Improved Approximation Algorithms for the Traveling Tournament Problem

Jingyang Zhao
University of Electronic Science and Technology of China, Chengdu, China

Mingyu Xiao[1]
University of Electronic Science and Technology of China, Chengdu, China

Chao Xu
University of Electronic Science and Technology of China, Chengdu, China

Abstract

The Traveling Tournament Problem (TTP) is a well-known benchmark problem in the field of tournament timetabling, which asks us to design a double round-robin schedule such that each pair of teams plays one game in each other's home venue, minimizing the total distance traveled by all n teams (n is even). TTP-k is the problem with one more constraint that each team can have at most k consecutive home games or away games. The case where $k = 3$, TTP-3, is one of the most investigated cases. In this paper, we improve the approximation ratio of TTP-3 from $(1.667 + \varepsilon)$ to $(1.598 + \varepsilon)$, for any $\varepsilon > 0$. Previous schedules were constructed based on a Hamiltonian cycle of the graph. We propose a novel construction based on triangle packing. Then, by combining our triangle packing schedule with the Hamiltonian cycle schedule, we obtain the improved approximation ratio. The idea of our construction can also be extended to $k \geq 4$. We demonstrate that the approximation ratio of TTP-4 can be improved from $(1.750+\varepsilon)$ to $(1.700+\varepsilon)$ by the same method. As an additional product, we also improve the approximation ratio of LDTTP-3 (TTP-3 where all teams are allocated on a straight line) from $4/3$ to $(6/5 + \varepsilon)$.

2012 ACM Subject Classification Theory of computation → Approximation algorithms analysis; Theory of computation → Scheduling algorithms

Keywords and phrases Sports scheduling, Traveling Tournament Problem, Approximation algorithm

Digital Object Identifier 10.4230/LIPIcs.MFCS.2022.83

Funding The work is supported by the National Natural Science Foundation of China, under grant 61972070.

1 Introduction

In the field of the tournament schedule, the traveling tournament problem (TTP) is a widely known benchmark problem that was first systematically introduced in [10]. This problem aims to find a double round-robin tournament satisfying some constraints, minimizing the total distance traveled by all participant teams. In a double round-robin tournament of n teams, each team will play 2 games against each of the other $n - 1$ teams, one at its home venue and one at its opponent's home venue. Additionally, each team should play one game a day, all games need to be scheduled on $2(n - 1)$ consecutive days, and so there are exactly $n/2$ games on each day. According to the definition, we know that n is always even. For TTP, we have the following two basic constraints or assumptions on the double round-robin tournament.

[1] Corresponding author

47th International Symposium on Mathematical Foundations of Computer Science (MFCS 2022).
Editors: Stefan Szeider, Robert Ganian, and Alexandra Silva; Article No. 83; pp. 83:1–83:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Traveling Tournament Problem (TTP).

- *No-repeat*: Two teams cannot play against each other on two consecutive days.
- *Direct-traveling*: Before the first game, all teams are at home, and they will return home after the last game. Furthermore, a team travels directly from its game venue on ith day to its game venue on $(i+1)$th day.

TTP-k, a well-known variant of TTP, is to add the following constraint on the maximum number of consecutive home games and away games.

- *Bounded-by-k*: Each team can have at most k consecutive home games or away games.

The smaller the value of k, the more frequently a team has to return home. By contrast, if k is very large, say $k = n-1$, then this constraint loses meaning, and TTP-k becomes TTP where a team can schedule their travel distance as short as that in the traveling salesman problem (TSP).

A weight function w on the complete undirected graph is called a *metric* if it satisfies the symmetry and triangle inequality properties: $w(a,b) = w(b,a)$ and $w(a,c) \leq w(a,b) + w(b,c)$ for all $a, b, c \in V$.

The input of TTP-k contains a complete graph where each vertex is a team, and the weight between two vertices u and v is the distance from the home of team u to the home of team v. In this paper, we only consider the case when the weight function w is a metric. Due to page limitations, the proofs of some lemmas and theorems marked with '*' may be omitted or incomplete. The full proofs can be found in the full version of this paper.

1.1 Related Work

TTP and TTP-k are difficult optimization problems. The NP-hardness of TTP and TTP-k with $k \geq 3$ has been established [2, 24, 5]. Although the hardness of TTP-2 is still not formally proved, it is believed that TTP-2 is also hard since it is also not easy to construct a feasible solution to it. In the literature there is a large number of contributions on approximation algorithms [25, 28, 31, 30, 6, 21, 29, 18, 27, 16, 17] and heuristic algorithms [11, 20, 1, 9, 13, 14].

For heuristic algorithms, most known works are concerned with the case of $k = 3$. Since the search spaces of TTP and TTP-k are usually very large, many instances of TTP-3 with more than 10 teams in the online benchmark [26, 3] have not been completely solved even by using high-performance machines.

In terms of approximation algorithms, most results are based on the assumption that the distance holds the symmetry and triangle inequality properties. This is natural and practical in the sports schedule. For $k = 2$, one significant contribution to TTP-2 was done by Thielen and Westphal [25]. They proposed a $(1 + 16/n)$-approximation algorithm for $n/2$ being even and a $(3/2 + O(1/n))$-approximation algorithm for $n/2$ being odd. Currently, approximation ratios for these two cases have been improved to $(1 + 3/n)$ and $(1 + 12/n)$, respectively [30, 31]. For $k = 3$, the first approximation algorithm, proposed by Miyashiro *et al.*, admits a $2 + O(1/n)$ approximation ratio using the Modified Circle Method [21]. Then, the approximation ratio was improved to $5/3 + O(1/n)$ by Yamaguchi *et al.* [29]. Their approximation algorithm also works for $3 < k \ll n$. For $k = 4$, the ratio is $1.750 + O(1/n)$ and for $4 < k \ll n$, the ratio is $(5k-7)/(2k) + O(k/n)$ [29]. For $k = \Theta(n)$, Westphal and Noparlik proposed a 5.875 approximation algorithm for any choice of $k \geq 4$ and $n \geq 6$ [27], and Imahori *et al.* gave a 2.75 approximation algorithm for $k = n-1$ where they also proved the approximation ratio can be further improved to 2.25 if the optimal TSP is given [18]. The current best approximation ratio for $k = 3$ is still $5/3 + O(1/n)$ [29].

We refer the readers to [3] for more variants of TTP (including the traveling tournament problem with predefined venues, the time-relaxed traveling tournament problem, etc.) with detailed benchmarks for each.

1.2 Our Results

In this paper, we consider TTP-k with the cases of $k = 3$ and $k = 4$. Our contributions can be summarized as follows.

We mainly focus on TTP-3. Firstly, we analyze the structural properties in more detail, which leads us to strengthen some of the current-known lower bounds and propose several new ones. Secondly, we design an approximation algorithm that improves the approximation ratio from $(5/3 + \varepsilon)$ to $(139/87 + \varepsilon)$. Our algorithm consists of two constructions. The first construction is based on the Hamiltonian cycle, which is a well-known method. The Hamiltonian cycle used is commonly generated by the Christofides-Serdyukov algorithm, while we propose a new Hamiltonian cycle that uses the minimum weight perfect matching. The second construction proposed by us is based on triangle packing which can be seen as a generalization of the matching packing schedule in TTP-2.

For a special case of TTP-3 where all teams are located on a line, Linear Distance TTP-3 (LDTTP-3), we prove that the approximation ratio of our triangle packing construction can achieve $(6/5 + \varepsilon)$, which improves the previous approximation ratio of 4/3 [15].

Finally, we extend our method to TTP-4 and show that we can improve the approximation ratio from $(7/4 + \varepsilon)$ to $(17/10 + \varepsilon)$.

2 Preliminaries

We will always use n to denote the total number of teams in the problem. The set of n teams is denoted by $V = \{t_1, t_2, \ldots, t_n\}$. Recall that n is even. For TTP-3, there are three cases of n we consider: $n \equiv 0 \pmod 6$, $n \equiv 2 \pmod 6$ and $n \equiv 4 \pmod 6$. Due to the different structural properties, these three cases have to be handled separately. We mainly describe the case of $n \equiv 0 \pmod 6$ due to page limitations. So from here on, we assume that n is a number divisible by 6.

We use $G = (V, E)$ to denote the complete graph on the n vertices representing the n teams. There is a positive weight function $w : E \to \mathbb{R}_{\geq 0}$ on the edges of G. We often write $w(u, v)$ to mean the weight of the edge uv, instead of $w(uv)$. Note that $w(u, v)$ would be the same as the distance between the home of team u and the home of team v. For any weight function $w : X \to \mathbb{R}_{\geq 0}$, we extend it to subsets of X. Define $w(Y) = \sum_{x \in Y} w(x)$ for $Y \subseteq X$.

The weight of a minimum weight spanning tree in G is denoted by $\mathrm{MST}(G)$. We use $\delta(u)$ to denote the set of edges incident on u in G. We also use $\deg(u)$ to denote the weighted degree of a vertex. That is the total weight of all edges incident on u in G, i.e., $\deg(u) = w(\delta(u))$. We also let Δ to be the sum of the weighted degrees, i.e., $\Delta = \sum_{u \in V} \deg(u) = 2w(E)$.

A cycle on k vertices is called a k-cycle. A triangle uvw is a 3-cycle on three different vertices $\{u, v, w\}$. Two subgraphs or sets of edges are *vertex-disjoint* if they do not share a common vertex. A *triangle packing* in G, denoted by T, is a set of edges such that each component is a triangle, and all vertices are covered. Equivalently, it can be seen as the edges of m vertex-disjoint triangles. Similarly, a P_3 *path* is a simple path on three different vertices $\{u, v, w\}$, which can be represented by u-v-w. A P_3 *path packing* in G is a set of edges such that every component is a P_3 path, and each vertex is covered. We can obtain a triangle packing from a P_3 packing by connecting the two non-adjacent vertices in each component.

Let $m = \frac{n}{3}$, and then m is an even number. For a fixed triangle packing T of G, let the components be $u_1, \ldots, u_m$. $U = \{u_1, \ldots, u_m\}$ is a partition of V. Each $u \in U$ is referred as a *super-team*. We define a complete graph $H = (U, F)$ on U. We will also have a cost function $c(u, v)$ defined on vertices in U. It is $c(u, v) = \sum_{u' \in u, v' \in v} w(u', v')$ for each edge $uv \in F$. For simplicity we also define $c(u, u) = 0$. Despite not using this property, it is worth noting that the cost function c is also a metric.

We also define $c(u)$ for $u \in U$ to be the same as $w(a, b) + w(b, c) + w(c, a)$ for $a, b, c \in u$. Note that $c(U) = \sum_{u \in U} c(u) = w(T)$. We can easily get

$$\frac{1}{2}\Delta = w(E) = c(F) + c(U) = c(F) + w(T). \tag{1}$$

The remaining parts of the paper are organized as follows. In Section 3, we focus on TTP-3. Specifically, in Section 3.1, we introduce some basic notations and propose several new lower bounds. In Section 3.2, we give a brief introduction to the well-known construction based on the Hamiltonian cycle. In Section 3.3, we propose a novel construction based on triangle packing. Although these two constructions cannot make any improvement separately, in Section 3.4, we show that together they can guarantee an improved approximation ratio for TTP-3. In Section 4, we analyze the approximation ratio of our algorithm for LDTTP-3. In Section 5, we extend our methods and prove that we can also improve the approximation ratio for TTP-4.

3 TTP-3

3.1 The Independent Lower Bounds

For TTP and TTP-k, a well-known method to obtain the lower bounds is to use an independent relaxation [4, 10]. The basic idea is to obtain a lower bound on the traveling distance of each team independently without considering the feasibility of other teams and then sum them together. Although there exist many independent lower bounds for TTP-3 [21, 29, 27], we can not use them directly to get our result. We are interested in a stronger bound. Before we make some observations on the independent lower bounds, we first need to explore some properties of TTP-k.

For TTP-k, each team needs to visit each other team's home once in the tournament. A *road trip* of a team v is a simple cycle starting and ending at v. A *k-itinerary* of v, is a connected subgraph of G that consists of road trips with each simple cycle of length at most $k + 1$, and each vertex other than v in V has degree 2.

For TTP-3, the length of each road trip is at most 4. For simplicity, we omit k when k is implicit. In the remainder of this section, $k = 3$.

An itinerary is *optimal* for a team if it is the itinerary of minimum weight.

Considering an optimal itinerary I_v for team v, we will use $w(I_v)$ to denote the weight of the optimal itinerary for team v. Then, $\psi = \sum_{v \in V} w(I_v)$ is a simple independent lower bound for the minimum weight solution of TTP-3. Note that this lower bound was also used in the experiment [10]. However, it is NP-hard to compute $w(I_v)$. Hence, we want to find an alternative lower bound for each team's optimal itinerary.

When $n \equiv 0 \pmod 6$ and $n \equiv 4 \pmod 6$, we can prove that there always exists an optimal itinerary with no 2-cycles.

▶ **Lemma 1.** *For TTP-3 with the cases of $n \equiv 0 \pmod 3$ and $n \equiv 1 \pmod 3$, there exists an optimal itinerary with no 2-cycles for each team.*

Proof. Consider an optimal itinerary I_v of team v with the minimum number of 2-cycles. Assume I_v contains a 2-cycle. Since all cycles share the vertex v, by the triangle inequality, we can get a 3-cycle by shortcutting two 2-cycles, and the 3-cycle has a weight no greater than the sum weight of the two 2-cycles. This shows that I_v contains exactly one 2-cycle. Similarly, by the triangle inequality, a 4-cycle can be obtained by shortcutting one 2-cycle and one 3-cycle without increasing the total weight. This shows there cannot be any 3-cycle. Thus, there exists only one 2-cycle, and the rest of the cycles are all 4-cycles. Thus, we will get $n \equiv 2 \pmod 3$, a contradiction to $n \equiv 0 \pmod 3$ or $n \equiv 1 \pmod 3$. ◀

3.1.1 Bounds on optimal itinerary

For this subsection, we fix a single team v and start to consider the optimal itinerary I_v for this team. We will write I as I_v to simplify the notation within this subsection. We will give a more refined analysis than previous results [29].

Recall that we consider the case of $n \equiv 0 \pmod 6$ here. By Lemma 1, there exists an optimal itinerary I with no 2-cycles, and it consists of a set of 4-cycles and a set of 3-cycles. Hence we will partition I into two sets of edges I_3 and I_4, where I_3 consists of all 3-cycles and I_4 consists of all 4-cycles. Define $0 \leq \gamma \leq 1$, such that $w(I_4) = \gamma w(I)$, and so $w(I_3) = (1-\gamma)w(I)$. Hence γ measures the proportion of weights of the 4-cycles compared to the entire itinerary. The edges in G incident to v in I are called *home-edges*, which is the same as $\delta(v) \cap I$. Let a and b be the proportion of weights of the home-edges in I_4 and I_3, respectively. Namely, $aw(I_4) = w(I_4 \cap \delta(v))$ is the weight of all home-edges in I_4, and $bw(I_3) = w(I_3 \cap \delta(v))$ is the weight of all home-edges in I_3.

Now, we are ready to give some stronger bounds for the optimal itinerary.

▶ **Lemma 2.** *Let C be a minimum weight Hamiltonian cycle in G. Then $w(I) \geq w(C)$.*

Proof. According to the definition of the itinerary, we know that each vertex in I has an even degree. Then, we can get an Euler tour in I and obtain a Hamiltonian cycle in graph G by shortcutting I. Hence $w(I) \geq w(C)$. ◀

We also recall the following result.

▶ **Lemma 3** ([7, 23]). *For a graph G, let C be a minimum weight Hamiltonian cycle, and C' be a Hamiltonian cycle obtained by the Christofides-Serdyukov algorithm. Then $w(C') \leq MST(G) + \frac{1}{2}w(C)$.*

For ease of proofs in the rest of the section, we also define V_3 and V_4 that partition the vertices in $V \setminus \{v\}$, where V_3 consists of all vertices in I_3 except v, and V_4 consists of all vertices in I_4 except v. Our results are mostly simple counting arguments.

▶ **Lemma 4.** $(1 - \frac{1}{2}\gamma + a\gamma)w(I) \geq (\frac{1}{2} + a)\gamma w(I) + b(1-\gamma)w(I) \geq \deg(v)$.

Proof. By the definition of b, we have $b \leq 1$. So, we have $(1 - \frac{1}{2}\gamma + a\gamma)w(I) = (\frac{1}{2} + a)\gamma w(I) + (1-\gamma)w(I) \geq (\frac{1}{2} + a)\gamma w(I) + b(1-\gamma)w(I)$. The left inequality holds. Now we show the right inequality.

First, one can see $\deg(v) = \sum_{u \in V_3} w(u,v) + \sum_{u \in V_4} w(u,v)$.

Next, recall that $b(1-\gamma)w(I) = w(I_3 \cap \delta(v))$ is the total weight of all home-edges in 3-cycles. Thus, we have $\sum_{u \in V_3} w(u,v) = b(1-\gamma)w(I)$. Similarly $a\gamma w(I) = w(I_4 \cap \delta(v))$.

For any 4-cycle $vv_1v_2v_3$ in I_4, we note that the edges v_1v and v_3v are home-edges which can be counted by $w(I_4 \cap \delta(v))$, but the edge v_2v is not a home-edge. By the triangle inequality, we know that $w(v,v_2) \leq \frac{1}{2}(w(v,v_1) + w(v_1,v_2) + w(v_2,v_3) + w(v_3,v))$. Therefore, the weight of the uncounted edge is at most half that of the 4-cycle.

Thus, we have $(\frac{1}{2}+a)\gamma w(I) \geq \sum_{u\in V_4} w(u,v)$. Then, we have $(\frac{1}{2}+a)\gamma w(I)+b(1-\gamma)w(I) \geq \sum_{u\in V_3} w(u,v) + \sum_{u\in V_4} w(u,v) = \deg(v)$. ◀

▶ **Lemma 5.** $(1-\frac{1}{2}a)\gamma w(I) + (1-\frac{1}{2}b)(1-\gamma)w(I) \geq MST(G)$.

Proof. In the optimal itinerary of v, we note that the total weight of all home-edges is $a\gamma w(I) + b(1-\gamma)w(I)$. For each cycle, there are exactly two home-edges. We can delete the longer edge from each cycle from I, and we can get a spanning tree with the total weight less than $(1-\frac{1}{2}a)\gamma w(I) + (1-\frac{1}{2}b)(1-\gamma)w(I)$. Since the weight of a minimum spanning tree is $\mathrm{MST}(G)$, we have that $(1-\frac{1}{2}a)\gamma w(I) + (1-\frac{1}{2}b)(1-\gamma)w(I) \geq \mathrm{MST}(G)$. ◀

Note that our bounds in Lemmas 4 and 5 are stronger than that in [29]. Next, we will propose two new lower bounds on minimum weight matching and triangle packing.

▶ **Lemma 6.** *Let M be a minimum weight perfect matching in G. Then $\frac{1}{2}\gamma w(I) + (1-b)(1-\gamma)w(I) \geq w(M)$.*

Proof. We note that the number of vertices in V_3 is even but odd in V_4. Since any pair of 3-cycles only share one common vertex v, after we delete both of home-edges for each 3-cycle, we can get a perfect matching M_1 in graph $G[V_3]$ with a total weight of $(1-b)(1-\gamma)w(I)$.

Then, by a similar argument with the proof of Lemma 2, we know that $\gamma w(I) = w(I_4)$, is greater than the weight of the minimum weight Hamiltonian cycle in graph $G[V_4 \cup \{v\}]$. Since the number of vertices in this graph is even, any Hamiltonian cycle in this graph can be decomposed into two perfect matching. Thus, we can get a perfect matching M_2 in this graph with a total weight less than $\frac{1}{2}\gamma w(I)$. Therefore, $M_1 \cup M_2$ is a perfect matching, and $w(M) \leq w(M_1 \cup M_2) \leq \frac{1}{2}\gamma w(I) + (1-b)(1-\gamma)w(I)$. ◀

▶ **Lemma 7.** *For a graph G, let P^* be a minimum weight P_3 packing, and T^* be a minimum weight triangle packing. Then $(\frac{2}{3}+\frac{1}{3}\gamma - a\gamma)w(I) = (1-a)\gamma w(I) + \frac{2}{3}(1-\gamma)w(I) \geq w(P^*) \geq \frac{1}{2}w(T^*)$.*

Proof. First, we show $w(P^*) \geq \frac{1}{2}w(T^*)$. Let T' be the triangle packing obtained by completing the P_3 packing. For any P_3 path in P^*, saying uvw, we obtain $w(u,v)+w(v,w) \geq w(u,w)$ by the triangle inequality. This shows $2w(P^*) \geq w(T') \geq w(T^*)$, and we are done.

We note that the number of vertices is divisible by 3 in V_4 but not in V_3. Since any pair of 4-cycles only share one common vertex v, after we delete both of home-edges for each 4-cycle, we can get a P_3 path packing P' in graph $G[V_4]$ such that $w(P') = (1-a)\gamma w(I)$.

Then, by a similar argument with the proof of Lemma 2, we know that $(1-\gamma)w(I) = w(I_3) \geq w(C)$ where C is the minimum weight Hamiltonian cycle in graph $G[V_3 \cup \{v\}]$. Since the number of vertices in this graph equals n minus the number of vertices in V_4, which is divisible by 3, we can delete some edges in C to get a vertex disjoint P_3 path packing P'' such that $w(P'') \leq \frac{2}{3}w(C)$. $P' \cup P''$ is a P_3 packing in $G[V]$. Thus, we have that $(1-a)\gamma w(I) + \frac{2}{3}(1-\gamma)w(I) \geq w(P' \cup P'') \geq w(P^*)$. ◀

3.1.2 Independent lower bounds

Now, we are ready for the independent lower bounds, which are found by summing the individual optimal itineraries. Note that the notations I, I_3, I_4, V_3, V_4, γ, a and b in the previous subsection refer to I_v, $I_{v,3}$, $I_{v,4}$, $V_{v,3}$, $V_{v,4}$, γ_v, a_v and b_v. We omitted the subscripts for the simplification.

For each team v, I_v is the optimal itinerary of v. Recall that $\psi = \sum_{v\in V} w(I_v)$. $I_{v,3}$ and $I_{v,4}$ consist of all 3-cycles and all 4-cycles of I_v, respectively.

Define γ, a and b so that $\gamma\psi = \sum_{v\in V} w(I_{v,4}) = \sum_{v\in V} \gamma_v w(I_v)$, $a\gamma\psi = \sum_{v\in V} w(\delta(v) \cap I_{v,4}) = \sum_{v\in V} a_v\gamma_v w(I_v)$, and $b(1-\gamma)\psi = \sum_{v\in V} w(\delta(v)\cap I_{v,3}) = \sum_{v\in V} b_v(1-\gamma_v)w(I_v)$ hold. Then, we have $0 \leq \gamma, a, b \leq 1$.

The lemmas we proved previously can be summed together to obtain different independent lower bounds.

▶ **Lemma 8.** *Let C be a minimum weight Hamiltonian cycle in G. Then $\psi \geq nw(C)$.*

Proof. Recall that $\psi = \sum_{v\in V} \psi_v$. By Lemma 2, we have that $\psi = \sum_{v\in V} w(I_v) \geq nw(C)$. ◀

▶ **Lemma 9.** $(1-\frac{1}{2}\gamma + a\gamma)\psi \geq (\frac{1}{2}+a)\gamma\psi + b(1-\gamma)\psi \geq \varDelta$.

Proof. By the definitions of γ, a and b, we know that $(1-\frac{1}{2}\gamma+a\gamma)\psi = \sum_{v\in V}(1-\frac{1}{2}\gamma_v + a_v\gamma_v)w(I_v)$ and $(\frac{1}{2}+a)\gamma\psi + b(1-\gamma)\psi = \sum_{v\in V}((\frac{1}{2}+a_v)\gamma_v w(I_v) + b_v(1-\gamma_v)w(I_v))$.

Recall that $\varDelta = \sum_{v\in V}\deg(v)$. By Lemma 4, it holds that $\sum_{v\in V}(1-\frac{1}{2}\gamma_v + a_v\gamma_v)w(I_v) \geq \sum_{v\in V}((\frac{1}{2}+a_v)\gamma_v w(I_v) + b_v(1-\gamma_v)w(I_v)) \geq \sum_{v\in V}\deg(v) = \varDelta$. Therefore, we have that $(1-\frac{1}{2}\gamma+a\gamma)\psi \geq (\frac{1}{2}+a)\gamma\psi + b(1-\gamma)\psi \geq \varDelta$. ◀

We note that $(1-\frac{1}{2}\gamma + a\gamma) \leq \frac{3}{2}$. Thus, we have that $\varDelta = O(1)\psi$.

▶ **Lemma 10.** $(1-\frac{1}{2}a)\gamma\psi + (1-\frac{1}{2}b)(1-\gamma)\psi \geq nMST(G)$.

Proof. Note that $(1-\frac{1}{2}a)\gamma\psi + (1-\frac{1}{2}b)(1-\gamma)\psi = \sum_{v\in V}((1-\frac{1}{2}a_v)\gamma_v w(I_v) + (1-\frac{1}{2}b_v)(1-\gamma_v)w(I_v))$.

By Lemma 5, we know that $(1-\frac{1}{2}a)\gamma\psi + (1-\frac{1}{2}b)(1-\gamma)\psi = \sum_{v\in V}((1-\frac{1}{2}a_v)\gamma_v w(I_v) + (1-\frac{1}{2}b_v)(1-\gamma_v)w(I_v)) \geq n\mathrm{MST}(G)$. ◀

By Lemmas 9 and 10, we have that

$$(1+\tfrac{1}{4}\gamma)\psi \geq \tfrac{1}{2}\varDelta + n\mathrm{MST}(G). \tag{2}$$

▶ **Lemma 11.** *Let M be a minimum weight perfect matching in G. Then $\frac{1}{2}\gamma\psi + (1-b)(1-\gamma)\psi \geq nw(M)$.*

Proof. Note that $\frac{1}{2}\gamma\psi + (1-b)(1-\gamma)\psi = \sum_{v\in V}(\frac{1}{2}\gamma_v w(I_v) + (1-b_v)(1-\gamma_v)w(I_v))$.

By Lemma 6, we know that $\frac{1}{2}\gamma\psi + (1-b)(1-\gamma)\psi = \sum_{v\in V}(\frac{1}{2}\gamma_v w(I_v) + (1-b_v)(1-\gamma_v)w(I_v)) \geq nw(M)$. ◀

By Lemmas 9 and 11, we have that

$$(1+a\gamma)\psi \geq \varDelta + nw(M) \tag{3}$$

for a minimum weight matching M.

▶ **Lemma 12.** *For a graph G, let P^* be a minimum weight P_3 packing, and T^* be a minimum weight triangle packing. Then $(\frac{2}{3}+\frac{1}{3}\gamma - a\gamma)\psi = (1-a)\gamma\psi + \frac{2}{3}(1-\gamma)\psi \geq nw(P^*) \geq \frac{1}{2}nw(T^*)$.*

Proof. Note that $(\frac{2}{3}+\frac{1}{3}\gamma - a\gamma)\psi = \sum_{v\in V}(\frac{2}{3}+\frac{1}{3}\gamma_v - a_v\gamma_v)w(I_v)$ and $(1-a)\gamma\psi + \frac{2}{3}(1-\gamma)\psi = \sum_{v\in V}((1-a_v)\gamma_v w(I_v) + \frac{2}{3}(1-\gamma_v)w(I_v))$.

By Lemma 7, it holds that $\sum_{v\in V}(\frac{2}{3}+\frac{1}{3}\gamma_v - a_v\gamma_v)w(I_v) \geq \sum_{v\in V}((1-a_v)\gamma_v w(I_v) + \frac{2}{3}(1-\gamma_v)w(I_v)) \geq nw(P^*) \geq \frac{1}{2}nw(T^*)$. Therefore, we have that $(\frac{2}{3}+\frac{1}{3}\gamma - a\gamma)\psi = (1-a)\gamma\psi + \frac{2}{3}(1-\gamma)\psi \geq nw(P^*) \geq \frac{1}{2}nw(T^*)$. ◀

Next, we will describe our algorithm. Our algorithm consists of two constructions, where the first is based on the Hamiltonian cycle and the second is based on the triangle packing. The approximation quality of each will be analyzed after showing the construction. Finally, we will make a trade-off between them and get the final approximation ratio.

3.2 The Hamiltonian Cycle Construction

In our algorithm, the idea of Hamiltonian cycle construction is to make use of the canonical schedule [19, 8] and a Hamiltonian cycle. For TTP-k, there are many approximation algorithms using this method [29, 27, 18], and hence we will directly use the well-analyzed schedule in [29]. However, we will give a tighter analysis. Next, we give a brief introduction to this construction.

Roughly speaking, this schedule is generated by a rotation scheme that can make sure that almost all road trips of each team t_i are 4-cycles and in each road trip, team t_i visits a set of consecutive teams along the Hamiltonian cycle.

▶ **Lemma 13** ([29]). *Let C be a Hamiltonian cycle in graph G. For TTP-3, there is a polynomial-time algorithm that can generate a solution with a total weight of at most $\frac{2}{3}nw(C) + \frac{2}{3}\Delta + O(\frac{1}{n})\psi$.*

Note that the Hamiltonian cycle used in [29] is generated by the Christofides-Serdyukov algorithm. In our algorithm, we also consider another Hamiltonian cycle that uses the minimum weight perfect matching. We will select the better one between these two cycles.

▶ **Lemma 14** (*). *Let M be a perfect matching in graph G. Then there is a polynomial-time algorithm that can generate a Hamiltonian cycle C such that $w(C) = w(M) + \frac{1}{n}w(E) + O(\frac{1}{n^2})w(E)$.*

Note that Lemma 14 holds for any perfect matching. We will consider the Hamiltonian cycle that uses a minimum weight perfect matching.

▶ **Theorem 15.** *For any $\varepsilon > 0$, there is a polynomial-time algorithm that can generate a feasible schedule for TTP-3 with an approximation ratio of $\min\{\frac{4}{3} + \frac{1}{3}a\gamma,\ 1 - \frac{1}{6}\gamma + a\gamma\} + \varepsilon$.*

Proof. Here we use C to denote a minimum weight Hamiltonian cycle in G. If we use the Hamiltonian cycle C' obtained by the Christofides-Serdyukov algorithm, we can construct a feasible schedule with a total weight of at most $\frac{2}{3}nw(C') + \frac{2}{3}\Delta + O(\frac{1}{n})\psi$ by Lemma 13. Then, we have that

$$\begin{aligned}\frac{2}{3}nw(C') + \frac{2}{3}\Delta + O(\frac{1}{n})\psi &\leq \frac{2}{3}n\left(\mathrm{MST}(G) + \frac{1}{2}w(C)\right) + \frac{2}{3}\Delta + O(\frac{1}{n})\psi \\ &= \frac{2}{3}n\mathrm{MST}(G) + \frac{1}{3}nw(C) + \frac{2}{3}\Delta + O(\frac{1}{n})\psi \\ &\leq (\frac{4}{3} + \frac{1}{3}a\gamma)\psi + O(\frac{1}{n})\psi,\end{aligned}$$

where the first inequality follows from Lemma 3 and the second follows from Lemmas 8 and 9, and (2).

Similarly, if we use a minimum weight perfect matching M to obtain the Hamiltonian cycle C_M in Lemma 14, we can construct a feasible schedule with a total weight of at most

$$\begin{aligned}\frac{2}{3}nw(C_M) + \frac{2}{3}\Delta + O(\frac{1}{n})\psi &\leq \frac{2}{3}n\left(w(M) + \frac{1}{n}w(E) + O(\frac{1}{n^2})w(E)\right) + \frac{2}{3}\Delta + O(\frac{1}{n})\psi \\ &\leq \frac{2}{3}nw(M) + \Delta + O(\frac{1}{n})\psi \\ &\leq (1 - \frac{1}{6}\gamma + a\gamma)\psi + O(\frac{1}{n})\psi,\end{aligned}$$

where the first inequality follows from Lemma 14, the second follows from $w(E) = \frac{1}{2}\Delta \leq O(1)\psi$, and the last follows from Lemma 9 and (3).

Since we select the better one between these two Hamiltonian cycles, then the approximation ratio is $\min\{\frac{4}{3}+\frac{1}{3}a\gamma,\ 1-\frac{1}{6}\gamma+a\gamma\}+O(\frac{1}{n})$. Hence, there exists a constant c such that the ratio is bounded by $\min\{\frac{4}{3}+\frac{1}{3}a\gamma,\ 1-\frac{1}{6}\gamma+a\gamma\}+\frac{c}{n}$. For an arbitrary $\varepsilon>0$, if $n\leq\frac{c}{\varepsilon}=O(1)$, we can find an optimal solution by brute force, otherwise we use the approximation algorithm. This establishes the approximation ratio of $\min\{\frac{4}{3}+\frac{1}{3}a\gamma,\ 1-\frac{1}{6}\gamma+a\gamma\}+\varepsilon$. ◀

Note that $\max_{0\leq a,\gamma\leq 1}\min\{\frac{4}{3}+\frac{1}{3}a\gamma,\ 1-\frac{1}{6}\gamma+a\gamma\}+\varepsilon$ is maximized when $a=\gamma=1$ with value $(\frac{5}{3}+\varepsilon)$. However, this would require $0=(\frac{2}{3}+\frac{1}{3}\gamma-a\gamma)\psi\geq nw(P^*)\geq\frac{1}{2}nw(T^*)$ by Lemma 12. If we use any constant ratio approximation algorithm for minimum weight P_3 path packing or minimum weight triangle packing, we may get a much better schedule based on them and therefore, we will show that we can do better than $(\frac{5}{3}+\varepsilon)$ by combining the Hamiltonian cycle construction with the triangle packing construction shown next.

3.3 The Triangle Packing and P_3 Path Packing Constructions

In this section, we will construct a feasible schedule based on a triangle packing or a P_3 path packing. The idea is similar to the packing schedule based on a minimum weight perfect matching for TTP-2 [25, 28, 30, 31, 17]. Given a triangle packing (resp., a P_3 path packing), we consider the three normal teams in a triangle (resp., a P_3 path) as a super-team. The packing construction is to first arrange a single round-robin for super-teams and then extend the single round-robin into a double round-robin for normal teams. Although the construction is similar for a given triangle packing and a given P_3 packing, the analysis and the approximation ratio may be different.

3.3.1 Construction

First, we will introduce the single round-robin of super-teams.

Given a triangle packing T (resp., P_3 path packing P) of G, recall that we take the three teams in each triangle (resp., P_3 path) as a super-team. There are n normal teams and then there are $m=\frac{n}{3}$ super-teams. The set of super-teams is $U=\{u_1,u_2,\dots,u_{m-1},u_m\}$. We relabel the n teams such that $u_i=\{t_{3i-2},t_{3i-1},t_{3i}\}$ for each i.

In the construction, the case of $n=6$ is easy, and hence we assume here that $n\geq 12$. Each super-team will attend $m-1$ super-games in $m-1$ time slots. Each super-game in the first $m-2$ time slots will be extended to normal games that span six days, and each super-game in the last time slot will be extended to normal games that span ten days. Therefore, we have $6\times(m-2)+10=6m-2=2n-2$ days of normal games in total. This is the number of days in a double round-robin. We will construct the schedule for super-teams from the first time slot to the last time slot $m-1$. In each of the $m-1$ time slots, we have $\frac{m}{2}$ super-games. The schedule in the last time slot is different from the schedules in the first $m-2$ time slots.

For the first time slot, the $\frac{m}{2}$ super-games are arranged as shown in Figure 1. The super-game including super-team u_m is called *left super-game* and we put a letter 'L' on the edge to indicate it. All other super-games are called *normal super-games*. Each super-game is represented by a directed edge, where a directed edge from u_i to u_j means a super-game between them at the home of u_j. The information of which will be used to extend super-games to normal games between normal teams.

In Figure 1, we can see that the last super-team u_m is denoted as a dark node and all other super-teams $u_1,\dots,u_{m-1}$ are denoted as white nodes which form a cycle. In the second time slot, we keep the position of u_m and change the positions of white super-teams in the cycle by moving one position in the clockwise direction, and we also change the direction of each edge. In the second time slot, there are still $\frac{m}{2}-1$ normal super-games and one left super-game.

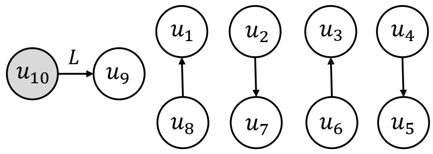

Figure 1 The super-game schedule in the first time slot for an instance with $m = 10$.

Table 1 Extending normal super-games where home games are marked in bold.

	$6q-5$	$6q-4$	$6q-3$	$6q-2$	$6q-1$	$6q$
a	x	z	y	$\boldsymbol{x}$	$\boldsymbol{z}$	$\boldsymbol{y}$
b	y	x	z	$\boldsymbol{y}$	$\boldsymbol{x}$	$\boldsymbol{z}$
c	z	y	x	$\boldsymbol{z}$	$\boldsymbol{y}$	$\boldsymbol{x}$
x	$\boldsymbol{a}$	$\boldsymbol{b}$	$\boldsymbol{c}$	a	b	c
y	$\boldsymbol{b}$	$\boldsymbol{c}$	$\boldsymbol{a}$	b	c	a
z	$\boldsymbol{c}$	$\boldsymbol{a}$	$\boldsymbol{b}$	c	a	b

Table 2 Extending left super-games where home games are marked in bold.

	$6q-5$	$6q-4$	$6q-3$	$6q-2$	$6q-1$	$6q$
a	x	z	$\boldsymbol{y}$	$\boldsymbol{x}$	$\boldsymbol{z}$	y
b	y	x	$\boldsymbol{z}$	$\boldsymbol{y}$	$\boldsymbol{x}$	z
c	z	y	$\boldsymbol{x}$	$\boldsymbol{z}$	$\boldsymbol{y}$	x
x	$\boldsymbol{a}$	$\boldsymbol{b}$	c	a	b	$\boldsymbol{c}$
y	$\boldsymbol{b}$	$\boldsymbol{c}$	a	b	c	$\boldsymbol{a}$
z	$\boldsymbol{c}$	$\boldsymbol{a}$	b	c	a	$\boldsymbol{b}$

The schedules for the other middle slots are derived analogously. Before we introduce the super-games in the last time slot, we first explain how to extend the super-games in the first $m-2$ time slots to normal games. In these time slots, we have two different kinds of super-games: normal super-games and left super-games. We first consider normal super-games.

Case 1. Normal super-games: Each normal super-game will be extended to eighteen normal games in six days. Assume that in a normal super-game, super-team u_i plays against the super-team u_j at the home of u_j in time slot q ($1 \leq i, j \leq m$ and $1 \leq q \leq m-1$). For ease of presentation, we let $u_i = \{t_{3i-2}, t_{3i-1}, t_{3i}\} = \{a, b, c\}$ and $u_j = \{t_{3j-2}, t_{3j-1}, t_{3j}\} = \{x, y, z\}$. The super-game will be extended to eighteen normal games in six corresponding days from $6q-5$ to $6q$, as shown in Table 1 where home games are marked in bold.

Case 2. Left super-games: Each left super-game will be extended to eighteen normal games in six days. Assume that in a left super-game, super-team $u_i = \{a, b, c\}$ plays against super-team $u_j = \{x, y, z\}$ at the home of u_j in time slot q ($2 \leq i \leq m-1$ and $1 \leq q \leq m-2$). The super-game will be extended to normal games in six corresponding days from $6q-5$ to $6q$, as shown in Table 2.

The first $m-2$ time slots will be extended to $6 \times (m-2) = 2n - 12$ days according to the above rules. Each normal team will have ten remaining games, which correspond to the super-games in the last time slot. Figure 2 shows the schedule in the last time slot. All super-games in the last time slot are *last super-game* where we put a letter 'Z' to indicate it.

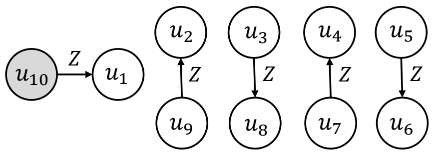

Figure 2 The super-game schedule in the last time slot for an instance with $m = 10$.

Table 3 Extending last super-games where home games are marked in bold.

	$6q-5$	$6q-4$	$6q-3$	$6q-2$	$6q-1$	$6q$	$6q+1$	$6q+2$	$6q+3$	$6q+4$
a	x	z	y	$\boldsymbol{x}$	$\boldsymbol{z}$	$\boldsymbol{y}$	c	b	$\boldsymbol{c}$	$\boldsymbol{b}$
b	y	x	c	$\boldsymbol{y}$	$\boldsymbol{x}$	$\boldsymbol{c}$	z	$\boldsymbol{a}$	$\boldsymbol{z}$	a
c	z	y	$\boldsymbol{b}$	$\boldsymbol{z}$	$\boldsymbol{y}$	b	$\boldsymbol{a}$	$\boldsymbol{x}$	a	x
x	$\boldsymbol{a}$	$\boldsymbol{b}$	z	a	b	$\boldsymbol{z}$	y	c	$\boldsymbol{y}$	$\boldsymbol{c}$
y	$\boldsymbol{b}$	$\boldsymbol{c}$	$\boldsymbol{a}$	b	c	a	$\boldsymbol{x}$	z	x	$\boldsymbol{z}$
z	$\boldsymbol{c}$	$\boldsymbol{a}$	$\boldsymbol{x}$	c	a	x	$\boldsymbol{b}$	$\boldsymbol{y}$	b	y

Case 3. Last super-games: Each normal super-game will be extended to thirty normal games in ten days. Assume that in the last time slot $q = m - 1$, super-team $u_i = \{a, b, c\}$ plays against super-team $u_j = \{x, y, z\}$ ($1 \leq i, j \leq m$) at the home of u_j. The super-game will be extended to thirty normal games in ten corresponding days from $6q - 5$ to $6q + 4$, as shown in Table 3.

▶ **Theorem 16 (*).** *For TTP-3 with n teams such that $n \equiv 0 \pmod 6$, the above construction can generate a feasible schedule.*

3.3.2 Analyzing the Approximation Quality

Note that all teams play three consecutive away games and three consecutive home games in a normal super-game. Indeed, all teams are at home before and after each normal super-game in the schedule, otherwise, it will break the bounded-by-3 property. Using this property, we can get the total cost of normal super-games by analyzing each normal super-game separately.

▶ **Lemma 17 (*).** *If there is a normal super-game between super-teams u_i and u_j at the home of u_j, then the cost of all normal teams in u_i and u_j is at most $\frac{4}{3}c(u_i, u_j) + 2c(u_i) + 2c(u_j)$.*

To analyze the total weight of our schedule, we first analyze the total cost of normal super-games. Recall that there is exactly one super-game between each pair of super-teams in U and then, there are $\frac{m(m-1)}{2}$ super-games in total. We define $R(u_i, u_j) = 1$ if the super-game between super-teams u_i and u_j is a normal super-game, and $R(u_i, u_j) = 0$ otherwise. By Lemma 17 and (1), we know that the total cost of all normal super-games E_0 satisfies that

$$\begin{aligned} E_0 &\leq \sum_{1 \leq i < j \leq m} \left(\tfrac{4}{3}c(u_i, u_j) + 2c(u_i) + 2c(u_j)\right) R(u_i, u_j) \\ &\leq \sum_{1 \leq i < j \leq m} \left(\tfrac{4}{3}c(u_i, u_j) + 2c(u_i) + 2c(u_j)\right) \\ &= \tfrac{4}{3}c(F) + 2(m-1)w(T) \\ &\leq \tfrac{2}{3}\Delta + \tfrac{2}{3}nw(T). \end{aligned} \tag{4}$$

The total cost of all left super-games and all last super-games are denoted by E_1 and E_2, respectively.

▶ **Lemma 18 (*).** *By using $O(n^3)$ time to reorder all super-teams, we can make*

$$E_1 + E_2 = O(\tfrac{1}{n})\psi.$$

Based on Lemma 18 and (4), we can get

▶ **Theorem 19.** *For TTP-3 with the case of* $n \equiv 0 \pmod 6$*, suppose there exists a triangle packing* T *in graph* G*, then there is a polynomial-time algorithm to generate a feasible schedule with a total weight of at most* $\frac{2}{3}\Delta + \frac{2}{3}nw(T) + O(\frac{1}{n})\psi$.

▶ **Theorem 20.** *Suppose there exist* ρ_t *and* ρ_p *approximation algorithms for minimum weight triangle packing and* P_3 *path packing problems respectively, then for an arbitrary* $\varepsilon > 0$*, there is a polynomial-time algorithm for TTP-3 with the case of* $n \equiv 0 \pmod 6$*, whose approximation ratio is* $\frac{8\rho+6}{9} + \frac{4\rho-3}{9}\gamma - \frac{4\rho-2}{3}a\gamma + \varepsilon$*, where* $\rho = \min\{\rho_p, \rho_t\}$.

Proof. First, we consider that the schedule uses a triangle packing T of G. By Theorem 19, we know that the total weight is bounded by $\frac{2}{3}\Delta + \frac{2}{3}nw(T) + O(\frac{1}{n})\psi$. Suppose the triangle packing T is obtained by using a ρ_t-approximation algorithm, i.e., $w(T) \leq \rho_t w(T^*)$, then by Lemmas 9 and 12, we have that

$$\begin{aligned}\frac{2}{3}\Delta + \frac{2}{3}nw(T) + O(\frac{1}{n})\psi &\leq \frac{2}{3}\Delta + \frac{2}{3}\rho_t nw(T^*) + O(\frac{1}{n})\psi\\ &\leq \frac{2}{3}\Delta + \frac{4}{3}\rho_t nw(P^*) + O(\frac{1}{n})\psi\\ &\leq (\frac{8\rho_t+6}{9} + \frac{4\rho_t-3}{9}\gamma - \frac{4\rho_t-2}{3}a\gamma)\psi + O(\frac{1}{n})\psi.\end{aligned}$$

Then, we consider that the schedule uses a P_3 path packing P of G. Suppose the P_3 path packing P is obtained by using a ρ_p-approximation algorithm, i.e., $w(P) \leq \rho_p w(P^*)$, then we can get a triangle packing T' by completing the P_3 path packing P such that $w(T') \leq 2w(P) \leq 2\rho_p w(P^*)$ by the triangle inequality. Thus, by Lemmas 9 and 12, the total weight of our schedule is bounded by

$$\begin{aligned}\frac{2}{3}\Delta + \frac{2}{3}nw(T') + O(\frac{1}{n})\psi &\leq \frac{2}{3}\Delta + \frac{4}{3}\rho_p nw(P^*) + O(\frac{1}{n})\psi\\ &\leq (\frac{8\rho_p+6}{9} + \frac{4\rho_p-3}{9}\gamma - \frac{4\rho_p-2}{3}a\gamma)\psi + O(\frac{1}{n})\psi.\end{aligned}$$

By selecting the better schedule, we can get the approximation ratio of $\frac{8\rho+6}{9} + \frac{4\rho-3}{9}\gamma - \frac{4\rho-2}{3}a\gamma + \varepsilon$, where $\rho = \min\{\rho_p, \rho_t\}$. Note that the algorithm of our triangle packing construction runs in polynomial time since it takes $O(n^3)$ time to reorder all super-teams and $O(n^2)$ time to construct the schedule. ◀

3.4 Trade-off between Two Constructions

▶ **Theorem 21 (*).** *Suppose there exist* ρ_t *and* ρ_p *approximation algorithms for minimum weight triangle packing and* P_3 *path packing problems respectively, for TTP-3 with an arbitrary* $\varepsilon > 0$*, there is an approximation algorithm whose ratio is* $\frac{11}{6} - \frac{5}{2(4\rho+1)} + \varepsilon$ *when* $\rho \leq \frac{9}{4}$*, and* $\frac{5}{3} - \frac{2}{3(4\rho-1)} + \varepsilon$ *otherwise, where* $\rho = \min\{\rho_t, \rho_p\}$*. Given* $\rho = \frac{8}{3}$*, the approximation ratio is* $\frac{139}{87} + \varepsilon < 1.598 + \varepsilon$ *which improves the previous ratio of* $\frac{5}{3} + \varepsilon < 1.667 + \varepsilon$.

Proof. For the case of $n \equiv 0 \pmod 6$, our algorithm will select the better schedule from the two constructions. By Theorem 15, the ratio of the Hamiltonian cycle construction is $\min\{\frac{4}{3} + \frac{1}{3}a\gamma,\ 1 - \frac{1}{6}\gamma + a\gamma\} + \varepsilon$, and by Theorem 20, the ratio of the triangle packing construction is $\frac{8\rho+6}{9} + \frac{4\rho-3}{9}\gamma - \frac{4\rho-2}{3}a\gamma + \varepsilon$. Therefore, the ratio of our algorithm in the worst case is

$$\max_{0\leq a,\gamma\leq 1} \min\left\{\frac{4}{3} + \frac{1}{3}a\gamma,\ 1 - \frac{1}{6}\gamma + a\gamma,\ \frac{8\rho+6}{9} + \frac{4\rho-3}{9}\gamma - \frac{4\rho-2}{3}a\gamma\right\} + \varepsilon.$$

We can transform it into the following linear programming where we let $\widetilde{\gamma} = a\gamma$.

$$\begin{aligned}
\max \quad & y \\
\text{s.t.} \quad & y \leq \frac{4}{3} + \frac{1}{3}\widetilde{\gamma}, \\
& y \leq 1 - \frac{1}{6}\gamma + \widetilde{\gamma}, \\
& y \leq \frac{8\rho+6}{9} + \frac{4\rho-3}{9}\gamma - \frac{4\rho-2}{3}\widetilde{\gamma}, \\
& 0 \leq \widetilde{\gamma} \leq \gamma \leq 1.
\end{aligned}$$

It shows that the ratio is $\frac{11}{6} - \frac{5}{2(4\rho+1)} + \varepsilon$ when $\rho \leq \frac{9}{4}$, and $\frac{5}{3} - \frac{2}{3(4\rho-1)} + \varepsilon$ otherwise.

To our best knowledge, both of the current best-known ratios for minimum weight triangle packing and minimum weight P_3 path packing problems are $\frac{8}{3}$ [12]. Therefore, given $\rho = \rho_p = \rho_t = \frac{8}{3}$, the final approximation ratio of our algorithm is $\frac{139}{87} + \varepsilon < 1.598 + \varepsilon$.

Note that the cases $n \equiv 2 \pmod 6$ and $n \equiv 4 \pmod 6$ have not been analyzed yet. Since n is not divisible by 3 for these two cases, there is no perfect P_3 path packing or triangle packing in G. However, with some modifications, we can extend the second construction and its analysis to get the same approximation ratio. ◀

By Theorem 21, we know that the approximation ratio can achieve $(4/3 + \varepsilon)$ if $\rho = 1$.

▶ **Corollary 22.** *If a minimum weight triangle packing or P_3 path packing is given, there exists a $(4/3+\varepsilon)$-approximation algorithm for TTP-3, for an arbitrary $\varepsilon > 0$.*

4 LDTTP-3

When all teams are located on a straight line, this problem is known as Linear Distance TTP-3 (LDTTP-3) [15]. For this problem, the minimum weight triangle packing and P_3 path packing can be found in polynomial time. Thus, for LDTTP-3, our algorithm has an approximation ratio of $(4/3 + \varepsilon)$, which also matches the current best-known ratio of 4/3 (however, note that their construction only works for the case of $n \equiv 4 \pmod 6$) [15]. Their ratio is based on a stronger lower bound of LDTTP-3. If we use this lower bound, the approximation ratio of our triangle packing construction can be proved to be $(6/5 + \varepsilon)$.

▶ **Theorem 23 (*).** *For LDTTP-3, there is an approximation algorithm whose ratio is $(6/5 + \varepsilon)$, for an arbitrary $\varepsilon > 0$.*

5 TTP-4

It is natural to use the same idea to solve TTP-4. For the problem of minimum weight P_4 path packing, to our best knowledge, the current best-known ratio is 3/2 [22]. If we use the construction based on the Hamiltonian cycle and the construction based on P_4 path packing, we can get a $(17/10+\varepsilon)$-approximation algorithm for TTP-4 which improves the previous approximation ratio of $(7/4 + \varepsilon)$ [29].

▶ **Theorem 24 (*).** *For TTP-4 with any $\varepsilon > 0$, there is an algorithm whose approximation ratio is $(17/10 + \varepsilon)$ which improves the previous approximation ratio of $(7/4 + \varepsilon)$.*

For TTP-k with $k \geq 5$, we note that both of the best-known approximation ratios for minimum weight k-cycle packing and P_k path packing are $4(1 - 1/k) > 3$ [12]. The approximation ratios are too large and we can not improve TTP-k by using the same idea.

References

1 Aris Anagnostopoulos, Laurent Michel, Pascal Van Hentenryck, and Yannis Vergados. A simulated annealing approach to the traveling tournament problem. *Journal of Scheduling*, 9(2):177–193, 2006.

2 Rishiraj Bhattacharyya. Complexity of the unconstrained traveling tournament problem. *Operations Research Letters*, 44(5):649–654, 2016.

3 David Van Bulck, Dries R. Goossens, Jörn Schönberger, and Mario Guajardo. Robinx: A three-field classification and unified data format for round-robin sports timetabling. *Eur. J. Oper. Res.*, 280(2):568–580, 2020.

4 Robert Thomas Campbell and Der-San Chen. A minimum distance basketball scheduling problem. *Management science in sports*, 4:15–26, 1976.

5 Diptendu Chatterjee. Complexity of traveling tournament problem with trip length more than three. *CoRR*, abs/2110.02300, 2021. `arXiv:2110.02300`.

6 Diptendu Chatterjee and Bimal Kumar Roy. An improved scheduling algorithm for traveling tournament problem with maximum trip length two. In *ATMOS 2021*, volume 96, pages 16:1–16:15, 2021.

7 Nicos Christofides. Worst-case analysis of a new heuristic for the travelling salesman problem. Technical report, Carnegie-Mellon Univ Pittsburgh Pa Management Sciences Research Group, 1976.

8 Dominique De Werra. Scheduling in sports. *Studies on graphs and discrete programming*, 11:381–395, 1981.

9 Luca Di Gaspero and Andrea Schaerf. A composite-neighborhood tabu search approach to the traveling tournament problem. *Journal of Heuristics*, 13(2):189–207, 2007.

10 Kelly Easton, George Nemhauser, and Michael Trick. The traveling tournament problem: description and benchmarks. In *7th International Conference on Principles and Practice of Constraint Programming*, pages 580–584, 2001.

11 Kelly Easton, George Nemhauser, and Michael Trick. Solving the travelling tournament problem: a combined integer programming and constraint programming approach. In *4th International Conference of Practice and Theory of Automated Timetabling IV*, pages 100–109, 2003.

12 Michel X. Goemans and David P. Williamson. A general approximation technique for constrained forest problems. *SIAM J. Comput.*, 24(2):296–317, 1995.

13 Marc Goerigk, Richard Hoshino, Ken Kawarabayashi, and Stephan Westphal. Solving the traveling tournament problem by packing three-vertex paths. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 2271–2277, 2014.

14 Marc Goerigk and Stephan Westphal. A combined local search and integer programming approach to the traveling tournament problem. *Ann. Oper. Res.*, 239(1):343–354, 2016.

15 Richard Hoshino and Ken-ichi Kawarabayashi. Generating approximate solutions to the TTP using a linear distance relaxation. *J. Artif. Intell. Res.*, 45:257–286, 2012.

16 Richard Hoshino and Ken-ichi Kawarabayashi. An approximation algorithm for the bipartite traveling tournament problem. *Mathematics of Operations Research*, 38(4):720–728, 2013.

17 Shinji Imahori. A 1+O(1/N) approximation algorithm for TTP(2). *CoRR*, abs/2108.08444, 2021. `arXiv:2108.08444`.

18 Shinji Imahori, Tomomi Matsui, and Ryuhei Miyashiro. A 2.75-approximation algorithm for the unconstrained traveling tournament problem. *Annals of Operations Research*, 218(1):237–247, 2014.

19 Thomas P Kirkman. On a problem in combinations. *Cambridge and Dublin Mathematical Journal*, 2:191–204, 1847.

20 Andrew Lim, Brian Rodrigues, and Xingwen Zhang. A simulated annealing and hill-climbing algorithm for the traveling tournament problem. *European Journal of Operational Research*, 174(3):1459–1478, 2006.

21 Ryuhei Miyashiro, Tomomi Matsui, and Shinji Imahori. An approximation algorithm for the traveling tournament problem. *Annals of Operations Research*, 194(1):317–324, 2012.

22 Jérôme Monnot and Sophie Toulouse. Approximation results for the weighted p_4 partition problem. *J. Discrete Algorithms*, 6(2):299–312, 2008.

23 Anatolii Ivanovich Serdyukov. Some extremal bypasses in graphs. *Upravlyaemye Sistemy*, 17:76–79, 1978.

24 Clemens Thielen and Stephan Westphal. Complexity of the traveling tournament problem. *Theoretical Computer Science*, 412(4):345–351, 2011.

25 Clemens Thielen and Stephan Westphal. Approximation algorithms for TTP(2). *Mathematical Methods of Operations Research*, 76(1):1–20, 2012.

26 Michael Trick. Challenge traveling tournament instances. Accessed: 2022-4-01, 2022.

27 Stephan Westphal and Karl Noparlik. A 5.875-approximation for the traveling tournament problem. *Annals of Operations Research*, 218(1):347–360, 2014.

28 Mingyu Xiao and Shaowei Kou. An improved approximation algorithm for the traveling tournament problem with maximum trip length two. In *MFCS 2016*, volume 58, pages 89:1–89:14, 2016.

29 Daisuke Yamaguchi, Shinji Imahori, Ryuhei Miyashiro, and Tomomi Matsui. An improved approximation algorithm for the traveling tournament problem. *Algorithmica*, 61(4):1077–1091, 2011.

30 Jingyang Zhao and Mingyu Xiao. A further improvement on approximating TTP-2. In *COCOON 2021*, volume 13025 of *Lecture Notes in Computer Science*, pages 137–149. Springer, 2021.

31 Jingyang Zhao and Mingyu Xiao. The traveling tournament problem with maximum tour length two: A practical algorithm with an improved approximation bound. In *IJCAI 2021*, pages 4206–4212, 2021.