Algorithms for long paths in graphs

Zhao ZHANG*

College of Mathematics and System Sciences

Xinjiang University

Urumqi, Xinjiang, 830046, CHINA

e-mail: zhzhao@xju.edu.cn

Hao LI†

Laboratoire de Recherche en Informatique

UMR 8623, C.N.R.S.-Université de Paris-sud

91405-Orsay cedex, FRANCE

e-mail: li@lri.fr

Abstract

We obtain a polynomial algorithm in $O(nm)$ time to find a long path in any graph with n vertices and m edges. The length of the path is bounded by a parameter defined on neighborhood condition of any three independent vertices of the path. Example is given to show that this bound is better than several classic results.

1 Introduction and notation

It is a classic problem to find a long path or cycle in a graph. Since finding a hamiltonian path/cycle in graphs is NP-hard, we are interested in finding a path with large length.

All graphs considered in this paper are undirected and simple. We follow the notation and terminology in [4]. For a graph $G = (V(G), E(G))$ and a subgraph H of G, the neighborhood of a vertex u in H is $N_H(u) = \{v \in V(H) \ : \ uv \in E(G)\}$. The degree of u in H is $d_H(u) = |N_H(u)|$. In the case $H = G$, we use $N(u)$ and $d(u)$ instead of $N_G(u)$ and $d_G(u)$. For simplicity, the graph itself is used to denote its set of vertices.

For a path $P = u_1u_2...u_p$ and two indices $i < j$, denote by $P[u_i, u_j] = u_iu_{i+1}...u_j$, and $\overline{P}[u_j, u_i] = u_ju_{j-1}...u_i$. Define $P(u_i, u_j] = P[u_{i+1}, u_j]$, $P[u_i, u_j) = P[u_i, u_{j-1}]$ and $P(u_i, u_j) = P[u_{i+1}, u_{j-1}]$. For any i, $u_i^+ = u_{i+1}$ and $u_i^- = u_{i-1}$. For $A \subseteq P$, $A^+ = \{v^+|v \in A\}, A^- = \{v^-|v \in A\}$.

*The work is partially supported by NSFC (10271101)

†Corresponding author. The work is partially supported by NNSF of China (60373012)

We use $|P|$ to denote the number of vertices in a path P. Denote by $\sigma_2(G) = \min\{d(u)+d(v):\ uv \notin E(G)\}$ and $\overline{\sigma}_3(G) = \min\{\sum_{i=1}^{3} d(u_i) - |\bigcap_{i=1}^{3} N(u_i)|: \{u_1, u_2, u_3\}$ is an independent set of $G\}$.

A Hamiltonian cycle (path, resp.) is a cycle (path, resp.) containing all vertices of the graph. A graph G is Hamiltonian if it has a Hamiltonian cycle. For an integer k, a graph is called k-connected if any two vertices can not be separated by deleting less than k vertices in the graph.

We begin with the following basic results in Hamiltonian graph theory, which are due to Dirac, Ore and Flandrin, Jung and Li, respectively.

Theorem 1. [5] *Let G be a graph on $n \geq 3$ vertices. If the minimum degree δ is at least $n/2$, then G is Hamiltonian.*

Theorem 2. [10] *Let G be a graph on $n \geq 3$ vertices. If $\sigma_2(G) \geq n$, then G is Hamiltonian.*

Theorem 3. [6] *If G is a 2-connected graph of order n such that $\overline{\sigma}_3(G) \geq n$, then G is hamiltonian.*

These results are generalized to circumferences of the graphs. The circumference $c(G)$ is the length of a longest cycle in G.

Theorem 4. [5] *If G is a 2-connected graph on $n \geq 3$ vertices, then $c(G) \geq$ min $\{n,\ 2\delta\}$.*

Theorem 5. [1] *Let G be a 2-connected graph on $n \geq 3$ vertices. Then $c(G) \geq min\{n, \sigma_2(G)\}$.*

Theorem 6. [11] *Let G be a 3-connected graph with n vertices. Then $c(G) \geq \min\{n, \overline{\sigma}_3(G)\}$.*

As a consequence of Theorem 6, we have the following

Corollary 1. *Let G be a 2-connected graph with n vertices. Then there exists a path of at least* $\min\{n, \overline{\sigma}_3(G)+1\}$ *vertices.*

Proof. Let D to be a graph obtained from G by adding a new vertex w which is adjacent to every vertex of G. Then D is 3-connected. By Theorem 6, $c(D) \geq \min\{n, \overline{\sigma}_3(D)\}$. Since $\overline{\sigma}_3(D) \geq \overline{\sigma}_3(G)+2$, we see that G has a path of at least $c(D)-1 \geq \min\{n, \overline{\sigma}_3(G)+1\}$ vertices. □

Since $\overline{\sigma}_3(G) \geq \sigma_2(G) \geq 2\delta$, we have the following two results:

Corollary 2. *Let G be a 2-connected graph with n vertices. Then there exists a path of at least* $\min\{n, \sigma_2(G)+1\}$ *vertices.*

Corollary 3. *Let G be a 2-connected graph with n vertices. Then there exists a path of at least* $\min\{n, 2\delta+1\}$ *vertices.*

In this paper, we will generalize the above corollaries by giving a new lower bound for the length of a longest path, using neighborhood condition of three independent vertices, one of which is an end of the path!

Since the problem of deciding whether a graph has a Hamiltonian path is NP-complete, it is interesting to find a long path in a network which can be realized by a polynomial algorithm. Such an algorithm with time complexity $O(nm)$ is given in this paper, by which we can find a long path with a length related with an end vertex of the path.

Some notation will be used in this paper. For a subgraph H and three vertices x, y, z, denote by

$$\Gamma_H(x, y, z) = d_H(x) + d_H(y) + d_H(z) - |N_H(x) \cap N_H(y) \cap N_H(z)|.$$

For $x \in H$, denote by

$$\Gamma_3(x, H) = \min\{\Gamma_H(x, y, z)| \ : \ y, z \in H \text{ and } x, y, z \text{ are independent}\}.$$

Clearly $\Gamma_3(x, H) \geq \overline{\sigma}_3(G)$.

The main result is the following:

Theorem 7. *Let G be a 2-connected graph of order $n \geq 3$. Then there exists a vertex x and a path P such that x is one end vertex of P and P contains at least* $\min\{n, \Gamma_3(x, P) + 1\}$ *vertices. Furthermore, P can be found in $O(nm)$ time.*

Theorem 7 is best possible in the following sense. Suppose d, f, r are three integers with $d \geq 8$, $3 \leq f \leq d-5$, and $r \geq 2$. Let G be the graph obtained from d disjoint graphs G_i $(1 \leq i \leq d)$ with $G_i \cong K_r$ $(1 \leq i \leq f)$ and $G_j \cong K_1$ $(f+1 \leq i \leq d)$, by adding edges from G_{d-1} and G_d to all the other vertices. It is easy to see that there is a path P containing all the vertices in G_i $(i = 1, 2, 3, d-1, d)$ with two end vertices $x_1 \in G_1$ and $x_2 \in G_2$ respectively. Clearly, P is a longest path with $3r + 2 = d(x_1) + d(x_2) + d(x_3) - |N(x_1) \cap N(x_2) \cap N(x_3)| + 1$ vertices, where x_3 is a vertex in G_3. So the bound in Theorem 7 is sharp. Furthermore, the same example shows that our result is better than the corollaries since $\overline{\sigma}_3(G) = 4 < |P| + 1$.

2 Proof of the main theorem

The idea of our proof of Theorem 7 is as follows. Let $P_1 = u_0u_1...u_p$ be a maximal path (in the sense of inclusion of vertices), and $P_2 = v_0v_1...v_q$ with

(a) $P_1 \cap P_2 = \{v_0\} = \{u_c\}$,
(b) subject to (a), c is as large as possible, and
(c) subject to (a) and (b), q is as large as possible.

Then a cycle P_V called vine of P_1 (which will be defined later) is found. Based on P_1, P_2 and P_V, a path P is constructed such that

$$v_q, u_p, u_0 \text{ are three independent vertices on } P \text{ with} \tag{1}$$
$$v_q \text{ or } u_0 \text{ being one end of } P\text{, and} \tag{2}$$
$$N(v_q) \cup N(u_p) \cup N(u_0) \subseteq P, \tag{3}$$
$$\Gamma_P(v_q, u_p, u_0) \leq |P| - 1. \tag{4}$$

With these properties, it is easy to see that P is a path with the desired length.

From algorithmic point of view, to find a maximal path P_1 needs a lot of work. However, to ensure that the path P we find has the desired length in Theorem 7, we do not need all properties of a maximal path. In fact, properties (1) to (4) are essential for our purpose, and to ensure that P satisfies properties (1) to (4), only nine operations to extend P_1 are sufficient, which are introduced in the following.

Circumstance 1: There is a vertex $v \in V(G) \setminus V(P_1)$ which is adjacent to one end of P_1.
Operation 1: Extend P_1 by adding v.

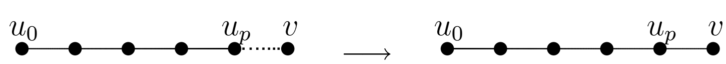


Figure 1

Circumstance 2: There is a vertex $v \in V(G) \setminus V(P_1)$ such that $u_i \in N_{P_1}(v)$ and u_{i+1} is connected to v by a path internally disjoint from P_1.
Operation 2: Reset $P_1 = u_0 u_1 ... u_i v ... u_{i+1} ... u_p$.

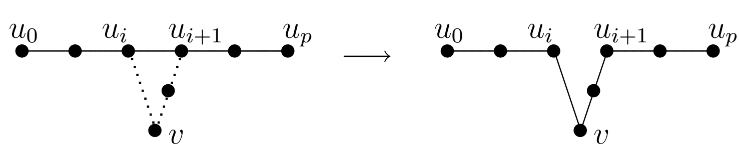


Figure 2

Circumstance 3: u_0 is adjacent to u_p, and $V(G) \setminus V(P_1) \neq \emptyset$.
Operation 3: Let v be a vertex in $V(G) \setminus V(P_1)$ which is adjacent to some vertex u_i on P_1. Reset $P_1 = v u_i u_{i-1} ... u_0 u_p u_{p-1} ... u_{i+1}$.

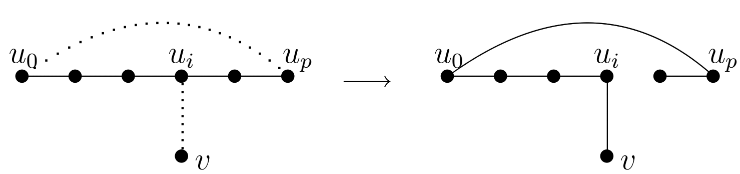


Figure 3

Circumstance 4: $u_i \in N_{P_1}(u_0) \cap N_{P_1}(u_p)^+ \neq \emptyset$ and $V(G) \setminus V(P_1) \neq \emptyset$.

Operation 4: Reset $P_1 = u_{i-1}u_{i-2}...u_0u_iu_{i+1}...u_p$,
and then extend it further by Operation 3.

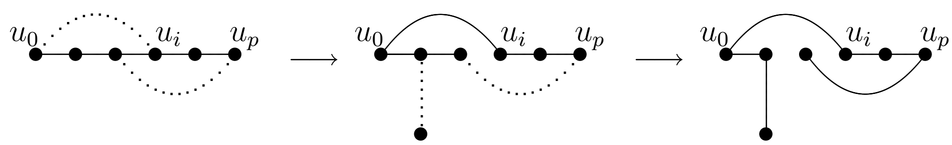


Figure 4

Circumstance 5: There is a vertex $u_i \in N(u_p)$ with u_{i+1} having some neighbor v outside of P_1.

Operation 5: Reset $P_1 = u_0u_1...u_iu_pu_{p-1}...u_{i+1}v$.

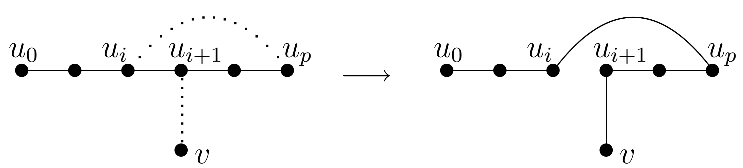


Figure 5

Circumstance 6: There is a vertex $u_i \in N(u_0)$ with u_{i-1} having some neighbor v outside of P_1.

Operation 6: Reset $P_1 = vu_{i-1}u_{i-2}...u_0u_iu_{i+1}...u_p$.

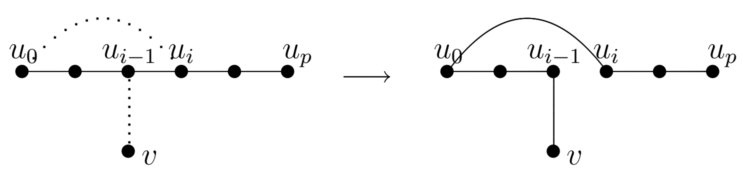


Figure 6

Circumstance 7: There is a vertex $u_i \in N(u_p)$ with u_{i-1} having some neighbor v outside of P_1, and there is an index $j > i$ such that $u_j \in N(u_0)$.

Operation 7: Reset $P_1 = vu_{i-1}u_{i-2}...u_0u_ju_{j-1}...u_iu_pu_{p-1}...u_{j+1}$.

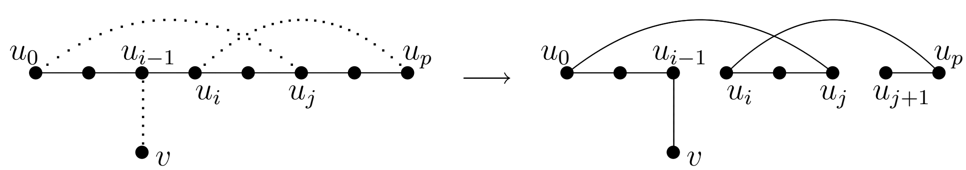


Figure 7

Circumstance 8: $u_i \in N_{P_1[u_1,u_c)}(u_p) \cap N_{P_1[u_1,u_c)}(v_q)^+ \neq \emptyset$.

Operation 8: Reset $P_1 = u_0u_1...u_{i-1}v_qv_{q-1}...v_1u_cu_{c+1}...u_pu_iu_{i+1}...u_{c-1}$.

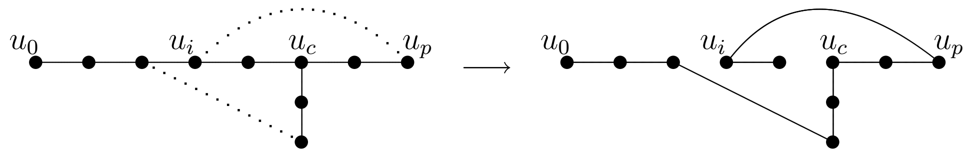


Figure 8

Circumstance 9: $u_i \in N_{P_1[u_1,u_c)}(u_p) \cap N_{P_1[u_1,u_c)}(u_p)^+ \neq \emptyset$.
Operation 9: Reset $P_1 = u_0u_1...u_{i-1}u_pu_iu_{i+1}...u_{p-1}$.

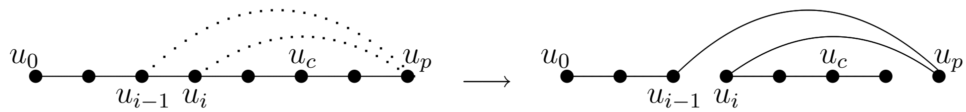


Figure 9

Note that except for Operation 9, all operations extend P_1 by at least one vertex. And Operation 9 increases c by one.

Algorithm 1.
Input: A connected graph G.
Output: Either a hamiltonian path P_1, or two paths P_1 and P_2 sharing only one common vertex u_c, and P_1 can not be extended by Operations 1 to 9.
Step 1. Set $P_1 = u_0$ where u_0 is an arbitrary vertex in G.
Step 2. Extend P_1 repeatedly by Operation 1 untill such operation can no longer be carried out.
Step 3. If $V(G) \setminus V(P_1) = \emptyset$, then output P_1 which is a hamiltonian path; stop. Else, if one of circumstances 2 to 7 happens, then extend P_1 by the corresponding operation; go to Step 2.
Step 4. If $V(G) \setminus V(P_1) = \emptyset$, then output P_1; stop. Else, let u_c be the last vertex on P_1 which has a neighbor outside of P_1; set $v_0 = u_c$; find a maximal path P_2 in $G - P_1$ starting at v_0, i.e., as long as there is a vertex $v \in V(G) - V(P_1 \cup P_2)$ adjacent to the other end of P_2, then extend P_2 by adding v.
Step 5. If circumstance 8 or circumstance 9 happens, then extend P_1 by the corresponding operation; go to Step 2. Else, output P_1, P_2 and u_c; stop. □

Given a path $P = u_0u_1...u_p$, let $\mathcal{Q} := \{Q_\ell[u_{i_\ell}, u_{j_\ell}] \ : \ 1 \le \ell \le m\}$ be a set of internally disjoint paths such that $Q_\ell \cap P = \{u_{i_\ell}, u_{j_\ell}\}$ and

$$0 = i_1 < i_2 < j_1 \le i_3 < j_2 \le i_4 \ ... \le i_m < j_{m-1} < j_m = p.$$

Denote by $\mathcal{P}$ the set of segments of P divided by u_{i_ℓ}'s and u_{j_ℓ}'s. A *vine* of P is composed of elements in $\mathcal{Q} \cup \mathcal{P}$ alternatively (see Figure 10).

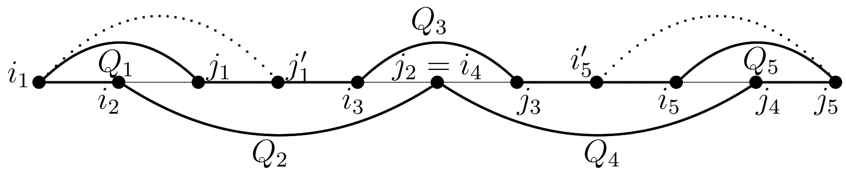


Figure 10 The vine is indicated by the bold lines.

For our purpose, we will find a vine P_V of P in a 2-connected graph with $N_P(u_0) \cup N_P(u_p) \subseteq P_V$, which can be realized by the following algorithm.

Algorithm 2.
Input: A path $P = u_0u_1...u_p$.
Output: A vine P_V with $N(u_0) \cup N(u_p) \subseteq P_V$.
Step 1. Set $i_1 = 0$. Let j_1' be the largest index such that $u_{j_1'}$ is adjacent to u_0. Set $\ell = 2$, $v = u_{j_1'}$, $w = u_0$.
Step 2. Find a path Q_ℓ in $G - v$ internally disjoint with P, connecting a vertex $u_{i_\ell} \in P[w, v^-]$ with a vertex $u_{j_\ell} \in P[v^+, u_p]$, such that j_ℓ is as large as possible (such a path always exists since G is 2-connected).
Step 3. If $j_\ell = p$, then choose i_ℓ as large as possible, go to Step 4. Else, set $w = v$, $v = u_{j_\ell}$, $\ell = \ell + 1$, go to Step 2.
Step 4. Set j_1 to be the first index in the segment $[u_{i_2}^+, u_{j_1'}]$ such that $u_{j_1} \in N_P(u_0)$.
Step 5. If ℓ is even, then let

$$\begin{aligned} P_V := \; & Q_1[u_{i_1}, u_{j_1})P[u_{j_1}, u_{i_3})Q_3[u_{i_3}, u_{j_3})P[u_{j_3}, u_{i_5})...Q_{\ell-1}[u_{i_{\ell-1}}, u_{j_{\ell-1}})P[u_{j_{\ell-1}}, u_{j_\ell}) \\ & \overline{Q_\ell}[u_{j_\ell}, u_{i_\ell})\overline{P}[u_{i_\ell}, u_{j_{\ell-2}})\overline{Q_{\ell-2}}[u_{j_{\ell-2}}, u_{i_{\ell-2}})\overline{P}[u_{i_{\ell-2}}, u_{j_{\ell-4}})...\overline{Q_2}[u_{j_2}, u_{i_2})\overline{P}[u_{i_2}, u_{i_1}], \end{aligned}$$

and if ℓ is odd, then let

$$\begin{aligned} P_V := \; & Q_1[u_{i_1}, u_{j_1})P[u_{j_1}, u_{i_3})Q_3[u_{i_3}, u_{j_3})P[u_{j_3}, u_{i_5})...Q_{\ell-2}[u_{i_{\ell-2}}, u_{j_{\ell-2}})P[u_{j_{\ell-2}}, u_{i_\ell}) \\ & Q_\ell[u_{i_\ell}, u_{j_\ell})\overline{P}[u_{j_\ell}, u_{j_{\ell-1}})\overline{Q_{\ell-1}}[u_{j_{\ell-1}}, u_{i_{\ell-1}})\overline{P}[u_{i_{\ell-1}}, u_{j_{\ell-3}})...\overline{Q_2}[u_{j_2}, u_{i_2})\overline{P}[u_{i_2}, u_{i_1}]. \end{aligned}$$

□

Suppose m is the ℓ-value at the end of the algorithm. Then $u_{j_m} = u_p$. By the choice of j_ℓ in Step 2, we see that $N_P(u_p) \subseteq P[u_{j_{m-2}}, u_{p-1}]$. By the choice of i_m in Step 3, we have $N_P(u_p) \cap P[u_{i_m}^+, u_{j_{m-1}}^-] = \emptyset$. So

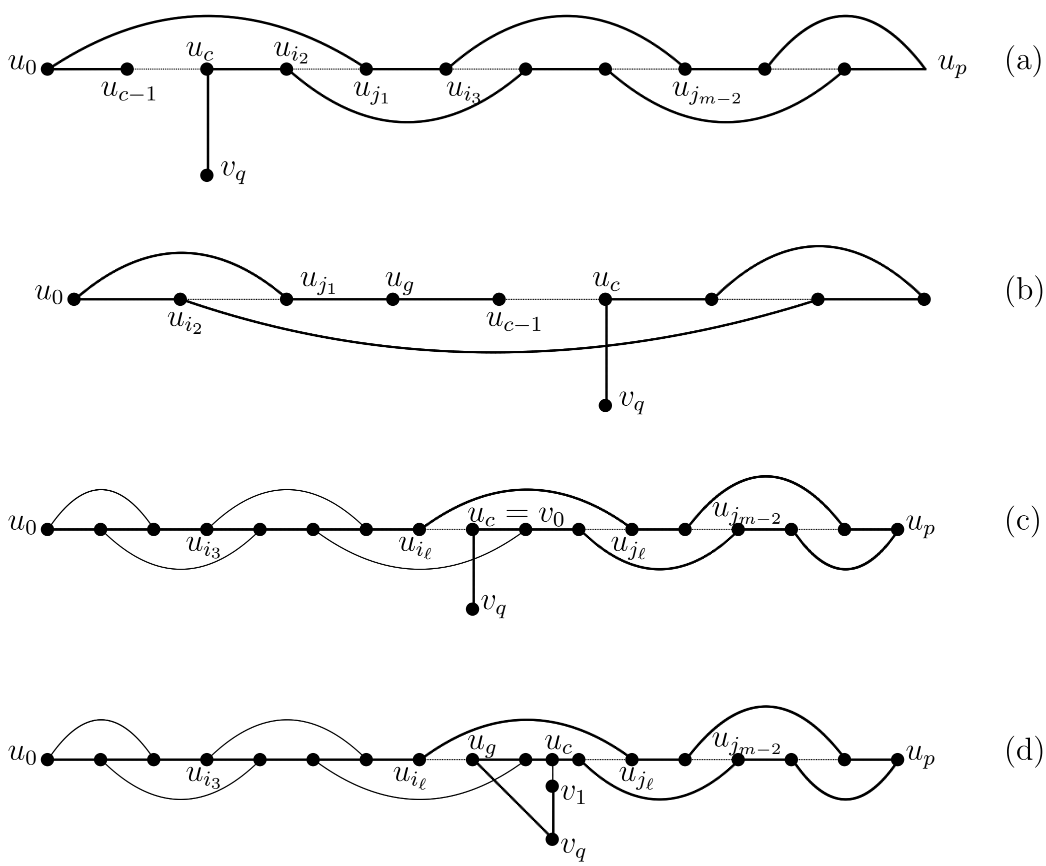
$$N_P(u_p) \subseteq P[u_{j_{m-2}}, u_{p-1}] - P[u_{i_m}^+, u_{j_{m-1}}^-] \subseteq P_V. \tag{5}$$

Similarly, by the choice of j_1' in Step 1 and the choice of j_1 in Step 4, we have

$$N_P(u_0) \subseteq P[u_1, u_{i_3}] - P[u_{i_2}^+, u_{j_1}^-] \subseteq P_V. \tag{6}$$

The next algorithm finds a path P satisfying conditions (1) to (4). For simplicity, we abuse the notation a little by, for example, using $P_V(u_{i_\ell}, u_c]$ to denote $P_V(u_{i_\ell}, u_{j_{\ell-1}}]$ $\overline{P}_1(u_{j_{\ell-1}}, u_c]$ when $u_c \in (u_{i_\ell}, u_{j_{\ell-1}})$. The same denotation is used in the remaining of this paper when there is no danger of confusion.

Algorithm 3.
Input: A 2-connected graph G.
Output: A vertex x and a path P with length at least $\min\{|G|, \Gamma_3(x, P) + 1\}$ such that x is one end vertex of P.
Step 1. Use Algorithm 1 to find P_1, P_2 and u_c. If P_1 is hamiltonian, then set $P = P_1$; stop.
Step 2. Use Algorithm 2 to find P_V.
Step 3. Let ℓ be the largest integer such that $u_c \in (u_{i_\ell}, u_{j_\ell})$. If $(u_{i_\ell}, u_c) \cap N(v_q) = \emptyset$, then set $u_g = u_c$, $tag = 0$. Else, let u_g be the first vertex in $(u_{i_\ell}, u_c) \cap N(v_q)$, set $tag = 1$.
Step 4. If $\ell = 1$, then set $x = v_q$ and $P = \overline{P_1}[u_{c-1}, u_0]P_V(u_0, u_c]P_2(v_0, v_q]$ (see Figure 11 (a)), stop.
Step 5. If $(u_{j_{\ell-1}}, u_g) \cap N(u_0) \neq \emptyset$, then set $x = v_q$ and $P = \overline{P_1}[u_{c-1}, u_{j_{\ell-1}}]P_1[u_0, u_{i_\ell}]P_V(u_{i_\ell}, u_c]P_2(v_0, v_q]$ (see Figure 11 (b)), stop.
Step 6. Set $x = u_0$. If $[u_{j_{\ell-1}}, u_g) \cap N(u_p) = \emptyset$, then set $P = P_1[u_0, u_{i_\ell}]P_V(u_{i_\ell}, u_c]$ (see Figure 11 (c) or (d)). Else, let u_f be the last vertex in $[u_{j_{\ell-1}}, u_g) \cap N(u_p)$ and set $P = P_1[u_0, u_f]\overline{P_1}[u_p, u_c]$ (see Figure 11 (e)).
Step 7. If $tag = 0$, then set $P = PP_2(v_0, v_q]$. Else, set $P = P\overline{P_1}(u_c, u_g]\overline{P_2}[v_q, v_1]$.



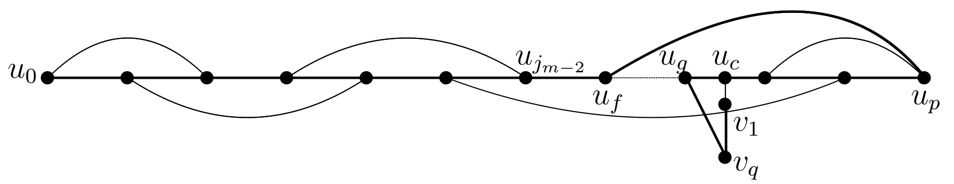


Figure 11. Path P is indicated by bold lines.

We will show that the path P found by Algorithm 3 indeed satisfies conditions (1) to (4). For this purpose, we need the following lemmas.

Lemma 1. *Let $P = u_0u_1....u_p$ be a path in G and $y, z \in V(G) - P$ such that $N_P(z) \cap N_P(y)^+ = \emptyset$. Then*

$$d_P(y) + d_P(z) \leq |P| + 1. \tag{7}$$

The equality holds only if $u_p \in N_P(y)$. Furthermore, if $N_P(y) \cap N_P(y)^+ = \emptyset$, then equality holds only when $u_p \in N_P(y) \cap N_P(z)$.

Proof. Since $(N_P(z) \cup (N_P(y) - \{u_p\})^+) \subseteq V(P)$ and $N_P(z) \cap N_P(y)^+ = \emptyset$, we have $|P| \geq |N_P(z)| + |(N_P(y) - \{u_p\})^+| \geq d_P(z) + d_P(y) - 1$. Equality holds only if

$$V(P) = N_P(z) \cup (N_P(y) - \{u_p\})^+ \tag{8}$$

and $u_p \in N_P(y)$. Furthermore, if $N_P(y) \cap N_P(y)^+ = \emptyset$ and equality holds, then it follows from $u_p \in N_P(y)$ that $u_p \notin N_P(y)^+$. By (8), we have $u_p \in N_P(z)$. □

Lemma 2. *Let $P = u_0u_1....u_p$ be a path in G and $x, y, z \in V(G) - P$ such that $N_P(x) \cap N_P(x)^+ = (N_P(y) \cup N_P(z)) \cap N_P(x)^+ = N_P(y) \cap N_P(y)^+ = N_P(z) \cap N_P(y)^+ = \emptyset$. Then*

$$\Gamma_P(x, y, z) \leq |P| + 1. \tag{9}$$

Furthermore, if equality holds and $u_p \notin N_P(x)$, then $u_p \in N_P(y) \cap N_P(z)$.

Proof. If $N_P(x) = \emptyset$, then it follows from Lemma 1 that

$$\Gamma_P(x, y, z) = d_P(y) + d_P(z) \leq |P| + 1, \tag{10}$$

with equality only when $u_p \in N_P(y) \cap N_P(z)$.

So, suppose $N_P(x) = \{u_{i_1}, u_{i_2}, ..., u_{i_t}\} \neq \emptyset$. Consider a segment $P(u_{i_j}, u_{i_{j+1}}]$, $1 \leq j < t$. By Lemma 1, noting that $u_{i_j+1} \notin N_P(y) \cup N_P(z)$, we see that

$$d_{P(u_{i_j}, u_{i_{j+1}}]}(y) + d_{P(u_{i_j}, u_{i_{j+1}}]}(z)) \leq |P(u_{i_j}, u_{i_{j+1}}]|, \tag{11}$$

with equality only when $u_{i_{j+1}} \in N(y) \cap N(z)$. Therefore

$$\begin{aligned} \Gamma_{P(u_{i_j}, u_{i_{j+1}}]}(x, y, z) &= 1 + d_{P(u_{i_j}, u_{i_{j+1}}]}(y) + d_{P(u_{i_j}, u_{i_{j+1}}]}(z) - |\{u_{i_{j+1}}\} \cap N(y) \cap N(z)| \\ &\leq |P(u_{i_j}, u_{i_{j+1}}]|. \end{aligned} \tag{12}$$

For the first segment $P[u_0, u_{i_1}]$ and the last segment $P(u_{i_t}, u_p]$, similar to the above we may get

$$\Gamma_{P[u_0,u_{i_1}]}(x, y, z) \leq |P[u_0, u_{i_1}]| + 1 \tag{13}$$

and

$$\Gamma_{P(u_{i_t},u_p]}(x, y, z) \leq |P(u_{i_t}, u_p]|. \tag{14}$$

Then (9) follows by adding (12), (13), (14) together. If equality holds for (9), then equality also holds for (14). If furthermore $u_p \notin N_P(x)$, then similar to the deduction of (11), we have

$$\Gamma_{P(u_{i_t},u_p]}(x, y, z) = d_{P(u_{i_t},u_p]}(y) + d_{P(u_{i_t},u_p]}(z) \leq |P(u_{i_t}, u_p]|,$$

with equality only when $u_p \in N_P(y) \cap N_P(z)$. □

Next, we will prove the main theorem.

Proof of Theorem 7 Since each of the nine operations either extends P_1 by at least one vertex or increases c by one, at most $O(n)$ extensions are needed. Furthermore, each extension can be completed in $O(m)$ time by graph searching (see for example [9]). For the same reason, the time complexity of Algorithm 2 and Algorithm 3 is also $O(m)$. So, P can be found in $O(nm)$ time. Next, we will prove that P satisfies conditions (1) to (4), and thus has the desired length.

Without loss of generality, we assume that G has no hamiltonian path. Let $P_1 = u_0u_1....u_p$ and $P_2 = v_0v_1...v_q$ be the paths found by Algorithm 1, P_V the vine found by Algorithm 2, and m the ℓ-value at the end of Algorithm 2. By Operations 1 and 3, u_0, u_p, v_q are independent (Condition (1)). Condition (2) is obviously satisfied by the definition of the path P in Algorithm 3. Furthermore,

$$N_{P_1}(v_q) \cap N_{P_1}(v_q)^+ = \emptyset \quad \text{(by Operation 2)}, \tag{15}$$
$$N_{P_1[u_1,u_c)}(u_p) \cap N_{P_1[u_1,u_c)}(v_q)^+ = \emptyset \quad \text{(by Operation 8)}, \tag{16}$$
$$N_{P_1[u_1,u_c)}(u_0) \cap N_{P_1[u_1,u_c)}(v_q)^+ = \emptyset \quad \text{(by Operation 6)}, \tag{17}$$
$$N_{P_1[u_1,u_c)}(u_p) \cap N_{P_1[u_1,u_c)}(u_p)^+ = \emptyset \quad \text{(by Operation 9)}, \tag{18}$$
$$N_{P_1}(u_0) \cap N_{P_1}(u_p)^+ = \emptyset \quad \text{(by Operation 4)}. \tag{19}$$

By (5) and (6),

$$N(u_0) \subseteq P_1(u_0, u_{i_2}] \cup P_1[u_{j_1}, u_{i_3}], \tag{20}$$
$$N(u_p) \subseteq P_1[u_{j_{m-2}}, u_{i_m}] \cup P_1[u_{j_{m-1}}, u_p). \tag{21}$$

By the definition in Algorithm 1,

$$N(v_q) \subseteq P_1[u_1, u_c] \cup P_2. \tag{22}$$

Recall that ℓ is such that $u_c \in P_1(u_{i_\ell}, u_{j_\ell})$. It follows from (22) that the only possible neighbors of v_q which may be missed lie in the segment (u_{i_ℓ}, u_c). However, this can be compensated by the choice of u_g (Step 3 and Step 7 of Algorithm 3). So, $N(v_q) \subseteq P$. If $\ell \geq 3$, then $N(u_0) \subseteq P$ by (20). If $\ell \leq 2$, then by noting that $[u_g, u_c] \subseteq P$ (Step 7), we also have $N(u_0) \subseteq P$ by the definition of P in Step 4 and Step 5. Similarly, u_f is taken to ensure that $N(u_p) \subseteq P$ (Step 6). So, Condition (3) is satisfied. In the following, we will show Condition (4). To this end, we first prove the following three claims.

Claim 1. Suppose $Q = u_i u_{i+1} ... u_{c-1}$ $(i > 0)$. Then $\Gamma_Q(v_q, u_p, u_0) \leq |Q|$.

By taking $x = v_q, y = u_p, z = u_0$ in Lemma 2, and by (1) and (15) to (19), we see that

$$\Gamma_Q(v_q, u_p, u_0) \leq |Q| + 1. \tag{23}$$

Note that $u_{c-1} \notin N(v_q)$ since otherwise P_1 can be extended by Operation 2. If equality holds in (23), then $u_{c-1} \in N(u_0) \cap N(u_p)$ by Lemma 2, and thus P_1 can be extended by Operation 5, a contradiction.

Claim 2. $\Gamma_{P_V[u_{j_\ell}, u_c]}(v_q, u_p, u_0) \leq |P_V[u_{j_\ell}, u_c]|$ when $\ell \geq 2$ and $\Gamma_{P_V[u_{j_1}, u_c]}(v_q, u_p, u_0) \leq |P_V[u_{j_1}, u_c]| + 1$ when $\ell = 1$.

If

$$d_{P_1(u_c, u_{i_{\ell+1}}]}(u_0) + d_{P_1(u_c, u_{i_{\ell+1}}]}(u_p) = |P_1(u_c, u_{i_{\ell+1}}]| + 1,$$

then by Lemma 1, $u_{c+1} \in N(u_0)$, which contradicts Operation 6. So,

$$d_{P_1(u_c, u_{i_{\ell+1}}]}(u_0) + d_{P_1(u_c, u_{i_{\ell+1}}]}(u_p) \leq |P_1(u_c, u_{i_{\ell+1}}]|.$$

Combining this with Lemma 1 and (20), we see that when $\ell = 1$,

$$\begin{aligned} & d_{P_V[u_{j_1}, u_c)}(u_0) + d_{P_V[u_{j_1}, u_c)}(u_p) \\ = \; & d_{P_1(u_c, u_{i_2}]}(u_0) + d_{P_1(u_c, u_{i_2}]}(u_p) + d_{P_1[u_{j_1}, u_{i_3}]}(u_0) + d_{P_1[u_{j_1}, u_{i_3}]}(u_p) + d_{P_1[u_{j_2}, u_p) \cap P_V}(u_p) \\ \leq \; & |P_1(u_c, u_{i_2}]| + |P_1[u_{j_1}, u_{i_3}]| + 1 + |P_1[u_{j_2}, u_p) \cap P_V| \\ = \; & |P_1(u_c, u_p) \cap P_V| + 1 = |P_1[u_c, u_p] \cap P_V| - 1 = |P_V[u_{j_1}, u_c]| - 1, \end{aligned}$$

and when $\ell \geq 2$,

$$\begin{aligned} & d_{P_V[u_{j_\ell}, u_c)}(u_0) + d_{P_V[u_{j_\ell}, u_c)}(u_p) \\ = \; & d_{P_1(u_c, u_{i_{\ell+1}}]}(u_0) + d_{P_1(u_c, u_{i_{\ell+1}}]}(u_p) + d_{P_1[u_{j_\ell}, u_p) \cap P_V}(u_p) \\ \leq \; & |P_1(u_c, u_{i_{\ell+1}}]| + |P_1[u_{j_\ell}, u_p) \cap P_V| \\ = \; & |P_1(u_c, u_p) \cap P_V| = |P_1[u_c, u_p] \cap P_V| - 2 = |P_V[u_{j_\ell}, u_c]| - 2. \end{aligned}$$

Then the claim follows from

$$\Gamma_{P_V[u_{j_\ell}, u_c]}(v_q, u_p, u_0) \;\; = \;\; d_{P_V[u_{j_\ell}, u_c)}(u_0) + d_{P_V[u_{j_\ell}, u_c)}(u_p) + \Gamma_{\{u_c\}}(v_q, u_p, u_0)$$

and the fact $\Gamma_{\{u_c\}}(v_q, u_p, u_0) \leq 2$.

Claim 3. Suppose $Q = u_0 u_1 ... u_i$. Then $\Gamma_Q(v_q, u_p, u_0) \leq |Q|$. If furthermore $i = c - 1$, then $\Gamma_Q(v_q, u_p, u_0) \leq |Q| - 1$.

In fact, by Lemma 2,

$$\Gamma_Q(v_q, u_p, u_0) = \Gamma_{Q \setminus u_0}(v_q, u_p, u_0) \leq |Q \setminus u_0| + 1 = |Q|.$$

If furthermore $i = c - 1$, then the above inequality becomes strict by Claim 1.

Clearly,

$$\Gamma_{P_2(v_0, v_q]}(v_q, u_p, u_0) = d_{P_2(v_0, v_q)}(v_q) \leq |P_2(v_0, v_q]| - 1. \tag{24}$$

By Claim 1, Claim 2, Claim 3 and inequality (24), the theorem is proved. □

References

[1] J-C. Bermond, On Hamiltonian Walks, *in "Proc. Fifth British Combinatorial Conference, Aberdeen, 1975," Utilitas Math. (1975) 41-51.*

[2] B. Bollobás, Extremal Graph Theory, *in "Handbook of combinatorics Volume II, pages 1231-1292, Elsevier, Amsterdam-Lausanne-New York-Oxford-Shannon-Tokyo, 1995".*

[3] J.A. Bondy, Basic Graph Theory: Paths and Circuits, *in "Handbook of combinatorics Volume I, pages 3-110, Elsevier, Amsterdam-Lausanne-New York-Oxford-Shannon-Tokyo, 1995".*

[4] J.A. Bondy and U.S.R. Murty, Graph Theory with Applications, *Macmillan Press, 1976.*

[5] G.A. Dirac, Some Theorems on Abstract Graphs, *Proc. London Math. Soc. (3) 2 (1952) 69-81.*

[6] E. Flandrin, H. A. Jung and H. Li, Degree Sum, Neighbourhood Intersections and Hamiltonism, *Discrete Math. 90 (1991) 41-52.*

[7] R.J. Gould, Updating the Hamiltonian Problem - A Survey, *Journal of Graph Theory, Vol.15, No.2 (1991) 121-157.*

[8] Jan Van den Heuvel, Degree and Toughness Conditions for Cycles in Graphs, *' Ph.D. Thesis, Faculty of Applied Math., University of Twente, 7500 AE Enschede, The netherland, 1993.*

[9] B. Korte and J. Vygen, Combinatorial Optimization: Theory and Algorithms, *Springer-Verlag-Berlin-Heidelberg-New York, 2000.*

[10] O. Ore, Note on Hamilton Circuits, *Amer. Math. Monthly 67 (1960) 55.*

[11] B. Wei, Longest Cycles in 3-connected Graphs, *Discrete Math. 170 No. 1-3 (1997) 195-201.*