A New Proof of Berge's Strong Path Partition Conjecture for Acyclic Digraphs

Irith Ben-Arroyo Hartman

Abstract. Berge's elegant strong path partition conjecture from 1982 extends the Greene-Kleitman Theorem and Dilworth's Theorem for all digraphs. The conjecture is known to be true for all digraphs for $k = 1$ by the Gallai-Milgram Theorem, and for $k > 1$ only for acyclic digraphs. We present a simple algorithmic proof for $k = 1$ which naturally extends to a new algorithmic proof for acyclic digraphs for all $k \geq 1$. We conclude with some ideas for extending the algorithm for all digraphs.

1. Introduction and terminology

Let $G = (V, E)$ be a directed graph containing no loops or multiple edges, defined by a set V of vertices and a set $E \subseteq V \times V$ of directed edges. A *path* P in G is a sequence of distinct vertices $(v_1, v_2, \ldots, v_l)$ such that $(v_i, v_{i+1}) \in E$, for $i = 1, 2, \ldots, l-1$. The set of vertices $\{v_1, v_2, \ldots, v_l\}$ of a path $P = (v_1, v_2, \ldots, v_l)$ is denoted by $V(P)$, and the set of edges by $E(P)$. The *cardinality* of P, denoted by $|P|$, is $|V(P)|$. For any path in G, (directed or undirected) we denote the first vertex by $in(P)$ and the last vertex by $ter(P)$. For a path P and $x \in V(P)$ we denote by $P_-(x)$ the vertex that precedes x on P, $P_+(x)$ the vertex that follows x on P, and $ter_-(P)$ denotes the vertex on P that precedes the terminal vertex of P. For a set of vertex disjoint paths $\mathcal{P}$, we write $V[\mathcal{P}] = \bigcup\{V(P); P \in \mathcal{P}\}$, $E[\mathcal{P}] = \bigcup\{E(P); P \in \mathcal{P}\}$, $in[\mathcal{P}] = \{in(P); P \in \mathcal{P}\}$, and $ter[\mathcal{P}] = \{ter(P); P \in \mathcal{P}\}$. We denote by $ter_-[\mathcal{P}] = \{ter_-(P); P \in \mathcal{P}\}$. If x is on a path $P \in \mathcal{P}$, then $\mathcal{P}_-(x)$ denotes the unique vertex which precedes x on P, unless x is the initial vertex of P, in which case $\mathcal{P}_-(x)$ is undefined. Similarly, $\mathcal{P}_+(x)$ denotes the unique vertex which follows x on P. We let $|V(G)| = n$. For other concepts, we adopt the terminology of [6].

A family $\mathcal{P}$ of paths is called a *path partition* of G if its members are vertex disjoint and $\cup\{V(P); P \in \mathcal{P} = V\}$. A directed graph, or for brevity, *digraph*, may have many path partitions. The *trivial* path partition, where every path is a single vertex, is an example of a path partition. For each nonnegative integer k, the k-norm $|\mathcal{P}|_k$ of a path partition $\mathcal{P} = \{P_1, \ldots, P_m\}$ is defined by

$$|\mathcal{P}|_k = \sum_{i=1}^{m} min\{|P_i|, k\}. \tag{1.1}$$

A partition which minimizes $|\mathcal{P}|_k$ is called *k-optimal.* Note that a 1-optimal path partition is a partition that contains a minimum number of paths. Denote by $\mathcal{P}^+$ the set of paths in $\mathcal{P}$ of cardinality at least k, (which we also call *long* paths), and by $\mathcal{P}^0$ the set of paths in $\mathcal{P}$ of cardinality less than

k, (called *short* paths). Then equation (1.1) can be alternatively written as

$$|\mathcal{P}|_k = \sum_{i=1}^{m} min\{|P_i|, k\} = k|\mathcal{P}^+| + |V[\mathcal{P}^0]|$$

For convenience, we draw all paths in a path partition vertically downwards, i.e. $\mathcal{P}^-(x)$ is drawn above x (see Figure 1(a)).

A *k-colouring* is a family $\mathcal{C}^k = \{C_1, C_2, \ldots, C_k\}$ of k disjoint independent sets called *colour classes.* (Some of the colour classes may be empty). The *cardinality* of a k-colouring is the sum of the sizes of the colour classes, i.e., $|\mathcal{C}^k| = \sum_{i=1}^{k} |C_i|$ and $\mathcal{C}^k$ is said to be *optimal* if $|\mathcal{C}^k|$ is as large as possible. A path partition $\mathcal{P}$ and a k-colouring $\mathcal{C}^k$ are *orthogonal* if every path P_i in $\mathcal{P}$ meets $min\{|P_i|, k\}$ different colour classes of $\mathcal{C}^k$. Note that this is the maximum number of different colour classes that a path can intersect a k-colouring.

Conjecture 1.1 (Berge's strong path partition conjecture [3]). Let G be a digraph and let k be a positive integer. Then for every k-optimal path partition $\mathcal{P}$ there exists a k-colouring orthogonal to it.

The conjecture holds for $k = 1$ for all digraphs by the Gallai-Milgram Theorem [14]. Berge's strong path partition conjecture has also been proved for acyclic digraphs (see [8], [18], [2], and [1]. It is not difficult to see, as was shown in [3], that the conjecture is true when $k \geq \lambda$, where λ is the cardinality of the longest path in G, and when the k-optimal path partition contains only short paths, (i.e. paths of cardinality less than k). In [1] it was proved that the conjecture holds also in the case that the given k-optimal path partition contains only long paths, i.e. $\mathcal{P} = \mathcal{P}^+$.

Denote by $\alpha_k(G)$ the cardinality of an optimal k-colouring in G, and by $\pi_k(G)$ the k-norm of a k-optimal path partition in G. Conjecture 1.1 implies the following:

Conjecture 1.2 (Weak path partition conjecture -Linial [17]). For any digraph G and positive integer k, $\alpha_k(G) \geq \pi_k(G)$.

If G is transitive and acyclic (i.e., the graph of a partially ordered set), then Green-Kleitman's theorem [15] states that $\alpha_k(G) = \pi_k(G)$, implying Conjecture 1.2.

In this paper, we will state and give a new and easy algorithmic proof of the Gallai-Milgram Theorem. We will also give a new algorithmic proof of Conjecture 1.1 for all k for acyclic digraphs which imitates and generalizes the proof of the Gallai-Milgram Theorem. The ideas in this proof may be the basis for a proof for all digraphs.

2. Proof for $k = 1$, the Gallai-Milgram Theorem

We begin with some more graph-theoretical notations and definitions: For a subgraph $H \subseteq G$ we denote by $d_H^-(x)$ and $d_H^+(x)$ the indegree and outdegree of x in H, respectively.

Definition 2.1 (Undirected trail, Improving trail). An *undirected trail* Q in G is a sequence $Q = (v_0, e_1, v_1, e_2, v_2, \ldots, e_l, v_l)$ such that, for each $1 \leq i \leq l$, either $e_i = (v_{i-1}, v_i) \in E(G)$ or $e_i = (v_i, v_{i-1}) \in E(G)$, and all the edges are distinct. We assign a direction to Q from v_0 to v_k, so that if $e_i = (v_{i-1}, v_i)$ then e_i is considered a *forward* edge, and if $e_i = (v_i, v_{i-1})$ then e_i is considered a *backward* edge.

Given a path partition $\mathcal{P}$, an undirected trail $Q = (v_0, e_1, v_1, e_2, v_2, \ldots, e_l, v_l)$ is *1-improving*, or simply, *improving* relative to $\mathcal{P}$ if the following conditions hold:

1. All forward edges of Q are in $E(G) - E[\mathcal{P}]$, all backward edges are in $E[\mathcal{P}]$, and the forward and backward edges alternate.
2. For every vertex $v \in V(Q)$ there exists at most one forward edge $(u, v) \in E(Q)$, entering v, and at most one forward edge $(v, w) \in E(Q)$, leaving v.
3. $v_0 \in ter[\mathcal{P}]$ and e_1 is a forward edge.
4. $v_l \in in[\mathcal{P}]$ and e_l is a forward edge.
5. Every internal vertex v_i of Q, $1 \leq i \leq l-1$, has $d^-_{\mathcal{P}}(v_i) = 1$.

We note that condition (2) follows from condition (1), and the fact that all edges in a trail are distinct.

We denote by $\mathcal{P} \bigoplus Q$ the spanning subgraph of G containing edges in the symmetric difference $E[\mathcal{P}] \bigoplus E(Q) = E[\mathcal{P}] \cup E(Q) \setminus (E[\mathcal{P}] \cap E(Q))$.

Lemma 2.2. Let Q be an improving trail relative to a path partition $\mathcal{P}$, and let $\mathcal{P}' = \mathcal{P} \bigoplus Q$. Then the following holds:

- **(a):** The subgraph $\mathcal{P}'$ is a spanning subgraph of G containing disjoint paths and cycles.
- **(b):** If $\mathcal{P}'$ contains no cycles, then $\mathcal{P}'$ is a path partition with $|\mathcal{P}'|_1 < |\mathcal{P}|_1$.

Proof: Let $Q = (v_0, e_1, v_1, e_2, v_2, \ldots, e_l, v_l)$.

- **(a):** Since every forward edge in Q is followed by a backward edge, except for the last edge e_l (by conditions (1) and (5) of Definition 2.1), it follows that for every $v \neq v_0, v_l$, $d^-_{\mathcal{P}'}(v) = d^-_{\mathcal{P}}(v) \leq 1$, $d^+_{\mathcal{P}'}(v) = d^+_{\mathcal{P}}(v) \leq 1$. For $v_0 \in ter[\mathcal{P}]$, we have $d^+_{\mathcal{P}'}(v_0) = 1$ by condition (3), and for $v_l \in in[\mathcal{P}]$, $d^-_{\mathcal{P}'}(v_l) = 1$, by condition (4). This proves (a).
- **(b):** By the proof of (a), $in[\mathcal{P}'] \subset in[\mathcal{P}]$ and $ter[\mathcal{P}'] \subset ter[\mathcal{P}]$, and $\mathcal{P}'$ has one fewer initial vertex and terminal vertex than $\mathcal{P}$. Therefore, if $\mathcal{P}'$ is acyclic, it has one fewer path than $\mathcal{P}$, proving the Lemma.

■

Definition 2.3 (1-transversal). Let $\mathcal{P} = \{P_1, \ldots, P_m\}$ be a path partition of G. A set of vertices $X \subseteq V$ is a *1-transversal*, (or a *transversal*), of $\mathcal{P}$ if for each $1 \leq i \leq m$, $|X \cap P_i| = 1$.

Conjecture 1.1 for $k = 1$ is equivalent to showing that every optimal path partition $\mathcal{P}$ contains a 1-transversal of $\mathcal{P}$ which is an independent set. A transversal which is an independent set is called an *independent transversal*.

Theorem 2.4 (Gallai-Milgram [14]). Let $\mathcal{P}$ be an optimal path partition in a directed graph G. Then G contains an independent transversal of $\mathcal{P}$.

Proof:

Let $\mathcal{P}$ be any path partition in G. We will demonstrate an algorithm that either finds an independent transversal of $\mathcal{P}$, or a path partition $\mathcal{P}'$ satisfying $|\mathcal{P}'| < |\mathcal{P}|$.

Initially, we define X to be $ter[\mathcal{P}]$, which is clearly, a transversal of $\mathcal{P}$. If X is an independent set, then we are done. Otherwise, there exists an edge $e = (x_1, x_2)$, where $x_1, x_2 \in X$. We use this edge to update X by replacing x_2 by $\mathcal{P}_-(x_2)$, the vertex that precedes x_2 in $\mathcal{P}$. We continue checking whether the new set X is independent. If not, then we continue to 'climb up' the vertices in $\mathcal{P}$, unless the vertex $P_-(x_2)$ does not exist, because x_2 was an initial vertex of a path in $\mathcal{P}$. In this case, we find an improving trail Q such that $\mathcal{P}' = \mathcal{P} \bigoplus Q$ contains no cycles, and hence, by Lemma 2.2, $\mathcal{P}'$ is a path partition satisfying $|\mathcal{P}'| < |\mathcal{P}|$.

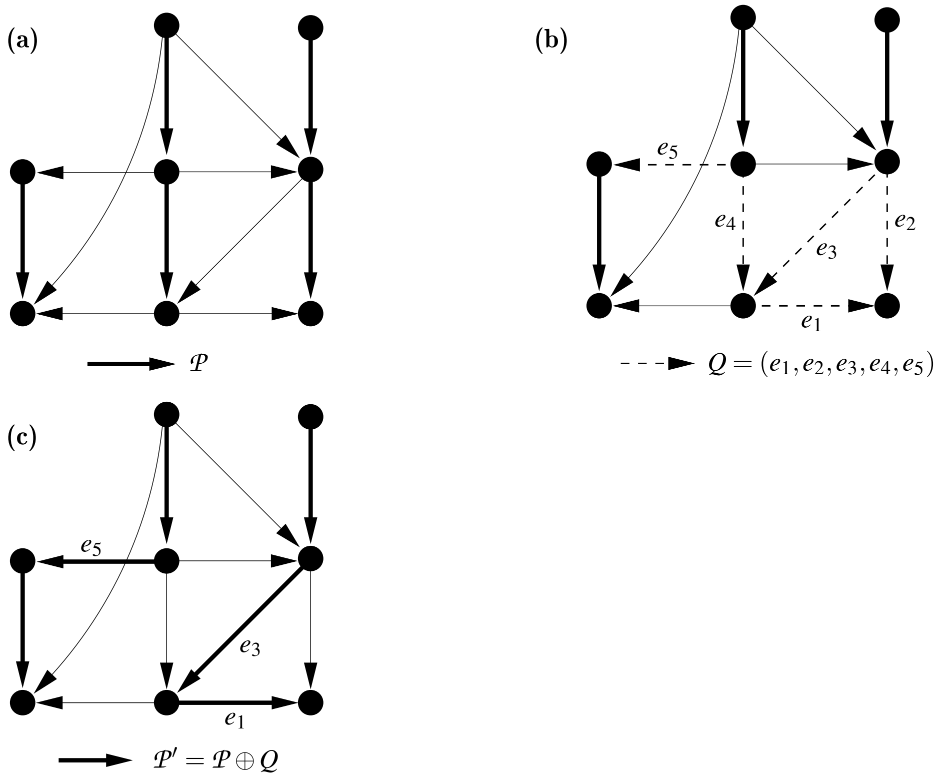

FIGURE 1. (a) A path partition $\mathcal{P}$. (b) An improving trail Q relative to $\mathcal{P}$. (c) The resultant path partition $\mathcal{P}'$.

The improving trail Q, is found by tracing the steps back from $x_2 \in in[\mathcal{P}]$ to x_1, down to $P_+(x_1)$ and so forth, until we reach a terminal vertex in $\mathcal{P}$. In order to trace Q, every time we update X, we remember the predecessor, $p[x]$ of x, the vertex that caused x to be added to X. For ease of notation, if x, y are elements in a set X, we use the short notation of $X - x + y$ to denote $X - \{x\} \cup \{y\}$. The details of the algorithm are given below:

2.1. Algorithm

initialize
    Transversal: $X \leftarrow ter[\mathcal{P}]$
    Predecessor: $p[x] \leftarrow x$, for all $x \in X$
while (there exists $e = (x_1, x_2), x_1, x_2 \in X$) **do**
  $p[x_2] \leftarrow x_1$
  if $(x_2 \notin in[\mathcal{P}])$
    $X \leftarrow X - x_2 + \mathcal{P}_-(x_2)$
    $p[\mathcal{P}_-(x_2)] \leftarrow x_2$
  else (i.e. $x_2 \in in[\mathcal{P}]$)
    backtrack from x_2 through the predecessors to retrieve an improving trail Q
    $\mathcal{P}' \leftarrow \mathcal{P} \bigoplus Q$
    stop
X is an independent set

Note that in the course of the algorithm, the set X is updated only in line 7, where a vertex in X is pushed up the path to be replaced by its predecessor. The Claim below easily follows:

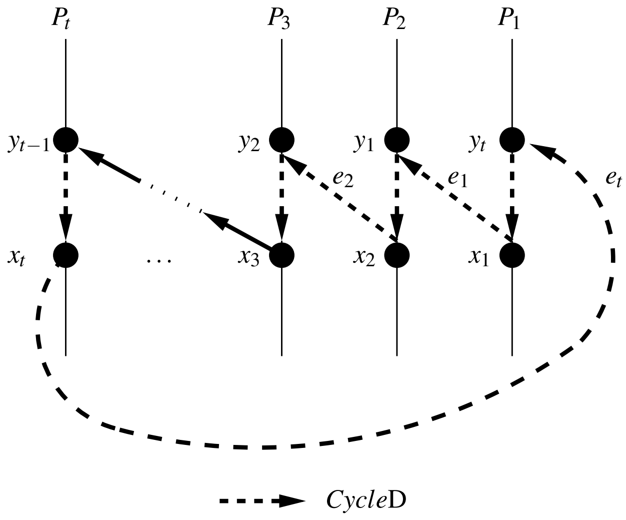

FIGURE 2

Claim 2.5 ("Climbing Transversal"). Let $P = (v_1, v_2, \ldots, v_l) \in \mathcal{P}$. If, at some stage of the algorithm, $v_i \in X$, then at any subsequent stage, if $v_j \in X$, then $j \leq i$.

The next Claim is sufficient to proving Theorem 2.4, since an optimal path partition $\mathcal{P}$ would have no path partition $\mathcal{P}'$ with $|\mathcal{P}'| < |\mathcal{P}|$.

Claim 2.6. Let $\mathcal{P}$ be some path partition in a digraph G. Algorithm 2.1 either finds an independent transversal X of $\mathcal{P}$, or another path partition $\mathcal{P}'$ with $|\mathcal{P}'| < |\mathcal{P}|$.

To prove the claim we have to show that $\mathcal{P}' = \mathcal{P} \bigoplus Q$ contains no cycles, and hence is a path partition with $|\mathcal{P}'| < |\mathcal{P}|$. Let $Q = (z_1, z_2, \ldots, z_t)$, where $z_1 \in ter[\mathcal{P}]$ and $z_t \in in[\mathcal{P}]$.

Assume, by contradiction, that $\mathcal{P}'$ contains a cycle D. Then $D = (e_1 \overline{P_2} e_2 \overline{P_3} \ldots e_t \overline{P_1})$, where $\overline{P_i}$ is a subpath of $P_i \in \mathcal{P}$ ($\overline{P_i}$ may be either a single vertex, or a non-trivial subpath of P_i or equal to P_i), see Figure 2. Without loss of generality, assume that $e_1 = (x_1, y_1)$ is the first edge of D encountered by the Algorithm. Let $e_2 = (x_2, y_2)$, where possibly $x_2 = y_1$. Then when e_1 is discovered, vertex y_1 gets 'bumped up' in line 7 to its predecessor $\mathcal{P}_-(y_1)$ (which exists since e_1 was the first edge of D). At a later stage of the algorithm the edge e_2 is discovered, which is impossible since x_2 is strictly below $\mathcal{P}_-(y_1)$, thus contradicting Claim 2.5. Hence $\mathcal{P}'$ contains no cycles.

This proves Claim 2.6 and Theorem 2.4. ∎

3. Proof of Berge's Conjecture for Acyclic Digraphs

The proof for all k generalizes the proof and algorithm given in the previous section. We remind the reader that in this case, for a given path partition $\mathcal{P}$, the set of all ('long') paths in $\mathcal{P}$ of cardinality at least k is denoted by $\mathcal{P}^+$, and the set of ('short') paths of cardinality less than k is denoted by $\mathcal{P}^0$. We assume now, that every short path in $\mathcal{P}$ is of cardinality one. This can be done by breaking up every short path of cardinality greater than one into single vertex paths.

Definition 3.1 (k-Alternating Trail). Given a path partition $\mathcal{P}$, an undirected trail $Q = (v_0, e_1, v_1, e_2, v_2, \ldots, e_l, v_l)$ is *k-alternating* relative to $\mathcal{P}$, if conditions 1-3 below hold:

1. All forward edges of Q are in $E(G) - E[\mathcal{P}^+]$, all backward edges are in $E[\mathcal{P}^+]$, and every forward edge (u, v), where $v \in V[\mathcal{P}^+]$ is followed by a backward edge, unless $v \in in[\mathcal{P}^+]$ and $v = v_l$.
2. There can be at most $k-1$ consecutive backward edges in Q, unless the first one follows a forward edge, in which case, there can be at most k consecutive backward edges.
3. For every vertex $v \in V(Q)$ there exists at most one forward edge $(u, v) \in E(Q)$, and at most one forward edge $(v, w) \in E(Q)$.

Note that conditions (1) and (2) above correspond to condition (1) of Definition 2.1, and condition (3) is identical to condition (2) of Definition 2.1.

Definition 3.2 (Prim and Proper k-Alternating Trail). A k-alternating trail $Q = (v_0, e_1, v_1, e_2, v_2, \ldots, e_l, v_l)$ is *proper* if either:

(a): $v_0 \in ter[\mathcal{P}^+]$
or
(b): $v_0 \in V[\mathcal{P}^0]$

A k-alternating trail is *prim* if either:

(1): $v_l \in in[\mathcal{P}^+]$
or
(2): $v_l \in V[\mathcal{P}^0]$

We say that a prim and proper k-alternating trail is of type **(a-1)** if it is of type **(a)** and type **(1)** defined above. Types **(a-2), (b-1)** and **(b-2)** are similarly defined.

Note that if $e_{i+1} = (v_i, v_{i+1})$ is a backward edge of Q, where $v_i, v_{i+1} \in V[\mathcal{P}^+]$, and e_{i+1} does not follow a forward edge, then v_i is an isolated vertex in $\mathcal{P}' = \mathcal{P} \bigoplus Q$. Also, if $v \in V[\mathcal{P}^0]$ is on a forward edge leaving v or entering v, then it is joined to a long path in $\mathcal{P}'$.

We define, therefore, the *weight* of a k-alternating trail Q, denoted by $w(Q)$ as the number of backward edges which do not follow forward edges in Q, minus the number of vertices in $V[\mathcal{P}^0]$ which meet Q. Note that the weight of Q is, in fact, $w(Q) = |V[\mathcal{P}'^0]| - |V[\mathcal{P}^0]|$, where $\mathcal{P}' = \mathcal{P} \bigoplus Q$.

Definition 3.3 (k-Improving Trail). A prim and proper k-alternating trail Q is *improving* if one of the following conditions holds:

1. Q is of type **(a-1)** with $w(Q) \leq k-1$.
2. Q is of type **(a-2)**, **(b-1)**, or **(b-2)** with $in(Q) = ter(Q)$, and $w(Q) \leq -1$.
3. Q is of type **(b-2)**, Q is not a cycle, and $w(Q) \leq -(k+1)$.

The next Lemma is the analogue of Lemma 2.2 for all k:

Lemma 3.4. Let Q be a k-improving trail relative to a path partition $\mathcal{P}$, and let $\mathcal{P}' = \mathcal{P} \bigoplus Q$. Then the following holds:

(a): The subgraph $\mathcal{P}'$ is a spanning subgraph of G containing disjoint paths and cycles.
(b): If $\mathcal{P}'$ contains no cycles, then $\mathcal{P}'$ is a path partition with $|\mathcal{P}'|_k < |\mathcal{P}|_k$.

Proof:

(a): Assume that $Q = (v_0, e_1, v_1, e_2, v_2, \ldots, e_l, v_l)$, where $v_0 \in ter[\mathcal{P}^+] \cup V[\mathcal{P}^0]$ and $v_l \in in[\mathcal{P}^+] \cup V[\mathcal{P}^0]$. Also note that $d^+_{\mathcal{P}}(v_0) = 0$. Therefore, by conditions (1) and (3) of Definition 3.1, for every $v \neq v_0, v_l$, $v \in V[\mathcal{P}^+]$, $d^-_{\mathcal{P}'}(v) \leq d^-_{\mathcal{P}}(v) = 1$, $d^+_{\mathcal{P}'}(v) \leq d^+_{\mathcal{P}}(v) = 1$. For $v_0 \in ter[\mathcal{P}^+]$, we have $d^+_{\mathcal{P}'}(v_0) = 1$, and for $v_k \in in[\mathcal{P}^+]$, $d^-_{\mathcal{P}'}(v_k) = 1$. Vertices in $V[\mathcal{P}^0]$ also receive indegree and outdegree at most one in $\mathcal{P}'$ by Definition 3.1-(3). We have shown that every vertex in $\mathcal{P}'$ has indegree and outdegree at most one, proving (a).

(b): We consider the different types of improving trails, according to their beginning and ending vertices:

Type(a-1): : In this case there is one fewer long path in $\mathcal{P}'$ than in $\mathcal{P}$, but since $w(Q) \leq k-1$, we have $|\mathcal{P}'^+| = |\mathcal{P}^+| - 1$ and $|V[\mathcal{P}'^0]| \leq |V[\mathcal{P}^0]| + k - 1$.

Type(a-2) or (b-1): : Here $|\mathcal{P}'^+| = |\mathcal{P}^+|$, but since $w(Q) \leq -1$, $|V[\mathcal{P}'^0]| \leq |V[\mathcal{P}^0]| - 1$.

Type(b-2) : : If Q is a cycle then $|\mathcal{P}'^+| = |\mathcal{P}^+|$, and since $w(Q) \leq -1$, $|V[\mathcal{P}'^0]| = |V[\mathcal{P}^0]| - 1$. Otherwise, $|\mathcal{P}'^+| \leq |\mathcal{P}^+| + 1$, and $|V[\mathcal{P}'^0]| \leq |V[\mathcal{P}^0]| - (k+1)$.

Since $\mathcal{P}'$ is acyclic, it is obviously a path partition. In all the cases above, it can easily be seen that $|\mathcal{P}'|_k \leq |\mathcal{P}|_k - 1$.

■

Figure 3 depicts a 3-improving trail of type (a-1), and the resultant path partition.

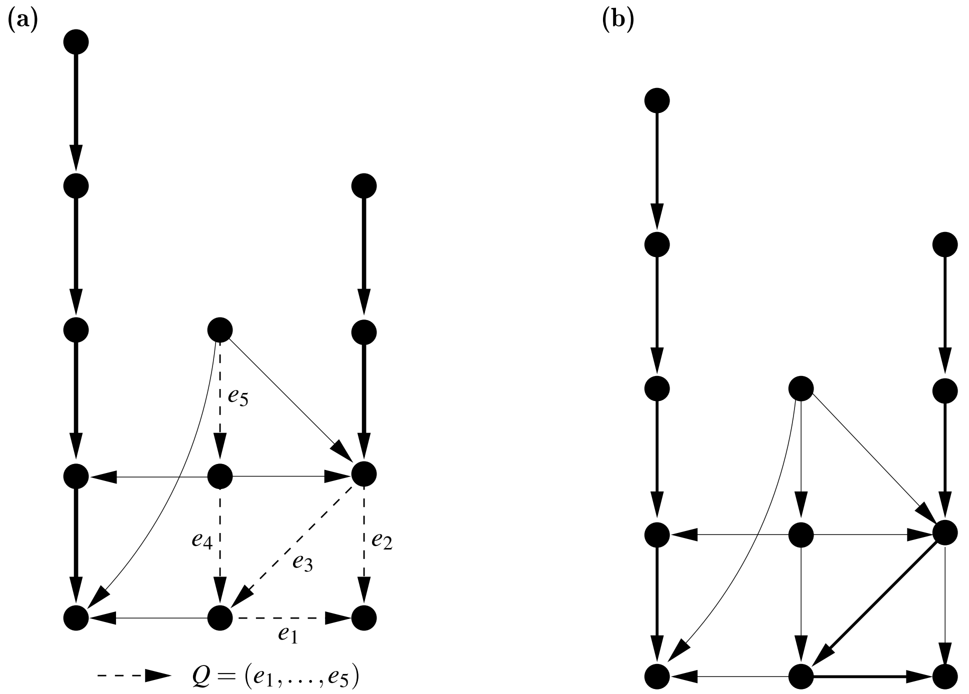

FIGURE 3. (a) $Q = (e_1, \ldots, e_5)$ a 3-improving trail (b) The resultant path partition $\mathcal{P}' = \mathcal{P}' \bigoplus Q$.

Definition 3.5 (k-transversal). Let $\mathcal{P} = \{P_1, \ldots, P_m\}$ be a path partition of G. A set of vertices $X = X_1 \cup X_2 \cup \ldots \cup X_k \subseteq V$, where $X_i \cap X_j = \emptyset$ for all $1 \leq i, j \leq k$, $i \neq j$ is a *k-transversal* of $\mathcal{P}$ if $V[\mathcal{P}^0] \subseteq X$ and for each $P_i \in \mathcal{P}^+$, $1 \leq i \leq k$, $1 \leq j \leq m$, $|X_i \cap P_j| = 1$.

Conjecture 1.1 is equivalent to showing that every k-optimal path partition $\mathcal{P}$ contains a k-transversal $X = X_1 \cup X_2 \cup \ldots \cup X_k$ to $\mathcal{P}$, where X_i for all $1 \leq i \leq k$ are independent sets, and for each $P \in \mathcal{P}^0$ and each $1 \leq i \leq k$, $V(P) \cap X_i \leq 1$. We use an algorithm which generalizes Algorithm 2.1. We remind the reader that for a disjoint set of paths $\mathcal{P}$ we denote by $ter_-[\mathcal{P}]$ the set of vertices in $\mathcal{P}$ preceding the terminal vertices of $\mathcal{P}$. Similarly, $ter_{--}[\mathcal{P}]$, or $ter_{-2}[\mathcal{P}]$ denote the set of vertices which precede the set $ter_-[\mathcal{P}]$ on $\mathcal{P}$, and by induction, $ter_{-i}[\mathcal{P}]$ denotes the vertices which precede the set $ter_{-(i-1)}[\mathcal{P}]$ on $\mathcal{P}$. We denote the vertex which follows v on $\mathcal{P}$ by $\mathcal{P}_+(v)$.

3.1. Description of Algorithm:

We initialize the sets $X_i, 1 \leq i \leq k$ as follows:

3.1.1. Initialize Transversals.

$$X_1 = ter[\mathcal{P}^+], X_2 = ter_-[\mathcal{P}^+], \ldots, X_i = ter_{-(i-1)}[\mathcal{P}^+], \ldots, X_k = ter_{-(k-1)}[\mathcal{P}^+] \cup V[\mathcal{P}^0] \tag{3.1}$$

In other words, initially, for every long path we colour the bottom k vertices with colours $1, 2, \ldots k$, and all vertices on short paths are coloured with colour k.

3.1.2. Updating Transversals. As long as there exists a set X_i which is not an independent set, we use the edge $e = (u, v)$ (where $u, v \in X_i$) to update the set X_i. There are two different criteria for updating X_i, depending whether the target vertex v, is on a long or short path in $\mathcal{P}$:

- **Case 1::** $v \in V[\mathcal{P}^+]$. Assume that $v \in P_j$. In this case v it gets 'bumped up' the path, i.e. replaced in the set X_i by its predecessor, as in Algorithm 2.1: $X_i \leftarrow X_i - v + \mathcal{P}_-(v)$. If the predecessor of v is a vertex x in X, then x also gets bumped up the path and replaced by its predecessor. The process continues until either a vertex x_1 is replaced by its predecessor which is not in X (**Case 1.1**), or $x_1 \in in[\mathcal{P}^+] \cap X_k$, i.e. x_1 has no predecessor, (**Case 1.2**). In this case , a k-improving trail Q is traced back from x_1.
- **Case 2::** $v \in V[\mathcal{P}^0]$. If $i > 1$ (**Case 2.1**) then v is relabeled as a vertex in X_{i-1}. Otherwise, $i = 1$, (**Case 2.2**) then v cannot be relabeled, and a k-improving trail Q is traced back from v.

In order to trace back the k-improving trail, every vertex x involved in the algorithm remembers its predecessor $p(x)$ defined as follows:

3.1.3. Initialize Predecessors: For each $v \in X_1 \cup V[\mathcal{P}^0]$, $p(v) \leftarrow v$.

For each $v \in V[\mathcal{P}^+]$, $v \in X_i$, $i > 1$, $p(v) \leftarrow \mathcal{P}_+(v)$. In other words, initially, the predecessor of a vertex on a long path in X_i is the vertex following it on the path which belongs to X_{i-1}.

3.1.4. Update Predecessors: For an edge $(u, v) \in X_i$, set $p(v) \leftarrow u$.

If v gets 'bumped up' the path as in Case 1.1, then set $p(\mathcal{P}_-(v)) \leftarrow v$.

3.2. The Algorithm

An outline of the algorithm is given below:

0. **input:** Graph $G = (V, E)$, path partition $\mathcal{P}$, integer $k \geq 1$.
1. **initialize**
2. Transversals: As in Section 3.1.1
3. Predecessors: As in Section 3.1.3
4. **while** (there exists $e = (u, v), u, v \in X_i$) **do**
5. $p[v] \leftarrow u$
6. **case 1:** $v \in V[\mathcal{P}^+]$ (Assume $v \in P_j$)
7. **case 1.1:** $in(P_j) \notin X_k$
8. Bump v up the path as in Case 1.1 of Section 3.1.2
9. Update predecessors as in Section 3.1.4
10. **case 1.2:** $in(P_j) \in X_k$
11. Backtrack from $in(P_j)$ to find Q
12. $\mathcal{P}' \leftarrow \mathcal{P} \bigoplus Q$
13. **stop**
14. **case 2:** $v \in V[\mathcal{P}^0]$
15. **case 2.1:** $i > 1$
16. $X_i \leftarrow X_i - v$
17. $X_{i-1} \leftarrow X_{i-1} + v$
18. **case 2.2:** $i = 1$
19. Backtrack to find Q
20. $\mathcal{P}' \leftarrow \mathcal{P} \bigoplus Q$
21. **stop**
22. X_i, $1 \leq i \leq k$ are independent sets

Theorem 3.6. Let $\mathcal{P}$ be a path partition, $k \geq 1$, and assume that every path in $\mathcal{P}^0$ is of cardinality one. Then Algorithm 3.2 finds either a path partition $\mathcal{P}'$ with $|\mathcal{P}'|_k < |\mathcal{P}|_k$ or a k-colouring orthogonal to $\mathcal{P}$.

Proof: If the algorithm stops at line 22 then the sets $\{X_1, X_2, \ldots, X_k\}$ form a k-colouring orthogonal to $\mathcal{P}$. Otherwise, a trail Q is found (Cases 1.2 and 2.2 of the Algorithm). Note that Q is a k-alternating trail. In addition, Q is a proper trail by the initialization of the predecessors, and prim of type (1) (in Case 1.2) or type (2) (in Case 2.2).

Observation: A trail Q with $in(Q) \in X_i$ and $ter(Q) \in X_j$ has weight $w(Q) = j - i$ if $in(Q), ter(Q) \in V[\mathcal{P}^+]$, weight $w(Q) = j - i - 1$ if one of $in(Q)$ or $ter(Q)$ is in $V[\mathcal{P}^0]$, and weight $w(Q) = j - i - 2$ if both $in(Q)$ and $ter(Q)$ are in $V[\mathcal{P}^0]$.

Now, if Q is of type (a) then $in(Q) \in V[\mathcal{P}^+] \cap X_1$ and if is of type (b) then $in(Q) \in V[\mathcal{P}^0] \cap X_k$. Similarly, if Q is of type (1) (Case 1.2 of Algorithm) then $ter(Q) \in V[\mathcal{P}^+] \cap X_k$, and of type (2)(Case 2.2 of Algorithm) then $ter(Q) \in V[\mathcal{P}^0] \cap X_1$. Using the Observation above, it can easily be checked that in all cases, Q has a weight as defined in Definition 3.3 and is therefore, a k-improving trail. By Lemma 3.4 this implies that $\mathcal{P}' = \mathcal{P} \bigoplus Q$ is a path partition with $|\mathcal{P}'|_k < |\mathcal{P}|_k$. ■

In Theorem 3.6 we assumed that all paths in $\mathcal{P}^0$ are of cardinality one. If we do not assume that, then Algorithm 3.2 may find a k- colouring where not all colours on a short path are distinct. To avoid this situation we make a slight modification to the Algorithm by forcing all the colours that meet a short path to be strictly decreasing. This can be done by replacing the condition in line 4 by the following modified condition:

4. **while** $\bar{(}$there exists $e = (u, v), u, v \in X_i$ or $e = (u, v)$ where $u, v \in V[\mathcal{P}^0], u \in X_j, v \in X_i$ and $i \geq j$) **do**

In the latter case, (i.e. there exists $e = (u, v)$ where $u, v \in V[\mathcal{P}^0], u \in X_j, v \in X_i$ and $i \geq j$) we update the colour of vertex v to be $j - 1$. More formally:

$X_i \leftarrow X_i - v$
$X_{j-1} \leftarrow X_{j-1} + v$.

Otherwise, there are no other changes to the algorithm. Thus the following is proved:

Theorem 3.7. Let G be an acyclic digraph. Let $\mathcal{P}$ be a path partition, $k \geq 1$. Then the modified Algorithm described above finds either a path partition $\mathcal{P}'$ with $|\mathcal{P}'|_k < |\mathcal{P}|_k$ or a k-colouring orthogonal to $\mathcal{P}$.

4. A Note on the General Case

If G is an arbitrary digraph with cycles, then Algorithm 3.2 may create a k-improving trail Q, where $\mathcal{P}' = \mathcal{P} \bigoplus Q$ contains a cycle, and hence does not have a smaller k-norm than $\mathcal{P}$. In the proof of the case $k = 1$ we used Claims 2.5 and 2.6. The k-analogue of Claim 2.5 is as follows:

Lemma 4.1 ("Climbing Transversals"). Let $\mathcal{P}$ be a path partition, and let $P = (v_1, v_2, \ldots, v_l)$ be a long path in $\mathcal{P}$, $l \geq k$. Then the following holds:

1. At any stage of Algorithm 3.2, as long as Q is not discovered, for each $1 \leq j \leq k$, there exist unique vertices $v_{j_1} \in X_j$, and $j_1 < i_1$ if $i > j$. In other words, each long path meets all k colour classes, and they remain in descending order.
2. If at some stage of the algorithm if $v_j \in X_j$, then at any subsequent stage, if $v_i \in X_j$ then $i \leq j$.
3. Throughout the algorithm, $V[\mathcal{P}^0] \subseteq X_1 \cup X_2 \cup \ldots \cup X_k = X$.

Theorem 4.2. Let $\mathcal{P}$ be a k-optimal path partition, and assume that $\mathcal{P}$ contains only 'long' paths (i.e. $\mathcal{P} = \mathcal{P}^+$). Then Algorithm 3.2 finds a k-colouring orthogonal to $\mathcal{P}$.

Outline of Proof:

It is sufficient to show that if, by contradiction, an improving trail Q is found, then $\mathcal{P}' = \mathcal{P} \bigoplus Q$ contains no cycles, contradicting the fact that $\mathcal{P}$ be k-optimal. The proof uses Lemma 4.1 and is similar to the proof of Claim 2.6: As in the proof of Claim 2.6, when the edge $e_2 = (x_2, y_2)$ is discovered which is strictly below $\mathcal{P}_-(y_1)$, vertex x_2 must belong to a lower colour class than y_1. But this is impossible, because having no vertices in $\mathcal{P}^0$, we have $w(Q(z_1, \ldots, y_1)) \leq w(Q(z_1, \ldots, y_1, \ldots, x_2))$. Hence $\mathcal{P}'$ contains no cycles and Berge's Conjecture is proved in this case. This result was previously proved in [1].

Figures 4(a) and (b) below are an example where Algorithm 3.2 outputs a 3-improving trail Q relative to $\mathcal{P}$, and $\mathcal{P}' = \mathcal{P} \bigoplus Q$ contains a cycle. The trail is discovered when the algorithm considers edges in the order: $(e_3, e_5, e_6, e_{10}, e_{11}, e_{12}, e_8)$. Note that each of the edges considered are of the type $(u, v), u, v \in X_i$. (e.g. e_3, e_5 and e_{10} connect two vertices in X_3, e_6, e_8, e_{11} and e_{12} connect two vertices in X_2.) However, if the edges in the algorithm are considered in a different order, such as: (e_{10}, e_{11}, e_{12}), (see Figure 4 (c)), then Algorithm 3.2 finds a 3-colouring orthogonal to $\mathcal{P}$. This gives hope to the fact that setting an appropriate order on the edges under consideration, may always yield either a k-colouring, or a path partition with a smaller k-norm, which will prove Conjecture 1.1. We leave this as an open question.

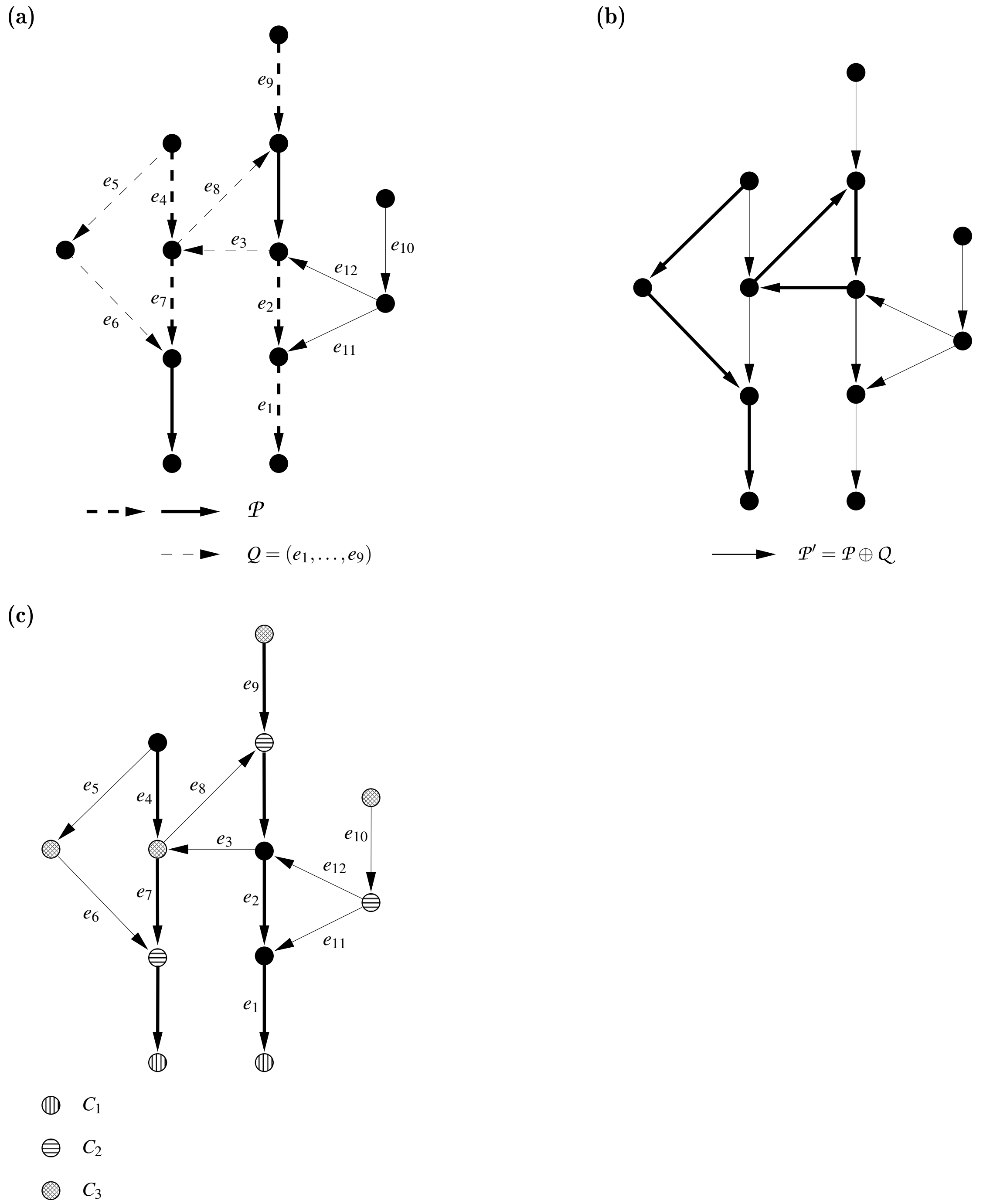

FIGURE 4. (a) $Q = (e_1, \ldots, e_8)$ is a 3-improving trail when the edges of the algorithm are considered in the order $(e_3, e_5, e_6, e_{10}, e_{11}, e_{12}, e_8)$. (b) The resultant path partition $\mathcal{P}' = \mathcal{P}' \bigoplus Q$. (c) An orthogonal 3-colouring $\mathcal{C}^3 = \{C_1, C_2, C_3\}$.

References

[1] R. Aharoni, and I. Ben-Arroyo Hartman, On Greene-Kleitman's theorem for general digraphs *Discrete Math.* 120 (1993), 13-24.

[2] R. Aharoni, I. Ben-Arroyo Hartman and A. J. Hoffman, Path partitions and packs of acyclic digraphs, *Pacific Journal of Mathematics* (2) vol.118 (1985), 249-259.

[3] C. Berge, k-optimal partitions of a directed graph, *Europ. J. Combinatorics* (1982) 3,97-101.

[4] C. Berge, A property of the k-optimal partitions, *Progress in Graph Theory* (J.A. Bondy and U. S. R. Murty, eds.), 105-108, Academic Press, Toronto, 1984.

[5] C. Berge, Path partitions in directed graphs and Posets, *Graphs and Orders, (Banff, Alta., 1984), 447-464, NATO Adv. Sci. Inst. Ser. C Math. Phys. Sci.*, 147, Reidel, Dordrecht, 1985.

[6] J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, MacMillan, London, 1972.

[7] K. Cameron, Polyhedral and Algorithmic Ramifications of Antichains, Ph.D. thesis, University of Waterloo, 1982.

[8] K. Cameron, On k-optimum dipath partitions and partial k-colourings of acyclic digraphs, *Europ. J. Combinatorics* 7 (1986), 115-118.

[9] K. Cameron, On Gallai-type min-max inequalities, *Combinatorics, Graph-Theory, Algorithms and Applications*(Beijing, 1993), 7-16, World Scientific Publishing, River Edge, NJ, 1994.

[10] P. Dilworth, A decomposition theorem for partially ordered sets, *Ann. of Math.* (1), vol.51(1950), 161-166.

[11] A. Frank, On chain and antichain families of a partially ordered set, *J. Comb. Theory*, Ser B, 29 (1980), 176-184.

[12] T. Gallai, Problem 15, in: *Theory of Graphs and its Applications,(eds. M. Fiedler)*, Proc. Symposium Smolenice 1963, p. 161.

[13] T. Gallai, On directed paths and circuits, in: *Theory of Graphs,(P. Erdos and G. Katona, eds.), Academic Press, New York,* 1968, 115-118.

[14] T. Gallai and A.N. Milgram, Verallgemeinerung eines graphentheoretischen satzes von Redei, *Acta Sc. Math.* 21 (1960), 181-186.

[15] C. Greene, D.J. Kleitman, The structure of Sperner k-families, *J. Combinatorial Theory, Ser.A* 20 (1976),41-68.

[16] N. Linial, Covering digraphs by paths, *Discrete Math.* 23 (1978), 257-272.

[17] N. Linial, Extending the Greene-Kleitman theorem to directed graphs, *J. Combinatorial Theory, Ser.A* 30 (1981), 331-334.

[18] M. Saks, A short proof of the k-saturated partitions, *Adv. In Math.* 33 (1979), 207-211.

[19] M. Saks, Duality properties of finite set systems, PH.D. Dissertation, Massachusetts Institute of Technology, Cambridge, MA, 1980.

[20] M. Saks, Some sequences associated with combinatorial structures, *Discrete Math.* 59 (1986), 135-166.

Acknowledgment

I thank Adrian Bondy for listening to a previous version of this algorithm, and insisting that there must be a simpler proof. Hopefully, this is simpler and more intuitive.

Irith Ben-Arroyo Hartman
Caesarea Rothschild Institute
for Interdisciplinary Applications of Computer Science
University of Haifa, Haifa 31905, Israel

e-mail: `irith@cs.haifa.ac.il`