Egerváry Research Group
on Combinatorial Optimization

Technical reportS

TR-2015-13. Published by the Egerváry Research Group, Pázmány P. sétány 1/C, H–1117, Budapest, Hungary. Web site: www.cs.elte.hu/egres. ISSN 1587–4451.

Berge's path partition conjecture: an algorithm for almost all known cases

Dávid Herskovics

August 2015

Berge's path partition conjecture: an algorithm for almost all known cases

Dávid Herskovics⋆

Abstract

Let D be a digraph. A *path partition* of D is called k-optimal if the sum of the k-norms of its paths is minimal. The *k-norm* of a path P is $\min(|V(P)|, k)$. Berge's path partition conjecture claims that for every k-optimal path partition $\mathcal{P}$ there are k disjoint stable sets orthogonal to $\mathcal{P}$. For general digraphs the conjecture has been proven for $k = 1, 2$ and $k \geq \lambda - 3$, where λ is the length of a longest path in the digraph. In this paper we give an algorithm which given a path partition $\mathcal{P}$, if it stops after a finite number of steps, it either finds k disjoint stable sets orthogonal to $\mathcal{P}$ or finds a better path partition. We prove that the algorithm stops after a finite number of steps for $k = 1$ and $k \geq \lambda - 3$.

Keywords: digraph, path partition, Berge's path partition conjecture

1 Introduction

Let $D = (V, A)$ be a digraph. A path partition of D is a set of disjoint (directed) paths $P_1, P_2, \ldots, P_m$ for which $V(P_1) \cup V(P_2) \cup \cdots \cup V(P_m) = V$. Throughout the paper by path we always mean directed path and a single vertex is also considered to be a path. Let $\mathcal{P}$ be a path partition and $\mathcal{S}$ be a set of k disjoint stable sets. We say that $\mathcal{P}$ and $\mathcal{S}$ are *orthogonal* if each path P_i intersects as many of the k stable sets as possible, i.e. $\min(|V(P_i)|, k)$. The Greene-Kleitman theorem [1] has shown that if the digraph is acyclic and transitive (i.e. represents a partially ordered set), then for each positive integer k and for each path partition $\mathcal{P}$ minimizing $\sum \min(|V(P_i)|, k)$ there are k disjoint stable sets orthogonal to $\mathcal{P}$. In 1982, Berge made his conjecture claiming the same for all digraphs ([2]).

The conjecture is known to be true for acyclic digraphs ([3]) and for $k \geq \lambda - \sqrt{\lambda}$ (where λ is the cardinality of a longest path in D) for strongly connected digraphs

⋆Department of Operations Research, Eötvös Loránd University, Budapest, Pázmány Péter stny. 1/C, Hungary. Email: hdavid@caesar.elte.hu. Research is supported by the National Development Agency of Hungary (grant no. K 109240), based on a source from the Research and Technology Innovation Fund.

([4]). However, for general digraphs only six cases are known: $k \leq 2$ and $k \geq \lambda - 3$. The cases $k = \lambda - 3, \lambda - 2$ have been proven recently in [6].

In [5] Eli Berger and Irith Hartman gave an algorithmic proof for the cases $k = 1, \lambda - 1, \lambda$.

In [6] we gave an algorithmic proof for the cases $k \geq \lambda - 3$.

Both algorithms are greedy algorithms and approach the problem with similar mind-set but from the opposite direction in some sense: given a path partition, the first algorithm either gives k disjoint stable sets orthogonal to it or "improves" the path partition – the problem is that this improvement might not result in a path partition. The second algorithm either finds a better path partition or finds k disjoint sets orthogonal to the path partition but not necessarily stable sets.

In this paper we propose a new algorithm which is a combination of these two. Given a path partition, if this algorithm stops after a finite number of steps, then it will either give k disjoint stable sets orthogonal to it or a better path partition. We will prove that the algorithm stops after a finite number of steps in the cases $k = 1$ and $k \geq \lambda - 3$.

We use the following definitions and notations:

Definition 1.1. The *k-norm* of a path partition $\mathcal{P} = \{P_1, \ldots, P_m\}$ is defined by:

$$|\mathcal{P}|_k = \sum \min(|V(P_i)|, k).$$

A path partition is *k-optimal* if its k-norm is minimal.

Definition 1.2. For a digraph D, $\pi_k(D)$ denotes $|\mathcal{P}|_k$ where $\mathcal{P}$ is a k-optimal path partition of D.

Definition 1.3. Let $\mathcal{P}$ be a path partition and $S^1, \ldots, S^k$ disjoint stable sets. We say that $S^1, \ldots, S^k$ are *orthogonal* to $\mathcal{P}$ if each path P of $\mathcal{P}$ intersects exactly $\min(|V(P)|, k)$ sets of $S^1, \ldots, S^k$.

Definition 1.4. Let $\mathcal{P}$ be a path partition. We denote by $\mathcal{P}^{\leq k}$ the set of paths in $\mathcal{P}$ with cardinality at most k. Similarly we denote by $\mathcal{P}^{\geq k}$ the set of paths in $\mathcal{P}$ with cardinality at least k.

Conjecture 1.5 (Berge's path partition conjecture)**.** *Let D be a digraph and k a positive integer. Then for every k-optimal path partition $\mathcal{P}$ there are k disjoint stable sets orthogonal to $\mathcal{P}$.*

Finding a k-optimal path partition in general digraphs is NP-complete as $\pi_k(D) = k$ for any $k < n$ if and only if there is a Hamiltonian path in D. However, if we also allow cycles in our partition and thus consider path-cycle partitions, then finding a k-optimal path-cycle partition and k disjoint stable sets orthogonal to its paths can be done. In the next section we show how.

2 A min-max theorem

The following definitions (except the standard stable-cut pair / standard stable-cut k-family) and the min-max theorem are from [6].

Definition 2.1. The k-norm of a path-cycle partition $\mathcal{P}^c = \{P_1, \ldots, P_r, C_1, \ldots, C_t\}$ where P_i are paths and C_i are cycles is the following:

$$|\mathcal{P}^c|_k = \sum_{i=1}^{r} \min(|V(P_i)|, k).$$

A path-cycle partition is k-optimal if its k-norm is minimal.

Definition 2.2. Let D be a digraph. Then $\pi_k^c(D) = |\mathcal{P}^c|_k$ where $\mathcal{P}^c$ is a k-optimal path-cycle partition of D.

Let S be a stable set in the digraph $D = (V, A)$. We call a vertex set $S_{cut} \subseteq V$ an S-cut set (or just cut-set if S is unambiguous) if $S \cap S_{cut} = \emptyset$ and all directed paths from S to S contain a vertex from S_{cut} as an internal point – we say S_{cut} *cuts* every S to S dipath. The pair (S, S_{cut}) is called *stable-cut pair.* For a pair of stable set S and one of its cut-sets S_{cut}, we use the notation $\langle S, S_{cut}\rangle = |S| - |S_{cut}|$.

Definition 2.3. An (S, S_{cut}) stable-cut pair is *optimal* if $\langle S, S_{cut}\rangle$ is the largest possible.

Definition 2.4. Two stable-cut pairs (S^1, S^1_{cut}) and (S^2, S^2_{cut}) are *disjoint* if S^1 and S^2 are disjoint.

Definition 2.5. An $\mathcal{S} = \{(S^1, S^1_{cut}), \ldots, (S^k, S^k_{cut})\}$ is a *stable-cut k-family* if each (S^i, S^i_{cut}) is a stable-cut pair, and they are pairwise disjoint. $\mathcal{S}$ is *optimal* if $\sum\langle S^i, S^i_{cut}\rangle$ is the largest possible.

Definition 2.6. For an optimal stable-cut family $\mathcal{S} = \{(S^1, S^1_{cut}), \ldots, (S^k, S^k_{cut})\}$, we use the notation $\alpha_k^* = \sum\langle S^i, S^i_{cut}\rangle$.

Lemma 2.7. *For every digraph $D = (V, A)$, $\pi_1^c = \alpha_1^*$.*

Proof. For the digraph $D = (V, A)$ we make a bipartite graph $G = (S \cup T, E)$ in the following way:

$$S = \{v' \;:\; v \in V\}, \qquad T = \{v'' \;:\; v \in V\}, \qquad E = \{(u', v'') \;:\; (u, v) \in A\}$$

A matching M in G corresponds to a path-cycle partition of D with $|V| - |M|$ paths and vice versa. From Kőnig's theorem we know that in a bipartite graph the size of a maximum matching equals the size of a minimum vertex cover. Let Z be a min vertex cover in G and $Z_1 = \{v \in V \;:\; v' \in Z\}$, $Z_2 = \{v \in V \;:\; v'' \in Z\}$. Now all arcs in D have either their tails in Z_1 or their heads in Z_2, so $S = V \setminus (Z_1 \cup Z_2)$ is a stable set. Let P be a path from S to S with k arcs. Let a be the first arc along P whose tail is in Z_1. This arc exists because P ends in a vertex of S. Moreover this arc is not the first arc of P. Hence, by definition the arc preceding a in P has its head in Z_2, and so the tail of a is in $Z_1 \cap Z_2$. So $S_{cut} = Z_1 \cap Z_2$ is an S-cut. For the pair of (S, S_{cut}) we have $\langle S, S_{cut}\rangle = |V| - |Z_1 \cup Z_2| - |Z_1 \cap Z_2| = |V| - |Z_1| - |Z_2| = |V| - |Z| = |V| - |M|$, where M is a maximum matching. Thus $\alpha_1^* \geq \langle S, S_{cut}\rangle = \pi_1^c$. The other direction is easy to see. □

In a digraph there might be several optimal stable-cut pairs and several 1-optimal partitions as well. For our algorithm to work we choose one specific (but not necessarily different) optimal stable-cut pair for each 1-optimal path partition. We do this by using the classic proof of the Kőnig's theorem with alternating paths:

Let $\mathcal{P}$ be a 1-optimal path partition of D. We construct the bipartite graph $G = (S \cup T, E)$ described in the above proof and the maximum matching $M_{\mathcal{P}}$ corresponding to $\mathcal{P}$. Let R_S be the set of unmatched vertices in S and let the set of Z be the set of vertices which can be reached from R_S via an alternating path (paths that alternate between edges that are in the matching and edges that are not in the matching). Then $(T \setminus S) \cup (Z \cap T)$ is a minimum vertex cover. We use this Z in the above proof to obtain the optimal stable-cut pair for $\mathcal{P}$.

Definition 2.8. We call the optimal stable-cut pair obtained by the above method the *standard* stable-cut pair (for the given path partition).

Theorem 2.9. *For every digraph $D = (V, A)$, $\pi_k^c = \alpha_k^*$*

We only show the outline of the proof, for the detailed proof please consult [6]:

Outline of the proof. To a digraph D and a positive integer k, we associate the digraph D^k defined by

$$V_k = \{(v, i) \ : \ v \in V, \ \ 1 \leq i \leq k\}$$

$$A_k = \{\big((u, i), (v, i)\big) \ : \ (u, v) \in A\} \cup \{\big((u, i), (u, j)\big) \ : \ i < j\}.$$

We prove that $\alpha_k^*(D) = \alpha_1^*(D^k) = \pi_1^c(D^k) = \pi_k^c(D)$. □

For $k = 1$ we defined the standard (optimal) stable-cut pair of D for each optimal path partition. We do the same for all k and define the standard (optimal) stable-cut k-family for each k-optimal path partition. For this the following remarks are needed:

The digraph D^k has k levels, each level is a copy of D. By looking at how it intersects each level a stable-cut pair in D^k corresponds to a stable-cut k-family in D. The proof shows that an optimal stable-cut pair in D^k corresponds to an optimal stable-cut k-family in D.

From a k-optimal path partition $\mathcal{P}$ in D we can construct a 1-optimal path partition $\mathcal{P}^*$ in D^k as it is shown on the figure below (each path P with at least k vertices corresponds to k horizontal copies and each path P' with less than k vertices corresponds to $|V(P')|$ vertical paths):

Given a k-optimal path partition in D, we construct D^k and the corresponding 1-optimal path partition in D^k. We take the standard stable-cut pair for this path partition in D^k and this gives us an (optimal) stable-cut k-family in D.

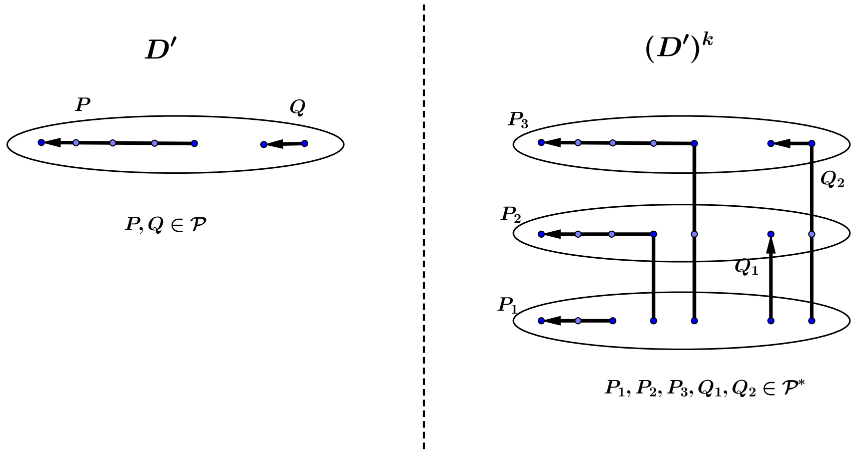


Figure 1: path partition $\mathcal{P}$ of D is on the left, the corresponding path partition $\mathcal{P}^*$ of D^k is on the rigth.

Definition 2.10. We call the optimal stable-cut k-family obtained by the above method the *standard* stable-cut k-family in D for the given path partition.

Using complementary slackness a k-optimal path-cycle partition $\mathcal{P}^c$ and an optimal stable-cut k-family $\mathcal{S} = \{(S^1, S^1_{cut}), \dots, (S^k, S^k_{cut})\}$ satisfy the following properties:

1. Each path P in $\mathcal{P}^c$ intersects exactly $\min(|V(P)|, k)$ of the $S^1, \dots, S^k$.
2. For each path P in $\mathcal{P}^c$ and i we have $|V(P) \cap S^i_{cut}| = \max(|V(P) \cap S^i| - 1, 0)$.
3. For each cycle C in $\mathcal{P}^c$ and i we have $|V(C) \cap S^i| = |V(C) \cap S^i_{cut}|$.

Definition 2.11. When the above properties are satisfied for a path-cycle partition $\mathcal{P}^c$ and a stable-cut k-family $\mathcal{S} = \{(S^1, S^1_{cut}), \dots, (S^k, S^k_{cut})\}$ we call them *orthogonal* to each other.

Remark If the digraph is acyclic, then a k-optimal path-cycle partition $\mathcal{P}^c$ is naturally a path partitions. Any optimal stable-cut k-family $\mathcal{S} = \{(S^1, S^1_{cut}), \dots, (S^k, S^k_{cut})\}$ is orthogonal to $\mathcal{P}^c$ (in the above sense) so the k disjoint stable sets $S^1, \dots, S^k$ are orthogonal to $\mathcal{P}^c$ (in the classic sense). This implies that the path partition conjecture holds for acyclic digraphs. Also, when the digraph is acyclic and transitive (i.e. it is a poset) then all cut-sets must be empty. In that case we get the Greene-Kleitman theorem.

The following lemma states that the standard stable-cut k family of digraph D doesn't have vertices from a cut-set along the "short" paths. For the proof please consult [6]:

Lemma 2.12. *Given a digraph $D = (V, A)$ and a k-optimal path partition $\mathcal{P}$, let $\mathcal{S} = \{(S^1, S^1_{cut}), \ldots, (S^k, S^k_{cut})\}$ be the standard stable-cut k-family. Then no path $P \in \mathcal{P}$ with at most $k+1$ vertices can contain a vertex from S^i_{cut} for any i.*

3 The algorithm

We use the same idea Eli Berger and Irith Hartman used in [5] as the base of our algorithm: we start with a path partition $\mathcal{P}$. We want to either find k disjoint stable sets orthogonal to $\mathcal{P}$ or find better path partition $\mathcal{Q}$, i.e. $|\mathcal{Q}|_k < |\mathcal{P}|_k$. The difference is that we use stable-cut k-families to make an additional step (deleting arcs if needed) and use it to prove the correctness of our algorithm for the cases $k = 1$ and $k \geq \lambda - 3$.

We maintain an acycic subdigraph D' containing $\mathcal{P}$. At each step we look at the standard stable-cut k-family and try to find k disjoint stable sets orthogonal to $\mathcal{P}$ with its help. If we succeed then we are done. If not then we update D'. We check if $\mathcal{P}$ is still k-optimal in this updated D'. If not then we can find a better path partition since D' is acyclic. If $\mathcal{P}$ is still k-optimal, then we repeat the steps in the updated D'.

1. Let D' be the digraph containing the arcs of the path partition $\mathcal{P}$.
2. Find the standard stable-cut k-family $\mathcal{S} = \{(S^1, S^1_{cut}), \ldots, (S^k, S^k_{cut})\}$ in D'. Let $S^i_{first} = \{\bigcup_{P \in \mathcal{P}}$ the first vertex along the path P that is in $S^i\}$.
   (a) If $S^1_{first}, \ldots, S^k_{first}$ are stable sets even in D, then we found k disjoint stable sets orthogonal to $\mathcal{P}$ and we are done.
   (b) If not then let (uv) be an arc in D such that $u, v \in S^j_{first}$ for some $1 \leq j \leq k$. Add (uv) to D' and delete other arcs if needed to maintain the acylicity of D. As to which arcs are to be deleted is explained below.
3. Check wether $\mathcal{P}$ is k-optimal in D'.
   (a) If not then we find a better path partition $\mathcal{Q}$ and we are done.
   (b) If yes then repeat the steps from 2.

Now what is left is to determine which arcs are to be deleted if needed: Let $u, v \in S^j_{first}$ in D' such that (uv) is an arc in D. If the addition of the arc (uv) creates a cycle in D', that means that there was a path Q from v to u in D'. Then Q must contain a vertex from S^j_{cut}. Let $x \in S^j_{cut}$ be the last one of such a vertex along Q. The vertex x is on a path P of the path partition. Since $u \in S^j_{first}$ so there is an arc (yz) in path Q such that $y \in V(P)$, $z \notin V(P)$ and y is after x along the path P. By deleting this (yz) arc we cut the Q path between v and uv. We repeat this till there are no more paths from v to u and thus the addition of the arc (uv) won't create a cycle.

4 Main result

If the algorithm stops after a finite number of steps, then we either find k disjoint stable sets orthogonal to $\mathcal{P}$ or a better path partition.

Theorem 4.1. *If the path partition $\mathcal{P}$ only contains paths with cardinality at least k, then the algorithm stops after a finite number of steps.*

Proof. We prove that our algorithm delete only a finite number of arcs. If this holds, then after deleting the last arc each step will increase the size of D' thus the algorithm can not go on infinitely.

Consider the moment we are about to add an arc (uv) to our current D' and have to delete the arc (yz) because of this. Then the vertices u, v are in the same S^i_{first} for some i. We first prove that the deletion of the arc (yz) from D' do not change the first i stable-cut pairs of the standard stable-cut k family, i.e. S^j, S^j_{cut} for $j \leq i$ remains the same after the deletion:

We construct D^k from D and the path partition $\mathcal{P}^*$ of D^k form the path partition $\mathcal{P}$ of D as it was descibed earlier and shown in Figure 1. We find the standard stable-cut pair (S, S_{cut}) as it was described earlier. This gives us the sets Z_1 and Z_2 in D^k where each arc of D^k has either its tail in Z_1 or its head in Z_2. From the way we chose which arc is to be deleted, it's not difficult to see that the vertex (y, j) of D^k is in Z_1 for all $j \leq i$. This means that deleting the arcs $\big((y, j)(z, j)\big)$ does not change the standard stable-cut pair (S, S_{cut}). Thus we only need to see that the deletion of the arcs $\big((y, j), (z, j)\big)$, $i < j$ doesn't change the standard stable-cut pair S, S_{cut} on the first (lowest) levels of D^k.

Since all paths of $\mathcal{P}$ have cardianlity at least k, the corresponding path partition $\mathcal{P}^*$ of D^k has only horizontal paths. Beacuse of this the alternating paths can only go upwards but not downwards in D^k. Beacuse of this and because of how we defined the standard stable-cut pair (S, S_{cut}) using alternating paths, deleting the arc (y, j) can only change (S, S_{cut}) on the j-th level and above in D^k.

Now we consider what happens when we add the arc (uv) to $D' - (yz)$. Let the paths in $\mathcal{P}$ be $P_1, P_2, \ldots, P_p$. For simplicity suppose that $v \in V(P_1)$. Let x^l_m denote the distance along the path P_m between its tail and the vertex w from $S^l_{first} \cap V(P_m)$. If the addition of the arc (uv) does not create a better path partition, then the new standard stable-cut k family remains orthogonal to $\mathcal{P}$. In this case the addition of the arc (uv) decreases x^i_1 and does not increases any of the x^l_m. So the addition of the arc (uv) and deletion of the arc (yz) increases $(x^1_1, \ldots, x^1_p, x^2_1 \ldots, x^2_p, \ldots, x^k_1, \ldots, x^k_p)$ according to the lexicographical order. Thus after a finite number of steps we either has to stop or find a better path partition.

□

Specially, if $k = 1$, then every path has cardinality at least k, thus

Theorem 4.2. *For $k = 1$ the algortihm stops after a finite number of steps.*

Theorem 4.3. *If each path in the path partition $\mathcal{P}$ has cardinality either not bigger than $k + 1$ or not smaller than $\lambda - 1$, then the algorithm stops after a finite number of steps.*

Proof. Let the arc (uv) be an arc we added to D' during our algorithm. We call this arc (uv) *problematic* if the addition of (uv) results in the deletion of some other arcs (i.e. $D' + (uv)$ is not acyclic). We prove that after a finite number of steps there will be no more problematic arcs added. From then on we don't need to delete arcs so each step of the algorithm increases the number of arcs in D' thus the algorithm stops after a finite number of steps.

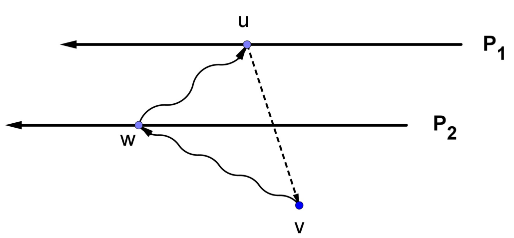
Let $V_1 \subset V$ be the set of vertices which are on a "short" path in $\mathcal{P}$, i.e. the vertex v is in V_1 if and only if the path $P \in \mathcal{P}$, $v \in V(P)$ has at most $k + 1$ vertices. The union of the vertices on "long" paths (i.e. paths with at least $k + 2$ vertices) wil form the set V_2. According to lemma 2.12 the set $\bigcup_i S^i_{cut} \subset V_2$. There can be 4 types of arcs (uv):

1. Both vertices u and v are in V_1,
2. $u \in V_1$, $v \in V_2$,
3. $u \in V_2$, $v \in V_1$,
4. both vertices u and v are in V_2.

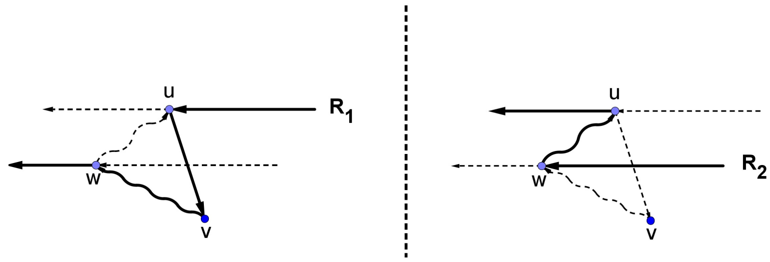
Let (xy) be an arc we delete during our algorihm. Let the path $P_x \in \mathcal{P}$ contain the vertex x. According to how we chose the arc (xy) to be deleted, the path P_x contains a vertex from S^i_{cut} for some i. Using Lemma 2.12 this means that P_x is a "long" path, i.e. it has at least $k + 2$ vertices. That means that the deleted arc (xy) can not be of type 1 or 2. This also means that after a finite number of steps we won't add any more arcs of type 1 or 2 thus the problematic arcs can only be of type 3 or 4 from there on.

Now we prove that there can not be a problematic arc of type 3 or 4 during our algorithm.

Suppose that we add the arc (uv) to D' during our algorithm and $u \in V_2$. Then $u, v \in S^i_{first}$ for some i. If (uv) is problematic, then there must be a $v \to u$ path Q in D' with a $w \in S^i_{cut}$ as an internal vertex. Both u and w are in V_2 (the latter because $S^i_{cut} \subset V_2$) so they are respectively on paths P_1, P_2, where $|V(P_1)|, |V(P_2)| \geq \lambda - 1$ and $P_1 \neq P_2$ (because $u \in S^i_{first}$ so if $P_1 = P_2$ then w would be after u along P_1, but that would mean that $P_1 \cup Q$ contains a cycle, which is impossible as D' is acyclic). The vertex v can possibly be on P_2, but not on P_1 as we took only one element of S^i from all paths in $\mathcal{P}$. On the figure below we choose not to place v on P_2 but that is of no importance in the proof and v could be on P_2 somewhere before w (but not after w as it would mean a cycle in D').



Now if we take a look at the figure, we can see that from the arc (u, v) and from the parts of P_1, P_2 and Q we can glue together two new paths: R_1 and R_2.



Thus $2\lambda \geq |V(R_1)| + |V(R_2)| \geq |V(P_1) + |V(P_2)| + 3$. But P_1, P_2 are both "long" paths, so $|V(P_1)| + |V(P_2)| + 3 \geq 2\lambda + 1$, which is a contradiction. □

Specially, if $k \geq \lambda - 3$, then every path has this property and thus

Theorem 4.4. *For $k \geq \lambda - 3$ the algorithm stops after finite number of steps.*

Acknowledgement

Supported by Hungarian Scientific Research Fund – OTKA, K109240.

References

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

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

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

[4] A. Sebő, Path Partitions, Cycle Covers and Integer Decomposition, *Lecture Notes in Computer Science Series* 5420 (2009), 183-199.

[5] E. Berger, I.B-A. Hartman, A unified approach to known and unknown cases of Berge's conjecture, *Journal of Graph Theory* Vol 71 Issue 3 (Nov 2012), 317-330.

[6] D. Herskovics, Proof of Berge' path partition conjecture for $k \geq \lambda - 3$, *Tech. Report TR-2013-08, Egerváry Research Group, Budapest (2013), http://www.cs.elte.hu/egres*