Equitable Partition of Graphs into Independent Sets and Cliques

Bruno Monteiro[1], **Vinicius F. dos Santos**[1]

[1]DCC - Universidade Federal de Minas Gerais (UFMG)
Belo Horizonte - MG - Brazil

{monteirobruno,viniciussantos}@dcc.ufmg.br

Abstract. *A graph is $(\boldsymbol{k}, \boldsymbol{l})$ if its vertex set can be partitioned into k independent sets and l cliques. Deciding if a graph is $(\boldsymbol{k}, \boldsymbol{l})$ can be seen as a generalization of coloring, since deciding is a graph belongs to $(\boldsymbol{k}, \boldsymbol{0})$ corresponds to deciding if a graph is k-colorable. A coloring is equitable if the cardinalities of the color classes differ by at most 1. In this paper, we generalize both the $(\boldsymbol{k}, \boldsymbol{l})$ and the equitable coloring problems, by showing that deciding whether a given graph can be equitably partitioned into k independent sets and l cliques is solvable in polynomial time if* $\max(k, l) \leq 2$*, and NP-complete otherwise.*

1. Introduction

The *Vertex Coloring Problem* (VCP) consists of partitioning the vertex set of a graph into k disjoint sets, such that no edges has both endpoints in the same set. This problem is known to be NP-complete [Karp 1972]. A widely studied variation of this problem is the *Equitable Coloring Problem*, in which we want the cardinalities of the sets to differ by at most 1. The latter is also NP-Complete, since it can be reduced from VCP by adding sufficiently many isolated vertices.

Let $(\boldsymbol{k}, \boldsymbol{l})$ be the set of graphs that can be partitioned into k independent sets and l cliques. Recognizing $(\boldsymbol{k}, \boldsymbol{l})$ graphs can be seen as a generalization of the *Equitable Coloring Problem* (ECP), since deciding if a graph belongs to $(\boldsymbol{k}, \boldsymbol{0})$ corresponds to deciding if the graph is k-colorable. This problem was shown to be solvable in polynomial time if $\max(k, l) \leq 2$, and NP-complete otherwise [Brandstädt 1996, Brandstädt 1998, Brandstädt 1984].

In this paper we introduce the equitable version of the $(\boldsymbol{k}, \boldsymbol{l})$ problem: we want the partition to be equitable. For that, let's define $(\boldsymbol{k}, \boldsymbol{l})_{\boldsymbol{E}}$ as the set of graphs that can be equitably partitioned into k independent sets and l cliques, that is, the cardinalities of the sets differ by at most 1. It is easy to see that this is a generalization of ECP.

If $\overline{G}$ is the complement graph of G, then $G \in (\boldsymbol{k}, \boldsymbol{l})_{\boldsymbol{E}}$ if and only if $\overline{G} \in (\boldsymbol{l}, \boldsymbol{k})_{\boldsymbol{E}}$. Checking whether $G \in (\boldsymbol{1}, \boldsymbol{0})_{\boldsymbol{E}}$ is to check if G has no edges, and to check whether $G \in (\boldsymbol{0}, \boldsymbol{1})_{\boldsymbol{E}}$ is to check if G is complete. Both can be easily done in linear time.

2. Recognition of $(1, 1)_E$ Graphs

For a vertex $x \in V$, let $N(x)$ be the neighborhood of x and $\overline{N}(x)$ be the set of non-neighbors of x. We will also denote $N[x]$ as $N(x) \cup x$ and $\overline{N}[x]$ as $\overline{N}(x) \cup x$.

Let $G = (V, E)$ be a graph, and $x \in V$ a vertex from G. We want to find an independent set I and a clique C such that $I \cup C = V$. If $x \in I$, then it does not have a neighbor in I. Therefore, $x \in I \Rightarrow N(x) \subseteq C$, that is, $N(x) \in \mathbf{(0,1)}$. Similarly, if $x \in C$, then it is neighbor with all other vertices in C: $x \in C \Rightarrow \overline{N}(x) \in \mathbf{(1,0)}$. So we check, for every vertex x from G, if **1)** $N(x) \in \mathbf{(0,1)}$ and **2)** $\overline{N}(x) \in \mathbf{(1,0)}$.

If there is a vertex that does not satisfy any of the previous conditions, then $G \notin \mathbf{(1,1)}_{\boldsymbol{E}}$. If there is a vertex x such that both **1** and **2** holds, then we can observe that $\overline{N}[x] \in \mathbf{(1,0)}$, and $N[x] \in \mathbf{(0,1)}$. Therefore, it is enough to insert x in the set that has fewer elements, and we get the partitions $I = \overline{N}[x]$ and $C = N(x)$, or $I = \overline{N}(x)$ and $C = N[x]$. If the partition is equitable, then $G \in \mathbf{(1,1)}_{\boldsymbol{E}}$. Otherwise, $G \notin \mathbf{(1,1)}_{\boldsymbol{E}}$.

If every vertex satisfy exactly one of the conditions, we can construct the sets I and C in the following manner: for each vertex $x \in G$, if x satisfies condition **1**, insert it in I. if x satisfies condition **2**, insert it in C. If at the end of the construction $I \in \mathbf{(1,0)}$ $\wedge$ $C \in \mathbf{(0,1)} \wedge ||I| - |C|| \leq 1$, then $G \in \mathbf{(1,1)}_{\boldsymbol{E}}$. Otherwise, $G \notin \mathbf{(1,1)}_{\boldsymbol{E}}$.

Checking conditions **1** and **2** can be easily done in $\mathcal{O}(|V|^2)$ time. Hence, deciding if $G \in \mathbf{(1,1)}_{\boldsymbol{E}}$ can be solved in $\mathcal{O}(|V|^3)$ time.

3. Recognition of $(2,0)_{\boldsymbol{E}}$ Graphs

Deciding whether a graph $G = (V, E) \in \mathbf{(2,0)}_{\boldsymbol{E}}$ is the same as deciding if G is equitably bipartite. For the sake of completeness we will prove this result here.

Verifying if $G \in \mathbf{(2,0)}$ can be easily done in $\mathcal{O}(|V|^2)$. If G is connected we check if the partition is equitable. Otherwise, we initially have two possibilities of partitioning for each connected component. In that case, we first find a bipartition for each of the c connected components of G. If any of them is not bipartite, then $G \notin \mathbf{(2,0)}_{\boldsymbol{E}}$. Let x_i and y_i be the number of vertices in each of the sets from the partition of the i-th connected component of G.

We define the function $f(i, S)$, that equals TRUE if starting from the connected component with index i, there is a partition in two independent sets such that the number of vertices in one of the two sets is equal to S. Otherwise, $f(i, S)$ equals FALSE. Now we can notice that $f(1, \lfloor\frac{|V|}{2}\rfloor)$ answers if $G \in \mathbf{(2,0)}_{\boldsymbol{E}}$. This strategy is similar to the standard pseudo-polynomial algorithm for the Partition problem, with the difference that, in our case, numbers are bounded by the instance size.

Theorem 1. *Deciding if $G = (V, E) \in \mathbf{(2,0)}_{\boldsymbol{E}}$ can be solved in $\mathcal{O}(|V|^2)$ time.*

Proof. To prove that $f(1, \lfloor\frac{|V|}{2}\rfloor)$ can be computed in $\mathcal{O}(|V|^2)$ time, we will show that we only have to compute $\mathcal{O}(|V|^2)$ values of f, and that each of them can be computed in $\mathcal{O}(1)$ time, using previously computed values.

To actually compute $f(i, S)$, we check both possibilities of partition of the i-th connected component. Depending on which one we choose, the number of vertices we will need starting from the $(i+1)$-th connected component is either $S - x_i$ or $S - y_i$. The full algorithm to compute $f(i, S)$ is shown is Algorithm 1.

Since the values in the first parameter of f varies from 1 to c, which is $\mathcal{O}(|V|)$, and the second parameter varies from $-\lceil\frac{|V|}{2}\rceil$ to $\lfloor\frac{|V|}{2}\rfloor$, which is also $\mathcal{O}(|V|)$, computing $f(1, \lfloor\frac{|V|}{2}\rfloor)$ depends on $\mathcal{O}(|V|^2)$ previous values of f. Moreover, the computation of f takes $\mathcal{O}(1)$ assuming the values it depends on are already computed. Therefore, as long as values are not recalculated, computing $f(1, \lfloor\frac{|V|}{2}\rfloor)$ can be done in $\mathcal{O}(|V|^2)$ time. □

Algorithm 1

Input: number c of connected components; $x_1, y_1, \ldots, x_c, y_c$, number of vertices in each set of the partition.

function $f(i, S)$
    if $i = c$ **then**
        if $S = x_c \vee S = y_c$ **then return** TRUE
        else return FALSE
    else return $f(i+1, S - x_i) \vee f(i+1, S - y_i)$

4. Recognition of $(2, 1)_E$ Graphs

For the recognition of $(\mathbf{2}, \mathbf{1})_{\boldsymbol{E}}$ graphs, we will use an algorithm to find a $(\mathbf{2}, \mathbf{1})$ partition of G. If $G \notin (\mathbf{2}, \mathbf{1})$, then $G \notin (\mathbf{2}, \mathbf{1})_{\boldsymbol{E}}$. Let I_1, I_2, and C be respectively the independent sets and the clique that partition G, such that $I_1 \cup I_2 \cup C = V(G)$. Let I'_1, I'_2, and C' be respectively the independent sets and the clique that equitably partition G. Since I_1 is an independent set, it is true that $\forall u, v \in I_1, (u, v) \notin E(G)$. Therefore, at most one vertex of I_1 can belong to C', given that C' is a clique. Similarly, at most two vertices of C can belong to $I'_1 \cup I'_2$, each one in one of the sets.

Now, we look at all the possibilities to insert I_1, I_2, and C in I'_1, I'_2, and C'. For each possibility, we check if $I'_1 \cup I'_2 \in (\mathbf{2}, \mathbf{0})_{\boldsymbol{E}}$, if $C' \in (\mathbf{0}, \mathbf{1})$, and if the partition is equitable. If that is the case for any of the possibilities, then $G \in (\mathbf{2}, \mathbf{1})_{\boldsymbol{E}}$. Otherwise, $G \notin (\mathbf{2}, \mathbf{1})_{\boldsymbol{E}}$.

There are $\mathcal{O}(|V|)$ ways to choose one vertex from I_1, $\mathcal{O}(|V|)$ ways for I_2 and $\mathcal{O}(|V|^2)$ ways to choose two vertices from C. For every choice, checking whether $I'_1 \cup I'_2 \in (\mathbf{2}, \mathbf{0})_{\boldsymbol{E}}$ can be done in $\mathcal{O}(|V|^2)$ time, from Theorem 1. It is trivial to check if $C' \in (\mathbf{0}, \mathbf{1})$ in $\mathcal{O}(|V|^2)$. Hence, checking each of the $\mathcal{O}(|V|^4)$ possibilities is $\mathcal{O}(|V|^2)$, which gives a total of $\mathcal{O}(|V|^6)$. It is known that a $(\mathbf{2}, \mathbf{1})$ partition of a graph can be found in $\mathcal{O}(|V|^4)$ time [Brandstädt 1996, Brandstädt 1998, Brandstädt 1984], so deciding if $G \in (\mathbf{2}, \mathbf{1})_{\boldsymbol{E}}$ can be done in $\mathcal{O}(|V|^6)$ time.

5. Recognition of $(2, 2)_E$ Graphs

We will use the same technique that was used in the $(\mathbf{2}, \mathbf{1})_{\boldsymbol{E}}$ recognition for the $(\mathbf{2}, \mathbf{2})_{\boldsymbol{E}}$ recognition. If $G \notin (\mathbf{2}, \mathbf{2})$, then $G \notin (\mathbf{2}, \mathbf{2})_{\boldsymbol{E}}$. Now suppose that $G \in (\mathbf{2}, \mathbf{2})$. Let I_1, I_2, C_1, and C_2 be respectively the independent sets and the cliques that equitably partition G.

Since I_1 is an independent set, then at most two of it's vertices can belong to $C'_1 \cup C'_2$, each one in one of the cliques, and the same is true for I_2. Moreover, since C_1 is a clique, then at most two of it's vertices can belong to $I'_1 \cup I'_2$, each one in one of the sets, and that also holds for C_2. So, there are $\mathcal{O}(|V|^8)$ possibilities of insertion in I'_1, I'_2, C'_1, and C'_2. If any of them is valid, that is, if they make an equitable partition of G, then $G \in (\mathbf{2}, \mathbf{2})_{\boldsymbol{E}}$. Otherwise, $G \notin (\mathbf{2}, \mathbf{2})_{\boldsymbol{E}}$.

For every one of the $\mathcal{O}(|V|^8)$ possibilities, we check if $I'_1 \cup I'_2 \in (\mathbf{2}, \mathbf{0})_{\boldsymbol{E}}$, if $C'_1 \cup C'_2 \in (\mathbf{0}, \mathbf{2})_{\boldsymbol{E}}$ and if the partition is equitable. Those conditions can be verified in $\mathcal{O}(|V|^2)$, from Theorem 1. Thus, given a $(\mathbf{2}, \mathbf{2})$ partition of G, we can answer if $G \in (\mathbf{2}, \mathbf{2})_{\boldsymbol{E}}$ in $\mathcal{O}(|V|^{10})$ time.

The best known algorithm capable of finding a $\boldsymbol{(2,2)}$ partition of a graph runs in $\mathcal{O}(|V|^{12})$ [Brandstädt 1996, Brandstädt 1998, Brandstädt 1984]. Hence, we are limited by that complexity, and answering whether $G \in \boldsymbol{(2,2)_E}$ can be done in $\mathcal{O}(|V|^{12})$ time.

6. $\boldsymbol{(k,l)_E}$ recognition for other values of k and l

We will now show that $\boldsymbol{(k,l)_E}$ recognition for $\max(k,l) \geq 3$ is NP-complete. For that, we will make a reduction from the problem EQUITABLE COLORING. It is known that deciding whether a given graph G can be equitably colored with k colors is NP-complete for $k \geq 3$.

Let's define the *join* of two graphs G and H as the graph obtained by the disjoint union of G and H and the addition of all edges between vertices of G and H. It suffices to show that $\boldsymbol{(k,l)_E}$ is NP-complete for $k \geq 3$, since that $G \in \boldsymbol{(k,l)_E} \Leftrightarrow \overline{G} \in \boldsymbol{(l,k)_E}$.

Theorem 2. *Deciding if* $G = (V,E) \in \boldsymbol{(k,l)_E}$ *for* $\max(k,l) \geq 3$ *is NP-complete.*

Proof. Since checking independent sets and cliques can be easily done in $\mathcal{O}(|V|^2)$, the problem belongs to NP. An instance of the EQUITABLE COLORING receives a graph $G = (V,E)$ and an integer k, and outputs YES if G can be equitably colored with k colors. We may assume that $|V| \geq 2k$, otherwise we can color G in polynomial time by finding a maximum matching in the complement, and that k divides $|V|$, by adding an isolated clique of size at most $k-1$. Let $r = \frac{|V|}{k}$. Notice that $r \geq 2$.

Let C a complete graph of $l \cdot r$ vertices and let G' be the join of G and C. We will show that $G' \in \boldsymbol{(k,l)_E}$ if and only if G can be equitable colored with k colors. If G can be equitably colored with k colors, then each color class of an equitable coloring of G has exactly r vertices. Hence we can partition the original vertices of G' into k independent sets, and vertices from C into the l cliques of size r.

Conversely, if $G' \in \boldsymbol{(k,l)_E}$, then we must make the following observation: if a vertex from $V(C)$ belongs to one of the k independent sets, then no other vertex of G' can belong to that independent set, since vertices of $V(C)$ are universal in G'. Now, due to the fact that $|V(G)| \geq 2k$, we have that $|V(G')| \geq 2(k+l)$, that is, each independent set has at least 2 vertices in any equitable partition. Therefore, none of the inserted vertices of G' belong to an independent set. So the original vertices of G form the k independent sets, that is, G can be equitably colored with k colors. □

References

Brandstädt, A. (1984). Partitions of graphs into one or two independent sets and cliques. Technical report, Forschungsergebnisse der Friedrich-Schiller-Universitatät Jena.

Brandstädt, A. (1996). Partitions of graphs into one or two independent sets and cliques. *Discrete Mathematics*, 152(1-3):47–54.

Brandstädt, A. (1998). Corrigendum. *Discrete Math.*, 186:295.

Karp, R. M. (1972). Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer.