Title	The NP-completeness of EULERIAN RECURRENT LENGTH (Algebra, Languages and Computation)
Author(s)	Jimbo, Shuji; Oshie, Yasuaki; Hashiguchi, Kosaburo
Citation	数理解析研究所講究録 (2005), 1437: 107-115
Issue Date	2005-06
URL	http://hdl.handle.net/2433/47477
Right	
Type	Departmental Bulletin Paper
Textversion	publisher

数理解析研究所講究録 1437 巻 2005 年 107-115

The NP-completeness of EULERIAN RECURRENT LENGTH

Shuji Jimbo* Yasuaki Oshie
Kosaburo Hashiguchi
Faculty of Engineering, Okayama University

February 22, 2005

Abstract

It is shown that it is NP-complete to determine the maximum length of the shortest cycles in Eulerian trails of an arbitrary Eulerian graph. By the authors, the maximum length of the shortest cycles in Eulerian trails of an Eulerian graph is referred to as Eulerian recurrent length of the Eulerian graph, and the decision problem above is named EULERIAN RECURRENT LENGTH.

Keywords: NP-complete, Eulerian graphs, cycles, path decompositions.

1 Introduction

Computations to find a parameter of an Eulerian trail of a Eulerian graph are discussed. We begin to define several technical terms in graph theory. A trail of a graph is a walk in which all the edges are distinct. An Eulerian trail of a graph is a closed trail containing all the edges of the graph. A connected graph is Eulerian if there exists an Eulerian trail of the graph. It is well known that a connected graph G is Eulerian if and only if the degree of each vertex of G is even. Hence, it is very easy to determine whether an arbitrary graph has an Eulerian trail or not. The Eulerian recurrent length of an Eulerian graph is the maximum length of a shortest cycle in an Eulerian trail of the Eulerian graph. More precisely, letting $l(c)$ denote the length of walk c, $C(t)$ the set of all cycles in walk t, and $E(G)$ the set of all Eulerian trails in graph G, the Eulerian length of a graph G is defined to be $\max_{t\in E(G)} \min_{c\in C(t)} l(c)$. The terminology of graph theory given in [5] is chiefly used in this paper.

We define the following decision problem referred to as **EULERIAN RECURRENT LENGTH**, and shall prove that it is NP-complete.

*E-mail: `jimbo@kiso.it.okayama-u.ac.jp`

EULERIAN RECURRENT LENGTH
INSTANCE: A graph $G = (V, E)$ and a positive integer $J \leqq |V|$.
QUESTION: Is there an Eulerian trail T of G such that the length of every cycle in T is greater than or equal to J?

The terminology of the theory of NP-completeness given in [2] is chiefly used in this paper.

The NP-completeness of EULERIAN RECURRENT LENGTH implies the intractability of determining whether an arbitrary graph has an Eulerian trail that have no cycle of length less than a given lower limit or not. It is clear that EULERIAN RECURRENT LENGTH is in the class NP. To prove that the problem is NP-complete, it suffices to exhibit a polynomial reduction from the known NP-complete problem 3SAT. The following definition of problem 3SAT is quoted from [3]. A set of clauses $C = \{C_1, C_2, \ldots, C_r\}$ in variables $u_1, u_2, \ldots, u_s$ is given, each clause C_i consisting of three literals $l_{i,1}, l_{i,2}, l_{i,3}$, where a literal $l_{i,j}$ is either a variable u_k or its negation $\overline{u_k}$. The problem is to determine whether C is satisfiable, that is, whether there is a truth assignment to the variables which simultaneously satisfies all the clauses in C. A clause is satisfied if one or more of its literals have value "true".

2 Definitions and a fundamental lemma

A path is a trail whose vertices are distinct, except that, possibly, the initial vertex is equal to the final one. If the initial and final vertices of a path are distinct, then the path is referred to as a *non-closed* path. Let k be a positive integer, and G a graph that has $2k$ vertices, say $v_1, v_2, \ldots, v_{2k}$, of odd degree. Since G is obtained from an Eulerian graph by deleting k edges such that no two of them share a common vertex, the edge family of G can be divided into k edge disjoint non-closed trails. The k trails may be k paths. For example, if G is the graph obtained from the complete graph K_{2k+1} by deleting such k edges, then the edge family of G can be divided into k edge disjoint non-closed paths, each of which is from one vertex incident with a deleted edge to the other one.

For a non-closed trail T of a graph, $I(T)$ and $F(T)$ denote the initial and final vertices of T, respectively.

Definition 1 *Let k be a positive integer, and G a graph that has $2k$ vertices, say $v_1, v_2, \ldots, v_{2k}$, of odd degree. A set of k non-closed trails $\mathcal{T} = \{T_1, T_2, \ldots, T_k\}$ in G is a trail decomposition of G if every edge of G belongs to exactly one trail in $\mathcal{T}$. A trail decomposition of a graph is a path decomposition of the graph if every trail in the trail decomposition is a path.*

For a trail decomposition $\mathcal{T} = \{T_1, T_2, \ldots, T_k\}$ of G, $\mathrm{IF}(\mathcal{T})$ denotes

$$\{\{I(T_1), F(T_1)\}, \{I(T_2), F(T_2)\}, \ldots, \{I(T_k), F(T_k)\}\},$$

that is to say the family of k sets each of which consists of the initial and final vertices of a trail in $\mathcal{T}$. For a path decomposition $\mathcal{P}$ of G, $\mathcal{F} = \mathrm{IF}(\mathcal{P})$ is referred to as an initial-final family associated with $\mathcal{P}$. If we need not specify the path decomposition $\mathcal{P}$, then we refer to $\mathcal{F}$ simply as an initial-final family of G.

For a finite set S, $|S|$ denotes the number of the elements belonging to S.

Lemma 1 *For any positive integer $k \geqq 2$, there exists a graph $H(k) = (V_k, E_k)$ such that*

1. *$H(k)$ has exactly $2k$ vertices $v_0(k), v_1(k), \ldots, v_{2k-1}(k)$ of degree 1,*
2. *$|V_k| = 4k^2 - 4k + 1$ and $|E_k| = 8k^2 - 10k$,*
3. *for any trail decomposition $\mathcal{T}$ of $H(k)$, if $\mathcal{T}$ includes a cycle, then $\mathcal{T}$ includes a cycle of length less than or equal to $8k - 13$,*
4. *all of the initial-final families of G are $\{\{v_0(k), v_1(k)\}, \{v_2(k), v_3(k)\}, \ldots, \{v_{2k-2}(k), v_{2k-1}(k)\}\}$ and $\{\{v_0(k), v_{2k-1}(k)\}, \{v_2(k), v_1(k)\}, \ldots, \{v_{2k-2}(k), v_{2k-3}(k)\}\}$,*
5. *for any path decomposition $\mathcal{P}$ of G, all of the paths in $\mathcal{P}$ share a common vertex w, and, for every initial or final vertex in a path P in $\mathcal{P}$, the length of the section between w and the vertex on the path is $4k - 5$, and,*
6. *for any path decomposition $\mathcal{P}$ of G, all of the paths in $\mathcal{P}$ are of the same length $8k - 10$.*

Proof. We shall configure the vertex-set of $H(k)$ as a set of points on a plane. Let $r_0, r_1, \ldots, r_{2k-1}$ be $2k$ distinct half-lines with the common origin o. For each $i \in \{0, 1, \ldots, 2k-1\}$, and for every positive integer d, let $r_i(d)$ denote the point z on r_i such that the distance between o and z is d. Let S denote the set of pairs of two non-negative integers $\{(x, y) \mid x \in \mathbb{Z},\ y \in \mathbb{Z},\ 0 \leqq x,\ 0 \leqq y,\ \text{and}\ y - 2k + 1 < x < y - 1 < 2k - 1\}$. The vertex-set V_k is defined to be $\{r_x(x+y) \mid (x, y) \in S\} \cup \{r_y(x+y) \mid (x, y) \in S\} \cup \{r_i(4k) \mid i \in \{0, 1, \ldots, 2k-1\}\} \cup \{o\}$. For $i \in \{0, 1, \ldots, 2k-1\}$, and for non-negative integer d with $d < 4k$, $\hat{r}_i(d)$ denote $r_i(\delta) \in V_k$ such that $d < \delta$ and, for any integer x with $d < x < \delta$, $r_i(x) \notin V_k$. The edge-set E_k is defined to be $\{o\hat{r}_0(0), o\hat{r}_1(0), \ldots, o\hat{r}_{2k-1}(0)\} \cup \bigcup_{(x,y) \in S} \{\hat{r}_x(x+y) r_y(x+y), r_x(x+y) \hat{r}_y(x+y), r_x(x+y) r_y(x+y), r_x(x+y) r_y(x+y)\}$, where symbol $\cup$ denotes multi-set union operation, and hence the arguments are multi-sets.

Statements (1) and (2) follow immediately from the construction of $H(k)$ above. Statements (3), (4), (5), and (6) are shown as follows.

Let $v = r_x(x+y)$, $v' = \hat{r}_x(x+y)$, $w = r_y(x+y)$, and $w' = \hat{r}_y(x+y)$ be vertices of $H(k)$. Notice that each of $x < y$ and $x > y$ may hold. Let v'' denote the vertex of $H(k)$ such that $v''v \in E_k$, $v'' \neq w$, and $v'' \neq w'$, and w'' the vertex of $H(k)$ such that $w''w \in E_k$, $w'' \neq v$, and $w'' \neq v'$. If a trail decomposition of $H(k)$ contains a trail that includes the sub-trail $v' \to w \to w''$, then the decomposition must contain a trail that includes the cycle $v \to w \to v$. Hence, if any trail in a trail decomposition of G has no cycles of length 2, then the subgraph of $H(k)$ induced by the six vertices $\{v, v', v'', w, w', w''\}$ is decomposed into two paths as either

1. $v' \to w \to v \to v''$ and $w' \to v \to w \to w''$, or
2. $v' \to w \to v \to w'$ and $v'' \to v \to w \to w''$.

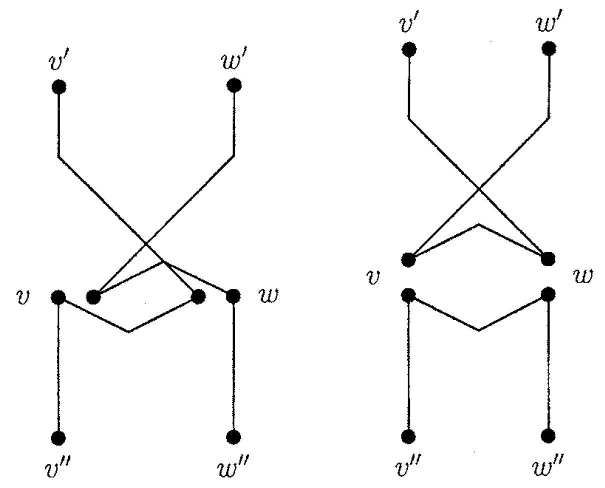

Figure 1: Two types of path decomposition.

Those types of decomposition are illustrated in Fig. 1.

If the latter decomposition occurs, then we choose such four vertices so that the distance between o and v, which is equal to the one between o and w, is minimum. By our choice, the trail that includes $v'' \to v \to w \to w''$ must contain a path that connects v and o and one that connects w and o. Those paths and edge vw compose a cycle of length at most $8k - 13 = 2 \cdot 2(2k-4) + 3$. Thus, if a trail decomposition T has no trail that includes a cycle of length less than or equal to $8k - 13$, then every trail of T consists of two paths each of which connects an end-vertex of $H(k)$ and o, and is of length $4k - 5 = 2(2k-3) + 1$. Furthermore, it is easy to see that, for every $(x, y) \in S$, the path that connects $r_x(4k)$ and o and the one that connect $r_y(4k)$ and o intersect at $r_x(x+y)$ and $r_y(x+y)$, composing a trail that includes a cycle of length less than or equal to $8k - 13$. By renaming vertex $r_i(4k)$ $v_i(k)$ for each $i \in \{0, 1, \ldots, 2k-1\}$, and vertex o w, statements (3), (4), (5), and (6) follow immediately from those facts. ■

3 The components used in the reduction

We will provide a positive integer constant μ and, given an instance C of the problem 3SAT, show how to construct a graph G such that the Eulerian recurrent length of G is greater than or equal to μ if and only if C is satisfiable.

The graph G will be put together from components which carry out specific tasks. There are three types of component, satisfaction-testing components, variable-setting components, and garbage-collecting components. Let $C = \{C_1, C_2, \ldots, C_r\}$ be a set of r clauses in variables $u_1, u_2, \ldots, u_s$.

Every clause in C is one-to-one corresponding to a satisfaction-testing component. For each $i \in \{1, 2, \ldots, r\}$, the satisfaction-testing component corresponding to C_i is denoted by $\Gamma_C(C_i)$, and is isomorphic to the graph that consists of three disjoint edges, namely

$$\begin{aligned}\Gamma_C(C_i) &= (\{a(i,1), b(i,1), a(i,2), b(i,2), a(i,3), b(i,3)\},\\ &\quad \{b(i,1)a(i,2), b(i,2)a(i,3), b(i,3)a(i,1)\}).\end{aligned}$$

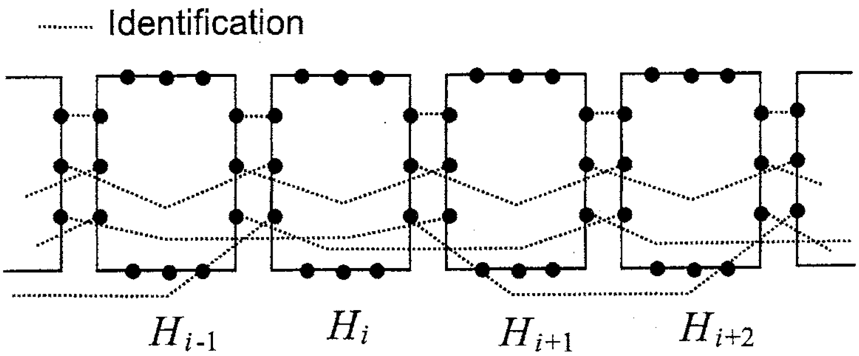

Figure 2: Connection around H_i.

Every variable in C is one-to-one corresponding to a variable-setting component. For each $j \in \{1, 2, \ldots, s\}$, the variable-setting component corresponding to u_j is denoted by $\Gamma_C(u_j)$, and is isomorphic to the graph $\Delta(m)$ defined below, where m denotes the number of clauses that includes variable u_k or its negation $\overline{u_k}$, or 3 if the number is less than 3. The definition of the garbage-collecting component Γ_C shall be stated in the next section.

For an integer x and a positive integer y, $x \bmod y$ denotes the unique integer z in $\{0, 1, \ldots, y-1\}$ such that $x - z$ is a multiple of y. Let m be an integer greater than 2. The $\Delta(m)$ is constructed as follows. Let H_0, H_1, $\ldots$, and H_{2m-1} be $2m$ distinct graphs isomorphic to $H(6)$. For each H_i and each $j \in \{0, 1, \ldots, 11\}$, v_j^i denotes the vertex of H_i corresponding to vertex $v_j(6)$ of $H(6)$.

Roughly speaking, $\Delta(m)$ is constructed by joining H_0, H_1, $\ldots$, and H_{2m-1} in a ring. Its precise definition is as follows. Graph $\Delta(m)$ is obtained from $H_0, H_1, \ldots, H_{2m-1}$ by identifying each vertex in $6m$ end-vertices with another one as follows:

$$\begin{aligned}
&\text{Identify } v_3^i \text{ with } v_{11}^{(i+1) \bmod 2m} \text{ for each } i \in \{0, 1, \ldots, 2m-1\},\\
&\text{identify } v_4^i \text{ with } v_{10}^{(i+2) \bmod 2m} \text{ for } i \in \{0, 1, \ldots, 2m-1\}, \text{ and}\\
&\text{identify } v_5^i \text{ with } v_9^{(i+3) \bmod 2m} \text{ for } i \in \{0, 1, \ldots, 2m-1\}.
\end{aligned}$$

Fig. 2 illustrates the connection around H_i in $\Delta(m)$.

By definition, the set of all the end-vertices of $\Delta(m)$ is $\left(\bigcup_{i=0}^{2m-1}\{v_0^i, v_1^i, v_2^i\}\right) \cup \left(\bigcup_{i=0}^{2m-1}\{v_6^i, v_7^i, v_8^i\}\right)$. For a graph G, $V(G)$ and $E(G)$ denote the vertex set and edge family of G, respectively.

Lemma 2 *Let m be an integer greater than 2. Then, the following statements hold.*

1. *Equations $|V(\Delta(m))| = 236m$ and $|E(\Delta(m))| = 456m$ hold.*

2. *For any trail decomposition $\mathcal{T}$ of $\Delta(m)$, if $\mathcal{T}$ includes a cycle, then $\mathcal{T}$ includes a cycle of length less than or equal to 152.*

3. *All of the initial-final families of* $\Delta(m)$ *are*

$$S_+(m) = \bigcup_{i=0}^{2m-1} \{\{v_0^i, v_1^i\}, \{v_2^i, v_8^{(i+2) \bmod 2m}\}, \{v_6^i, v_7^i\}\} \text{ and}$$
$$S_-(m) = \bigcup_{i=0}^{2m-1} \{v_1^i, v_2^i\}, \{\{v_0^i, v_6^{(i-2) \bmod 2m}\}, \{v_7^i, v_8^i\}\}.$$

4. *For any path decomposition* $\mathcal{P}$ *of* $\Delta(m)$ *such that* $\mathrm{IF}(\mathcal{P}) = S_+(m)$ *and any* $i \in \{0, 1, \ldots, 2m-1\}$, *every edge in the path in* $\mathcal{P}$ *that connects either* v_0^i *and* v_1^i *or* v_6^i *and* v_7^i *belongs to* $E(H_i)$, *and every edge in the path in* $\mathcal{P}$ *that connects* v_2^i *and* $v_8^{(i+2) \bmod 2m}$ *belongs to* $H_i \cup H_{(i+1) \bmod 2m} \cup H_{(i-1) \bmod 2m} \cup H_{(i+2) \bmod 2m}$. *For any path decomposition* $\mathcal{P}$ *of* $\Delta(m)$ *such that* $\mathrm{IF}(\mathcal{P}) = S_-(m)$ *and any* $i \in \{0, 1, \ldots, 2m-1\}$, *every edge in the path in* $\mathcal{P}$ *that connects either* v_1^i *and* v_2^i *or* v_7^i *and* v_8^i *belongs to* $E(H_i)$, *and every edge in the path in* $\mathcal{P}$ *that connects* v_0^i *and* $v_6^{(i-2) \bmod 2m}$ *belongs to* $H_i \cup H_{(i-1) \bmod 2m} \cup H_{(i+1) \bmod 2m} \cup H_{(i-2) \bmod 2m}$.

5. *For any path decomposition* $\mathcal{P}$ *of* $\Delta(m)$, *the length of a path in* $\mathcal{P}$ *that connects two vertices in* $\bigcup_{i=0}^{2m-1}\{v_0^i, v_1^i, v_2^i\}$ *or two vertices in* $\bigcup_{i=0}^{2m-1}\{v_6^i, v_7^i, v_8^i\}$ *is* 38, *and that of a path in* $\mathcal{P}$ *that connects a vertex in* $\bigcup_{i=0}^{2m-1}\{v_0^i, v_1^i, v_2^i\}$ *and one in* $\bigcup_{i=0}^{2m-1}\{v_6^i, v_7^i, v_8^i\}$ *is* 152.

Proof. Statement (1) follows immediately from the structure of $V(\Delta(m))$.

Let $\mathcal{P}$ be a path decomposition of $\Delta(m)$. For each $i \in \{0, 1, \ldots, 2m-1\}$, the restriction of $\mathcal{P}$ to H_i, say $\mathcal{P}_i$, is a path decomposition of H_i. We say that $\mathcal{P}_i$ is of the positive type if $\mathrm{IF}(\mathcal{P}_i) = \{\{v_0^i, v_1^i\}, \{v_2^i, v_3^i\}, \ldots, \{v_{10}^i, v_{11}^i\}\}$, that $\mathcal{P}_i$ is of the negative type otherwise. Statement (2), (3), (4), and (5) follow from the fact that if there are $\mathcal{P}_i$ of the positive type and $\mathcal{P}_j$ of the negative type in $\mathcal{P}$, then there is a path in $\mathcal{P}$ that includes a cycle of length less than or equal to 152.

For instance, assume that every $\mathcal{P}_i$ is of the positive type. It is easy to see that, for each $i \in \{0, 1, \ldots, 2m-1\}$, $\{v_0^i, v_1^i\}$ and $\{v_6^i, v_7^i\}$ belong to $S_+(m)$. Furthermore, it can be shown that there is a path that connects v_2^i and $v_8^{(i+2) \bmod 2m}$ in $\mathcal{P}$ as follows. Since $\mathcal{P}_i$, $\mathcal{P}_{(i+1) \bmod 2m}$, $\mathcal{P}_{(i-1) \bmod 2m}$, and $\mathcal{P}_{(i+2) \bmod 2m}$ are all of the positive type, $\mathcal{P}_i$ includes a path that connects v_2^i and v_3^i, $\mathcal{P}_{(i+1) \bmod 2m}$ includes a path that connects $v_{11}^{(i+1) \bmod 2m}$ and $v_{10}^{(i+1) \bmod 2m}$, $\mathcal{P}_{(i-1) \bmod 2m}$ includes a path that connects $v_4^{(i-1) \bmod 2m}$ and $v_5^{(i-1) \bmod 2m}$, and $\mathcal{P}_{(i+2) \bmod 2m}$ includes a path that connects $v_9^{(i+2) \bmod 2m}$ and $v_8^{(i+2) \bmod 2m}$. Those four paths compose a path in $\mathcal{P}$ that connects v_2^i and $v_8^{(i+2) \bmod 2m}$. In the case where every $\mathcal{P}_i$ is of the negative type, we can obtain similar results. Statement (3), (4), and (5) are readily follow from those results.

Now, assume that there are $\mathcal{P}_i$ of the positive type and $\mathcal{P}_j$ of the negative type in $\mathcal{P}$. To prove Statement (2), it suffices to show that there is a path in $\mathcal{P}$ that includes a cycle of length less than or equal to 152. We can choose i and j above so that $j = (i-1) \bmod 2m$. It is easy to see that $\mathcal{P}_i$ has a path that connects v_{11}^i and v_{10}^i, $\mathcal{P}_j$ has a path that connects v_3^j and v_4^j, and $\mathcal{P}_{(j-1) \bmod 2m}$ has a path that connects either $v_4^{(j-1) \bmod 2m}$ and $v_3^{(j-1) \bmod 2m}$ or $v_4^{(j-1) \bmod 2m}$ and $v_5^{(j-1) \bmod 2m}$. Furthermore, it follows from Lemma 1 that, for each $h \in \{j, (i+1) \bmod 2m\}$, every path in $\mathcal{P}_h$ passes through the vertex w_h that corresponds to w in Lemma 1 and the length from w_h to the end-vertex on the path is 19. Those facts guarantee that the existence of a cycle of length less than or equal to 152 in a path in $\mathcal{P}$, concluding the proof. ■

4 The main theorem

Let $C = \{C_1, C_2, \ldots, C_r\}$ be a set of clauses in variables $u_1, u_2, \ldots, u_s$ such that each clause C_i consists of three literals $l_{i,1}, l_{i,2}, l_{i,3}$, where the variables of the three literals are distinct.

The Eulerian graph $G(C)$ corresponding to C is constructed as follows.

For each $k \in \{1, 2, \ldots, s\}$, $v_j^i(k)$ denotes a vertex of the variable-setting component $\Gamma_C(u_k)$ corresponding to the vertex v_j^i of $\Delta(\max\{m(k), 3\})$, where $m(k)$ denotes the number of clauses that includes variable u_k or its negation $\overline{u_k}$. Let $M(k)$ denote $\max\{m(k), 3\}$. For each $k \in \{1, 2, \ldots, s\}$ and $t \in \{1, 2, \ldots, m(k)\}$, $i(k,t)$ and $j(k,t)$ denote positive integers such that the literal $l_{i(k,t),j(k,t)}$ is the t-th occurrence of literals whose variable is v_k.

First, we shall connect satisfaction-testing components and variable-setting components as follows. For each $k \in \{1, 2, \ldots, s\}$ and $t \in \{1, 2, \ldots, m(k)\}$, apply the following procedure:

> If $l_{i(k,t),j(k,t)} = u_k$, then
> identify $v_2^t(k)$ with $v_1^{t+M(k)}$, identify $v_1^t(k)$ with $a(i(k,t), j(k,t))$, and identify $v_2^{t+M(k)}(k)$ with $b(i(k,t), j(k,t))$.
>
> Otherwise,
> identify $v_0^t(k)$ with $v_1^{t+M(k)}$, identify $v_1^t(k)$ with $a(i(k,t), j(k,t))$, and identify $v_0^{t+M(k)}(k)$ with $b(i(k,t), j(k,t))$.

The graph $G_0(C)$ obtained by applying the procedure above has many end-vertices. Notice that if a vertex of $G_0(C)$ is not an end-vertex, then the degree of the vertex is even.

Next, we shall construct the garbage-collecting component for C, Γ_C, and join it to $G_0(C)$ to obtain $G(C)$, as follows. Let x_1, x_2, …, and x_N be all of the end-vertices of $G_0(C)$. Component Γ_C is the tree with N end-vertices $y_1, y_2, \ldots, y_N$, $75N$ vertices of degree 2, and one vertex z of degree N such that the length of the path from z to any end-vertex is 76. Graph $G(C)$ is obtained by identifying y_i of Γ_C with x_i of $G_0(C)$ for each $i \in \{1, 2, \ldots, N\}$.

The positive integer constant μ is defined to be 153. Notice that the length of any cycle passing through the vertex z of Γ_C is greater than or equal to μ. The following is the main theorem of this paper.

Theorem 3 *It is NP-complete to determine whether a graph given has an Eulerian trail that includes no cycles of length less than μ or not.*

Proof. The problem is clearly in the class NP. Furthermore, it is easy to see that the Eulerian graph $G(C)$ can be constructed from an instance C of 3SAT in polynomial time. It therefore suffices to show that C is satisfiable if and only if $G(C)$ has an Eulerian trail that includes no cycles of length less than μ.

First, assume that $G(C)$ has an Eulerian trail T that includes no cycles of length less than μ. For any variable u_k, a truth value is assigned to u_k as follows. By Lemma 2, a path decomposition $\mathcal{P}$ of $\Gamma_C(u_k)$ is obtained from T by deleting all of the edges not contained in $\Gamma_C(u_k)$. If $\mathrm{IF}(\mathcal{P}) = S_+(M(k))$, then assign "true" to u_k, otherwise assign "false". Then, every clause C_i must be satisfied by a literal, otherwise $\Gamma_C(C_i)$ and the six paths in the path decompositions of the variable-setting components that share vertices with $\Gamma_C(C_i)$ compose a cycle. It is impossible for the Eulerian trail T to include such a cycle.

Next, assume that there is a truth assignment to the variables which simultaneously satisfies all the clauses in C. For any variable u_k, if the truth value of u_k is "true", then a path decomposition $\mathcal{P}$ of $\Gamma_C(u_k)$ is made so that $\mathrm{IF}(\mathcal{P}) = S_+(M(k))$ holds. Otherwise, a path decomposition $\mathcal{P}$ of $\Gamma_C(u_k)$ is made so that $\mathrm{IF}(\mathcal{P}) = S_-(M(k))$ holds. Since every clause C_i in C is satisfied by at least one literal, by Lemma 2 and the connection between satisfaction-testing components and variable-setting components, it follows that, for each clause C_i in C, there exists a path P in the path decomposition of a variable-setting component such that P connects an end-vertex of the satisfaction-testing component $\Gamma_C(C_i)$ and one of the garbage-collecting component Γ_C. Furthermore, the following holds. Let P_1 and P_2 be paths in the path decomposition of one variable-setting component. If an end-vertex of P_1 is identical with one of P_2 or an end-vertex of P_1 and one of P_2 are both end-vertices of one satisfaction-testing component $\Gamma_C(C_i)$, or both, then P_1 does not intersect P_2 at any vertex except their end-points.

It is therefore easy to see that the path decompositions of the variable-setting components defined above can be uniquely extended to a path decomposition of $G(C) - z$, the graph obtained from $G(C)$ by deleting z, where z is the unique vertex of the garbage-collecting component Γ_C with degree greater than 2. Furthermore, we obtain an Eulerian trail T from the path decomposition $\mathcal{P} = \{P_1, P_2, \ldots, P_{N/2}\}$ of $G(C) - z$ by plugging z between P_i and P_{i+1} for each $i \in \{1, 2, \ldots, (N/2) - 1\}$, and between $P_{N/2}$ and P_1. Notice that all of the paths in $\mathcal{P}$ are contained in T and, furthermore, any cycle in T includes vertex z of Γ_C. Thus, it follows that T includes no cycles of length less than μ, concluding the proof. ■

5 Concluding remarks

The graph $G(C)$ constructed from an instance C of 3SAT contains only one vertex z whose degree tends to infinity as the size of C tends to infinity. For any vertex v of $G(C)$, the degree of v is at most a constant, except z. We conjecture that there is a reduction from 3SAT to EULERIAN RECURRENT LENGTH so that, for any C, the degree of any vertex in $G(C)$ does not exceed some constant. Furthermore, we conjecture that μ, the lower limit of the length of a cycle in an Eulerian trail, may be vastly decreased. It is an interesting challenge to determine to what extent the instance $(G(C), \mu)$ of EULERIAN RECURRENT LENGTH transformed from C can be simplified. Let $\mathcal{G}$ be the class of simple graphs with maximum degree at most 4.

According to [1], the problem to determine whether the Eulerian recurrent length of a graph in $\mathcal{G}$ is greater than 3 or not can be solved in polynomial time.

Lastly, we remark that this paper is written by correcting and touching in the previous article [4].

References

[1] Tobias Adelgren. Triangle-free Eulerian tours in graphs with maximum degree at most 4. *Discrete Mathematics*, 138:5–14, 1995.

[2] Michael R. Garey and David S. Johnson. *Computers and intractability — A Guide to the Theory of NP-Completeness.* W. H. Freeman and Company, New York, 1979. ISBN 0-7167-1045-5.

[3] Ian Holyer. The NP-completeness of edge-colouring. *SIAM Journal on Computing*, 10(4):718–720, November 1981.

[4] Shuji Jimbo, Yasuaki Osie, and Kosaburo Hashiguchi. It is NP-complete to find the maximum length of the shortest cycles in Eulerian trails. *Technical Reports of IEICE*, COMP2004-53:7–10, December 2004.

[5] Robin J. Wilson. *Introduction to graph theory.* Pearson Education Limited, Harlow, Essex, England, fourth edition, 1996. ISBN 0-582-24993-7.