EuroComb 2005 DMTCS proc. **AE**, 2005, 251–256

Packing triangles in low degree graphs and indifference graphs

Gordana Manić† and Yoshiko Wakabayashi‡

Instituto de Matemática e Estatística da Universidade de São Paulo — Departamento de Ciência da Computação — Rua do Matão 1010, CEP 05508-090, São Paulo, SP – Brazil, {*gocam,yw*}*@ime.usp.br*

We consider the problems of finding the maximum number of vertex-disjoint triangles (VTP) and edge-disjoint triangles (ETP) in a simple graph. Both problems are NP-hard. The algorithm with the best approximation guarantee known so far for these problems has ratio $3/2 + \varepsilon$, a result that follows from a more general algorithm for set packing obtained by Hurkens and Schrijver in 1989. We present improvements on the approximation ratio for restricted cases of VTP and ETP that are known to be APX-hard: we give an approximation algorithm for VTP on graphs with maximum degree 4 with ratio slightly less than 1.2, and for ETP on graphs with maximum degree 5 with ratio $4/3$. We also present an exact linear-time algorithm for VTP on the class of indifference graphs.

Keywords: triangle packing, approximation algorithm, polynomial algorithm, low degree graph, indifference graph

1 Introduction

For a given family $\mathcal{F}$ of sets, any collection of pairwise disjoint sets is called a *packing* of $\mathcal{F}$. The *maximum k-set packing* problem is defined as follows: given a family $\mathcal{F}$ of sets of size $k \geq 2$ over a certain domain, find a largest packing of $\mathcal{F}$. This problem is a fundamental combinatorial problem that underlies a range of practical and theoretical problems. The case $k = 2$ is the well-known maximum matching problem. We study two special cases of the maximum 3-set packing problem that are NP-hard.

A cycle of length 3 in a graph $G = (V_G, E_G)$ is called a *triangle*. Let $\mathcal{T}_V(G)$ (resp. $\mathcal{T}_E(G)$) denote the collection of the sets of vertices (resp. edges) of all triangles in G. We address the following problems on simple graphs. *Vertex-Disjoint Triangle Packing* (VTP): given a graph G, find a maximum size packing of $\mathcal{T}_V(G)$, and *Edge-Disjoint Triangle Packing* (ETP): given a graph G, find a maximum size packing of $\mathcal{T}_E(G)$. For simplicity, we may also refer to a collection of vertex-disjoint (resp. edge-disjoint) triangles of a graph G as a packing of $\mathcal{T}_V(G)$ (resp. $\mathcal{T}_E(G)$).

The problem VTP arises in scheduling airline flight crew (say pilots, copilots, and navigators) to airplanes, while ETP has applications in computational biology (2). As both problems are NP-hard (7; 5), we wish to find good approximation algorithms or special instances amenable to polynomial algorithms. Consider the following local search algorithm HS$(\mathcal{T}, t)$, where $\mathcal{T}$ is $\mathcal{T}_V(G)$ for VTP (resp. $\mathcal{T}_E(G)$ for ETP) and t is a positive integer.

†Partially supported by CAPES, Brazil.
‡Partially supported by CNPq (Proc. 490333/04-4 and 308138/04-0) and PRONEX–FAPESP/CNPq (Proc. 2003/09925-5).

Algorithm $\mathrm{HS}(\mathcal{T}, t)$: Given a collection $\mathcal{C}$ of disjoint sets constructed so far, check whether there are $p \leq t$ disjoint sets in $\mathcal{T} \setminus \mathcal{C}$ that intersect at most $p-1$ sets that are in $\mathcal{C}$. If this happens, swap the sets to form a larger collection $\mathcal{C}$, and repeat; otherwise, return $\mathcal{C}$.

A general result of Hurkens and Schrijver (6) on the maximum k-set packing problem implies that the above algorithm is a $(\frac{3}{2}+\varepsilon)$-approximation algorithm for both VTP and ETP (ε is inversely proportional to t). This ratio is tight and is the best approximation ratio known so far for both problems. There are only a few more results concerning maximum triangle packings. Both problems admit a polynomial-time approximation scheme on planar graphs (1) and λ-precision unit disk graphs (a result proved by Hunt et al. in 1998). The problem VTP is NP-complete when restricted to chordal graphs, while it is polynomially solvable on split graphs and cographs (4).

For a given integer $k \geq 3$, we denote by VTP-k (resp. ETP-k), the problem VTP (resp. ETP) on graphs with maximum degree k. Both VTP-3 and ETP-4 can be solved in polynomial time, whereas VTP-4 and ETP-5 are APX-hard (2). We present improvements on the approximation ratios of these APX-hard instances: a $(3-\frac{\sqrt{13}}{2}+\varepsilon)$-approximation algorithm for VTP-4, and a $\frac{4}{3}$-approximation algorithm for ETP-5. We also give an exact linear-time algorithm for VTP on indifference graphs (or, equivalently, unit interval graphs and proper interval graphs). This result is of interest in view of many applications of such graphs in management, psychology, scheduling, etc.

1.1 Basic definitions and notation

A natural reduction for both VTP and ETP consists in deleting the edges that do not belong to any triangle. We, thus, restrict our attention to simple graphs in which every edge belongs to some triangle; these graphs will be called *irredundant*. The terminology we use is standard. One exception is that, when we write $G-U$ (for $U \subseteq V_G$ or $U \subseteq E_G$) we assume that isolated vertices and edges that do not belong to any triangle on the graph obtained by deleting U from G have been removed as well. Graphs G and H *intersect* if $G \cap H$ is a non-empty graph. The *degree* of a triangle T in a graph G, $d_G(T)$, is the number of triangles in G, different from T, that intersect T. We denote by $\mathcal{T}_G$ the collection of all triangles in G, and by $[u,v,w]$ the triangle with vertices u, v and w. If two triangles T_1 and T_2 of G have only one vertex in common and there is no other triangle in G that intersects both T_1 and T_2, we say that the subgraph $T_1 \cup T_2$ is a *butterfly* in G, and denote by $v_{T_1T_2}$ the only vertex in common to T_1 and T_2. A collection $\mathcal{T}$ of vertex-disjoint triangles in G is *locally optimal* in G if $\{V_T\colon\ T \in \mathcal{T}\}$ is a maximum packing of the family $\{V_T\colon\ T \in \mathcal{T}_G,\ T \text{ intersects a triangle in } \mathcal{T}\}$.

The *intersection graph* of a collection of sets $\mathcal{T}$ is the graph H with $V_H := \mathcal{T}$ and such that $XY \in E_H \Leftrightarrow X \cap Y \neq \emptyset$. A graph G is an *indifference graph* if there exists a positive number δ and a real-valued function f on V_G such that for all $u, v \in V_G$ $(u \neq v)$, uv is an edge in G whenever $|f(u)-f(v)| < \delta$.

2 The problem VTP on graphs with maximum degree 4

We describe in this section an algorithm, called $\mathrm{VT4}_k$, for VTP on graphs with maximum degree 4. This algorithm performs some approximation-preserving reductions to transform the input graph G into another graph G' in which every triangle intersects at most 3 other triangles. Then, on the intersection graph of $\mathcal{T}_{G'}$ it applies the $(3-\frac{\sqrt{13}}{2}+\frac{13-\sqrt{13}}{52k})$-approximation algorithm of Chlebík and Chlebíková (3), which we denote by $\mathrm{MIS3}_k$ (where k is a fixed integer parameter), for the problem of finding a maximum cardinality independent set of vertices on graphs with maximum degree 3. We note that for $k=4$ the above ratio is slightly less than 1.25; and for $k > 65$ it is slightly less than 1.2.

In each iteration of the algorithm $\mathrm{VT4}_k$, a set $\mathcal{T} \subseteq \mathcal{T}_G$, $|\mathcal{T}| \leq 2$, locally optimal in G is repeatedly added to $\mathcal{A}^*$ (the set to be returned by the algorithm) and G is updated. If G contains a triangle T with degree greater than 3, the algorithm finds a certain subgraph H that contains T and applies an appropriate reduction (in a way that in the reduced graph the triangles obtained by this reduction have degree at most 3). The reduction is based on the number of triangles in H that forms a butterfly with a triangle not in H (which is at most 2).

Algorithm $\mathrm{VT4}_k$
Input: A graph G with maximum degree 4.

$\mathcal{A}^* \leftarrow \emptyset$
while exists a triangle in G with degree greater than 3
  do while exists $\mathcal{T} \subseteq \mathcal{T}_G$, $|\mathcal{T}| \leq 2$, locally optimal in G **do** $\mathit{Accept}(\mathcal{T})$
    if exists a triangle $T \in \mathcal{T}_G$ with $d_G(T) > 3$
      then $H \leftarrow$ maximal connected irredundant subgraph of G that
          contains T and does not contain any butterfly
        $B_H \leftarrow \{T' \in \mathcal{T}_H\colon$ exists a triangle in $\mathcal{T}_G \setminus \mathcal{T}_H$ that forms a butterfly with T' in $G\}$
        if $|B_H| = 2$ **then** apply $\mathit{Reduction}(H)$
          else if $|B_H| = 0$
            then { take a triangle $\widetilde{T}$ in $\mathcal{T}_H$, $\mathrm{Sol}_H \leftarrow \widetilde{T} \cup \mathit{Commit}(H - V_{\widetilde{T}})$ }
          if $|B_H| = 1$ **then** $\mathrm{Sol}_H \leftarrow \mathit{Commit}(H)$
          $\mathcal{A}^* \leftarrow \mathcal{A}^* \cup \mathrm{Sol}_H$
if $G \neq \emptyset$ **then** $\mathcal{A}^* \leftarrow \mathcal{A}^* \cup \mathrm{MIS3}_k$(intersection graph of $\mathcal{T}_G$)
for every application of $\mathit{Reduction}(H)$ **do** $\mathit{Restauration}(H)$
return $\mathcal{A}^*$

Each of the procedures is described next in more detail.

1. $\mathit{Accept}(\mathcal{T})$: Add $\mathcal{T}$ to $\mathcal{A}^*$ and delete from G the vertices of all triangles in $\mathcal{T}$.
2. $\mathit{Commit}(H)$: Set $\mathcal{E} := \emptyset$. While $H \neq \emptyset$, find a triangle T locally optimal in H, add T to $\mathcal{E}$ and delete V_T from H. Return $\mathcal{E}$.
3. $\mathit{Reduction}(H)$: Take $T', T'' \in B_H$ and $\widetilde{T}', \widetilde{T}'' \in \mathcal{T}_G \setminus \mathcal{T}_H$ such that $T' \cup \widetilde{T}'$ and $T'' \cup \widetilde{T}''$ are butterflies in G (possibly $\widetilde{T}' = \widetilde{T}''$). Let

$$\mathrm{Sol}_{T'T''} := \{T', T''\} \cup \mathit{Commit}(H - V_{T'} - V_{T''}), \quad \mathrm{Sol}_{T'\overline{T''}} := \{T'\} \cup \mathit{Commit}(H - V_{T'} - v_{T''\widetilde{T}''}),$$
$$\mathrm{Sol}_{\overline{T'}T''} := \{T''\} \cup \mathit{Commit}(H - V_{T''} - v_{T'\widetilde{T}'}), \quad \mathrm{Sol}_{\overline{T'}\overline{T''}} := \mathit{Commit}(H - v_{T'\widetilde{T}'} - v_{T''\widetilde{T}''}).$$

   (a) If $|\mathrm{Sol}_{T'T''}| = |\mathrm{Sol}_{T'\overline{T''}}| = |\mathrm{Sol}_{\overline{T'}T''}| = |\mathrm{Sol}_{\overline{T'}\overline{T''}}|$, then *Accept* $(\mathrm{Sol}_{\overline{T'}\overline{T''}})$.
   (b) If the equalities in (a) are not satisfied and $\widetilde{T}' = \widetilde{T}''$, then *Accept* $(\mathrm{Sol}_{T'T''})$.
   (c) If $|\mathrm{Sol}_{T'T''}| - 1 = |\mathrm{Sol}_{T'\overline{T''}}| = |\mathrm{Sol}_{\overline{T'}T''}| = |\mathrm{Sol}_{\overline{T'}\overline{T''}}|$ and $\widetilde{T}' \neq \widetilde{T}''$ then apply *Reduction* $1(H)$:

$$G \leftarrow \big(G - (E_H \setminus \{E_{T'} \cup E_{T''}\})\big) \cup T_H\ ,$$

   where $T_H := [v', w, v'']$, w is a new vertex, v' is any vertex of T' different from $v_{T'\widetilde{T}'}$, and v'' is any vertex of T'' different from $v_{T''\widetilde{T}''}$. Thus, *Reduction* $1(H)$ replaces all triangles of H, except T' and T'', with a new triangle T_H.

(d) If $|\mathrm{Sol}_{T'T''}| = |\mathrm{Sol}_{T'\overline{T}''}| = |\mathrm{Sol}_{\overline{T}'T''}| = |\mathrm{Sol}_{\overline{T}'\overline{T}''}| + 1$ and $\widetilde{T}' \neq \widetilde{T}''$, then apply $Reduction\,2(H)$:

$$G \leftarrow (G - E_H) \cup T_H^1 \cup T_H^2\ ,$$

where $T_H^1 := [v_{T'\widetilde{T}'}, w_1, w]$, $T_H^2 := [w, w_2, v_{T''\widetilde{T}''}]$ and w_1, w, w_2 are new vertices. Hence, this reduction replaces all triangles of H with the new triangles T_H^1 and T_H^2.

4. $Restauration(H)$:
   (a) If the reduction applied to H was $Reduction\,1(H)$, then if T_H belongs to $\mathcal{A}^*$ before applying $Restauration(H)$, this procedure removes T_H from $\mathcal{A}^*$ and adds to it the set $\mathrm{Sol}_{\overline{T}'\overline{T}''}$ (computed in the procedure $Reduction(H)$); if $T', T'' \in \mathcal{A}^*$, then $\mathcal{A}^* \leftarrow \mathcal{A}^* \cup \mathrm{Sol}_{T'T''}$; if $T' \in \mathcal{A}^*$, $T'' \notin \mathcal{A}^*$, then $\mathcal{A}^* \leftarrow \mathcal{A}^* \cup \mathrm{Sol}_{T'\overline{T}''}$; and if $T' \notin \mathcal{A}^*$, $T'' \in \mathcal{A}^*$, then $\mathcal{A}^* \leftarrow \mathcal{A}^* \cup \mathrm{Sol}_{\overline{T}'T''}$.
   (b) If, however, the reduction applied to H was $Reduction\,2(H)$, then if T_H^1 belongs to $\mathcal{A}^*$ before applying $Restauration(H)$, this procedure adds $\mathrm{Sol}_{T'\overline{T}''}$ to $\mathcal{A}^*$ and removes T_H^1; if $T_H^2 \in \mathcal{A}^*$, then adds $\mathrm{Sol}_{\overline{T}'T''}$ to $\mathcal{A}^*$ and removes T_H^2; and if $T_H^1, T_H^2 \notin \mathcal{A}^*$, then adds $\mathrm{Sol}_{\overline{T}'\overline{T}''}$ to $\mathcal{A}^*$.

Making use of the structural properties of the input graph, maximum degree 4 and irredundancy (that are maintained in each iteration), we can prove that the graph H defined in the algorithm is isomorphic to one of the graphs in Figure 1. Thus, for each iteration of $\mathrm{VT4}_k$, the cardinality of B_H in line 8 is less than 3. If $|B_H| \leq 1$, then $G[V_H]$ is a component of G and Sol_H is an optimal solution in that component. We can also prove that $Reduction\,1$, $Reduction\,2$ (and corresponding restaurations) and $Accept$ are all approximation-preserving reductions, and thus the approximation ratio of $\mathrm{VT4}_k$ is that of $\mathrm{MIS3}_k$.

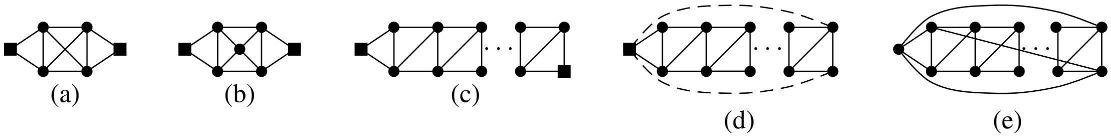


Fig. 1: Possible configurations of graph H. Each square vertex is a vertex common to two triangles in G whose union is a butterfly. The graph (c) has at least 7 vertices.The graphs (d) and (e) have at least 9 vertices, and $G[V_H]$ is a component of G (in (d) dashed lines indicate edges not in E_H).

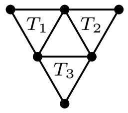
Theorem 2.1 *The algorithm* $\mathrm{VT4}_k$ *is a* $(3 - \frac{\sqrt{13}}{2} + \frac{13-\sqrt{13}}{52k})$*-approximation algorithm for* VTP-4.

The time complexity of $\mathrm{VT4}_k$ is dominated by that of $\mathrm{MIS3}_k$, which is $O(|V_G|^{O(k)})$.

3 The problem ETP on graphs with maximum degree 5

We restrict now our attention to graphs with maximum degree 5 and describe an approximation algorithm, called ET5, for the problem ETP on such graphs.

Algorithm ET5
Input: A graph G with maximum degree 5.
$\mathcal{A}^* \leftarrow \emptyset$
while G contains a Hajos graph $H = H[T_1, T_2, T_3]$ (see the figure beside)
  do $\{\mathcal{A}^* \leftarrow \mathcal{A}^* \cup \{T_1, T_2, T_3\},\ G \leftarrow G - E_H\}$
return $\mathcal{A}^* \cup \{T :\ E_T \in \mathrm{HS}(\mathcal{T}_E(G), 3)\}$



Lemma 3.1 *The algorithm* $\mathrm{HS}(\mathcal{T}_E(G), 3)$ *is a* $\frac{4}{3}$*-approximation algorithm for the problem* ETP-5 *on a graph G that does not contain a Hajós graph.*

The proof of Lemma 3.1 is by induction on the number of triangles in G. For that, one should prove first that if $\mathcal{T}^*$ is a maximum packing of $\mathcal{T}_E(G)$, then there is a subcollection $\mathcal{A} \subseteq \mathrm{HS}(\mathcal{T}_E(G), 3)$ with $|\mathcal{A}| \leq 3$, such that, the ratio of the number of triangles from $\mathcal{T}^*$ that share an edge with a triangle in $\mathcal{A}$ to $|\mathcal{A}|$, is at most $\frac{4}{3}$; and then apply the induction hypothesis on $G - \bigcup_{T \in \mathcal{A}} E_T$. Now, using Lemma 3.1 and the fact that *if G is a graph that contains a Hajós graph H, then the number of triangles in any maximum packing of $\mathcal{T}_E(G)$ that share an edge with H is at most* 4, we obtain the approximation ratio of ET5.

Theorem 3.2 *The algorithm* ET5 *is a $\frac{4}{3}$-approximation algorithm for the problem* ETP-5. *Furthermore, the ratio $\frac{4}{3}$ is tight and the algorithm can be implemented to run in $O(|V_G|^3)$ time.*

4 The problem VTP on indifference graphs

For the next result we use the following characterization obtained by Looges and Olariu (8): a graph G is an indifference graph if, and only if, there exists a linear order $<$ (which we call *canonical*) on V_G such that, for every choice of vertices u, v, w we have that if $u < v < w$ and $uw \in E_G$, then $uv, vw \in E_G$.

Algorithm VTindifference
Input: An indifference graph G.
Find a canonical order $v_1 < v_2 < \cdots < v_n$ on V_G
$\mathcal{A}^* \leftarrow \emptyset$
for $i \leftarrow 1$ to $|V_G| - 2$
  do if $v_i v_{i+2} \in E_G$ **then** $\{ T \leftarrow [v_i, v_{i+1}, v_{i+2}],\ \mathcal{A}^* \leftarrow \mathcal{A}^* \cup T,\ G \leftarrow G - V_T \}$
return $\mathcal{A}^*$

One may prove by contradiction that the algorithm above solves VTP on indifference graphs. It suffices to consider an optimal solution that has the maximum number of triangles in common with the solution found by the algorithm. Since the canonical order can be computed in linear time (8), it follows that the algorithm is linear.

References

[1] B. S. Baker. Approximation algorithms for NP-complete problems on planar graphs. *J. Assoc. Comput. Mach.*, 41(1):153–180, 1994.

[2] A. Caprara and R. Rizzi. Packing triangles in bounded degree graphs. *Inform. Process. Lett.*, 84(4):175–180, 2002.

[3] M. Chlebík and J. Chlebíková. On approximability of the independent set problem for low degree graphs. In *LNCS 3104*, pages 47–56. Springer, 2004.

[4] V. Guruswami, C. Pandu Rangan, M. S. Chang, G. J. Chang, and C. K. Wong. The K_r-packing problem. *Computing*, 66(1):79–89, 2001.

[5] I. Holyer. The NP-completeness of some edge-partition problems. *SIAM J. Comput.*, 10(4):713–717, 1981.

[6] C. A. J. Hurkens and A. Schrijver. 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(1):68–72, 1989.

[7] R. M. Karp. On the computational complexity of combinatorial problems. *Networks*, 5(1):45–68, 1975.

[8] P. J. Looges and S. Olariu. Optimal greedy algorithms for indifference graphs. *Comput. Math. Appl.*, 25(7):15–25, 1993.