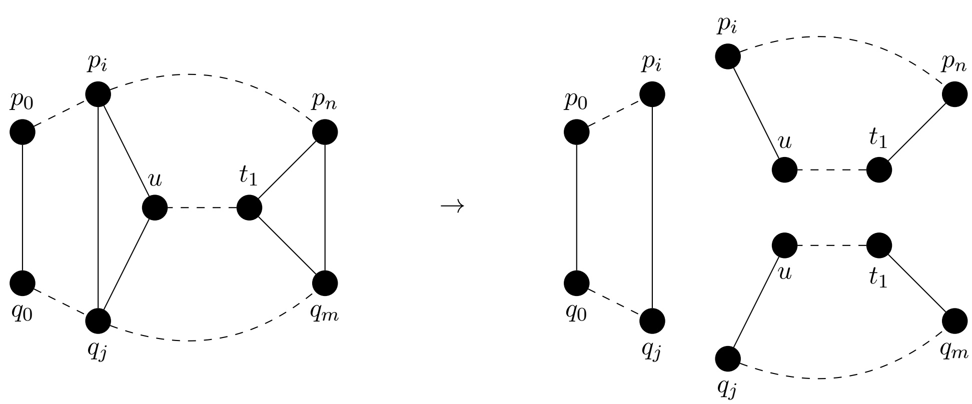

Figure 2.2 The case of no t_0t_1-path.

In locating t_0 and t_1 we make a neighbor lookup once for each. The amortized complexity of a neighbor lookup is $O(|E|/|V|)$. Since a vertex may only be t_0 or t_1 once in the algorithm we perform at most $|V|$ neighbor lookups and the amoritized complexity of this step is $O(|V||E|/|V|) = O(|E|)$. In planar graphs $|E| \leq 3|V| - 6$, and $O(|E|) = O(|V|)$. Therefore, the amortized complexity is $O(|V|)$. We also perform at most one breadth first search from each t_1, each with complexity $O(|V|)$. Therefore the complexity of the serach step over the entire graph is $O(|V|^2)$. This gives us an overall amortized complexity of $O(|V| + |V|^2) = O(|V|^2)$.

3 Path 3-Choosing Plane Graphs

The following theorem is a significant restatement of the work of Hartman [r] and Skrekovski [s]. The restatement limits each inductive step to considering a single vertex and its set of neighbors.

Suppose C is the outer cycle of a weakly triangulated plane graph G. Using notation from [r] for $u, v \in V(C)$ we let $C[u, v]$ denote the path from u to v clockwise along the outer face. If we wish to exclude u or v from this path we will use parenthesis, $C(u, v)$. Similarly, for $v \in V(G)$ and $u, w \in N(v)$ we let $[u, w]_v$ denote the path from u to w clockwise around v, assuming triangulated faces.

Theorem 2. Let G be a 2-connected weakly triangulated plane graph with outer face C, or a complete graph on two vertices. Let $x, y \in V(C)$ be not necessarily distinct, potentially precolored vertices. Let $p \in V(C[x, y])$ be precolored some color α. Suppose $L(v)$ assigns a list of colors to each $v \in V(G) \setminus \{p\}$ such that

$$\begin{array}{ll} |L(v)| \geq 1 & \text{if } v = x \text{ or } v = y; \\ |L(v)| \geq 2 & \text{if } v \in V(C) \setminus \{x, y\}; \\ |L(v)| \geq 3 & \text{otherwise.} \end{array}$$

If $p \neq x$, let $\alpha \notin L(v)$ for any $v \in V(C[x, p))$.

The coloring may then be extended to a path choosing of G from L such that x, y, and p recieve at most one same color neighbor. If $x = y$ then x and y recieve no same color neighbors. If $y = p$, or y is immediately prior to p on the outer face and $\alpha \notin L(y)$, then y recieves no same color neighbors.

Proof. Suppose $|V(G)| = 2$. If $x = y = p$ color the remaining vertex any color in its list other than α. Otherwise, color the remaining color any color in its list if it has not been precolored.

Suppose $|V(G)| > 2$ and the theorem holds for all graphs H with $|V(H)| < |V(G)|$. Let $C = c_0c_1 \ldots c_n$ denote, in clockwise order, the outer face of G with $p = c_0$. There are several cases to consider. Let $c_i \in V(C)$ be the next neighhbor of p counterclockwise from c_n. Let G_0 be the subgraph bounded by the cycle formed from $C[c_i, c_n]$ and $[c_n, c_i]_p$. If $c_i \neq c_1$ let G_1 be the sugbraph bounded by the cycle formed from $C[p, c_i]$ and the edge pc_i. As seen in Figure 3.1 $G = G_0 \cup G_1$ and $V(G_0) \cap V(G_1) = \{c_i\}$. We will display in each case that the inductive hypothesis holds for each subgraph and show their union still forms a path choosing of G from L. If $c_i = c_1$ then G_1 will not exist and we handle this case specially, noting $G_0 = G - p$.

Note that in all figures solid circles denote vertices yet to be colored, and colored vertices will be labeled with their assigned color. A label β will denote a color distinct from α. If a vertex is unlabled it represents arbitrary color assignment from L.

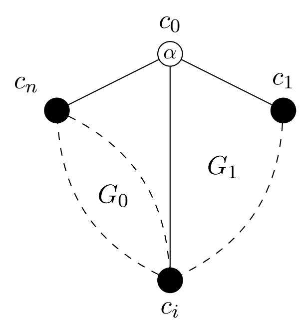

Figure 3.1 The subdivision of G into G_0 and G_1.

For all $v \in N(p) \cap V(G_0) \setminus \{x, y\}$ we note if $v = c_n$ or $v = c_i$ then $|L(v)| \geq 2$, and $|L(v)| \geq 3$ otherwise. We define a new list assignment L_0 such that $L_0(v) = L(v) \setminus \{\alpha\}$ for $v \in N(p) \cap V(G_0)$, and $L_0(v) = L(v)$ for $v \in V(G_0) \setminus N(p)$. Note that all $v \in N(p) \cap V(G_0)$ will be on the outer face of G_0. Thus $|L_0(v)| \geq 3$ for all interior vertices v of G_0. Also, $|L_0(v)| \geq 1$ for all $v \in \{x, y, c_i, c_n\}$. The case $\alpha \in L(y)$ when $y \in V(G_i) \cap N(p)$ will potentially result in $|L_0(y)| = 0$, but in this case $y = c_i$ and y will be colored α. Finally, $|L_0| \geq 2$ for all other v on the outer face of G_0. By choosing G_0 from L_0 we ensure p recieves no α colored neighbors in G_0, except in a few specially handled cases where c_i is colored α.

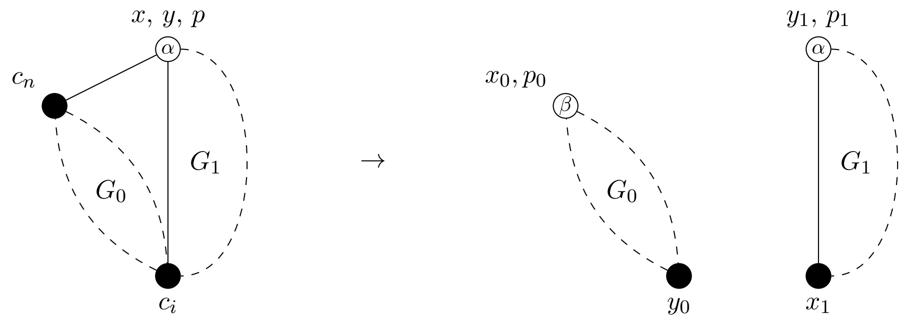

Figure 3.2 The case $x = y = p$.

Suppose $x = y = p$. Define $p_0 = c_n$ and precolor p_0 from $L_0(p_0)$. By our note above, clearly $L_0(v)$ satisfies the inductive hypothesis with G_0, $x_0 = c_n$, $y_0 = c_i$, and p_0. After choosing G_0 from L_0, if G_1 exists we apply the inductive hypothesis again to choose G_1 from L with $x_1 = c_i$, $y_1 = y$, and $p_1 = p$. Since c_i recieves at most one neighbor in each G_0 and G_1, the combined coloring forms a path choosing of G from L.

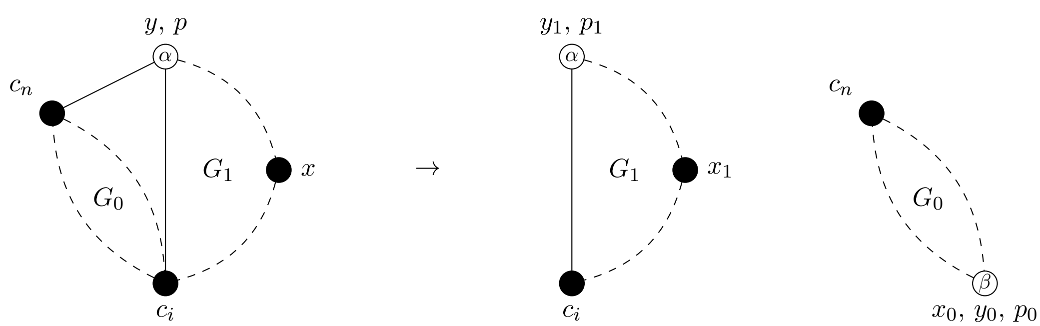

Figure 3.3 The case $y = p$, $x \neq p$, and $c_i \in V(C[x, p))$ (shown is the case $x \neq c_i$).

Suppose $y = p$, $x \neq p$, and $c_i \in V(C[x, p))$. I G_1 exists, first apply the inductive hypothesis to choose G_1 from L with $x_1 = x$, $y_1 = y$, and $p_1 = p$. Since $\alpha \notin L(v)$ for any $v \in V(C[x, p))$ we may apply the inductive hypothesis to choose G_0 from L_0 with $x_0 = y_0 = p_0 = c_i$. If G_1 exists note c_i precolored from our choosing of G_1 and recieves no same color neighbors in G_0, thus the combined coloring forms a path choosing of G from L.

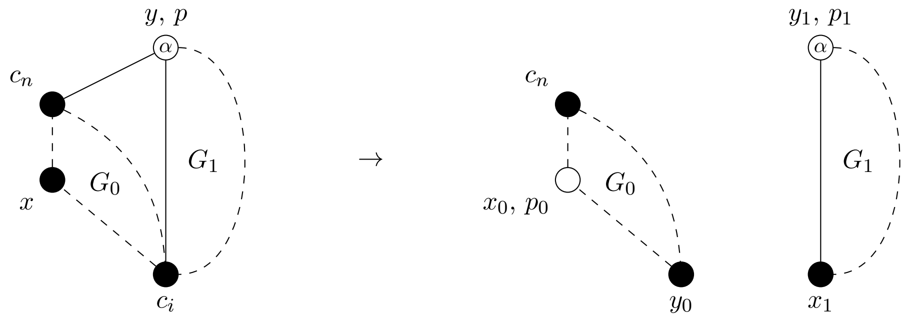

Figure 3.4 The case $y = p$, $x \neq p$, and $c_i \notin V(C[x, p))$.

Suppose $y = p$, $x \neq p$, and $c_i \notin V(C[x, p))$. Since $\alpha \notin L(v)$ for any $v \in V(C[x, p))$ we precolor x from $L_0(x)$ and apply the inductive hypothesis to choose G_0 from L_0 with $x_0 = p_0 = x$ and $y_0 = c_i$. If G_1 exists we apply the inductive hypothesis again to choose G_1 from L with $x_1 = c_i$, $y_1 = y$, and $p_1 = p$. Notice c_i recieves at most one same color neighbor in each G_0 and G_1.

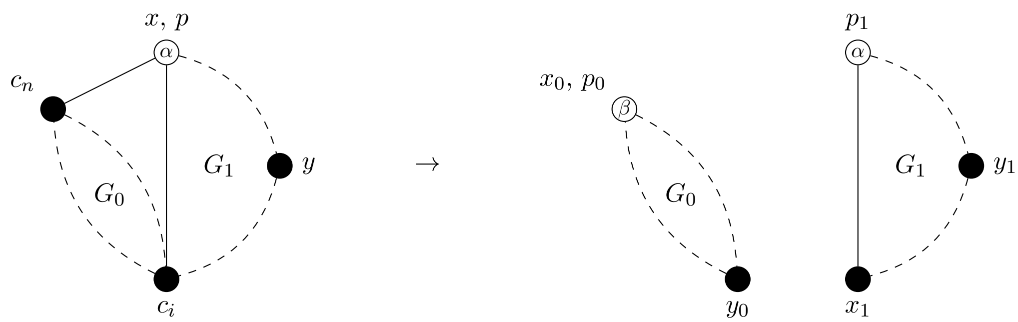

Figure 3.5 The case $x = p$, $y \neq p$, and $c_i \in V(C(y, x))$.

Suppose $x = p$, $y \neq p$, and $c_i \in V(C(y, x))$. In this case G_1 must exist, so first apply the inductive hypothesis to choose G_0 from L_0 with $x_0 = p_0 = c_n$ and $y_0 = c_i$. We again apply the inductive hypothesis to choose G_1 from L with $x_1 = c_i$, $y_1 = y$, and $p_1 = p$. Notice c_i recieves at most one same color neighbor in each G_0 and G_1.

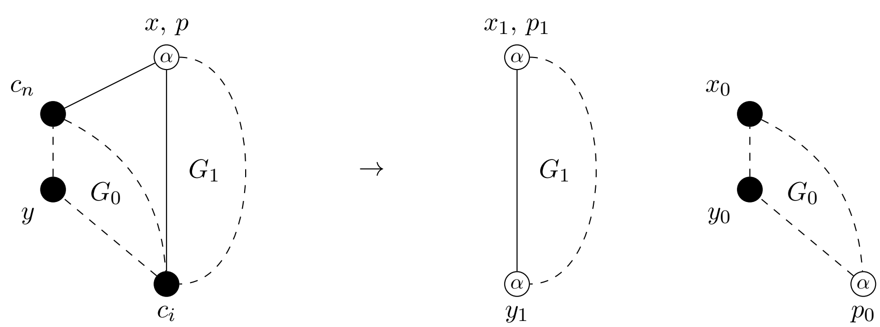

Figure 3.6 The case $x = p$, $y \neq p$, and $c_i \notin V(C(y, x))$ (shown is the case $y \neq c_i$ and $\alpha \in L(c_i)$).

Suppose $x = p$, $y \neq p$, and $c_i \notin V(C(y, x))$. If $\alpha \in L(c_i)$ we set $p_0 = c_i$ and color c_i with α. Otherwise, set $p_0 = c_n$ and color it with the first color in $L_0(c_n)$, note $|L_0(c_i)| \geq 2$ in this case. We then apply the inductive hypothesis to choose G_0 from L_0 with $x_0 = c_n$, $y_0 = y$, and p_0. If G_1 exists, we again apply the inductive hypothesis to color G_1 with $x_1 = p_1 = p$ and $y_1 = c_i$. Notice if $p_0 = c_i$, c_i recieves at most one same color neighbor in G_0 and the single same color neighbor p in

G_1. Furthermore, p will recieve no same color neighbor in G_1 other than c_i. If $p_0 \neq c_i$, then $y_1 = c_i$ is immediatley prior to $p_1 = p$ and $\alpha \notin L(c_i)$. Thus c_i will recieve no same color neighbors in G_1.

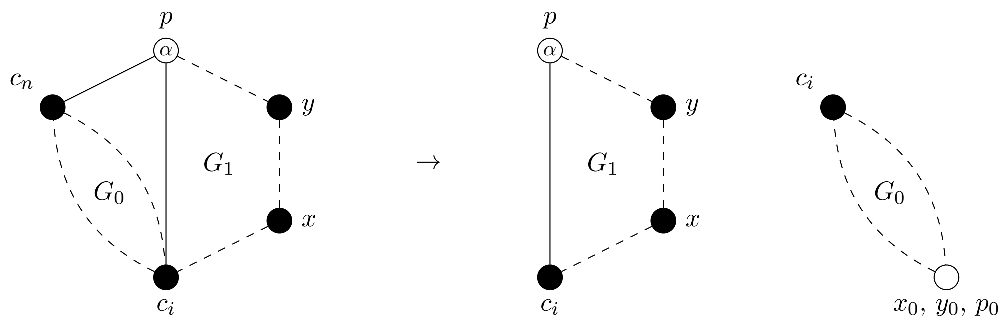

Figure 3.7 The case $x \neq p$, $y \neq p$, and $c_i \in V(C[x, p))$ (shown is the case $x \neq c_i$).

Suppose $x \neq p$, $y \neq p$, and $c_i \in V(C[x, p))$. In this case G_1 must exist, so first apply the inductive hypothesis to choose G_1 from L with $x_1 = x$, $y_1 = y$, and $p_1 = p$. Since $\alpha \notin L(v)$ for any $v \in V(C[x, p))$ we may apply the inductive hypothesis to choose G_0 from L_0 with $x_0 = y_0 = p_0 = c_i$. Note that c_i was precolored from our choosing of G_1 and recieves no same color neighbors in G_0 so the combined coloring forms a path choosing of G from L.

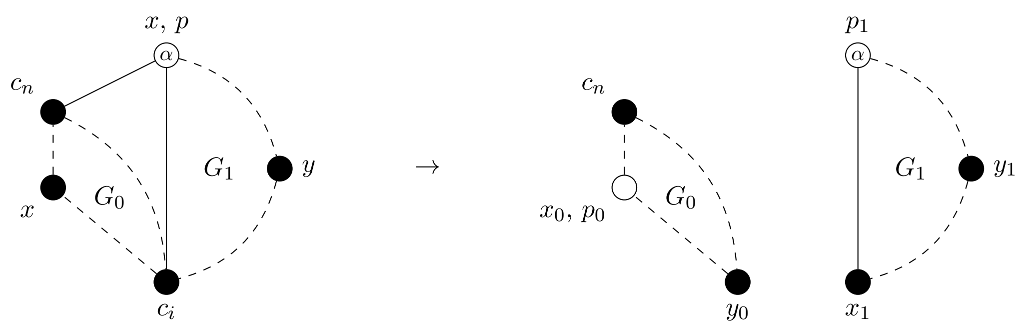

Figure 3.8 The case $x \neq p$, $y \neq p$, and $c_i \in V(C(y, x))$.

Suppose $x \neq p$, $y \neq p$, and $c_i \in V(C(y, x))$. In this case G_1 must exist. We apply the inductive hypothesis to color G_0 from L_0 with $x_0 = p_0 = x$ and $y_0 = c_i$. Then apply the inductive hypothesis again to choose G_1 from L with $x_1 = c_i$, $y_1 = y$, and $p_1 = p$. Notice c_i recieves at most one same color neighbor in each G_0 and G_1.

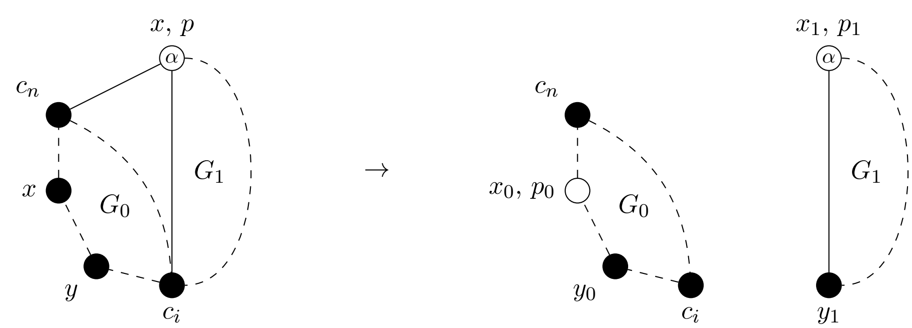

Figure 3.9 The case $x \neq p$, $y \neq p$, and $c_i \in V(C(p, y])$ (shown is the case $y \neq c_i$ and $\alpha \notin L(c_i)$).

Finally, suppose $x \neq p$, $y \neq p$, and $c_i \in V(C(p, y])$. If $\alpha \in L(c_i)$ we set $p_0 = c_i$ and color c_i with α. Otherwise, set $p_0 = c_1$ and color it with the first color in $L_0(c_1)$, noting $|L_0(c_i)| \geq 2$ in this case.

We then apply the inductive hypothesis to choose G_0 from L_0 with $x_0 = x$, $y_0 = y$, and p_0. If G_1 exists, we again apply the inductive hypothesis to choose G_1 from L with $x_1 = p_1 = p$ and $y_1 = c_i$. Note that c_i was precolored by our choosing of G_0. If $p_0 = c_i$, c_i recieves at most one same color neighbor in G_0 and the single same color neighbor p in G_1. Furthermore, p will recieve no same color neighbor in G_1 other than c_i. If $p_0 \neq c_i$, then $y_1 = c_i$ is immediatley prior to $p_1 = p$ and $\alpha \notin L(c_i)$. Thus c_i will recieve no same color neighbors in G_1. □

Implementation

To implement the coloring procedure described in Theorem 2 we must be able to efficiently achieve each of the following:

1. maintain the current subgraph G and its outer face C;
2. remove a single vertex $v \in V(C)$ maintaining the new outer face of $G - v$;
3. split G into two subgraphs along an edge bridging the outer face;
4. for any $v \in V(C)$, determine the region of C that v lies in with respect to x, y, and p.

Each vertex will marked with a *state* indicating whether it is currently interior, uncolored on the outer face, or colored. Since G is a plane graph we are provided with an embedded ordering of incident edges for each vertex. For each vertex we will maintain a pair of indices into this embedded ordering of edges, initialized with the start and end of the ordered lists. When removing a vertex $v \in C$ from G we must simply remove v from the incidence list of each $u \in N(v)$. Since v is on the outer face, it will be the the first or last index

References

[1] Hartman, C., ”Extremal problems in graph theory,” Ph.D. thesis, Department of Mathematics, University of Illinois at Urbana-Champaign, 1997.

[2] Skrekovski, R., ”List improper colourings of planar graphs,” *Combinatorics, Probability and Computing*, vol. 8, pp. 293-299, 1999.

[3] Eaton, N., N. Hull, ”Defective list colorings of planar graphs,” *Bulletin of the Institute for Combinatorics and its Applications*, 1999.

[4] Poh, K., ”On the linear vertex-arboricity of a planar graph,” *Journal of Graph Theory*, vol. 14, pp. 73-75, 1990

[5] Kant, G., ”Algorithms for drawing planar graphs,” Ph.D. thesis, Department of Computer Science, Utrecht University, 1993.

[6] Tarjan, R., ”Depth-first search and linear graph algorithms,” SIAM Journal on Computing, vol. 1, pp. 146-160, 1972.

[6] Tarjan, R., ”Depth-first search and linear graph algorithms,” SIAM Journal on Computing, vol. 1, pp. 146-160, 1972.