UNIVERSITY OF LJUBLJANA
FACULTY OF MATHEMATICS AND PHYSICS

Financial mathematics – 1st cycle

Anej Rozman, Tanja Luštrek

Rich-Neighbor Edge Colorings

Term Paper in Finance Lab

Short Presentation

Advisers: Assistant Professor Janoš Vidali,
Professor Riste Škrekovski

Ljubljana, 2023

1. Introduction

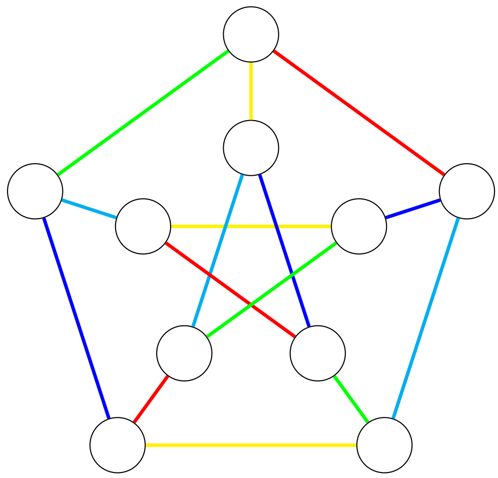
In this paper we set out to analyse an open conjecture in a modern graph theory problem known as rich-neighbor edge coloring.

Definition 1.1. In an edge coloring, an edge e is called *rich* if all edges adjacent to e have different colors. An edge coloring is called a *rich-neighbor edge coloring* if every edge is adjacent to some rich edge.

Definition 1.2. $X'_{rn}(G)$ denotes the smallest number of colors for which there exists a rich-neighbor edge coloring.

Conjecture 1.3. *For every graph G of maximum degree Δ, $X'_{rn}(G) \leq 2\Delta - 1$ holds.*

Example 1.4. Let's take a look at the Petersen graph and an example of a rich-neighbor edge coloring.



We can see that for the Petersen graph (which is 3-regular) $X'_{rn} \leq 5$. $\diamondsuit$

2. Plan

Our plan is to create an integer program that "proves" the conjecture for (small) regular graphs of degree 4 $\geq$ (it finds a rich-neighbor edge coloring for every k-regular graph on n verticies) and to make a random search algorythm for checking classes of graphs that are too large to be checked individually.

2.1. **Integer Programming.**

Using SageMath we plan to create an integer programming model, that checks all smaller graphs for a rich-neighbor edge coloring with $\leq 2\Delta - 1$ colors. Our interger program looks like this:

minimize t — we minimize the number of colors we need

subject to $\forall e: \quad \sum_{i=1}^{k} x_{ei} = 1$ — each edge is exactly one color

$\forall i \; \forall u \; \forall v, w \sim u, v \neq u: \quad x_{uv,i} + x_{uw,i} \leq 1$ — edges with the same vertice are a different color

$\forall e \; \forall i: \quad x_{ei} \cdot i \leq t$ — we use less or equal to t colors

$\forall i \; \forall uv \; \forall w \sim u, w \neq v \; \forall z \sim v, z \neq u, w: \quad x_{uw,i} + x_{vz,i} + y_{uv} \leq 2$ — uv is a rich edge $\Leftrightarrow$ all adjacent edges are a different color

$\forall e: \quad \sum_{f \sim e} y_f \geq 1$ — every edge is adjacent to some rich edge

$\forall e: \quad 0 \leq y_e \leq 1, \; y_e \in \mathbb{Z}$

$\forall e \; \forall i: \quad 0 \leq x_{ei} \leq 1, \; x_{ei} \in \mathbb{Z},$

where

$$x_{ei} = \begin{cases} 1, & \text{if edge } e \text{ has color } i \\ 0, & \text{otherwise} \end{cases} \quad \text{and} \quad y_e = \begin{cases} 1, & \text{if edge } e \text{ is rich} \\ 0, & \text{otherwise.} \end{cases}$$

We will determine at what point the computation of rich-neighbor edge coloring becomes too intense for this technique and we will then use the random search algorythm.

2.2. **Random Search.**

We will construct a random search algorythm that will check if the conjecture holds for a class of graphs that are too large to be checked individually. Our plan is to create a random graph generator that will generate a regular graph with a given number of verticies and edges. Then we will check if the conjecture holds for the generated graph and we will repeat this process for a given number of times. Then we will repeat this process for different classes of graphs. The algorythm will be implemented in SageMath and its pseudocode is as follows: