Journal of Combinatorial Optimization (2021) 41:12–27
https://doi.org/10.1007/s10878-020-00660-7

A randomized approximation algorithm for metric triangle packing

Yong Chen[1] · Zhi-Zhong Chen[2] · Guohui Lin[3] · Lusheng Wang[4] · An Zhang[1]

Accepted: 7 October 2020 / Published online: 13 October 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

Given an edge-weighted complete graph G on $3n$ vertices, the maximum-weight triangle packing problem asks for a collection of n vertex-disjoint triangles in G such that the total weight of edges in these n triangles is maximized. Although the problem has been extensively studied in the literature, it is surprising that prior to this work, no nontrivial approximation algorithm had been designed and analyzed for its metric case, where the edge weights in the input graph satisfy the triangle inequality. In this paper, we design the first nontrivial polynomial-time approximation algorithm for the maximum-weight metric triangle packing problem. Our algorithm is randomized and achieves an expected approximation ratio of $0.66768 - \epsilon$ for any constant $\epsilon > 0$.

Keywords Triangle packing · Metric · Approximation algorithm · Randomized algorithm · Maximum cycle cover

---

An extended abstract appears in the Proceedings of COCOA 2019. LNCS 11949, pages 119–129.

✉ Zhi-Zhong Chen
zzchen@mail.dendai.ac.jp

✉ Guohui Lin
guohui@ualberta.ca

Yong Chen
chenyong@hdu.edu.cn ; anzhang@hdu.edu.cn

Lusheng Wang
cswangl@cityu.edu.hk

1 Department of Mathematics, Hangzhou Dianzi University, Hangzhou, China

2 Division of Information System Design, Tokyo Denki University, Saitama, Japan

3 Department of Computing Science, University of Alberta, Edmonton, AB T6G 2E8, Canada

4 Department of Computer Science, City University of Hong Kong, Hong Kong SAR, China

1 Introduction

An instance of the *maximum-weight triangle packing* problem (MWTP for short) is an edge-weighted complete graph G on $3n$ vertices, where n is a positive integer. Given G, the objective of MWTP is to compute n vertex-disjoint triangles (a.k.a. cycles of length 3) such that the total weight of edges in these n triangles is maximized.

The unweighted (i.e., edge uniformly weighted) variant (MTP for short) is to compute the maximum number of vertex-disjoint triangles in the input graph, which is an edge-unweighted but incomplete graph.

In their classic book, (Garey and Johnson 1979, GT11) show that MTP (in fact, a special case called *partition into triangles*) is NP-hard. Kann (1991) and van Rooij et al. (2013) show that MTP is APX-hard even restricted on graphs of maximum degree 4. Chlebík and Chlebíková (2003) show that unless P = NP, no polynomial-time approximation algorithm for MTP can achieve an approximation ratio of 0.9929. Moreover, Guruswami et al. (1998) show that MTP remains NP-hard even restricted on chordal, planar, line, or total graphs.

MTP can be easily cast as a special case of the *unweighted* 3-*set packing* problem (U3SP for short). Recall that an instance of U3SP is a family $\mathcal{F}$ of sets each of size at most 3 and the objective is to compute a maximum-sized family of disjoint sets in $\mathcal{F}$. On the positive approximation results, Hurkens and Schrijver (1989) (also see Halldórsson 1995) present a nontrivial polynomial-time algorithm for U3SP which achieves an approximation ratio of $\frac{2}{3} - \epsilon$ for any constant $\epsilon > 0$. This ratio has been improved to $\frac{3}{4} - \epsilon$ (Cygan 2013; Fürer and Yu 2014). Therefore, MTP can also be approximated within $\frac{3}{4} - \epsilon$. When restricted to graphs of maximum degree 4, Manic and Wakabayashi (2008) present a polynomial-time 0.833-approximation algorithm for MTP.

Analogously, MWTP can be cast as a special case of the *weighted* 3-*set packing* problem (W3SP for short). Two different algorithms both based on local search have been designed for W3SP (Arkin and Hassin 1998; Berman 2000) and they happen to achieve the same approximation ratio of $\frac{1}{2} - \epsilon$ for any constant $\epsilon > 0$. For MWTP specifically, Hassin and Rubinstein (2006a, b) present a better randomized approximation algorithm with an expected approximation ratio of $\frac{43}{83} - \epsilon$ (≈ 0.518) for any constant $\epsilon > 0$. This ratio has been improved to roughly 0.523 by Chen et al. (2009, 2010) and van Zuylen (2013).

The current paper focuses on a common special case of MWTP, namely, the *metric* MWTP problem (MMWTP for short), where the edge weights in the input graph satisfy the triangle inequality. Note that both NP-hardness and APX-hardness of MMWTP follow from a trivial reduction from the MTP problem (Garey and Johnson 1979; Kann 1991; van Rooij et al. 2013), in which one assigns a weight of 2 to each edge inside the instance graph or a weight of 1 otherwise. Also, one can easily (for example, as in Sect. 2.1) design a polynomial-time approximation algorithm for MMWTP to achieve an approximation ratio of $\frac{2}{3}$; but surprisingly, prior to our work, no nontrivial approximation algorithm had been designed and analyzed. In this paper, we design the first nontrivial $O(n^3)$-time approximation algorithm for MMWTP. Our algorithm is randomized and achieves an expected ratio of $0.66768 - \epsilon$ for any constant $\epsilon > 0$.

At the high level, given an instance graph G and an assumed optimal triangle packing B, our algorithm first computes a triangle packing T_1 based on two maximum-weight matchings, which turns out a good (better than $\frac{2}{3}$) approximation when there is a non-trivial portion of *unbalanced* triangles (defined in Sect. 2.1) in B. Next, noting that B is a cycle cover, our algorithm computes a maximum-weight cycle cover and transforms it into another cycle cover $\mathcal{C}$ of almost the same weight but containing only *short* cycles. Using $\mathcal{C}$, our algorithm then constructs a *partial* triangle packing (defined in Sect. 2.2) which has at least as many vertex-components as edge-components, and augments it into a triangle packing T_2 of weight greater than or equal to one subset of edges in B with respect to $\mathcal{C}$. Our algorithm lastly constructs another triangle packing T_3 based on a random matching and a maximum-weight matching in $\mathcal{C}$, and it is shown that T_1 and T_3 together are able to pick up the weight of the rest of the edges in B not picked up by T_2. The computation of T_1 and T_2 is deterministic but that of T_3 is randomized. Our algorithm returns the best among the three triangle packings T_1, T_2 and T_3, which has the desired quality by a performance ratio analysis using a simple linear program.

The details of our algorithm and its performance analysis are presented in the next section. We conclude the paper in the last Sect. 3, with some remarks.

2 The randomized approximation algorithm

Hereafter, let G be a given instance of the MMWTP problem. We fix an optimal triangle packing B of G for the following argument. Recall that G is an edge-weighted complete graph on $3n$ vertices and the edge weights satisfy the triangle inequality. Let $w(\cdot)$ denote the edge weight function. We extend it to $w(S)$ to denote the total weight of edges in S, where S can be either an edge subset or a subgraph.

The algorithm starts by computing a maximum-weight cycle cover $\mathcal{C}$ of G in $O(n^3)$ time (Hartvigsen 1984). Obviously, $w(\mathcal{C}) \geq w(B)$, because B is also a cycle cover. Let ϵ be any constant such that $0 < \epsilon < 1$. A cycle C in $\mathcal{C}$ is *short* if its length (measured as the number of edges therein) is at most $\lceil \frac{1}{\epsilon} \rceil$; otherwise, it is *long*. It is easy to transform each long cycle C in $\mathcal{C}$ into two or more short cycles whose total weight is at least $(1-\epsilon) \cdot w(C)$. So, we hereafter assume that we have modified the long cycles in $\mathcal{C}$ in this way. Then, $\mathcal{C}$ is a collection of short cycles and $w(\mathcal{C}) \geq (1-\epsilon) \cdot w(B)$.

We will compute below three triangle packings T_1, T_2, T_3 in G. The computation of T_1 and T_2 will be deterministic but that of T_3 will be randomized. Our goal is to prove that there is a constant $\rho > 0.001$, such that $\max\{w(T_1), w(T_2), \mathcal{E}[w(T_3)]\} \geq (\frac{2}{3} + \rho) \cdot w(B)$, where $\mathcal{E}[X]$ denotes the expected value of a random variable X.

2.1 Computing T_1

We first compute a maximum-weight matching M_1 of size n (i.e., n edges) in G in $O(n^3)$ time (Gabow 1976). We then construct an auxiliary complete bipartite graph H_1 as follows. One part of $V(H_1)$ is $V \setminus V(M_1)$, which consists of the vertices of G that are not endpoints of M_1; the vertices of the other part of $V(H_1)$, still denoted as M_1, one-to-one correspond to the edges in M_1. For each edge $\{x, e = \{u, v\}\}$ in the bipartite graph H_1, where $x \in V \setminus V(M_1)$ and $e \in M_1$, its weight is set to $w(u, x) + w(v, x)$. Next, we compute a maximum-weight matching M_1' in H_1 in another $O(n^3)$ time and transform it back into a triangle packing T_1 of G with its weight $w(T_1) = w(M_1) + w(M_1')$.

To compare $w(T_1)$ against $w(B)$, we fix a constant δ with $0 \le \delta < 1$ and classify the triangles in B into two types as follows. A triangle t in B is *balanced* if the minimum weight of an edge in t is at least $1 - \delta$ times the maximum weight of an edge in t; otherwise, t is *unbalanced*.

Lemma 1 *Let $B_{\bar{b}}$ be the set of unbalanced triangles in B, and $\gamma = \frac{w(B_{\bar{b}})}{w(B)}$. Then, $w(T_1) \ge \left(\frac{2}{3} + \frac{2\gamma\delta}{9-3\delta}\right) \cdot w(B)$.*

Proof For each t in B, let a_t (respectively, b_t) be the maximum (respectively, minimum) weight of an edge in t. Further let $a = \sum_{t\in B} a_t$ and $b = \sum_{t\in B} b_t$. If $t \in B_{\bar{b}}$, then $b_t < (1-\delta)a_t$ and in turn $(3-\delta)a_t > w(t)$. Thus,

$$\sum_{t\in B_{\bar{b}}} a_t \ge \frac{1}{3-\delta} w(B_{\bar{b}}) = \frac{\gamma}{3-\delta} w(B).$$

When t is balanced, we still have $b_t \le a_t$; hence,

$$w(B) \le 2a + b \le 3a - \delta \sum_{t\in B_{\bar{b}}} a_t \le 3a - \frac{\delta\gamma}{3-\delta} w(B)$$

and in turn $a \ge \left(\frac{1}{3} + \frac{\delta\gamma}{9-3\delta}\right) w(B)$. On the other hand, the triangle inequality implies $w(T_1) \ge 2a$, and consequently,

$$w(T_1) \ge \left(\frac{2}{3} + \frac{2\gamma\delta}{9-3\delta}\right) \cdot w(B).$$

This proves the lemma. □

We remark that by Lemma 1, the above algorithm for computing the triangle packing T_1 is an $O(n^3)$-time $\frac{2}{3}$-approximation for the MMWTP problem.

2.2 Computing T_2

We start with several definitions.

A *partial-triangle packing* in a graph is a subgraph P of the graph such that each connected component of P is a vertex, an edge, or a triangle. A connected component C of P is a *vertex-component* (respectively, *an edge-component*, or *a triangle-component*) of P if C is a vertex (respectively, an edge, or a triangle). The *augmented weight* of P, denoted by $\hat{w}(P)$, is $\sum_t w(t) + 2\sum_e w(e)$, where t (respectively, e) ranges over all triangle-components (respectively, edge-components) of P. Intuitively speaking, if P has at least as many vertex-components as edge-components, then we can augment P into a triangle packing P', so that $w(P') \geq \hat{w}(P)$, as follows:

1. Fix an arbitrary injective function $f(\cdot)$ from the edge-components of P to the vertex-components of P.
2. For each edge-component e of P, connect the endpoints of e to $f(e)$ by adding two new edges. (*Comment:* At the end of this step, P can have only vertex- or triangle-components.)
3. Arbitrarily partition the set of vertex-components of P into disjoint subsets of size 3, and further connect the three vertex-components in each subset into a triangle by adding the three edges. This constructs P'.

Recall that B is the optimal triangle packing we fixed for discussion, and $\mathcal{C}$ is a computed cycle cover consisting of short cycles only. We classify the triangles t in B into three types as follows.

- t is *completely internal* if all its vertices fall on the same cycle in $\mathcal{C}$.
- t is *partially internal* if exactly two of its vertices fall on the same cycle in $\mathcal{C}$.
- t is *external* if no two of its vertices fall on the same cycle in $\mathcal{C}$.

An edge e of B is *external* if the endpoints of e fall on different cycles in $\mathcal{C}$; otherwise, e is *internal*. In particular, an internal edge e of B is *completely* (respectively, *partially*) *internal* if e appears in a completely (respectively, partially) internal triangle in B. A vertex v of G is *external* if it is incident to no internal edges of B. See Fig. 1 for an illustration. Let $B_{\bar{e}}$ be the partial-triangle packing in G obtained from B by deleting all external edges.

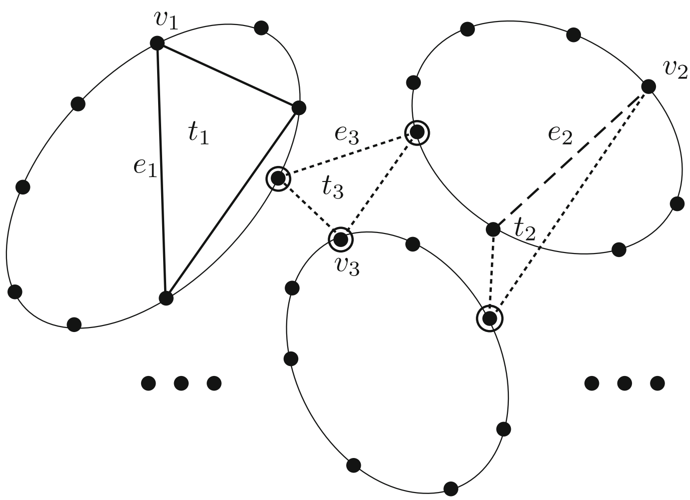

Fig. 1 An illustration of the three types of triangles, the three types of edges, and two types of vertices. The cycles of $\mathcal{C}$ are in ovals and the edges of B are straight lines. The triangle t_1, t_2, and t_3 are completely internal, partially internal, and external, respectively; the edges e_1 (solid), e_2 (dashed), and e_3 (dotted) are completely internal, partially internal, and external, respectively; the circled vertices such as v_3 are external

Now, we are ready to explain how to construct T_2 so that $w(T_2) \geq \hat{w}(B_{\bar{e}})$. Let C_1, …, C_ℓ be the cycles in $\mathcal{C}$, and V_1, …, V_ℓ be their vertex sets. For each $i \in \{1, \ldots, \ell\}$, let $n_i = |V_i|$, p_i be the number of partially internal edges e in B such that both endpoints of e appear in C_i, q_i be the number of external vertices in C_i, and E_i be the set of edges $\{u, v\}$ in G with $\{u, v\} \subseteq V_i$ (i.e., the edge set of the subgraph $G[V_i]$ induced on V_i). Obviously, $n_i - 2p_i - q_i$ is the number of vertices on the completely internal triangles in $G[V_i]$, and is hence a multiple of 3. For each $i \in \{1, 2, \ldots, \ell\}$, let $\tilde{n}_i = \sum_{h=1}^{i} n_h$, $\tilde{p}_i = \sum_{h=1}^{i} p_h$, and $\tilde{q}_i = \sum_{h=1}^{i} q_h$.

Although we do not know p_i and q_i, we easily see that $0 \leq q_i \leq n_i$ and $0 \leq p_i \leq \lfloor \frac{n_i - q_i}{2} \rfloor$. So, for every $j \in \{0, 1, \ldots, n_i\}$ and every $k \in \{0, 1, \ldots, \lfloor \frac{n_i - j}{2} \rfloor\}$, we compute the maximum-weight (under $\hat{w}(\cdot)$) partial-triangle packing $P_i(j, k)$ in $G[V_i]$ such that $P_i(j, k)$ has exactly j vertex-components and exactly k edge-components (and exactly $\frac{1}{3}(n_i - 2k - j)$ triangles). Since $n_i = |V_i| \leq \lceil \frac{1}{\epsilon} \rceil$, the computation of $P_i(j, k)$ can be done by enumeration in $O(1)$ time.

Similarly, although we do not know $\tilde{p}_i$ and $\tilde{q}_i$, we easily see that $0 \leq \tilde{q}_i \leq \tilde{n}_i$ and $0 \leq \tilde{p}_i \leq \lfloor \frac{\tilde{n}_i - \tilde{q}_i}{2} \rfloor$. For every $j \in \{0, 1, \ldots, \tilde{n}_i\}$ and every $k \in \{0, 1, \ldots, \lfloor \frac{\tilde{n}_i - j}{2} \rfloor\}$, we want to compute a maximum-weight (under $\hat{w}(\cdot)$) partial-triangle packing $\tilde{P}_i(j, k)$ in $G[\bigcup_{h=1}^{i} V_h]$ such that $\tilde{P}_i(j, k)$ has exactly j vertex-components and exactly k edge-components. This can be done by dynamic programming in $O(n^3)$ time as follows:

1. In the boundary case, we clearly have $\tilde{P}_1(j, k) = P_1(j, k)$ for every $j \in \{0, 1, \ldots, \tilde{n}_1\}$ and every $k \in \{0, 1, \ldots, \lfloor \frac{\tilde{n}_1 - j}{2} \rfloor\}$.
2. To develop the recurrence, suppose that $1 \leq i < \ell$ and we have computed $\tilde{P}_i(j, k)$ for every $j \in \{0, 1, \ldots, \tilde{n}_i\}$ and every $k \in \{0, 1, \ldots, \lfloor \frac{\tilde{n}_i - j}{2} \rfloor\}$. For every $j \in \{0, 1, \ldots, \tilde{n}_{i+1}\}$ and every $k \in \{0, 1, \ldots, \lfloor \frac{\tilde{n}_{i+1} - j}{2} \rfloor\}$, we can compute $\tilde{P}_{i+1}(j, k)$ by finding a pair (j', k') such that $j' \in \{0, 1, \ldots, n_{i+1}\}$, $k' \in \{0, 1, \ldots, \lfloor \frac{n_{i+1} - j'}{2} \rfloor\}$, and $\hat{w}(P_{i+1}(j', k')) + \hat{w}(\tilde{P}_i(j - j', k - k'))$ is maximized; and let $\tilde{P}_{i+1}(j, k) = P_{i+1}(j', k') \cup \tilde{P}_i(j - j', k - k')$.

Note that $\tilde{n}_\ell = 3n$. Finally, we have $\tilde{P}_\ell(j, k)$ for every $j \in \{0, 1, \ldots, 3n\}$ and every $k \in \{0, 1, \ldots, \lfloor \frac{3n - j}{2} \rfloor\}$. We now find a pair (j', k') such that $j' \in \{0, 1, \ldots, 3n\}$, $k' \in \{0, 1, \ldots, \lfloor \frac{3n - j'}{2} \rfloor\}$, $k' \leq j'$, and $\hat{w}(\tilde{P}_\ell(j', k'))$ is maximized. It follows that $\hat{w}(\tilde{P}_\ell(j', k')) \geq \hat{w}(B_{\bar{e}})$. Since $\tilde{P}_\ell(j', k')$ is a partial-triangle packing containing at least as many vertex-components as edge-components, we can transform $\tilde{P}_\ell(j', k')$ into a triangle packing T_2 of G with $w(T_2) \geq \hat{w}(\tilde{P}_\ell(j', k'))$ the same as before.

In summary, we have shown the following lemma:

Lemma 2 *A triangle packing T_2 of G with $w(T_2) \geq \hat{w}(B_{\bar{e}})$ can be constructed out of $\mathcal{C}$ in $O(n^3)$ time.*

2.3 Computing a random matching in $\mathcal{C}$

We next compute a random matching M in $\mathcal{C}$ as follows, in $O(n)$ time.

1. Initialize two sets $L = \emptyset$ and $M = \emptyset$.
2. For each even cycle C_i in $\mathcal{C}$, perform the following three steps:

(a) Partition $E(C_i)$ into two matchings $M_{i,1}$ and $M_{i,2}$.
(b) Select a $j_i \in \{1, 2\}$ uniformly at random.
(c) Add the edges in M_{i,j_i} to L.

3. For each odd cycle C_i in $\mathcal{C}$, perform the following five steps:

(a) Select an edge $e_i \in E(C_i)$ uniformly at random.
(b) Partition $E(C_i) \setminus \{e_i\}$ into two matchings $M_{i,1}$ and $M_{i,2}$.
(c) Select a $j_i \in \{1, 2\}$ uniformly at random.
(d) Select an edge $e'_i \in M_{i,j_i}$ uniformly at random and add e'_i to M.
(e) Add the edges in $M_{i,j_i} \setminus \{e'_i\}$ to L.

4. Select two thirds of edges from L uniformly at random and add them to M.

In the sequel, unless otherwise explicitly stated, L and M mean the sets L and M obtained at the end of Steps 3 and 4 , respectively.

Lemma 3 *Let c_o be the number of odd cycles in $\mathcal{C}$. Then,* $|L| = \frac{3}{2} \cdot (n - c_o)$.

Proof Each even cycle C_i contributes $\frac{1}{2}|V_i|$ edges to L, and each odd cycle C_i contributes $\frac{1}{2}(|V_i| - 1) - 1 = \frac{1}{2}(|V_i| - 3)$ edges to L, where V_i is the vertex set of C_i. Hence $|L| = \frac{1}{2}(3n - 3c_o) = \frac{3}{2} \cdot (n - c_o)$. □

Lemma 4 *$L \cup M$ is a matching and $|M| = n$.*

Proof One sees that for each cycle C in $\mathcal{C}$, the edges of C selected into $L \cup M$ in Step 2 or 3 form a matching, and thus $L \cup M$ is a matching of G.

At the end of Step 3, each odd cycle C_i contributes one edge to M and that is it; that is, $|M| = c_o$. So, by Lemma 3, at the end $|M| = c_o + (n - c_o) = n$. □

Lemma 5 *For every vertex v of G,* $\Pr[v \notin V(M)] = \frac{1}{3}$.

Proof First consider the case where v appears in an even cycle in $\mathcal{C}$. In this case, $v \in V(L)$ at the end of Step 3. So, $\Pr[v \notin V(M)] = \frac{1}{3}$ because the edge incident at v is added to M with probability $\frac{2}{3}$ in Step 4.

Next consider the case where v appears in an odd cycle C_i in $\mathcal{C}$. There are two subcases, depending on whether or not v is an endpoint of the edge e_i selected in Step 3a. If v is incident to e_i, then $\Pr[v \notin V(M_{i,j_i})] = \frac{1}{2}$ and $\Pr[v \in V(M_{i,j_i}) \wedge v \notin V(e'_i)] = \frac{1}{2} \cdot \left(1 - \frac{2}{n_i - 1}\right)$, where $n_i = |V_i|$. Hence, the conditional probability $\Pr[v \notin V(M) \mid v \in V(e_i)] = \Pr[v \notin V(M_{i,j_i}) \mid v \in V(e_i)] + \Pr[v \in V(M_{i,j_i}) \wedge v \notin V(e'_i) \mid v \in V(e_i)] \cdot \Pr[v \notin V(M) \mid v \in V(M_{i,j_i}) \wedge v \notin V(e'_i)] = \frac{1}{2} + \frac{1}{2} \cdot \left(1 - \frac{2}{n_i - 1}\right) \cdot \frac{1}{3} = \frac{2n_i - 3}{3(n_i - 1)}$.

On the other hand, if v is not an endpoint of e_i, then $\Pr[v \in V(M_{i,j_i})] = 1$ and $\Pr[v \in V(M_{i,j_i}) \wedge v \notin V(e'_i)] = 1 \cdot \left(1 - \frac{2}{n_i - 1}\right)$. Thus, the conditional probability $\Pr[v \notin V(M) \mid v \notin V(e_i)] = \Pr[v \in V(M_{i,j_i}) \wedge v \notin V(e'_i) \mid v \notin V(e_i)] \cdot \Pr[v \notin V(M) \mid v \in V(M_{i,j_i}) \wedge v \notin V(e'_i)] = \left(1 - \frac{2}{n_i - 1}\right) \cdot \frac{1}{3} = \frac{n_i - 3}{3(n_i - 1)}$.

It follows from $\Pr[v \in V(e_i)] = \frac{2}{n_i}$ that $\Pr[v \notin V(M)] = \frac{2}{n_i} \cdot \frac{2n_i - 3}{3(n_i - 1)} + \left(1 - \frac{2}{n_i}\right) \cdot \frac{n_i - 3}{3(n_i - 1)} = \frac{1}{3}$. □

Lemma 6 *For every edge e of $\mathcal{C}$,* $\Pr[e \in M] = \frac{1}{3}$.

Proof First consider the case where e appears in an even cycle in $\mathcal{C}$. In this case, $\Pr[e \in M] = \frac{1}{2} \cdot \frac{2}{3} = \frac{1}{3}$ because the edge is added to L with probability $\frac{1}{2}$ and then is added to M with probability $\frac{2}{3}$.

Next consider the case where e appears in an odd cycle C_i in $\mathcal{C}$. There are two subcases, depending on whether or not e is the edge e_i selected in Step 3a. If $e = e_i$, then $\Pr[e \notin M] = 1$. Hence, the conditional probability $\Pr[e \notin M \mid e = e_i] = 1$. On the other hand, if $e \neq e_i$, then $\Pr[e \notin M_{i,j_i}] = \frac{1}{2}$ and $\Pr[e \neq e'_i \mid e \in M_{i,j_i}] = 1 - \frac{2}{n_i-1} = \frac{n_i-3}{n_i-1}$. Thus, the conditional probability $\Pr[e \notin M \mid e \neq e_i] = \Pr[e \notin M_{i,j_i} \mid e \neq e_i] + \Pr[e \neq e'_i \mid e \in M_{i,j_i}] \cdot \Pr[e \in M_{i,j_i} \mid e \neq e_i] \cdot \Pr[e \notin M \mid e \in M_{i,j_i} \wedge e \neq e'_i] = \frac{1}{2} + \frac{n_i-3}{n_i-1} \cdot \frac{1}{2} \cdot \frac{1}{3} = \frac{2n_i-3}{3(n_i-1)}$.

It follows from $\Pr[e = e_i] = \frac{1}{n_i}$ that $\Pr[e \notin M] = \frac{1}{n_i} \cdot 1 + \left(1 - \frac{1}{n_i}\right) \cdot \frac{2n_i-3}{3(n_i-1)} = \frac{2}{3}$, and therefore, $\Pr[e \in M] = \frac{1}{3}$. □

Lemma 7 *For every vertex v of G and every edge e of $\mathcal{C}$ such that v and e appear in different cycles in $\mathcal{C}$,* $\Pr[e \in M \wedge v \notin V(M)] \geq \frac{1}{9}$.

Proof Suppose that v and e appear in $C_{i'}$ and $C_{i''}$, respectively. The two events $e \in M$ and $v \notin V(M)$ are not independent because it is possible that both e and at least one edge incident to v in $\mathcal{C}$ are added to L in Step 2 or 3.

We distinguish four cases according to the parities of $n_{i'}$ and $n_{i''}$, as follows.

Case 1 Both *$n_{i'}$ and $n_{i''}$ are even*. In this case, $\Pr[v \in V(M_{i',j_{i'}})] = 1$ and $\Pr[e \in M_{i'',j_{i''}}] = \frac{1}{2}$. So, $\Pr[v \in V(M_{i',j_{i'}}) \wedge e \in M_{i'',j_{i''}}] = \frac{1}{2}$. Moreover, by Lemma 3, $\Pr[e \in M \wedge v \notin V(M) \mid v \in V(M_{i',j_{i'}}) \wedge e \in M_{i'',j_{i''}}] = \frac{\binom{|L|-2}{\frac{2}{3}|L|-1}}{\binom{|L|}{\frac{2}{3}|L|}} = \frac{(n-c_o)\cdot\frac{1}{2}(n-c_o)}{\frac{3}{2}(n-c_o)\cdot\left(\frac{3}{2}(n-c_o)-1\right)} \geq \frac{2}{9}$. Thus, $\Pr[e \in M \wedge v \notin V(M)] \geq \frac{2}{9} \cdot \frac{1}{2} = \frac{1}{9}$.

Case 2 *$n_{i'}$ is even but $n_{i''}$ is odd*. In this case, $\Pr[v \in V(M_{i',j_{i'}})] = 1$ and $\Pr[e \in M_{i'',j_{i''}}] = \frac{1}{2} \cdot \frac{n_{i''}-1}{n_{i''}} = \frac{n_{i''}-1}{2n_{i''}}$. Moreover, $\Pr[e = e'_{i''} \mid e \in M_{i'',j_{i''}}] = \frac{2}{n_{i''}-1}$, $\Pr[e = e'_{i''}] = \frac{1}{n_{i''}}$, and $\Pr[e \in M_{i'',j_{i''}} \setminus \{e'_{i''}\}] = \frac{n_{i''}-1}{2n_{i''}} \cdot \left(1 - \frac{2}{n_{i''}-1}\right) = \frac{n_{i''}-3}{2n_{i''}}$. Furthermore, $\Pr[v \notin V(M) \mid e = e'_{i''}] = \frac{1}{3}$ by Lemma 5, and $\Pr[v \notin V(M) \wedge e \in M \mid e \in M_{i'',j_{i''}} \setminus \{e'_{i''}\}] = \frac{\binom{|L|-2}{\frac{2}{3}|L|-1}}{\binom{|L|}{\frac{2}{3}|L|}} = \frac{(n-c_o)\cdot\frac{1}{2}(n-c_o)}{\frac{3}{2}(n-c_o)\cdot\left(\frac{3}{2}(n-c_o)-1\right)} \geq \frac{2}{9}$. Thus, $\Pr[e \in M \wedge v \notin V(M)] \geq \frac{1}{3} \cdot \frac{1}{n_{i''}} + \frac{2}{9} \cdot \frac{n_{i''}-3}{2n_{i''}} = \frac{1}{9}$.

Case 3 *$n_{i'}$ is odd but $n_{i''}$ is even*. In this case, $\Pr[v \in V(M_{i',j_{i'}})] = \frac{2}{n_{i'}} \cdot \frac{1}{2} + \left(1 - \frac{2}{n_{i'}}\right) \cdot 1 = \frac{n_{i'}-1}{n_{i'}}$ and $\Pr[e \in M_{i'',j_{i''}}] = \frac{1}{2}$. So, $\Pr[v \in V(M_{i',j_{i'}}) \wedge e \in M_{i'',j_{i''}}] = \frac{n_{i'}-1}{2n_{i'}}$ and $\Pr[v \notin V(M_{i',j_{i'}}) \wedge e \in M_{i'',j_{i''}}] = \frac{1}{2n_{i'}}$. Moreover, by Lemma 3, $\Pr[e \in M \wedge v \notin V(M) \mid v \in V(M_{i',j_{i'}}) \wedge e \in M_{i'',j_{i''}}] = \frac{\binom{|L|-2}{\frac{2}{3}|L|-1}}{\binom{|L|}{\frac{2}{3}|L|}} = \frac{(n-c_o)\cdot\frac{1}{2}(n-c_o)}{\frac{3}{2}(n-c_o)\cdot\left(\frac{3}{2}(n-c_o)-1\right)} \geq \frac{2}{9}$ and

$\Pr[e \in M \wedge v \notin V(M) \mid v \notin V(M_{i',j_{i'}}) \wedge e \in M_{i'',j_{i''}}] = \frac{2}{3}$. Thus, $\Pr[e \in M \wedge v \notin V(M)] \geq \frac{n_{i'}-1}{2n_{i'}} \cdot \frac{2}{9} + \frac{1}{2n_{i'}} \cdot \frac{2}{3} \geq \frac{1}{9}$.

Case 4 Both $n_{i'}$ *and* $n_{i''}$ *are odd.* In this case, $\Pr[v \in V(M_{i',j_{i'}})] = \frac{n_{i'}-1}{n_{i'}}$ and $\Pr[e \in M_{i'',j_{i''}}] = \frac{1}{2} \cdot \frac{n_{i''}-1}{n_{i''}} = \frac{n_{i''}-1}{2n_{i''}}$. Moreover, $\Pr[e = e'_{i''} \mid e \in M_{i'',j_{i''}}] = \frac{2}{n_{i''}-1}$, $\Pr[e = e'_{i''}] = \frac{1}{n_{i''}}$, and $\Pr[e \in M_{i'',j_{i''}} \setminus \{e'_{i''}\}] = \frac{n_{i''}-1}{2n_{i''}} \cdot \left(1 - \frac{2}{n_{i''}-1}\right) = \frac{n_{i''}-3}{2n_{i''}}$. So, $\Pr[v \notin V(M_{i',j_{i'}}) \wedge e = e'_{i'}] = \frac{1}{n_{i'}n_{i''}}$, $\Pr[v \notin V(M_{i',j_{i'}}) \wedge e \in M_{i'',j_{i''}} \setminus \{e'_{i''}\}] = \frac{n_{i''}-3}{2n_{i'}n_{i''}}$, $\Pr[v \in V(M_{i',j_{i'}}) \wedge e = e'_{i'}] = \frac{n_{i'}-1}{n_{i'}n_{i''}}$, $\Pr[v \in V(M_{i',j_{i'}}) \wedge e \in M_{i'',j_{i''}} \setminus \{e'_{i''}\}] = \frac{(n_{i'}-1)(n_{i''}-3)}{2n_{i'}n_{i''}}$. Obviously, $\Pr[e \in M \wedge v \notin V(M) \mid v \in V(M_{i',j_{i'}}) \wedge e = e'_{i''}] = \frac{1}{3}$, $\Pr[e \in M \wedge v \notin V(M) \mid v \notin V(M_{i',j_{i'}}) \wedge e = e'_{i''}] = 1$, and $\Pr[e \in M \wedge v \notin V(M) \mid v \notin V(M_{i',j_{i'}}) \wedge e \in M_{i'',j_{i''}} \setminus \{e'_{i''}\}] = \frac{2}{3}$. Furthermore, by Lemma 3, $\Pr[e \in M \wedge v \notin V(M) \mid v \in V(M_{i',j_{i'}}) \wedge e \in M_{i'',j_{i''}} \setminus \{e'_{i''}\}] = \frac{\binom{|L|-2}{\frac{2}{3}|L|-1}}{\binom{|L|}{\frac{2}{3}|L|}} = \frac{(n-c_o)\cdot\frac{1}{2}(n-c_o)}{\frac{3}{2}(n-c_o)\cdot\left(\frac{3}{2}(n-c_o)-1\right)} \geq \frac{2}{9}$.
Thus, $\Pr[e \in M \wedge v \notin V(M)] \geq \frac{1}{3} \cdot \frac{n_{i'}-1}{n_{i'}n_{i''}} + 1 \cdot \frac{1}{n_{i'}n_{i''}} + \frac{2}{3} \cdot \frac{n_{i''}-3}{2n_{i'}n_{i''}} + \frac{2}{9} \cdot \frac{(n_{i'}-1)(n_{i''}-3)}{2n_{i'}n_{i''}} \geq \frac{1}{9}$. □

2.4 Computing T_3

Fix a constant τ with $0 < \tau < 1$. A *good triplet* is a triplet $(x, y; z)$, where $\{x, y\}$ is an edge of some cycle C_i in $\mathcal{C}$ and z is a vertex of some other cycle C_j in $\mathcal{C}$ with $i \neq j$ such that $w(x, y) \leq (1 - \tau) \cdot (w(x, z) + w(y, z))$.

To compute T_3, we initialize $T_3 = \emptyset$ and proceed as follows. One sees that the total running time for computing T_3 is in $O(n^3)$, dominated by computing a maximum-weight matching.

1. Construct an auxiliary edge-weighted and edge-labeled multi-digraph H_3 as follows. The vertex set of H_3 is $V(G)$. For each good triplet $(x, y; z)$, H_3 contains the two arcs (z, x) and (z, y), each of these two arcs has a weight of $w(x, z) + w(y, z)$ in H_3, the label of (z, x) is y, and the label of (z, y) is x.
2. Compute a maximum-weight matching M_3 in H_3 (by ignoring the direction of each arc).
3. Compute a random matching M in $\mathcal{C}$ as in Sect. 2.3.
4. Let N_3 be the set of all arcs $(z, x) \in M_3$ such that $z \notin V(M)$ and $\{x, y\} \in M$, where y is the label of (z, x).
(*Comment:* Since both M and N_3 are matchings, no two arcs in N_3 can share a label. Moreover, the endpoints of each edge $e \in M$ can be the heads of at most two arcs in N_3 because e has only two endpoints and N_3 is a matching.)
5. Initialize $N'_3 = N_3$. For every two arcs (z, x) and (z', y) in N'_3 such that $\{x, y\} \in M$, select one of (z, x) and (z', y) uniformly at random and delete it from N'_3.
6. For each $(z, x) \in N'_3$, let T_3 include the triangle t with $V(t) = \{x, y, z\}$, where y is the label of (z, x).

(*Comment:* By Step 5 and the comment on Step 4, the triangles included in T_3 in this step are vertex-disjoint.)

7. Let M' be the set of edges (x, y) in M such that neither x nor y is the head or the label of an arc in N_3'. Further let Z be the set of vertices z in G such that $z \notin V(M)$ and z is not the tail of an edge in N_3'.
(*Comment:* Since $|M| = n$ by Lemma 4, the comment on Step 6 implies $|Z| = |M'|$.)
8. Select an arbitrary one-to-one correspondence between the edges in M' and the vertices in Z. For each $z \in Z$ and its corresponding edge (x, y) in M', let T_3 include the triangle t with $V(t) = \{x, y, z\}$.

We classify the external balanced triangles in B into two types as follows. An external balanced triangle t in B is of *Type 1* if for every vertex v of t, the weight of each edge incident to v in $\mathcal{C}$ is at least $\frac{1}{2}(1 - \frac{1}{2}\delta)(1 - \tau)w(t)$; otherwise, t is of *Type 2*. We use B_1^e and B_2^e to denote these two types of external balanced triangles in B, respectively.

Similarly, we classify the partially internal balanced triangles in B into two types as follows. A partially internal balanced triangle t in B is of *Type 1* if the weight of each edge incident to the external vertex of t in $\mathcal{C}$ is at least $\frac{1}{2}(1 - \frac{1}{2}\delta)(1 - \tau)w(t)$; otherwise, t is of *Type 2*. We use B_1^p and B_2^p to denote these two types of partially internal balanced triangles in B, respectively.

Lemma 8 $w(T_1) \geq \frac{2}{3}w(B) + \frac{2-3\delta-6\tau+3\delta\tau}{54}w(B_1^e) + \frac{2-3\delta-6\tau+3\delta\tau}{162}w(B_1^p)$.

Proof For the analysis, we use the triangles in $B_1^e \cup B_1^p$ to construct a random matching N in $\mathcal{C}$ as follows.

1. Initialize $N' = \emptyset$. For each triangle t in B, select one edge e_t of t uniformly at random and add it to N'.
2. For each triangle t in B_1^e, choose one neighbor v_t' of v_t in $\mathcal{C}$ uniformly at random, where v_t is the vertex of t not incident to e_t.
3. For each triangle t in B_1^p such that e_t is internal, choose one neighbor v_t' of v_t in $\mathcal{C}$ uniformly at random, where v_t is the external vertex of t.
4. Initialize $X = \emptyset$. For each $t \in B_1^e \cup B_1^p$, if $v_t' \notin V(N')$, then add the (ordered) pair (v_t, v_t') to X.
(*Comment:* Suppose that t_1 and t_2 are different triangles in $B_1^e \cup B_1^p$ with $\{v_{t_1}', v_{t_2}'\} \cap V(N') = \emptyset$. Then, it holds that $(v_{t_1}, v_{t_1}') \neq (v_{t_2}, v_{t_2}')$ because $v_{t_1} \neq v_{t_2}$. However, it is possible that $(v_{t_1}, v_{t_1}') = (v_{t_2}', v_{t_2})$ or $(v_{t_1}', v_{t_1}) = (v_{t_2}, v_{t_2}')$.)
5. Let D be the digraph with vertex set $V(G) \setminus V(N')$ and arc set X. Partition X into three matchings X_1, X_2, X_3 in D.
(*Comment:* We will later show that this step can be done.)
6. Select a set Y among X_1, X_2, X_3 uniformly at random.
7. Initialize $N = \{e_t \mid t \in B \setminus (B_1^e \cup B_1^p)\}$. For each $t \in B_1^e$, if $(v_t, v_t') \notin Y$, then add e_t to N; otherwise add $\{v_t, v_t'\}$ to N. Similarly, for each $t \in B_1^p$, if e_t is external or $(v_t, v_t') \notin Y$, then add e_t to N; otherwise add $\{v_t, v_t'\}$ to N.

In this paragraph, we show that Step 5 can be done. By the comment on Step 4, we see that for each vertex v in D, there is at most one arc leaving v in D. Moreover, since

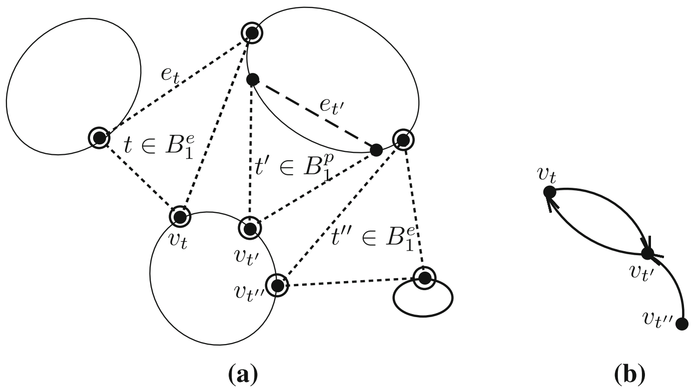

Fig. 2 Triangles $t, t'' \in B_1^e$ and $t' \in B_1^p$ for the proof of Lemma 8, and the subgraph D induced by $\{v_t, v_{t'}, v_{t''}\}$ when $v'_t = v_{t'}$, $v'_{t'} = v_t$ and $v'_{t''} = v_{t'}$

v is incident to only two edges in $\mathcal{C}$, there are at most two arcs entering v in D (see Fig. 2b for an illustration). Thus, if we ignore the direction of each edge in D, then we obtain an undirected multigraph G_D in which each vertex is incident to at most three edges and there are at most two parallel edges between each pair of vertices. Let C be a connected component of G_D, and C' be the simple graph obtained from C by deleting exactly one edge from each pair of parallel edges. If C has no parallel edges, then C is a subgraph of a cycle in $\mathcal{C}$ and in turn its edges can be trivially partitioned into three disjoint matchings; otherwise, we can claim that C' is not a cycle. By the claim, C' is a collection of vertex-disjoint paths; this together with the fact that each vertex is incident to at most three edges in C implies that the edges of C can be trivially partitioned into three disjoint matchings as well. To see the claim, we assume, on the contrary, that C has at least one pair of parallel edges and C' is a cycle. Recall that each edge of C' has a direction in D. If we restore the directions of the edges in C', then we must obtain a directed cycle C'' because C' is a cycle and there is at most one arc leaving each vertex of C' in D. Now, since there is already one arc leaving each vertex in C'', we have no way to restore the direction of each arc in $C \setminus C'$ without violating the condition that there is at most one arc leaving each vertex of C in D.

We next analyze $\mathcal{E}[w(N)]$. For each triangle $t \in B_1^e$ (see Fig. 2a for illustrations of the triangles), let E_t be the set of edges e in $\mathcal{C}$ such that e is incident to a vertex of t. Similarly, for each triangle $t \in B_1^p$, let E_t be the set of edges e in $\mathcal{C}$ such that e is incident to the external vertex of t. Consider a $t \in B_1^e \cup B_1^p$ and an $e = \{x, y\} \in E_t$ with $x \in V(t)$. Since v_t takes on any of the vertices of t with equal probability, $\Pr[x = v_t] = \frac{1}{3}$. Similarly, since v'_t takes on any of the two neighbors of v_t in $\mathcal{C}$ with equal probability, $\Pr[y = v'_t \mid x = v_t] = \frac{1}{2}$. Hence, $\Pr[\{v_t, v'_t\} = e] = \frac{1}{6}$. Moreover, $\Pr[v'_t \notin V(N')] = \frac{1}{3}$ because v'_t appears in a triangle t' in B and $v_{t'}$ takes on any of the vertices in t' with equal probability. Thus, $\Pr[\{v_t, v'_t\} = e \wedge v'_t \notin V(N')] = \frac{1}{6} \cdot \frac{1}{3} = \frac{1}{18}$. Furthermore, $\Pr[e \in N \mid \{v_t, v'_t\} = e \wedge v'_t \notin V(N')] = \Pr[e \in Y \mid \{v_t, v'_t\} = e \wedge v'_t \notin V(N')] = \frac{1}{3}$. So, $\Pr[e \in N] = \frac{1}{3} \cdot \frac{1}{18} = \frac{1}{54}$. Now, if $t \in B_1^e$, then $|E_t| = 6$ and in turn

$\Pr[e_t \notin N] = 6 \cdot \frac{1}{54} = \frac{1}{9}$. On the other hand, if $t \in B_1^p$, then $|E_t| = 2$ and in turn $\Pr[e_t \notin N] = 2 \cdot \frac{1}{54} = \frac{1}{27}$.

By the discussions in the last paragraph, $\mathcal{E}[w(N)] \geq \frac{1}{3}\sum_{t \in B \setminus (B_1^e \cup B_1^p)} w(t) + \frac{8}{9} \cdot \frac{1}{3}\sum_{t \in B_1^e} w(t) + \frac{1}{9} \cdot \frac{1}{2}(1 - \frac{1}{2}\delta)(1-\tau)\sum_{t \in B_1^e} w(t) + \frac{26}{27} \cdot \frac{1}{3}\sum_{t \in B_1^p} w(t) + \frac{1}{27} \cdot \frac{1}{2}(1 - \frac{1}{2}\delta)(1-\tau)\sum_{t \in B_1^p} w(t) = \frac{1}{3}w(B) + \frac{2-3\delta-6\tau+3\delta\tau}{108}w(B_1^e) + \frac{2-3\delta-6\tau+3\delta\tau}{324}w(B_1^p)$. So, $w(T_1) \geq 2 \cdot \mathcal{E}[w(N)] \geq \frac{2}{3}w(B) + \frac{2-3\delta-6\tau+3\delta\tau}{54}w(B_1^e) + \frac{2-3\delta-6\tau+3\delta\tau}{162}w(B_1^p)$. □

Lemma 9 *Let t be a balanced triangle in B, and e_1 and e_2 be any two edges in t. Then we have* $\frac{w(e_1)+0.5w(e_2)}{w(t)} \geq \frac{3(1-\delta)}{6-4\delta}$.

Proof Let e_3 be the edge in t other than e_1 and e_2. Since $w(t)$ is independent of the choice of e_1 and e_2, one can easily see that in order to prove the lemma, it suffices to consider the case where e_1 is the lightest edge and e_2 is the second lightest edge in T. So, we may assume $w(e_1) \leq w(e_2) \leq w(e_3)$. Since B is balanced, $w(e_1) \geq (1-\delta)w(e_3)$. An easy inspection shows that the ratio $\frac{w(e_1)+0.5w(e_2)}{w(t)}$ is minimized when $w(e_1) = w(e_2) = (1-\delta)w(e_3)$. Thus, the ratio is at least $\frac{1.5(1-\delta)}{1+2(1-\delta)} = \frac{3(1-\delta)}{6-4\delta}$. □

Lemma 10 $\mathcal{E}[w(T_3)] \geq \frac{2(1-\epsilon)}{3}w(B) + \frac{(1-\delta)\tau}{36-24\delta} \cdot w(B_2^e) + \frac{(1-\delta)\tau}{36-24\delta} \cdot w(B_2^p)$.

Proof For a set F of edges in H_3, let $\tilde{w}(F)$ denote the total weight of edges of F in H_3. Further let W_2 be the total weight of triangles in $B_2^e \cup B_2^p$.

Consider an arbitrary $t \in B_2^e \cup B_2^p$ (see Fig. 3 for illustrations of the triangles). Since t is of Type 2, t has a vertex v_t such that some neighbor v_t' of v_t in $\mathcal{C}$ satisfies $w(v_t, v_t') < \frac{1}{2}(1 - \frac{1}{2}\delta)(1-\tau)w(t)$. Let z_t and z_t' be the vertices in $V(t) \setminus \{v_t\}$. By the triangle inequality, $w(z_t, v_t') \geq \frac{1}{2}w(z_t, z_t')$ or $w(z_t', v_t') \geq \frac{1}{2}w(z_t, z_t')$. Without loss of generality, we may assume that $w(z_t, v_t') \geq \frac{1}{2}w(z_t, z_t')$. We claim that $(v_t, v_t'; z_t)$ is a good triplet. To see this, first recall that $(1-\delta)w(z_t', v_t) \leq w(z_t, v_t)$ because t is balanced. Moreover, by the triangle inequality, $\frac{1}{2}\delta w(z_t', v_t) \leq$

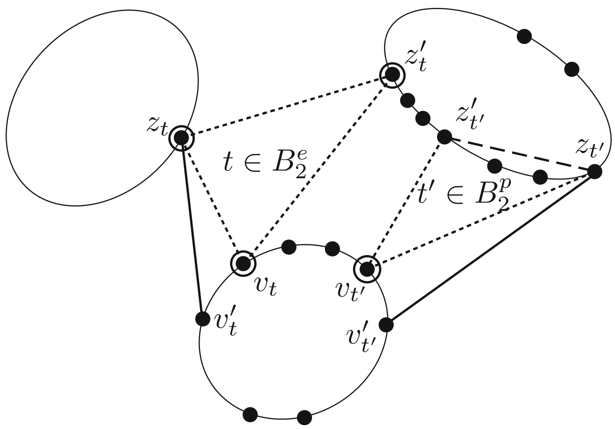

Fig. 3 Triangles $t \in B_2^e$ and $t' \in B_2^p$ for the proof of Lemma 10

$\frac{1}{2}\delta w(z_t, v_t) + \frac{1}{2}\delta w(z_t, z_t')$. So, $(1 - \frac{1}{2}\delta)w(z_t', v_t) \le (1 + \frac{1}{2}\delta)w(z_t, v_t) + \frac{1}{2}\delta w(z_t, z_t')$. Thus, $(1 - \frac{1}{2}\delta)\left(w(z_t, v_t) + w(z_t, z_t') + w(z_t', v_t)\right) \le 2w(z_t, v_t) + w(z_t, z_t') \le 2w(z_t, v_t) + 2w(z_t, v_t')$. Hence, $\frac{1}{2}(1 - \frac{1}{2}\delta)w(t) \le w(z_t, v_t) + w(z_t, v_t')$. Therefore, $w(v_t, v_t') < \frac{1}{2}(1 - \frac{1}{2}\delta)(1 - \tau)w(t) \le (1 - \tau)\left(w(z_t, v_t) + w(z_t, v_t')\right)$. Consequently, the claim holds.

By the claim in the last paragraph, the set X of all $\{z_t, v_t\}$ with $t \in B_2^e \cup B_2^p$ is a matching in H_3. Moreover, $\tilde{w}(M_3) \ge \tilde{w}(X) = \sum_{t \in B_2^e \cup B_2^p} \tilde{w}(z_t, v_t) \ge \frac{3(1-\delta)}{6-4\delta} \sum_{t \in B_2^e \cup B_2^p} w(t) = \frac{3(1-\delta)}{6-4\delta} W_2$, where the second inequality holds by Lemma 9. Now, by Lemma 7, $\mathcal{E}[\tilde{w}(N_3)] \ge \frac{1}{9}\tilde{w}(M_3) \ge \frac{1-\delta}{18-12\delta} W_2$ and in turn $\mathcal{E}[\tilde{w}(N_3')] \ge \frac{1-\delta}{36-24\delta} W_2$. Obviously, $w(T_3) \ge 2w(M') + w(M \setminus M') + \tilde{w}(N_3') \ge 2w(M) + \tau \cdot \tilde{w}(N_3')$, where the first inequality holds by the triangle inequality and the second inequality holds because each edge in N_3' corresponds to a good triplet. Therefore, by Lemma 6, $\mathcal{E}[w(T_3)] \ge \frac{2}{3} \cdot w(\mathcal{C}) + \frac{(1-\delta)\tau}{36-24\delta} W_2 \ge \frac{2(1-\epsilon)}{3} \cdot w(B) + \frac{(1-\delta)\tau}{36-24\delta} W_2$. □

2.5 Analyzing the approximation ratio

Let B^i be the set of completely internal balanced triangles in B, and let $\alpha_1 = \frac{w(B^i)}{w(B)}$. Recall that B_1^e (B_2^e, respectively) is the set of Type 1 (Type 2, respectively) external balanced triangles in B, and B_1^p (B_2^p, respectively) is the set of Type 1 (Type 2, respectively) partially internal balanced triangles in B. For convenience, let $\alpha_2 = \frac{w(B_1^e)}{w(B)}$, $\alpha_3 = \frac{w(B_2^e)}{w(B)}$, $\alpha_4 = \frac{w(B_1^p)}{w(B)}$, and $\alpha_5 = \frac{w(B_2^p)}{w(B)}$. Recall from Lemma 1 that $B_{\bar{b}}$ is the set of unbalanced triangles in B and $\gamma = \frac{w(B_{\bar{b}})}{w(B)}$. Therefore, $\gamma + \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 = 1$.

Suppose that we have fixed δ and τ to certain constants, respectively. Then, to use Lemmas 1, 2, 8 and 10 to obtain the best lower bound on the approximation ratio achieved by our algorithm, it suffices to solve the following linear program (denoted by $\text{LP}_{\delta,\tau}$):

$$\begin{aligned}
\text{Minimize} \quad & b; \\
\text{Subject to} \quad & b \ge \frac{2}{3} + \frac{2\delta}{9 - 3\delta}\gamma, \\
& b \ge \alpha_1 + \frac{2}{3}\alpha_4 + \frac{2}{3}\alpha_5, \\
& b \ge \frac{2}{3} + \frac{2 - 3\delta - 6\tau + 3\delta\tau}{54}\alpha_2 + \frac{2 - 3\delta - 6\tau + 3\delta\tau}{162}\alpha_4, \\
& b \ge \frac{2}{3} + \frac{(1-\delta)\tau}{36 - 24\delta}\alpha_3 + \frac{(1-\delta)\tau}{36 - 24\delta}\alpha_5, \\
& \gamma + \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 = 1, \\
& \gamma, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5 \ge 0.
\end{aligned}$$

For each pair (δ, τ) with $0 \le \delta \le 1$ and $0 \le \tau \le 1$, let $b_{\delta,\tau}$ be the optimal value of the objective function of $\text{LP}_{\delta,\tau}$. Since we can freely choose δ and τ, we can find the pair (δ, τ) by a (100×100)-grid search such that $b_{\delta,\tau}$ is maximized among the pairs (δ, τ) with $\delta \in \{0.01 \cdot k \mid k = 0, 1, \ldots, 100\}$ and $\tau \in \{0.01 \cdot \ell \mid \ell = 0, 1, \ldots, 100\}$. It turns out the best $b_{\delta,\tau}$ is at least 0.66768. So, we can conclude that the expected approximation ratio achieved by our randomized approximation algorithm is at least $0.66768 - \epsilon$.

The discussion in the last paragraph may not look rigorous. So, we next rigorously prove that the expected approximation ratio achieved by our randomized approximation algorithm is at least $0.66768 - \epsilon$. We choose $\delta = 0.1$ and $\tau = 0.2$. Then, by Lemmas 1, 2, 8 and 10 , we have the following inequalities:

$$\frac{w(T_1)}{w(B)} \geq \frac{2}{3} + \frac{0.2}{8.7}\gamma \tag{1}$$

$$\frac{w(T_2)}{w(B)} \geq \alpha_1 + \frac{2}{3}\alpha_4 + \frac{2}{3}\alpha_5 \tag{2}$$

$$\frac{w(T_1)}{w(B)} \geq \frac{2}{3} + \frac{0.56}{54}\alpha_2 + \frac{0.56}{162}\alpha_4 \tag{3}$$

$$\frac{\mathcal{E}[w(T_3)]}{w(B)} \geq \frac{2(1-\epsilon)}{3} + \frac{0.18}{33.6}\alpha_3 + \frac{0.18}{33.6}\alpha_5. \tag{4}$$

Suppose that we multiply both sides of Inequalities (1), (2), (3) and (4) by 0.1327, 0.00305, 0.2943 and 0.5698, respectively. Then, one can easily verify that the summation of the left-hand sides of the resulting inequalities is

$$0.1327 \cdot \frac{w(T_1)}{w(B)} + 0.00305 \cdot \frac{w(T_2)}{w(B)} + 0.2943 \cdot \frac{w(T_1)}{w(B)} + 0.5698 \cdot \frac{\mathcal{E}[w(T_3)]}{w(B)},$$

while the summation of the right-hand sides is at least

$$\frac{1.9936}{3} - \frac{1.1396}{3}\epsilon + 0.00305(\gamma + \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5).$$

Now, using $\gamma + \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 = 1$, we finally have

$$(0.1327 + 0.00305 + 0.2943 + 0.5698) \cdot \max\left\{\frac{w(T_1)}{w(B)}, \frac{w(T_2)}{w(B)}, \frac{\mathcal{E}[w(T_3)]}{w(B)}\right\} \geq \frac{2.00275}{3} - \frac{1.1396}{3}\epsilon,$$

which can be simplified as

$$\max\{w(T_1), w(T_2), \mathcal{E}[w(T_3)]\} \geq (0.66768 - 0.38\epsilon) \cdot w(B).$$

In summary, we have proven the following theorem, stating that the MMWTP problem admits a better approximation algorithm than the trivial $\frac{2}{3}$-approximation if ϵ is sufficiently small. Note that each of the three triangle packings T_1, T_2 and T_3 is computed in $O(n^3)$ time.

Theorem 1 *For any constant* $\epsilon > 0$*, the expected approximation ratio achieved by our* $O(n^3)$*-time randomized approximation algorithm is at least* $0.66768 - \epsilon$.

3 Conclusions

We studied the maximum-weight triangle packing problem on an edge-weighted complete graph G, in which the edge weights satisfy the triangle inequality. Although the non-metric variant has been extensively studied in the literature, it is surprising that prior to our work, no nontrivial approximation algorithm had been designed and analyzed for this common metric case. We designed the first nontrivial cubic-time approximation algorithm for MMWTP, which is randomized and achieves an expected approximation ratio of $0.66768 - \epsilon$ for any positive constant $\epsilon > 0$. This improves the almost trivial deterministic $\frac{2}{3}$-approximation. It seems that completely new ideas are needed to improve our approximation ratio.

Acknowledgements YC and AZ are supported by the NSFC Grants 11971139, 11771114 and 11571252; and supported by the CSC Grants 201508330054 and 201908330090, respectively. ZZC is supported by the Grant-in-Aid for Scientific Research of the Ministry of Education, Science, Sports and Culture of Japan, under Grant No. 18K11183. GL is supported by the NSERC Canada. LW is supported by a Grant for Hong Kong Special Administrative Region, China (CityU 11210119).

References

Arkin EM, Hassin R (1998) On local search for weighted packing problems. Math Oper Res 23:640–648
Berman P (2000) A $d/2$ approximation for maximum weight independent set in d-claw free graphs. In: Proceedings of the 7th scandinavian workshop on algebraic theory (SWAT'00), LNCS 1851, p 214–219,
Chen Z-Z, Tanahashi R, Wang L (2009) An improved randomized approximation algorithm for maximum triangle packing. Discrete Appl Math 157:1640–1646
Chen Z-Z, Tanahashi R, Wang L (2010) Erratum to an improved randomized approximation algorithm for maximum triangle packing. Discrete Appl Math 158:1045–1047
Chlebík M, Chlebíková J (2003) Approximation hardness for small occurrence instances of NP-hard problems. Proc ISAAC 2003:152–164
Cygan M (2013) Improved approximation for 3-dimensional matching via bounded pathwidth local search. Proc FOCS 2013:509–518
Fürer M, Yu H (2014) Approximating the k-set packing problem by local improvements. Proc ISCO 2014:408–420
Gabow HN (1976) An efficient implementation of Edmonds' algorithm for maximum matching on graphs. J ACM 23:221–234
Garey MR, Johnson DS (1979) Computers and intractability: a guide to the theory of NP-completeness. W. H. Freeman and Company, San Francisco
Guruswami V, Rangan CP, Chang MS, Chang GJ, Wong CK (1998) The vertex disjoint triangles problem. Proc WG 1998:26–37
Halldórsson MM (1995) Approximating discrete collections via local improvement. In: ACM-SIAM Proceedings of the 6th annual symposium on discrete algorithms (SODA'95), pp 160–169
Hartvigsen D (1984) Extensions of matching theory. PhD thesis, Department of Mathematics, Carnegie-Mellon University
Hassin R, Rubinstein S (2006a) An approximation algorithm for maximum triangle packing. Discrete Appl Math 154:971–979
Hassin R, Rubinstein S (2006b) Erratum to an approximation algorithm for maximum triangle packing. Discrete Appl Math 154:2620

Hurkens CAJ, Schrijver A (1989) On the size of systems of sets every t of which have an SDR, with an application to the worst-case ratio of heuristics for packing problems. SIAM J Discrete Math 2:68–72
Kann V (1991) Maximum bounded 3-dimensional matching is MAX SNP-complete. Inf Process Lett 37:27–35
Manic G, Wakabayashi Y (2008) Packing triangles in low degree graphs and indifference graphs. Discrete Math 308:1455–1471
van Rooij JMM, van Kooten Niekerk ME, Bodlaender HL (2013) Partition into triangles on bounded degree graphs. Theory Comput Syst 52:687–718
van Zuylen A (2013) Deterministic approximation algorithms for the maximum traveling salesman and maximum triangle packing problems. Discrete Appl Math 161:2142–2157

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.