SIAM J. DISCRETE MATH.
Vol. 38, No. 2, pp. 2001–2003

ERRATUM: MULTITASKING CAPACITY: HARDNESS RESULTS AND IMPROVED CONSTRUCTIONS*

NOGA ALON†, JONATHAN D. COHEN‡, THOMAS L. GRIFFITHS§, PASIN MANURANGSI¶, DANIEL REICHMAN‖, IGOR SHINKAR#, AND TAL WAGNER††

Abstract. We correct an error in the appendix of [N. Alon et al., *SIAM J. Discrete Math.*, 34 (2020), pp. 885–903] and prove that it is NP-hard to approximate the size of a maximum induced matching of a bipartite graph within any constant factor.

Key words. connected matching, hardness of approximation, multitasking

MSC code. 68R05

DOI. 10.1137/23M1603856

1. Result. Recall the following definitions and results from [1].

DEFINITION 2.3. *For a given graph G, a connected matching in G is a matching M such that every two edges of M are connected by an edge of G. Let $\nu_c(G)$ denote the size of the maximum cardinality of a connected matching in G.*

THEOREM 3.1. *Given a bipartite graph G with n vertices on each side, it is NP-hard to approximate $\nu_c(G)$ within a factor of $n^{1-\varepsilon}$ for any $\varepsilon > 0$ under a randomized polynomial time reduction.*

DEFINITION 3.2. *Fix $n \in N$. A bipartite graph $HC_n = (A = \{u_1, \ldots, u_n\}, B = \{v_1, \ldots, v_n\}, E_H)$ is said to be a bipartite half-cover of K_n if (1) for every $\{i, j\} \subseteq [n]$, $(u_i, v_j) \in E_H$ or $(u_j, v_i) \in E_H$, and (2) for every $i \in [n]$, $(u_i, v_i) \notin E_H$.*

CLAIM 3.3. *There is an $O(n)$-time randomized algorithm that on input $n \in N$ outputs a graph HC_n, which is a bipartite half-cover of K_n such that $\nu_c(HC_n) \leq O(\log n)$ with probability $1 - o(1)$.*

As remarked in [1] a deterministic polynomial time construction of such graphs would imply that the hardness result in Theorem 3.1 holds under a deterministic reduction (as opposed to the randomized reduction stated). Here we show how to modify the proof of Theorem 3.1 and get that a weaker hardness result holds under deterministic reduction. In particular, this gives that the problem of computing $\nu_c(G)$

*Received by the editors September 22, 2023; accepted for publication September 29, 2023; published electronically June 26, 2024.
https://doi.org/10.1137/23M1603856
†Department of Mathematics, Princeton University, Princeton, NJ 08544 USA (nogaa@post.tau.ac.il).
‡Princeton Neuroscience Institute, Princeton University, Princeton, NJ 08540 USA (jdc@princeton.edu).
§Departments of Computer Science and Psychology, Princeton University, Princeton, NJ 08544 USA (tomg@princeton.edu).
¶Google Research, Mountain View, CA 94043 USA (pasin@google.com).
‖Department of Computer Science, Worcester Polytechnic Institute, Worcester, MA 01609 USA (daniel.reichman@gmail.com).
#School of Computing Science at Simon Fraser University, Burnaby, BC V5A 1S6, Canada (ishinkar@sfu.ca).
††Amazon, Seattle, WA 98109 USA Work done prior to being affiliated with Amazon (agner@gmail.com).

for a given input bipartite graph G is APX-hard (and hence also NP-hard). This is stated in the following result.

THEOREM 1.1. *Given a bipartite graph G with n vertices on each side, the problem of computing $\nu_c(G)$ is APX-hard.*

2. Proof. We need the fact that it is NP-hard to approximate the clique number of a graph even when this number is linear. This is well known; in particular we state the following result of Hastad [2] (see the proof of Theorem 8.1 there).

THEOREM 2.1 (see [2]). *Given an n-vertex graph G, it is NP-hard to distinguish between the case that the size $w(G)$ of the maximum clique of G is at least $\frac{n}{4}(1-\varepsilon)$ and the case that $w(G)$ is at most $\frac{n}{8}(1+\varepsilon)$.*

The main tool in the proof is the following (weak) derandomized version of Claim 3.3 above.

THEOREM 2.2. *There is a deterministic polynomial time algorithm that on input $n \in N$ (of the form s^{2^k} for some integers s, k) outputs a graph HC_n, which is a bipartite half-cover of K_n such that*

$$\nu_c(HC_n) \leq \frac{n}{e^{\Omega(\sqrt{\log n})}}.$$

The proof of this theorem is based on two lemmas. The first is an efficient deterministic algorithm for constructing an initial relatively small half-cover, and the second is an efficient procedure for squaring a half-cover. The desired graph is obtained from the small initial graph by repeated squaring.

LEMMA 2.3. *There is a deterministic algorithm that on input $n \in N$ outputs, in time polynomial in n, a graph H, which is a bipartite half-cover of K_m for $m = e^{\sqrt{\log n}}$ such that*

$$\nu_c(H) \leq O(\log m) = O(\sqrt{\log n}).$$

Proof (sketch). Apply the method of conditional expectations [3] to the proof of Claim 3.3 given in [1]. The running time is $m^{O(\log m)} = n^{O(1)}$. ∎

LEMMA 2.4. *There is a deterministic polynomial time algorithm which, given as an input a bipartite half-cover F of K_p with $\nu_c(F) \leq \varepsilon p$, outputs a bipartite half-cover F' of K_{p^2} satisfying $\nu_c(F') \leq 4\varepsilon p^2$.*

Proof. Let the vertex classes of F be $A = \{a_1, a_2, \ldots, a_p\}$ and $B = \{b_1, b_2, \ldots, b_p\}$, where for every i $a_i b_i$ is not an edge. We construct the graph F' by blowing up H as follows. Replace each vertex a_i by a set U_i of p vertices and each vertex b_j by a set V_j of p vertices, where all these sets are pairwise disjoint. The vertex classes of F' are $U = \cup_i U_i$ and $V = \cup_j V_j$. For each $1 \leq i \leq p$, the bipartite graph between U_i and V_i is a copy of F. For every $1 \leq i \neq j \leq p$, the bipartite graph between U_i and V_j is complete if $a_i b_j$ is an edge of F; otherwise it is edgeless.

It is easy to see that the constructed graph F' is a bipartite half-cover of K_{p^2}. In order to complete the proof we show that

$$\nu_c(F') \leq 4\varepsilon p^2. \tag{1}$$

Indeed, let M be a maximum connected matching in F'. Construct an auxiliary graph F'' on the classes of vertices A, B by letting $a_i b_j$ be an edge if and only if $i \neq j$ and there is at least one edge of M connecting a vertex of U_i and a vertex of V_j. Any matching in F'' can be partitioned into three disjoint matchings, where none of these three matchings saturates both a_i and b_i.

Note that each of these three matchings in F'' must be a connected matching in F, and hence its size is at most εp. This shows that the size of the maximum matching in F'' is at most $3\varepsilon p$. By König's Theorem this means that F'' has a vertex cover of size at most $3\varepsilon p$, implying that all edges of M that do not connect a vertex of U_i with one of V_i (for some i, with the same index i) are covered by the vertices in at most $3\varepsilon p$ of the blocks U_i, V_j. This gives a total of at most $3\varepsilon p^2$ edges of M. In addition, for each fixed i the connected matching M can contain at most εp edges connecting a vertex in U_i and one in V_i, adding at most εp^2 additional edges and establishing (1). This completes the proof of the lemma. ∎

Starting with the graph $H_1 = H$ in Lemma 2.3 and applying Lemma 2.4 repeatedly k times, where $2^k = \sqrt{\log n}$ we get that there is a deterministic algorithm that on input $n \in N$ (of the form s^{2^k} for some integers s, k) outputs, in time polynomial in n, a graph HC, which is a bipartite half-cover of K_n such that

$$\nu_c(HC) \le 4^k \frac{O(\sqrt{\log n})}{e^{\sqrt{\log n}}} n = \frac{n}{e^{\Omega(\sqrt{\log n})}}.$$

This establishes the statement of Theorem 2.2.

The assertion of Theorem 1.1 follows from Theorems 2.2 and 2.1 by following the argument given in [1] for proving Theorem 3.1, based on Claim 3.4 in [1]. We omit the details.

Acknowledgment. We thank Cyriac Antony and Daniel Paulusma for pointing out a flaw in a previous proof we suggested (Theorem A.1 in [1]) for a weaker version of Theorem 1.1. The result here is stronger as it shows that the problem of computing $\nu_c(G)$ for a given bipartite input graph G is not only NP-hard but is also APX-hard.

REFERENCES

[1] N. Alon, J. D. Cohen, T. L. Griffiths, P. Manurangsi, D. Reichman, I. Shinkar, T. Wagner, and A. Yu, *Multitasking capacity: Hardness results and improved constructions*, SIAM J. Discrete Math., 34 (2020), pp. 885–903, https://doi.org/10.1137/18M1224672.

[2] J. Hastad, *Some optimal inapproximability results*, J. ACM, 48 (2001), pp. 798–859.

[3] N. Alon and J. Spencer, *The Probabilistic Method*, John Wiley & Sons, 2016.