INSTITUTO DE COMPUTAÇÃO
UNIVERSIDADE ESTADUAL DE CAMPINAS

Using Latin Squares to Color Split Graphs

S.M. Almeida *C.P. de Mello* *A. Morgana*

Technical Report - IC-09-01 - Relatório Técnico

January - 2009 - Janeiro

Using Latin Squares to Color Split Graphs

Sheila Morais de Almeida* Célia Picinin de Mello* Aurora Morgana †

Abstract

An *edge-coloring* of a graph is an assignment of colors to its edges such that no adjacent edges have the same color. A *split graph* is a graph whose vertex set admits a partition into a stable set and a clique. Split graphs have been introduced by Földes and Hammer [4] and it is a well-studied class of graphs. However, the problem of deciding the chromatic index of any split graph remains unsolved. Chen, Fu, and Ko [1] use a latin square to color any split graph with odd maximum degree. In this work, we also use a latin square to color some split graphs with even maximum degree and we show that these graphs are Class 1.

1 Introduction

An *edge-coloring* of G is an assignment of one color to each edge of G such that no adjacent edges have the same color. The *chromatic index*, $\chi'(G)$, is the minimum number of colors for which G has an edge-coloring.

An easy lower bound for the chromatic index is the maximum vertex degree Δ. A celebrated theorem by Vizing [12] states that, for a simple graph, the chromatic index is at most $\Delta+1$. Graphs whose chromatic index equals the maximum degree are said to be *Class* 1; graphs whose chromatic index exceeds the maximum degree by one are said to be *Class* 2. Despite the restriction imposed by Vizing, it is NP-complete to determine, in general, if a graph is Class 1 [7]. There are not many graph classes for which the problem is known to be polynomial; see [6, 8, 10] for examples. The complexity of the problem is open for very structured classes of graphs such as cographs, proper interval graphs and split graphs.

A graph G satisfying the inequality $|E(G)| > \Delta(G)\left\lfloor\frac{|V(G)|}{2}\right\rfloor$, is said to be an *overfull* graph. A graph G is *subgraph-overfull* when it has an overfull subgraph H with $\Delta(H) = \Delta(G)$ [5]. When the overfull subgraph H can be chosen to be a neighborhood of a vertex of degree $\Delta(G)$, we say that G is *neighborhood-overfull* [3]. Overfull, subgraph-overfull, and neighborhood-overfull graphs are in Class 2.

A *split graph* is a graph whose vertex set admits a partition into a stable set and a clique. Split graphs is a well-studied class of graphs for which most combinatorial problems are solved [2, 9, 11]. It has been shown that every odd maximum degree split graph is

*Instituto de Computação, Universidade Estadual de Campinas, 13081-970 Campinas, SP. This research was partially supported by CAPES and CNPq (482521/2007-4 and 300934/2006-8).

†Department of Mathematics, University of Rome "La Sapienza", Italy

Class 1 [1] and that every subgraph-overfull split graph is in fact neighborhood-overfull [3]. It has been conjectured that every Class 2 split graph is neighborhood-overfull [3]. The validity of this conjecture implies that the edge-coloring problem for split graphs is in P. The goal of this paper is to investigate this conjecture by giving another positive evidence for its validity. We describe a new subset of split graphs with even maximum degree that is Class 1. Using latin squares, we construct a polynomial edge-coloring for these graphs.

2 Theoretical framework

In this paper, G denotes a simple, finite, undirected and connected graph with vertex set $V(G)$ and edge set $E(G)$. Write $n = |V(G)|$ and $m = |E(G)|$. For any v in $V(G)$, the set of vertices adjacent to v is denoted by $N(v)$ and $N[v] = \{v\} \cup N(v)$. The *degree* of a vertex v is $d_G(v) = |N(v)|$. The *maximum degree* of G is, then, $\Delta(G) = \max\limits_{v \in V(G)} \{d_G(v)\}$. When there is no ambiguity, we remove the symbol G from the notations. A *clique* is a set of pairwise adjacent vertices of a graph. A *maximal clique* is a clique that is not properly contained in any other clique. A *stable set* is a set of pairwise non adjacent vertices. A subgraph of G is a graph H with $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. For $X \subseteq V(G)$, denote by $G[X]$ the subgraph induced by X, that is, $V(G[X]) = X$ and $E(G[X])$ consists of those edges of $E(G)$ having both ends in X. Let $D \subseteq E(G)$. The subgraph induced by D is the subgraph H with $E(H) = D$ and $V(H)$ is the set of vertices such that each one of them has at least one edge of D incident to it.

In order to study edge-coloring of split graphs, the following results are useful tools. We recall them from [1] for the reader's convenience.

A *color diagram* $\mathcal{C}$ is a sequence of color arrays $(C_1, \ldots, C_k)$, where each color array $C_i = [c_{i,1}, \ldots, c_{i,d_i}]$, $1 \leq i \leq k$, consists of distinct colors. A color diagram $\mathcal{C}$ is *monotonic* if the color $c_{i,j}$ occurs at most $d_i - j$ times in $C_1, \ldots, C_{i-1}$ for all $1 \leq i \leq k$ and $1 \leq j \leq d_i$.

A monotonic color diagram can be used to provide an edge-coloring for a bipartite graph.

Let B be a bipartite graph with bipartition $\{U, V\}$. Consider $|U| = k$ and $U = \{u_1, u_2, \ldots, u_k\}$. For each vertex $u_i \in U$, let C_{u_i} be a set of $d_B(u_i)$ distinct colors. Consider the set $\mathcal{C} = (C_{u_1}, \ldots, C_{u_k})$ and suppose that $\mathcal{C}$ is a monotonic color diagram. In this case, it is possible to color the edges of B using the set of colors $C_{u_i} = [c_{i,1}, \ldots, c_{i,d_B(u_i)}]$ to color the edges incident to u_i, for each $u_i \in U$. The algorithm is a greedy one. It considers the vertices of U ordered according to $\mathcal{C}$ and, for each u_i, using consecutively the colors in C_{u_i} it choose a neighbor of u_i where does not incide an edge with color $c_{i,j}$ and colors this edge with the color $c_{i,j}$. Since $\mathcal{C}$ is a monotonic color diagram, when it is coloring an edge of u_i and is considering the color $c_{i,j}$, this color has already occurred at most $d_B(u_i) - j$ times in $C_{u_1}, \ldots, C_{u_{i-1}}$. But there are at least $d_B(u_i) - j + 1$ not colored edges incident to u_i. Then there is at least one edge incident to u_i which can be colored with $c_{i,j}$. This result is in the following lemma and we use it in our study of edge-coloring of split graphs.

Lemma 1 *[1] Let B be a bipartite graph with bipartition $\{U, V\}$. Let $\mathcal{C} = (C_{u_1}, \ldots, C_{u_k})$ be a monotonic color diagram with each C_{u_i} defined as above, then B has an edge-coloring that uses the colors of C_{u_i}, $u_i \in U$.*

A $k \times k$-matrix with entries from $\{0, \ldots, k-1\}$ is called *latin square of order* k if every element of $\{0, \ldots, k-1\}$ appears in each row and column exactly once. A latin square $M = [m_{i,j}]$ is *commutative* if $m_{i,j} = m_{j,i}$ for $0 \leq i \leq j \leq k-1$ and it is *idempotent* if $m_{i,i} = i$, for $0 \leq i \leq k-1$.

Consider the $k \times k$-matrix, $M = [m_{i,j}]$, $0 \leq i \leq k-1$ and $0 \leq j \leq k-1$, defined as following.

$$m_{i,j} = (i + j) \pmod k \tag{1}$$

Note that M is a commutative latin square and it is possible to construct monotonic color diagrams, using the entries of the lines of M as a sequence of color arrays. For instance, it is easy to see that, for a fixed j, $0 \leq j \leq k-1$, the sequence $\mathcal{C} = (C_0, \ldots, C_j)$, where $C_i = [m_{i,j}, m_{i,j+1}, \ldots, m_{i,k-1}]$, $0 \leq i \leq j$, is a monotonic color diagram.

From now on, G is a split graph with a partition $\{Q, S\}$, where Q is a maximal clique and S is a stable set. Note that Q is also a maximum clique of G. To every split graph G we shall associate the bipartite graph B obtained from G by removing all edges of the subgraph of G induced by Q. Let $d(Q)$ be the maximum degree of vertices of Q in the bipartite graph B, i.e., $d(Q) = \max\limits_{v \in Q}\{d_B(v)\}$. Then $\Delta(G) = |Q| - 1 + d(Q)$.

Chen, Fu and Ko [1], use an odd order idempotent commutative latin square to show that an odd maximum degree split graph is Class 1. Since it is known that there is an idempotent commutative latin square of order n if, only if, n is odd, this technique could not be directly applied on split graphs with even maximum degree.

In order to provide an edge-coloring with $\Delta(G)$ colors for some split graphs when $\Delta(G)$ is even, we consider the matrix M defined in (1). From now on, the entries of a matrix are called colors. The matrix M is a commutative latin square of order $\Delta(G) - 1$, so we need a new color. We replace some entries of M with the new color, as described in Algorithm ColorDiagrams. (The effect of these replacements will be the coloring of a selected set of independent edges of $G[Q]$ with the new color.) Then, we consider a vertex v in S with degree at least $\frac{|Q|}{2}$ and we label the vertices of Q as $u_1, u_2, \ldots, u_{|Q|}$ such that u_i is adjacent to v, $0 \leq i \leq d_G(v)$. Hence, we use the color $a_{i,i}$ of the submatrix $A = [a_{i,j}]$ formed by the first $|Q|$ rows and columns of M to color the edge (v, u_i), $0 \leq i \leq d_G(v)$. After, we use the color $a_{i,j}$ to color the edge (u_i, u_j) of the subgraph of G induced by Q, $0 \leq i, j \leq |Q|$. Now, it remains to color at most $d(Q) - 1$ edges of B that are incident to u_i, for each u_i, $1 \leq i \leq d_B(v)$, and at most $d(Q)$ edges of B incident to u_i, for each u_i, $d_B(v) < u_i \leq |Q|$. We use the monotonic color diagram C produced by our algorithm to color these edges. The constraints of Theorem 4 are given by this strategy. The algorithm used in our approach is given in the next section.

3 A split graph subset that is not neighborhood-overfull

In this section we describe a subset of split graphs that is Class 1. For this, we present the Algorithm ColorDiagrams that constructs a latin square M and derives from it a matrix A and a sequence of colors arrays $\mathcal{C}$. If a split graph G has some special conditions, the sequence $\mathcal{C}$ is a color diagram. In this case, we can perform an edge-coloring of G with Δ colors using the matrix A and the color diagram $\mathcal{C}$ returned by the algorithm.

Algorithm ColorDiagrams(Δ, $|Q|$, $d(v)$)
Input: The positive integers $\Delta \geq 3$, $|Q| < \Delta$, and $d(v) < |Q|$.
Construct a $(\Delta - 1) \times (\Delta - 1)$-matrix M where
$m_{i,j} = (i + j) \pmod{\Delta - 1}$.
Construct a sequence $\mathcal{C} = (C_0, \ldots, C_{|Q|-1})$, where
$C_i = [m_{i,|Q|}, \ldots, m_{i,\Delta-2}]$, $0 \leq i < |Q|$.
Add $m_{i,i}$ of M as the first element of C_i, $d(v) \leq i < |Q|$.
Add $\Delta - 1$ as the first element of C_i, $0 \leq i < |Q|$.
Construct a matrix $A_{|Q|,|Q|}$, where $a_{i,j} \leftarrow m_{i,j}$, $0 \leq i, j < |Q|$;
$l \leftarrow 0$; $l' \leftarrow |Q| - 1$; $x \leftarrow -1$; $c \leftarrow |Q| + x$;
If c is odd $count \leftarrow \lfloor \frac{\Delta - |Q| - x - 1}{2} \rfloor$;
If c is even $count \leftarrow \lfloor \frac{\Delta - |Q| - x - 2}{2} \rfloor$;
While $(l < l')$ and $(c < \Delta - 2)$ do
  Replace the color c from $a_{l,l'}$ and $a_{l',l}$ of A by $\Delta - 1$;
  Replace the color $\Delta - 1$ of C_l and $C_{l'}$ by c;
  $l \leftarrow l + 1$; $l' \leftarrow l' - 1$; $count \leftarrow count - 1$;
  if $count = 0$ then
    $x \leftarrow x + 1$; $c \leftarrow |Q| + x$;
    if c is odd then $count \leftarrow \lfloor \frac{\Delta - |Q| - x - 1}{2} \rfloor$;
    if c is even then $count \leftarrow \lfloor \frac{\Delta - |Q| - x - 2}{2} \rfloor$;
    $l \leftarrow l + 1$;
Return(A, $\mathcal{C}$).

The following results will be used in Theorem 4. The Lemma 2 shows some proprerties of the matrix A returned by the Algorithm ColorDiagrams and the Lemma 3 exhibits the conditions on Δ, $|Q|$, and $d(v)$ of a split graph G which are necessaries to get $\mathcal{C}$ as a color diagram.

Lemma 2 *The matrix A returned by Algorithm ColorDiagrams is commutative, its elements are from $\{0, ..., \Delta - 1\}$, and it has pairwise distinct elements in each line and column. Moreover, if Δ is even, the elements of the main diagonal of A are pairwise distinct.*

Proof. The algorithm constructs the matrix A using the first $|Q|$ lines and the first $|Q|$ columns of matrix the M. This is possible, because $|Q| \leq \Delta - 1$. So, since M is a commutative latin square, before the replacements, the matrix A is commutative and its elements in any line or column are pairwise distinct. Since the matrix A, before the replacements, is a submatrix of M, then its elements are in the set $\{0, 1, \ldots, \Delta - 2\}$. (Note that A is not a latin square.) The only color used for replacements on the cells of A is the color $\Delta - 1$. Therefore, the elements of matrix A returned by Algorithm ColorDiagrams are from $\{0, ..., \Delta - 1\}$.

The loop *while* of the algorithm considers each color that belongs to $F = \{|Q| - 1, |Q|, \ldots, \Delta - 3\}$ in increasing order. It starts with color $c = |Q| - 1$ and replaces it by $\Delta - 1$ in the cells $a_{l,l'} = a_{0,|Q|-1}$ and $a_{l',l} = a_{|Q|-1,0}$. Note that when l' is decreased by one, l is increased by one or two and two cells receive the color $\Delta - 1$. This loop finishes when all the colors were used, i.e. when $c = \Delta - 2$, or when all lines of A were visited, i.e. $l \geq l'$. Note that when the loop finishes each line and column were considered by the algorithm at most one time. Hence each line and each column receives at most one color $\Delta - 1$. Since the elements in each line and column of A, before the replacements, are pairwise distinct and after each replacement the line l and column l' are incremented and decremented, respectively, the matrix A returned by the Algorithm ColorDiagrams has also pairwise distinct elements in each line and column.

Since before the replacements the matrix A is commutative and each replacement is performed at $a_{l,l'}$ and at $a_{l',l}$, the matrix A returned by Algorithm ColorDiagrams is commutative too.

By hypothesis, Δ is even. Thus the matrix M that is used to construct A has odd order $(\Delta - 1)$. So, before the replacements, the elements of the main diagonal of A are pairwise distinct. Since the algorithm does not perform any replacement in the main diagonal of A (it stops when $l \geq l'$), these elements remain pairwise distinct. ■

Lemma 3 *Let $d(v) \geq \frac{|Q|}{2}$ and Δ even. Suppose that $(d(Q))^2 \geq 2|Q|+1$. Then the sequence $\mathcal{C}$ returned by the Algorithm ColorDiagrams is a monotonic color diagram.*

Proof. First of all, the Algorithm ColorDiagrams uses a subset of the elements of line i of matrix M to inicialize C_i, $0 \leq i < |Q|$. By construction, each C_i has size $\Delta - |Q| - 1 = d(Q) - 2$. (Remember that $\Delta = |Q| - 1 + d(Q)$.) Since M is a latin square, at this moment, the colors in each C_i are pairwise distinct and, since $0 \leq i < |Q|$, the sequence $\mathcal{C}$ is a monotonic color diagram. Hence, each color that belongs to $\mathcal{C}$ appears in this color diagram at most $|C_i| = d(Q) - 2$ times.

The Algorithm ColorDiagrams includes the element $m_{i,i}$ of the main diagonal of M as the first element of each sequence C_i, $d(v) \leq i < |Q|$. We show that, after this operation, the sequence $\mathcal{C}$ remains a monotonic color diagram. The colors that are in each new $C_i = \{m_{i,i}, m_{i,|Q|}, \ldots, m_{i,\Delta-1}\}$ are pairwise distinct, since they belong to the line i of M and M is a latin square. By hypothesis, Δ is even, then M is a latin square with odd order and, therefore, the elements of the main diagonal of M are pairwise distinct. So no color is included twice in this step. Moreover, since $d(v) \geq \frac{|Q|}{2}$, then no C_j with $j > i$ has the color $m_{i,i}$. Remember that before the inclusion of the elements of the main diagonal in C_i, each color appeared in $\mathcal{C}$ at most $d(Q) - 2$ times. Since $m_{i,i}$ is the first element of each C_i and the cardinality of each one of these C_i is $d(Q) - 1$, then each $m_{i,i}$ can appear in $C_0, \ldots, C_j$, $j < i$, at most $d(Q) - 2$ times. Therefore, the sequence $\mathcal{C}$ remains a monotonic color diagram.

In the last step, the Algorithm ColorDiagrams includes a new element at the first position of some C_i, $0 \leq i \leq |Q| - 1$. It inicializes the first position of each C_i as $\Delta - 1$ and, after, replaces some of them by one color of the set $F = \{|Q| - 1, |Q|, \ldots, \Delta - 3\} = \{|Q| + x, -1 \leq x \leq \Delta - |Q| - 3 = d(Q) - 4\}$. In order to guarantee the sequence $\mathcal{C}$ returned by the Algorithm ColorDiagrams is a monotonic color diagram, the Algorithm replaces the color $\Delta - 1$ of some sequences C_i by a color $|Q| + x \in F$ such that the total number of times that the color $|Q| + x$ appears in $\mathcal{C}$ is at most $\Delta - |Q| = d(Q) - 1$. So, each color $|Q| + x \in C_i$ appears at most $d(Q) - 2$ times in $C_0, \ldots, C_j$, $j < i$.

Now we show how many times the color $Q + x \in F$ appears in $\mathcal{C} = C_i$, $0 \le i \le |Q| - 1$, before of the execution of the loop while of the algorithm. There are two cases:

- if the color $|Q| + x$ is odd, it appears at most $x + 1$ times.
- if the color $|Q| + x$ is even, it appears at most $x + 1$ or $x + 2$ times. In fact, when $\frac{|Q|}{2} < d(v) < \frac{\Delta}{2}$, the color $|Q| + x$ belongs to the main diagonal of M and it had been included in $\mathcal{C}$ in the previous steps of the algorithm. So, in this case, it appears at most $x + 2$ times in $\mathcal{C}$. Otherwise, the color $|Q| + x$ appears at most $x + 1$ times.

So, if $|Q|+x$ is odd, this color can be included in $\mathcal{C}$ at most $d(Q)-1-(x+1) = d(Q)-2-x$ times. Since the Algorithm ColorDiagrams performs the replacement of the color $\Delta - 1$ by the color $|Q| + x$ in C_l and $C_{l'}$ at the same time, this pair of replacements occurs exactly $\left\lfloor \frac{d(Q)-2-x}{2} \right\rfloor = \left\lfloor \frac{\Delta-|Q|-x-1}{2} \right\rfloor$ times, unless l becomes greater than l' and the algorithm finishes.

Analogously, if $|Q|+x$ is even, the color $\Delta-1$ can be replaced at most $d(Q)-1-(x+2) = d(Q) - 3 - x$ times by $|Q| + x$ and the pair of replacements (in C_l and $C_{l'}$) occurs exactly $\left\lfloor \frac{d(Q)-3-x}{2} \right\rfloor$ times, unless l becomes greater than l' and the algorithm finishes.

Considere the sequences C_j that contain $c_{j,k} = |Q| + x$, $k \neq 1$. Note that each sequence C_i that contains $c_{i,1} = |Q| + x$ has $i > j$.

Now we show that, after the last step of the algorithm, there is at most $d(Q) - 1$ colors $\Delta - 1$ in $\mathcal{C}$.

The algorithm uses at most $|F| = d(Q)-2$ distinct colors to perform these replacements. For each distinct color used, the algorithm leaves a sequence with the color $\Delta - 1$. (This occurs when *count* $= 0$.)

If the loop *while* finishes because $l \ge l'$ but with $c < \Delta - 2$, then the number of times that the color $\Delta - 1$ appears in $\mathcal{C}$ at the end of the algorithm is less than $d(Q) - 2$ plus 1 (a color $\Delta - 1$ that is left in C_l, when $l = l' = \left\lfloor \frac{|Q|}{2} \right\rfloor + 1$). Thus, the total number of occorrences of the color $\Delta - 1$ in $\mathcal{C}$ is less than $d(Q) - 1$.

If the loop *while* finishes because $c = \Delta - 2$, then the number times that the color $\Delta - 1$ appears in $\mathcal{C}$ is given bellow.

By hypothesis, $(d(Q))^2 \ge 2|Q| + 1$. If $|Q|$ is even, the first and the last elements of F is odd. Therefore,

$$|Q| - \left(d(Q) - 1 + 2\left(\frac{(d(Q)-3)}{2} + \frac{(d(Q)-3)}{2} + \frac{(d(Q)-5)}{2} + \ldots + \frac{2}{2}\right)\right) =$$

$$|Q| - (d(Q) - 1 + 2(d(Q) - 3 + d(Q) - 5 + \cdots + d(Q) - (d(Q) - 2))) =$$

$$|Q| - \left(d(Q) - 1 + \frac{(d(Q)-1)(d(Q)-3)}{2}\right) =$$

$$|Q| - \frac{(d(Q)-1)^2}{2} \le \frac{(d(Q))^2 - 1}{2} - \frac{(d(Q)-1)^2}{2} = d(Q) - 1$$

If $|Q|$ is odd, then $(d(Q))^2$ is even and then $(d(Q))^2 \geq 2|Q| + 2$. Hence,

$$|Q| - 2(d(Q) - 2 + d(Q) - 4 + \cdots + d(Q) - (d(Q) - 2)) =$$
$$|Q| - \left(\frac{d(Q)(d(Q) - 2)}{2}\right) \leq$$
$$\frac{(d(Q))^2 - 2}{2} - \frac{d(Q)(d(Q) - 2)}{2} = d(Q) - 1$$

Remember that each color $\Delta - 1 \in C_i$ appears in the first position of C_i. Since there is at most $d(Q) - 1$ colors $\Delta - 1$ in $\mathcal{C}$, then there is at most $d(Q) - 2$ colors $\Delta - 1$ in $C_0, \ldots, C_j$ with $j < i$. Therefore, the sequence $\mathcal{C}$ returned by the Algorithm ColorDiagrams is a monotonic color diagram. This conclusion follows from the counting of the total number of each color $|Q| + x$ and each color $\Delta - 1$ that appears in $\mathcal{C}$. ■

Now, we are ready to prove the Theorem 4.

Theorem 4 *Let G be a split graph with even maximum degree. If G has a vertex v in S with degree at least $\frac{|Q|}{2}$ and $(d(Q))^2 \geq 2|Q| + 1$, then G is Class 1.*

Proof. Let G be a split graph with $\Delta = \Delta(G)$ even. Let $\{Q, S\}$ be a partition of the vertex set of G, where Q is a maximal clique and S is a stable set. Note that removing edges from a graph cannot increase its chromatic index, so it suffices to show that $\chi'(G) = \Delta$ when all vertices of Q has degree equal to Δ. Remember that B_G is the bipartite graph obtained from G by removing all edges of $G[Q]$, $d(Q) = \max\limits_{v \in Q}\{d_B(v)\}$, and $\Delta(G) = |Q| - 1 + d(Q)$.

Suppose that G has a vertex v in S with $d(v) \geq \frac{|Q|}{2}$ and $(d(Q))^2 \geq 2|Q| + 1$.

We order the vertices in $S = \{v_0, \ldots, v_{|S|-1}\}$ such that $v_0 = v$ and we order the vertices in $Q = \{w_0, \ldots, w_{|Q|}\}$ such that the $d(v_0)$ first vertices in Q are adjacent to v_0.

Let A be the matrix returned by Algorithm ColorDiagrams. For $0 \leq i, j \leq |Q| - 1$, we use the color $a_{i,j}$ to color the edge $\{w_i, w_j\}$. By Lemma 2, the elements of any line of matrix A are pairwise distinct and A is commutative. Hence, this is an edge-coloring of $G[Q]$. Then it remains to color the edges of B_G.

Now we color the edges of B_G that are incident to $v_0 = v$. By hypothesis, $d(v_0) \geq \frac{|Q|}{2}$. We use the color $a_{i,i}$ of the main diagonal of A to color the edge $\{v_0, w_i\}$, $0 \leq i < d(v_0)$. Since Δ is even, by Lemma 2, the elements of the main diagonal of A are pairwise distinct. Hence the edges incident to v_0 have distinct colors. Since G is a simple graph, only colors $a_{i,j}$ with $i \neq j$ are used to color the edges of $G[Q]$. Then each color $a_{i,i}$ differs of all the other colors that are in edges incident to w_i. Therefore, we have an edge-coloring of $G[Q \cup \{v_0\}]$.

Let B' the bipartite graph induced by the edges with a vertex in Q and another one in $S \backslash \{v_0\}$. Consider now the color diagram $\mathcal{C}$ returned by the Algorithm ColorDiagrams. The first $d(v)$ sequences of $\mathcal{C}$ have size $d(Q) - 1$ and the other ones have size $d(Q)$. In fact, $C_i = [c, m_{i,|Q|}, \ldots, m_{i,\Delta-2}]$, for $0 \leq i < d(v)$, and $C_i = [c, m_{i,i}, m_{i,|Q|}, \ldots, m_{i,\Delta-2}]$, for $d(v) \leq i < |Q|$, where $c \in F = \{|Q| - 1, |Q|, \ldots, \Delta - 3, \Delta - 1\}$.

Remember that $d_{B_G}(w_i) \leq d(Q)$, $0 \leq i \leq |Q| - 1$. Since we label the vertices of Q such that the first $d(v_0)$ vertices are the neighbors of v_0, then these vertices have degree at most $d(Q) - 1$ in B'. The size of a sequence C_i is at least $d_{B'}(w_i)$, $0 \leq i < |Q|$. By Lemma 3, $\mathcal{C}$ is a monotonic color diagram. Thus, by Lemma 1, we can color the edges of B' using the elements of the color diagram $\mathcal{C}$ and we have an edge-coloring of B'.

Now, we have to show that the colors of the edges incident to w_i in G are pairwise distinct, for $0 \leq i < |Q|$. To see this, remember that for each i, $0 \leq i < |Q|$, the colors of $\{w_i, w_j\}$ belong to line i of A, the colors of $\{w_i, v_j\}$, $v_j \in S$, belong to C_i, and each C_i has at most one color $\Delta - 1$. Remember, also, that the elements of line i of matrix A and the elements of C_i, except $\Delta - 1$, belong to line i of M (the matrix defined in (1)). Since M is a latin square, the colors of the edges incident to each w_i, $0 \leq i < |Q|$, are pairwise distinct. Then we have a an edge-coloring with Δ colors for G. Therefore, G is Class 1. ■

A split graph satisfying the conditions of Theorem 4 is not neighborhood-overfull. So, our result gives a positive evidence to the conjecture that for any split graph neighborhood-overfullness is equivalent to being Class 2.

References

[1] B-L. Chen, H-L. Fu and M. T. Ko, *Total chromatic number and chromatic index of split graphs*, Journal of Combinatorial Mathematics and Combinatorial Computing, **17**, 137–146 (1995).

[2] D. G. Corneil and Y. Perl, *Clustering and domination in perfect graphs*, Discrete Applied Mathematics, **9**, 27–39 (1984).

[3] C. M. H. de Figueiredo, J. Meidanis and C. P. de Mello, *Local conditions for edge-coloring*, Journal of Combinatorial Mathematics and Combinatorial Computing, **32**, 79–91 (2000).

[4] S. Földes and P. L. Hammer, *Split Graphs*, Congressus Numerantium, **19**, 311–315 (1977).

[5] A. J. W. Hilton, *Two conjectures on edge-coloring*, Discrete Mathematics, **74**, 61–64 (1989).

[6] D. G. Hoffman and C. A. Rodger, *The chromatic index of complete multipartite graphs*, Journal of Graph Theory, **16**, 159–163 (1992).

[7] I. Holyer, *The NP-completeness of edge-coloring*, SIAM J. Comput., **10**, 718–720 (1981).

[8] D. König, *Über Graphen und ihre Anwendung auf Determinantentheorie und Mengenlehre*, Mathematische Annalen, **77**, 453–465 (1916).

[9] C. Ortiz, N. Maculan and J. L. Szwarcfiter, *Characterizing and edge-coloring split-indiference graphs*, Discrete Applied Mathematics, **82**, 209–217 (1998).

[10] M. Plantholt, *The chromatic index of graphs with a spanning star*, Journal of Graph Theory, **5**, 45–53 (1981).

[11] G. F. Royle, *Counting set covers and split graphs*, Journal of Integer Sequences, **3**, 1–5 (2000).

[12] V. G. Vizing, *On an estimate of the chromatic class of a p-graph*, Diket. Analiz., **3**, 25–30 (1964).