Distance-2 Edge Coloring is NP-Complete

Jeff Erickson Shripad Thite David P. Bunde
Department of Computer Science, University of Illinois at Urbana-Champaign
{jeffe,thite,bunde}@uiuc.edu

March 1, 2002

Abstract

We prove that it is NP-complete to determine whether there exists a distance-2 edge coloring (strong edge coloring) with 5 colors of a bipartite 2-inductive graph with girth 6 and maximum degree 3.

Let G be a simple, undirected graph. We say that two edges of G are within distance 2 of each other if either they are adjacent or there is some other edge that is adjacent to both of them. A *distance-2-edge-coloring* of G is an assignment of colors to edges so that any two edges within distance 2 of each other have distinct colors, or equivalently, a vertex-coloring of the square of the line graph of G. If the coloring uses only k colors, it is called a *k-D2-edge-coloring*, and the graph G is said to be *k-D2-edge-colorable*. Any k-D2-edge-colorable graph is also $(k+1)$-D2-edge-colorable. A distance-2-edge-coloring is also known as a *strong edge coloring*. Mahdian [2, 3] proved, via a reduction from Graph k-colorability, that it is NP-complete to determine, for every fixed g, whether a bipartite graph with girth g has a strong edge coloring with k colors for $k \geq 4$. We present a new proof that shows that the strong edge coloring problem is NP-complete for bipartite 2-inductive graphs of maximum degree 3.

Definition 1 (c-inductive graphs). *A graph G is said to be c-inductive if the vertices can be numbered so that at most c neighbors of any vertex v have higher numbers than v.*

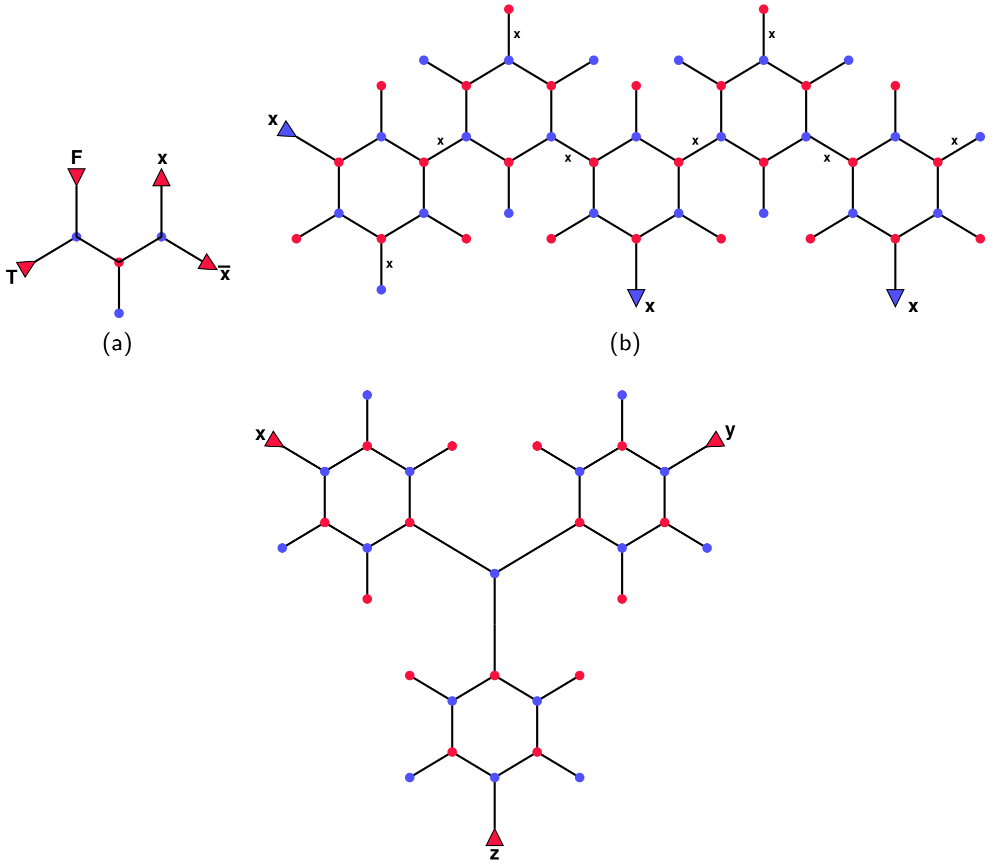
Theorem 2. *Determining whether a bipartite 2-inductive graph with girth 6 and maximum degree 3 is 5-D2-edge-colorable is NP-complete.*

Proof: The problem is clearly in NP since a coloring can be verified in polynomial time. To prove NP-hardness, we describe a reduction from Not-All-Equal-3SAT [1, Problem LO3].

> Instance: A set $X = \{x_1, x_2, \ldots, x_n\}$ of variables, and a collection $C = \{C_1, C_2, \ldots, C_m\}$ of boolean clauses over X, each with exactly three literals.
>
> Question: Is there a truth assignment for X such that each clause in C has at least one true literal and at least one false literal?

Given an instance (X, C) of Not-All-Equal-3SAT, we will reduce it to a graph $G(X, C)$ that has a 5-D2-edge-coloring (hereafter, a *valid coloring*) if and only if (X, C) has a satisfying assignment. We will label the five colors $\{T, F, 1, 2, 3\}$, where T and F represent the boolean values True and False.

Our reduction uses three types of gadgets, all illustrated in Figure 1.



(a)

(b)

(c)

Figure 1. (a) A variable gadget. (b) A fanout gadget. (c) A clause gadget. Triangles indicate input and output edges.

- **Fanout gadget:** This extendible gadget has a valid coloring with five colors such that the output edges have the same color as the input edges, regardless of which two colors are used on the two edges adjacent to the output edge. In fact, all marked edges in the figure are required to be the same color. Our reduction uses $2n + 2$ fanout gadgets: a *truth* gadget, a *falsehood* gadget, and two *literal* gadgets for each variable in X. Without loss of generality, the output edges of the truth gadget are colored T, and the output edges of the falsehood gadget are colored F.

- **Variable gadget:** This gadget has two input edges, three internal edges, and two output edges. One input edge is connected to the truth gadget, the other connected to the falsehood gadget. (Thus, the truth and falsehood gadgets must output different colors.) The output edges are colored T and F in any valid coloring, but either assignment of colors is possible. Each output is connected to one of the literal fanout gadgets. The reduction uses n variable gadgets, one for each variable in X.

- **Clause gadget:** This gadget has three input edges, each connected to an output edge of the appropriate literal gadget. The clause gadget has a valid coloring if and only if the three input edges are *not* all the same color. The reduction uses m clause gadgets, one for each clause in C.

Overall, our graph $G(X, C)$ has complexity $O(n + m)$ vertices and edges, and we can easily construct it in linear time. The graph is bipartite because its vertices can be consistently colored with two colors, red and blue, as shown in Figure 1. The graph has girth 6 and maximum degree 3. It is easy to see that the graph is also 2-inductive—repeatedly delete a vertex of smallest degree, which has degree 1 or 2, and consider the vertices in the reverse order. Any satisfying assignment for (X, C) can be transformed into a valid coloring for $G(X, C)$, and vice versa, in $O(n+m)$ time. $\square$

References

[1] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman and Co., 1979.

[2] M. Mahdian. The strong chromatic index of graphs. Master's thesis, Department of Computer Science, University of Toronto, 2000.

[3] M. Mahdian. On the computational complexity of strong edge coloring. *Discrete Applied Mathematics*, 118:239–248, 2002.