Tight Bound for Matching[1]

Yijie Han

School of Computing and Engineering
University of Missouri at Kansas City
5100 Rockhill Road
Kansas City, MO 64110
hanyij@umkc.edu

Abstract

Let M be the number of edges in a maximum matching in graphs with m edges, maximum vertex degree k and shortest simple odd-length cycle length L. We show that

$$M \geq \begin{cases} \frac{m}{2} - \frac{m}{2L}, & \text{if } k = 2 \\ \frac{m}{k} - \frac{m}{(k+L)k}, & \text{if } k > 2 \end{cases}$$

This lower bound is tight.

When no simple odd-length cycle exists it is known previously that $M \geq \frac{m}{k}$.

Keywords: Combinatorial problems, graph algorithms, matching, lower bound.

1 Introduction

Matching is an extensively studied topic. The question we want to study here is the lower bound of a maximum matching in a graph. Earlier researchers studied the problem of the existence of a perfect matching (i.e. a matching of size $n/2$ with n vertices in a graph). Petersen [7] showed that a bridgeless cubic graph has a perfect matching. König [4] showed that there exists a perfect matching in any k-regular bipartite graph. Tutte [8] characterizes when a graph has a perfect matching. For graphs without a perfect matching the size of a maximum matching is studied. Nishizeki and Baybars [6] showed that any 3-connected planar graph has a matching of size at least $(n + 4)/3$ for $n \geq 22$. Biedl et al. [1] raised the question whether a bound better than $m/(2k - 1)$ can be obtained for the size of a maximum matching in a graph of m edges and degree k. Feng et al. [2] showed a lower bound of $2m/(3k - 1)$ (for $k \geq 3$) for this problem. Recently Han [3] showed a lower bound of $4m/(5k + 3)$.

Actually Vizing has shown [9] that graphs with degree k can be edge colored with $k + 1$ colors. Misra and Gries gave a constructive proof of this [5]. This result will give a matching of size $m/(k + 1)$ and is better than the results shown in [1, 2, 3]. The author forgot about this known result and worked after the results in [1, 2]. After presenting the paper [3] at FAW'2008 Daniel Lokshtanov reminded me of Vizing's theorem. After examining our techniques used in [3] I informed Daniel Lokshtanov right way that the technique employed in [3] will allow me to outperform the result obtainable from Vizing's theorem. Here we present our result of a tight bound on matching which outperforms that achievable via Vizing's theorem.

[1] A previous version with weaker results was published in [3].

We prove that

$$M \geq \begin{cases} \frac{m}{2} - \frac{m}{2L}, & \text{if } k = 2 \\ \frac{m}{k} - \frac{m}{(k+L)k}, & \text{if } k > 2 \end{cases}$$

where M is the number of edges in a maximum matching, m is the number of edges, k is the maximum vertex degree and L is the length of the shortest simple odd-length cycle (cycles in which there are no repeating vertices or edges) in the input graph. This bound is better than $m/(k+1)$ which is derivable from Vizing's theorem [9]. We show that this lower bound is tight.

Note that when no odd-length cycle exists the graph can be viewed as a bipartite graph and therefore it can be edge colored with k colors. This will give a matching of $\geq m/k$ edges.

We believe that the quality of the result presented in this paper is comparable to that of Vizings theorem [9] (every graph with vertex degree k can be edge colored with k or $k+1$ colors.) or that of Königs result [4] (every regular bipartite graph has a perfect matching).

2 The Bound

Assume that a maximum matching is obtained for the input graph with m edges. Vertices incident to an edge in the matching are saturated vertices. Vertices not incident to any edge in the matching are unsaturated vertices. Without loss of generality we can assume that the input graph is connected. We have the following lemma:

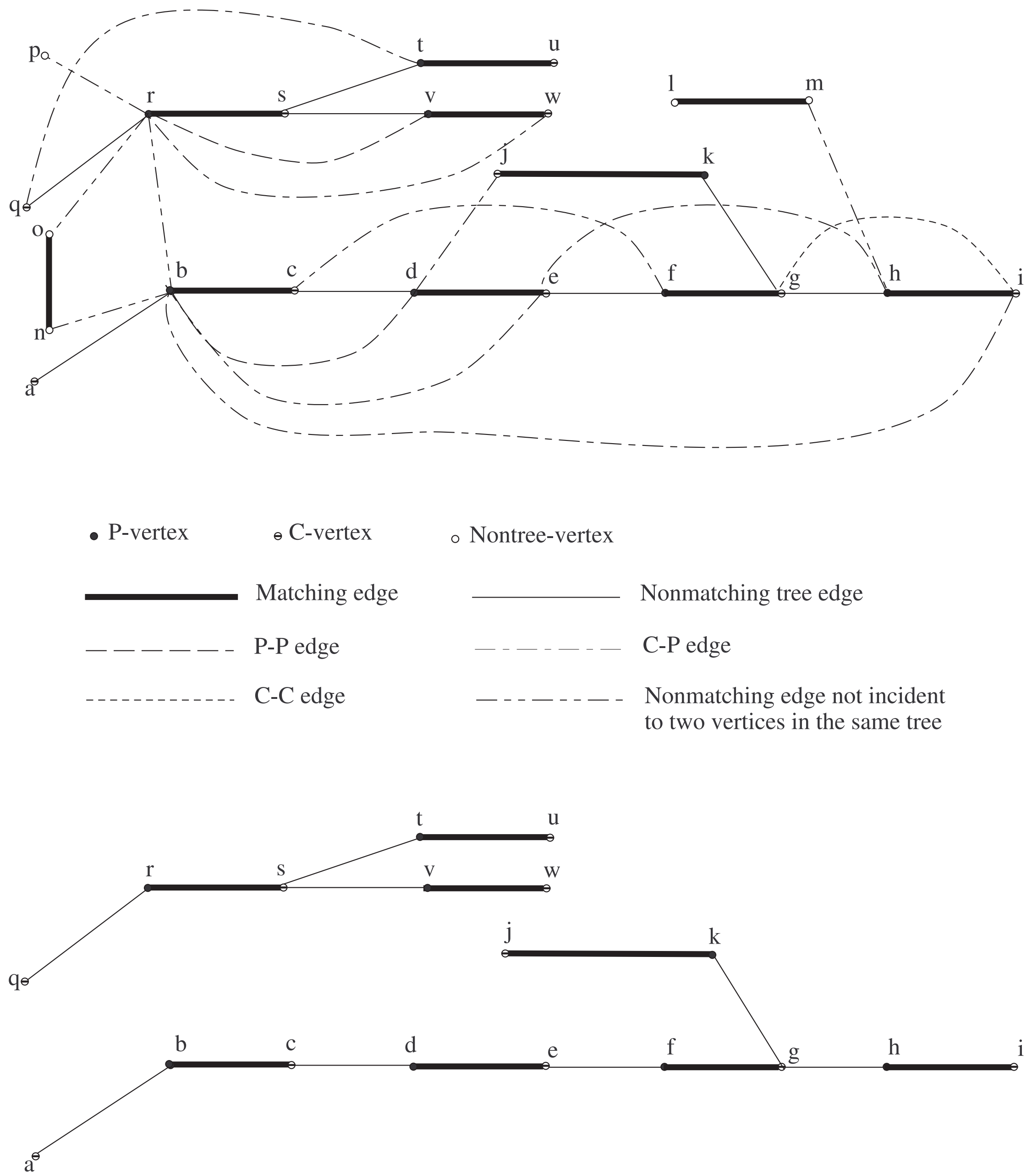
Lemma 1: If the graph has no unsaturated vertex then $M \geq m/k$.

Proof: For each edge e in the matching we can have at most $1 + (2k-2)$ edges incident to e. Among which 1 is the edge in the matching and $2k-2$ are nonmatching edges. However, when we are counting this way the nonmatching edges are counted twice, once from each vertex they are incident to. Therefore we have that $M(2 + (2k-2)) \geq 2m$, i.e. $M \geq m/k$. □

Therefore we assume that there is at least one unsaturated vertex.

We now perform depth-first search along alternating paths starting from each unsaturated vertex. After starting from an unsaturated vertex and building a depth-first tree T along alternating paths, we remove T and all edges incident to vertices in T from the input graph. We then start building another tree starting from another unsaturated vertex. We keep doing this until no trees of alternating paths with root being unsaturated vertices can be built. Fig. 1. illustrates this. For each tree T obtained we orient each matching edge in the tree from the parent to the child. Therefore one vertex of each matching edge in a tree is a parent and the other vertex of the matching is a child. We also classify the roots of all trees as children vertices. Thus for all vertices in the trees they are either parent vertices (call them P-vertices) or child vertices (call them C-vertices). Notice that all edges at odd levels in a tree are nonmatching edges and all edges at even levels in the tree are matching edges. Also notice that each P-vertex has only one child and each C-vertex can have none or one or more than one children. See Fig. 1.

Every nontree edge connecting P-vertices and/or C-vertices in the same tree can be classified as a C-P edge with one end being a C-vertex and the other end being a P-vertex, C-C edge with both ends being C-vertices, or a P-P edge with both ends being P-vertices. (Note that there could



Trees generated from the graph above.

Fig. 1.

be edges which are not incident to two vertices in the same tree. These edges are not classified here.) See Fig. 1. for an example. We have the following lemmas:

Lemma 2: There cannot be an alternating path connecting a C-vertex in one tree to a C-vertex in another tree.

Proof: For otherwise there is an augmenting path from the root of one tree to the root of the other tree. □

Lemma 3: There cannot be an alternating path containing at least one matching edge, disjoint from a tree and yet connects a C-vertex of the tree to another vertex of the tree.

Proof: If such an alternation path exists the matching edges in it will become part of the tree because of the depth-first search property. □.

The loop formed starting from a C-vertex a and goes to another C-vertex b in the same tree plus the nontree C-C edge (a, b) (when exist) is called an enclosing loop. Here a could be either an ancestor or a descendant of b, or a and b could be in different branches of the same tree. Any vertex in an enclosing loop is called an enclosing vertex. Note that an enclosing loop must have odd length.

In Fig. 1. there is an enclosing loop g, h, i, g.

Lemma 4: Let T be a tree constructed by the method mentioned above. Let a be the root of T. Let b be an enclosing vertex in T (note that b can be either a P-vertex or a C-vertex). Then there cannot be an alternating path starting from an unsaturated vertex c other than a and reaching b when this alternating path first reaches T.

Proof: If such a path exists then an augmenting path or an augmenting tree can be formed. Note that without an enclosing loop such an augmenting path also exists when b is a C-vertex. When b is a P-vertex this enclosing loop allows us to "go the other way around" and form an augmenting path or augmenting tree. □

In Fig. 2. an alternating path is added which starts from an unsaturated vertex A and leads to the enclosing vertex h. Fig. 2 shows the augmenting path resulted.

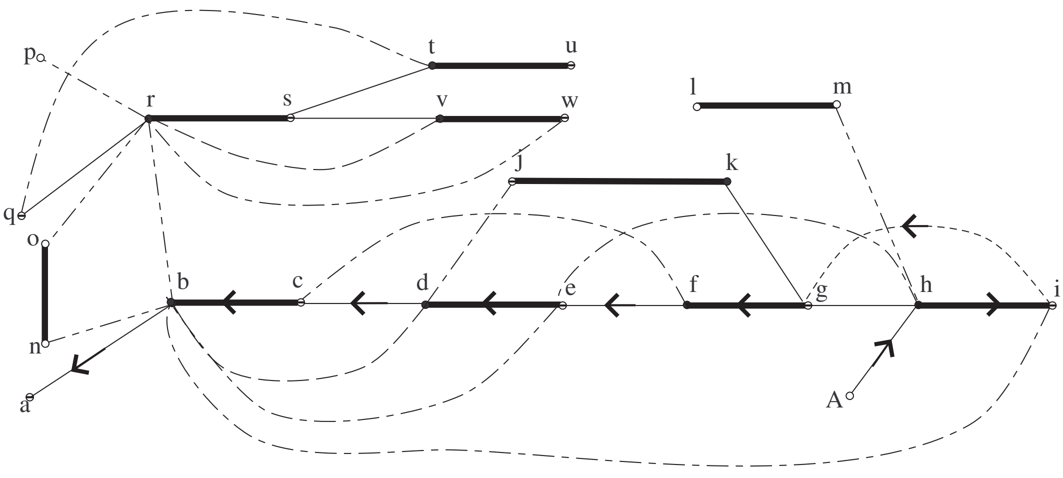


Fig. 2.

The loop formed starting from a C-vertex a and goes to an enclosing P-vertex b in the same tree plus the nontree edge (a, b) (when exists) is called an auxiliary enclosing loop. Vertices in an auxiliary enclosing loop are also enclosing vertices. Here a could be an ancestor or a descendent of b, or a and b could be in different branches of the same tree. Note that Lemma 4 holds for these newly defined enclosing vertices as well. Again if the vertex b in Lemma 4 is a C-vertex then an augmenting path exists without enclosing loops and auxiliary enclosing loops in the proof of Lemma 4. When vertex b in Lemma 4 is a P-vertex the auxiliary enclosing loop will allow us to "go the other way around" and form an augmenting path or augmenting tree.

Note that whenever we identify an auxiliary enclosing loop more vertices are identified as enclosing vertices and therefore new additional auxiliary enclosing loops may again be identified.

In Fig. 1. loop e, f, g, h, e is an auxiliary enclosing loop because h is an enclosing vertex. This makes e and f enclosing vertices in addition to g, h, i. Now loop c, d, e, f, c is an auxiliary enclosing loop because f is an enclosing vertex. This adds e, d to enclosing vertices. Now d, e, f, g, k, j is an auxiliary enclosing loop because d is an enclosing vertex. This adds k, j to enclosing vertices. Thus we have $c, d, e, f, g, h, i, j, k$ as enclosing vertices in the graph. In Fig. 3. an alternating path is added which starts from an unsaturated vertex A and leads to the enclosing vertex d which is in an auxiliary enclosing loop. Fig. 3. shows the augmenting path resulted.

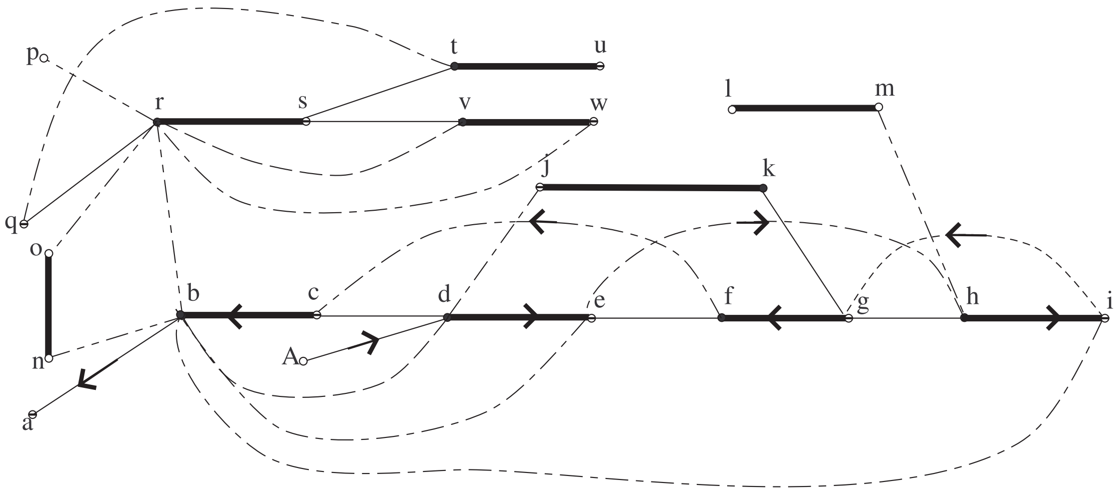


Fig. 3.

Vertices on trees which are not enclosing vertices are called non-enclosing vertices. Note that there could be also vertices which are not in any tree and therefore they are not identified as either enclosing or non-enclosing. In Fig. 1. $c, d, e, f, g, h, i, j, k$ are enclosing vertices, $a, b, q, r, s, t, u, v, w$ are non-enclosing vertices, l, m, n, o, p are nontree vertices. Also note that by the way we forming enclosing loops and auxiliary enclosing loops the boundary enclosing vertex adjacent to a non-enclosing vertex must be a C-vertex.

For each non-enclosing P-vertex a we can associate all nonmatching edges incident to a to the matching edge incident to a. This represents a ratio of at most k edges versus 1 matching edge. We can thus remove all edges incident to a. For each non-enclosing C-vertex b each nonmatching edge incident to b cannot have the other end incident to an enclosing vertex or a C-vertex in the same tree (for otherwise it will form an enclosing loop or an auxiliary enclosing loop), neither can it have the other end incident to an enclosing vertex in another tree (see Lemma 4), neither can it have the other end incident to a saturated nontree vertex (because of depth-first search), neither can it have the other end incident to an unsaturated vertex other than the root of tree (for otherwise there is an augmenting path). Therefore each nonmatching edge incident to b has the other end incident to a non-enclosing P-vertex c (not necessarily in the same tree). Thus the nonmatching edge (b, c) can be associated with the matching edge incident to c, and then it can also be removed.

In this way we remove all edges incident to non-enclosing vertices. We then remove all non-enclosing vertices. What remain are enclosing vertices and nontree vertices. If a connected component C of the remaining graph contains nontree vertices only then the ratio of the number of edges in C and the number of matching edges in C is no large than $k/1$ because C has no unsaturated vertices (see Lemma 1). Therefore such a component can be removed. (From this we know that if there are no enclosing loops then $M \geq m/k$.)

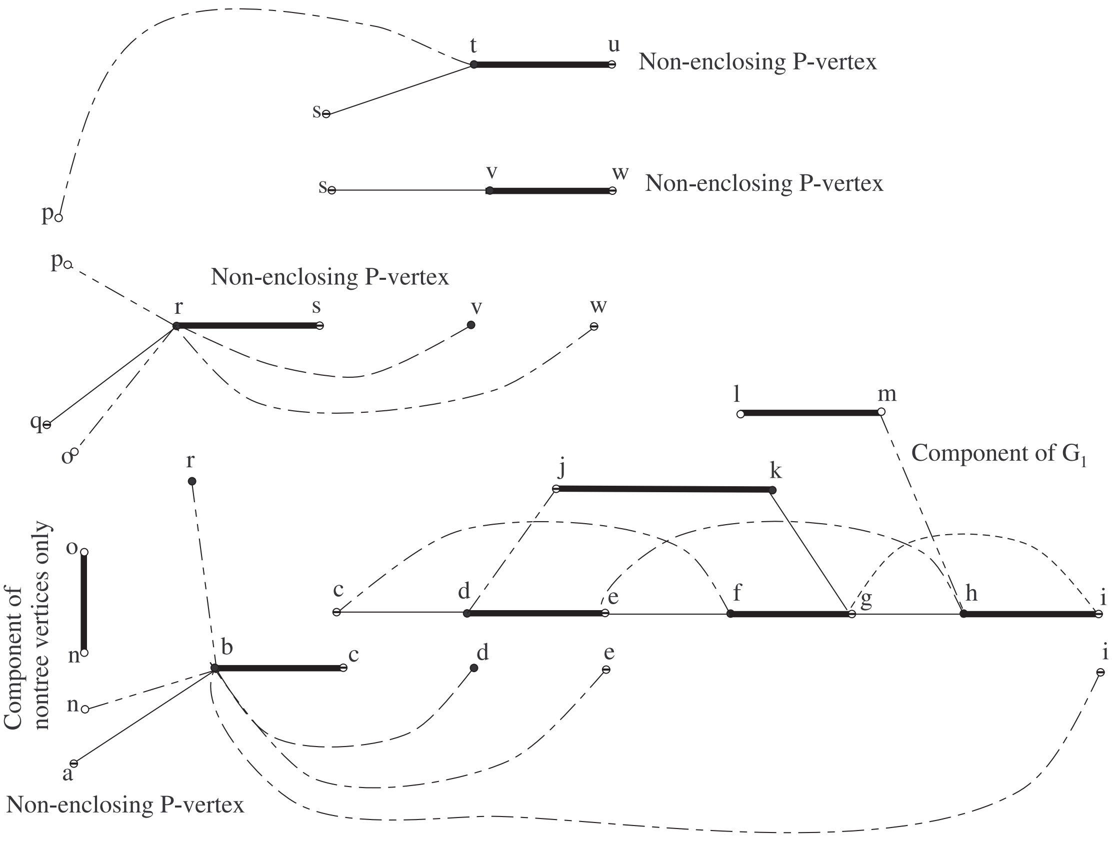
Fig. 4. shows the decomposition of Fig. 1. In Fig. 4. there are four (connected) components labeled "Non-enclosing P-vertex". In each of these components the nonmatching edges (only incident to the P-vertex) are associated with the matching edge with ratio no more than $k/1$. These components can be removed as we stated above. Note edge (r, v) can be associated with either r or v and edge (r, b) can be associated with either r or b. Component having vertices n and o is a component of nontree vertices only. This component can be removed also.

Let the remaining graph be G_1. Each connected component of G_1 contains at least one enclosing loop (for otherwise it contains nontree vertices only and is already treated) and cannot contain vertices from more than one tree because of Lemma 4 (note here all these vertices from both trees are enclosing vertices because non-enclosing vertices have been removed). Fig. 4. shows a component of G_1.

Now for each connected component C of G_1 there is a lowest common ancestor A in the original tree (before removing non-enclosing vertices) where these enclosing vertices of C are in. A is a root in G_1. A must be a C-vertex because it is a boundary vertex. Such a vertex A is called an outer vertex. There is exactly one outer vertex in each connected component of G_1 because of the way enclosing loop and auxiliary enclosing loops are formed and the way non-enclosing vertices are removed. There is at least one enclosing loop *Loop* in C which is uniquely associated with A and not associated with other outer vertices (because there is one outer vertex in each component of G_1 and *Loop* is contained in component C and the other outer vertices belong to other connected components). Also note that outer vertices are the only unsaturated vertices in G_1. In Fig. 4. the outer vertex in the component of G_1 is vertex c.

Let L be the length of the shortest simple odd-length cycle in the input graph. If a component C in G_1 has n vertices then $n \geq L$ because the component contains at least one enclosing loop (we mentioned before that enclosing loops are of odd length).

Note that n has to be odd because if n is even then there is no perfect matching among $n - 1$ vertices. Thus when n is even then, after removing the root (outer vertex) which is an unsaturated vertex in C, there are odd number of vertices left and therefore among them there is another unsaturated vertex, and thus Lemma 4 will be violated. Note also that we have mentioned before



t
u
Non-enclosing P-vertex
s
s
v
w
Non-enclosing P-vertex
p
p
Non-enclosing P-vertex
r
s
v
w
q
o
l
m
Component of G_1
r
j
k
o
Component of nontree vertices only
c
d
e
f
g
h
i
n
b
c
d
e
i
n
a
Non-enclosing P-vertex

Fig. 4.

that L is odd.

If we remove the root of C and incident edges then the matching in the remaining part of C is a perfect matching. Therefore the size of the matching in C is $(n-1)/2$. We analyze the following cases:

Case 1: $k < n - L$.

There can be no more than $nk/2$ edges in C. Thus we have that:

$$\frac{M}{(n-1)/2} \geq \frac{m}{nk/2}$$

i.e.:

$$M \geq \frac{m}{k} - \frac{m}{nk} \geq \frac{m}{k} - \frac{m}{(k+L)k}$$

Case 2: $k \geq n - L$.

If $n = L$:

The number of edges in C is no more than L (only one cycle can exists). Therefore we have:

$$\frac{M}{(L-1)/2} \geq \frac{m}{L}$$

i.e.:

$$M \geq \frac{m}{2} - \frac{m}{2L}$$

It can be verified that when $k > 2$ this leads to:

$$M \geq \frac{m}{2} - \frac{m}{2L} \geq \frac{m}{k} - \frac{m}{(k+L)k}$$

$n = L + 1$ cannot hold because both n and L are odd.

If $n \geq L + 2$:

The number of edges in C cannot be more than $n(n-L)/2$ because of the shortest simple cycle length (we leave readers to verify this for themselves). Therefore we have:

$$\frac{M}{(n-1)/2} \geq \frac{m}{n(n-L)/2}$$

i.e.:

$$M \geq \frac{m}{n-L} - \frac{m}{n(n-L)} = \frac{m(n-1)}{n(n-L)} \geq \frac{m(k+L-1)}{nk}$$
$$\geq \frac{m(k+L-1)}{(k+L)k} = \frac{m}{k} - \frac{m}{(k+L)k}$$

Thus we have:

Theorem 1: Let m be the number edges, k be the maximum degree of any vertex, L be the length of the shortest simple odd-length cycle in the input graph, let M be the number of edges in a maximum matching, then

$$M \geq \begin{cases} \dfrac{m}{2} - \dfrac{m}{2L}, & \text{if } k = 2 \\ \dfrac{m}{k} - \dfrac{m}{(k+L)k}, & \text{if } k > 2 \end{cases} \qquad \square$$

The bound proved in Theorem 1 is reached when $n = L + k$. As we analyzed above that both n and L have to be odd. In Fig. 5 we show a case which demonstrates the tightness of the bound proved in Theorem 1.

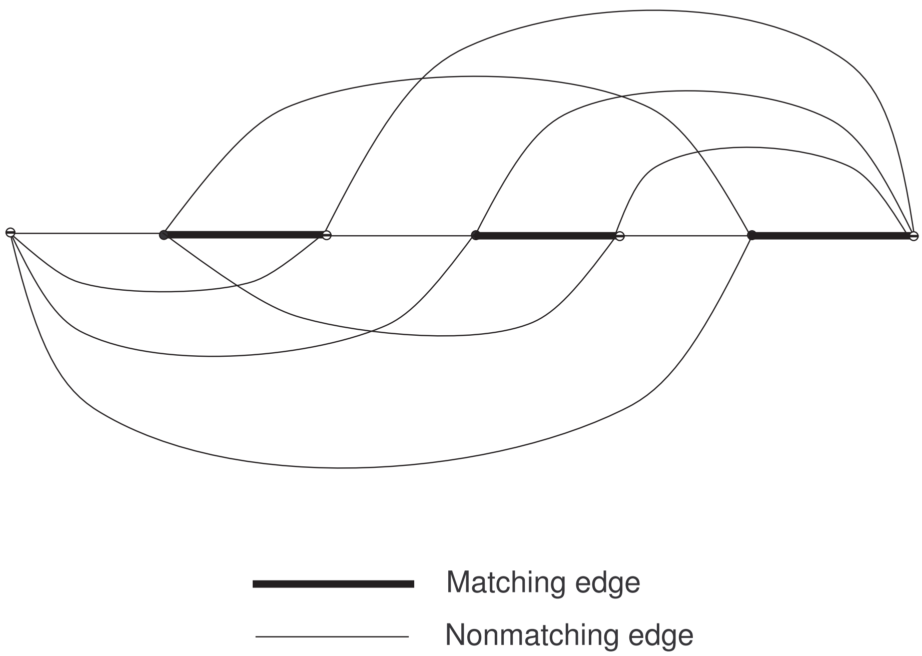


Fig. 5. $3 = M = m/k - m/((k+L)k) = 14/4 - 14/((4+3)*4)$.

The bound provided in Theorem 1 is larger than $m/(k+1)$. Thus Theorem 1 provides a stronger result than that could be provided by Vizing's theorem [9].

Acknowledgment

I would like to thank Daniel Lokshtanov for reminding me of the Vizing's theorem which provided a motive for me to fully exploit the capability of my techniques. Vizing's theorem was known to me before and was cited in my previous publication.

References

[1] T. Biedl, E. Demaine, C. Duncan, R. Fleischer, S. Kobourov. Tight bounds on the maximal and maximum matchings. Discrete Math. 285(1-3), 7-15(2004).

[2] W. Feng, W. Qu, H. Wang. Lower bounds on the cardinality of maximum matchings in graphs with bounded degrees. Proc. 2007 Int. Conf. on Foundations of Computer Science, Las Vegas, 110-113(2007).

[3] Y. Han. Matching for graphs of bounded degree. Proc. Second Int. Workshop on Frontiers in Algorithmics (FAW'2008), Changsha, China, Lecture Notes in Computer Science 5059, 171-173(June 2008).

[4] D. König. Über Graphen und ihre Anwendung auf Determinantentheorie und Mengenlehre. Math. Ann. 77, 453-456(1916).

[5] J. Misra, D. Gries. A constructive proof of Vizings theorem. Information Processing Letters, 41, 131-133(1992).

[6] T. Nishizeki, I. Baybars. Lower bounds on the cardinality of the maximum matchings of planar graphs. Discrete Math. 28(3), 255-267(1979).

[7] J. Petersen. Die Theorie der regulären graphs (The theory of regular graphs). Acta Math. 15, 193-220(1891).

[8] W. T. Tutte. The factorization of linear graphs. J. London Math. Soc. 22, 107-111(1947).

[9] V. G. Vizing. On an estimate of chromatic class of a p-graph. Diskret. Anal. 3(1964), 25-30. (in Russian).