The Flow Coloring Problem*

Manoel Campêlo† Ricardo Corrêa Cristiana Huiban‡
Carlos Diego Rodrigues

Abstract

Suppose a graph $G = (V, E)$ with one destination node g (the gateway) and a set of source nodes with integer demands defining the input of the problem. Let Φ stand for the set of all possible flows $\phi : E \rightarrow \mathbb{Z}_+$ sending all demands from the sources to the gateway. Then each $\phi \in \Phi$ defines a multigraph $G_\phi = (V, E, \phi)$, where ϕ represents the multiplicity of the edges.

In the *flow coloring problem*, the objective is to find the *flow chromatic index* $\chi'_\Phi(G) = \min_{\phi\in\Phi} \chi'(G_\phi)$, where $\chi'(G_\phi)$ is the chromatic index of G_ϕ. We relate $\chi'_\Phi(G)$ to the *flow fractional chromatic index* $\chi'_{\Phi,f}(G) = \min_{\phi\in\Phi} \chi'_f(G_\phi)$, where $\chi'_f(G_\phi)$ is the fractional chromatic index of G_ϕ. We are interested in proving that the inequality $\chi'_{\Phi,f}(G) \leq \chi'_\Phi(G) \leq \chi'_{\Phi,f}(G) + 1$ is valid, following the classical *Goldberg's Conjecture* for arbitrary multigraphs. When G is 2-connected, we propose a polynomial algorithm to show that the inequality holds for several cases. Moreover, we prove that this algorithm is optimal for 3-connected graphs and gives a $\frac{3}{2}$-approximation for arbitrary 2-connected graphs.

1 Problem introduction

Let $G = (V, E)$ be a graph with a special node $g \in V$, to be called *destination node* or *gateway*. Each other node $v \in V \setminus g$ is associated with an integer demand $b_v \geq 0$ to be sent to g. We will call *source node* a node v with $b_v > 0$. Let Φ stand for the set of all possible integer flows $\phi : E \rightarrow \mathbb{Z}_+$ sending the total demand from the sources to the gateway. Each $\phi \in \Phi$ defines a multigraph $G_\phi = (V, E, \phi)$, where ϕ represents the multiplicity of the edges. In other words, the edge multiset of G_ϕ is defined by each element $e \in E$ replicated $\phi(e)$ times.

The *flow coloring problem* (FCP) in G consists in finding the *flow chromatic index* $\chi'_\Phi(G, b) = \min_{\phi\in\Phi} \chi'(G_\phi)$, where $\chi'(G_\phi)$ is the chromatic index of G_ϕ, i.e. the minimum number of colors assigned to the edges of G_ϕ such that every edge receives at least one color and no two edges with the same color meet at a node. Notice that an edge $e \in E$ with multiplicity $\phi(e) = 0$ does not appear in G_ϕ, so it does not need to be colored. When the vector of demands b has not a particular definition, we simplify the notation by using $\chi'_\Phi(G)$ to denote $\chi'_\Phi(G, b)$.

The edges of G_ϕ receiving the same color induces a matching in G. The number $c(e)$ of matchings covering the edge $e \in E$ is at least the flow $\phi(e)$. Thus, $c(e)$ can be seen as the capacity assigned to e. This observation leads to a restatement of the FCP as a minimum weighting of the matchings of G such that the sum of the (integer) weights of the matchings covering an edge defines its capacity, and these capacities allow a flow sending the total demand from the sources to the gateway. We will say that the weighted matchings cover the flow.

Actually, the term flow coloring can be used in more general contexts involving other combinations of flows (e.g. single or multi-commodity, single or multiple sources and destinations etc) and colorings (edge or node coloring, distance-d coloring - meaning that nodes/edges at distance at most d cannot share a color). Moreover, either the flow or the coloring need not to be integer. Each possible combination leads to a variant of FCP. Particularly, we will relate $\chi'_\Phi(G, b)$ to the *flow fractional chromatic index* $\chi'_{\Phi,f}(G, b) = \min_{\phi\in\Phi} \chi'_f(G_\phi)$, where $\chi'_f(G_\phi)$ is the fractional chromatic index of G_ϕ.

Some scenarios of flow coloring have been studied in the literature under the name of *Round Weighting Problem* - RWP [KMP08]. The coloring usually used in RWP is a kind of *fractional edge*

*This work is partially supported by CNPq Universal and FUNCAP/CNPq PRONEM Projects.
†Partially supported by CNPq, Brazil
‡C. Huiban is funded by FUNCAP/CNPq, Brazil.

coloring of the edge-weighted graph G_ω (we say "kind of" because the used "*matchings*" may not be the classical ones; they call them *rounds*, see in [KMP08]). The edge-weights of G_ω are defined by a *fractional* flow (multi-commodity or single-flow). So, *real-valued* weights are assigned to the "*matchings*" with the objective of covering G_ω. The RWP with classical matchings is treated in [Gom09, Rey09, BGR09].

Now, we adopt the term *flow coloring* so as to make the relation between the problem and the classical flow and coloring problems more evident. In particular, we want to define and study flow coloring parameters that come out as counterparts of classical coloring parameters.

In this work, we deal with the specific case of single flow to *one* destination node and *integer*[1] edge-coloring. We will also assume that G is 2-connected. We are interested in proving that the inequality $\chi'_{\Phi,f}(G) \leq \chi'_\Phi(G) \leq \chi'_{\Phi,f}(G)+1$ is valid, following the classical *Goldberg's Conjecture* that states a similar inequality for an arbitrary multigraph [Gol73]. In our case we are dealing with a particular multigraph G_ϕ, defined by the optimal flow ϕ. We list several cases satisfying the inequality and show the exact value of $\chi'_\Phi(G)$ for some of them (including the case where G is a 3-connected graph). For these last cases, we also give a polynomial-time algorithm to find, besides $\chi'_\Phi(G)$, the optimal flow and coloring.

2 Preliminaries

In the edge coloring multigraph literature, we can find many results that can be used in the context of FCP. We summarize the most important ones for our case. Let $\mathfrak{G} = (\mathfrak{V}, \mathfrak{E})$ be a multigraph. A b-fold edge-coloring of $\mathfrak{G}$ assigns a set of at least b colors to each edge of $\mathfrak{G}$ so that any two edges sharing an endpoint receive disjoint sets of colors. The b-fold coloring index, denoted by $\chi'_b(\mathfrak{G})$, is the least a such that $\mathfrak{G}$ admits a b-fold coloring using a colors in total. The chromatic index and the fractional chromatic index of $\mathfrak{G}$ are $\chi'(\mathfrak{G}) = \chi'_1(\mathfrak{G})$ and $\chi'_f(\mathfrak{G}) = \inf_b \frac{\chi'_b(\mathfrak{G})}{b}$, respectively. While finding $\chi'(\mathfrak{G})$ is an NP-Hard problem, it is is polynomial to determine $\chi'_f(\mathfrak{G})$, which is clearly a lower bound for $\chi'(\mathfrak{G})$ [PW84]. There is $\frac{4}{3}$-approximation algorithm for $\chi'(\mathfrak{G})$ [HNS86], and it is NP-hard to obtain any better constant factor approximation [Hol81].

Let $\Delta = \max_{v\in V}\delta(v)$, where $\delta(v)$ is the degree of the node v in $\mathfrak{G}$, i.e. the number of edges (counting multiplicity) that are incident to v in $\mathfrak{G}$. Let $\Gamma = \max_{H\subseteq\mathfrak{V},|H|=2k+1,k\geq 1} \frac{|\mathfrak{E}(H)|}{k}$ be the *odd density* of $\mathfrak{G}$, where $\mathfrak{E}(H)$ is the subset of edges of $\mathfrak{G}$ with both endpoints in H. It is known that $\chi'_f(\mathfrak{G}) = \max(\Delta, \Gamma)$ and that $\max\{\Delta, \lceil\Gamma\rceil\} \leq \chi'(\mathfrak{G}) \leq \min\{\frac{3}{2}\Delta, \Delta+\mu\}$, where μ is the maximum edge multiplicity of $\mathfrak{G}$ [Sha49, Viz64]. The classical *Goldberg's Conjecture* claims that $\chi'(\mathfrak{G}) \leq \max\{\Delta+1, \lceil\Gamma\rceil\}$[Gol73],[Sey79]. We can restate the conjecture as $\chi'_f(\mathfrak{G}) \leq \chi'(\mathfrak{G}) \leq \chi'_f(\mathfrak{G})+1$.

In the *flow coloring problem*, we are edge-coloring a special kind of multigraph that has the multiplicity of the edges defined by a flow, G_ϕ. Notice that the flow graph G^*_ϕ giving the optimal solution $\chi'_\Phi(G)$ may be different from the flow graph giving the optimal fractional solution $\chi'_{\Phi,f}(G)$. Therefore, even if *Goldberg's Conjecture* is true for the edge-coloring, it is not clear that $\chi'_{\Phi,f}(G) \leq \chi'_\Phi(G) \leq \chi'_{\Phi,f}(G)+1$ is true for the flow coloring problem. If this inequality holds, finding $\chi'_\Phi(G)$ becomes a decision problem.

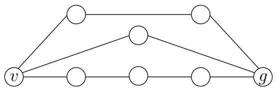
Now focusing on the existing results for flow coloring, we use the lower bound defined by the unavoidable number of edges that are incident to g in any feasible flow [Gom09, BGR09]. Indeed, in any $\phi\in\Phi$, the number of edges incident to g is $B = \sum_{v\in V} b_v$. This gives a lower bound for the maximum degree Δ of G_ϕ and consequently:

Fact 1 $B \leq \chi'_{\Phi,f}(G) \leq \chi'_\Phi(G)$.

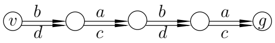
To compute an initial upper bound, we apply the strategy of iteratively sending flow through *a couple of node-disjoint paths*[2] from one or two sources to g. (Recall that given a graph and pairs $(s_1,t_1),...,(s_k,t_k)$, for fixed k, the problem of deciding if exist k mutually vertex-disjoint paths is tractable [RS95]). Such node-disjoint paths always exists in 2-connected graphs [Men27]. This

[1] Notice that assuming *integer* edge-coloring corresponds to defining integer capacities $c(e)$ for the edges of G. As we are considering single-flow, we can restrict ourselves to integer flow by the *Integrality Theorem* (there is an optimum flow ϕ whose values are all integers, $\phi : E \to \mathbb{Z}_+$). It was proved by [HK65] and follows from the observation that the constraints matrix respects the property of *total unimodularity*.

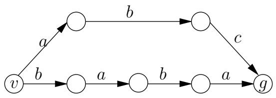
[2] For the sake of simplicity, we say node-disjoint paths (or simply disjoint) to mean node-disjoint except for the endnodes.



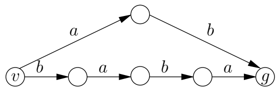
(a) Graph G.



(b) G_{ϕ_1}: Sending $b_v = 2$ with 1 path. We need 4 colors a, b, c and d, that is $2\ c/\phi$.



(c) G_{ϕ_2}: Sending $b_v = 2$ with 2 paths in an odd cycle. We need 3 colors a, b and c, that is $\frac{3}{2}\ c/\phi$.



(d) G_{ϕ_3}: Sending $b_v = 2$ with 2 paths in an even cycle. We need 2 colors a and b, that is $1\ c/\phi$.

Figure 1: Sending $b_v = 2$ from v to g. Flow ϕ_3 gives the optimal solution $\chi'_\Phi(G) = \chi'(G_{\phi_3}) = 2$.

strategy was used in [BGR09] to show that $\chi'_\Phi(G) = B$ if B is even and G is 3-connected. In this paper, we prove it remains B also for the case with B odd. In [Rey09], it is shown the result is B for 3-connected graphs in the case with *fractional* edge coloring and *fractional* simple-flow with *one* destination node, and it is obtained a $\frac{6}{5}$-approximation for general graphs in this same case.

The same strategy of upper bound computation was also used in [Gom09, BGR09] to approximate the distance-d[3] fractional/integer edge coloring problem. They present a $\frac{d+1}{\lceil \frac{d+1}{2} \rceil}$-approximation algorithm for any d. In [Gom09], it is also shown that the FCP with fractional flow and fractional edge coloring can be solved in polynomial time for $d = 1$.

In this work, we extend the results cited here from [Gom09, BGR09]. Section 3 explain the strategy for obtaining an upper bound for FCP. This strategy is the basis of Algorithm 2-by-2 proposed in Subsection 3.1. We show the quality of the obtained upper bound with respect to the basic lower bound given by Fact 1. Section 4 proposes an extension of the strategy proposed by Algorithm 2-by-2 showing it is not the best possible way to solve the problem. With the extended strategy, we obtain some additional cases that can be solved satisfying the inequality $\chi'_{\Phi,f}(G) \leq \chi'_\Phi(G) \leq \chi'_{\Phi,f}(G) + 1$. For instance, we get the optimal solution for FCP in a 3-connected graph. Finally, Section 5 presents a $\frac{3}{2}$-approximation for the problem improving the factor of 2 presented in [Gom09]. Section 6 sums up our results.

3 Coloring a flow

Fact 1 establishes a minimum of B colors to sent B units of flow to the gateway, i.e. a minimum ratio of one color per unit of flow ($1\ c/\phi$, for short). The strategy for upper bound computation is to iteratively send flow to the gateway at the ratio of $1\ c/\phi$ as much as possible.

In order to illustrate possible strategies for iteratively send flow to the gateway, let us consider the graph G of Figure 1(a) with a unique source node v with demand $b_v = 2$. Figure 1(b) presents a possible flow in G that uses only one path per iteration. We can use two iterations to send $b_v = 2$. At each iteration, a flow of 1 is sent through a path colored alternately with 2 colors: $\{a, b\}$ (for the $1st$ unit of flow) and $\{c, d\}$ (for the $2nd$ unit of flow). This strategy needs 4 colors to send a demand of 2, that is $2\ c/\phi$. Figure 1(c) shows another possible flow using two disjoint paths (from an odd cycle). It needs 3 colors to send $b_v = 2$ in one iteration (a flow of 1 is sent in each path), thus achieving a ratio of $1.5\ c/\phi$. The flow giving the optimal solution is presented in Figure 1(d): $b_v = 2$ is sent also using two disjoint paths, but now from an even cycle. Since it can be colored alternately with 2 colors, it makes $1\ c/\phi$.

Let us consider other scenarios in graph G. If $b_v = 4$, we can apply two iterations depicted in Figure 1(d) to keep the ratio of $1\ c/\phi$. If $b_v = 5$, we can send 4 units of flows as before, but it will remain one unit of flow in v. If there were another source u with $b_u = 1$, we could use any two node-disjoint paths P_{ug} (from u to g) and P_{vg} (from v to g) to send the remaining demand of v and the demand of u. Since these two paths can be colored with 2 colors, we again could keep the

[3] In a distance-d edge coloring, we cover the graph with distance-d matchings (set of edges mutually at distance d, also called *δ-separated matchings* in [SV82]).

minimum ratio of 1 c/ϕ.

The situations presented above justify our basic algorithm to generate an upper bound. At each iteration, we use a couple of node-disjoint paths to send a same amount of flow in each path at the ratio of 1 c/ϕ. These paths may link two different sources to g or define an even cycle containing a source and g. Since we are assuming that G is 2-connected, there always exist node-disjoint paths, one from u to g and one from v to g, for any two (maybe equal) sources u and v. Eventually, this algorithm may stop without sending the total demand B at the minimum ratio of 1 c/ϕ.

To describe more precisely the algorithm, we present two categories of node:

- *Autonomous*: represented by set $\mathcal{A}$. A source v belongs to $\mathcal{A}$ if there exist an *even* cycle containing v and g (that is, node v can use two disjoint paths of the same parity to send its demand to the gateway) or v is in the neighborhood $N(g)$ of g (that is, node v can send its demand directly to g).

- *Dependent*: represented by set $\mathcal{D}$, which is composed by all sources that are not in $\mathcal{A}$.

Let $\mathcal{S} = \mathcal{A} \cup \mathcal{D}$ be the set of all sources.

In [iKLR10], they show that two disjoint paths of the same parity between two nodes (making an even cycle) can be found in polynomial time. The problem is called *2 parity disjoint paths problem* [iKLR10]. Then, we can classify in polynomial time a source node of G into autonomous or dependent.

3.1 Algorithm 2-by-2

Algorithm 2-by-2 defines a sequence of pairs of sources to send flow together using two disjoint paths; we say it makes *combinations*. The existence of a pair of node-disjoint paths from two different nodes to any other node is always possible in 2-connected graphs [Men27]. As two node-disjoint paths from different sources to g do not close a cycle, they can be colored alternately with 2 colors. So, combining two different nodes of any kind (even both dependent) makes it possible to send a flow of 2 (a unit of flow per path) with 2 colors. Similarly, an autonomous node can send 2 units of flow with 2 colors. In both cases, we can keep the rate of 1 c/ϕ.

To describe Algorithm 2-by-2, denote by b'_v be the current demand in node v during the execution of the iterative process. Of course, $b'_v = b_v$ initially. Also, let v_k be such that $b_{v_k} = \max\{b_v : v \in \mathcal{S}\}$. The algorithms acts differently depending on $b_{v_k} \geq \sum_{v\in\mathcal{S}\setminus v_k} b_v$ (**Case** I) or $b_{v_k} < \sum_{v\in\mathcal{S}\setminus v_k} b_v$ (**Case** II).

Algorithm 1 Algorithm 2-by-2

if $b_{v_k} \geq \sum_{v\in\mathcal{S}\setminus v_k} b_v$ **then** {**Case** I}
  for all $u \in \mathcal{S} \setminus v_k$ **do**
    Send to g a flow of b_u from each node of the pair (v_k, u);
  end for
  Send from v_k to g a flow of $2\lfloor b'_{v_k}/2 \rfloor$, if $v_k \in \mathcal{A}$;
else {**Case** II}
  for all $u, v \in \mathcal{S}$, $u \neq v$, $b'_v, b'_u \geq 1$ **do**
    Send to g a flow of $\min\{b'_v, b'_u\}$ from each node of the pair (v, u);
  end for
end if

Let ρ be the remaining total demand after the execution of Algorithm 2-by-2. Since the minimum rate of 1 c/ϕ is kept in each iteration, an amount of $B - \rho$ units of flow is sent to g using $B - \rho$ colors. In **Case** II, the value of ρ may be dependent on the sequence of combinations. The next result shows that we can manage to have $\rho \in \{0, 1\}$ in this case.

Lemma 1 *If $b_{v_k} < \sum_{v\in\mathcal{S}\setminus v_k} b_v$ there is separator node q with the following properties:*

1. *$b_q = b_q^1 + b_q^2$, for some integers $b_q^1, b_q^2 \geq 0$;*
2. *$\mathcal{S} \setminus q = \mathcal{S}' \cup \mathcal{S}''$, with $\mathcal{S}' \cap \mathcal{S}'' = \emptyset$;*

3. $\left|\left(b_q^1 + \sum_{v\in\mathcal{S}'} b_v\right) - \left(b_q^2 + \sum_{v\in\mathcal{S}''} b_v\right)\right| \le 1$

4. $b_q^1 \le \sum_{v\in\mathcal{S}''} b_v$ *and* $b_q^2 \le \sum_{v\in\mathcal{S}'} b_v$*;*

Proof: Notice that $B = \sum_{v\in\mathcal{S}} b_v$ and $b_{v_k} < B/2$. Then, there exist a subset $\mathcal{S}' \subset \mathcal{S}$ such that $v_k \in \mathcal{S}'$, $\sum_{v\in\mathcal{S}'} b_v \le \lfloor B/2 \rfloor$ and $\sum_{v\in\mathcal{S}'} b_v + b_q > \lfloor B/2 \rfloor$, for any $q \in \mathcal{S}\setminus\mathcal{S}'$. Take $\mathcal{S}'' = \mathcal{S}\setminus(\mathcal{S}'\cup q)$, $b_q^1 = \lfloor B/2\rfloor - \sum_{v\in\mathcal{S}'} b_v$ and $b_q^2 = b_q - b_q^1$. We have that $b_q^1 + \sum_{v\in\mathcal{S}'} b_v = \lfloor B/2 \rfloor$ and $b_q^2 + \sum_{v\in\mathcal{S}''} b_v = \lceil B/2 \rceil$. Then, we clearly get items 1–3.

Since $v_k \in \mathcal{S}'$ and $b_{v_k} = \max\{b_v : v \in \mathcal{S}\}$, it follows that $\sum_{v\in\mathcal{S}'} b_v \ge b_{v_k} \ge b_q \ge b_q^2$ and $\sum_{v\in\mathcal{S}''} b_v = \lceil B/2\rceil - b_q^2 \ge \lfloor B/2 \rfloor - b_q^2 = b_q^1 + \sum_{v\in\mathcal{S}'} b_v - b_q^2 \ge b_q^1$. This shows item 4. ■

Motivated by Lemma 1, we chose the following sequence of combinations in loop 7–9:

(i) first, take $u = q$ and $v \in \mathcal{S}''$ until b_q^1 is vanished;

(ii) then, take $u = q$ and $v \in \mathcal{S}'$ until b_q^2 is vanished;

(iii) finally, take $u \in \mathcal{S}'$ and $v \in \mathcal{S}''$.

Notice that Lemma 1 (items 1 and 4) guarantees the accomplishment of steps (i)–(ii). Lemma 1(1–3) also implies that the residual demand after step (iii) is at most one. This refinement of Algorithm 2-by-2 leads to the following result.

Lemma 2 *Algorithm* 2-by-2 *stops with the following residual demand* ρ*:*

If ***Case*** *I and* $v_k \in \mathcal{D}$*:* $\rho = b_{v_k} - \sum_{v\neq v_k} b_v \ge 0$*;* ρ *and* B *have the same parity;*

If ***Case*** *II or* $v_k \in \mathcal{A}$*:* $\rho = 0$ *if* B *is even, or* $\rho = 1$ *if* B *is odd;*

Proof: Since each iteration of Algorithm 2-by-2 always send an even amount of flow, ρ and B have the same parity. It remains to show the possible values of ρ in each case. We start with **Case** *I*. As $b_{v_k} \ge \sum_{v\in\mathcal{S}\setminus v_k} b_v$, the demand b'_{v_k} will be always greater than the demand of the other nodes. So, Algorithm 2-by-2 combines b_{v_k} with *all* other demands. The remaining demand of v_k after loop 2–4 is $b_{v_k} - \sum_{v\neq v_k} b_v \ge 0$. If node v_k is dependent, Step 5 is not executed and this is the final residual demand. Otherwise, the remaining demand can be sent through an even cycle or directly through one edge, leading to $\rho \in \{0,1\}$.

In **Case** *II*, $b_{v_k} < \sum_{v\in\mathcal{S}\setminus v_k} b_v$. By Item 3 of Lemma 1, Algorithm 2-by-2 has the total demand patitioned into two parts: $\left(b_q^1 + \sum_{v\in\mathcal{S}'} b_v\right)$ and $\left(b_q^2 + \sum_{v\in\mathcal{S}''} b_v\right)$. We have only to guarantee that node q is not combined with itself (as it may be dependent). Item 3 of Lemma 1 guarantees that the whole demand of node q can be combined with the demand of other nodes. Then, Algorithm 2-by-2 can combine the remaining demand of the two partitions. So, $\rho \in \{0,1\}$ as the difference between the two partitions (Item 3) is at most 1. ■

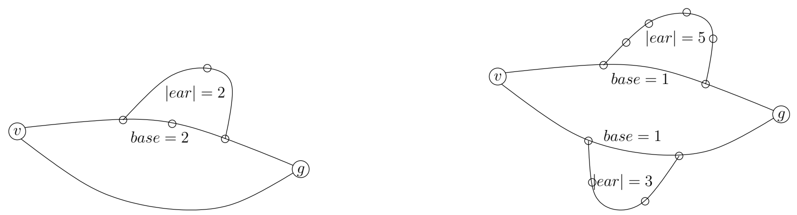
A unitary demand at a source v can only be sent to g at the rate of 1 c/ϕ if combined with other demand or if v is a neighbor of g. Whenever the second alternative of Lemma 2 holds, we can slightly modify Algorithm 2-by-2 to end with the residual demand at a source in $N(g)$, if any.

3.2 Dealing with a $\rho > 1$

We saw in Lemma 2 that ρ can be greater than 1 in **Case** *I* and $v_k \in \mathcal{D}$. In this section, we prove that if node v_k participates in at least one odd cycle with g satisfying an *EarCondition* (defined below) it can send a flow of ρ to g with $\rho + 1$ colors in a combination with itself. Let an *ear of a cycle* be a path p disjoint of the cycle between two nodes x,y of the cycle. The *size* of an ear is the number of edges in p. The *base* of an ear is represented by a path from y to z in the cycle.

Dependent node with ears ($\mathcal{D}'$): Set of nodes $v \in \mathcal{D}$ satisfying one of the following conditions.
-**EarCondition 1**: the node has an odd cycle to g with one ear of size $|ear| \ge 2$ and $|base| \ge 2$. The base of the ear can contain v or g only as an endpoint (not in the interior).
-**EarCondition 2**: the node has an odd cycle C to g with two ears of size $|ear| \ge 3$ and $|base| = 1$. Assume C as the cycle using the bases of the ears. Both ears can share edges if satisfying the following conditions:

* $\exists$ an edge $e_1 \in ear1$ such that the distance (in number of edges) $d(e_1, e) \ge 1, \forall e \in C$; and $d(e_1, e_2) \ge 1, \forall e_2 \in ear2$; and



(a) EarCondition 1. (b) EarCondition 2.

Figure 2: When a node $v \in \mathcal{D}$ can send a flow of ρ with $\rho + 1$ colors in a combination with itself.

* $\exists$ an edge $e_2 \in ear2$ such that the distance (in number of edges) $d(e_2, e) \geq 1, \forall e \in C$; and $d(e_2, e_1) \geq 1, \forall e_1 \in ear1$.

Notice that, the base of the ear has to have the same parity of the size of the ear, otherwise the node would be an autonomous node (it would be an even cycle to g). Figure 2 presents some examples.

Lemma 3 *Let G be a 2-connected graph and ρ be the residual demand in node v returned by Algorithm 2-by-2. If $v \in D'$ then ρ can be sent to g using $\rho + 1$ colors.*

Proof: Node v has (at least) two variants of the same odd cycle to send its remaining flow ρ: C_1 and C_2. Let us say that cycle C_1 uses the $ear1$ and cycle C_2 does not use the $ear1$. Cycle C_2 may have same edges of C_1 except that C_2 uses the base instead of the own $ear1$. Suppose node v is in an odd cycle satisfying **EarCondition 1**. So, node v can send the first time using cycle C_2 with three colors $b, a, b, a, b, ..., c$ starting in the first edge of the base of the $ear1$. Second iteration, v uses C_1 with the colors $a, d, e, d, e, ...$ that is reusing color a of the previous cycle. This reused color can always be used on the first edge of the $ear1$ (the adjacent colors are c and b). Next iteration, v uses C_2 with the colors $d, f, g, f, g, ...$ reusing color d of the previous cycle, and so on. This reused color can always be used on the first edge of the base of the $ear1$ in C_2 (the adjacent colors are a, b, c, e).

Thus, if ρ is even we are using $\rho+1$ colors (the $+1$ comes from color c used at the first iteration). Otherwise it remains one unit of flow in v_k but we can use the side of C_2 that does not use color c to alternate color c with one more color sending this last unit with a total of also $\rho + 1$ colors.

Suppose node v in an odd cycle satisfying **EarCondition 2**. Let C_2 be the cycle using $ear2$. We repeat the same algorithm explained before, now putting the reused colors only on the ears. ∎

If node $v_k \notin \mathcal{A} \cup \mathcal{D}'$, G is a graph which is formed by a cycle C that has the following.

\- Only one ear of size ≥ 3 and base of size 1. (It can be seen as one *chord* assuming that C uses the ear, not the base); or

\- Two or more ears with base of size 1 sharing edges such that all edges of $ear1$ are at distance ≤ 1 of C or $ear2$, and vice-versa.

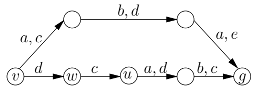
Besides that, C may have ears with base containing g or v_k as an interior node. There are other cases implying multi-edges in G, these cases are not a matter for FCP.

4 Flow coloring extension

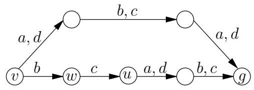
Algorithm 2-by-2 defines *simple* combinations (v, u) using any pair of node-disjoint paths P_{vg} and P_{ug} (between v and g, and between u and g, respectively). Here, we use extended combinations (v, u, w) that uses an additional node-disjoint path with node w. The objective is using a mix of simple and extended combinations to send a flow greater than $B - \rho$ keeping a rate of 1 c/ϕ. Two types of extended combinations are described in the next subsections.

4.1 Edge-extended combination

In this subsection, the extended combination (v, u, w) uses three node-disjoint paths P_{vg},P_{ug},P_{vw} and, the additional path P_{vw} is exactly an edge (see Figure 4(a) and 4(b)). We use edge-extended combinations to reduce the residual demand ρ, when it is at least 2. By Lemma 2, a $\rho \geq 2$ happens

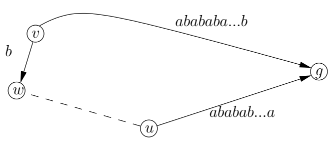


(a) Solution given by Algorithm 2-by-2 and Theorem 1(5 colors).

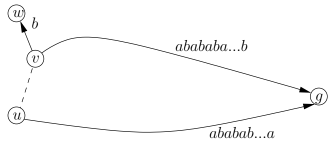


(b) Optimal solution obtained by extending the combinations of Algorithm 2-by-2 (4 colors)

Figure 3: Graph G with $b_v = 3$ and $b_u = 1$, $\chi'_\Phi(G) = 4$.



(a) $|N(v)| = 2$ and $w \in X$.



(b) $|N(v)| \geq 3$ and $w \in X$.

Figure 4: Extending the combination to edge P_{vw}.

if, and only if, $b_{v_k} \geq \sum_{u \in \mathcal{S} \setminus v_k} b_u + 2$ and $v_k \in \mathcal{D} \setminus \mathcal{D}'$. Let ρ' be the remaining from ρ after using edge-extended combinations.

Let us consider the example of Figure 3(a) with $b_{v_k} = 3$ and $b_u = 1$. Algorithm 2-by-2 will use two colors, say ab, to send 2 units of flow (1 unit from v_k to g and 1 unit from u to g). The residual demand is $\rho = 2$ at v_k. We saw how to send this residual demand with 3 colors, leading to a $\rho = 0$ with 5 colors in total. However, we can do better.

An optimal strategy using 4 colors is depicted in Figure 3(b). Instead of coloring only two paths in the first iteration, we color three paths with the same colors a, b. The extra path links v_k to a neighbor w. We say we are *extending* the coloring given by Algorithm 2-by-2. This way we can transfer demand from v_k to another node w. In a second iteration, we can combine v_k and w, which corresponds to combine v_k with itself, but now at the minimal rate $1c/\phi$. In Figure 3(b), this corresponds to color the edge $v_k w$ with the used color b and to transfer 1 unit of flow from v_k to w. After that, we can send 1 unit of v_k and 1 unit of w together, through two disjoint paths, with two adtional colors c, d. In total, we use 4 colors to obtain a $\rho' = 0$.

We will use the edge-extended combinations in a operation as described below.

Definition 1 ***Extension to an edge:*** *We use an edge P_{vw} to send a flow of $f' \leq b_u$ reusing one of the $2f'$ colors already used in the paths P_{vg} and P_{ug} of a combination (v, u). After that, we define a new combination (v, w) with two disjoint paths P_{wg} and P_{vg}, using $2f'$ new colors, to send a flow of $2f'$ (that is, f' per path) to g using then 1 c/ϕ.*

A combination (v, u) can be extended to (v, u, w), $w \in N(v)$, if it admits an *extension to an edge*. Lemma 4 characterizes the node $u \in \mathcal{S} \setminus v_k$ that can participate with v_k in an edge-extended combination. Let X be the set of source nodes u that can participate with v_k in an edge-extended combination (v_k, u, w). Recall that when $v_k \in \mathcal{D}$, Algorithm 2-by-2 combines v_k with each other source and, a flow of b_u is sent in each combination (v_k, u). Then, we should have $\lfloor \frac{\rho}{2} \rfloor \leq \sum_{u \in X} b_u$ to be abble to send flow from ρ leaving a remaining of $\rho' \leq 1$. That is, extending $|X|$ simple combinations to edges incident to v_k.

Lemma 4 *Let $u \in \mathcal{S} \setminus v_k$. The following assertions are equivalent:*

1. *$u \in X$, i.e. there exist a pair of node-disjoint paths P_{ug} and $P_{v_k g}$ (between u and g, and between v_k and g, respectively), and a node $y \in N(v_k)$ not belonging to P_{ug} neither $P_{v_k g}$;*

2. *there is a subset $C \subseteq N(v_k)$ with $|C| \geq 2$ such that $C \cup v_k$ does not separate u and g, i.e. there is still a path between u and g in $G \setminus (C \cup v_k)$.*

Proof: Assume item 1. Since v_k is not a neighbor of g (as it is a dependent node), there is a $w \in N(v_k)$, $w \notin \{g, y\}$, in the path $P_{v_k g}$. Then, $C = \{y, w\}$ satisfies item 2, provided that P_{ug} is a path in $G \setminus (C \cup v_k)$.

Now, assume item 2. Let $C = \{y, w\}$ and P'_{ug} be a path in $G \setminus (C \cup v_k)$. Since G is 2-connected, there are two disjoint paths P_{yg} and P_{wg}. We consider two cases. First, suppose that v_k does not belong to P_{yg} nor P_{wg}. Notice that both P_{yg} and P_{wg} intercepts P'_{ug} at least at g. W.l.o.g, assume that P'_{ug} intersects P_{yg} before than it intersects P_{wg}, and let z be this node intersection. Let $P_{v_k g}$ be composed by the edge $v_k w$ and P_{wg}, and P_{ug} be composed by the subpath P_{uz} of P'_{ug} and the subpath P_{zg} of P_{yg}. The two formed paths are node-disjoint. They together with y satisfy item 1.

The complementary case occurs when $v_k \in P_{wg}$. We can then assume that the edge wv_k is in P_{wg}. Otherwise, we can shorten P_{wg} to include this edge. If P'_{ug} intersects P_{yg} before than it intersects P_{wg} (at a node z), we can define P_{ug} as before and take $P_{v_k g}$ as a subpath of P_{wg}. These two paths with w satisfy item 1. Otherwise, P'_{ug} intersects P_{wg} at z (after v_k) and the subpath P'_{uz} has no intersection with P_{yg}. In this case, form P_{ug} by P'_{uz} together with the subpath P_{zg} of P_{wg}, and $P_{v_k g}$ is composed by the edge $v_k y$ and P_{yg}. Again item 1 holds for these two paths and w. ■

Eventhough finding the the sources in X can be done in polynomial time, this task may become easier in the following case.

Lemma 5 *Assume that G is 2-connected. If $|N(v_k)| \geq 3$, every source outside $N(v_k) \cup v_k$ is in X.*

Proof: let $u \in \mathcal{S} \setminus (N(v_k) \cup v_k)$. Let P_1 and P_2 be two disjoint paths between u and g. Remember that $v_k \notin N(v) \cup N(g)$. If v_k participates in one of these paths, then it must also have two neighbors in this path. If v_k is not in P_1 nor P_2, we can take two neighbors of v_k that do not belong to one of these paths. This is always possible because $|N(v_k)| \geq 3$. In any cases, the chosen two neighbors satisfy Lemma 4(2). ■

Now, we show a limit on the amount of b_{v_k} (in function of the other demands) such that ρ can be reduced to a $\rho' \in \{0, 1\}$ using the *extensions to edges*. We have the following:

Lemma 6 *Assume that G is 2-connected. Let $\bar{X} = \{v \in \mathcal{S} : v \notin X\}$. If $b_{v_k} \leq 3\sum_{v \in \mathcal{S} \setminus v_k} b_v - 2\sum_{v \in \bar{X} \setminus v_k} b_v$. Then, if ρ is even, $\rho' = 0$. Otherwise, $\rho' = 1$. Consequently, $B \leq \chi'_\Phi(G) \leq B + 1$.*

Proof: We have a flow of $B' = \sum_{u \in X} b_u$ that is sent using simple combinations of Algorithm 2-by-2. We rewrite as $B' = B - \sum_{v \in \bar{X} \setminus v_k} b_v - b_{v_k}$. Now, we need to send just $\lfloor \frac{\rho}{2} \rfloor$ to neighbors of v_k using edge-extensions of the combinations used to send B'. Then, we can combine both halves of ρ as explained in Definition 1.

From $b_{v_k} \leq 3\sum_{v \in \mathcal{S} \setminus v_k} b_v - 2\sum_{v \in \bar{X} \setminus v_k} b_v$, we can derive that $b_{v_k} - \sum_{v \in \mathcal{S} \setminus v_k} b_v \leq 2(\sum_{v \in \mathcal{S} \setminus v_k} b_v + b_{v_k}) - 2(\sum_{v \in \bar{X} \setminus v_k} b_v) - 2b_{v_k}$. Thus, $b_{v_k} - \sum_{v \in \mathcal{S} \setminus v_k} b_v \leq 2B - 2(\sum_{v \in \bar{X} \setminus v_k} b_v) - 2b_{v_k}$. It implies that $\lfloor \frac{\rho}{2} \rfloor \leq B'$, then we have enough extended combinations to reach a $\rho' \leq 1$. ■

Using Lemma 5, we may reduce the residual demand ρ described in Case I of Lemma 2 improving the resulting upper bound.

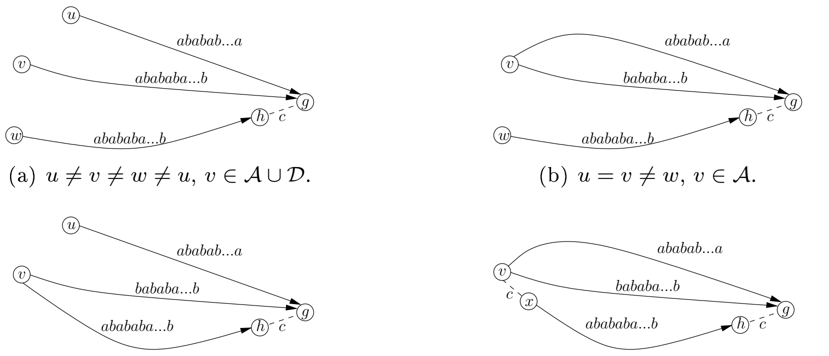
Corollary 1 *Assume that G is 2-connected and $|N(v_k)| \geq 3$. If $b_{v_k} \leq 3\sum_{v \in \mathcal{S} \setminus v_k} b_v - 2\sum_{v \in N(v_k)} b_v$. Then, if ρ is even, $\rho' = 0$. Otherwise, $\rho' = 1$. Consequently, $B \leq \chi'_\Phi(G) \leq B + 1$.*

Dealing with $|N(v_k)| = 2$

By Lemma 5, when $|N(v_k)| = 3$ then $\bar{X} \subseteq \{N(v_k) \cup v_k\}$. When $|N(v_k)| = 2$, it may exist some source nodes in $\bar{X} \setminus \{N(v_k) \cup v_k\}$. Item 2 of Lemma 4 says that, a node $u \in \bar{X}$ does not have a path to g in $G \setminus (C \cup v_k)$ for *any* subset $C \subseteq N(v_k)$ with $|C| \geq 2$. We give a strategy to allow the combination of v_k with these nodes that are in $\bar{X} \setminus \{N(v_k) \cup v_k\}$.

Lemma 7 *Assume that G is 2-connected and $|N(v_k)| = 2$. If $b_{v_k} \leq 3\sum_{v \in \mathcal{S} \setminus v_k} b_v - 2\sum_{v \in \{\bar{X} \cap N(v_k)\}} b_v$ then ρ can be sent to g using $\rho + 1$ colors.*

Proof: If $\bar{X} \setminus v_k = \emptyset$, Lemma 6 solves it. Then, assume there are some nodes in $\bar{X} \setminus v_k$. Suppose that all nodes $u \notin \bar{X} \setminus v_k$ had their flow already combined with v_k by Algorithm 2-by-2 and, there are only nodes in $\bar{X}$ with a remaining flow. Let $N(v_k) = \{y, w\}$, if node $u \in \bar{X} \setminus v_k$ then all paths P_{ug} use a node in $\{y, w\}$ (if P_{ug} uses v_k then it uses before a node in $\{y, w\}$ as v_k has degree 2). So u is in an ear with base yv_kw of size 2 and, the ear has size ≥ 2 as it contains u. So, it does not satisfy **EarCondition 1** only by the fact that node v_k is in the interior of the base of the ear. For



(a) $u \neq v \neq w \neq u$, $v \in \mathcal{A} \cup \mathcal{D}$. (b) $u = v \neq w$, $v \in \mathcal{A}$.

(c) $u \neq v = w$, $v \in \mathcal{A} \cup \mathcal{D}$. (d) $u = v = w$, $v \in \mathcal{A}$.

Figure 5: Extension to path P_{wh} using colors ab.

this case, use an odd cycle to send 2 units from v_k using the colors $ababab, ..., c$, after that reuse one color to send 2 units with an odd cycle from a node $u \in \bar{X} \setminus \{N(v_k) \cup v_k\}$ and so on. It is similar to the strategy used by Lemma 3 for the case satisfying **EarCondition 1**, the only difference is that here C_1 is used by u and C_2 is used by v_k. ■

Corollary 2 *Assume that G is 2-connected. If $b_{v_k} \leq 3\sum_{v \in \mathcal{S}\setminus v_k} b_v - 2\sum_{v \in N(v_k)} b_v$ then $B \leq \chi'_\Phi(G) \leq B + 1$.*

4.2 Path-extended combination

In this subsection, the extended combination (v, u, w) uses three node-disjoint paths P_{vg}, P_{ug}, P_{wg} and, the additional path P_{wg} is a path from a source node w to g. A path-extended combination has four configurations of the *three* disjoint paths as illustrated in Figures 5(a), 5(b), 5(c) and 5(d). Let f be the amount of flow sent in a simple combination of Algorithm 2-by-2. We will use the path-extended combinations in a operation as described below.

Definition 2 ***Extension to a path:*** *We use a path P_{wh} to send a flow of $f' \leq f$ reusing $2f'$ colors already used in the paths P_{vg} and P_{ug} of a combination (v, u). After that, we color edge P_{hg} with f' new colors to send a flow of f' to g using then 1 c/ϕ. For the case where $u = v = w$, we color only P_{xh} (with the $2f'$ colors from P_{vg} P_{ug}) instead of P_{wh}, $x \in N(v)$. After, we color not only edge P_{hg} but also edge P_{wx} with the same f' new colors.*

Figures 5(b), 5(d) show a possible extension for the case where there are two disjoint paths P_{vg} and P_{ug} that can be combined with P_{wg}. It represents the basic case where the *extension to a path* may be applied: there are three (possibly non-distinct) sources u, v and w linked to g by three node-disjoint paths. No matter these 3 paths close a cycle or not, it is possible to color them with only 3 colors. Indeed, we can color all edges with a, b, alternately, except for one of the three edges adjacent to g and possibly one edge adjacent to v (see Figure 5(d)), that has to be color c.

In case $u = v = w$ (Figure 5(d)) where v has three disjoint paths to g, at least two paths have the same parity and close an even cycle (see Lemma 9). For instance, these paths always exist if G is 3-connected. Corollary 3 gives an optimal solution of FCP in 3-connected graphs.

Lemma 8 *A path-extended combination can be defined if $\exists$ three paths P_{vg}, P_{ug}, P_{wg} such that $P_{vg} \cap P_{ug} \cap P_{wg} \in \{g, u, v, w\}$ where $\{g, u, v, w\}$ are only end-nodes in the paths P_{vg},P_{ug},P_{wg}.*

Lemma 9 *Assume node v_r has three node-disjoint paths to g and a remaining of ρ after Algorithm 2-by-2. If $B > 1$ then $\chi'_\Phi(G) = B$. Otherwise, there is only one source v with $b_v = 1$. In this case, $\chi'_\Phi(G) = 1$, if $v \in N(g)$, and $\chi'_\Phi(G) = 2$, otherwise.*

Proof: Since v_r has 3 disjoint paths to g then it is autonomous (there are necessarily 2 paths of the same parity). Then the remainig flow is $\rho = 1$ (if $\rho = 0$ we are done). The case $B = 1$ is trivial. If $B > 1$, we had at least one combination in Algorithm 2-by-2. We want to prove that the combination can be extended to a path. As we do not know if v_r participates in this combination, assume w.l.o.g v_r is the node w. So, node w has three disjoint paths P_1,P_2,P_3 to g and we have a combination (v, u). Name the nodes of the combination v,u and choose P_v,P_u such that nodes x,y satisfy the following conditions. Node x is the first node in the intersection $P_v \cap P_x$, $P_x \in \{P_1, P_2, P_3\}$; y is the first node in the intersection $P_u \cap P_y$, $P_y \in \{P_1, P_2, P_3\}$ and $y \notin P_{xg}$ from P_x. If $\exists x, y$ then adopt P_v as P_{vx} from P_v followed by P_{xg} from P_x, adopt P_u as P_{uy} from P_u followed by P_{yg} from P_y, and $P_w \in \{P_1, P_2, P_3\} \setminus \{P_x, P_y\}$. Otherwise (if $\nexists x$ or y), there is at least one path in $\{P_v, P_u\}$ that is disjoint from $\{P_1, P_2, P_3\}$. Let us say P_v is disjoint and P_u uses at least a node in $\{P_1, P_2, P_3\}$. So, use P_v, P_u as P_{uy} from P_u followed by P_{yg} from P_y, and $P_w \in \{P_1, P_2, P_3\} \setminus \{P_y\}$. Otherwise, adopt Pv, P_u and $P_w \in \{P_1, P_2, P_3\}$. So, we prove that any combination (v, u) can be extended to a path P_{wg} that sends ρ without using new colors. ■

Corollary 3 *Let G be 3-connected. If $B > 1$ then $\chi'_\Phi(G) = B$. Otherwise, there is only one source v with $b_v = 1$. In this case, $\chi'_\Phi(G) = 1$, if $v \in N(g)$, and $\chi'_\Phi(G) = 2$, otherwise.*

We did not explore this strategy fully, only to solve the cases above. Notice that, P_{wg} could use colors from more than one combination, then only some edges of P_{wg} need to be disjoint from the paths in the corresponding color provider combination. In other words, for each edge e of one path P_{wg}, it must exist at least one combination (v, u) where e is disjoint from P_{vg},P_{ug}.

5 Approximating the flow coloring index

After applying Algorithm 2-by-2, we saw several ways to send the residual demand ρ with extra colors (see Subsection 3.2 and Section 4). The $B - \rho$ used colors plus the number of additional colors provide an upper bound.

Lemma 10 *Let $v \in \mathcal{A}$. If **Case** II or $v_k \in \mathcal{A}$, we can adapt Algorithm 2-by-2 to end with the residual demand ρ at v.*

Proof: We can assume that $\rho = 1$. It follows that B is odd. Let $\bar{v} \in \mathcal{A}$. In order to show that ρ can end at $\bar{v}$, we initialize Algorithm 2-by-2 with $b'_v = b_v$, for $v \neq \bar{v}$, and $b'_{\bar{v}} = b_{\bar{v}} - 1$ (instead of $b_{\bar{v}}$). Let ρ' be the new residual demand. We have to show that $\rho' = 0$. Lemma 2 still holds for the new initialization. Notice that ρ' is even, the same parity as $B - 1$. Consider the two possible cases according to the hypothesis.

First suppose that $b_{v_k} < \sum_{v \in \mathcal{S} \setminus v_k} b_v$ (**Case** *II*). Clearly, $\rho' = 0$ if the second alternative of Lemma 2 still holds for the new initialization. Otherwise, we must have that $\bar{v} \neq v_k$ and $b'_{v_k} = b_{v_k} = \sum_{v \in \mathcal{S} \setminus v_k} b_v - 1 = \sum_{v \in \mathcal{S} \setminus v_k} b'_v$. By Lemma 2, we get $\rho' = 0$. In the complementary case, $v_k \in \mathcal{A}$ and $b_{v_k} \geq \sum_{v \in \mathcal{S} \setminus v_k} b_v + 1$ (because B is odd). Since we reduce only of the two sides of this inequality of 1 unit, we still have $b'_{v_k} \geq \sum_{v \in \mathcal{S} \setminus v_k} b'_v$ and b'_{v_k} continues to be the greatest demand. Again By Lemma 2, we have $\rho' = 0$. ■

Let A' be the set of nodes in $N(g)$ or nodes having three disjoint paths to g.

Theorem 1 *Let G be a 2-connected graph. Let $\sigma = b_{v_k} - \sum_{v \in \mathcal{S} \setminus v_k} b_v$. Then, $B \leqslant \chi'_\Phi(G) \leqslant B + \epsilon$, where*

$$\epsilon = \begin{cases} 0, & \text{if } (\sigma \leq 0 \text{ or } v_k \in \mathcal{A}) \text{ and } (B \text{ is even or } \mathcal{S} \cap A' \neq \emptyset); \\ 1, & \text{if } (\sigma = 1 \text{ and } v_k \in \mathcal{D}) \text{ or } (B \text{ is odd and } \mathcal{S} \cap A' = \emptyset) \text{ or } (\sigma \geq 2 \text{ and } v_k \in \mathcal{D}') \text{ or} \\ & (\sigma \leq 2\sum_{v \in \mathcal{S} \setminus v_k} b_v - 2\sum_{v \in N(v_k)} b_v \text{ and } v_k \in \mathcal{D} \setminus \mathcal{D}'); \\ \lfloor \frac{\sigma}{2} \rfloor & \text{if } \sigma > 2\sum_{v \in \mathcal{S} \setminus v_k} b_v - 2\sum_{v \in N(v_k)} b_v \text{ and } v_k \in \mathcal{D} \setminus \mathcal{D}'. \end{cases}$$

Proof: By Fact 1, we only need to prove the upper bound. For, we use Algorithm 2-by-2. Since it sends $B - \rho$ units of flow with $B - \rho$ colors, it suffices to show that the residual demand ρ can be sent with at most $\rho + \epsilon$ extra colors.

If $\sigma = 0$ (which implies B even), we get $\rho = 0$. So, we trivially have $\epsilon = 0$. If $\sigma < 0$ or $v_k \in \mathcal{A}$, the first alternative of Lemma 2 holds. In addition, B even or $\mathcal{S} \cap A' \neq \emptyset$ implies $\rho = 0$ or $\rho = 1$ concentrated at a node in A' by Lemma 10. If the node is in $N(g) \cap A'$ we can send directly using ρ colors. If the node is in $A' \setminus N(g)$, we only need ρ colors by Lemma 9. In both cases, $\epsilon = 0$.

If $\sigma = 1$ and $v_k \in \mathcal{D}$, Lemma 2 implies that $\rho = 1$, concentrated at a source outside $N(g)$. The same occurs if B is odd and $\mathcal{S} \cap N(g) = \emptyset$. In both cases, $\rho + \epsilon = 2$ colors are enough, because we can always send 1 unit of flow by a path using 2 colors (see Figure 1(b)). If $\sigma \geq 2$ and $v_k \in \mathcal{D}'$, $\epsilon \leq 1$ by Lemma 3. If $\sigma \leq 2\sum_{v\in\mathcal{S}\setminus v_k} b_v - 2\sum_{v\in N(v_k)} b_v$ and $v_k \in \mathcal{D} \setminus \mathcal{D}'$, we also obtain $\epsilon \leq 1$ by Corollary 2.

Finally, suppose that $\sigma \geq 2\sum_{v\in\mathcal{S}\setminus v_k} b_v - 2\sum_{v\in N(v_k)} b_v$ and $v_k \in \mathcal{D} \setminus \mathcal{D}'$. Since the first alternative of Lemma 2 holds, $\rho = \sigma$ is concentrated in a source $v \notin N(g)$ that participates in an odd cycle with g that does not respect any **EarCondition**. To show that the demand ρ can be sent with $\rho + \epsilon = \lfloor \frac{3\rho}{2} \rfloor$, we proceed as follows. First, we send $2\lfloor \frac{\rho}{2} \rfloor$ units of flow through an odd cycle with $3\lfloor \frac{\rho}{2} \rfloor$ colors (see Figure 1(c)). If ρ is even, we are done. Otherwise, there is a remaining flow of 1 in v. It can be sent to g using only one of the sides of the cycle (that is a path) with 2 colors. However, one of these colors can be a color used in a previous iteration to color a single edge from the other side of the cycle (e.g. color c in Figure 1(c)). Such a color exits, because $\rho \geq 2$. In this case, the number of colors used is $3\frac{\rho-1}{2} + 1 = \frac{3\rho-1}{2} = \lfloor \frac{3\rho}{2} \rfloor$. ■

The above bounds on $\chi'_\Phi(G)$ lead to an approximation factor for FCP related to Algorithm 2-by-2. We derive different approximation factors according to the cases of Theorem 1.

Corollary 4 *Given an instance (G, b) of FCP, let $\sigma = b_{v_k} - \sum_{v\in\mathcal{S}\setminus v_k} b_v$. Algorithm 2-by-2 provides an α-approximation factor, where:*

$$\alpha = \begin{cases} 1, & \text{if } (\sigma \leq 0 \text{ or } v_k \in \mathcal{A}) \text{ and } (B \text{ is even or } \mathcal{S} \cap A' \neq \emptyset); \\ 1 + \frac{1}{B}, & \text{if } (\sigma = 1 \text{ and } v_k \in \mathcal{D}) \text{ or } (B \text{ is odd and } \mathcal{S} \cap A' = \emptyset) \text{ or } (\sigma \geq 2 \text{ and } v_k \in \mathcal{D}') \text{ or} \\ & (\sigma \leq 2\sum_{v\in\mathcal{S}\setminus v_k} b_v - 2\sum_{v\in N(v_k)} b_v \text{ and } v_k \in \mathcal{D} \setminus \mathcal{D}'); \\ 1 + \lfloor \frac{\sigma}{2} \rfloor \frac{1}{B} & \text{if } \sigma > 2\sum_{v\in\mathcal{S}\setminus v_k} b_v - 2\sum_{v\in N(v_k)} b_v \text{ and } v_k \in \mathcal{D} \setminus \mathcal{D}'. \end{cases}$$

In general, the algorithm yields a $\frac{3}{2}$-approximation factor.

Proof: By Theorem 1, Algorithm 2-by-2 provides an α-approximation, for $\alpha = \frac{B+\epsilon}{B}$. Then, we get the desired value of α for each case. In all cases, we have $\alpha \leq 1 + \frac{\sigma}{2B} \leq \frac{3}{2}$, provided that $\sigma \leq B$. ■

6 Conclusion

In this work, we deal with a flow coloring problem with single-flow to *one* destination node and *integer* edge-coloring in 2-connected graphs. We identify several cases where $B \leq \chi'_\Phi(G) \leq B + 1$. An upper bound is given by a polynomial-time algorithm that finds a feasible flow and coloring, thus providing a $1 + \frac{1}{B}$–approximation for FCP. In some of these cases, we actually determine an upper bound of B (instead of $B+1$) and show that FCP is polynomial. In all these cases it follows, also by Fact 1, that $\chi'_{\Phi,f}(G) \leq \chi'_\Phi(G) \leq \chi'_{\Phi,f}(G) + 1$.

More precisely, we show the problem is polynomial with $\chi'_\Phi(G) = B$ when the node v_r with remaining demand from Algorithm 2-by-2 has three node-disjoint paths to g. A specific case that verifies this condition is the FCP in 3-connected graphs. It is worth stressing that we did not need the assumption of B even used in [BGR09] to show this result.

When node v_r does not have three node-disjoint paths to g or does not participate in a cycle to g satisfying any **EarCondition**, we consider some subcases. For instance, if $b_{v_r} \leq 3\sum_{v\in\mathcal{S}\setminus v_r} b_v - 2\sum_{v\in N(v_r)} b_v$, then $\chi'_\Phi(G) \leq B + 1$. For the complementary subcases, we get an approximation factor of $\frac{3}{2}$, thus improving the factor 2 presented in [Gom09]. For the moment, the complexity of the FCP is still open.

References

[BGR09] Jean-Claude Bermond, Cristiana Gomes, and Patricio Reyes. Round Weighting Problem and gathering in wireless networks with symmetrical interference. Rapport de recherche, UNSA, 2009.

[Gol73] M. K. Goldberg. On multigraphs of almost maximal chromatic class (in russian). *Diskret Analiz*, 23:3–7, 1973.

[Gom09] C. Gomes. *Radio Mesh Networks and the Round Weighting Problem.* PhD thesis, Université de Nice-Sophia Antipolis (UNS), Sophia Antipolis, France, December 2009.

[HK65] A. J. Hoffman and J.B. Kruskal. *Integral Boundary Points of Convex Polyhedra.* Linear Inequalities and Related Systems. Princeton University Press, 1965.

[HNS86] Dorit S. Hochbaum, Takao Nishizeki, and David B. Shmoys. A better than "best possible" algorithm to edge color multigraphs. *J. Algorithms*, 7(1):79–104, 1986.

[Hol81] I. Holyer. The np-completeness of edge-coloring. *SIAM Journal on Computing*, 10:718–720, 1981.

[iKLR10] Ken ichi Kawarabayashi, Zhentao Li, and Bruce A. Reed. Recognizing a totally odd k_4-subdivision, parity 2-disjoint rooted paths and a parity cycle through specified elements. In *SODA*, pages 318–328, 2010.

[KMP08] R. Klasing, N. Morales, and S. Pérennes. On the complexity of bandwidth allocation in radio networks. *Theoretical Computer Science*, 406(3):225–239, October 2008.

[Men27] Karl Menger. Zur allgemeinen Kurventheorie. *Fund. Math.*, 10:96–115, 1927.

[PW84] M. W. Padberg and L. A. Wolsey. Fractional covers for forests and matchings. *Math. Programming*, 29:1–14, 1984.

[Rey09] P. Reyes. *Data Gathering in Radio Networks.* PhD thesis, Ecole doctorale STIC, Université de Nice-Sophia Antipolis, August 2009.

[RS95] N. Robertson and P. D. Seymour. Graph minors XIII: The disjoint path problem. *Journal of Combinatorial Theory(Series B)*, 63:65–110, 1995.

[Sey79] P. D. Seymour. In graph theory and related topics. *J. A. Bondy and U. S. R. Murty, Eds. Academic Press*, pages 367–368, 1979.

[Sha49] C. E. Shannon. A theorem on colouring lines of a network. *J. Math. Phys.*, 28:148–151, 1949.

[SV82] Larry J. Stockmeyer and Vijay V. Vazirani. Np-completeness of some generalizations of the maximum matching problem. *Inf. Process. Lett.*, 15(1):14–19, 1982.

[Viz64] V. G. Vizing. On an estimate of the chromatic class of a p-graph (in russian). *Diskret. Analiz*, 3:23–30, 1964.