

# Determining a Graph's Chromatic Number for Part Consolidation in Axiomatic Design

Jeffery Cavallaro

03 September 2019

## Abstract

Mechanical engineering design practices are increasingly moving towards a framework called *axiomatic design* that starts with a set of independent *functional requirements* (FRs) for a manufactured product. A key tenet of axiomatic design is to decrease the *information content* of a design in order to increase the chance of manufacturing success. One important way to decrease information content is to fulfill multiple functional requirements by a single part.

But what is the best way to allocate FRs to parts? One possible answer is to represent the problem by a graph, where the vertices are the FRs and the edges represent the need to separate their endpoint FRs into separate parts. The answer then becomes the solution to a vertex coloring problem. This research will investigate a new algorithm that determines the chromatic number for such a graph, thus determining the minimum possible number of parts in a design. During the design phase, determining the minimum number of parts is much more important than determining the actual allocation of FRs to parts because the former is a key attribute used to compare competing designs. Once the design is finalized, traditional coloring algorithms can then be used to obtain an actual chromatic coloring.

This algorithm was initially developed as part of an NSF grant in collaboration with two mechanical engineering researchers at NYU Buffalo. A subsequent paper describing the algorithm was presented at the International Design Engineering Technical Conference (IDETC) in August 2019.

The primary goals of this research project are to present the proposed algorithm with formal theoretical support, as was done in the previously-mentioned paper, and then perform theoretical and empirical runtime complexity comparisons to a standard Zykov-like exhaustive NP-complete algorithm for vertex partitioning. Since previous work has relied on manual execution of the algorithm under study, an additional goal is to develop software solutions that can extend the ability to try and compare various examples. The availability of a good software tool will provide the ability to run the algorithm on sets of random graphs in order to empirically demonstrate the theoretical results.

## 297 Summary

This work began as a Math-297 project. The goals of that project were to:

1. Learn the basics of axiomatic design.
2. Understand the steps of the proposed algorithm and make corrections where necessary.
3. Formalize the theoretical support for the steps of the algorithm.
4. Provide some initial software support to execute the algorithm on example graphs.
5. Collaborate with the research team on a paper to be presented at IDETC 2019.

The normal approach for an exhaustive chromatic coloring algorithm is to adopt a recursive technique. For a graph  $G$ :

$$\chi(G) = \min \{ \chi(G \cdot uv), \chi(G + uv) \}$$

where  $u, v \in V(G)$  and  $uv \notin E(G)$ . The first recursive call uses vertex contraction to indicate that two non-adjacent vertices are assigned the same color. The second recursive call uses edge addition to indicate that two previously non-adjacent vertices are assigned different colors. Thus, the leaves of the resulting Zykov tree are complete graphs whose orders enumerate all of the possible proper colorings of  $G$ . Picking a leaf with the smallest order represents a chromatic coloring. Note that this Turing Machine-type approach results in NP-complete running time.

The proposed algorithm takes a somewhat different approach. Instead of relying solely on recursive calls, it loops for increasing values of candidate  $k$ -colorability. Focusing on a candidate  $k$  value allows various subgraph simplification, branch success, and branch failure checks to be applied before having to resort to the same recursive calls. The first success condition encountered indicates a minimum  $k$ . In fact, running the algorithm on the examples provided by the ME collaborators resulted in solutions with at most one recursive call.

The software developed to run the algorithm and display the stepwise results and final solutions in  $\text{\LaTeX}$  TikZ format provided paper-ready results in seconds.

The paper was successfully completed and presented at IDETC 2019, where it was awarded a best paper award.

## Timeline

The thesis will be organized into the following sections:

Topic/Section	weeks
Axiomatic design	1
Graph theory basics	2
Chromatic number theorems	2
Zykov trees and runtime complexity	1
Proposed algorithm and runtime complexity	1
Possible shortcuts for starting $k$ (cliques, Hoffman)	2

The topic of spectral graph theory may be of some use (especially in the case of Hoffman), but its impact is not clear to the author or his advisor at this time. Reading in spectral graph theory will occur in parallel to the above efforts and will be incorporated where appropriate.

## Reading List

- [1] G. Agnarsson and R. Greenlaw, *Graph theory: Modeling, applications, and algorithms*, Prentice Hall, Upper Saddle River, NJ, 2007.
- [2] S. Behdad, P.K. Gopalakrishnan, S. Jahanbekam, and J. Cavallaro, *A graph coloring technique for identifying the minimum number of parts for physical integration in product design*, Proceedings of the ASME 2019 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference (2019), no. DETC2018-98251.
- [3] D. Chartrand and P. Zhang, *A first course in graph theory*, Dover Publications, Inc., Mineola, NY, 2012.
- [4] C. Godsil and G. Royle, *Algebraic graph theory*, Springer, New York, 2001.
- [5] C. McDiarmid, *Determining the chromatic number of a graph*, Siam Journal on Scientific Computing **8** (1979), no. 1.
- [6] B. Nica, *A brief introduction to spectral graph theory*, European Mathematical Society, Zurich, 2018.
- [7] R.C. Reed (ed.), *Graph theory and computing*, Academic Press, New York and London, 1972.
- [8] N.P. Suh, *The principles of design*, Oxford University Press, New York, 1990.
- [9] ———, *Axiomatic design theory for systems*, Research in Engineering Design **10** (1998), no. 4, 189–209.
- [10] D.B. West, *Introduction to graph theory*, 2<sup>nd</sup> ed., Prentice Hall, Upper Saddle River, NJ, 2001.