On the (k, i)-coloring of cacti and complete graphs[1]

Flavia Bonomo[a], Guillermo Durán[b], Ivo Koch[c] and Mario Valencia-Pabon[d,2]

[a] *CONICET and Dep. de Computación, FCEyN, Universidad de Buenos Aires, Argentina*
[b] *CONICET and Dep. de Matemática and Instituto de Cálculo, FCEyN, Universidad de Buenos Aires, Argentina*
Dep. de Ingeniería Industrial, FCFM, Universidad de Chile, Santiago, Chile
[c] *Instituto de Ciencias, Universidad Nacional de General Sarmiento, Argentina*
[d] *Université Paris-13, Sorbonne Paris Cité LIPN, CNRS UMR7030, Villetaneuse, France*

Abstract

In the (k, i)-*coloring problem*, we aim to assign sets of colors of size k to the vertices of a graph G, so that the sets which belong to adjacent vertices of G intersect in no more than i elements and the total number of colors used is minimum. This minimum number of colors is called the (k, i)-*chromatic number*. We present in this work a very simple linear time algorithm to compute an optimum (k, i)-coloring of cycles and we generalize the result in order to derive a polynomial time algorithm for this problem on cacti. We also perform a slight modification to the algorithm in order to obtain a simpler algorithm for the close coloring problem addressed in [R.C. Brigham and R.D. Dutton, Generalized k-tuple colorings of cycles and other graphs, J. Combin. Theory B 32:90–94, 1982]. Finally, we present a relation between the (k, i)-coloring problem on complete graphs and weighted binary codes.

Keywords: Generalized k-tuple coloring, (k, i)-coloring, cactus, complete graphs.

[1] Partially supported by UBACyT Grant 20020130100808BA, CONICET PIP 112-201201-00450CO and ANPCyT PICT 2012-1324 (Argentina), FONDECyT Grant 1140787 and Millennium Science Institute "Complex Engineering Systems" (Chile), and MathAmSud Project 13MATH-07 (Argentina–Brazil–Chile–France).

[2] Currently in "Délégation" at the INRIA Nancy - Grand Est, 2013-2015.

1 Introduction

We consider finite undirected graphs without loops. A classic *coloring* (i.e. *proper coloring*) of a graph G is an assignment of colors (or natural numbers) to the vertices of G such that any two adjacent vertices are assigned different colors. The smallest number t such that G admits a coloring with t colors (a *t-coloring*) is called the *chromatic number* of G and is denoted by $\chi(G)$.

Several generalizations of the coloring problem were introduced in the literature, in particular, cases in which each vertex is assigned not only a color but a set of colors, under different restrictions. One of these variations is the *k-tuple coloring* introduced independently by Hilton, Rado and Scott [13], Stahl [19], and Bollobás and Thomason [3]. A k-tuple coloring of a graph G is an assignment of k colors to each vertex in such a way that adjacent vertices are assigned distinct colors.

Brigham and Dutton [4] generalize the concept of k-tuple coloring by introducing the concept of *$k : i$-coloring*, in which the sets of colors assigned to adjacent vertices intersect in exactly i colors. The *$k : i$-coloring problem* consists into finding the minimum number of colors in a $k : i$-coloring of a graph G, which we denote by $\chi_k^{(i)}(G)$. By using a theorem of Stahl ([19], p.193), Brigham and Dutton obtain the following result on cycles.

Theorem 1.1 *[4] Let C_n be a cycle, with $n = 2t + 1$. Then,*

$$\chi_k^{(i)}(C_n) = \begin{cases} 2k - i & \text{if } k \leq i(t+1), \\ 2k - i + 1 + \lfloor \frac{k-i(t+1)-1}{t} \rfloor = \lceil \frac{n(k-i)}{t} \rceil & \text{if } k > i(t+1). \end{cases}$$

However, it is not so evident how to construct efficiently a $k : i$-coloring of an odd cycle C_n with $\chi_k^{(i)}(C_n)$ colors in polynomial time.

In this paper, we will deal with another generalization, known as (k, i)-*coloring*, introduced by Méndez-Díaz and Zabala in [17], in which the sets of colors assigned to adjacent vertices intersect in at most i colors. Formally, let $G = (V, E)$ be a graph and let k, i, j be non-negative integers, with $0 \leq i \leq k \leq j$ then, a (k, i)-coloring of G with j colors is a function $c : V(G) \to [j]^k$ that assigns to each vertex $v \in V$ a k-subset of the set $[j] = \{1, 2, \ldots, j\}$ such that each pair of adjacent vertices u, v verifies $|c(v) \cap c(u)| \leq i$. The minimum positive integer j such that G admits a

(k,i)-coloring with j colors is called the (k,i)-*chromatic number* and it is denoted by $\chi_k^i(G)$. Note that for $k=1$, $i=0$, we have the classical coloring problem and thus $\chi_1^0(G)=\chi(G)$ for any graph G. For arbitrary k and $i=0$, we have the k-tuple coloring.

Note that $\chi_k^i(G) \leq \chi_k^{(i)}(G)$, since every $k:i$-coloring is in particular a (k,i)-coloring, but they are not necessarily equal, even for complete graphs. We will provide an example in Section 3.

From another point of view, a graph G is t-colorable if and only if there is a graph homomorphism from G to the complete graph on t vertices K_t, where an homomorphism from a graph G to a graph H is an edge preserving map between G and H. Denley [5] introduced the generalized Kneser graphs $K(j,k,i)$ as follows. Let i,j,k be integers such that $0 \leq i \leq k \leq j$. Define the graph $K(j,k,i)$ as the graph having as set of vertices the family of k-subsets of $\{1,\ldots,j\}$, and where two k-subsets A and B are adjacent if and only if $|A \cap B| \leq i$. When $i=0$, the graphs $K(j,k,0)$ are the well known Kneser graphs [7]. It is not difficult to see that a graph G admits a (k,i)-coloring with j colors if and only if there is a graph homomorphism from G to $K(j,k,i)$.

Méndez-Díaz and Zabala solved in [17] the (k,i)-coloring problem for some values of k and i on complete graphs, studied the notion of perfectness and criticality for the (k,i)-coloring problem and gave general bounds for the (k,i)-chromatic number. The authors proposed also an heuristic approach and a linear programming model for the problem, which they further developed and generalized in [18].

Graph coloring problems on cactus graphs were also studied in several articles (see for example [6, 16, 15]). We present in this work a linear time algorithm to compute the (k,i)-chromatic number of cycles and we generalize the result in order to derive a polynomial algorithm for this problem on cacti. We also show that these results hold for the $k:i$-chromatic number of cycles and cacti. Finally, we present a relation between the (k,i)-coloring problem on complete graphs and weighted binary codes.

1.1 Definitions and preliminary results

For standard definitions in graph theory not included in this section, we refer to [2]. The *line graph* $L(G)$ of a graph $G=(V,E)$ is the graph having as its vertex set the set E of edges, two vertices in $L(G)$ being adjacent if their corresponding edges in G are incident.

A *multigraph* is a graph where parallel edges are allowed. *Multicycles* are cycles in which we can have parallel edges between two consecutive vertices. A multigraph is *k-uniform* if the number of parallel edges between any two adjacent vertices is exactly k.
An *edge coloring* of a (multi)graph G is an application from the edge set E to a set of colors such that incident edges are assigned different colors. The minimum number of colors in an edge coloring of G is called the *chromatic index* $\chi'(G)$.
An *independent set* (respectively *matching*) of a graph G is a subset of vertices (respectively edges) pairwise non-adjacent (respectively non-incident). Clearly, a matching in G corresponds to an independent set in $L(G)$ and vice-versa.
A vertex v in a connected graph G is called a *cut-vertex* if $G \setminus \{v\}$ is unconnected. A *block* is a maximal biconnected subgraph (i.e., a maximal connected subgraph without cut-vertices) of a graph. An *end-block* is a block containing exactly one cut-vertex. It is known that every connected graph that is not biconnected has an end-block.

Let G be a (multi)cycle on n vertices, $m \geq n$ edges and maximum degree equal to Δ. It is well known that $\chi'(G) = \Delta$ if n is even. In fact, it follows from König's Theorem on edge-coloring of bipartite (multi)graphs. When n is odd, we have the following result due to Berge.

Theorem 1.2 *[2] Let $G = (V, E)$ be a multicycle on n vertices with m edges and maximum degree Δ. Let $\tau = \lfloor \frac{n}{2} \rfloor$ denote the maximum cardinality of a matching in G. Then*

$$\chi'(G) = \begin{cases} \Delta & \text{if } n \text{ is even,} \\ max\{\Delta, \lceil \frac{m}{\tau} \rceil\} & \text{if } n \text{ is odd} \end{cases}$$

Let G be a k-uniform multicycle on n vertices. It is not difficult to see that the line graph $L(G)$ of G can be seen as the cycle C_n where each vertex is replaced by a clique of size k and all edges between two disjoint copies of K_k associated with two adjacent vertices in C_n are added. Therefore, we can rephrase Theorem 1.2 for k-uniform multicycles in terms of a vertex coloring problem of $L(G)$ as follows.

Corollary 1.3 *Let $L(G)$ be the line graph of a k-uniform multicycle G on n vertices. Let $\alpha = \lfloor \frac{n}{2} \rfloor$ denote the maximum cardinality of an inde-*

pendent set in $L(G)$. *Then*

$$\chi(L(G)) = \begin{cases} 2k & \textit{if } n \textit{ is even,} \\ max\{2k, \lceil \frac{nk}{\alpha} \rceil\} & \textit{if } n \textit{ is odd} \end{cases}$$

Corollary 1.3 has been obtained independently by Stahl [19].

2 (k, i)-coloring of cycles

It was already noticed in [17] that a bipartite graph has (k, i)-chromatic number at most $2k-i$, and that this is also the trivial lower bound for the (k, i)-chromatic number of any graph with at least one edge. Since even cycles are bipartite, this case is solved, and we will turn our attention to the odd case. In this section, we obtain a similar result as the one found by Brigham and Dutton [4] on odd cycles (Theorem 1.1). We prove that the (k, i)-chromatic number and the $k : i$-chromatic number are equal on odd cycles. Furthermore, we derive a simple linear time algorithm to (k, i)-color an odd cycle with the minimum number of colors, and we adapt it also for $k : i$-coloring.

We will compute first a lower bound for the (k, i)-chromatic number of C_n as follows.

Lemma 2.1 *Let* C_n *be a cycle on* $n = 2t + 1$ *vertices. Then, for any non-negative integers* i, k *with* $0 \leq i \leq k$, *we have that :* $\chi_k^i(C_n) \geq \max\{2k - i, \lceil \frac{n(k-i)}{t} \rceil\}$.

Proof. Notice that $2k - i$ is a trivial lower bound for any graph with at least one edge. So, we only need to prove that $\chi_k^i(C_n) \geq \lceil \frac{n(k-i)}{t} \rceil$, where $n = 2t + 1$. Assume that the vertices of C_n are labeled consecutively by $v_0, \ldots, v_{n-1}$. Arithmetic operations will be taken modulo n. Let c be an optimum (k, i)-coloring of the vertices of C_n, that is, for each vertex v_i we have that $|c(v_i)| = k$; for each pair of adjacent vertices v_i, v_{i+1} we have that $|c(v_i) \cap c(v_{i+1})| \leq i$; and the maximum color included in the set $\bigcup_{v \in V(C_n)} c(v)$ is equal to χ_k^i. Now, for each vertex v_i in C_n, let $c'(v_i) = c(v_i) \setminus (c(v_i) \cap c(v_{i+1}))$. Notice that the size of each set $c'(v_i)$ is at least $k-i$, and that $c'(v_i) \cap c'(v_{i+1}) = \emptyset$ for every $i = 1, \ldots, n$. Therefore, it is not difficult to deduce that the sets c' can be used in order to color the vertices of the line graph of a multicycle on n vertices having at least $k-i$ parallel edges between each pair of adjacent vertices. By Corollary 1.3, the result follows. □

Now, in order to compute an upper bound for the (k,i)-chromatic number of cycles, we will construct a (k,i)-coloring for these graphs. First, we need the following lemma.

Lemma 2.2 *Let n, n' be two odd integers, with $n' > n \geq 3$. Then any (k,i)-coloring of C_n can be extended to a (k,i)-coloring of $C_{n'}$ without using additional colors.*

Proof. Let $v_1, \ldots, v_n$ be the vertices of C_n and let c be a (k,i)-coloring of C_n. Let $v'_1, \ldots, v'_{n'}$ be the vertices of $C_{n'}$ and define c' as $c'(v'_i) = c(v_i)$ for $i = 1, \ldots, n$; $c'(v_{n+j}) = c(v_{n-1})$ if j is odd, $c'(v_{n+j}) = c(v_n)$ if j is even, for $j = 1, \ldots, n'-n$. It is easy to check that c' is a (k,i)-coloring of $C_{n'}$. □

Based on this, we propose the following simple algorithm.

Lemma 2.3 *Let $n = 2t+1$ with $t \geq 1$. Then, $\chi^i_k(C_n) \leq \max\{2k-i, \lceil \frac{n(k-i)}{t} \rceil\}$. Moreover, a (k,i)-coloring of C_n with $\max\{2k-i, \lceil \frac{n(k-i)}{t} \rceil\}$ colors can be obtained by Algorithm 1.*

Proof. Let us see that the assignment c obtained by Algorithm 1 on C_{2t+1} defines a (k,i)-coloring.

Note that the algorithm assigns circular intervals of size k (i.e., either intervals of k consecutive numbers or intervals formed by the last d and the first $k-d$ numbers) to each vertex of the cycle in such a way that $c(v_1) = \{1, 2, \ldots, k\}$ and for $2 \leq j \leq 2t'+1$, $c(v_j)$ is the circular interval whose first i colors are the last i colors of $c(v_{j-1})$. As we have at least $2k-i$ colors, the intersection of $c(v_j)$ and $c(v_{j-1})$ are exactly those i colors. The property $|c(v_j) \cap c(v_{j-1})| = i$ holds also for $2t'+2 \leq j \leq 2t+1$, when $t' < t$, since they use alternately $c(v_{2t'})$ and $c(v_{2t'+1})$. Therefore, in order to ensure that c is a valid (k,i)-coloring of C_{2t+1}, we just need to check that $|c(v_{2t'+1}) \cap c(v_1)| \leq i$.

By construction, the first number in the circular interval $c(v_{2t'+1})$ is the number d in $[1,N]$ that is congruent to $2t'(k-i)+1$ modulo N. We

Algorithm 1

Input: A cycle C_n, $n = 2t+1$, with vertices $v_1, v_2, \ldots, v_n$, integers k and i with $0 \leq i < k$.

Output: An assignment of k colors to each vertex v_i, i.e. $c(v_i)$, for $i = 1, 2, \ldots, n$, from the set $\{1, \ldots, \max\{2k - i, \lceil \frac{n(k-i)}{t} \rceil\}\}$ to each vertex of C_n.

1: Let $N = \max\{2k-i, \lceil \frac{n(k-i)}{t} \rceil\}$; $\ell = 1$. Let t' be the minimum positive integer value such that $\lceil \frac{(2t'+1)(k-i)}{t'} \rceil = \lceil \frac{n(k-i)}{t} \rceil$, i.e., either $t' = 1$ or $t' > 1$ and $\lceil \frac{(2t'-1)(k-i)}{t'-1} \rceil > \lceil \frac{n(k-i)}{t} \rceil$. (This value can be obtained by binary search.)

2: For $j = 1$ to $2t' + 1$ do:
  If $\ell + k - 1 \leq N$ then
    $c(v_j) = [\ell, \ell + k - 1]$
  else
    $c(v_j) = [\ell, N] \cup [1, \ell + k - 1 - N]$
  end if
  If $\ell + k - i \leq N$ then
    $\ell = \ell + k - i$
  else
    $\ell = \ell + k - i - N$
  end if
end for

3: For $j = t' + 1$ to t do:
  $c(v_{2j}) = c(v_{2t'})$
  $c(v_{2j+1}) = c(v_{2t'+1})$
end for

should prove

$$k - i + 1 \leq d \leq N - (k - i) + 1.$$

If $t' = 1$, then $2t'(k-i)+1 = 2(k-i)+1$ and it holds $k-i+1 \leq 2(k-i)+1$. Also, $2(k-i) + 1 \leq N - (k-i) + 1$ if and only if $3(k-i) \leq N$, but $N = \max\{2k-i, \lceil \frac{n(k-i)}{t} \rceil\}$ and $\lceil \frac{n(k-i)}{t} \rceil = \lceil \frac{(2t'+1)(k-i)}{t'} \rceil = 3(k-i)$, so $d = 2(k-i)+1$ and this finishes the case $t' = 1$. Assume from now on that $t' > 1$ and $\lceil \frac{n(k-i)}{t} \rceil = \lceil \frac{(2t'+1)(k-i)}{t'} \rceil$ but $\lceil \frac{n(k-i)}{t} \rceil < \lceil \frac{(2t'-1)(k-i)}{t'-1} \rceil$, so $\lceil \frac{n(k-i)}{t} \rceil < \frac{(2t'-1)(k-i)}{t'-1}$. We will split now the proof into two cases, depending on the value of N.

Case 1: $N = 2k-i$. Note that $\lceil \frac{n(k-i)}{t} \rceil = \lceil \frac{(2t+1)(k-i)}{t} \rceil = \lceil \frac{2tk-2ti+k-i}{t} \rceil = 2k - i + \lceil \frac{(k-(t+1)i)}{t} \rceil$. So, $\lceil \frac{n(k-i)}{t} \rceil \leq 2k - i \Leftrightarrow \frac{(k-(t+1)i)}{t} \leq 0 \Leftrightarrow k \leq$

$(t+1)i$. In particular, this will not be the case if $i = 0$. Thus, $\max\{2k - i, \lceil \frac{n(k-i)}{t} \rceil\} = 2k - i$ if and only if $i > 0$ and $\frac{k}{i} \leq t + 1 \Leftrightarrow \lceil \frac{k}{i} \rceil - 1 \leq t$. By our assumption about t' and as we have discarded the case $t' = 1$, it should be $t' = \lceil \frac{k}{i} \rceil - 1$.

But $2t'(k-i)+1 = t'(2k-i) - t'i + 1 \equiv 2k - i - t'i + 1 \pmod{2k-i}$. Since $t'i < k \leq (t'+1)i$, it holds $k - i + 1 < k - i + k - t'i + 1 = k + k - (t'+1)i + 1 \leq k + 1$, so $d = 2k - i - t'i + 1$ and this closes Case 1.

Case 2: $N = \lceil \frac{n(k-i)}{t} \rceil = \lceil \frac{(2t'+1)(k-i)}{t'} \rceil$. By the analysis in Case 1, that means $\frac{(k-(t'+1)i)}{t'} > 0$. Let $b = \lceil \frac{(k-(t'+1)i)}{t'} \rceil$, thus $N = 2k - i + b$. By our assumption about t' and as we have discarded the case $t' = 1$, it should be $\frac{(k-t'i)}{t'-1} > b$.

In this case, $2t'(k-i)+1 = t'N - t'i - t'b + 1 \equiv N - t'i - t'b + 1$ modulo N. On one hand,

$$\begin{aligned} N - t'i - t'b + 1 &\leq N - (k-i) + 1 \quad \Leftrightarrow \\ -t'(i+b) &\leq -(k-i) \quad \Leftrightarrow \\ \frac{k-(t'+1)i}{t'} &\leq b \end{aligned}$$

and this is satisfied because $b = \lceil \frac{(k-(t'+1)i)}{t'} \rceil$. On the other hand,

$$\begin{aligned} k - i + 1 &\leq N - t'i - t'b + 1 = (2k - i + b) - t'(i+b) + 1 \quad \Leftrightarrow \\ (t'-1)b &\leq k - t'i \quad \Leftrightarrow \\ b &\leq \frac{(k-t'i)}{t'-1} \end{aligned}$$

and we have observed that this inequality already holds. So $d = N - t'i - t'b + 1$ and this ends the proof of this lemma. □

Lemma 2.4 *Let C_n be a cycle on $n = 2t+1$ vertices. The running time of Algorithm 1 with input C_n, integers k and i, with $0 \leq i < k$, is $O(n)$.*

Proof. Let $M = \lceil \frac{n(k-i)}{t} \rceil$. We analyze first Step 1 of Algorithm 1. Let t'_c be the current value processed by the binary search, $1 \leq t'_c \leq t$. The

procedure is guaranteed to work due to the fact that the possible values for t'_c contain at least one value that satisfies $\lceil \frac{(2t'_c+1)(k-i)}{t'_c} \rceil = M$, namely t, and to the fact that $\lceil \frac{(2t'_c+1)(k-i)}{t'_c} \rceil$ is non-increasing in t'_c. This means that if $\lceil \frac{(2t'_c+1)(k-i)}{t'_c} \rceil > M$, we may safely discard all values in the set $\{1, \ldots, t'_c\}$. In turn, $\lceil \frac{(2t'_c+1)(k-i)}{t'_c} \rceil \leq M$ implies that values in the set $\{t'_c + 1, \ldots, N\}$ should be disregarded. The search occurs thus in no more than $O(\log t) = O(\log n)$ steps.
For Step 2 of the algorithm, note that the assignment c is built for every vertex v_i of C_n from consecutive colors intervals. Hence, we only need to store the first and last color as a compact representation of the whole color interval. If $c(v_j) = [\ell, \ell + k - 1]$, we assign to $c(v_j)$ a reference to a tuple $< \ell, \ell + k - 1 >$ in $O(1)$. Analogously, if $c(v_j) = [\ell, N] \cup [1, \ell + k - 1 - N]$, we assign to $c(v_j)$ references to tuples $< \ell, N >$ and $< 1, \ell + k - 1 - N >$. There are $2t' + 1$ iterations, and hence this step is $O(2t + 1) = O(n)$.
Step 3 of Algorithm 1 is analogous to Step 2. Tuples are copied and assigned in $O(1)$ during $t - t'$ iterations, clearly resulting in no more than $O(n)$ operations.
Therefore, the total execution time of all three steps (executed sequentially) is $O(max\{\log n, n, n\}) = O(n)$, as desired. □

By the proofs of Lemmas 2.1, 2.3 and 2.4, we have the following result.

Theorem 2.5 *Let C_n be a cycle on $n = 2t+1$ vertices. Then, $\chi_k^i(C_n) = max\{2k - i, \lceil \frac{n(k-i)}{t} \rceil\}$ and a (k, i)-coloring of C_n with $\chi_k^i(C_n)$ colors can be obtained in $O(n)$ time.*

For example, the $(4, 1)$-coloring of C_3 obtained by Algorithm 1 is $\{1, 2, 3, 4\}$, $\{4, 5, 6, 7\}$, $\{7, 8, 9, 1\}$, the $(4, 1)$-coloring of C_5 obtained by Algorithm 1 is $\{1, 2, 3, 4\}$, $\{4, 5, 6, 7\}$, $\{7, 8, 1, 2\}$, $\{2, 3, 4, 5\}$, $\{5, 6, 7, 8\}$, the $(4, 1)$-coloring of C_7 obtained by Algorithm 1 is $\{1, 2, 3, 4\}$, $\{4, 5, 6, 7\}$, $\{7, 1, 2, 3\}$, $\{3, 4, 5, 6\}$, $\{6, 7, 1, 2\}$, $\{2, 3, 4, 5\}$, $\{5, 6, 7, 1\}$, and the $(4, 1)$-coloring of C_{11} obtained by Algorithm 1 is an extension of the coloring of C_7, namely, $\{1, 2, 3, 4\}$, $\{4, 5, 6, 7\}$, $\{7, 1, 2, 3\}$, $\{3, 4, 5, 6\}$, $\{6, 7, 1, 2\}$, $\{2, 3, 4, 5\}$, $\{5, 6, 7, 1\}$, $\{2, 3, 4, 5\}$, $\{5, 6, 7, 1\}$, $\{2, 3, 4, 5\}$, $\{5, 6, 7, 1\}$.

2.1 Extension to the $k : i$-coloring problem

Note that an optimal (k, i)-coloring of C_{2t} is always a $k : i$-coloring, since it uses $2k - i$ colors, but for odd cycles this is not always the case. Indeed, the $(4,1)$-coloring of C_5 obtained by Algorithm 1 is not a $4:1$-coloring, since $c(v_5) \cap c(v_1) = \emptyset$.

Note also that an analogous to Lemma 2.2 can be proved for the $k:i$-coloring problem. We will show now that, if a (k,i)-coloring c of C_{2t+1} is obtained by Algorithm 1, one can modify the set $c(v_{2t+1})$ by a simple procedure, in order to obtain a $k:i$-coloring of C_{2t+1} with the same number of colors.

First notice that $|c(v_i) \cap c(v_{i+1})| = i$ for $i = 1, \dots, 2t$, and $|c(v_{2t+1}) \cap c(v_1)| \leq i$. Assume $|c(v_{2t+1}) \cap c(v_1)| < i$, otherwise we are done. We have to show how to increase $|c(v_{2t+1}) \cap c(v_1)|$ without decreasing $|c(v_{2t+1}) \cap c(v_{2t})|$.

Let us define the following sets: $A = c(v_1) \cap c(v_{2t}) \setminus c(v_{2t+1})$, $B = c(v_1) \cap c(v_{2t}) \cap c(v_{2t+1})$, $C = c(v_1) \setminus (c(v_{2t}) \cup c(v_{2t+1}))$, $D = c(v_{2t}) \setminus (c(v_1) \cup c(v_{2t+1}))$, $E = c(v_{2t}) \cap c(v_{2t+1}) \setminus c(v_1)$, $F = c(v_1) \cap c(v_{2t+1}) \setminus c(v_{2t})$, $G = c(v_{2t+1}) \setminus (c(v_1) \cup c(v_{2t}))$ (see Figure 1), and let $x = |X|$ for $X = A, \dots, G$.

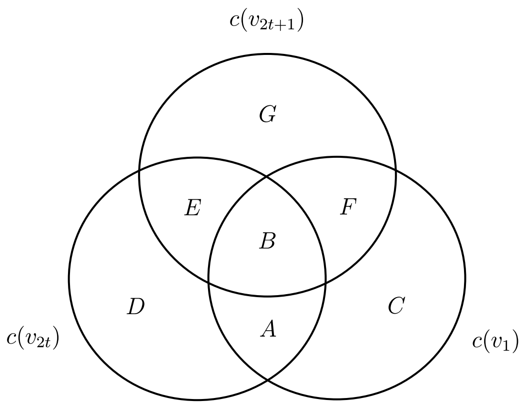

Figure 1: Diagram for the definition of color sets.

If $g > 0$ and $c > 0$, we can replace in $c(v_{2t+1})$ a color from G by a color from C, and if $e > 0$ and $a > 0$, we can replace in $c(v_{2t+1})$ a color from E by a color from A. In both cases, we are increasing $|c(v_{2t+1}) \cap c(v_1)|$ without decreasing $|c(v_{2t+1}) \cap c(v_{2t})|$.

If $c = 0$, the total number of colors used by v_1, v_{2t}, and v_{2t+1} is $2k - i$, so $|c(v_{2t+1}) \cap c(v_1)| \geq i$, a contradiction to our assumption. So, $c > 0$. If $g = 0$ then $e > 0$, otherwise $|c(v_{2t+1})| = b + f < i \leq k$, a contradiction.

Therefore, we only have to show that if $g = 0$ then $a > 0$. Suppose $g = a = 0$. Then $c > k - i$, $d = k - i$, and $b + e + f = k$. So, the total number of colors used by v_1, v_{2t}, and v_{2t+1} is strictly greater than $3k - 2i$. We will show that, instead, the number of colors used by Algorithm 1 is at most $3k - 2i$. It is clear that $2k - i \leq 3k - 2i$ since $i \leq k$, so we will assume that the number of colors used is $2k - i + \lceil \frac{(k-(t+1)i)}{t} \rceil$.

$$2k - i + \lceil \frac{(k - (t+1)i)}{t} \rceil \leq 3k - 2i \;\Leftrightarrow\; \lceil \frac{(k - (t+1)i)}{t} \rceil \leq k - i \;\Leftrightarrow$$

$$\frac{(k - (t+1)i)}{t} \leq k - i \;\Leftrightarrow\; 0 \leq (t-1)k + i$$

And this completes the argument.

In the previous example, the $(4, 1)$-coloring of C_5 obtained by Algorithm 1 would be modified as to obtain, for instance, the following $4 : 1$-coloring: $\{1, 2, 3, 4\}$, $\{4, 5, 6, 7\}$, $\{7, 8, 1, 2\}$, $\{2, 3, 4, 5\}$, $\{5, 6, 7, 1\}$.

It may be interesting to characterize in general the graphs G such that $\chi_k^i(G) = \chi_k^{(i)}(G)$, or those graphs G such that $\chi_k^i(H) = \chi_k^{(i)}(H)$ for each induced subgraph H of G.

2.2 Generalization to cacti

These results can be easily generalized for cacti. A graph G is a *cactus* if it does not contain two cycles that share an edge. It is a known fact that every block (maximal 2-connected subgraph) of a cactus is either an edge or a chordless cycle. We will base our proof on the following easy lemma, that holds for many coloring problems.

Lemma 2.6 *Let G be a graph. The (k, i)-chromatic number of G is the maximum of the (k, i)-chromatic numbers of its blocks.*

Proof. Clearly, it is enough to prove it for connected graphs. We proceed by induction on the number of blocks m of G. If G has only one block, the result trivially holds. For the inductive case, suppose the lemma holds for all graphs with fewer than m blocks. Let B be an end-block of G and let v be the cut-vertex of G that belongs to B. Let G' be the subgraph of G induced by $(V(G) \setminus B) \cup \{v\}$. By inductive hypothesis, the (k,i)-chromatic number of G' is the maximum of the (k,i)-chromatic numbers of its blocks.

Let f' be a (k,i)-coloring of G' with the minimum number of colors, and f'' be an optimal (k,i)-coloring of the subgraph of G induced by B. By renaming the colors in f'' in such a way that $f''(v) = f'(v)$, we can combine f' and f'' in order to obtain a (k,i)-coloring of G without adding any new colors. This proves the lemma. □

By Theorem 2.5 and Lemma 2.6, we obtain directly the following result.

Corollary 2.7 *Let G be a cactus. Then, a (k,i)-coloring of G with $\chi_k^i(G)$ colors can be computed in linear time.*

Note that Lemma 2.6 and Corollary 2.7 can be proved analogously for the $k:i$ coloring problem.

3 (k,i)-coloring of cliques

Brigham and Dutton proved the next partial results on the $k:i$-coloring of cliques:

Theorem 3.1 *[4]*
(a) If $n \leq \frac{k}{i} + 1$ then $\chi_k^{(i)}(K_n) = kn - \frac{n(n-1)i}{2}$.
(b) If $n \geq k^2 - k + 2$ then $\chi_k^{(i)}(K_n) = kn - (n-1)i$.

Part (a) of Theorem 3.1 also holds for $\chi_k^i(K_n)$. This was proved by Méndez-Díaz and Zabala in [17]. Part (b), however, does not. For a counterexample, let $n = 4$, $k = 2$ and $i = 1$. We have that $\chi_2^{(1)}(K_4) = 5$, but $\chi_2^1(K_4) = 4$. Indeed, by Theorem 3.1 part (b), we have that $\chi_2^{(1)}(K_4) = 5$ and $\{\{1,2\},\{1,3\},\{1,4\},\{1,5\}\}$ is a proper $2:1$ coloring of K_4. On the other hand, $\{\{1,2\},\{1,3\},\{1,4\},\{3,4\}\}$ is a proper $(2,1)$-coloring of K_4, and thus $\chi_2^1(K_4) \leq 4$. By Proposition 3.5 below, we will obtain that $\chi_2^1(K_4) \geq 4$.

The general problem of (k,i)-coloring cliques is still open, and it is also closely related to one of the central concerns in coding theory. We give now some definitions we need to present this relation. A *binary code* (or just a *code*, for brevity) is a set of binary vectors (or *codewords*) of length j. If a position in a binary vector contains a one, it will be called a *1-position* and a *0-position* otherwise. The *size* of a code is its cardinality. The *Hamming distance* of two codewords a and b is the number of positions in which they differ. The *distance* d_C of a code C is the smallest Hamming distance between any two codewords of C. A (j,d,k)-*constant weight code* is a set of codewords of length j and exactly k ones in each of them, with Hamming distance at least equal to d.
Given j, d and k, the question of determining the largest possible size $A(j,d,k)$ of a (j,d,k)-constant weight code has been studied for almost forty years, and remains one of the most basic questions in coding theory. The general answer is not known, but several upper and lower bounds on $A(j,d,k)$ have been found (see [1, 8] and references therein). We study now the relation between $A(j,d,k)$ and the k,i-coloring of cliques in the following Theorem:

Theorem 3.2 *A (k,i)-coloring for K_n with j colors does exist if and only if $A(j,2(k-i),k) \geq n$.*

Proof. We start with the proof of necessity. Let f be a (k,i)-coloring of K_n with j colors. Construct a set $B = \{b_1, b_2, \ldots, b_n\}$ of n binary vectors, each of length j, such that every vector is the characteristic function of the set of colors associated with each vertex of K_n. That is, for every vertex v_s of K_n we have vector $b_s = (b_s^1, b_s^2, \ldots, b_s^j)$, where $b_s^t = 1$ if and only if color t belongs to $f(v_s)$. We will show that $d_B \geq 2(k-i)$. Let v_x and v_y be any two vertices of K_n, and b_x and b_y their associated binary vectors in B. Since $|f(v_x) \cap f(v_y)| \leq i$, b_x and b_y have at most i 1-positions in common. Vector b_x has k 1-positions in total, so at least $(k-i)$ 1-positions of b_x must be distributed along positions where b_y holds a 0. Analogously, vector b_y must also accommodate at least $(k-i)$ 1's along positions that store a 0 in b_x. This means that they differ in at least $2(k-i)$ positions, so $d(b_x, b_y) \geq 2(k-i)$. Since v_x and v_y are two arbitrary vertices of K_n, we have by definition of distance that $d_B \geq 2(k-i)$, so $A(j,2(k-i),k) \geq n$.
We prove now sufficiency. Suppose $A(j,2(k-i),k) \geq n$. Let B be a code that realizes $A(j,2(k-i),k)$. Choose any n-subset of B. We have now only to interpret each binary vector $b \in B$ as a color set S_b, where a color c belongs to S_b if and only if $b_c = 1$. We obtain n color sets, each of cardinality k. By the same argument as before, no two of them have more than i colors in common, otherwise their corresponding binary

vectors would be at a distance smaller than $2(k-i)$. Assign each set to a vertex of K_n. This is a valid (k,i)-coloring f that uses no more than j colors. □

By Theorem 3.2, we can rephrase the definition of the (k,i)-chromatic number of a complete graph K_n as the minimum positive integer j such that $A(j, 2(k-i), k) \geq n$. This fact is used in the following straightforward corollary.

Corollary 3.3 *If $A(j, 2(k-i), k) \leq n$ and $m > n$, then $\chi_k^i(K_m) > j$.*

Thanks to Corollary 3.3, any upper bound on $A(j, d, k)$ for an even number d, can be used for generating new lower bounds for the $(k, k-\frac{d}{2})$-chromatic number of complete graphs. We will do so with the well known Johnson bound, presented in the next theorem:

Theorem 3.4 *[14] $A(j, 2r, k) \leq \lfloor \frac{rj}{k^2-kj+rj} \rfloor$, if the denominator is positive.*

Let j be an integer such that $\frac{k^2}{i} > j$ (1). By Theorem 3.4 applied to $A(j, 2(k-i), k)$, we have that $A(j, 2(k-i), k) \leq \lfloor \frac{(k-i)j}{k^2-ij} \rfloor$. Note that by our choice of j, the denominator is a positive number. Corollary 3.3 applied on this bound yields $\chi_k^i(K_n) > j$, if $n > \lfloor \frac{(k-i)j}{k^2-ij} \rfloor$ (2). We are interested in the largest possible lower bound on $\chi_k^i(K_n)$, so we will find the maximum value for j that meets the given inequalities (1) and (2). For (2), we may write:

$$\begin{aligned}
n &> \lfloor \frac{(k-i)j}{k^2-ij} \rfloor \\
n &> \frac{(k-i)j}{k^2-ij} \qquad (\text{If } x \in \mathbb{R}, n \in \mathbb{N}, n > x \iff n > \lfloor x \rfloor) \\
nk^2 &> (k-i)j + nij \\
\frac{nk^2}{(n-1)i+k} &> j
\end{aligned}$$

For any real number x and any natural number j, we have $x > j \iff \lceil x \rceil > j$, so the largest possible value for j is $\lceil \frac{nk^2}{(n-1)i+k} \rceil - 1$. We show now that this value of j also meets (1):

$$\begin{aligned} \lceil \frac{nk^2}{(n-1)i+k} \rceil - 1 &\leq \lceil \frac{nk^2}{(n-1)i+i} \rceil - 1 \qquad \text{(Because } k \geq i\text{)} \\ &= \lceil \frac{k^2}{i} \rceil - 1 < \frac{k^2}{i} \end{aligned}$$

The second line holds since for all $x \in \mathbb{R}$, $\lceil x \rceil - x < 1$.
We have thus calculated our maximum possible j. Replacing this value of j in $\chi_k^i(K_n) > j$ gives rise to the following new lower bound on $\chi_k^i(K_n)$:

Proposition 3.5 $\chi_k^i(K_n) > \lceil \frac{nk^2}{(n-1)i+k} \rceil - 1$

We may as well take advantage of results on specific values of $A(j, d, k)$ found in the literature for achieving bounds on $\chi_k^i(K_n)$, for some values of n, k and i. We choose as an example a theorem due to Hanani:

Theorem 3.6 *[9, 10, 11, 12]*
(a) $A(j, 6, 4) = \frac{j(j-1)}{12}$*, if and only if* $j \equiv 1$ *or 4 (mod 12).*
(b) $A(j, 8, 5) = \frac{j(j-1)}{20}$*, if and only if* $j \equiv 1$ *or 5 (mod 20).*

Proposition 3.7 *Let* $j \equiv 1$ *or 4 (mod 12). Then*
(a) $\chi_4^1(K_n) > j$*, if* $n > \frac{j(j-1)}{12}$*.*
(b) $\chi_4^1(K_n) \leq j$*, if* $n \leq \frac{j(j-1)}{12}$*.*

Proof. Part (a) is a direct consequence of Theorem 3.6 (a) and Corollary 3.3. Part (b) follows from Theorem 3.6 (a) and Theorem 3.2. □

Proposition 3.8 *Let* $j \equiv 1$ *or 5 (mod 20). Then*
(a) $\chi_5^1(K_n) > j$*, if* $n > \frac{j(j-1)}{20}$*.*
(b) $\chi_5^1(K_n) \leq j$*, if* $n \leq \frac{j(j-1)}{20}$*.*

Proof. The proof is analogous to Proposition 3.7, using now Part (b) of Theorem 3.6. □

4 Conclusions

We have presented in this work a simple linear time algorithm to compute the (k,i)-chromatic number and an optimum (k,i)-coloring of cycles, and we have generalized the result in order to derive a polynomial time algorithm for this problem on cacti. We have furthermore adapted the algorithm in order to obtain an optimum $k:i$-coloring of cycles and cacti (the $k:i$-chromatic number of cycles was already known [4]).

We also present a relation between this problem on complete graphs and weighted binary codes. However, computing the (k,i)-chromatic number and the $k:i$-chromatic number of complete graphs are both still open problems.

Acknowledgement

We would like to thank the anonymous referee for his/her careful reading and suggestions that helped us to improve the paper.

References

[1] E. Agrell, A. Vardy, and K. Zeger. Upper bounds for constant-weight codes. *IEEE Transactions on Information Theory*, 46(7):2373–2395, 2000.

[2] C. Berge. *Graphs and Hypergraphs*. North–Holland, Amsterdam, 1976.

[3] B. Bollobás and A. Thomason. Set colourings of graphs. *Discrete Mathematics*, 25(1):21–26, 1979.

[4] R.C. Brigham and R.D. Dutton. Generalized k-tuple colorings of cycles and other graphs. *Journal of Combinatorial Theory. Series B*, 32:90–94, 1982.

[5] T. Denley. The odd girth of the generalized Kneser graph. *European Journal of Combinatorics*, 18(6):607–611, 1997.

[6] C. Friedmann, L. Markenzon, and C. Waga. Total coloring of block-cactus graphs. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 78:273–283, 2009.

[7] C.D. Godsil and G. Royle. *Algebraic graph theory.* Graduate Texts in Mathematics. Springer, 2001.

[8] R.L. Graham and N.J.A. Sloane. Lower bounds for constant weight codes. *IEEE Transactions on Information Theory*, 26:37–43, 1980.

[9] H. Hanani. On quadruple systems. *Canadian Journal of Mathematics*, 12:145–157, 1960.

[10] H. Hanani. The existence and construction of balanced incomplete block designs. *Annals of Mathematical Statistics*, 32:361–386, 1961.

[11] H. Hanani. On some tactical configurations. *Canadian Journal of Mathematics*, 15:702–722, 1963.

[12] H. Hanani. A balanced incomplete block design. *Annals of Mathematical Statistics*, 36:711, 1965.

[13] A. Hilton, R. Rado, and S. Scott. A (< 5)-colour theorem for planar graphs. *Bulletin of the London Mathematical Society*, 5:302–306, 1973.

[14] S.M. Johnson. A new upper bound for error-correcting codes. *IEEE Transactions on Information Theory*, 8:203–207, 1962.

[15] N. Khan, and M. Pal. Adjacent vertex distinguishing edge colouring of cactus graphs. *International Journal of Engineering and Innovative Technology*, 4:62–71, 2013.

[16] N. Khan, A. Pal, and M. Pal. Edge colouring of cactus graphs. *Advanced Modeling and Optimization*, 4:407–421, 2009.

[17] I. Méndez-Díaz and P. Zabala. A generalization of the graph coloring problem. *Investigación Operativa*, 8:167–184, 1999.

[18] I. Méndez-Díaz and P. Zabala. Solving a multicoloring problem with overlaps using integer programming. *Discrete Applied Mathematics*, 158:349–354, 2010.

[19] S. Stahl. n-Tuple colorings and associated graphs. *Journal of Combinatorial Theory. Series B*, 20:185–203, 1976.