Strong edge-coloring of $(3, \Delta)$-bipartite graphs[☆]

Julien Bensmail[a], Aurélie Lagoutte[a], Petru Valicov[b]

[a] *LIP, UMR 5668 ENS Lyon, CNRS, UCBL, INRIA, Université de Lyon, France*
[b] *Aix-Marseille Université, CNRS, LIF UMR 7279, 13288, Marseille, France*

Abstract

A strong edge-coloring of a graph G is an assignment of colors to edges such that every color class induces a matching. We here focus on bipartite graphs whose one part is of maximum degree at most 3 and the other part is of maximum degree Δ. For every such graph, we prove that a strong 4Δ-edge-coloring can always be obtained. Together with a result of Steger and Yu, this result confirms a conjecture of Faudree, Gyárfás, Schelp and Tuza for this class of graphs. graphs.

Keywords: Strong edge-colouring, bipartite graphs, maximum degree

1. Introduction

One common notion of graph theory is the one of *proper edge-coloring*, which is, given an undirected simple graph $G = (V, E)$, an assignment of colors to the edges such that no two adjacent edges receive the same color. A proper edge-coloring can equivalently be seen as a partition of the edges into *matchings*. One can easily convince himself that these matchings are generally not induced. If we want each matching of the partition to be induced, then in every part all edges must be sufficiently far apart in the graph. In this perspective, Fouquet and Jolivet introduced the following stronger notion [7]: a *strong edge-coloring* of G is a proper edge-coloring such that every two edges joined by another edge are colored differently. Clearly, every matching of a strong edge-coloring is induced. The least number of colors in a strong edge-coloring is referred to as the *strong chromatic index*, denoted $\chi'_s(G)$ for G.

We denote by $\Delta(G)$ (or simply Δ when no ambiguity is possible) the *maximum degree* of G. If S is a subset of vertices of a graph, we refer to $\Delta(S)$ as the maximum degree of the vertices of S. Greedy coloring arguments show that $2\Delta^2 - 2\Delta + 1$ is a naive upper bound on the strong chromatic index of any graph. But so many colors are generally not necessary to obtain a strong edge-coloring. Actually, the tightest upper bound on $\chi'_s(G)$ involving Δ is believed to be the following.

Conjecture 1 (Erdős and Nešetřil [5]). *For every graph G, we have*

$$\chi'_s(G) \leq \begin{cases} \frac{5}{4}\Delta^2 & \text{if } \Delta \text{ is even,} \\ \frac{1}{4}(5\Delta^2 - 2\Delta + 1) & \text{if } \Delta \text{ is odd,} \end{cases}$$

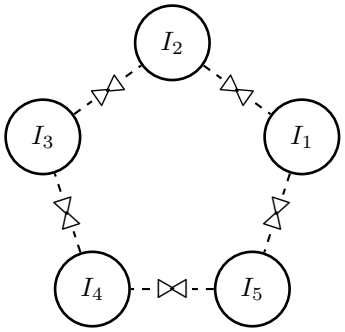
which, if true, would be tight as the graphs described on Figure 1 achieve these bounds.

This conjecture was verified for graphs of maximum degree at most 3 [1, 8], and also considered in other situations [9, 3]. But it remains still widely open in general.

In this paper we focus on strong edge-coloring of *bipartite graphs*, which are graphs whose vertex set admits a bipartition into two independent sets. In this context, Conjecture 1 was strengthened to the following by Faudree, Gyárfás, Schelp and Tuza:

Conjecture 2 (Faudree *et al.* [6]). *For every bipartite graph G, we have $\chi'_s(G) \leq \Delta^2$.*

[☆] This research is partially supported by ANR Grant STINT - ANR-13-BS02-0007.



- Every I_j is an independent set.
- "$I_j \bowtie I_{j'}$" means that I_j is complete to $I_{j'}$.
- If $\Delta = 2k$, then $|I_j| = k$.
- If $\Delta = 2k+1$, then $|I_1| = |I_2| = |I_3| = k$ and $|I_4| = |I_5| = k+1$.

Figure 1: Erdős and Nešetřil's construction.

Brualdi and Quinn Massey introduced a new notion of edge-coloring – the incidence coloring of graphs [2]. They showed a connection of this notion with the one of strong edge-coloring, which made them refine Conjecture 2.

Conjecture 3 (Brualdi and Quinn Massey [2]). *For every bipartite graph G with bipartition A and B, we have $\chi'_s(G) \leq \Delta(A)\Delta(B)$.*

In the spirit of this conjecture, we define a (d_A, d_B)-*bipartite graph* to be a bipartite graph with parts A and B such that $\Delta(A) \leq d_A$ and $\Delta(B) \leq d_B$. Conjectures 2 and 3 are still widely open, the second being proved to hold in two specific non-trivial situations. It is first known to hold whenever G is subcubic bipartite:

Theorem 1 (Steger and Yu [11]). *For every $(3,3)$-bipartite graph G, we have $\chi'_s(G) \leq 9$.*

Later on, Nakprasit solved the case where one part of the bipartition is of small maximum degree, namely at most 2.

Theorem 2 (Nakprasit [10]). *For every $(2,\Delta)$-bipartite graph G, we have $\chi'_s(G) \leq 2\Delta$.*

Theorems 1 and 2 were proved using a similar proof scheme, first used in [11]. Reusing this idea, we prove the following which, together with the aforementioned previous results, settles a special case of Conjecture 2.

Theorem 3. *For every $(3,\Delta)$-bipartite graph G, we have $\chi'_s(G) \leq 4\Delta$.*

2. Proof of Theorem 3

Let G be a $(3,\Delta)$-bipartite graph with bipartition A and B such that $\Delta(A) \leq 3$. We set $n_B = |V(B)|$. It is sufficient to prove the result for the case where all vertices of A are of degree exactly 3, so let us make this assumption.

We describe G by a (non-unique) $(n_B \times \Delta)$-matrix constructed in the following way:

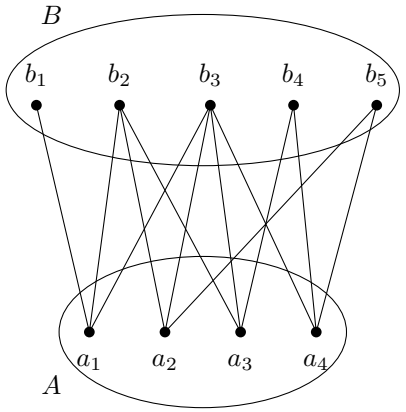
- the rows are indexed by the vertices of B and the columns are indexed by $1, 2, \ldots, \Delta$;
- every row with index $b \in B$ contains exactly once every edge incident to b (some cells will be empty if b is of degree strictly less than Δ).

We give an example of a bipartite graph and two such associated matrices in Figure 2. Note that the order of the edges (and the empty cells, if any) in any row of a matrix can be arbitrary, and we will explain later how to take advantage of it. Assuming an edge e of G lies in cell (i,j) of a matrix, we often refer to the index j as the "column of e" (with respect to this matrix).

Every matrix describing G yields a classification of the vertices of A into three types:

Type 1: vertices whose all incident edges are in the same column,

Type 2: vertices whose only two incident edges are in the same column,



$$\begin{array}{c} \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} b_1a_1 & \llcorner\lrcorner & \llcorner\lrcorner & \llcorner\lrcorner \\ b_2a_1 & b_2a_2 & \llcorner\lrcorner & b_2a_3 \\ b_3a_1 & b_3a_2 & b_3a_3 & b_3a_4 \\ \llcorner\lrcorner & b_4a_4 & \llcorner\lrcorner & b_4a_3 \\ b_5a_2 & \llcorner\lrcorner & b_5a_4 & \llcorner\lrcorner \end{pmatrix} \end{array}$$

$$\begin{array}{c} \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{pmatrix} b_1a_1 & \llcorner\lrcorner & \llcorner\lrcorner & \llcorner\lrcorner \\ b_2a_1 & b_2a_2 & \llcorner\lrcorner & b_2a_3 \\ b_3a_1 & b_3a_2 & b_3a_4 & b_3a_3 \\ \llcorner\lrcorner & b_4a_4 & \llcorner\lrcorner & b_4a_3 \\ \llcorner\lrcorner & b_5a_2 & b_5a_4 & \llcorner\lrcorner \end{pmatrix} \end{array}$$

Figure 2: Example of $(3, \Delta = 4)$-bipartite graph and two associated matrices. A "$\llcorner\lrcorner$" indicates that the content of the cell is empty.

Type 3: vertices whose all incident edges are in different columns.

Since every Type 1 vertex v has all of its three incident edges of the same column, say i, calling i the "column of v" directly makes sense. When considering a Type 2 vertex, we say that its two incident edges located in the same column are *paired*. Its third incident edge is called *lonely*.

Since the order of the edges and the empty cells in a given row is arbitrary, different matrices can describe G. However, some of them will be better for us, so let us define an order on the matrices and, from now on, consider a *maximum matrix* M_G of G. The order is defined as the lexicographical order on (T_1, T_2), where T_i $(i = 1, 2)$ is the number of Type i vertices. As an illustration of this order, note that, with the first (top) matrix of Figure 2, only a_1 is Type 1, the vertices a_2 and a_3 are Type 2, while only a_4 is Type 3. But this matrix is not maximum as the second (bottom) matrix of Figure 2 describes the same graph but yields three Type 1 vertices (a_1, a_2 and a_3), one Type 2 vertex (a_4), and no Type 3 vertex. This second matrix is actually maximum.

Now we give some observations on M_G which will be useful for the coloring process. Most of these observations are straightforward and can be proved by just showing that if some particular situation occurs, then we can perform switches (*i.e.* exchange two edges in a same row) in M_G to get a matrix contradicting the maximality of M_G. We provide the proof of Observation 3 as an illustration of this statement.

Observation 1. *For every $i \in \{1, \ldots, \Delta\}$, every vertex of B has at most one incident edge in column i.*

Let e and e' be two edges of G. We say that e is *visible* from e' (or e' *sees* e) if $\{e, e'\}$ is not an induced matching, or, rephrased differently, if neither e and e' are adjacent nor share a common adjacent edge. So equivalently a strong edge-coloring is an assignment of colors such that every two edges which are mutually visible are assigned different colors.

Observation 2. *If two vertices a_0 and a_1 of A have no common neighbor in B, then every edge incident to a_0 sees no edge incident to a_1.*

Observation 3. *Let a_0 and a_1 be two Type 3 vertices with incident edges a_0b_0, a_0b_1, a_0b_2 and a_1b_3, a_1b_4, a_1b_2, respectively. Note that b_2 is a common neighbor of a_0 and a_1. Let i, j, k (respectively i', j', k') be the columns of a_0b_0, a_0b_1, a_0b_2 (respectively a_1b_3, a_1b_4, a_1b_2). Then $k \notin \{i', j', k'\}$ and $k' \notin \{i, j, k\}$.*

Proof
Assume by contradiction that one of the situations described in the statement occurs, *e.g.* that

$$
\begin{array}{c} \\ b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \end{array}
\begin{array}{c}
\begin{array}{ccccc} i & j & k=i' & j' & k' \end{array} \\
\left(\begin{array}{ccccc}
b_0a_0 & - & - & - & - \\
- & b_1a_0 & - & - & - \\
- & - & b_2a_0 & - & b_2a_1 \\
- & - & b_3a_1 & - & - \\
- & - & - & b_4a_1 & -
\end{array}\right) \\
M_G
\end{array}
\qquad
\begin{array}{c} \\ b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \end{array}
\begin{array}{c}
\begin{array}{ccccc} i & j & k=i' & j' & k' \end{array} \\
\left(\begin{array}{ccccc}
b_0a_0 & - & - & - & - \\
- & b_1a_0 & - & - & - \\
- & - & \mathbf{b_2a_1} & - & \mathbf{b_2a_0} \\
- & - & b_3a_1 & - & - \\
- & - & - & b_4a_1 & -
\end{array}\right) \\
M'_G
\end{array}
$$

Figure 3: An illustration of the proof of Observation 3. The first (left) matrix M_G cannot be maximum, since one can switch two edges on a same row to get the second (right) matrix M'_G, which has more Type 2 vertices and the same number of Type 1 vertices. A "$-$" indicates that the content of the cell can be arbitrary (either filled or empty).

without loss of generality we have $k = i'$. Then M_G looks like the first (left) matrix depicted in Figure 3. But then, by switching b_2a_0 and b_2a_1 in the row indexed by b_2, we get the second (right) matrix M'_G depicted in Figure 3 which yields the same number of Type 1 vertices, but one extra Type 2 vertex a_1. Therefore, M_G is not maximum – a contradiction. □

Observation 4. *Let a_0 be a Type 3 vertex with incident edges e_1, e_2, e_3 in columns i, j, k, respectively. Let a_1 be a Type 2 vertex with incident edges e_4, e_5, e_6 in columns i', i', j', respectively. If e_6 is adjacent to e_3, then $j' \notin \{i, j, k\}$ and $i' \neq k$.*

Observation 5. *Let a_0 and a_1 be two Type 2 vertices. Let e_1, e_2, e_3 (respectively e_4, e_5, e_6) be their incident edges in columns i, i, j (respectively i', i', j'). If e_3 is adjacent to e_4 or e_5, then $i \neq i'$.*

Observation 6. *Let a_0 and a_1 be two Type 2 vertices. Let e_1, e_2, e_3 (respectively e_4, e_5, e_6) be their incident edges in columns i, i, j (respectively i', i', j'). If e_3 is adjacent to e_6, then $j \neq i'$.*

Observation 7. *Let a_0 and a_1 be two Type 1 vertices of columns i and j, respectively. If a_0 and a_1 have a common neighbor, then $i \neq j$.*

Observation 8. *Let a be a Type 2 vertex with incident edges e_1, e_2, e_3, where e_1 is the lonely edge in column j. Then at least one of e_2 or e_3 is not adjacent to a lonely edge of column j different from e_1.*

We now describe the coloring process which will yield a strong 4Δ-edge-coloring c of G. Each edge e will be given a color $c(e) = (i, j)$, where $j \in \{1, \ldots, \Delta\}$ is fixed as the column of e in M_G and $i \in \{1, 2, 3, *\}$ is to be set in the coloring process. So, in what follows, by "coloring an edge" we mean assigning a value to i.

The coloring process mainly consists in coloring the edges of G successively without creating any conflict, *i.e.* in such a way that every resulting partial edge-coloring remains strong. Its successive steps are the following:

Coloring Procedure:

Step 1: color the edges incident to Type 1 vertices.
Step 2: color the paired edges incident to Type 2 vertices.
Step 3: color the edges incident to Type 3 vertices.
Step 4: color the lonely edges incident to Type 2 vertices.

In order to show that this coloring procedure is almost optimal somehow, we will impose ourselves the constraint that the "special" color $*$ is used during Step 4 only. This will show that 3Δ colors are sufficient to color all edges considered during Steps 1 to 3.

The first three steps will be performed greedily, while the last one requires a careful analysis of the structure of the remaining non-colored edges. The rest of this section is dedicated to explanations on why this procedure can be achieved correctly, *i.e.* why there is always an available color for an edge considered at any of the four steps.

Step 1: *color the edges incident to Type 1 vertices.*

For each Type 1 vertex with incident edges e_1, e_2 and e_3, just color e_1, e_2 and e_3 greedily (*i.e.* properly) with $\{1, 2, 3\}$. The obtained edge-coloring is also strong, which follows directly from Observations 2 and 7.

Step 2: *color the paired edges incident to Type 2 vertices.*

Once again, for each Type 2 vertex with incident paired edges e_1 and e_2, we just color e_1 and e_2 greedily, in such a way that no conflict arises with the already colored edges. The following lemma shows that this is always possible, *i.e.* that, after Step 1 and at any moment of Step 2, there is always (at least) one color available for any considered paired edge.

Lemma 1. *After performing Step 1 and any number of iterations of Step 2, for each Type 2 vertex there are always at least two colors available among* $\{1, 2, 3\}$ *for each of its paired edges.*

Proof
Let a be a Type 2 vertex, ab_1 and ab_2 be its paired edges situated in column, say, j of M_G, and ab_0 be the lonely edge. Let us count the number of already colored edges in column j visible from ab_1 or ab_2. We prove that there is at most one such edge, which moreover is incident to b_0. First recall that, according to Observation 1, none of the edges incident to b_1 or b_2, except ab_1 and ab_2, are in column j. Consider the neighbors of b_1 and b_2 distinct from a. Without loss of generality we consider one of them, say a_1 - neighbor of b_1, and assume for contradiction that a_1 has at least one already colored incident edge e in column j. As mentioned previously, according to Observation 1, e cannot be a_1b_1. Thus a_1 cannot be a Type 1 vertex. Moreover, if a_1 is a Type 3 vertex, then e has not been colored yet. The same happens if a_1 is a Type 2 vertex and e is a lonely edge. The last case occurs when a_1 is a Type 2 vertex and e is a paired edge: by Observation 1, edge a_1b_1 has to be lonely, and then Observation 5 yields a contradiction. Now observe that b_0 has at most one incident edge in column j by Observation 1. Consequently, at any moment while performing Step 2 of the procedure, two colors among $\{1, 2, 3\}$ are available for ab_1 and ab_2. □

Step 3: *color the edges incident to Type 3 vertices.*

Once again, a correct extension of the partial strong edge-coloring to the edges incident to the Type 3 vertices can be obtained greedily. The following lemma shows that available colors exist for any edge considered during the procedure.

Lemma 2. *After performing Step 2 and any given number of iterations of Step 3, for each edge incident to any given Type 3 vertex there is at least one available color among* $\{1, 2, 3\}$.

Proof
Let a be a Type 3 vertex with neighbors b_0, b_1 and b_2, and let j be the column of ab_0. Let us count the number of edges visible from ab_0, which are already colored and in column j. We prove that there can be at most two of them. Due to Observation 1, vertices b_1 and b_2 can each have at most one incident edge in column j. Let a_0 be a neighbor of b_0 and suppose for contradiction that a_0 has an incident edge e in column j. By Observation 1, edge a_0b_0 cannot be in column j, and thus a_0 is not a Type 1 vertex. For the same reason, if a_0 is of Type 2, edge a_0b_0 cannot be paired with e; moreover, by Observation 4, edge a_0b_0 cannot be a lonely edge, so e is the lonely edge of a_0, and thus is not colored yet. Finally a_0 cannot be a Type 3 vertex according to Observation 3. Thus at least one color among $\{1, 2, 3\}$ is available for a_0b_0. □

Step 4: *color the lonely edges incident to Type 2 vertices.*

Before explaining how to color the lonely edges explicitly, we first introduce some notions and raise some observations about how these edges appear in G.

Let F be a subset of edges of G. The *subgraph induced by* F is the subgraph induced by the endpoints of the edges of F. For each column j of M_G, we define the *component of* j, denoted $\mathcal{C}_j$, as the subgraph of G induced by the set of lonely edges of column j. Since G is bipartite, observe that every cycle of $\mathcal{C}_j$ have even length. We call a cycle $v_0v_1v_2 \dots v_{k-1}v_0$ of $\mathcal{C}_j$ *alternate* if exactly half of its edges are lonely in column j and, for every pair of consecutive edges v_iv_{i+1} and $v_{i+1}v_{i+2}$

(where i is taken modulo k), one is lonely in column j and the other is not (*i.e.* the lonely edges of column j on the cycle are non-adjacent). Similarly, we say that a path of $\mathcal{C}_j$ is *alternate* if for every pair of adjacent edges of the path, one of them is lonely in column j and the other is not. We prove below that each $\mathcal{C}_j$ has a very specific structure. We first start with a direct consequence of Observation 8.

Observation 9. *Let j be a column of M_G. Every Type 2 vertex $a \in A$ appearing in $\mathcal{C}_j$ cannot have both its paired edges in $\mathcal{C}_j$.*

Proof
Since $a \in \mathcal{C}_j$, its lonely edge is in column j. Moreover, if both its paired edges ab_1 and ab_2 belong to $\mathcal{C}_j$, then both b_1 and b_2 are incident to a distinct lonely edge in column j. But this contradicts the maximality of M_G according to Observation 8. □

Lemma 3. *Let j be a column of M_G. Every connected component of $\mathcal{C}_j$ has at most one cycle. Moreover, if this cycle exists, then it must be alternate.*

Proof
Observe that $\mathcal{C}_j$ has no lonely edge e which is not in column j. Otherwise, since we are considering the component $\mathcal{C}_j$, the endpoint of e in part A would be incident to a lonely edge of column j contradicting the definition of a Type 2 vertex. Therefore, from now on in this proof, when speaking about lonely edges of $\mathcal{C}_j$ we will refer to lonely edges of column j.

First we show that all cycles in $\mathcal{C}_j$ are alternate. Suppose by contradiction that there is a cycle in $\mathcal{C}_j$ which is non-alternate. Observe first that there cannot be two adjacent lonely edges in $\mathcal{C}_j$ (otherwise there would be two lonely edges incident to a same vertex in A or B, which is impossible by the definition of a Type 2 vertex and Observation 1). Thus by hypothesis the non-alternate cycle must have two non-lonely adjacent edges e and e' sharing a same vertex v.

Observe first that v cannot be in part A: otherwise, these two non-lonely edges e and e' would be the paired edges of v, a contradiction with Observation 9. Now suppose that v is in part B. We denote the non-alternate cycle by $C = b_0a_0b_1a_1...b_ka_kb_0$, where each vertex a_i (resp. b_i) belongs to A (resp. B). Assume $v = b_0$, as well as $e = b_0a_0$ and $e' = b_0a_k$. Then, since no vertex a_i has its two paired edges along C (according to Observation 9), we get that a_0b_1 is lonely. Now, since two lonely edges cannot be adjacent, b_1a_1 is not lonely. Repeating these arguments along the edges of C, we get that every edge b_ia_i with $0 \leq i \leq k$ is non-lonely, while every a_ib_{i+1} for $0 \leq i \leq k-1$ is lonely. Then we get that the two edges incident to a_k along C are not lonely, which contradicts Observation 9.

Therefore, all the cycles of the component are alternate.

Now we prove that there can be only one alternate cycle (if any) in every connected component of $\mathcal{C}_j$. Suppose by contradiction that there are two alternate cycles in a connected component of $\mathcal{C}_j$. We show the following properties about these two cycles to end up with a contradiction:

1. the two cycles cannot share a vertex without sharing an edge,
2. the two cycles cannot share an edge,
3. the two cycles cannot be joined by a path in the component.

The first property follows from the fact that the two cycles are alternate and there cannot be two adjacent lonely edges in a same component. Suppose by contradiction that the second property is false, *i.e.* that two cycles share an edge. Let $P = v_1 \cdots v_k$ be one longest alternate path shared by theses cycles. Observe that v_1v_2 must be lonely (since otherwise there would be two adjacent lonely edges) and v_1 must have two other incident non-lonely edges - one in each of the two cycles. We call these edges e and e' respectively, and observe then that $v_1 \notin A$ thanks to Observation 9. However, by the same arguments, v_k must be in B as well, and $v_{k-1}v_k$ must be a lonely edge. Then P is an alternate path of odd length between v_1 and v_k which are both in part B, a contradiction since $\mathcal{C}_j$ is bipartite.

Finally, in order to show the third property, suppose by contradiction that there is a path connecting the two cycles in the component. Consider in particular the shortest path P with

extremities u and v, where u lies on the first cycle while v lies on the second one. Recall that the cycles are alternate, and thus one edge of the first cycle incident to u is lonely, and similarly for v with respect to the second cycle. Recall also that P is a subgraph of $\mathcal{C}_j$. Therefore, by Observation 9, none of u and v can be in part A: otherwise, they would have two paired edges in $\mathcal{C}_j$ - one on the cycle, and one on P.

Let us hence denote $P = b_0a_0b_1a_1...b_k$, where $u = b_0$ and $v = b_k$, and $k \geq 2$ is even. Now consider the successive vertices of P, *i.e.* from b_0 to b_k. By Observation 1, the edge b_0a_0 cannot be lonely since b_0 already has an incident lonely egde on the first cycle. Now, a_0b_1 has to be lonely, since otherwise a_0 would have both its paired edges in $\mathcal{C}_j$. Repeating the same arguments until we reach b_k, we get that every edge b_ia_i is not lonely, while every edge a_ib_{i+1} is lonely, for $0 \leq i \leq k-1$. Then b_k is incident to two lonely edges (one is $a_{k-1}b_k$ and the other one is on the second cycle), a contradiction with Observation 1. □

We now explain how to color the lonely edges in order to finish the coloring of G. Recall that during this step, we allow the use of the special color $*$. Consider every successive value of $j \in \{1, ..., \Delta\}$. We may assume that $\mathcal{C}_j$ is connected (if not, apply the procedure below component-wisely). The lonely edges of $\mathcal{C}_j$ are colored in up to two phases as follows:

Phase 1: In case $\mathcal{C}_j$ has an induced cycle C, it is unique and alternate according to Lemma 3. Let $C = a_1b_1a_2b_2...a_kb_ka_1$ be this cycle, where $a_1b_1, a_2b_2, \ldots, a_kb_k$ are its lonely edges and $a_i \in A$ (resp. $b_i \in B$) for $1 \leq i \leq k$. Then greedily color the edges $a_1b_1, \ldots, a_{k-1}b_{k-1}$, in this order, with colors among $\{1, 2, 3\}$ in order to obtain a partial strong edge-coloring. Color the remaining lonely edge a_kb_k with color $*$.

Phase 2: If C exists, then first remove its edges to get a (possibly empty) forest. Each tree T of the forest will have as a root a vertex r of C. If the component had no cycle C, we designate an arbitrary node of T to be the root r. Then greedily color with colors among $\{1, 2, 3, *\}$ the remaining uncolored lonely edges of T as they are encountered during a Breadth-First Search (BFS) algorithm performed from r.

The following two results show that Phases 1 and 2 can always be performed correctly.

Lemma 4. *During Phase 1, for every lonely edge of C there is at least one available color among $\{1, 2, 3, *\}$.*

Proof
Assume that the edges $a_1b_1, \ldots, a_{i-1}b_{i-1}$ have already been colored and let $a_ib_i \neq a_kb_k$ be the considered lonely edge. Recall that, due to our ordering, the edge $a_{i+1}b_{i+1}$ is uncolored. Recall also that no other edge adjacent to a_ib_i in G is in column j of M_G (according to Observation 1 and the definition of a Type 2 vertex). Let us count the number of edges visible from a_ib_i which are already colored and in column j. Let us prove that there can be at most two of them. One of them if $a_{i-1}b_{i-1}$. Let b be the third neighbor of a_i. By Observation 1, at most one of the edges incident to b can be in column j. Now consider a neighbor a of b_i (different from a_i) and assume it has an incident edge e in column j. By Observation 1, a cannot be a Type 1 vertex, nor a Type 2 vertex where e would be paired with ab_i. By Observation 6, if a is a Type 2 vertex, then e is lonely and thus not yet colored: indeed, there exists at most one cycle per component (according to Lemma 3), and, for now, we have colored only lonely edges involved in a cycle. Finally a cannot be a Type 3 vertex according to Observation 4. Therefore, one color among $\{1, 2, 3\}$ is available to color a_ia_{i+1}. As for a_kb_k, no other edge of the same connected component of $\mathcal{C}_j$ is colored with $*$, so coloring this edge cannot create any conflict (note that a_kb_k can have three visible edges in column j and thus none of $\{1, 2, 3\}$ may be available). This completes the proof. □

Lemma 5. *During the BFS algorithm in Phase 2, for every lonely edge of T there is at least one available color among $\{1, 2, 3, *\}$.*

Proof
Consider a Type 2 vertex $a \in A$ with lonely edge $ab_0 \in T$, where $a \in A$ and $b_0 \in B$. So a is Type 2 with paired edges a_0b_1 and a_0b_2. Then b_1 and b_2 can be each incident to at most one edge

in column j, and each of these two edges may be colored already (for example, if both b_1 and b_2 are adjacent to a Type 1 vertex in column j).

We now prove that the other edges visible from ab_0 and in column j have to be lonely, and that at most one of them is already colored. Consider any edge b_0a_0 different from ab_0 and adjacent to an edge e in column j. By Observation 1, b_0a_0 cannot be in column j. Then a_0 is either of Type 2 or Type 3. Actually, a_0 cannot be of Type 3 according to Observation 4. Also, according to Observation 6, a_0 cannot be of Type 2 with b_0a_0 being lonely and its paired edges being of column j. So necessarily a_0 is of Type 2 with lonely edge e in column j. Observe that b_0 may have several neighbors playing the same role as a_0, *i.e.* incident to an edge e' in column j, but then the same argument applies and e' is lonely.

The important remark to raise is that the BFS algorithm performed on T from r ensures that, whenever a lonely edge e is treated, at most one lonely edge of T (and C, if it exists) visible from e has already been colored: indeed, assume first that e is not adjacent to the root and call $v_\uparrow$ (resp. $v_\downarrow$) the endpoint of e which is closer (resp. further) to the root r. Then e is the only lonely edge adjacent to $v_\uparrow$; call $v_{\uparrow\uparrow}$ the father of $v_\uparrow$ in the tree: $v_{\uparrow\uparrow}$ has only one incident lonely edge in column j (which happens to be colored before e); finally, any subtree rooted at a son of $v_\uparrow$, or rooted at $v_\downarrow$ is not colored yet. So e sees at most one already colored lonely edge. Let us now deal with a lonely edge e incident to the root r: if $\mathcal{C}_j$ had a cycle, then we chose the root r to be on C; consequently r already has an incident lonely edge on the cycle, a contradiction with Observation 1 and the definition of a Type 2 vertex. Otherwise, the component had no cycle, and thus no lonely edge visible from e has already been colored.

□

3. Conclusion and possible improvements

In this paper, we have proved that, for every $(3, \Delta)$-bipartite graph G, we have $\chi'_s(G) \leq 4\Delta$. This result, together with Theorem 1, confirms Conjecture 2 for this specific family of bipartite graphs. We however believe that our upper bound should not be tight, as stated in Conjecture 3 where 3Δ is conjectured to be the right bound.

Avoiding using $*$

Maybe the upper bound we have obtained, could be improved by refining the coloring procedure introduced in Section 2. To do so, one would have to find a way to do without color $*$, *i.e.* color every lonely edge of $\mathcal{C}_j$ with "regular" color 1, 2 or 3. One optimistic reason why this should be possible is that each such color is only used when 1, 2 and 3 are all forbidden, that is when coloring the connected components of the $\mathcal{C}_j$'s during Step 4.

On the one hand, color $(*, j)$ is always used, in Phase 1 of Step 4, once for each alternate cycle of $\mathcal{C}_j$. But this use of $(*, j)$ is sometimes not necessary. The main purpose for us to systematically use it is to facilitate and lighten the proof of Theorem 3 by avoiding a tedious case analysis. But one may note that the only bad situation, that is when the use of color $(*, j)$ might be necessary to color the lonely edges of C, is when the following three conditions are satisfied

- C is of length $2k$ with $k \geq 3$ odd;
- every vertex $a_i \in A$ of C is at distance 2 from a Type 1 vertex a'_i of column j – call e'_i the edge which is not in C and which joins a'_i and the common neighbor of a_i and a'_i;
- and every edge e'_i has been assigned exactly the same color (i, j) in Step 1 with $i \in \{1, 2, 3\}$.

On the other hand, color $(*, j)$ may also be used during Phase 2 of Step 4 to color a lonely edge of a tree T of $\mathcal{C}_j - C$. A careful analysis shows that actually color $(*, j)$ may only be needed for lonely edges incident to a leaf of T, and if the around vertices are colored in an unfavourable way (typically when several Type 1 vertices surround the leaf).

We believe that if it would be possible to decrease the number of colors used in our procedure, these two bad cases above should be the ones to tackle. To this regard, choosing M_G among all maximal matrices so that it meets additional convenient properties such as minimizing then number of alternate cycles would be interesting to investigate. Also, it is worth pointing out that

many tasks of the coloring process are performed arbitrarily (*e.g.* coloring the edges during Steps 1 to 3, the choice of r during Phase 2 of Step 4, etc.). Searching for better choices would be another promising perspective.

From 3 to higher values of $\Delta(A)$

An interesting perspective of research is to investigate whether the coloring scheme we have used herein may be generalized to larger values of $\Delta(A)$. One could indeed, based on some maximum matrix M_G describing G, organize the incident edges of every vertex in $A(G)$ into *maximal groups of paired edges*, *i.e.* edges in a same column of M_G, and generalize the coloring scheme described in Section 2. Namely, one could first color the maximal groups of $\Delta(A)$ paired edges (which correspond to the notion of Type 1 vertex herein), then color the maximal groups of $\Delta(A)-1$ paired edges, and so on, and show that such strong extensions exist according to generalized versions of Observations 1 to 9. Following the same idea as in our proof of Theorem 3, the algorithm to color the graph would require $\Delta(A)+1$ steps. But the success of this task does not seem immediate to us. In particular, the last step of the new procedure seems hard to define, a simple adaptation of Step 4 from the proof of Theorem 3 being not clear. This is due to the fact that expressing accurately how the maximal groups of paired edges are organized in G in general, is not easy.

Complexity matters

For computational complexity, one could wonder whether our proof yields a polynomial-time algorithm to deduce a strong 4Δ-edge-coloring of a given $(3,\Delta)$-bipartite graph G. The important thing to see is that the coloring can easily be obtained once M_G is known since assigning a color to an edge then just requires to check what are its neighboring colors. However, it turns out that obtaining M_G is an NP-hard problem in general. So that we prove this statement, let us introduce the following problem.

Maximum Number of Type 1 Vertices
Instance: a bipartite graph G and an integer $\ell \geq 1$.
Question: does there exist a matrix describing G yielding at least ℓ Type 1 vertices?

Our statement above follows from a polynomial-time reduction from the following problem, where a *properly k-vertex-colorable graph* is a graph admitting a *proper k-vertex-coloring*, that is a partition of its vertices into k independent sets (*i.e.* with no adjacent vertices).

Maximum Properly k-Vertex-Colorable Subgraph
Instance: a graph G and an integer $\ell \geq 1$.
Question: does there exist a properly k-vertex-colorable subgraph of G with at least ℓ vertices?

Maximum Properly 2-Vertex-Colorable Subgraph is known to remain NP-complete when its input graph is of maximum degree 3 (see [4]). Using this fact, we prove the following result establishing the hardness of Maximum Number of Type 1 Vertices.

Theorem 4. Maximum Number of Type 1 Vertices *is NP-complete, even when restricted to $(3,2)$-bipartite graphs.*

Proof
Given a matrix M_G describing a graph G (which, obviously, has size polynomial in the number of vertices of G), one can compute in polynomial time the number of Type 1 vertices yielded by M_G. So Maximum Number of Type 1 Vertices is an NP problem.

We now prove the NP-hardness of Maximum Number of Type 1 Vertices. Consider an instance of Maximum Properly 2-Vertex-Colorable Subgraph, *i.e.* a graph G of maximum degree 3 together with an integer $\ell \geq 1$. From G, we construct a $(3,2)$-bipartite graph H such that the number of vertices in a maximum properly k-vertex-colorable subgraph of G is exactly equal to the number of Type 1 vertices yielded by a maximum matrix M_H describing H. Hence, (H,ℓ) will be a positive instance of Maximum Number of Type 1 Vertices if and only if (G,ℓ) is a positive instance of Maximum Properly 2-Vertex-Colorable Subgraph. We construct H as the 1-subdivision of G, namely $H = A \cup B$, as follows:

- for every vertex u of G, add a vertex a_u to H,

- for every edge uv of G, add one vertex $b_{u,v}$ to H,
- $A = \{a_u : u \in V(G)\}$ and $B = \{b_{u,v} : uv \in E(G)\}$,
- for every edge uv of G, add the edges $a_u b_{u,v}$ and $a_v b_{u,v}$ to H.

Clearly $\Delta(A) \leq 3$ and $\Delta(B) = 2$, so H is a $(3, 2)$-bipartite graph. Besides, the reduction is achieved in polynomial time since the number of vertices of H is $|V(G)| + |E(G)|$. Note that every two adjacent vertices u and v of G are directly depicted in H by the two vertices a_u and a_v which are at distance exactly 2 (because of $b_{u,v}$). So u and v cannot be assigned the same color by a partial proper 2-vertex-coloring of G while a_u and a_v cannot be Type 1 vertices of a same column of M_H, and vice-versa. From this fact, assuming color, say, 1 is liken to column 1 of M_H, coloring 1 a vertex u of G is equivalent to having a_u being a Type 1 vertex of column 1 of M_H. Because $\Delta(B) = 2$, note that M_H has exactly two columns by definition, and so we can define a straight equivalence between the two colors used to color G and the two columns of M_H. The equivalence between the two instances then follows. □

References

[1] L.D. Andersen. The strong chromatic index of a cubic graph is at most 10. *Discrete Mathematics*, 108:231–252, 1992.

[2] A.T. Brualdi and J.J. Quinn Massey. Incidence and strong edge colorings of graphs. *Discrete Mathematics*, 122:51–58, 1993.

[3] D.W. Cranston. Strong edge-coloring of graphs with maximum degree 4 using 22 colors. *Discrete Mathematics*, 306(21):2772–2778, 2006.

[4] H.-A. Choi, K. Nakajima and C.S. Rim. Graph bipartization and via minimization. *SIAM Journal of Discrete Mathematics*, 2:38–47, 1989.

[5] P. Erdős and J. Nešetřil. Irregularities of partitions (G. Halász, V.T. Sós, Eds.), [Problem], 162–163, 1989.

[6] R.J. Faudree, A. Gyárfás, R.H. Schelp and Zs. Tuza. The strong chromatic index of graphs. *Ars Combinatoria*, 29B:205–211, 1990.

[7] J.L. Fouquet and J.L. Jolivet. Strong edge-colorings of graphs and applications to multi-k-gons. *Ars Combinatoria*, 16A:141–150, 1983.

[8] P. Horák, H. Qing and W.T. Trotter. Induced matchings in cubic graphs. *Journal of Graph Theory*, 17:151–160, 1993.

[9] M. Molloy and B. Reed. A bound on the strong chromatic index of a graph. *Journal of Combinatorial Theory, Series B*, 69(2):103–109, 1997.

[10] K. Nakprasit. A note on the strong chromatic index of bipartite graphs. *Discrete Mathematics*, 308:3726–3728, 2008.

[11] A. Steger and M.-L. Yu. On induced matchings. *Discrete Mathematics*, 120:291–295, 1993.