Packing Cycles in Undirected Graphs

Alberto Caprara[+] Alessandro Panconesi[†] Romeo Rizzi[o]

[+] DEIS, University of Bologna
Viale Risorgimento 2, I-40136 Bologna, Italy
e-mail: acaprara@deis.unibo.it

[†] Dipartimento di Informatica, University of Bologna
Mura Anteo Zamboni 7, I-40127, Bologna, Italy
e-mail: ale@cs.unibo.it

[o] BRICS[*] Department of Computer Science, University of Aarhus
Ny Munkegade, DK-8000 Aarhus C, Denmark
e-mail: rrizzi@science.unitn.it

Abstract

Given an undirected graph G with n nodes and m edges, we address the problem of finding a largest collection of *edge-disjoint cycles* in G. The problem, dubbed Cycle Packing, is very closely related to a few genome rearrangement problems in computational biology. In this paper, we study the complexity and approximability of Cycle Packing, about which very little is known although the problem is natural and has practical applications. We show that the problem is $\mathcal{APX}$-hard but can be approximated within a factor of $O(\log n)$ by a simple greedy approach. We do not know whether the $O(\log n)$ factor is tight, but we give a nontrivial example for which the ratio achieved by greedy is not constant, namely $\Omega\left(\sqrt{\frac{\log n}{\log\log n}}\right)$. We also show that, for "not too sparse" graphs, i.e. graphs for which $m = \Omega(n^{1+\frac{1}{t}+\delta})$ for some positive integer t and for any fixed $\delta > 0$, we can achieve an approximation arbitrarily close to $\frac{2t}{3}$ in polynomial time. In particular, for any $\varepsilon > 0$, this yields a $\frac{4}{3}+\varepsilon$ approximation when $m = \Omega(n^{\frac{3}{2}+\delta})$, therefore also for dense graphs. Finally, we briefly discuss a natural linear programming relaxation for the problem.

Key words: packing, edge-disjoint cycles, complexity, approximation, linear programming relaxation.

Introduction

Several combinatorial optimization problems in computational biology are related to the following natural packing problem, dubbed here Cycle Packing: given an undirected graph G with n nodes and m edges, find a largest collection of *edge-disjoint cycles* in G. In particular, Sorting by Reversals, a basic problem arising in the reconstruction of evolutionary trees [14, 2], is closely related to the following variant of Cycle Packing. The input is a graph

[*]Basic Research in Computer Science,
Centre of the Danish National Research Foundation.

Graph Class	Complexity	Hardness of Approx.	Approx. Achievable
General	$\mathcal{NP}$-hard [13]	$\mathcal{APX}$-hard	$O(\log n)$
Planar	$\mathcal{NP}$-hard	?	$2+\varepsilon$ [7]
Planar Eulerian	$O(mn)$ [11, 7]	—	—
$\mathcal{G}(t,\alpha,\delta)$	$\mathcal{NP}$-hard [13]	?	$\frac{2t}{3}+\varepsilon$

Table 1: Summary of the results (and open questions) on Cycle Packing.

whose edge set is *bicolored*, i.e. partitioned into, say, *grey* and *black* edges. One is to find the largest collection of edge-disjoint *alternating* cycles. A cycle is alternating if it has even length and its edges are alternately black and grey [2]. The connection of this problem with Cycle Packing is discussed in [6].

Despite its connections and basic character, very little seems to be known about the complexity and approximability of Cycle Packing. To the best of our knowledge, the only known result is the $\mathcal{NP}$-hardness of the problem, implied by an old result of Holyer concerning the packing of triangles [13].

In this paper we study the approximability of Cycle Packing. We show that the problem is $\mathcal{APX}$-hard, but that it can be approximated within a factor of $O(\log n)$ by a simple greedy approach. Whether this bound is tight we do not know and leave it as an interesting open problem. What we are able to show is that the performance of greedy is not constant. More precisely, it must be $\Omega\left(\sqrt{\frac{\log n}{\log\log n}}\right)$. We also show that Cycle Packing is $\mathcal{NP}$-hard for planar graphs. We finally show that the problem can be approximated in polynomial time for "dense" graphs where by dense we mean graphs defined as follows. Given an integer constant $t \geq 2$ and positive constants α and δ, $\mathcal{G}(t,\alpha,\delta)$ consists of those graphs such that $m \geq \alpha n^{1+\frac{1}{t}+\delta}$. For these graphs, we can achieve an approximation arbitrarily close to $\frac{2t}{3}$ (i.e. independent of α and δ) in polynomial time. In particular, for any $\varepsilon > 0$, this yields a $\frac{4}{3}+\varepsilon$ approximation for graphs in $\mathcal{G}(2,\alpha,\delta)$, that include dense graphs. The running time is $O(c_{t\alpha\delta\varepsilon}+p(n))$, where $c_{t\alpha\delta\varepsilon}$ is a (huge) constant depending on $t,\alpha,\delta,\varepsilon$, and $p(n)$ is a polynomial independent of $t,\alpha,\delta,\varepsilon$.

Table 1 summarizes the results proved in this paper, along with the previously-known results. The latter can be recognized by the presence of a reference. Entries without references correspond to results of this paper, while question marks correspond to open problems. In a companion paper [7] we study the following problem, dubbed Cut Packing: given an undirected graph G, find a largest collection of *edge-disjoint cuts* in G. Differently from Cycle Packing, Cut Packing is essentially as hard as Independent Set. However, for planar graphs, Cycle Packing and Cut Packing are equivalent, and our study in [7] yields some of the results in the table. Besides those in the table, the main open question remains, namely the existence of a constant factor approximation algorithm for Cycle Packing. Another intriguing question concerning Cycle Packing is to characterize the exact performance of greedy.

Basic definitions and notation

Consider a maximization problem P. Given a parameter $\rho \geq 1$, a *ρ-approximation algorithm* for P is a polynomial-time algorithm which returns a solution whose value is at least $\frac{1}{\rho} \cdot opt$, where *opt* denotes the optimal solution value. We will also say that the *approximation guarantee* of the algorithm is ρ. Similarly, given a parameter k and a function $f(\cdot)$, an $O(f(k))$-approximation algorithm is a polynomial-time algorithm that returns a solution whose value is $\Omega(\frac{1}{f(k)} \cdot opt)$ (with approximation guarantee $O(f(k))$). A problem P is *$\mathcal{APX}$-hard* if there exists some constant $\sigma > 1$ such that there is no σ-approximation algorithm for P unless $\mathcal{P} = \mathcal{NP}$. We remark that this definition is non-standard but it is adopted here because it simplifies the exposition, while at the same time it allows us to reach the conclusions we are interested in.

Although our results apply to undirected multigraphs, we shall focus on *connected*, *simple* graphs. Connectedness can be assumed without loss of generality, for the problem we study can be solved separately on each connected component and the solutions be pasted together. As for multiple edges they can be removed in pairs, counting one cycle per removed pair.

Given an undirected graph $G = (V, E)$, we will let $n_G := |V|$ and $m_G := |E|$. The subscript G will be omitted when no ambiguity arises.

Definition 1 *A* cycle *of G is a sequence of edges $v_1v_2, v_2v_3, \ldots, v_{k-1}v_k, v_kv_1$, $k \geq 2$, such that $v_i \neq v_j$ for $i \neq j$.*

Sometimes, for convenience, the cycle will be denoted simply by the sequence of nodes it visits, namely $v_1, v_2, \ldots, v_{k-1}, v_k$. A *packing of cycles* is a collection of edge-disjoint cycles, and Cycle Packing is the problem of finding a packing of cycles in G of maximum cardinality.

Definition 2 *ψ_G denotes the maximum size of a packing of cycles in G, d_G denotes the minimum degree of G, and g_G the* girth *of G, i.e. the length of the shortest cycle of G.*

Note that $g_G \geq 3$, as we are dealing with simple graphs only.

Finally, the notation "log" without subscripts stands for the logarithm base 2.

1 The hardness of Cycle Packing

For the $\mathcal{APX}$-hardness proof, the reduction is from Max-2-Sat-3. The input to this problem is a Boolean formula φ in conjunctive normal form in which each clause is the OR of at most 2 literals. Each literal is a variable or the negation of a variable taken from a ground set of Boolean variables $\{x_1, \ldots, x_n\}$, with the additional restriction that each variable appears in at most 3 of the clauses, counting together both positive and negative occurrences. The optimization problem calls for a truth assignment that satisfies as many clauses as possible. It is known that Max-2-Sat-3 is $\mathcal{APX}$-hard [1, 4].

Theorem 1 Cycle Packing *is $\mathcal{APX}$-hard, even for graphs with maximum degree* 3.

Proof: Given an input φ to Max-2-Sat-3, let $\{x_1, \ldots, x_n\}$ denote the set of variables and $\{c_1, \ldots, c_m\}$ the set of clauses. Furthermore, denote by m_i the number of occurrences of x_i.

We will show how to transform φ into a graph $G(\varphi)$ in polynomial time in such a way that every truth assignment for φ that satisfies k clauses can be transformed, again in polynomial time, into a packing of $G(\varphi)$ of value $\sum_{i=1}^{n} 2m_i + k$, and viceversa. Once this is done, the claim follows from the easy observation that $\sum_{i=1}^{n} 2m_i \leq 4m$ and at least half of the clauses can be satisfied by a simple greedy approach. This implies that any $(\frac{1}{1-\varepsilon})$-approximated solution of CYCLE PACKING on $G(\varphi)$ can be transformed in polynomial time into a $(\frac{1}{1-9\varepsilon})$-approximated solution of MAX-2-SAT-3 on φ.

To each clause c_j we associate a *test component*, shown in Fig. 1. The left-hand side shows the test component when the clause has two literals, while the right-hand side shows it when the clause has one literal. The test component of a clause with two literals consists of two *squares* (i.e. cycles of length 4) $s_j^1, r_j^1, r_j^2, t_j^1$ and $s_j^2, r_j^1, r_j^2, t_j^2$ with the edge $r_j^1 r_j^2$ in common. The test component associated with a clause c_j with one literal consists of a single square $s_j^1, r_j^1, r_j^2, t_j^1$.

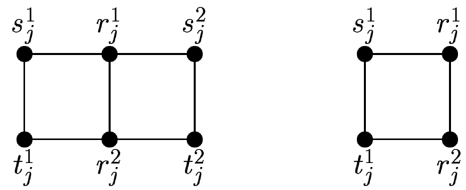

Figure 1: To the left, the test component associated with the clause c_j when it has two literals and, to the right, when it has one literal.

To each variable x_i is associated a *truth setting component.* Please refer to Fig. 2. Without loss of generality we can assume that x_i appears either two or three times, for if it appears only once we can set it to the value which satisfies the clause it belongs to. The truth setting component can be thought of as a "wheel" consisting of two cycles, one "inside" the other, connected by "spikes". The outer, or external, cycle has $12m_i$ nodes, $E := z_i^1, u_i^1, v_i^1, \ldots, z_i^{4m_i}, u_i^{4m_i}, v_i^{4m_i}$, while the inner, or core, cycle has $4m_i$ nodes, $C := b_i^1, \ldots, b_i^{4m_i}$. When referring to a truth setting component, it is understood that all indices are modulo $4m_i$.

Each node b_i^q is connected to z_i^q, $1 \leq q \leq 4m_i$. Each cycle

$$S_q := b_i^q, z_i^q, u_i^q, v_i^q, z_i^{q+1}, b_i^{q+1}$$

is called a *sector*. (Note that there are four sectors in the component for each occurrence of x_i.) A sector is *odd* if q is odd, and *even* otherwise. The *parity* of edge $u_i^q v_i^q$ is the parity of the sector it belongs to (i.e. that of q). Clearly, at most half of the sectors can belong to the same packing, and this is possible only if they all have the same parity.

The graph $G(\varphi)$ is obtained by connecting test and truth setting components as follows. Let x_i be a variable appearing, say, in clauses $c_1, \ldots, c_{m_i}$. For $j = 1, \ldots, m_i$, if x_i appears positive (resp. negated) in c_j, we identify an edge $s_j^h t_j^h$ ($h \in \{1, 2\}$) of the test component of c_j with edge $u_i^{4j-2} v_i^{4j-2}$ (resp. $u_i^{4j-1} v_i^{4j-1}$) of the truth setting component of x_i. We perform identifications in an arbitrary order, so as to guarantee that, for each clause c_j with two literals, both edges $s_j^1 t_j^1$ and $s_j^2 t_j^2$ are identified with one edge in the truth setting components of the two variables occurring in c_j.

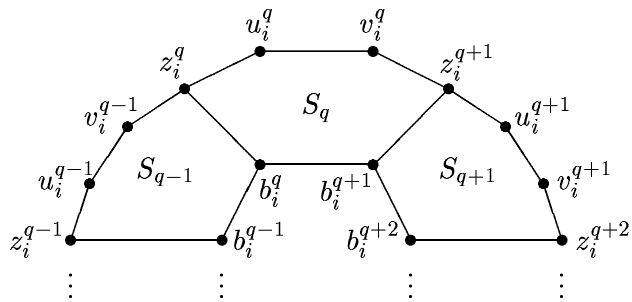

Figure 2: The truth setting component associated with the variable x_i (only three consecutive sectors are depicted).

By the above construction, it is easy to see that, given two (cyclically) consecutive sectors S_q, S_{q+1} in the truth setting component associated with x_i, at least one edge among $u_i^q v_i^q, u_i^{q+1} v_i^{q+1}$ is *not* identified with any edge in a test component. This property simplifies the case analysis below.

A maximal packing $\mathcal{P}$ is called *canonical* if (a) for each truth setting component, it contains either all even sectors or all odd sectors of the component, and (b) for every test component it contains at most one of its squares.

A canonical packing $\mathcal{P}$ naturally corresponds to a truth assignment in the following way. If $\mathcal{P}$ contains a square of the test component of c_j this means that c_j is "satisfied" by $\mathcal{P}$. If $\mathcal{P}$ contains all odd (resp. even) sectors of the truth setting component of x_i this means that x_i is "set to true (resp. false)" by $\mathcal{P}$. Therefore a maximal canonical packing of $G(\varphi)$ which contains $\Sigma_{i=1}^n 2m_i + k$ cycles, k of which are squares, corresponds to a truth assignment of φ satisfying exactly k clauses, and viceversa. The following three claims conclude the proof.

Claim 1 *If k clauses of φ can be simultaneously satisfied, then there is a canonical packing of $G(\varphi)$ with at least $\sum_{i=1}^n 2m_i + k$ edge-disjoint cycles.*

Proof: Consider a truth assignment T which satisfies k clauses. In the truth setting component of variable x_i, take all odd sectors if x_i is *true*. Otherwise, take all even sectors. Now take a square from each test component associated with a clause which is satisfied by T. □

Claim 2 *Given a packing $\mathcal{P}$ we can find in polynomial time a packing $\mathcal{P}'$ which is at least as large and in which each cycle is either fully contained in some test component or in some truth setting component.*

Proof: Consider a cycle C in $\mathcal{P}$ having at least one edge in a truth setting component, say associated with variable x_i, and one edge in a test component, say associated with clause c_j. We assume w.l.o.g. that C contains edge $r_j^1 s_j^1$, where s_j^1 coincides with u_i^q for some $q \in \{1, \ldots, 4m_i\}$. We consider all possible cases (and subcases), in increasing order of complication, and show how to replace C (and possibly other cycles) in $\mathcal{P}$ by squares and sectors so as to get a packing at least as large as $\mathcal{P}$. The case analysis can be checked by looking

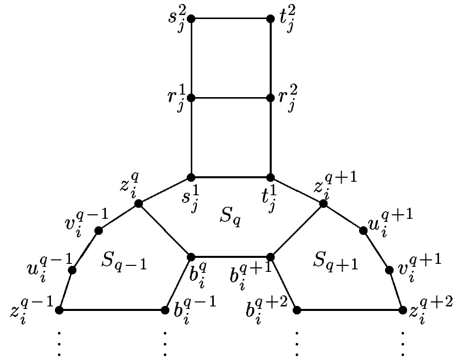

Figure 3: Reference picture for the case analysis in the proof of Claim 2.

at Fig. 3. Given two adjacent nodes u, v of degree 3, we say that edge uv is *covered* by a cycle C if C visits either u or v (or both). Clearly, any edge covered by a cycle C cannot be contained in any other cycle.

Case A: c_j has only one literal. In this case, C must contain also edges $r_j^1 r_j^2, r_j^2 t_j^1$ and covers edge $s_j^1 t_j^1$, therefore a packing as large as $\mathcal{P}$ is obtained by replacing C by the square $s_j^1, r_j^1, r_j^2, t_j^1$.

Case B: c_j has two literals.

Case B.1: C contains also edge $s_j^1 t_j^1$. In this case, note that both edges $r_j^1 r_j^2$ and $r_j^2 t_j^1$ are covered by C. Therefore, a packing as large as $\mathcal{P}$ is obtained by replacing replacing C by the square $s_j^1, r_j^1, r_j^2, t_j^1$.

Case B.2: C contains also edge $z_i^q s_j^1$.

Case B.2.1: C contains also edge $r_j^1 r_j^2$. In this case, both edges $s_j^1 t_j^1$ and $r_j^2 t_j^1$ are covered by C, therefore a packing as large as $\mathcal{P}$ is obtained by replacing C by the square $s_j^1, r_j^1, r_j^2, t_j^1$.

Case B.2.2: C contains also edges $r_j^1 s_j^2$ and $s_j^2 t_j^2$. In this case, both edges $r_j^1 r_j^2$ and $t_j^2 r_j^2$ are covered by C, therefore a packing as large as $\mathcal{P}$ is obtained by replacing C by the square $r_j^1, s_j^2, t_j^2, r_j^2$.

Case B.2.3: C contains also edges $r_j^1 s_j^2$ and $s_j^2 z_h^p$, where x_h is the other variable occurring in c_j and $p \in \{1, \ldots, 4m_h\}$. In this case, edges $s_j^1 t_j^1, r_j^1 r_j^2, s_j^2 t_j^2$ are covered by C.

Case B.2.3.1: Edges $t_j^1 r_j^2$ and $r_j^2 t_j^2$ are not contained in any cycle in $\mathcal{P} \setminus \{C\}$. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C by the square $s_j^1, r_j^1, r_j^2, t_j^1$.

Case B.2.3.2: There exists a cycle C' in $\mathcal{P} \setminus \{C\}$ containing edges $t_j^1 r_j^2$ and $r_j^2 t_j^2$. In this case, C' contains also edge $t_j^1 z_i^{q+1}$. Note that C covers edge $z_i^q b_i^q$ and C' covers edge $z_i^{q+1} b_i^{q+1}$.

Case B.2.3.2.1: Edge $b_i^q b_i^{q+1}$ is not contained in any cycle in $P \setminus \{C, C'\}$. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C, C' by the square $r_j^1, s_j^2, t_j^2, r_j^2$ and the sector $S_q = b_i^q, z_i^q, s_j^1, t_j^1, z_i^{q+1}, b_i^{q+1}$.

Case B.2.3.2.2: There exists a cycle C'' in $\mathcal{P} \setminus \{C, C'\}$ containing edge $b_i^q b_i^{q+1}$. In this case, C'' contains also edges $b_i^{q-1} b_i^q$ and $b_i^{q+1} b_i^{q+2}$. Moreover, since edges $u_i^{q-1} v_i^{q-1}$ and $u_i^{q+1} v_i^{q+1}$ are not in any test component, C contains also path $z_i^q, v_i^{q-1}, u_i^{q-1}, z_i^{q-1}$ and covers edge $b_i^{q-1} z_i^{q-1}$, and C' contains also path $z_i^{q+1}, u_i^{q+1}, v_i^{q+1}, z_i^{q+2}$ and covers edge $b_i^{q+2} z_i^{q+2}$. Therefore, a packing as large as $\mathcal{P}$ is obtained by replacing C, C', C'' by the square $r_j^1, s_j^2, t_j^2, r_j^2$ and the sectors

$S_{q-1} = b_i^{q-1}, z_i^{q-1}, u_i^{q-1}, v_i^{q-1}, z_i^q, b_i^q$ and $S_{q+1} = b_i^{q+1}, z_i^{q+1}, u_i^{q+1}, v_i^{q+1}, z_i^{q+2}, b_i^{q+2}$. □

Claim 3 *Given a packing $\mathcal{P}$ we can find in polynomial time a canonical packing $\mathcal{Q}$ which is at least as large.*

Proof: If $\mathcal{P} = C_1, \ldots, C_\ell$ is a packing of cycles in $G(\varphi)$ we can derive a canonical packing $\mathcal{Q}$ such that:

(1) each cycle of $\mathcal{Q}$ is either fully contained in some test component or in some truth setting component;

(2) every cycle of $\mathcal{Q}$ inside a test component is a square of that test component;

(3) at most one square per test component is in $\mathcal{Q}$;

(4) every cycle of $\mathcal{Q}$ inside a truth setting component is a sector of that truth setting component;

(5) for each truth setting component, $\mathcal{Q}$ contains either all even sectors or all odd sectors of the component.

Condition (1) is shown by Claim 2. Conditions (2) and (3) follow immediately from (1) and the structure of the test components.

In order to show (4), assume $\mathcal{P}$ contains a cycle C inside a truth setting component which is not a sector of the component. If C contains path $b_i^q, z_i^q, u_i^q, v_i^q, z_i^{q+1}, b_i^{q+1}$ for some q, replacing C by the sector S_q yields a packing as large as $\mathcal{P}$. Otherwise, C contains either edges $b_i^q b_i^{q+1}$ and $b_i^{q+1} b_i^{q+2}$ or path $z_i^q, u_i^q, v_i^q, z_i^{q+1}, u_i^{q+1}, v_i^{q+1}, z_i^{q+2}$ for some q. We consider only the first case since the second is perfectly analogous. If C contains also edge $b_i^q z_i^q$, i.e. if C contains *at all* an edge of the form $b_i^p z_i^p$ for some p, then note that C contains either path $z_i^q, u_i^q, v_i^q, z_i^{q+1}$ and covers edge $b_i^{q+1} z_i^{q+1}$, or path $z_i^q, v_i^{q-1}, u_i^{q-1}, z_i^{q-1}$ and covers edge $b_i^{q-1} z_i^{q-1}$. Replacing C by the sector S_q in the first case and by the sector S_{q-1} in the second yields a packing as large as $\mathcal{P}$. The case which remains is the one in which C is the inner cycle $b_i^1, \ldots, b_i^{4m_i}$. In this case, there is at most another cycle in the component, namely the outer cycle C'. Replacing C (and possibly C') by two arbitrarily chosen sectors which are not consecutive and whose edges $u_i^p v_i^p$ are not shared by any test component (two such sectors always exist) yields a packing at least as large as $\mathcal{P}$.

We finally show (5). It is here that we make use of the special structure of Max-2-Sat-3 formulae. Focus on a truth setting component K associated with x_i. Suppose that x_i appears negated in at most one clause c_j (the case in which x_i appears positive in at most one clause being identical). If $\mathcal{P}$ does not contain all even or all odd sectors of K, then replacing the sectors of K in $\mathcal{P}$ by all odd sectors and possibly removing from $\mathcal{P}$ the square in the test component of c_j with one edge in K, yields a packing at least as large as $\mathcal{P}$. □

We conclude this section with the $\mathcal{NP}$-hardness proof for the planar case, reducing the following Planar 3-Sat problem to Cycle Packing. As customary $\{x_1, \ldots, x_n\}$ and $\{c_1, \ldots, c_m\}$ denote, respectively, the set of variables and clauses in a Boolean formula φ

in conjunctive normal form, where each clause has *exactly* 3 literals. Consider the bipartite graph $G_\varphi = (U \cup V, E_\varphi)$, with color classes $U := \{x_1, \ldots, x_n\}$ and $V := \{c_1, \ldots, c_m\}$ and edge set $E_\varphi = \{xc$: variable x occurs in clause $c\}$. The Boolean formula φ is called *planar* when G_φ is planar. PLANAR 3-SAT is the problem of finding, if any, a truth assignment that satisfies all clauses in a planar Boolean formula, where each clause has exactly three literals. It is known that PLANAR 3-SAT is $\mathcal{NP}$-complete [15].

Theorem 2 CYCLE PACKING *is $\mathcal{NP}$-hard for planar graphs with maximum degree 3.*

Proof: The proof follows the same lines as that of Theorem 1, transforming a PLANAR 3-SAT instance φ to a graph $G(\varphi)$ in polynomial time by connecting certain gadgets together. The truth setting component associated with variable x_i is perfectly analogous to the one in the reduction of Theorem 1, displayed in Fig. 2, noting that in this case m_i is not constant in general. The test component associated with a clause c_j is displayed in Fig. 4. Each cycle $r_j^0, r_j^k, s_j^k, t_j^k, r_j^{k+1}$ is called a *pentagon*. This component is connected to truth setting components by sharing the edges ${s^1}_j{t^1}_j$, ${s^2}_j{t^2}_j$ and ${s^3}_j{t^3}_j$, as in the reduction of Theorem 1.

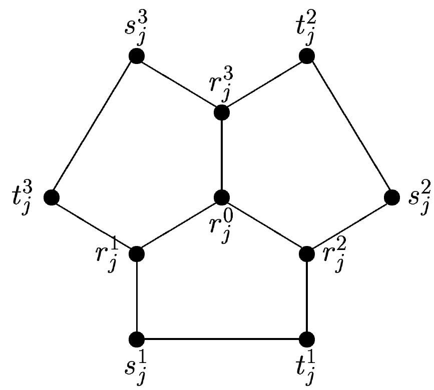

Figure 4: The test component associated with the clause c_j in the $\mathcal{NP}$-hardness proof for the planar case.

In particular, we define the graph $G(\varphi)$ as follows. Consider a planar embedding of G_φ. Such an embedding defines, for each variable x_i a (cyclic) topological order of the clauses in which x_i appears, corresponding to the order in which the associated edges appear in clockwise order around node x_i in the planar embedding. Analogously, for each clause c_j, the embedding defines a topological order of the variables appearing in c_j.

Starting from G_φ and its planar embedding, we first replace each edge x_ic_j by two parallel edges $(x_ic_j)^1$ and $(x_ic_j)^2$. Then, for each variable x_i appearing, say, in clauses $c_1, \ldots, c_{m_i}$, given in topological order, we replace the corresponding node by the truth setting component. Within this operation, for $j = 1, \ldots, m_i$, edges $(x_ic_j)^1$ and $(x_ic_j)^2$ are replaced, respectively, by edges $u_i^{4j-2}c_j$ and $v_i^{4j-2}c_j$ if x_i appears positive in c_j, and by edges $u_i^{4j-1}c_j$ and $v_i^{4j-1}c_j$ if x_i appears negated in c_j. Finally, we replace each clause c_j, containing, say, variables x_1, x_2, x_3, again given in topological order, and incident edges $u_i^{q_i}c_j$, $v_i^{q_i}c_j$ $(j = 1, 2, 3)$, by the four nodes r_j^i, $i = 0, 1, 2, 3$. Here, for $i = 1, 2, 3$, we introduce edges $r_j^0 r_j^i$ and replace edges $u_i^{q_i}c_j$, $v_i^{q_i}c_j$ by $u_i^{q_i}r_j^i$, $v_i^{q_i}r_j^{i+1}$ (with $r_j^4 \equiv r_j^1$). (Note that $s_j^i \equiv u_i^{q_i}$ and $t_j^i \equiv v_i^{q_i}$ for $i = 1, 2, 3$.) The resulting graph is the required $G(\varphi)$.

It is immediate to see that all the operations in the definition of $G(\varphi)$ from G_φ preserve planarity, i.e. $G(\varphi)$ is planar.

We now show that $G(\varphi)$ has a packing of size $\sum_{i=1}^{n} 2m_i + m$ if and only if φ is satisfiable.

The "if" part being trivial, we show the remaining. Let $\mathcal{P} = C_1, \ldots, C_\ell$ be a packing of edge-disjoint cycles with $\ell := \sum_{i=1}^{n} 2m_i + m$. We show that $\mathcal{P}$ can be transformed in polynomial time into "canonical" packing $\mathcal{Q}$ that is at least as large as $\mathcal{P}$ and moreover satisfies Conditions (1)–(5) in the proof of Claim 3 above, with "square" replaced by "pentagon" in (2) and (3). As in the proof of Theorem 1, Conditons (2) and (3) follow immediately from (1). Moreover, the proof of (4) is identical since that proof holds for any m_i, whereas (5) is immediately impled here by the requirement on the size of the packing.

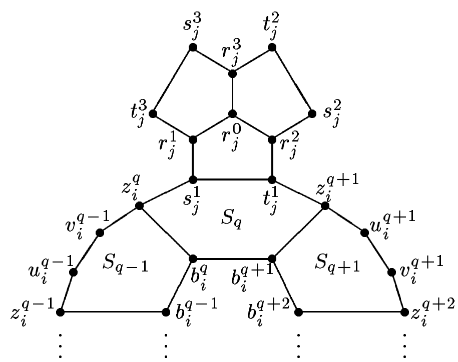

Figure 5: Reference picture for the case analysis in the proof of Claim 4.

Therefore, we only need to prove (1).

Claim 4 *Given a packing $\mathcal{P}$ we can find in polynomial time a packing $\mathcal{P}'$ which is at least as large and in which each cycle is either fully contained in some test component or in some truth setting component.*

Proof: The proof is analogous to that of Claim 2, but the case analysis is more involved. We consider again a cycle C in $\mathcal{P}$ having at least one edge in a truth setting component, say associated with variable x_i, and one edge in a test component, say associated with clause c_j, and assume w.l.o.g. that C contains edge $r_j^1 s_j^1 \equiv r_j^1 u_i^q$ for some $q \in \{1, \ldots, 4m_i\}$. As before, we consider all possible cases and show how to replace C (and possibly other cycles) in $\mathcal{P}$ by pentagons and sectors so as to get a packing at least as large as $\mathcal{P}$. Note that c_j contains exactly three literals in this case. The reference picture is Fig. 5.

Case 1: C contains also edges $s_j^1 t_j^1$ and $t_j^1 z_i^{q+1}$ and hence covers edges $r_j^1 r_j^0$, $r_j^2 t_j^1$ and $z_i^q s_j^1$.

Case 1.1: Edge $r_j^0 r_j^2$ is not contained in a cycle in $\mathcal{P} \setminus \{C\}$. In this case, a packing at least as large as $\mathcal{P}$ is obtained by replacing C by the pentagon $s_j^1, r_j^1, r_j^0, r_j^2, t_j^1$.

Case 1.2: There exists a cycle C' in $\mathcal{P} \setminus \{C\}$ containing edge $r_j^0 r_j^2$. In this case, C' contains also edges $r_j^2 s_j^2$ and $r_j^0 r_j^3$ and C contains also edge $r_j^1 t_j^3$.

Case 1.2.1: C contains also path $z_i^{q+1}, u_i^{q+1}, v_i^{q+1}, z_i^{q+2}$ and hence covers edge $z_i^{q+2} b_i^{q+2}$.

Case 1.2.1.1: Edge $b_i^{q+1} b_i^{q+2}$ is not contained in any cycle in $P \setminus \{C\}$. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C by the sector S_{q+1}.

Case 1.2.1.2: Edge $b_i^{q+1}b_i^{q+2}$ is contained in a cycle C'' in $P \setminus \{C\}$ (possibly $C'' \equiv C'$). In this case, C'' contains also edges $b_i^q b_i^{q+1}$ and $b_i^{q+2}b_i^{q+3}$, as well as either edge $b_i^q b_i^{q-1}$ or path $b_i^q z_i^q v_i^{q-1} u_i^{q-1} z_i^{q-1}$. In both cases, no edge in path $z_i^q v_i^{q-1} u_i^{q-1} z_i^{q-1}$ is contained in any cycle in $P \setminus \{C, C'\}$, and a packing as large as $\mathcal{P}$ is obtained by replacing C, C' by the sectors S_{q-1} and S_{q+1}.
Case 1.2.2: C contains also edge z_i^{q+1}, b_i^{q+1}.
Case 1.2.2.1: C contains also edge $b_i^{q+1}b_i^q$ and hence covers edge $b_i^q z_i^q$. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C by the sector S_q.
Case 1.2.2.2: C contains also edge $b_i^{q+1}b_i^{q+2}$ and covers edge $b_i^{q+2}z_i^{q+2}$. In this case, no edge in the path $z_i^{q+1}, u_i^{q+1}, v_i^{q+1}, z_i^{q+2}$ is contained in any cycle in $\mathcal{P}$, and a packing as large as $\mathcal{P}$ is obtained by replacing C by the sector S_{q+1}.

Case 2: C contains also edge $z_i^q s_j^1$ and hence covers edges $r_j^1 r_j^0$ and $s_j^1 t_j^1$.
Case 2.1: Edge $t_j^1 z_i^{q+1}$ is not contained in any cycle in $\mathcal{P} \setminus \{C\}$.
Case 2.1.1: C contains also path $z_i^{q-1}, u_i^{q-1}, v_i^{q-1}, z_i^q$ and hence covers edge $z_i^{q-1}b_i^{q-1}$.
Case 2.1.1.1: Edge $b_i^{q-1}b_i^q$ is not contained in any cycle in $\mathcal{P} \setminus \{C\}$. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C by the sector S_{q-1}.
Case 2.1.1.2: There exists a cycle C' in $\mathcal{P}\setminus\{C\}$ containing edge $b_i^{q-1}b_i^q$. In this case, C' contains also edges $b_i^q b_i^{q+1}$ and $b_i^{q-2}b_i^{q-1}$, as well as either edge $b_i^{q+1}b_i^{q+2}$ or path $b_i^{q+1}, z_i^{q+1}, u_i^{q+1}, v_i^{q+1}, z_i^{q+2}$. In both cases, a packing as large as $\mathcal{P}$ is obtained by replacing C, C' by the sectors S_{q-1} and S_{q+1}.
Case 2.1.2: C contains also edge $z_i^q b_i^q$.
Case 2.1.2.1: C contains also edge $b_i^q b_i^{q+1}$ and hence covers edge $b_i^{q+1}z_i^q$. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C by the sector S_q.
Case 2.1.2.2: C contains also edge $b_i^{q-1}b_i^q$ and hence covers edge $b_i^{q-1}z_i^{q-1}$. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C by the sector S_{q-1}.

Case 2.2: There exists a cycle C' in $\mathcal{P}\setminus\{C\}$ containing edge $t_j^1 z_i^{q+1}$. In this case, C' contains also edge $t_j^1 r_j^2$ and covers edge $r_j^0 r_j^2$.
Case 2.2.1: C contains also path $z_i^{q-1}, u_i^{q-1}, v_i^{q-1}, z_i^q$ and hence covers edge $z_i^{q-1}b_i^{q-1}$.
Case 2.2.1.1: Edge $b_i^{q-1}b_i^q$ is not contained in any cycle in $\mathcal{P} \setminus \{C\}$. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C by the sector S_{q-1}.
Case 2.2.1.2: Edge $b_i^{q-1}b_i^q$ is contained in C'. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C, C' by the pentagon $s_j^1, r_j^1, r_j^0, r_j^2, t_j^1$ and the sector S_{q-1}.
Case 2.2.1.3: There exists a cycle C'' in $\mathcal{P} \setminus \{C, C'\}$ containing edge $b_i^{q-1}b_i^q$. In this case, C'' contains also edges $b_i^q b_i^{q+1}$. $b_i^{q+1}b_i^{q+2}$ and covers edge $b_i^{q+2}z_i^{q+2}$, and C' contains path $z_i^{q+1}, u_i^{q+1}, v_i^{q+1}, z_i^{q+2}$. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C, C', C'' by the pentagon $s_j^1, r_j^1, r_j^0, r_j^2, t_j^1$ and the sectors S_{q-1}, S_{q+1}.
Case 2.2.2: C contains also edge $z_i^q b_i^q$.
Case 2.2.2.1: C contains also edge $b_i^{q-1}b_i^q$ and hence covers edge $b_i^{q-1}z_i^{q-1}$. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C by the sector S_{q-1}.
Case 2.2.2.2: C contains also path $b_i^q, b_i^{q+1}, b_i^{q+2}$ and C' path $z_i^{q+1}, u_i^{q+1}, v_i^{q+1}, z_i^{q+2}$. In this case, a packing as large as $\mathcal{P}$ is obtained by replacing C, C' by the pentagon $s_j^1, r_j^1, r_j^0, r_j^2, t_j^1$

and the sector S_{q+1}. □

Now, since for each truth setting component, at most half of its sectors can be packed, $\sum_{i=1}^{n} 2m_i + m$ is an upper bound on the size of any packing of $G(\varphi)$. If we have a packing of this size then, for each test component, exactly $2m_i$ sectors are in the packing and they all have the same parity. Therefore the packing corresponds to a truth assignment for φ that satisfies all clauses. □

Corollary 1 Cut Packing *is $\mathcal{NP}$-hard for planar graphs.*

2 An $O(\log n)$ approximation algorithm

Given the results above, we investigate the approximability of the problem. Recalling that the shortest cycle in a graph can be found efficiently, consider the following intuitive idea: repeatedly pick a cycle of smallest length in the solution and remove the associated edges. The resulting algorithm will be called *basic greedy.* As such, this algorithm does not work well. The simple example in Fig. 6 demonstrates that the approximation guarantee can be as bad as $\Omega(\sqrt{n})$, and in fact this is (asymptotically) the worst case. This is stated formally in the following.

Theorem 3 *The approximation guarantee of basic greedy is* $\Theta(\sqrt{n})$.

Proof: We first illustrate the bad example. A *sunflower* (see Fig. 6) is a graph S consisting of a *core* cycle $C = v_1, \ldots, v_p$ and of p *petals*, namely p cycles $P_i = v_i, u_1^i, \ldots, u_{p-2}^i, v_{i+1}$, where indices are modulo p. Note that $n_S = \Theta(p^2)$. The optimum packing consists of the p petals, therefore $\psi_S = p$. Basic greedy on the other hand, may select the core first. Thereafter there is only the *external* cycle remaining, $E := v_1, u_1^1, \ldots, u_{p-2}^1, v_2, u_1^2, \ldots, u_{p-2}^2, v_3, \ldots, v_1$. The approximation guarantee is therefore $\Omega(\sqrt{n})$.

In order to show that the approximation guarantee is $O(\sqrt{n})$, let C be the first cycle found by basic greedy and let $\mathcal{S}$ be any maximum packing of cycles. We prove that C intersects (i.e. has at least one edge in common with) at most $O(\sqrt{n})$ cycles of $\mathcal{S}$. This shows that the optimal Cycle Packing value is decreased by at most $O(\sqrt{n})$ by taking C in the solution, as is done in the first iteration of basic greedy. Iterating this argument on the graph obtained by removing C yields the claim.

If C has at most $\sqrt{n}$ edges, we are done. Otherwise, assuming $|C| > \sqrt{n}$, let $S_1, \ldots, S_k$ be the cycles in $\mathcal{S}$ intersected by C and G' be the subgraph of G induced by the nodes visited by at least one cycle among $C, S_1, \ldots, S_k$. Consider a breadth-first search over G' which starts from all nodes visited by C, i.e. contains all these nodes at level 0. At level i ($i > 0$) of the breadth-first search we have the nodes in some cycle among $S_1, \ldots, S_k$ that are at distance i from the closest node of C in G'.

Now, let ℓ be the smallest level in which there is a node u having at least two neighbors in levels $\leq \ell$. This means that there are two distinct paths from the nodes at level 0 to u, of length at most ℓ and $\ell + 1$, respectively. Since all nodes at level 0 are in C, this means that there is a cycle visiting u whose length is at most $2\ell + |C|/2 + 1$. The fact that C is a shortest cycle in G' implies $|C| \leq 2\ell + |C|/2 + 1$, i.e. $\ell = \Omega(|C|)$. Since $S_1, \ldots, S_k$ are edge disjoint, there are $\Omega(k)$ nodes in level 1. Moreover, since each node in levels $2, \dots, \ell-1$ is adjacent to exactly one node in the previous level, there are $\Omega(k)$ nodes in each level $0, \dots, \ell-1$. This implies

$$n \geq n_{G'} = \Omega(\ell k) = \Omega(|C|k).$$

Since $|C| = \Omega(\sqrt{n})$, we get $k = O(\sqrt{n})$, which completes the proof. $\square$

We next show that a small modification of basic greedy, called *modified greedy* works "reasonably well". We use the following well-known result (see for example [5]):

Fact 1 *For a graph G with $d_G \geq 3$, $g_G \leq 2\lceil \log n \rceil$.*

Modified greedy iteratively performs the following three steps: (1) while G contains a node w of degree 1, w is removed from G along with the incident edge; (2) while G contains a node w of degree 2, with neighbors u and v, w is removed from G along with the incident edges, and edge uv is added to G (note that edge uv may already be present, in which case a cycle of length 2 is formed, which is removed and added to the solution); (3) an arbitrarily chosen shortest cycle C of G is added to the solution and the corresponding edges are removed from G. Steps (1), (2) and (3) are repeated until there are no edges left in G.

Theorem 4 *Modified greedy is a $O(\log n)$-approximation algorithm for* Cycle Packing.

Proof: This proof is similar to the second part of the proof of Theorem 3, based on the observation that the first cycle C found by modified greedy intersects at most $O(\log n)$ cycles of a maximum packing $\mathcal{S}$. In particular, C is taken from the graph G_C obtained from G by applying Steps (1) and (2). Since $d_{G_C} \geq 3$ and $n_{G_C} \leq n_G = n$, by Theorem 1 C has $O(\log n)$ edges. This completes the proof along with the observation that Steps (1) and (2) do not change the optimal Cycle Packing value. $\square$

If one wants to get rid of the $O(\cdot)$ notation in the approximation ratio, it is easy to see that the approximation guarantee is $(2 + o(1)) \log n$.

We do not know of any examples for which the approximation ratio achieved by modified greedy is $\Omega(\log n)$. We remark that if g_G is $\Omega(\log n)$ it is fairly easy to show that the approximation guarantee of modified greedy is constant. The same holds if g_G is bounded by a constant during the whole execution of modified greedy. What we are able to show is the following.

Theorem 5 *The approximation guarantee of modified greedy is $\Omega\left(\sqrt{\frac{\log n}{\log \log n}}\right)$.*

Proof: In order to build the bad example we shall make use of the sunflower graph S described above. Recall that C denotes the core cycle and $P_1, \dots, P_p$ denote the petals. Consider a $p(p-2)$-regular graph H having girth $p(p-1)$. That such graphs exist follows from the following fact (see for example [5]).

Fact 2 *For every positive integers $d \geq 3$, g and n, if $\log_{d-1} n \geq g$ then there exists a d-regular graph G with n nodes such that $g_G \geq g$.*

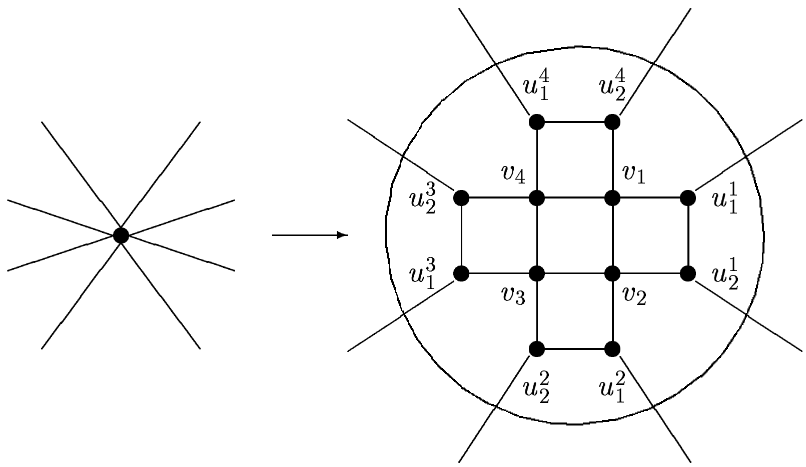

Figure 6: The sunflower graph (within the circle on the right) and the construction of a bad example for modified greedy.

Therefore, setting $d := p(p-2)$ and $g := p(p-1)$, there exists a d-regular graph H with girth at least g, provided that n_H has, say, $(p^2)^{p^2}$ nodes. Given H, we replace each node w of H with a copy of the sunflower S, denoted as S_w (see Fig. 6). The degree-2 nodes of different copies of S are connected as follows. If $uv \in E(H)$, connect an arbitrarily chosen degree-2 node of S_u to an arbitrarily chosen degree-2 node of S_v, provided that no node has ever degree greater than 3. Let G be the resulting graph.

The optimal Cycle Packing solution for G has value at least pn_H. A solution of this size can be obtained by taking cycles $P_1, \ldots, P_p$ for each copy of S. On the other hand, modified greedy applied to G may for the first n_H iterations select the n_H core cycles in each copy of S. Afterwards, as we will show, the optimal solution value for the left-over graph is at most $\frac{3}{2}n_H$. To see this, consider the graph G' obtained by removing all the core cycles from G, *without removing degree 2 nodes*. We have $g_{G'} = p(p-1)$, since the only cycle left within each copy of S is the external cycle, whose length is $p(p-1)$, while the length of any cycle visiting more than one copy of S is at least equal to the girth of H. We have that,

$$m_{G'} = m_H + (\#\text{ of edges remaining in all copies of } S) = \frac{1}{2}p(p-2)n_H + p(p-1)n_H.$$

The optimal Cycle Packing solution cannot be larger than the ratio between the number of edges and the girth, i.e.

$$\psi_{G'} \leq \frac{m_{G'}}{g_{G'}} \leq \frac{\frac{1}{2}p(p-2)n_H + p(p-1)n_H}{p(p-1)} \leq \frac{3}{2}n_H.$$

Therefore, overall greedy may find a solution of value at most $\frac{5}{2}n_H$, with an approximation ratio of

$$\frac{pn_H}{\frac{5}{2}n_H} = \frac{2p}{5}.$$

Now, the relation $n_H \leq (p^2)^{p^2}$ implies

$$p \geq \sqrt{\frac{\log n_H}{\log \log n_H}}$$

as $\left(\frac{\log n_H}{\log \log n_H}\right)^{\frac{\log n_H}{\log \log n_H}} \leq n_H$ if and only if $\frac{\log n_H}{\log \log n_H}(\log \log n_H - \log \log \log n_H) \leq \log n_H$, which is immediately verified. Hence, noting that $\log n_H = \Theta(\log n_G)$, we get the desired claim. $\square$

3 Constant approximation guarantee for "nonsparse" graphs

In this section, we consider the approximation ratio achieved by a version of basic greedy for graphs that have (asymptotically) "many more" edges than nodes. More precisely, given an integer constant $t \geq 2$ and constants $\alpha, \delta > 0$, let $\mathcal{G}(t, \alpha, \delta)$ be the family of graphs with n nodes and m edges such that

$$m \geq \alpha n^{1+\frac{1}{t}+\delta}. \tag{1}$$

We will show that a constant approximation ratio can be achieved for these graphs, where the constant depends only on t.

Our approach is based on showing that, if n is "not too small" (i.e. not bounded by a constant) the girth of the considered graph G will remain "small" for a large number of iterations. In particular, our proof will be based on the following result, given in [5], p. 158:

Theorem 6 *For every graph G with*

$$m > 90tn^{1+\frac{1}{t}},$$

where t is an integer ≥ 2, $g_G \leq 2t$.

Moreover, we will use the following obvious remark: if n is bounded by a constant, Cycle Packing can be solved in constant time.

The algorithm, called *dense greedy*, is as follows. Let $\varepsilon > 0$ be arbitrary, but fixed. If

$$n \leq \left(\frac{90t(1+\varepsilon)}{\alpha\varepsilon}\right)^{\frac{1}{\delta}}, \tag{2}$$

then n is bounded by a constant and dense greedy finds an optimal solution by complete enumeration. Otherwise, dense greedy simply proceeds as basic greedy in the previous section.

Theorem 7 *For any fixed $\varepsilon > 0$, dense greedy is a ρ-approximation algorithm for* Cycle Packing *on graphs in $\mathcal{G}(t, \alpha, \delta)$, where*

$$\rho := (1+\varepsilon)\frac{2t}{3}.$$

The running time is $O(c_{t\alpha\delta\varepsilon} + p(n))$, where $c_{t\alpha\delta\varepsilon}$ is a constant depending on $t, \alpha, \delta, \varepsilon$ and $p(n)$ is a polynomial independent of $t, \alpha, \delta, \varepsilon$.

Proof: Clearly, the constant $c_{t\alpha\delta\varepsilon}$ corresponds to the time taken to solve by complete enumeration CYCLE PACKING on graphs with a "small" number of nodes (see (2)). Hence, the only case to be considered in the proof is the one in which (2) does not hold, i.e.

$$n > \left(\frac{90t(1+\varepsilon)}{\alpha\varepsilon}\right)^{\frac{1}{\delta}},$$

which is equivalent to

$$\alpha n^{1+\frac{1}{t}+\delta} > \frac{90t(1+\varepsilon)}{\varepsilon} n^{1+\frac{1}{t}}.$$

In other words, we have

$$m \geq \alpha n^{1+\frac{1}{t}+\delta} > \frac{90t(1+\varepsilon)}{\varepsilon} n^{1+\frac{1}{t}}.$$

Consider the *last* iteration of basic greedy, say the p-th, in which the remaining number of edges (after removal of the edges in the cycles found) exceeds $90tn^{1+\frac{1}{t}}$. This means that, in iterations $1, \ldots, p$, basic greedy found p cycles and that, after removal of these cycles, the number of edges is at most $90tn^{1+\frac{1}{t}}$. In other words, these p cycles contain, overall, at least

$$m - 90tn^{1+\frac{1}{t}} > \frac{1}{1+\varepsilon} m$$

edges. Moreover, by Theorem 6 the length of each of these cycles is at most $2t$.

Therefore, the number of cycles found by dense greedy is at least

$$p \geq \frac{\frac{1}{1+\varepsilon} m}{2t}. \tag{3}$$

Furthermore, the optimal CYCLE PACKING solution ψ_G has value at most $\frac{m}{3}$ as each cycle in G contains at least 3 edges (recall that G is simple). Therefore,

$$\frac{\psi_G}{p} \leq \frac{\frac{m}{3}}{\frac{\frac{1}{1+\varepsilon} m}{2t}} = (1+\varepsilon)\frac{2t}{3}.$$

□

Note that for $t = 2$, i.e. for all graphs for which $m = \Omega(n^{\frac{3}{2}+\delta})$ for some $\delta > 0$, which include dense graphs, the approximation achieved is arbitrarily close to $\frac{4}{3}$.

A reduction of the constant $c_{t\alpha\delta\varepsilon}$ can be achieved by using results which are tighter than Theorem 6, limiting the complete enumeration to smaller values of n than the right-hand side of (2). (For instance, it is known that every graph with at least $\frac{n}{4}(1 + \sqrt{4n-3})$ edges has a cycle of length 4.) However, the exact determination of the minimum value of the right-hand side in (2) that preserves the approximation guarantee is out of the scope of this paper.

Note that the use of modified greedy instead of basic greedy in the above scheme would yield the same guarantee.

4 Cycle Packing and Linear Programming

A simple, natural *Integer Linear Programming* (ILP) formulation of Cycle Packing is the following. Let $\mathcal{C}$ be the family of all cycles of G. We associate with each $C \in \mathcal{C}$ a binary variable x_C equal to 1 if and only if cycle C is in the optimal Cycle Packing solution. The ILP model reads

$$\max \sum_{C \in \mathcal{C}} x_C, \tag{4}$$

subject to

$$\sum_{C \ni e} x_C \leq 1, \qquad e \in E, \tag{5}$$

$$x_C \geq 0, \qquad C \in \mathcal{C}, \tag{6}$$

$$x_C \text{ integer}, \qquad C \in \mathcal{C}. \tag{7}$$

The associated LP relaxation (4)–(6) can be solved in polynomial time as its dual reads

$$\min \sum_{e \in E} y_e, \tag{8}$$

subject to

$$\sum_{e \in C} y_e \geq 1, \qquad C \in \mathcal{C}, \tag{9}$$

$$y_e \geq 0, \qquad e \in E, \tag{10}$$

and the separation problem for (9) calls for a cycle of minimum weight (the weighted counterpart of the girth), which can be found by shortest path techniques if weights are nonnegative. Note also that each optimal integer solution of the dual corresponds to a set of edges that intersects each cycle at least once, i.e. to the complement of a spanning tree of G. The corresponding value is $m - n + 1$.

A question closely related with the approximability of Cycle Packing is how large can be the difference between the value of integer and fractional solutions of (4)–(6). In particular, let ψ_G^* be the optimal value of LP (4)–(6) and define the *integrality gap* as the ratio between ψ_G^* and ψ_G. The analysis of modified greedy in Section 2 shows an upper bound on this gap. In fact, this result was already shown by Erdös and Pósa [9]. Here we give a constructive proof analogous to theirs.

Theorem 8 *For any graph G, $\psi_G^* \leq O(\log n) \cdot \psi_G$.*

Proof: We already noted that $\psi_G^* \leq m - n + 1$. We show that $m - n + 1 \leq O(\log n) \cdot \psi_G$, which yields the proof. In particular, we show that $h \geq \frac{m-n+1}{O(\log n)}$, where h is the value of the solution produced by modified greedy. Indeed, at each iteration of modified greedy, h is increased by 1. Moreover, in Steps (1) and (2) (removal of nodes of degree 1 and 2) the quantity $m - n + 1$ is not changed. Finally, in Step (3) m is decreased by $O(\log n)$ and n is unchanged. Hence, every time h is increased by 1, $m - n + 1$ is decreased by $O(\log n)$. □

A main question is, again, whether the $O(\log n)$ bound is tight. Clearly, Fact 2 mentioned in Section 2 shows that $\psi_G = \Theta(\log n) \cdot (m - n + 1)$ for some 3-regular graph G for which

$g_G = \Theta(\log n)$. However, since $\psi^*_G \leq \frac{m}{g_G}$ (note that $y^*_e := \frac{1}{g_G}$ is a feasible dual solution) and $\psi_G \geq \frac{m}{O(\log n)}$, $g_G = \Theta(\log n)$ implies that the integrality gap for G is bounded by a constant. Graph $K_{3,3}$ shows that the integrality gap is at least 2.25. In particular, among the graphs G with $\psi_G = 1$, $K_{3,3}$ is the one for which $\psi^*_G (= \frac{9}{4})$ is largest.

Note that the proof of Theorem 7 implicitly shows that the integrality gap for "nonsparse" graphs is asymptotically constant, as clearly $\psi^*_G \leq \frac{m}{g_G}$ and $\frac{m}{g_G} \leq O(1) \cdot \psi_G$ as ψ_G tends to infinity.

Acknowledgments

Part of this work was done while the first and third authors were visiting BRICS. The first author was also partially supported by CNR and MURST, Italy. We thank Maxim Sviridenko for helpful discussions on the subject. Moreover, we are grateful to two anonymous referees, whose comments led to an improvement of the presentation as well as to a strengthening (and simplification) of the original version of Theorem 7.

References

[1] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, M. Protasi, *Combinatorial Optimization Problems and their Approximability Properties*, Springer-Verlag, Berlin (1999).

[2] V. Bafna and P.A. Pevzner, Genome Rearrangements and Sorting by Reversals. *SIAM J. on Computing* 25 (1996) 272–289.

[3] B.S. Baker, Approximation Algorithms for $\mathcal{NP}$-Complete Problems on Planar Graphs. *J. ACM* 41 (1994) 153–180.

[4] P. Berman and M. Karpinski, On Some Tighter Inapproximability Results. *Proceedings of the 26th International Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science 1644, Springer-Verlag, Berlin (2000) 200–209.

[5] B. Bollobás, *Extremal Graph Theory*, Academic Press, New-York (1978).

[6] A. Caprara, Sorting Permutations by Reversals and Eulerian Cycle Decompositions. *SIAM J. on Discrete Mathematics* 12 (1999) 91–110.

[7] A. Caprara, A. Panconesi and R. Rizzi, Packing Cuts in Undirected Graphs. Research Report OR/01/14, DEIS, University of Bologna (2001). Available at `http://www.or.deis.unibo.it/alberto/online.htm`

[8] P. Crescenzi, V. Kann, A Compendium of NP Optimization Problems. Available at `http://www.nada.kth.se/~viggo/problemlist/compendium.html`

[9] P. Erdös and L. Pósa. On the Maximal Number of Disjoint Circuits of a Graph. *Publ. Math. Debrecen* 9 (1962) 3–12.

[10] P. Erdös and H. Sachs. Regulare Graphen Gegebener Taillenweite mit Minimaler Knotenzahl. *Wittenberg Math. – Natur. Reine* 12 (1963) 251–257.

[11] A. Frank, Conservative Weightings and Ear-Decompositions of Graphs. *Combinatorica* 13 (1993) 65–81.

[12] M. Grötschel, L. Lovász, A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Springer-Verlag, Berlin (1988).

[13] I. Holyer, The $\mathcal{NP}$-Completeness of Some Edge-Partition Problems. *SIAM J. on Computing* 10 (1981) 713–717.

[14] J. Kececioglu and D. Sankoff, Exact and Approximation Algorithms for Sorting by Reversals, with Application to Genome Rearrangement. *Algorithmica* 13 (1995) 180–210.

[15] D. Lichtenstein, Planar formulae and their uses. *SIAM J. on Computing* 11 (1982) 329–343.

[16] C.H. Papadimitriou and M. Yannakakis (1991), Optimization, Approximation, and Complexity Classes *J. Comput. System Sci.* 43 (1991) 425–440.