Theoretical Computer Science 410 (2009) 4543–4553

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Approximating maximum edge 2-coloring in simple graphs via local improvement

Zhi-Zhong Chen*, Ruka Tanahashi

Department of Mathematical Sciences, Tokyo Denki University, Hatoyama, Saitama 350-0394, Japan

ARTICLE INFO

Keywords:
Approximation algorithms
Graph algorithms
Edge coloring
NP-hardness

ABSTRACT

We present a polynomial-time approximation algorithm for legally coloring as many edges of a given simple graph as possible using two colors. It achieves an approximation ratio of $\frac{24}{29} \approx 0.828$.

1. Introduction

Given a graph G and a natural number t, the *maximum edge t-coloring problem* (called MAX EDGE t-COLORING for short) is to find a maximum set F of edges in G such that F can be partitioned into at most t matchings of G. Motivated by call admittance issues in satellite based telecommunication networks, Feige et al. [2] introduced the problem and proved its APX-hardness. They also observed that MAX EDGE t-COLORING is obviously a special case of the well-known maximum coverage problem (see [4]). Since the maximum coverage problem can be approximated by a greedy algorithm within a ratio of $1-(1-\frac{1}{t})^t$ [4], so can MAX EDGE t-COLORING. In particular, the greedy algorithm achieves an approximation ratio of $\frac{3}{4}$ for MAX EDGE 2-COLORING which is the special case of MAX EDGE t-COLORING where the input number t is fixed to 2. Feige et al. [2] has improved the trivial ratio $\frac{3}{4} = 0.75$ to $\frac{10}{13} \approx 0.769$ by an LP approach.

The APX-hardness proof for MAX EDGE t-COLORING given by Feige et al. [2] indeed shows that the problem remains APX-hard even if we restrict the input graph to a simple graph and fix the input integer t to 2. We call this restriction (special case) of the problem MAX SIMPLE EDGE 2-COLORING. Feige et al. [2] also pointed out that for MAX SIMPLE EDGE 2-COLORING, an approximation ratio of $\frac{4}{5}$ can be achieved by the following *simple algorithm*: Given a simple graph G, first compute a maximum subgraph H of G such that the degree of each vertex in H is at most 2 and there is no 3-cycle in H, and then remove one *arbitrary* edge from each odd cycle of H.

In [1], the authors have improved the ratio to $\frac{468}{575} \approx 0.814$. Essentially, the algorithm in [1] differs from the simple algorithm only in the handling of 5-cycles where instead of removing one arbitrary edge from each 5-cycle of H, we remove a *random* edge from each 5-cycle of H. The intuition behind the algorithm is as follows: If we delete a random edge from each 5-cycle of H, then for each edge $\{u, v\}$ in the optimal solution such that u and v belong to different 5-cycles, both u and v become of degree 1 in H (after handling the 5-cycles) with a probability of $\frac{4}{25}$ and hence can be added into H without losing the edge 2-colorability of H.

In this paper, we further improve the ratio to $\frac{24}{29} \approx 0.828$. The basic idea behind our algorithm is as follows: Instead of removing a random edge from each 5-cycle of H and removing an arbitrary edge from each other odd cycle of H, we remove one edge from each odd cycle of H with more care in the hope that after the removal, a lot of edges $\{u, v\}$ (in the optimal solution) with u and v belonging to different odd cycles of H can be added to H. The new algorithm is even more difficult to analyze than the algorithm in [1].

* Corresponding author.
E-mail address: zzchen@mail.dendai.ac.jp (Z.-Z. Chen).

0304-3975/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2009.07.008

Kosowski et al. [8] also considered Max Simple Edge 2-Coloring. They presented an approximation algorithm that achieves a ratio of $\frac{28\Delta-12}{35\Delta-21}$, where Δ is the maximum degree of a vertex in the input simple graph. This ratio can be arbitrarily close to the trivial ratio $\frac{4}{5}$ because Δ can be very large. In particular, this ratio is smaller than $\frac{24}{29}$ when $\Delta \geq 6$.

Kosowski et al. [8] showed that approximation algorithms for Max Simple Edge 2-Coloring can be used to obtain approximation algorithms for certain packing problems and fault-tolerant guarding problems. Combining their reductions and our improved approximation algorithm for Max Simple Edge 2-Coloring, we can obtain improved approximation algorithms for their packing problems and fault-tolerant guarding problems immediately.

The remainder of this paper is organized as follows. Section 2 gives some basic definitions and notations in graph theory that will be used throughout this paper. Section 3 presents our algorithm and its analyses. The analysis in Section 3.1 is simpler but proves a smaller ratio, while the analysis in Section 3.2 is more complicated but proves a larger ratio. Section 4 describes an application of our algorithm.

2. Basic definitions

Throughout the remainder of this paper, a graph means a simple undirected graph (i.e. it has neither parallel edges nor self-loops).

Let G be a graph. We denote the vertex set of G by $V(G)$, and denote the edge set of G by $E(G)$. The *neighborhood* of a vertex v in G, denoted by $N_G(v)$, is the set of all vertices adjacent to v in G. The *degree* of a vertex v in G, denoted by $d_G(v)$, is $|N_G(v)|$. A vertex v of G with $d_G(v) = 0$ is called an *isolated vertex*. For a subset U of $V(G)$, let $G[U]$ denote the graph (U, E_U) where E_U consists of all edges $\{u, v\}$ of G with $u \in U$ and $v \in U$. We call $G[U]$ the *subgraph of G induced by U*. For a subset U of $V(G)$, we use $G - U$ to denote $G[V(G) - U]$.

A *cycle* in G is a connected subgraph of G in which each vertex is of degree 2. A *Hamiltonian cycle* of G is a cycle C of G with $V(C) = V(G)$. A *path* in G is a connected subgraph of G in which exactly two vertices are of degree 1 and the others are of degree 2. Each vertex of degree 1 in a path P is called an *endpoint* of P, while each vertex of degree 2 in P is called an *inner vertex* of P. An edge $\{u, v\}$ of a path P is called an *inner edge* of P if both u and v are inner vertices of P. The *length* of a cycle or path C is the number of edges in C. A cycle of odd (respectively, even) length is called an *odd* (respectively, *even*) cycle. The *distance* between two vertices u and v in G, denoted by $dist_G(u, v)$, is the length of the shortest paths between u and v in G.

A *k-cycle* is a cycle of length k. Similarly, a *k^+-cycle* is a cycle of length at least k. A *path component* (respectively, *cycle component*) of G is a connected component of G that is a path (respectively, cycle). Note that an isolated vertex of G is not a path component of G. A *path-cycle cover* of G is a subgraph H of G such that $V(H) = V(G)$ and $d_H(v) \leq 2$ for every $v \in V(H)$. Note that each connected component of a path-cycle cover of G is a single vertex, path, or cycle. A *cycle cover* of G is a path-cycle cover of G in which each connected component is a cycle. A path-cycle cover C of G is *triangle-free* if C does not contain a 3-cycle. A path-cycle cover C of G is *maximum* if the number of edges in C is maximized over all path-cycle covers of G.

G is *edge-2-colorable* if each connected component of G is an isolated vertex, a path, or an even cycle. Note that Max Simple Edge 2-Coloring is the problem of finding a maximum edge-2-colorable subgraph in a given graph.

3. The algorithm

Throughout this section, fix a graph G and a maximum edge-2-colorable subgraph *Opt* of G. For convenience, for each path-cycle cover K of G, we define two numbers as follows:

- $n_0(K)$ is the number of isolated vertices in K.
- $p(K)$ is the number of path components in K.

Like the simple algorithm described in Section 1, our algorithm starts by performing the following step:

1. Compute a maximum triangle-free path-cycle cover H of G.

Since $|E(H)| \geq |E(Opt)|$, it suffices to modify H into an edge-2-colorable subgraph of G without significantly decreasing the number of edges in H. The simple algorithm achieves an approximation ratio of $\frac{4}{5}$ because it simply removes an arbitrary edge from each odd cycle in H. In order to improve this ratio, we have to treat 5-cycles (and other short odd cycles) in H more carefully. In more details, when removing edges from odd cycles in H, we also want to add some edges of $E(G) - E(H)$ to H. For this purpose, when we decide which edge should be deleted from an odd cycle C in H, we cannot concentrate solely on C; rather, we have to explore the neighborhood of C in the input graph G. So, we will define eleven types of operations on H each of which breaks one or two odd cycles in H (by edge removal) and may also add to H one or two edges in the neighborhood of the cycle(s) in G without creating new odd cycles or vertices of degree larger than 2 in H. In a nutshell, all the operations will aim to decrease the number of odd cycles in H at the possible risk of decreasing the number of edges in H by 1. To tighten the analysis of the approximation ratio achieved by our algorithm, we set up a charging scheme that charges the net loss of edges from H (due to the operations) to some edges still remaining in H. Whenever we do this, we will always maintain the following invariants:

I1. Every edge of H is charged a real number smaller than or equal to $\frac{1}{9}$.
I2. The total charge on the edges of H equals the total number of operations performed on H that decrease the number of edges in H.

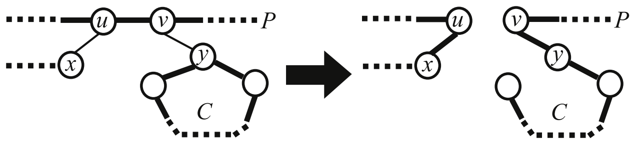

Fig. 1. A Type-2 operation, where bold edges are in H.

I3. No cycle component of H contains a charged edge.
I4. If a path component P of H contains a charged edge, then the length of P is at least 6.

Initially, every edge of H is charged nothing. However, as our algorithm modifies H by performing the operations, some edges of H will be charged.

The rest of our algorithm can now be sketched as follows: It consists of only two stages. In the first stage, the algorithm performs the operations (to be defined below) on H until none is applicable. In the second stage, it simply deletes one arbitrary edge from each odd cycle still remaining in H.

Basically, the operations will be designed based on two intuitive ideas. The first idea is that if an odd-cycle component C of H is long enough, then we can simply transform C into a path by deleting one of its edges; the loss of this edge can be charged evenly to the leftover edges of C. The second idea is the following: If $\mathcal{K}$ is a collection of two or three connected components of H such that

- at least one component in $\mathcal{K}$ is an odd cycle and
- we can transform $\mathcal{K}$ into an edge-2-colorable subgraph $\mathcal{K}'$ of G such that $V(\mathcal{K}) = V(\mathcal{K}')$, $|E(\mathcal{K}')| = |E(\mathcal{K})|$ in case $|E(\mathcal{K})|$ is not large enough, and $|E(\mathcal{K}')| \geq |E(\mathcal{K})| - 1$ in case $|E(\mathcal{K})|$ is large enough,

then we can transform $\mathcal{K}$ into $\mathcal{K}'$; the loss of one edge, if any, can be charged evenly to some edges of $\mathcal{K}'$. We believe that by exploring larger collections of components of H, our algorithm can achieve a better ratio. However, by doing so, our algorithm will be more complicated and slower.

We next proceed to the definitions of the operations. We first define those operations on H that decrease the number of odd cycles in H but do not decrease the number of edges in H. In order to do this, the following three concepts are necessary:

A quintuple (x, y, P, u, v) is a 5-*opener* for an odd cycle C of H if the following hold:

- $d_H(x) \leq 1$ and $y \in V(C)$.
- P is a path component of H, both u and v are inner vertices of P, and x is not a vertex of P.
- Both $\{u, x\}$ and $\{v, y\}$ are contained in $E(G) - E(H)$.

A sextuple (x, y, Q, P, u, v) is a 6-*opener* for an odd cycle C of H if the following hold:

- $x \in V(C)$ and $y \in V(C)$. Moreover, if $x = y$, then Q is a cycle cover of $G[V(C) - \{x\}]$ in which each connected component is an even cycle; otherwise, Q is a path-cycle cover of $G[V(C)]$ in which one connected component is a path from x to y and each other connected component is an even cycle.
- P is a path component of H and both u and v are inner vertices of P.
- Both $\{u, x\}$ and $\{v, y\}$ are contained in $E(G) - E(H)$.

An operation (to be performed) on H is *robust* if the following holds:

- If G has no edge $\{u, v\}$ before the operation such that u is an isolated vertex in H and either v is an isolated vertex in H or v appears in a cycle component of H, then neither does it after the operation.

Based on the above concepts, we are now ready to define six robust operations on H that decrease the number of odd cycles in H but do not decrease the number of edges in H.

Type 1: Suppose that $\{u, v\}$ is an edge in $E(G) - E(H)$ such that $d_H(u) \leq 1$ and v is a vertex of some cycle C of H. Then, a *Type*-1 *operation* on H using $\{u, v\}$ modifies H by deleting one (arbitrary) edge of C incident to v and adding edge $\{u, v\}$. Obviously, this operation is robust and does not change $|E(H)|$.

(*Comment:* If $d_H(u) = 0$ before a Type-1 operation, then $n_0(H)$ decreases by 1 and $p(H)$ increases by 1 after the operation. Similarly, if $d_H(u) = 1$ before a Type-1 operation, then neither $n_0(H)$ nor $p(H)$ changes after the operation.)

Type 2: Suppose that some odd cycle C of H has a 5-opener (x, y, P, u, v) with $\{u, v\} \in E(H)$ (see Fig. 1). Then, a *Type*-2 *operation* on H using (x, y, P, u, v) modifies H by deleting edge $\{u, v\}$, deleting one (arbitrary) edge of C incident to y, and adding edges $\{u, x\}$ and $\{v, y\}$. Obviously, this operation is robust and does not change the number of edges in H. However, edge $\{u, v\}$ may have been charged before this operation. If that is the case, we move its charge to $\{u, x\}$. Moreover, if the path component Q of H containing edge $\{u, x\}$ after this operation is of length at most 5, then we move the charges on the edges of Q to edge $\{v, y\}$ and the edges of C still remaining in H.

(*Comment:* A Type-2 operation on H maintains Invariants I1 through I4. Moreover, if $d_H(x) = 0$ before a Type-2 operation, then $n_0(H)$ decreases by 1 and $p(H)$ increases by 1 after the operation. Similarly, if $d_H(x) = 1$ before a Type-2 operation, then neither $n_0(H)$ nor $p(H)$ changes after the operation.)

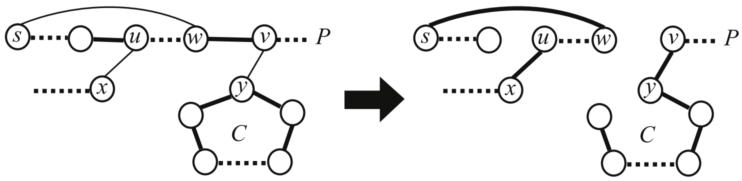

Fig. 2. A Type-3 operation, where bold edges are in H.

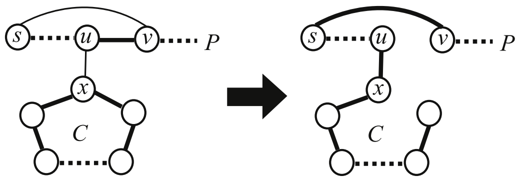

Fig. 3. A Type-4 operation, where bold edges are in H.

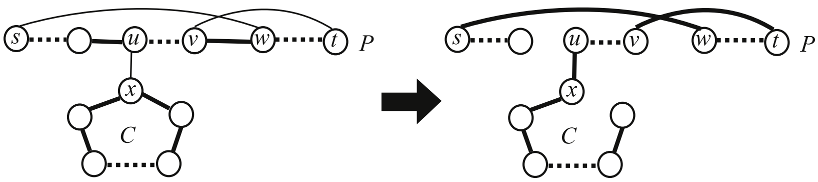

Fig. 4. A Type-5 operation, where bold edges are in H.

Type 3: Suppose that some odd cycle C of H has a 5-opener (x, y, P, u, v) such that $\{u, v\} \notin E(H)$ and $E(G) - E(H)$ contains the edge $\{w, s\}$, where w is the neighbor of v in the subpath of P between u and v and s is the endpoint of P with $dist_P(s, u) < dist_P(s, v)$ (see Fig. 2). Then, a *Type*-3 *operation* on H using (x, y, P, u, v) modifies H by deleting edge $\{v, w\}$, deleting one (arbitrary) edge e_u of P incident to u, deleting one (arbitrary) edge of C incident to y, and adding edges $\{u, x\}$, $\{v, y\}$, and $\{s, w\}$. Obviously, this operation is robust and does not change the number of edges in H. Note that $\{v, w\}$ or e_u may have been charged before this operation. If that is the case, we move their charges to edges $\{u, x\}$ and $\{s, w\}$, respectively. Moreover, if the path component Q of H containing edge $\{u, x\}$ after this operation is of length at most 5, then we move the charges on the edges of Q to edge $\{v, y\}$ and the edges of C still remaining in H.

(*Comment:* A Type-3 operation on H maintains Invariants I1 through I4. Moreover, if $d_H(x) = 0$ before a Type-3 operation, then $n_0(H)$ decreases by 1 and $p(H)$ increases by 1 after the operation. Similarly, if $d_H(x) = 1$ before a Type-3 operation, then neither $n_0(H)$ nor $p(H)$ changes after the operation.)

Type 4: Suppose that there is a quadruple (x, P, u, v) satisfying the following conditions (see Fig. 3):

- x is a vertex of a cycle component C of H.
- P is a path component of H and $\{u, v\}$ is an inner edge of P.
- $E(G) - E(H)$ contains both $\{u, x\}$ and $\{s, v\}$, where s is the endpoint of P with $dist_P(s, u) < dist_P(s, v)$.

Then, a *Type*-4 *operation* on H using (x, P, u, v) modifies H by deleting edge $\{u, v\}$, deleting one (arbitrary) edge of C incident to x, and adding edges $\{u, x\}$ and $\{s, v\}$. Obviously, this operation is robust and does not change the number of edges in H. However, $\{u, v\}$ may have been charged before this operation. If that is the case, we move its charge to $\{u, x\}$.

(*Comment:* A Type-4 operation on H maintains Invariants I1 through I4, and changes neither $n_0(H)$ nor $p(H)$.)

Type 5: Suppose that there is a quintuple (x, P, u, v, w) satisfying the following conditions (see Fig. 4):

- x is a vertex of a cycle component C of H.
- P is a path component of H, u is an inner vertex of P, $\{v, w\}$ is an inner edge of P, $dist_P(u, v) < dist_P(u, w)$, and $u \neq v$.
- $E(G) - E(H)$ contains $\{u, x\}$, $\{s, w\}$, and $\{t, v\}$, where s is the endpoint of P with $dist_P(s, u) < dist_P(s, v)$ and t is the other endpoint of P.

Then, a *Type*-5 *operation* on H using (x, P, u, v, w) modifies H by deleting edge $\{v, w\}$, deleting one (arbitrary) edge e_u of P incident to u, deleting one (arbitrary) edge of C incident to x, and adding edges $\{s, w\}$, $\{t, v\}$, and $\{u, x\}$. Obviously, this operation is robust and does not change the number of edges in H. However, edges $\{v, w\}$ and e_u may have been charged before this operation. If that is the case, we move their charges to $\{u, x\}$ and $\{s, w\}$, respectively.

(*Comment:* A Type-5 operation on H maintains Invariants I1 through I4, and changes neither $n_0(H)$ nor $p(H)$.)

Type 6: Suppose that some odd cycle C of H with length at most 9 has a 6-opener (x, y, Q, P, u, v) such that $\{u, v\} \in E(H)$ (see Fig. 5). Then, a *Type*-6 *operation* on H using (x, y, Q, P, u, v) modifies H by deleting edge $\{u, v\}$, deleting all

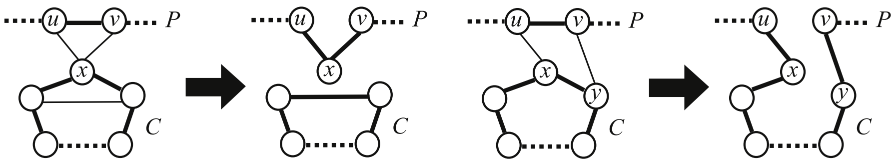

Fig. 5. A Type-6 operation, where bold edges are in H.

edges of C, adding edges $\{u, x\}$ and $\{v, y\}$, and adding all edges of Q. Obviously, this operation does not change the number of edges in H, and is robust because (1) it does not create a new isolated vertex in H and (2) if it creates one or more new cycles in H then $V(C') \subseteq V(C)$ for each new cycle C'. However, $\{u, v\}$ may have been charged before this operation. If that is the case, we move its charge to $\{u, x\}$.

(*Comment:* A Type-6 operation on H maintains Invariants I1 through I4, and changes neither $n_0(H)$ nor $p(H)$.)

Using the above operations, our algorithm then proceeds to modifying H by performing the following step:

2. Repeat performing a Type-i operation on H with $1 \leq i \leq 6$, until none is applicable.

Obviously, H remains a triangle-free path-cycle cover of G. Moreover, the following fact holds:

Fact 3.1. *After Step 2, G has no edge $\{u, v\}$ such that u is an isolated vertex in H and either v is an isolated vertex in H or v appears in a cycle component of H.*

Unfortunately, H may still have odd cycles after Step 2. So, we need to perform new types of operations on H that always decrease the number of odd cycles in H but may also decrease the number of edges in H. Before defining the new operations on H, we define two concepts as follows. Two cycles C_1 and C_2 of H are *pairable* if at least one of them is odd and their total length is at least 10. A quintuple (x, y, P, u, v) is an *opener* for two pairable cycles C_1 and C_2 of H if the following hold:

- x is a vertex of C_1 and y is a vertex of C_2.
- P is either a path component of H or a 4-cycle of H different from C_1 and C_2.
- u and v are distinct inner vertices of P.
- Both $\{u, x\}$ and $\{v, y\}$ are in $E(G) - E(H)$.

Now, we are ready to define the new types of robust operations on H as follows:

Type 7: Suppose that C is an odd cycle of H with length at least 11. Then, a *Type*-7 *operation* on H using C modifies H by deleting one (arbitrary) edge from C. Clearly, the net loss in the number of edges in H is 1. We charge this loss evenly to the edges of C still remaining in H. In more details, if C was a k-cycle before the operation, then a charge of $\frac{1}{k-1}$ is charged to each edge of C still remaining in H after the operation. Since $k \geq 11$, the charge assigned to one edge here is at most $\frac{1}{10}$. Obviously, this operation is robust.

(*Comment:* A Type-7 operation on H maintains Invariants I1 through I4, does not change $n_0(H)$, and increases $p(H)$ by 1.)

Type 8: Suppose that C_1 and C_2 are two pairable cycles of H such that there is an edge $\{u, v\} \in E(G)$ with $u \in V(C_1)$ and $v \in V(C_2)$. Then, a *Type*-8 *operation* on H using $\{u, v\}$ modifies H by deleting one (arbitrary) edge of C_1 incident to u, deleting one (arbitrary) edge of C_2 incident to v, and adding edge $\{u, v\}$. Note that this operation decreases the number of edges in H by 1. So, the net loss in the number of edges in H is 1. We charge this loss evenly to edge $\{u, v\}$ and the edges of C_1 and C_2 still remaining in H. In more details, if C_1 was a k-cycle and C_2 was an ℓ-cycle in H before the operation, then a charge of $\frac{1}{k+\ell-1}$ is assigned to $\{u, v\}$ and each edge of C_1 and C_2 still remaining in H after the operation. Since $k \geq 5$ and $\ell \geq 5$, the charge assigned to one edge here is at most $\frac{1}{9}$. Obviously, this operation is robust.

(*Comment:* A Type-8 operation on H maintains Invariants I1 through I4, does not change $n_0(H)$, and increases $p(H)$ by 1.)

Type 9: Suppose that two odd cycles C_1 and C_2 of H have an opener (x, y, P, u, v) with $\{u, v\} \in E(H)$ (see Fig. 6). Then, a *Type*-9 *operation* on H using (x, y, P, u, v) modifies H by deleting edge $\{u, v\}$, deleting one (arbitrary) edge of C_1 incident to x, deleting one (arbitrary) edge of C_2 incident to y, and adding edges $\{u, x\}$ and $\{v, y\}$. Note that edge $\{u, v\}$ may have been charged before this operation. If that is the case, we move its charge to edge $\{u, x\}$. Moreover, the operation decreases the number of edges in H by 1. So, the net loss in the number of edges in H is 1. We charge this loss evenly to edge $\{v, y\}$ and the edges of C_1 and C_2 still remaining in H. Obviously, the charge assigned to one edge here is at most $\frac{1}{9}$. It is also clear that this operation is robust.

(*Comment:* A Type-9 operation on H maintains Invariants I1 through I4, does not change $n_0(H)$, and increases $p(H)$ by 1.)

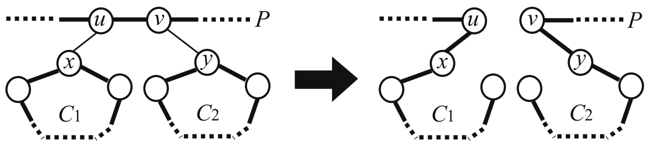

Fig. 6. A Type-9 operation, where bold edges are in H.

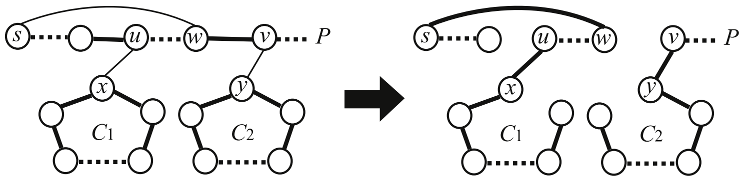

Fig. 7. A Type-10 operation, where bold edges are in H.

Type 10: Suppose that two odd cycles C_1 and C_2 of H have an opener (x, y, P, u, v) such that $\{u, v\} \notin E(H)$ and $E(G) - E(H)$ contains the edge $\{w, s\}$, where w is the neighbor of v in the subpath of P between u and v and s is the endpoint of P with $dist_P(s, u) < dist_P(s, v)$ (see Fig. 7). Then, a *Type*-10 *operation* on H using (x, y, P, u, v) modifies H by deleting edge $\{v, w\}$, deleting one (arbitrary) edge e_u of P incident to u, deleting one (arbitrary) edge of C_1 incident to x, deleting one (arbitrary) edge of C_2 incident to y, and adding edges $\{u, x\}$, $\{v, y\}$, and $\{s, w\}$. Note that $\{v, w\}$ or e_u may have been charged before this operation. If that is the case, we move their charges to edges $\{u, x\}$ and $\{v, y\}$, respectively. Moreover, the operation decreases the number of edges in H by 1. So, the net loss in the number of edges in H is 1. We charge this loss evenly to edge $\{s, w\}$ and the edges of C_1 and C_2 still remaining in H. Obviously, the charge assigned to one edge here is at most $\frac{1}{9}$. It is also clear that this operation is robust.

(*Comment:* A Type-10 operation on H maintains Invariants I1 through I4, does not change $n_0(H)$, and increases $p(H)$ by 1.)

After Step 2, no matter how many times we perform Type-i operations on H with $1 \leq i \leq 10$, G cannot have an edge $\{u, v\}$ such that u is an isolated vertex in H and either v is an isolated vertex in H or v appears in a cycle component of H. This follows from Fact 3.1 and the fact that every Type-i operation on H with $1 \leq i \leq 10$ is robust. However, after performing a Type-i operation on H with $7 \leq i \leq 10$, the following new type of robust operations on H may be applicable:

Type 11: Suppose that $\{u, v\}$ is an edge in $E(G) - E(H)$ such that $d_H(u) = 1$, $d_H(v) \leq 1$, and no connected component of H contains both u and v. Then, a *Type*-11 *operation* on H using $\{u, v\}$ modifies H by adding edge $\{u, v\}$. Obviously, this operation is robust and increases the number of edges in H by 1.

(*Comment:* If $d_H(v) = 0$ before a Type-11 operation, then $p(H)$ does not change and $n_0(H)$ decreases by 1 after the operation. Similarly, if $d_H(v) = 1$ before a Type-11 operation, then $n_0(H)$ does not change and $p(H)$ decreases by 1 after the operation.)

Using the above operations, our algorithm then proceeds to modifying H by performing the following steps:

3. Repeat using a Type-i operation to modify H with $1 \leq i \leq 11$, until none is applicable.
4. For each odd cycle C of H, remove one (arbitrary) edge from C. (*Comment:* Each odd cycle modified in this step is a 5-, 7-, or 9-cycle.)
5. Output H.

For $1 \leq i \leq 4$, let H_i be the triangle-free path-cycle cover H of G immediately after Step i of our algorithm. In order to analyze the approximation ratio achieved by our algorithm, we need to define several notations as follows:

- Let n, m, n_{is}, and n_{pc} be the numbers of vertices, edges, isolated vertices, and path components in H_2, respectively. (*Comment:* $m \geq |E(Opt)|$.)
- Let m_- be the number of Type-i operations with $7 \leq i \leq 10$ performed in Step 3.
- Let $m_{+,-1}$ be the number of Type-11 operations performed in Step 3 that decrease the number of isolated vertices in H by 1.
- Let $m_{+,0}$ be the number of Type-11 operations performed Step 3 that do not change the number of isolated vertices in H.
- Let $n_{0,-1}$ be the number of Type-i operations with $1 \leq i \leq 3$ performed in Step 3 that decrease the number of isolated vertices in H by 1.
- For each $i \in \{5, 7, 9\}$, let c_i be the number of i-cycles in H_3.
- Let m_c and m_{uc} be the numbers of charged edges and uncharged edges in H_3, respectively.

Lemma 3.2. *The following statements hold:*

1. $m_- \leq \frac{1}{10}(m + m_{+,0} + m_{+,-1} - m_{uc})$.

2. $|E(H_4)| = m - m_- + m_{+,0} + m_{+,-1} - c_5 - c_7 - c_9$.
3. $|E(H_4)| \geq \frac{9}{10}(m + m_{+,0} + m_{+,-1}) - (\frac{1}{2}c_5 + \frac{3}{10}c_7 + \frac{1}{10}c_9)$.

Proof. By the algorithm, $|E(H_3)| = m - m_- + m_{+,0} + m_{+,-1}$. On the other hand, $|E(H_3)| = m_c + m_{uc}$ by definition. So, $m_c = m - m_- + m_{+,0} + m_{+,-1} - m_{uc}$. We also have $m_- \leq \frac{1}{9}m_c$ by Invariant I2. Thus, $m_- \leq \frac{1}{10}(m + m_{+,0} + m_{+,-1} - m_{uc})$. This establishes Statement 1.

By Step 3, $|E(H_4)| = |E(H_3)| - c_5 - c_7 - c_9$. So, by the first equality in the last paragraph, $|E(H_4)| = m - m_- + m_{+,0} + m_{+,-1} - c_5 - c_7 - c_9$. This establishes Statement 2.

By Statements 1 and 2, $|E(H_4)| \geq \frac{9}{10}(m + m_{+,0} + m_{+,-1}) + \frac{1}{10}m_{uc} - c_5 - c_7 - c_9$. We also have $m_{uc} \geq 5c_5 + 7c_7 + 9c_9$, because each edge in a cycle component of H_3 is uncharged according to Invariant I3. Combining these two inequalities, we have Statement 3. □

Lemma 3.3. *The following statements hold:*

1. $n - n_0(H_3) - 2p(H_3) = m - n_{pc} - 2m_- + 2m_{+,0} + m_{+,-1} - n_{0,-1}$.
2. $p(H_3) = n_{pc} + m_- - m_{+,0} + n_{0,-1}$.

Proof. Immediately before Step 3, $n - n_0(H) - 2p(H) = m - n_{pc}$ because $p(H) = n_{pc}$ and the number of vertices on a path is 1 plus the number of edges on the path. Now, to prove the lemma, it suffices to see how the values of $n - n_0(H) - 2p(H)$ and $p(H)$ change when performing an operation in Step 3. The comment on the definition of each type of operations helps. □

3.1. The first analysis

We start by giving several definitions:

- For $i \in \{0, 1\}$, let T_i be the set of all vertices v in H_3 with $d_{H_3}(v) = i$.
- Let T_2 be the set of all vertices v in H_3 such that v appears in an odd cycle of H_3.
- Let $T = T_0 \cup T_1 \cup T_2$.
- For $i \in \{0, 1, 2\}$, let $\overline{T_i}$ be the set of vertices $u \in V(G) - T$ such that the number of edges $\{u, v\} \in E(Opt)$ with $v \in T$ is exactly i. (*Comment:* $V(G) - T = \overline{T_0} \cup \overline{T_1} \cup \overline{T_2}$.)
- Let E^T_{opt} be the set of all edges $\{u, v\}$ in Opt such that both u and v are vertices of T.
- Let $\mathcal{C}_{-2}$ be the set of all odd cycles C in H_3 such that Opt contains at most $|V(C)| - 2$ edges $\{u, v\}$ with $\{u, v\} \subseteq V(C)$.

The next lemma gives an upper bound on $|E^T_{opt}|$.

Lemma 3.4. $|E^T_{opt}| \leq p(H_3) + 4c_5 + 6c_7 + 8c_9 - |\mathcal{C}_{-2}| \leq n_{pc} + m_- - m_{+,0} + n_{0,-1} + 4c_5 + 6c_7 + 8c_9 - |\mathcal{C}_{-2}|$.

Proof. First, we claim that each vertex $u \in T_0$ is an isolated vertex in $G[T]$. To see this, consider an arbitrary $u \in T_0$. Because of Fact 3.1 and the fact that all Type-i operations with $1 \leq i \leq 11$ are robust, there is no vertex $v \in T_0 \cup T_2$ with $\{u, v\} \in E(G)$. Moreover, since no Type-11 operation can be applied to H_3, there is no vertex $v \in T_1$ with $\{u, v\} \in E(G)$. So, the claim holds.

Next, we claim that there is no edge $\{u, v\} \in E(G)$ such that $u \in T_1$ and $v \in T_2$. This follows from the fact that no Type-1 operation can be applied to H_3.

By the above two claims, each edge in E^T_{opt} is either in $G[T_1]$ or in $G[T_2]$. Since no Type-11 operation can be applied to H_3, there is no edge $\{u, v\} \in E(G)$ with $\{u, v\} \subseteq T_1$ such that u and v belong to different connected components of H_3. So, there are at most $p(H_3)$ edges in $G[T_1]$. Consequently, to show the first inequality in the lemma, it remains to show that E^T_{opt} contains at most $4c_5 + 6c_7 + 8c_9 - |\mathcal{C}_{-2}|$ edges $\{u, v\}$ with $\{u, v\} \subseteq T_2$.

Suppose that $\{u, v\}$ is an edge in E^T_{opt} with $\{u, v\} \subseteq T_2$. Since no Type-8 operation can be applied to H_3, u and v belong to the same cycle component of $H_3[T_2]$. On the other hand, since each cycle C in $H_3[T_2]$ is an odd cycle, E^T_{opt} can contain at most $|E(C)| - 1$ edges $\{u, v\}$ with $\{u, v\} \subseteq V(C)$. In particular, for each cycle $C \in \mathcal{C}_{-2}$, E^T_{opt} can contain at most $|E(C)| - 2$ edges $\{u, v\}$ with $\{u, v\} \subseteq V(C)$. Hence, E^T_{opt} contains at most $4c_5 + 6c_7 + 8c_9 - |\mathcal{C}_{-2}|$ edges $\{u, v\}$ with $\{u, v\} \subseteq T_2$. This completes the proof of the first inequality in the lemma. The second inequality follows from the first and Statement 2 in Lemma 3.3 immediately. □

The next lemma gives a lower bound on $|E^T_{opt}|$.

Lemma 3.5. $|E^T_{opt}| \geq |E(Opt)| - 2m + 2n_{pc} + 4m_- - 4m_{+,0} - 2m_{+,-1} + 2n_{0,-1} + 10c_5 + 14c_7 + 18c_9 + |\overline{T_0}| + \frac{1}{2}|\overline{T_1}|$.

Proof. The idea behind the proof is to obtain an upper bound on $|E(Opt) - E^T_{opt}|$. A trivial upper bound is $2(n - |T|)$, because each edge in $E(Opt) - E^T_{opt}$ must be incident to a vertex in $V(G) - T$ and each vertex in $V(G) - T$ can be adjacent to at most two edges in Opt. This bound is not good enough because each edge $\{u, v\} \in E(Opt)$ with $\{u, v\} \subseteq V(G) - T$ is counted twice.

To get a better bound, we set up a savings account for each vertex in $V(G) - T$. Initially, we deposit two credits to the account of each vertex. The total credits amount to $2(n - |T|)$ (namely, the trivial upper bound). Next, for each edge $\{u, v\} \in E(Opt)$ with $\{u, v\} \subseteq V(G) - T$, we pay a half credit from the account of u and another half credit from the account of v. After this, for each edge $\{u, v\} \in E(Opt)$ with $u \in V(G)-T$ and $v \in T$, we pay one credit from the account of u. Obviously, we paid a total of $|E(Opt) - E_{opt}^T|$ credits. We want to estimate the number of credits that are still left in the accounts of the vertices in $V(G) - T$. First, for each vertex $u \in \overline{T_0}$, we paid at most one credit, because u is incident to at most two edges in $E(Opt) - E_{opt}^T$ and each of them has both of its endpoints in $V(G) - T$. So, the total number of credits still left in the accounts of the vertices in $\overline{T_0}$ is at least $|\overline{T_0}|$. Second, for each vertex $u \in \overline{T_1}$, we paid at most one and a half credits, because u is incident to at most two edges in $E(Opt) - E_{opt}^T$ and one of them has an endpoint in T. Thus, the total number of credits still left in the accounts of the vertices in $\overline{T_1}$ is at least $\frac{1}{2}|\overline{T_1}|$. In summary, we have shown that $|E(Opt) - E_{opt}^T| \le 2(n - |T|) - |\overline{T_0}| - \frac{1}{2}|\overline{T_1}|$.

Obviously, $|T| = n_0(H_3) + 2p(H_3) + 5c_5 + 7c_7 + 9c_9$. So, by Statement 1 in Lemma 3.3, $|T| = n - (m - n_{pc} - 2m_- + 2m_{+,0} + m_{+,-1} - n_{0,-1}) + 5c_5 + 7c_7 + 9c_9$. In other words, $n - |T| = m - n_{pc} - 2m_- + 2m_{+,0} + m_{+,-1} - n_{0,-1} - 5c_5 - 7c_7 - 9c_9$. Hence, by the last inequality in the last paragraph, $|E(Opt) - E_{opt}^T| \le 2m - 2n_{pc} - 4m_- + 4m_{+,0} + 2m_{+,-1} - 2n_{0,-1} - 10c_5 - 14c_7 - 18c_9 - |\overline{T_0}| - \frac{1}{2}|\overline{T_1}|$. So, the lemma holds. □

The following lemma shows that the approximation ratio achieved by our algorithm is at least $\frac{37}{45}$.

Lemma 3.6. $|E(H_4)| \ge \frac{37}{45}|E(Opt)|$.

Proof. Combining Lemmas 3.4 and 3.5, we have

$$|E(Opt)| \le 2m - n_{pc} - 3m_- + 3m_{+,0} + 2m_{+,-1} - n_{0,-1} - 6c_5 - 8c_7 - 10c_9.$$

So, by Statement 2 in Lemma 3.2,

$$3|E(H_4)| - |E(Opt)| \ge m + m_{+,-1} + n_{pc} + n_{0,-1} + 3c_5 + 5c_7 + 7c_9.$$

Thus, $3c_5 + 5c_7 + 7c_9 \le 3|E(H_4)| - |E(Opt)| - m$. Hence, $\frac{1}{2}c_5 + \frac{3}{10}c_7 + \frac{1}{10}c_9 \le \frac{1}{6}(3c_5 + 5c_7 + 7c_9) \le \frac{1}{2}|E(H_4)| - \frac{1}{6}|E(Opt)| - \frac{1}{6}m$. Therefore, by Statement 3 in Lemma 3.2, we have $|E(H_4)| \ge \frac{9}{10}m - (\frac{1}{2}|E(H_4)| - \frac{1}{6}|E(Opt)| - \frac{1}{6}m)$. Rearranging this inequality and using the fact that $m \ge |E(Opt)|$, we finally obtain $|E(H_4)| \ge \frac{37}{45}|E(Opt)|$. □

In the next subsection, we will refine the analysis to obtain a better ratio.

3.2. The second analysis

To obtain a better approximation ratio, our idea is to show that $|\mathcal{C}_{-2}| + |\overline{T_0}| + \frac{1}{2}|\overline{T_1}|$ is large.

For convenience, we say that a path component of H_3 is *short* if its length is at most 5. The point is that each short path component of H_3 does not contain a charged edge. This follows from Invariant I4.

Let C be an odd cycle component of H_3. We call an edge $\{u, v\} \in E(Opt)$ an *antenna* of C if exactly one of u and v is a vertex of C. The *tip* of an antenna $\{u, v\}$ of C is the vertex in $\{u, v\} - V(C)$. Note that the tip of each antenna of C must be either an inner vertex of a path component of H_3 or a vertex of a 4-cycle in H_3, because neither Type-1 nor Type-8 operations can be applied to H_3. An antenna of C is *short* if its tip appears either in a short path component of H_3 or in a 4-cycle of H_3; otherwise, it is *long*. Note that the tip of a long antenna of C must be an inner vertex of a long path component of H_3 because no Type-i operation with $i \in \{1, 8\}$ can be applied to H_3. C is *antenna-sensitive* if it has at least one antenna. On the contrary, C is *antenna-free* if it has no antenna. Similarly, C is *short-antenna-free* if it has no short antenna.

For each antenna-sensitive odd cycle component C of H_3, we define the *representative antenna* of C as follows: If C is short-antenna-free, then we choose an arbitrary antenna of C to be its representative antenna; otherwise, we choose an arbitrary short antenna of C to be its representative antenna. We denote the representative antenna of C by $A(C)$. For each $i \in \{5, 7, 9\}$, we define three sets as follows:

- Let $\mathcal{F}_i$ be the set of all antenna-free i-cycles in H_3.
- Let $\mathcal{S}_i$ be the set of all antenna-sensitive i-cycles C in H_3 such that $A(C)$ is short.
- Let $\mathcal{L}_i$ be the set of all antenna-sensitive i-cycles C in H_3 such that $A(C)$ is long.

Lemma 3.7. $m \ge |E(Opt)| + |\mathcal{F}_5| + |\mathcal{F}_7| + |\mathcal{F}_9|$.

Proof. For convenience, let $\mathcal{F} = \mathcal{F}_5 \cup \mathcal{F}_7 \cup \mathcal{F}_9$. Let U be the set of all vertices in G that appear in cycles in $\mathcal{F}$. Consider an arbitrary cycle $C \in \mathcal{F}$. Since C is antenna-free, there is no edge $\{u, v\}$ in Opt with $|\{u, v\} \cap V(C)| = 1$. So, $E(Opt)$ can be partitioned into two sets E_1 and E_2, where E_1 consists of those edges $\{u, v\} \in E(Opt)$ with $\{u, v\} \subseteq V(G) - V(C)$ and E_2 consists of those edges $\{u, v\} \in E(Opt)$ with $\{u, v\} \subseteq V(C)$. Hence, $(V(C), E_2)$ must be a maximum edge-2-colorable subgraph of C or else Opt would not be a maximum edge-2-colorable subgraph of G. Obviously, a maximum edge-2-colorable subgraph of C must contain exactly $|V(C)| - 1$ edges. Therefore, $|E_2| = |V(C)| - 1$. Consequently, $Opt - U$ is an edge-2-colorable subgraph of $G - U$ and contains exactly $|E(Opt)| - \sum_{C\in\mathcal{F}}(|V(C)| - 1)$ edges.

Since no Type-i operation with $1 \le i \le 11$ creates an odd cycle, every cycle in $\mathcal{F}$ is also a cycle in H_2. So, $H_2 - U$ is a maximum triangle-free cycle-path cover of $G - U$ because H_2 is a maximum triangle-free cycle-path cover of G. This together with the fact that $Opt - U$ is an edge-2-colorable subgraph of $G - U$ implies that the number of edges in $H_2 - U$ is larger than or equal to the number of edges in $Opt - U$. Now, because $H_2 - U$ has exactly $m - \sum_{C\in\mathcal{F}} |V(C)|$ edges and $Opt - U$ has exactly $|E(Opt)| - \sum_{C\in\mathcal{F}}(|V(C)| - 1)$ edges, we have $m - \sum_{C\in\mathcal{F}} |V(C)| \ge |E(Opt)| - \sum_{C\in\mathcal{F}}(|V(C)| - 1)$. Thus, $m \ge |E(Opt)| + |\mathcal{F}_5| + |\mathcal{F}_7| + |\mathcal{F}_9|$. □

Lemma 3.8. $m_{uc} \geq \sum_{i\in\{5,7,9\}}(i|\mathcal{F}_i| + (i+1)|\mathcal{S}_i| + i|\mathcal{L}_i|)$.

Proof. By Invariant I3, every edge in an odd cycle of H_3 is not charged. Note that the total number of edges in odd cycles of H_3 is $\sum_{i\in\{5,7,9\}}(i|\mathcal{F}_i| + i|\mathcal{S}_i| + i|\mathcal{L}_i|)$. Further note that each edge in a 4-cycle of H_3 or a short path component of H_3 is not charged because of Invariant (I4). Thus, it remains to show the following claim:

Claim. The total number of edges in 4-cycles or short path components of H_3 is at least $\sum_{i\in\{5,7,9\}}|\mathcal{S}_i|$.

For each short path component or 4-cycle P in H_3, let $\mathcal{S}_P$ be the set of all cycles $C \in \mathcal{S}_5 \cup \mathcal{S}_7 \cup \mathcal{S}_9$ such that the tip of $A(C)$ is a vertex of P. To show the claim, it suffices to show that for each short path component or 4-cycle P in H_3, $|E(P)| \geq |\mathcal{S}_P|$.

First, consider an arbitrary 4-cycle P in H_3. Recall that the degree of each vertex in *Opt* is at most 2. So, for each vertex v of P, there are at most two cycles $C \in \mathcal{S}_P$ such that v is the tip of $A(C)$. Moreover, since no Type-9 operation can be applied to H_3, there do not exist two cycles C_1 and C_2 in $\mathcal{S}_P$ such that the tips of $A(C_1)$ and $A(C_2)$ are adjacent in P. Therefore, there are at most 4 odd cycles C in H_3 such that $A(C)$ is a vertex of P, or equivalently $|\mathcal{S}_P| \leq 4$. Hence, $|E(P)| = 4 \geq |\mathcal{S}_P|$.

Next, consider an arbitrary short path component P in H_3. Since no Type-1 operation can be applied to H_3, there is no cycle $C \in \mathcal{S}_P$ such that the tip of $A(C)$ is an endpoint of P. Moreover, since no Type-9 operation can be applied to H_3, there do not exist two cycles C_1 and C_2 in $\mathcal{S}_P$ such that the tips of $A(C_1)$ and $A(C_2)$ are adjacent in P. Hence, there are at most $\lceil\frac{|V(P)|-2}{2}\rceil$ vertices $v \in V(P)$ such that $v = A(C)$ for some $C \in \mathcal{S}_P$. Consequently, there are at most $2 \cdot \lceil\frac{|V(P)|-2}{2}\rceil$ odd cycles $C \in \mathcal{S}_P$ such that the tip of $A(C)$ is a vertex of P, because the degree of each vertex in *Opt* is at most 2. Or equivalently, $|\mathcal{S}_P| \leq 2 \cdot \lceil\frac{|V(P)|-2}{2}\rceil$. Therefore, $|E(P)| = |V(P)| - 1 \geq 2 \cdot \lceil\frac{|V(P)|-2}{2}\rceil \geq |\mathcal{S}_P|$. □

Lemma 3.9. *Suppose that C is a cycle in $(\mathcal{L}_5 \cup \mathcal{L}_7 \cup \mathcal{L}_9) - \mathcal{C}_{-2}$ and has two antennas $\{u, x\}$ and $\{v, y\}$ whose tips u and v appear in the same path component P of H_3. Then, (x, y, Q, P, u, v) is a 6-opener for C, where Q is the subgraph of Opt induced by $V(C)$.*

Proof. Since $\{u, x\} \in E(Opt)$ and $d_{Opt}(x) \leq 2$, we have $d_Q(x) \leq 1$. Similarly, we have $d_Q(y) \leq 1$. On the other hand, since C is odd and is not contained in $\mathcal{C}_{-2}$, Q has exactly $|V(C)| - 1$ edges. So, $d_Q(z) = 2$ for every vertex $z \in V(C) - \{x, y\}$. Hence, if $x = y$, then Q is a cycle cover of $G[V(C) - \{x\}]$ in which each connected component is an even cycle; otherwise, Q is a path-cycle cover of $G[V(C)]$ in which one connected component is a path from x to y and each other connected component is an even cycle. □

Lemma 3.10. $|\mathcal{C}_{-2}| + |\overline{T_0}| + \frac{1}{2}|\overline{T_1}| \geq \frac{1}{3}(|\mathcal{L}_5| + |\mathcal{L}_7| + |\mathcal{L}_9|)$.

Proof. For each $i \in \{0, 1\}$, let $\overline{T_{i,\ell}}$ be the set of all vertices $v \in \overline{T_i}$ such that v appears in a long path component of H_3. For convenience, let $\mathcal{L} = \mathcal{L}_5 \cup \mathcal{L}_7 \cup \mathcal{L}_9$. Obviously, the following claim is stronger than the lemma:

Claim 1. $|\overline{T_{0,\ell}}| + \frac{1}{2}|\overline{T_{1,\ell}}| \geq \frac{1}{3}|\mathcal{L} - \mathcal{C}_{-2}|$.

To prove Claim 1, we set up a savings account for each inner vertex in a long path component of H_3. Initially, every account is empty. Then, we deposit a total of $|\overline{T_{0,\ell}}| + \frac{1}{2}|\overline{T_{1,\ell}}|$ credits in the accounts as follows:

- For each vertex $v \in \overline{T_{0,\ell}}$, deposit one credit in the account of v.
- For each vertex $v \in \overline{T_{1,\ell}}$, deposit a half credit in the account of v.

For convenience, for each long path component P of H_3, let $\mathcal{L}_P$ be the set of all cycles $C \in \mathcal{L} - \mathcal{C}_{-2}$ such that the tip of $A(C)$ is a vertex of P. Recall that for each $C \in \mathcal{L}$, the tip of $A(C)$ is an inner vertex of a long path component of H_3 because of the choice of $A(C)$ and the fact that no Type-i operation with $i \in \{1, 8\}$ can be applied to H_3. So, to show Claim 1, it suffices to show the following claim:

Claim 2. For each long path component P of H_3, the credits in the accounts of inner vertices of P sum up to at least $\frac{1}{3}|\mathcal{L}_P|$.

To prove Claim 2, consider an arbitrary long path component P of H_3 with $\mathcal{L}_P \neq \emptyset$. Let s and t be the endpoints of P, and let Q be the path $P - \{s, t\}$. Since P is long, the length of Q is at least 4. For convenience, we call s the *left endpoint* of P and call t the *right endpoint* of P. Moreover, for each inner vertex v of P, we define its *left neighbor* in P to be the vertex $u \in N_P(v)$ with $dist_P(s, u) < dist_P(s, v)$, and define its *right neighbor* in P to be the vertex $w \in N_P(v)$ with $dist_P(t, w) < dist_P(t, v)$. Furthermore, we color each vertex v of Q white or black as follows: If there is a cycle $C \in \mathcal{L}_P$ such that v is the tip of $A(C)$, then color v white; otherwise, color v black. Then, to prove Claim 2 for P, it suffices to show the following claim:

Claim 3. Let W be the set of all white vertices v of Q. Then, the credits in the accounts of inner vertices of P sum up to at least $\frac{2}{3}|W|$.

Claim 3 implies Claim 2, because for each vertex $v \in W$, there are at most two cycles $C \in \mathcal{L}_P$ such that v is the tip of $A(C)$. To prove Claim 3, we first observe the following useful property:

Property 1. Suppose that $u \in W$ and v is a neighbor of u in Q. Then, $v \in \overline{T_0} \cup \overline{T_1}$. Moreover, if $v \in \overline{T_1}$ and v is the left (respectively, right) neighbor of u in P, then the vertex in $N_{Opt}(v) \cap T$ is the left (respectively, right) endpoint of P.

To see Property 1, let C be a cycle in $\mathcal{L}_P$ such that u is the tip of $A(C)$. Since $C \notin \mathcal{C}_{-2}$, v cannot be the tip of an antenna of C or else a Type-6 operation could be applied to H_3 (cf. Lemma 3.9). Moreover, there is no edge $\{v, x\} \in E(Opt)$ with $x \in T_2 - V(C)$ because no Type-9 operation can be applied to H_3. Similarly, there is no edge $\{v, x\} \in E(Opt)$ with $x \in (T_0 \cup T_1) - \{s, t\}$ because no Type-2 operation can be applied to H_3. Furthermore, v cannot be adjacent to both s and t in G because no Type-4 operation can be applied to H_3. More specifically, if v is the left (respectively, right) neighbor of u in P, then v cannot be adjacent to the right (respectively, left) endpoint of P in G because no Type-4 operation can be applied to H_3. Thus, Property 1 holds.

We next use Property 1 to prove Claim 3. For clarity, we distinguish two cases as follows:

Case 1: $|W| = 1$. In this case, we want to show that the credits in the accounts of inner vertices of P sum up to at least 1. Toward this goal, consider the vertex u in W. If u is not an endpoint of Q, then Property 1 guarantees that both neighbors of u in Q belong to $\overline{T_0} \cup \overline{T_1}$, implying that the accounts of the neighbors of u in Q have at least one credit in total. So, suppose that u is an endpoint of Q. Without loss of generality, we may assume that u is the right endpoint of Q. Then, since P is long, the left neighbor v of u in Q exists and so does the left neighbor w of v in Q. By Property 1, either $v \in \overline{T_0}$, or $v \in \overline{T_1}$ and the vertex in $N_{Opt}(v) \cap T$ is s. Hence, if $\{s, v\} \notin E(G) - E(H)$, then $v \in \overline{T_0}$, implying that the account of v has one credit. Therefore, it remains to consider the case where $\{s, v\} \in E(G) - E(H)$. In this case, since no Type-i with $i \in \{3, 4, 5\}$ can be applied to H_3, w belongs to $\overline{T_0} \cup \overline{T_1}$ (in particular, w cannot be adjacent to t or a vertex of C in G). Thus, the accounts of v and w have at least one credit in total.

Case 2: $|W| \geq 2$. For convenience, we define the *leftmost* (respectively, *rightmost*) white vertex in Q to be the white vertex that is closest to s (respectively, t) in P. Moreover, we say that two white vertices u and v are *consecutive* if no inner vertex of the subpath of Q between u and v is white. Note that if u and v are two consecutive white vertices, then Property 1 guarantees that the subpath of Q between u and v contains at least one inner vertex. Again, for convenience, we call a vertex $w \in V(Q) - W$ a *delimiter*, if either (1) its left neighbor in Q is white but is not the rightmost white vertex, or (2) its right neighbor in Q is white but is not the leftmost white vertex. The crucial point is that each delimiter belongs to $\overline{T_0}$. This follows from Property 1 and the fact that no Type-10 operation can be applied to H_3. We further distinguish two subcases as follows:

Case 2.1: There do not exist two consecutive white vertices u and v in Q with $dist_Q(u, v) > 2$. In this case, there are exactly $|W| - 1$ delimiters in Q and hence we are done if $|W| \geq 3$ (because $|W| - 1 \geq \frac{2}{3}|W|$ when $|W| \geq 3$). So, assume that $|W| = 2$. Let u and v be the vertices in W. We may assume that $dist_P(s, u) < dist_P(s, v)$. Since $|E(P)| \geq 6$, u has a left neighbor in Q or v has a right neighbor in Q. We assume that u has a left neighbor in Q; the other case is similar. Let w be the left neighbor of u in Q. Then, by Property 1, w belongs to $\overline{T_0} \cup \overline{T_1}$ and hence its account has at least a half credit. Also recall that the (unique) delimiter in Q has one credit in its account. Therefore, the total credits in the accounts of w and the delimiter is at least 1.5, which is not smaller than $\frac{2}{3} \cdot |W|$.

Case 2.2: There exist two consecutive white vertices u and v in Q with $dist_Q(u, v) > 2$. Without loss of generality, we may assume that $dist_P(s, u) < dist_P(s, v)$. Then, for each white vertex w on the subpath of P between s and u, the right neighbor of w in Q is a delimiter. Similarly, for each white vertex w on the subpath of P between t and v, the left neighbor of w in Q is a delimiter. Moreover, the right neighbor of u in Q and the left neighbor of v in Q are different delimiters for $dist_Q(u, v) > 2$. Hence, there are at least as many delimiters as white vertices. Therefore, the total credits in the accounts of the delimiters is at least $|W|$ because each delimiter belongs to $\overline{T_0}$ and so its account has one credit. □

Lemma 3.11. $|E(H_4)| \geq \frac{24}{29}|E(Opt)|$.

Proof. The proof is similar to that of Lemma 3.6. Combining Lemmas 3.4 and 3.5, we have $|E(Opt)| \leq 2m - n_{pc} - 3m_{-} + 3m_{+,0} + 2m_{+,-1} - n_{0,-1} - 6c_5 - 8c_7 - 10c_9 - (|\mathcal{C}_{-2}| + |\overline{T_0}| + \frac{1}{2}|\overline{T_1}|)$. So, by Statement 2 in Lemma 3.2, $3|E(H_4)| - |E(Opt)| \geq m + n_{pc} + m_{+,-1} + n_{0,-1} + 3c_5 + 5c_7 + 7c_9 + |\mathcal{C}_{-2}| + |\overline{T_0}| + \frac{1}{2}|\overline{T_1}|$. Thus, by Lemma 3.10, $3|E(H_4)| - |E(Opt)| \geq m + 3c_5 + 5c_7 + 7c_9 + \frac{1}{3}(|\mathcal{L}_5| + |\mathcal{L}_7| + |\mathcal{L}_9|)$. Consequently, $3|E(H_4)| - |E(Opt)| \geq m + 3|\mathcal{S}_5| + 5|\mathcal{S}_7| + 7|\mathcal{S}_9| + \frac{10}{3}|\mathcal{L}_5| + \frac{16}{3}|\mathcal{L}_7| + \frac{19}{3}|\mathcal{L}_9|$ because $c_i = |\mathcal{S}_i| + |\mathcal{F}_i| + |\mathcal{L}_i|$ for each $i \in \{5, 7, 9\}$.

On the other hand, by Statements 1 and 2 in Lemma 3.2, $|E(H_4)| \geq \frac{9}{10}(m + m_{+,0} + m_{+,-1}) + \frac{1}{10}m_{uc} - c_5 - c_7 - c_9$. So, by Lemma 3.8, $|E(H_4)| \geq \frac{9}{10}m - \frac{2}{5}|\mathcal{S}_5| - \frac{1}{5}|\mathcal{S}_7| - \frac{1}{2}|\mathcal{F}_5| - \frac{3}{10}|\mathcal{F}_7| - \frac{1}{10}|\mathcal{F}_9| - \frac{1}{2}|\mathcal{L}_5| - \frac{3}{10}|\mathcal{L}_7| - \frac{1}{10}|\mathcal{L}_9|$. Consequently, by Lemma 3.7, $|E(H_4)| \geq \frac{9}{10}|E(Opt)| - \frac{2}{5}|\mathcal{S}_5| - \frac{1}{5}|\mathcal{S}_7| - \frac{1}{2}|\mathcal{L}_5| - \frac{3}{10}|\mathcal{L}_7| - \frac{1}{10}|\mathcal{L}_9|$. This together with the last inequality in the last paragraph implies that $\frac{3}{20}(3|E(H_4)| - |E(Opt)|) + |E(H_4)| \geq \frac{9}{10}|E(Opt)| + \frac{3}{20}m$. Rearranging this inequality and using the fact that $m \geq |E(Opt)|$, we finally obtain $|E(H_4)| \geq \frac{24}{29}|E(Opt)|$. □

Theorem 3.12. *There is an $O(n^2m^2)$-time approximation algorithm for* MAX SIMPLE EDGE 2-COLORING *achieving a ratio of $\frac{24}{29}$, where n (respectively, m) is the number of vertices (respectively, edges) in the input graph.*

Proof. We estimate the running time of the algorithm as follows. Step 1 can be done in $O(n^2m^2)$ time [3]. Obviously, Steps 2 through 4 can be done in $O(n(n + m))$ time, because each operation can be performed in $O(n + m)$ time and each step performs a total of $O(n)$ operations. □

4. An application

Let G be a graph. An *edge cover* of G is a set F of edges of G such that each vertex of G is incident to at least one edge of F. For a natural number k, a $[1, \Delta]$-*factor k-packing* of G is a collection of k disjoint edge covers of G. The *size* of a $[1, \Delta]$-factor

k-packing $\{F_1, \ldots, F_k\}$ of G is $|F_1| + \cdots + |F_k|$. The problem of deciding whether a given graph has a $[1, \Delta]$-factor k-packing was considered in [5,6]. In [8], Kosowski et al. defined the *minimum* $[1, \Delta]$-*factor k-packing problem* (Min-k-FP) as follows: Given a graph G, find a $[1, \Delta]$-factor k-packing of G of minimum size or decide that G has no $[1, \Delta]$-factor k-packing at all.

According to [8], Min-2-FP is of special interest because it can be used to solve a fault tolerant variant of the guards problem in grids (which is one of the art gallery problems [9,10]). Indeed, they proved the NP-hardness of Min-2-FP and the following lemma:

Lemma 4.1. *If* Max Simple Edge 2-Coloring *admits an approximation algorithm A achieving a ratio of* α, *then* Min-2-FP *admits an approximation algorithm B achieving a ratio of* $2 - \alpha$. *Moreover, if the time complexity of A is* $T(n)$, *then the time complexity of B is* $O(T(n))$.

So, by Theorem 3.12, we have the following immediately:

Theorem 4.2. *There is an* $O(n^2m^2)$*-time approximation algorithm for* Min-2-FP *achieving a ratio of* $\frac{34}{29}$, *where n (respectively, m) is the number of vertices (respectively, edges) in the input graph.*

Previously, the best ratio achieved by a polynomial-time approximation algorithm for Min-2-FP was $\frac{682}{575}$ [1], although Min-2-FP admits a polynomial-time approximation algorithm achieving a ratio of $\frac{42\Delta-30}{35\Delta-21}$, where Δ is the maximum degree of a vertex in the input graph [8].

Note added in proof. One referee pointed out a recent paper by Kosowski [7] to us, in which a new approximation algorithm for Max Simple Edge 2-Coloring is given. His algorithm has the same time complexity as ours but achieves a slightly better ratio (namely, $\frac{5}{6} \approx 0.833$). It also seems that his algorithm was made public slightly earlier than ours. However, his work and ours are completely independent. Indeed, his algorithm and ours are based on completely different approaches. We believe that, by combining the two approaches, we can improve both his result and ours.

Acknowledgment

First author was supported in part by the Grant-in-Aid for Scientific Research of the Ministry of Education, Science, Sports and Culture of Japan, under Grant No. 20500021.

References

[1] Z.-Z. Chen, R. Tanahashi, L. Wang, An improved approximation algorithm for maximum edge 2-coloring in simple graphs, Journal of Discrete Algorithms 6 (2) (2008) 205–215. A preliminary version appeared in Proceedings of 3rd International Conference on Algorithmic Aspects in Information and Management, Lecture Notes in Computer Science, vol. 4508, 2007, pp. 27–36.
[2] U. Feige, E. Ofek, U. Wieder, Approximating maximum edge coloring in multigraphs, in: Proceedings of the 10th International Conference on Integer Programming and Combinatorial Optimization, IPCO, in: Lecture Notes in Computer Science, vol. 2462, 2002, pp. 108–121.
[3] D. Hartvigsen, Extensions of matching theory, Ph.D. Thesis, Carnegie-Mellon University, 1984.
[4] D. Hochbaum, Approximation Algorithms for NP-Hard Problems, PWS Publishing Company, Boston, 1997.
[5] D.P. Jacobs, R.E. Jamison, Complexity of recognizing equal unions in families of sets, Journal of Algorithms 37 (2000) 495–504.
[6] K. Kawarabayashi, H. Matsuda, Y. Oda, K. Ota, Path factors in cubic graphs, Journal of Graph Theory 39 (2002) 188–193.
[7] A. Kosowski, Approximating the maximum 2- and 3-edge-colorable subgraph problems. Discrete Applied Mathematics (in press).
[8] A. Kosowski, M. Malafiejski, P. Zylinski, Packing $[1, \Delta]$-factors in graphs of small degree, Journal of Combinatorial Optimization 14 (2007) 63–86.
[9] J. O'Rourke, Art Gallery Theorems and Algorithms, Oxford University Press, 1987.
[10] J. Urrutia, Art Gallery and Illumination Problems, in: Handbook on Computational Geometry, Elsevier Science, Amsterdam, 2000.