On the list coloring of 1-band buffering cellular graphs

Marine Collery[1] and Benjámin Martin Seregi[2]

Abstract— **The optimal channel allocation problem (CAP) in cellular networks is often formulated in a graph theoretic framework. One of its variants–where each access point knows the list of its free channels–is related to the so-called list coloring problem. In spite of the fact that the list coloring problem is NP-complete for arbitrary graphs, we showed that there exists a polynomial time algorithm for 1-band buffering cellular graphs; and as a corollary it turned out the choice number of such a graph is at most 4. In addition, we carried out performance comparisons between the existing integer linear programming solution and our implementation.**

I. Introduction

In telecommunication one of the most challenging problems is the efficient allocation of the available frequency. Since the available bandwidth is always limited (and expensive), the efficient utilization of the frequency spectrum is a major concern. Due to the fastest growing number of mobile Internet users, the optimal channel allocation in cellular networks and their variants have been heavily researched in recent years [1].

Several variants of the channel allocation problem (CAP) have been defined based on the different channel constraints that a particular service might require. One of them is the so-called co-channel constraint where the same channel is not allowed to be assigned to neighboring cells simultaneously. This problem have been formalized as a graph coloring problem by many authors [2]. Unfortunately, graph coloring is a well-known NP-complete problem [3] and therefore we do not know if a polynomial time algorithm for co-channel constraint satisfaction exists. Therefore various heuristic algorithms have been developed, the list of methods includes genetic algorithms, neural networks, graph-based and other approaches as well [1].

Cellular network topologies usually admit certain geometric structure. The most common network topology is the hexagonal grid topology where each cell is represented by a regular hexagon (two cells are neighbors if they share a common boundary). In [4], the authors exploited this special structure and proposed an algorithm that optimally solves the CAP in k-band buffering systems where $k = 1, 2$. Moreover, the algorithm has polynomial running time $O(p)$ where p is the number of cells.

The authors of [5] introduced a *distinctly different* CAP from all the above mentioned problems. Assuming a 2-band buffering hexagonal cell topology (the interference graph created from this topology is called cellular graph) where each cell has a fixed number of frequency channel (channels are either busy or free), they asked the following question: "*What is smallest size of the set of free channels associated with the cells (nodes of the cellular graph) that can guarantee interference free channel assignment to all the nodes?*". This problem is related to one of the generalizations of the graph coloring problem, called list coloring. It turned out that the required number of free channels is between 8 and 10. In addition, two algorithms have been proposed to create an interference free assignment, that is, a list-coloring of the cellular graph. The first one is the integer linear programming formulation of the list coloring problem (and therefore it is not a polynomial algorithm), the second one is a heuristic linear time algorithm that is, according to their experiment, within 12% of the optimal solutions.

In what follows, we formulate the same problem in 1-band buffering systems, then we outline a possibly polynomial time algorithm that optimally solves the list coloring problem in cellular graphs.

II. Problem statement

Before we state the problem it is necessary to introduce some definitions. A graph G is a *cellular graph* if it is constructed from the hexagonal cell topology in the following way: each cell is a node and two nodes are connected if and only if they share a common boundary. A cellular network is *k-band buffering* if the interference extends up to k cells. Let G be a graph and $L(v)$ a set of colors for all $v \in V(G)$ such that $|L(v)| = k$. We say that G is *k-choosable* if G is colorable such that the color of v is in $L(v)$ for all $v \in V(G)$, such colorings called *k-list coloring* of G. The *choice number* of G is the smallest $k \in \mathbb{N}$ (notated by $\mathrm{ch}(G)$) such that G is k-choosable.

Problem statement. Let G be a cellular graph of an 1-band buffering cellular network. Find the k-list coloring of G where $k = 4$ in polynomial time of the size (edges, nodes) of G.

III. Research questions

Let G be a cellular graph of an 1-band buffering cellular network.

1) What is the choice number of G?
2) How can we find an orientation of G such that the newly constructed directed graph G' does not contain directed cycles such that $d^+(v) \leqslant 3$ ($d^+(v)$ is the indegree of v) for all $v \in V(G')$ in polynomial time of the size of G'?
3) Does there exist a polynomial time algorithm that finds a kernel in the graph G', that is, an independent set

[1] student of Research Methodology and Scientific Writing Course at KTH Kista P1P22017. e-mail: collery@kth.se

[2] student of Research Methodology and Scientific Writing Course at KTH Kista P1P22017. e-mail: seregi@kth.se

$K \subseteq V(G)$ that satisfies the following: for each node $u \in V(G) \setminus K$ there is a node $v \in K$ such that $(u, v) \in E(G)$?

IV. Hypotheses

Our hypothesis is that it is possible to construct a polynomial time algorithm that can find a k-list coloring of G (G and k are as above) The hypothesis is based on the following theorems and conjectures.

The following theorem [6] (non-multigraph version from [7], Lemma 5.4.3) will play a central role in our algorithm.

Theorem 1 (Galvin, 1995): Let G be a graph and $\{L(v) : v \in V(G)\}$ given color sets. If G has an orientation D such that $d^+(v) < |L(v)|$ for all $v \in V(D)$ and every induced subgraph of D has a kernel, then G can be colored from the given color sets.

The proof of this theorem can be transformed into a polynomial time algorithm that solves the 4-list coloring problem by assuming the following hypotheses:

1) The choice number of a cellular graph is either 3 or 4 (see Appendix II).
2) It is possible to find such an orientation that is defined in Research questions section in polynomial time. This is necessary since finding a kernel in arbitrary graphs is NP-complete [8] (see Appendix I).
3) In cycle-free directed graphs (DAG), it is possible to find a kernel in polynomial time (DAGs are kernel-perfect by Richardson's theorem [9]).

V. Purpose

The purpose of this research is to reduce the interference in 1-band buffering cellular graphs and therefore achieve higher network throughput. To be more specific, it can be used to reduce the interference in 802.11 wireless systems where each access point has a list of free channels.

VI. Goals

The goal is to implement a polynomial time algorithm that computes interference free channel assignments in 1-band buffering cellular graphs and to prove that the choice number of cellular graphs is at most 4, that is, if each node has at least 4 free channels then there exists an interference-free channel assignment.

VII. Research Methodology

Research methods: theory and implementation: For the theoretical part, the method is analytical since the 1-band buffering cellular topology can be easily modeled as a graph. In order to prove that our solution is optimal and fast we cannot use empirical methods as it would not cover all the possible topologies that might arise in network deployments. For the implementation part–where we compare our solution with existing solution(s)–is empirical, that is, we compare different implementations using randomly generated topologies by measuring key resource metrics like CPU time or memory usage. The reason of carrying out experiments is twofold: (1) it is a way to test (not prove!) the correctness of the proofs, (2) the problem that the paper addresses is an engineering problem, that is, it is very important to provide key information on how the proposed algorithms would possibly perform in real-world scenarios.

In what follows we explain how we are conducting our research and how we are going to ensure that our research is ethical and reproducible.

Rules, ethics and reproducibility: During the research we are going to maintain a repository on GitHub. The authors–Marine Collery and Benjámin Seregi–are responsible for maintaining this repository that is available at `https://github.com/Benmartin92/research-kth`. It is a public space as we believe that being open-source is a good way to make a research *reproducible*.

The repository contains different branches each of them is related to a particular task in our research:

- research-plan (group)
- final-report-theory (Benjámin Seregi)
- final-report-implementation (Marine Collery)
- *additional tasks that are not related to our research: e.g. peer-review* (group)

The authors inside the parenthesis are responsible for the particular branch. The repository *research-plan* is the working repository for this document. Theory (*final-report-theory*) and implementation (*final-report-implementation*) reflect to the responsibilities that are already documented in our research proposal (IX. Tasks). A branch must contain individual contributions from its author. The authors are responsible to publish all of their work in the form of commits. The commit messages should be descriptive and informative and if it is necessary credit the source of the feedback that the particular patch is based on e.g. *"Initial value of variable x has been changed to π (feedback from Prof X.Y.)"* Since *final-report-implementation* is going to contain the source code of our implementation, the reproducibility is guaranteed.

On the ethics part, documenting our progress is of great importance. This way, we ensure that the team members pulling their weight in the research therefore we avoid ethical issues like crediting authorship.

VIII. Risks

In our research everything that is in the Hypothesis part is risk. In order to implement a polynomial time algorithm that solves our research problem we need to verify all hypotheses. Another source of risk can be the implementation of the proposed algorithms as the software development process can be delayed by some unforeseen implementation issues.

However, we would like to note that Hypothesis 1 has already been eliminated by Lemma 2. In addition, by using Lemma 2 we proved that if G is a cellular graph then $\mathrm{ch}(G) \leqslant 4$ (Theorem 2). This is important since from Thomassen theorem [10] we only knew that $\mathrm{ch}(G) \leqslant 5$ (since G is planar).

IX. Tools

For our implementation we will use the JGraphT Library [11]. It will provides us the essentials structure of graphs and help us generate our cellular graph. displaying our graph will be done through a Java Applet.

APPENDIX I
HYPOTHESIS 1

In this appendix we prove hypothesis 2.

Lemma 1: Let G be a cellular graph. Then G has a node v such that $\deg(v) \leqslant 3$.

Proof: Let us consider the planar drawing of the graph G which corresponds to its hexagonal topology. Let v be any node of G and we assign $(0, 0)$ to this node. We assign coordinates to every node starting from node v in the following way. If a node w is to the north of node v then the coordinates of node w is equal to the coordinates of node v plus $(0, 1)$. We summarize this method in Table I according to the cardinal directions. By applying this method to every

Direction	Change
N	$(0, 1)$
NE	$(1, 1)$
SE	$(1, 0)$
S	$(0, -1)$
SW	$(-1, -1)$
NW	$(-1, 0)$

TABLE I

RULES OF THE HEXAGONAL COORDINATE SYSTEM

node, we get a coordinate system within our hexagonal topology. Since there are finitely many node in $V(G)$, we can select the maximal node by selecting the node with maximal x- and y-coordinate, that is, a node w with the following property: if (x_1, y_1) is the coordinates of w and $v \neq w$ is another node with the coordinates (x_2, y_2) then $x_1 \geqslant x_2$ and $y_1 > y_2$ are satisfied.

To obtain a contradiction, suppose that the maximal node w has more than 3 neighbors. It follows then at least one if its neighbors is either to the north, northeast or southeast to w. However, this neighbor would violate the maximality of w. Therefore we proved that $\deg(w) \leqslant 3$. ■

Definition 1: We say that the orientation of a digraph G is *k-bounded* if for each node $v \in V(G)$ its outdegree is bounded by k, that is, $\deg^+(v) \leqslant k$.

Lemma 2: If G is a cellular graph, then G has a 3-bounded acyclic orientation.

Proof: Since $G = (V, E)$ is a cellular graph, we can make use of Lemma 1, that is, let v be a node of $G^{(0)} := G$ with $\deg(v) \leqslant 3$. We construct a function $f : V(G) \to \mathbb{N}$ by setting $f(v) := 0$ and then remove the node v from $G^{(0)}$ yielding the graph $G^{(1)}$. By repeating the same procedure - at step ith - we select a node v in $G^{(i)}$ such that $\deg(v) \leqslant 3$ and we set $f(v) := i$. Then we remove v from $G^{(i)}$ which yields the graph $V(G^{(i+1)}) := V(G^{(i)}) \setminus \{v\}$. It is trivial that the graphs $G^{(1)}, \ldots, G^{(|V(G)|)}$ are all cellular therefore it was valid using Lemma 1.

We note that the procedure ends after $|V(G)|$ steps.

Now we construct a digraph $H = (V, A)$ from G using the function f. Let $(u, v) \in E$ be an arbitrary edge. If $f(u) < f(v)$ then $A := A \cup (u, v)$ otherwise $A := A \cup (v, u)$. By repeating this procedure for all edges in E we get an orientation of G, that is, the digraph H. We need to prove that H is

1) acyclic and
2) $\deg^+(v) \leqslant 3$ holds for all $v \in V(G)$.

To prove (1), we will obtain a contradiction by assuming that H contains a directed cycle. Let C be a directed cycle in H with the nodes $C := \{v_1, v_2, \ldots, v_k\}$ that are ordered such that $(v_i, v_{i+1 \bmod k}) \in E$ $(i = 1, 2, \ldots, k)$ which means that $f(v_1) < f(v_2) < \ldots < f(v_{k-1}) < f(v_k)$ but also $f(v_k) < f(v_1)$ which is impossible.

What is left to show is that $\deg^+(v) \leqslant 3$. At each step we select a node with no more than 3 neighbors and those neighbors will be assigned a number that is greater than $f(v)$. Therefore the outdegree of v cannot be larger than 3. ■

It is easy to see that the proof of this lemma can be transformed into a polynomial time algorithm. In what follows we outline the algorithm (Algorithm 1) that is based on the proof of Lemma 2 and then we prove that its running time is $O(|V(G)| + |E(G)|)$.

```
Data: A cellular graph $G = (V, E)$
Result: An acyclic 3-bounded orientation of $G$
$G^{(0)} := G$;
Initialize $f : V(G) \to \mathbb{N}$;
for $i \leftarrow 0$ to $|V(G)| - 1$ do
    Select $v \in V(G^i)$ such that $\deg(v) \leqslant 3$;
    $f(v) := i$;
    $V(G^{(i+1)}) := V(G^i) \setminus \{v\}$;
end
Initialize $H := (V, A)$;
forall $e = (u, v) \in E(G)$ do
    if $f(u) < f(v)$ then
        $A := A \cup (u, v)$;
    else
        $A := A \cup (v, u)$;
    end
end
```

Algorithm 1: Constructing an acyclic 3-bounded orientation of a cellular graph

Proposition 1: Algorithm 1 has a running time of $O(|V(G)| + |E(G)|)$.

Proof: It is enough to prove that "Select $v \in V(G^i)$ such that $\deg(v) \leqslant 3$" can be done in $O(|V(G)|)$ since the rest is obvious. Let us initialize a queue $Q := \{v \mid \deg(v) \leqslant 3\}$ before we start running the algorithm. We pop a node v from G at every iteration and push new ones after the removal of v if there are new nodes in G that satisfy the degree condition. The pop and push methods can be implemented in constant time which completes the proof. ■

It is important to note that the idea of Algorithm 1 came from [12] where the authors proved that every planar graph has a 5-bounded acyclic orientation. They used the fact that every planar graph has a node with at most 5 neighbors which follows from Euler's formula.

APPENDIX II
CHOICE NUMBER OF CELLULAR GRAPHS

In this appendix we prove hypothesis 1).

Theorem 2: If G is a cellular graph then $ch(G) \leqslant 4$.

Proof: Since G is cellular, we can consider its 3-bounded acyclic orientation by Lemma 2. This orientation is kernel-perfect by Richardson's theorem [9]. Therefore we can apply Theorem 1 (with $d^+(v) \leqslant 3$ and $L(v) = 4$) which concludes the proof. ■

We note that $3 = \chi(G) \leqslant \mathrm{ch}(G) \leqslant 4$ (if G is cellular then $\chi(G) = 3$ [4] (Theorem 1)) therefore the choice number of a cellular graph is either 3 or 4.

REFERENCES

[1] G. K. Audhya, K. Sinha, S. C. Ghosh, and B. P. Sinha, "A survey on the channel assignment problem in wireless networks," *Wirel. Commun. Mob. Comput.*, vol. 11, no. 5, pp. 583–609, May 2011. [Online]. Available: http://dx.doi.org/10.1002/wcm.898

[2] W. K. Hale, "Frequency assignment: Theory and applications," *Proceedings of the IEEE*, vol. 68, no. 12, pp. 1497–1514, Dec 1980.

[3] R. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*, R. Miller and J. Thatcher, Eds. Plenum Press, 1972, pp. 85–103.

[4] A. Sen, T. Roxborough, and S. Medidi, "Upper and lower bounds of a class of channel assignment problems in cellular networks," in *INFOCOM '98. Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, Mar 1998, pp. 1284–1291 vol.3.

[5] R. Wang, C. Zhou, A. Mazumder, A. Das, H. A. Kierstead, and A. Sen, "Upper and lower bounds of choice number for successful channel assignment in cellular networks," in *2015 IEEE International Conference on Communications (ICC)*, June 2015, pp. 3370–3375.

[6] F. Galvin, "The list chromatic index of a bipartite multigraph," *J. Comb. Theory Ser. B*, vol. 63, no. 1, pp. 153–158, Jan. 1995. [Online]. Available: http://dx.doi.org/10.1006/jctb.1995.1011

[7] R. Diestel, *Graph Theory (Graduate Texts in Mathematics)*. Springer, August 2005.

[8] V. Chvátal, "On the computational complexity of finding a kernel," *Report No. CRM-300, Centre de Recherches Mathematiques, Universite de Montreal*, 1973.

[9] M. Richardson, "On weakly ordered systems," *Bull. Amer. Math. Soc.*, vol. 52, no. 2, pp. 113–116, 02 1946. [Online]. Available: https://projecteuclid.org:443/euclid.bams/1183507698

[10] C. Thomassen, "Every planar graph is 5-choosable," *J. Comb. Theory Ser. B*, vol. 62, no. 1, pp. 180–181, Sep. 1994. [Online]. Available: http://dx.doi.org/10.1006/jctb.1994.1062

[11] "Welcome to JGraphT - a free Java Graph Library." [Online]. Available: http://jgrapht.org/

[12] M. Chrobak and D. Eppstein, "Planar orientations with low out-degree and compaction of adjacency matrices," *Theoretical Computer Science*, vol. 86, no. 2, pp. 243 – 266, 1991. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0304397591900203