BUEM119H7 – Mathematics Essay

L(2,1)-labelling

Lecturer: Steve Noble

Mair Allen-Williams

March 18, 2022

1 Introduction and context

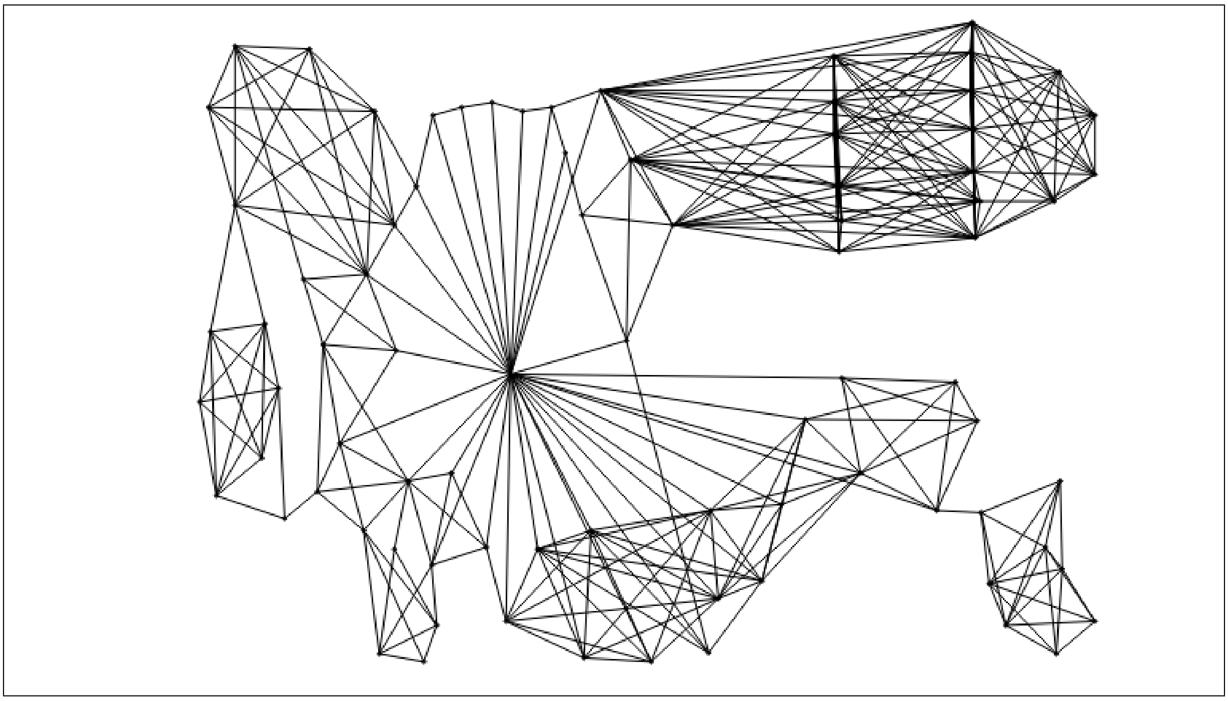


Figure 1: Interference graph provided by the Centre d'ELectronique de l'Armement for the CALMA project (Combinatorial ALgorithms for Military Applications), taken from [1] (Provided under license: CC BY-NC 2.0)

Suppose we have a set $V = \{1, 2, \ldots n\}$ of nodes to which we wish to assign wireless frequency channels, while ensuring that interference between channels is kept to a minimum – in other words, that nearby channels do not have the same or similar frequencies. This "channel assignment problem" motivates a mathematical problem known as L(2,1)-labelling. First introduced by [23] in 1980, the problem was presented by [39] as a graph-colouring problem in 1988. In 1992, Griggs and Yeh [22] published a seminal study into the mathematics of the problem, in which they focused on minimising the range of channels required, and also examined the complexity of the allocation task. Since then, the problem has been extensively studied from both these angles for various graph structures – see [9] for a comprehensive overview.

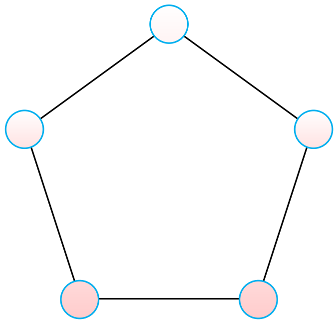
This essay will highlight key results regarding the range of channels needed and sketch their proofs, before briefly examining complexity results, approximate solutions and the current state of the art. We begin with a reminder of graph-theoretic preliminaries.

2 Rudiments of graph theory

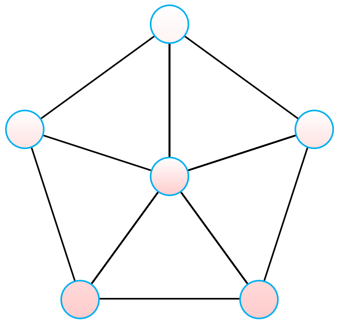
(See any standard graph theory textbook for more information on the concepts in this section, e.g. [7], available online).

We consider a connected, undirected graph $G = (V, E)$; Figure 2 shows some common graph families. The *order* of G is the number of vertices $|V(G)|$ (often written as $|V|$); the *degree* of a vertex $v \in V$ is the number of edges meeting at v; the degree Δ of G is the maximum degree of any vertex of G. The *complement* G^C of G has the same vertices V as G, and an edge between the vertices v_i, v_j if and only if there is no edge $v_i v_j$ in G (Figure 3b). A subset $V_{sub} \subseteq V$ generates an *induced subgraph* of G with vertices V_{sub} and edges $E_{sub} = \{v_i v_j | v_i, v_j \in V_{sub}, v_i v_j \in E\}$. A *path* in a graph is a sequence of unique vertices linked by edges (Figure 4a). The *distance* between two vertices is the number of edges in the shortest path between the vertices; the *diameter* of G is the longest distance between any pair of vertices in G. A *spanning tree* of G is a tree T (see Figure 2) with vertices $V(T) = V(G)$ and edges $E(T) \subseteq E(G)$ (Figure 5). Any connected graph has at least one spanning tree. A graph is *complete* if there is an edge between every pair of vertices; it is *planar* if it can be drawn on a plane without any edges crossing. We limit ourselves to *simple* graphs, which have no loops (edges $v_i \to v_i$) or duplicate edges.

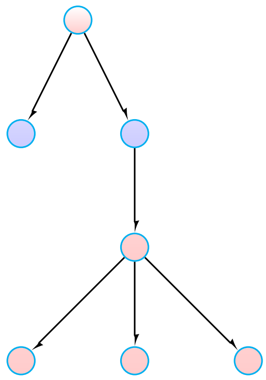
Two concepts used in the proofs below relate to edge adjacency: a *matching* in G is a set M (Figure 6) such that no vertex appears in M more than once; a *maximum matching* contains the largest possible number of edges. A *Hamilton path* is a path visiting each vertex of G exactly once (Figure 7). Not every graph has a Hamilton path; a graph with a Hamilton path is *Hamiltonian*.



(a) A cycle



(b) A wheel



(c) A tree

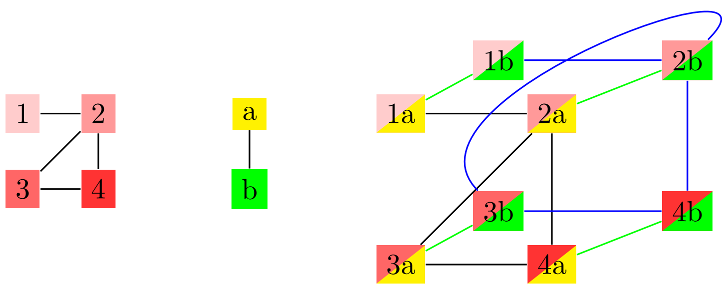
Common graph families include:

a) The *cycle*.

b) A cycle plus a vertex is a *wheel*.

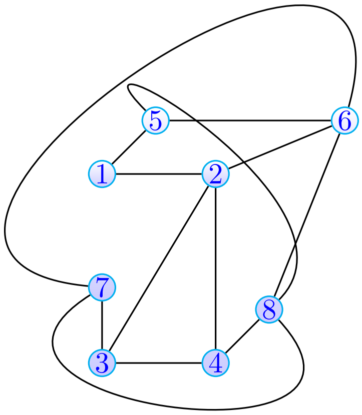
c) The *tree*: a graph in which any two vertices are connected by **exactly one** path.

d) Various *product* families: the product of $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ has vertices $V_p = V_1 \times V_2$; different types of product have different conditions defining the edges.

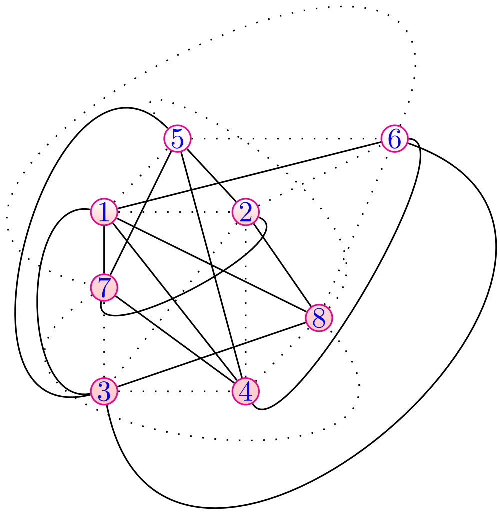


(d) A simple Cartesian product of graphs

Figure 2: Some common families of graph

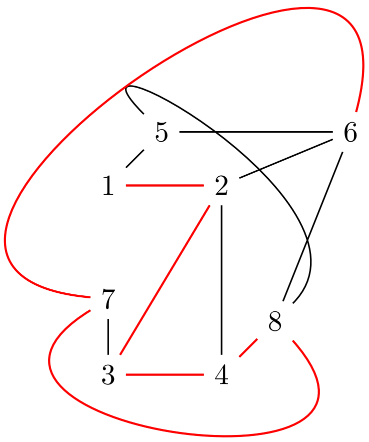


(a) G: A simple example graph



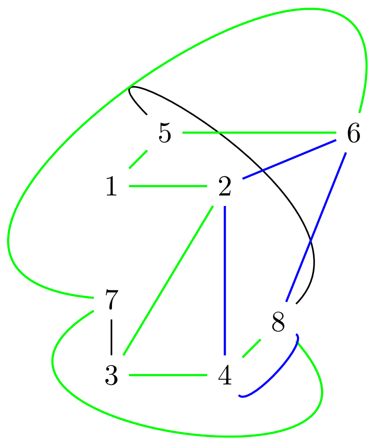
(b) Complement of G

$|V| = 8$, $\Delta(G) = 4$ (v_2, v_6, v_8); $\Delta(G^C) = 5$ (v_1).



(a) A path in G:

$P = \{1, 2, 3, 4, 8, 7, 6\}$



$P_1 = \{1, 2, 3, 4, 8, 7, 6, 5, 1\}$;

$P_2 = \{2, 6, 8, 4, 2\}$

(b) Cycles in G

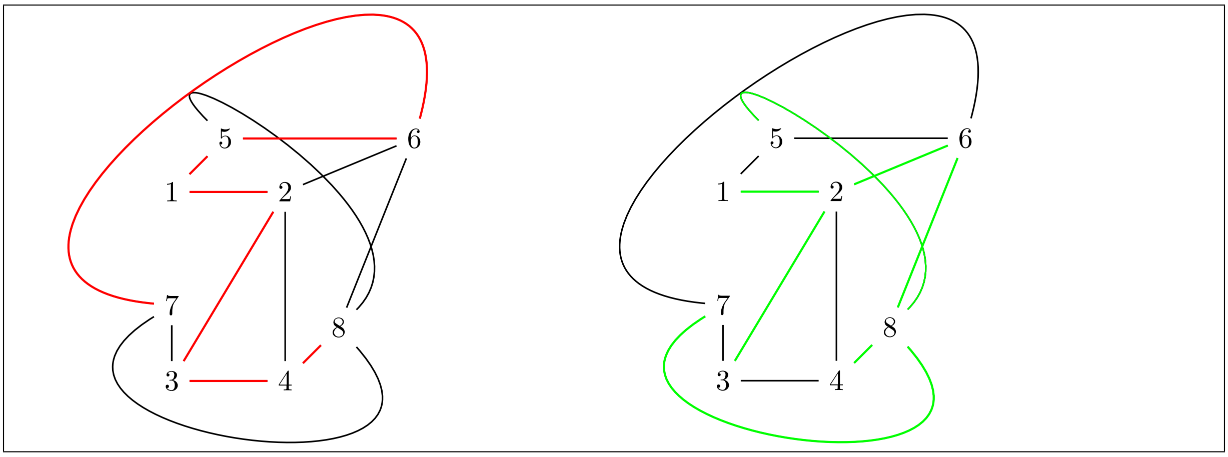
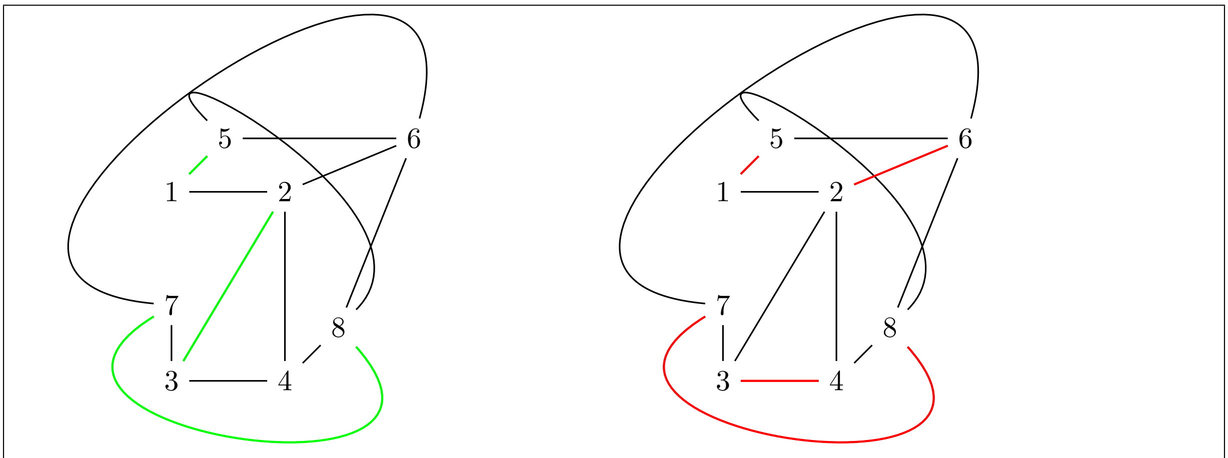


Figure 5: Spanning trees in G



(a) A non-maximum matching in G (b) One of various maximum matchings

Figure 6: Matchings in G

A *labelling* of G is a mapping from V to labels subject to a set of constraints. The best-known labelling problem is *graph colouring*, in which vertices must be assigned colours such that no two neighbouring vertices have the same colour (Figure 8). The *chromatic number* $\chi(G)$ of a graph G is the minimum number of colours needed to colour G.

An L(2,1)-labelling of G – also called a *radiocolouring* or *radio colouring* – is a mapping f from V to integers $\{1, \dots n\}$ such that $|f(v_j) - f(v_i)| \geq 2$ if the vertices v_j and v_i are adjacent, and $|f(v_j) - f(v_i)| \geq 1$ if the vertices v_j and v_i have a common neighbour. The *labelling number* associated with a given labelling $f \mapsto \{1, \dots, n\}$ is n; the *span* λ of the graph is the minimum labelling number over all valid labellings f for that graph[1].

[1] *Caveat lector*: although this definition seems natural as n corresponds to the number of labels needed if there are no gaps, the literature is not consistent: the labelling number is also commonly defined as $\max(f) - \min(f)$, i.e. $n - 1$ above (or n using labels $\{0, \dots, n\}$).

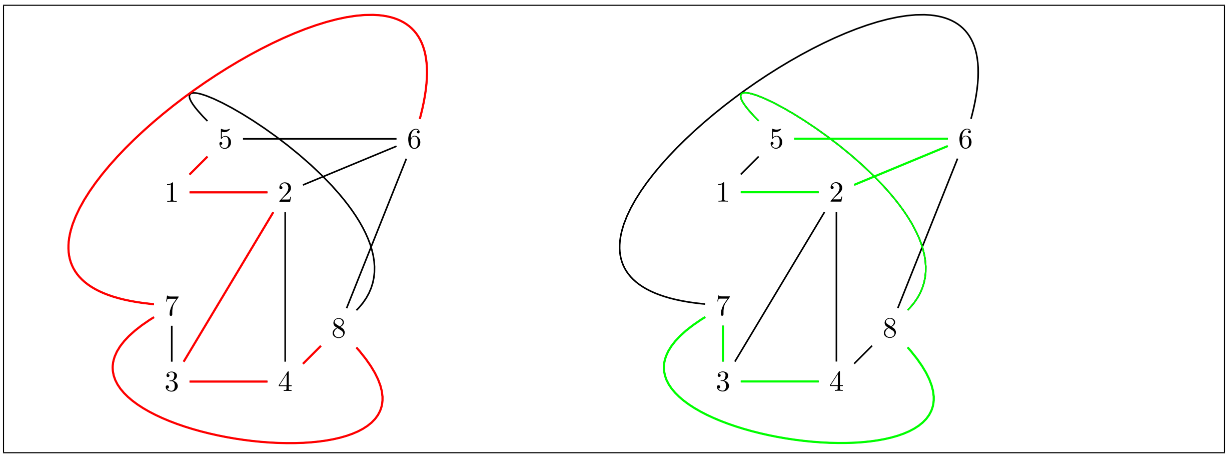
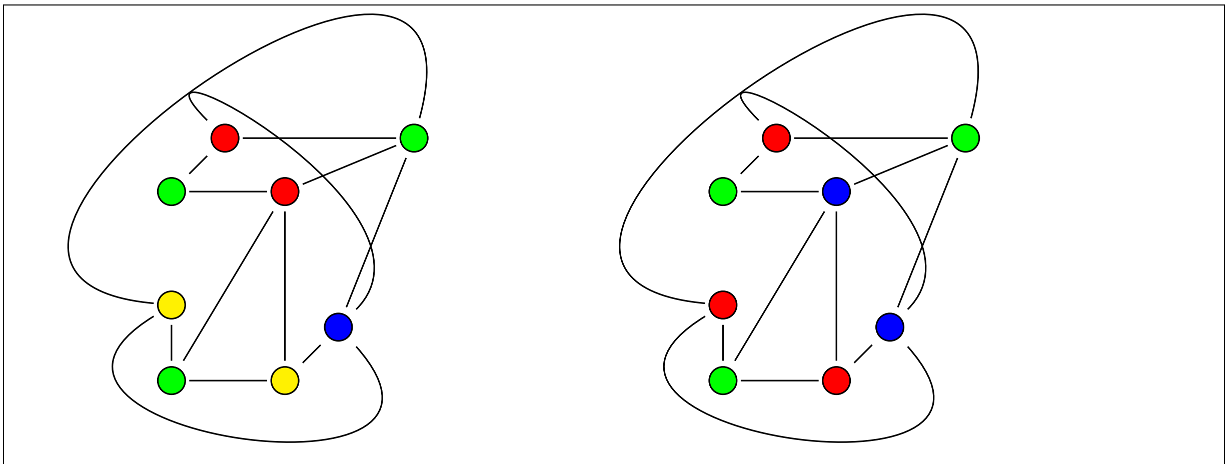
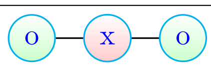


Figure 7: Hamilton paths in G



(a) A 4-colouring of G (b) A 3-colouring of G

Figure 8: Colourings of G



Consider an L(2,1)-labelling of this graph: the two 'o' nodes must have distinct labels. If 'x' has label i, and one 'o' node has label $i-2$, then the other 'o' node must have label $\leq i-3$ or $\geq i+2$. The minimum labelling number is then $\lambda = |i - i - 3| + 1 = 4$, even though there are only three nodes.

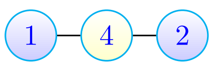
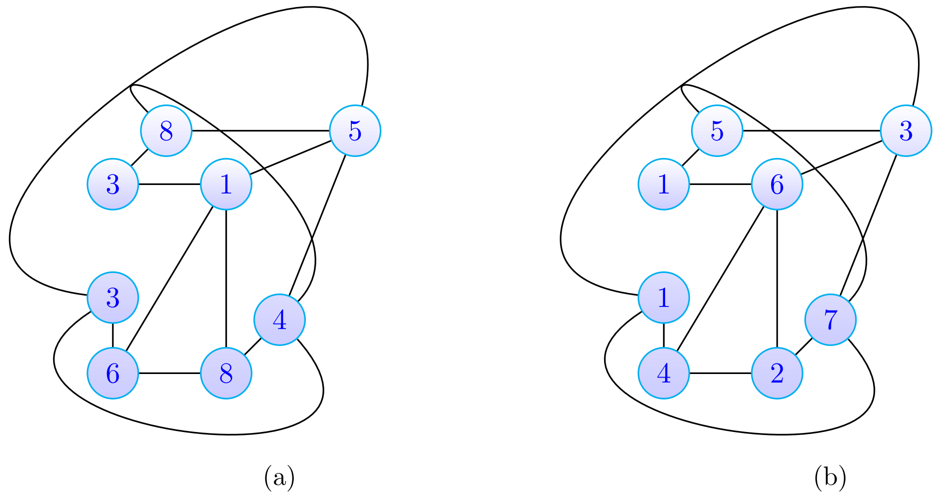


Figure 9: Simple labelling illustration



The difference between labelling number and number of labels: (a) uses a total of 6 labels; the labelling number is 8 and channels 2,7 are unused; in (b), the labelling number is 7 and every channel is used. It is not possible to achieve a labelling number < 8 using just 6 labels for this graph.

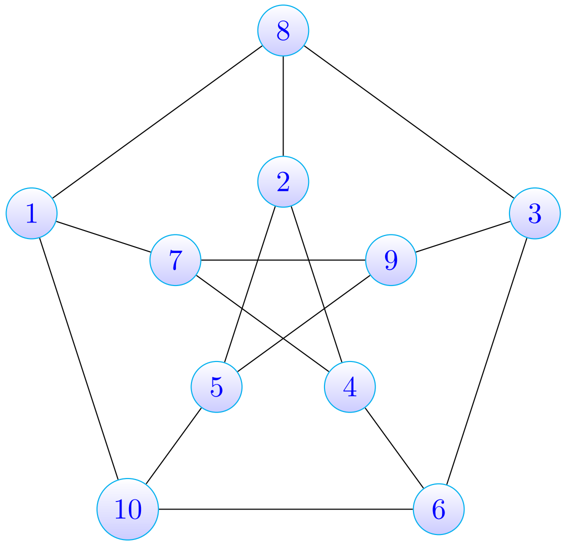
Figure 10: L(2,1)-labellings

It is L(2,1)-labelling that will be our focus in this essay. The next section outlines what mathematicians currently understand about this problem and where research may be headed in the next decade.

3 Algorithms and bounds

Given a graph $G = (V, E)$ to which we wish to assign an L(2,1)-labelling, there are a number of invariants that may be of interest (see e.g. [30]). In traditional graph colouring, we want to minimise the number of colours used. For an L(2,1)-labelling, the focus is generally on minimising the range of labels, i.e. the labelling number, rather than the number of labels (see Figures 9, 10). We might also be interested in efficient ways to assign the labels.

In this section, we discuss the research into upper bounds for the span λ as a function of $\Delta(G)$. Without getting too deep into details, we sketch the ideas of some key proofs to obtain an insight into the problem. We then briefly outline the state of the art regarding the complexity of L(2,1)-labelling, and allude to compromise solutions using heuristics.



The Petersen graph has $\Delta = 3$, so $\Delta^2 = 9$, and is known to have $\lambda = 10$. An example L(2,1)-labelling using 10 labels (given by [4]) is shown.

Figure 11: The Petersen graph

3.1 A bound on λ

It is clear intuitively that the degree $\Delta(G)$ must be relevant to the span of G. Unlike in the classical colouring problem, every neighbour of a vertex v must have a different label, and so as the number of neighbours of v increases, the number of labels required in the neighbourhood of v likewise increases. In their 1992 paper, [22] prove that λ can be directly related to Δ and give a general upper bound on λ of $\Delta^2 + 2\Delta + 1$. They furthermore conjecture that a better bound of $\lambda \leq \Delta^2 + 1$ exists. Certain classes of graph, such as the Petersen graph (Figure 11) are known to have $\lambda = \Delta^2 + 1$ [4], so this would be the minimum possible general bound. [22] also present tighter bounds for certain families of graph, including:

- Any path P_n with n vertices ($n \geq 5$) has $\lambda(P_n) = 5$.
- Any wheel W_n has $\lambda(W_n) = n + 2$.
- Any tree T with degree Δ has $\lambda(T) = \Delta + 2$ or $\lambda(T) = \Delta + 3$.
- For any k-colourable graph, i.e. $\chi(G) = k$, $\lambda(G) \leq |V(G)| + k - 1$.
- Any graph with diameter 2 has $\lambda(G) \leq \Delta^2$.

λ has been established for other special categories of graph over the years. We will look briefly next at one in particular: for a *triangular lattice* L, $\lambda(L) = 9$ [44]. This graph is not hard to label, but the triangular lattice is particularly relevant to the motivating frequency assignment problem, and the proof for $\lambda(L)$ will serve a simple example before we embark on the general case.

3.2 λ for a triangular lattice

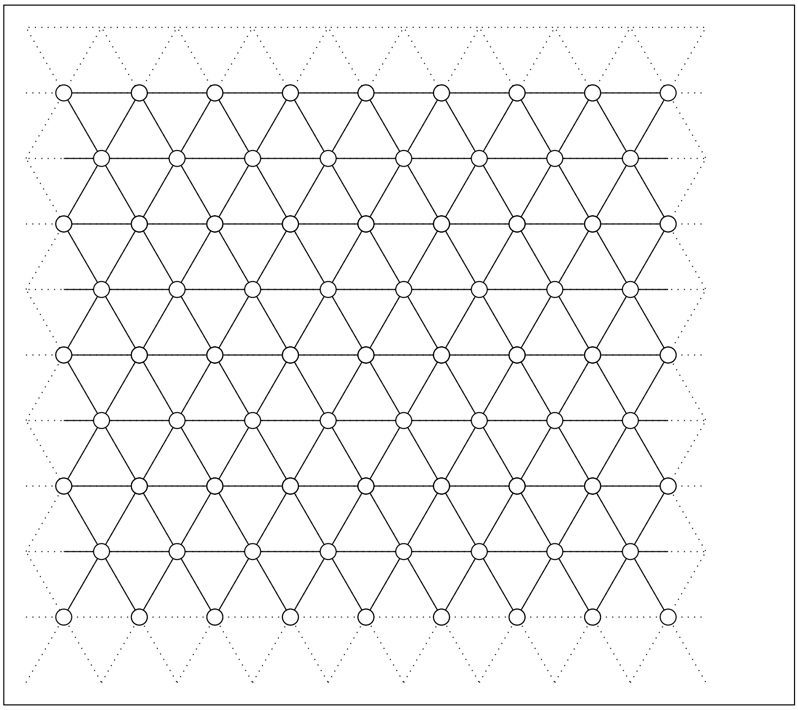


Figure 12: A triangular lattice

The triangular lattice (Figure 12) is of practical interest as it represents an efficient way to distribute transmitters with circular ($\approx$ hexagonal) coverage evenly across a geographical area, represented on a plane. The infinite lattice has a repeating pattern, and can be labelled by considering the small finite subgraph H shown in Figure 13 (due to [44]). To do this, we assign the vertices coordinates i, j as shown in Figure 14a. Then it is easy to verify with simple arithmetic that the function $f : V \to 0, 1, \ldots, 8$ defined by $f(i, j) = (-3i + 2j) \mod 9$ (Figure 14b) is an L(2,1)-labelling and so $\lambda \leq 9$. It is also easy to verify that the subgraph H cannot be labelled with any fewer labels, and so $\lambda = 9$.

3.3 General graphs

The above list of graph families for which bounds on λ have been established is by no means exhaustive. Another important category that has been extensively studied covers various families

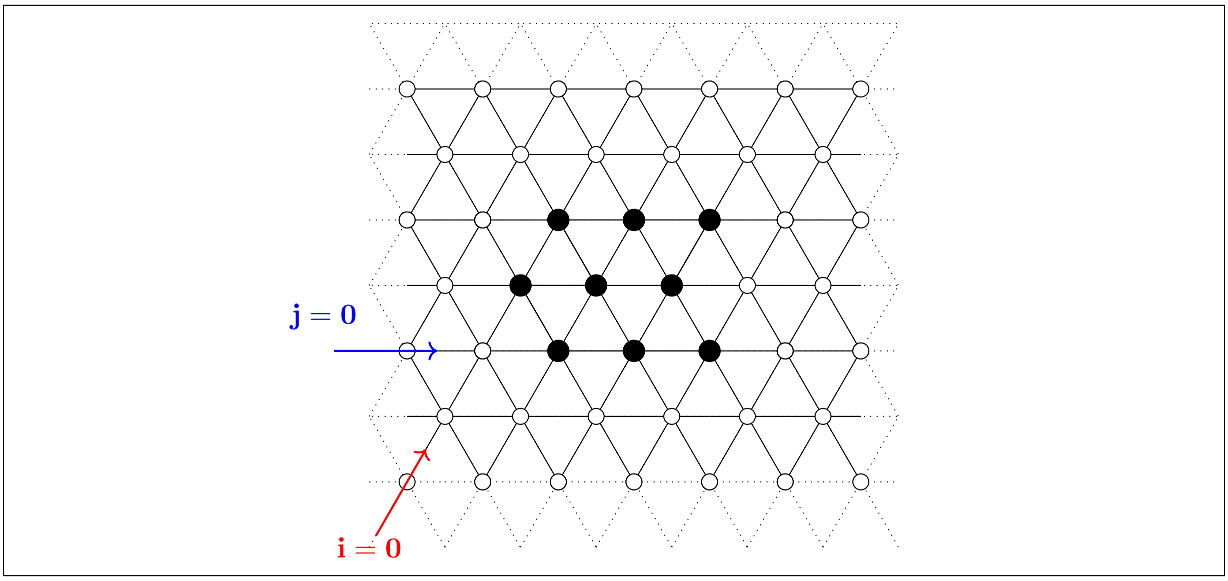
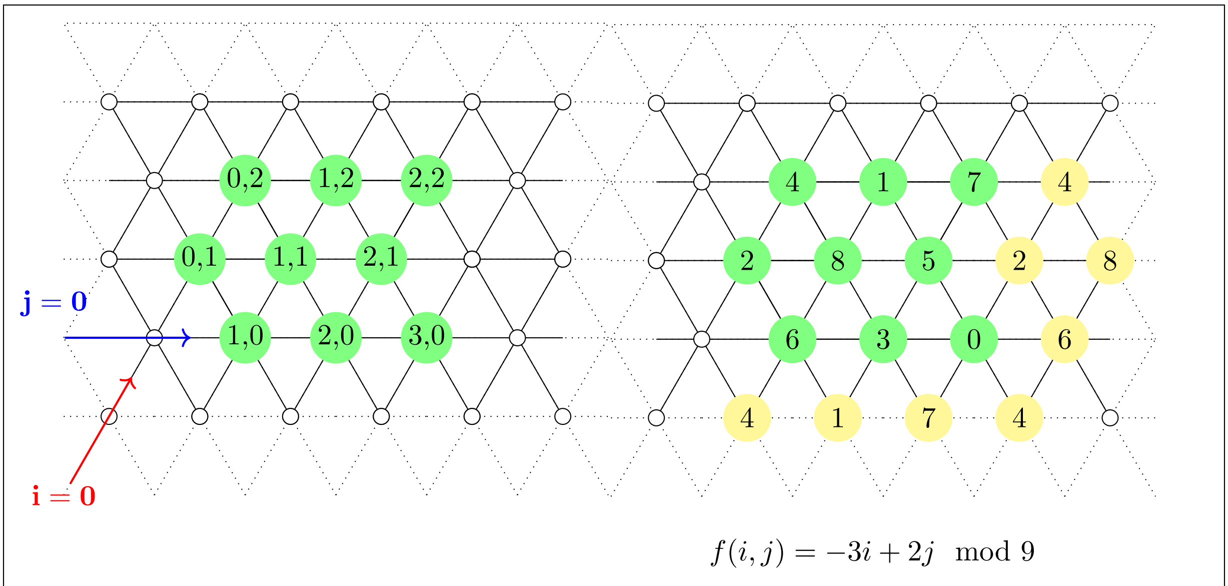


Figure 13: Subgraph H of a triangular lattice – the shaded nodes



(a) Subgraph with coordinates (b) Labelled subgraph

Figure 14: An L(2,1)-labelling for the triangular lattice

of combined graphs: products [34] [33] [28] of certain types of graph, *amalgamations* – graphs created by identifying certain vertices of two graphs [3] [31][2], intersections [13]. However, the conjecture of Griggs and Yeh that $\lambda \leq \Delta^2 + 1$ for *any* simple graph remains unproven.

An initial bound on λ for an arbitrary graph is given by a greedy algorithm [22]: simply assign each vertex in turn (in any order) the smallest label compatible with the already-labelled vertices. A vertex v has at most Δ immediate neighbours, which each rule out three labels (their own label l_n, $l_n - 1$ and $l_n + 1$). Each of these neighbours has at most $\Delta - 1$ neighbours distinct from v, i.e. there are at most $\Delta(\Delta - 1) = \Delta^2 - \Delta$ vertices whose label must be distinct from l_v. In the worst case, therefore, there may be $(\Delta^2 - \Delta) + (3\Delta) = \Delta^2 + 2\Delta$ labels ruled out, and thus after labelling v we may have a labelling number of $\Delta^2 + 2\Delta + 1$.

By 2005, this greedy bound had been whittled down to $\Delta^2 + \Delta - 1$ by Gonçalves [21], who reached back to 1996 and refined an algorithm due to [11] to improve on the 2003 bound of [35]. The proof in [21] is based on a constructive algorithm. The key to the technique is to choose a smart ordering in which to label the vertices, based on a spanning tree of G.

Another step forward followed in 2008, when [26] succeeded in proving the Griggs and Yeh conjecture for "sufficiently large" Δ, using probabilistic tools. Their result is framed as follows:

(Havet et al. [26])

There is some Δ_c such that for every graph G of maximum degree $\Delta \geq \Delta_c$:

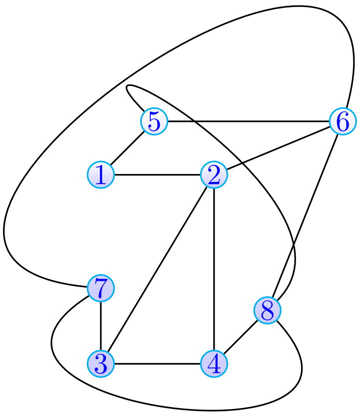
$$\lambda(G) \leq \Delta^2 + 1$$

A corollary of this result is that $\lambda(G) \leq \Delta^2 + C$ for some absolute constant C [25].

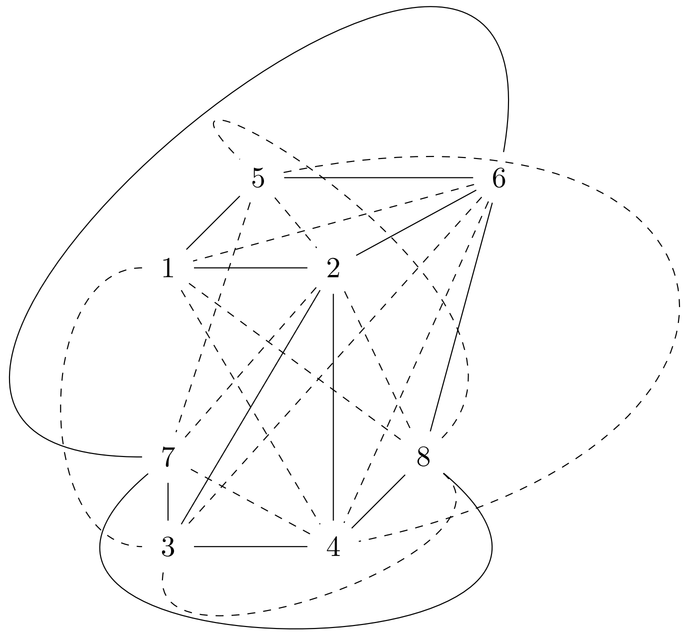
The proof is based on a random trial approach to labelling, subject to certain conditions. Simplifying somewhat: suppose that E_x is the "bad" event that, in some random assignment, the vertex v_x does not have a valid label. The *Lovász local lemma* states that under certain conditions (in particular, each such E_x should depend mainly on a limited number of other "local" variables, not on the overall size of the assignment), there will be a positive (> 0) probability that no such events will occur, i.e. that the random assignment will be a valid labelling. This lemma has been used for other graph-colouring proofs (see e.g. [16] [20]), but its application in [26] is complex and relies on additional probabilistic results: the details are beyond the scope of this essay.

However, we will sketch an outline of the setup. The first step is to identify the *square* G^2 of G: this graph has the same vertices as G but additional edges between any two nodes that

[2]NB: The precise definition of amalgamation used in these papers is the authors' own.



(a) Our example graph G



(b) G^2 – new edges are shown as dashed

Figure 15: G^2 for our simple example graph

share a neighbour in G (see Figure 15). The maximum degree of any vertex in G^2 is thus the familiar value $\Delta(\Delta - 1) = \Delta^2 - \Delta$.

Next, [26] use a decomposition of the vertices of G^2 into disjoint sets with certain properties – for example, all but one of the sets must have between $\Delta^2 - 8000\Delta$ and $\Delta^2 + 4000\Delta$ vertices – to construct three disjoint sets of vertices V_1, V_2, V_3 which are labelled in turn. At each stage, the Lovász local lemma is applied (in conjunction with other probabilistic machinery) to prove that a suitable labelling must exist (the labelling for V_1 and V_2 is subject to additional constraints designed to ensure that the conditions of the lemma are met as the labelling is extended from V_1 to $V_1 \cup V_2$ and then to the whole graph).

The exact requirements for the decomposition and sub-labellings are designed to ensure that the probability calculations work out, and involve considerable counting of neighbours and edges. The large numbers used rule out a proper worked example – for one thing, over 1000 edges are needed for any decomposition satisfying the required properties. However, to get a feel for the procedure, Appendix A steps through the main parts of the process to label an example graph, albeit with an arbitrary decomposition. Highlights to note: the first set of vertices V_1 is derived from G^2, but disconnected pairs of vertices in each local decomposition are merged and labelled as one. Other than these merged pairs, only the most-connected vertices from G

are retained in this first derived graph, so the strategy has echoes of a greedy approach.

Although [26] certainly represents a step towards proving the Griggs and Yeh conjecture, the proof is complex and requires somewhat opaque constructions to make the numbers work; furthermore, the suggested value of $\Delta \approx 10^{69}$ is outlandishly large.

Working "from the other direction", in 2015 Franks [18] succeeded in proving the conjecture for graphs of "small order"; namely,

> **(Franks [18])**
> $\lambda(G) \leq \Delta^2 + 1$ if
> $$|V| \leq \left(\left\lfloor \frac{\Delta}{2} \right\rfloor + 1\right)(\Delta^2 - \Delta + 1) - 1$$
> .

This is actually a corollary of [18]'s main theorem, which states that given some $L \geq \Delta^2 + 1$, $\lambda(G) \leq L$ if

$$|V| \leq (L - \Delta)\left(\left\lfloor \frac{L-1}{2\Delta} \right\rfloor + 1\right) - 1. \tag{1}$$

The corollary follows by choosing $L = \Delta^2 + 1$. The proof is anchored in a series of graph transformations which generate a valid labelling, as follows:

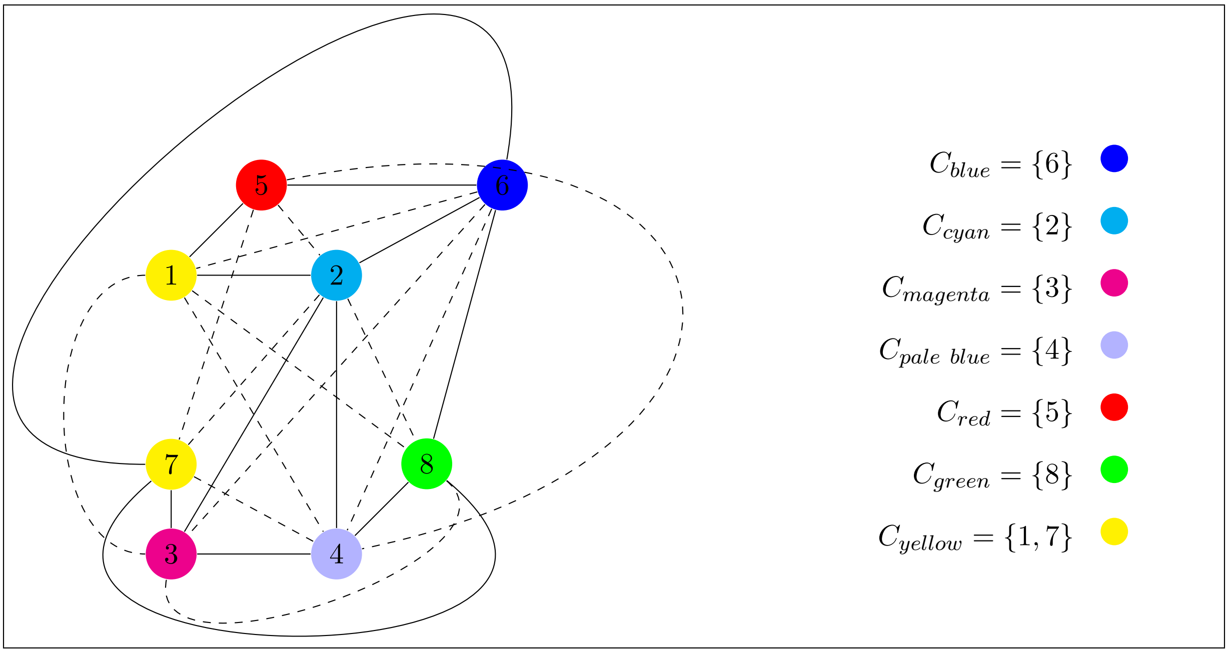


Figure 16: A 7-colouring of G^2

Given a graph G, Franks begins, like [26] above, by computing the square graph G^2 (Figure 15). However, Franks' next step is to find a standard colouring of G^2. Now let C_i be the set of vertices coloured with colour i (the C_i thus partition V). Figure 16 illustrates this on our example graph. Next, Franks introduces a novel concept: the *square colour graph* $\mathcal{G}$ has the C_i as vertices and an edge C_iC_j if the induced subgraph of G with vertices $C_i \cup C_j$ contains an edge of G (Figure 17). Finally, $\mathcal{G}^C$ is the complement of $\mathcal{G}$ (Figure 17b).

Franks then shows that if $\mathcal{G}^C$ is Hamiltonian with Hamilton path $P = \{p_0, p_1, \ldots, p_n\}$, an L(2,1)-labelling for the original G can be constructed (Figure 18a) by giving the vertex $v \in G$ the label i if $p_i \in P$ contains v (recall the p_i are simply an ordering of the C_i, which partition V): since C is a colouring of G^2, vertices in different sets C_i, C_j are exactly those whose labels must differ by at least one. If v_i and v_j are neighbours in G, there will also be an edge C_iC_j in $\mathcal{G}$ and accordingly no edge in $\mathcal{G}^C$, so C_i and C_j cannot be adjacent in P; the corresponding labels must therefore differ by at least two. Figure 18b shows the corresponding labelling of our example graph.

Since this construction relies on the existence of a Hamilton path in $\mathcal{G}^C$, the proof [18] shows (using a well-known result due to Pósa; we omit the details) that such a path always exists under Condition (1), provided the k-colouring of G^2 is *equitable* (i.e. the sizes of the C_i as defined above differ by at most one – our 7-colouring in Figure 16 is equitable with $|C_i| \in \{1, 2\}$). A result of Szemerédi-Hajnal states that if the degree of a graph $G \leq L$ for some L, then G can be equitably coloured with $L + 1$ colours; since the degree of $G^2 \leq \Delta^2 - \Delta$, an equitable colouring of G^2 with $L = \Delta^2 + 1 \geq \Delta^2 - \Delta$ colours must exist, and this completes the proof.

A comparison between the rather involved proof [26] for "sufficiently large" graphs and the lean proof [18] for "sufficiently small" graphs is interesting: both start from G^2, but [18] subsequently transforms the labelling stage to a standard colouring problem – albeit with the "equitable" requirement. While [26] use the complement of G^2 to identify "locally matched" vertices, [18] only takes the complement at the square colour graph stage. Indeed, by contrast with the elegant "square colour graph", the tools used by [26] are rather heavy machinery.

Another difference is that while the proof of [18] yields an algorithm for assigning the labels (more on this below), [26] is merely an existence proof: the Lovász local lemma does not provide a constructive solution approach. In the next section we examine further this issue of assigning the labels.

$C_{blue} = \{6\}$

$C_{cyan} = \{2\}$

$C_{magenta} = \{3\}$

$C_{pale\ blue} = \{4\}$

$C_{red} = \{5\}$

$C_{green} = \{8\}$

$C_{yellow} = \{1, 7\}$

$C_{b,c} = \{6, 2\}$ $E = \{v_6v_2\}$

$C_{b,m} = \{6, 3\}$ $E = \{\}$

$C_{b,p} = \{6, 4\}$ $E = \{\}$

$C_{b,r} = \{6, 5\}$ $E = \{v_6v_5\}$

$C_{b,g} = \{6, 8\}$ $E = \{v_6v_8\}$

$C_{b,y} = \{1, 6, 7\}$ $E = \{v_6v_7\}$

$C_{c,m} = \{2, 3\}$ $E = \{v_2v_3\}$

$C_{c,p} = \{2, 4\}$ $E = \{v_2v_4\}$

$C_{c,r} = \{2, 5\}$ $E = \{\}$

$C_{c,g} = \{2, 8\}$ $E = \{\}$

$C_{c,y} = \{1, 2, 7\}$ $E = \{v_1v_2\}$

$C_{m,p} = \{3, 4\}$ $E = \{v_3v_4\}$

$C_{m,r} = \{3, 5\}$ $E = \{\}$

$C_{m,g} = \{3, 8\}$ $E = \{\}$

$C_{m,y} = \{1, 3, 7\}$ $E = \{v_3v_7\}$

$C_{p,r} = \{4, 5\}$ $E = \{\}$

$C_{p,g} = \{4, 8\}$ $E = \{v_4v_8\}$

$C_{p,y} = \{1, 4, 7\}$ $E = \{\}$

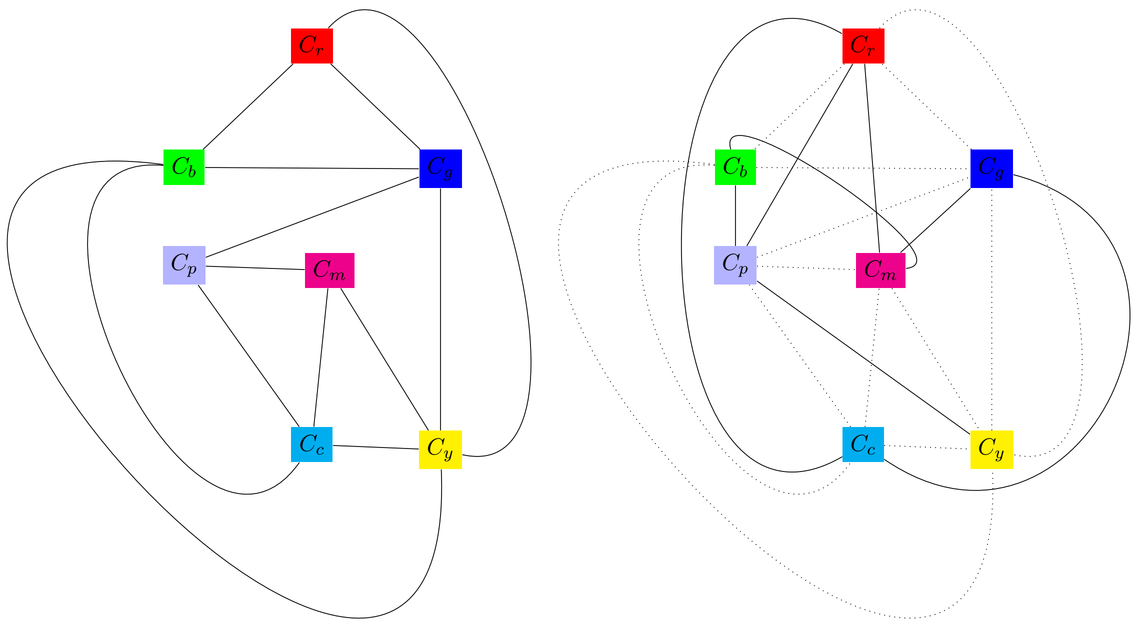
$C_{r,g} = \{5, 8\}$ $E = \{v_5v_8\}$

$C_{r,y} = \{1, 5, 7\}$ $E = \{v_1v_5\}$

$C_{g,y} = \{1, 7, 8\}$ $E = \{v_7v_8\}$

Edges in $\mathcal{G}$ correspond to the (13) non-empty sets:

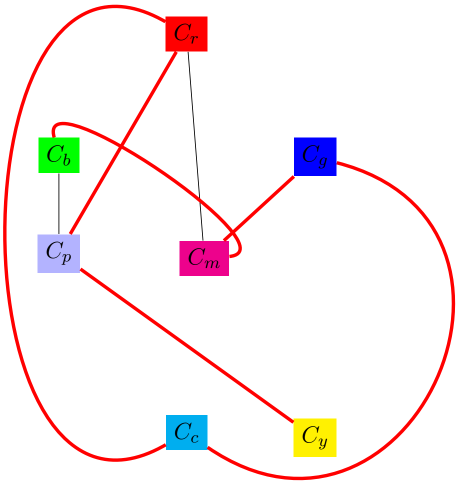
$C_bC_c, C_bC_r, C_bC_g, C_bC_y, C_cC_m, C_cC_p, C_cC_y, C_mC_p, C_mC_y, C_pC_g, C_rC_g, C_rC_y, C_gC_y$



(a) Square colour graph $\mathcal{G}$

(b) $\mathcal{G}^C$ (solid edges)

Figure 17: Square colour graph $\mathcal{G}$ and its complement $\mathcal{G}^C$ for our 7-colouring of G



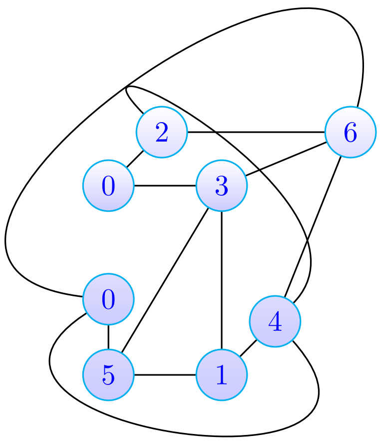
$P = C_y \to C_p \to C_r \to C_c \to C_g \to C_m \to C_b$

(a) Hamilton path in $\mathcal{G}^C$

p_0	p_1	p_2	p_3	p_4	p_5	p_6
C_y	C_p	C_r	C_c	C_g	C_m	C_b

This Hamilton path induces the following L(2,1)-labelling:

vertex v	1	2	3	4	5	6	7	8
$v \in C_i$	C_y	C_c	C_m	C_p	C_r	C_b	C_y	C_g
$j : P_j = C_i$	0	3	5	1	2	6	0	4



Notice that this is distinct from the labelling in Figure 10b, although the pair of vertices that share a label is the same.

(b) G labelled using Franks' method [18]

Figure 18: Hamilton path in $\mathcal{G}^C$ and the induced labelling of G

3.4 Complexity and algorithms

The *complexity*, or $\mathcal{O}$-number, of a problem is an order of magnitude for the number of steps it takes to find a solution, as a function of the problem parameters – in this case, $|V(G)|$ and $|E(G)|$. For example, the algorithm in §3.2 for the triangular lattice is *linear* in $|V|$, since we require one step per vertex. Efficient algorithms have been found for other specific types of graph [6], including a linear time algorithm for trees [24]. The most time-consuming stage of the square colour graph technique in [18] is finding an equitable k-colouring of the square graph; it is known that this can be done in $\mathcal{O}(k|V|^2)$ time [32]; subsequently finding a Hamilton path can also be done in polynomial time under [18]'s conditions [8], so there is a polynomial-time algorithm for graphs satisfying Condition (1).

In general, however, the L(2,1)-labelling problem is known to be *NP-complete* [36]: it is possible to check in polynomial time whether a candidate labelling f of a graph G is an L(2,1)-labelling, but finding a labelling may require exponential time. The brute force approach takes some $\mathcal{O}(k^{|V|})$ steps: simply test all possible candidate labellings using labels $\{1, \ldots, k\}$. Since each of the $|V|$ vertices can be assigned any one of the k labels, there are $k^{|V|}$ candidates, each requiring polynomial time to check. Furthermore, to find an optimum k, we would have to start with an implausibly low value for k, exhaust all possible assignments, increment k by 1, and repeat until hitting on a solution. Indeed, [17] show that for a general graph, and $k \leq k_0$ for some k_0, deciding whether there is an L(2,1)-labelling of G using k labels is itself an NP-complete problem. Although this means that any exact labelling algorithm for an arbitrary graph must be exponential, i.e. $\mathcal{O}(c^n)$, we can do much better than the $c = k$ of the brute force approach. [27] explore the problem and bring c to a little above 3. This was reduced again in 2013 to the current record of $c = 2.6488$ [29]; we are not aware of any further breakthroughs in the subsequent nine years.

Suppose, however, that we relax the requirement for the "best" labelling, and simply look for "good" solutions. A *heuristic algorithm* seeks "near-optimal" solutions to an otherwise intractable problem. A simple heuristic for L(2,1)-labelling might assign labels greedily to vertices ordered by decreasing degree [38]. Recall that the proof in [26] relied on the Lovász local lemma, a result using the fact that dependence between labels is largely *local*. Good results are often found by combining *local optimisation* with an element of randomisation, frequently imitating natural processes – for example, genetic algorithms mimic "survival of the fittest"; artificial bee colonies exploit "swarm intelligence" [10]. Tools which have been explored for

L(2,1)-labelling include genetic algorithms [40] [14], simulated annealing [37] and self-organising systems [15]. [1] has an overview. These algorithms can only ever be as good as the heuristic chosen; in other words, the more we can learn and prove about the relationships between substructures of a graph and locally good labellings, the better we can make our approximate solutions.

Many open questions remain: What is the best upper bound we can achieve for the labelling number while guaranteeing, say, polynomial time? What are the most effective heuristics? And what is the best way to assign frequencies in a real network – such as Figure 1 – today? In the next section, we revisit the practical problem that prompted the mathematical exploration before summarising the results covered in this essay and highlighting key open problems.

4 Perspectives and conclusion

4.1 Channel assignment in the digital age

The L(2,1)-labelling problem was initially motivated by a real-world problem posed in the 1980s. Forty years on, the core problem of assigning frequencies with minimum interference remains current, but communication technology has changed. In today's mobile world, frequency assignments must often be carried out dynamically, rapidly, and adaptively for moving devices [1] [5]. The channel assignment problem has thus largely become the domain of computer science, fast approximate algorithms, hybrid approaches and parallel computing [12] [41] [15] [1].

Meanwhile, mathematicians continue to study L(2,1)-labelling for its theoretical interest, seeking new insights, new tools, and new ways of looking at the problem that could lead to a final proof of the Griggs and Yeh conjecture. It is fascinating to realise how many different mathematical tools have been brought to bear on this problem. As well as classical graph-theoretic concepts, we find probability lemmas [26], coding theory (an approach for labelling n-cubes introduced by [42] and exploited by [19]), etc. What seams remain yet to be mined? A recent preprint [2] concerning the Lovász local lemma may harbour possibilities for improving on [26]: the authors of [2] introduce a powerful hierarchy of related lemmata that can be explored with relatively little computational effort. Indeed, perhaps modern computing power could have more roles to play here? The map-colouring theorem ("every planar graph can be coloured using no more than four colours") was proved by breaking down the infinity of possible graphs into a few thousand configurations, which were checked by computer [43]. Could something similar be done for L(2,1)-labelling? In the next section, we sum up the current state of the art and highlight some open problems.

4.2 Summary and open problems

We introduced L(2,1)-labelling as a graph-colouring problem motivated by a need to assign radio channels to nodes in a geographical network. We defined the span λ of a graph as a key parameter of interest and outlined the current state of the art regarding an upper bound for λ in a few specific families of graph and in general graphs. We then looked briefly at the complexity of finding a good L(2,1)-labelling and alluded to heuristic algorithms. We noted that the intersection of different areas of mathematics makes L(2,1)-labelling particularly interesting and suggested that there may be new insights to be found by using tools from other fields to illuminate the problem from fresh angles.

Certainly L(2,1)-labelling still harbours plenty of open problems: many concern bounds on λ for specific categories of graph. Furthermore, although Δ^2+1 is known to be a tight bound on λ for certain categories of graph such as the Petersen graph, it is not known how many such graphs exist: categorising all graphs with $\lambda \geq \Delta^2+1$ remains an open problem. Other open problems relate to the complexity: efficiently determining the span λ of a given graph, efficiently finding a L(2,1)-labelling with a given labelling number $\geq \lambda$, and finding good approximate solutions. The most tantalising open problem, however, must remain the 1992 conjecture of Griggs and Yeh that $\lambda \leq \Delta^2+1$.

References

[1] Karen I. Aardal et al. "Models and solution techniques for frequency assignment problems". In: *Annals of Operations Research* 153 (2007), pp. 79–129.

[2] Dimitris Achlioptas and Kostas Zampetakis. *The Lovász Local Lemma is Not About Probability*. Preprint. [Accessed 2022-03-15]. Nov. 16, 2021. arXiv: https://arxiv.org/pdf/2111.08837.pdf.

[3] Sarah Spence Adams et al. "On the L(2,1)-labelings of amalgamations of graphs". In: *Discrete Applied Mathematics* 161.7 (2013), pp. 881–888.

[4] Sarah Spence Adams et al. "The minimum span of L(2,1)-labelings of certain generalized Petersen graphs". In: *Discrete Applied Mathematics* 155.10 (2007), pp. 1314–1325.

[5] Goutam K. Audhya et al. “A survey on the channel assignment problem in wireless networks”. In: *Wireless Communications and Mobile Computing* 11.5 (2011), pp. 583–609. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/wcm.898.

[6] Hans Bodlaender et al. “Approximations for λ-Coloring of Graphs”. In: *Lecture Notes in Computer Science* 1770 (Aug. 2007), pp. 395–406.

[7] J. A. Bondy and U. S. R. Murty. *Graph Theory with Applications*. Available online at http://www.maths.lse.ac.uk/Personal/jozef/LTCC/Graph_Theory_Bondy_Murty.pdf [Accessed 2022-03-14]. New York: Elsevier, 1976.

[8] J.A. Bondy and V. Chvátal. “A method in graph theory”. In: *Discrete Mathematics* 15.2 (1976), pp. 111–135.

[9] Tiziana Calamoneri. “The L(h, k)-Labelling Problem: An Updated Survey and Annotated Bibliography”. In: *The Computer Journal* 54.8 (May 2011), pp. 1344–1371. eprint: https://academic.oup.com/comjnl/article-pdf/54/8/1344/1101365/bxr037.pdf.

[10] Vahid Chahkandi and Omid Mirzaei. “Vertex coloring based on artificial bee colony algorithm”. In: *2015 International Congress on Technology, Communication and Knowledge (ICTCK)* (2015), pp. 312–317.

[11] Gerard J. Chang and David Kuo. “The L(2,1)-Labeling Problem on Graphs”. In: *SIAM Journal on Discrete Mathematics* 9.2 (1996), pp. 309–316. eprint: https://doi.org/10.1137/S0895480193245339.

[12] Jose M. Chaves-González et al. “Optimizing a realistic large-scale frequency assignment problem using a new parallel evolutionary approach”. In: *Engineering Optimization* 43 (Aug. 2011), pp. 813–842.

[13] Joanna Chybowska-Sokół, Konstanty Junosza-Szaniawski, and Paweł Rza żewski. “L(2,1)-labeling of disk intersection graphs”. In: *Discrete Applied Mathematics* 277 (2020), pp. 71–81.

[14] Christine Crisan and Heinz Mühlenbein. “The Breeder Genetic Algorithm for frequency assignment”. In: *Parallel Problem Solving from Nature — PPSN V*. Ed. by Agoston E. Eiben et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 897–906.

[15] Jamal Elhachimi and Zouhair Guenoun. “Distributed Frequency Assignment Using Hierarchical Cooperative Multi-Agent System”. In: *International Journal of Communications, Network and System Sciences* 4.11 (2011), pp. 727–734.

[16] András Faragó. “A Meeting Point of Probability, Graphs, and Algorithms: The Lovász Local Lemma and Related Results — A Survey”. In: *Algorithms* 14.12 (2021).

[17] Jirí Fiala, Ton Kloks, and Jan Kratochvíl. “Fixed-parameter complexity of λ-labelings”. In: *Discret. Appl. Math.* 113 (2001), pp. 59–72.

[18] Cole Franks. “The Δ^2 conjecture holds for graphs of small order”. In: *Involve, a Journal of Mathematics* 8.4 (4 2015), pp. 541–549.

[19] Ophir Frieder, Frank Harary, and Pengjun Wan. “A Radio Coloring of a Hypercube”. In: *International Journal of Computer Mathematics* 79 (2002), pp. 665–670.

[20] Ioannis Giotis et al. “Acyclic edge coloring through the Lovász Local Lemma”. In: *Theor. Comput. Sci.* 665 (2017), pp. 40–50.

[21] D. Gonçalves. “On the L(p,1)-labelling of graphs”. In: *Discrete Mathematics* 308.8 (2008). Third European Conference on Combinatorics, pp. 1405–1414.

[22] Jerrold R. Griggs and Roger K. Yeh. “Labelling Graphs with a Condition at Distance 2”. In: *SIAM Journal on Discrete Mathematics* 5.4 (1992), pp. 586–595. eprint: https://doi.org/10.1137/0405048.

[23] W.K. Hale. “Frequency assignment: Theory and applications”. In: *Proceedings of the IEEE* 68.12 (1980), pp. 1497–1514.

[24] Toru Hasunuma et al. “A linear time algorithm for L(2,1)-labeling of trees”. In: *CoRR* abs/0810.0906 (2008). arXiv: 0810.0906.

[25] Frédéric Havet, Bruce Reed, and Jean-Sébastien Sereni. “Griggs and Yeh’s Conjecture and L(p,1)-labelings”. In: *SIAM Journal on Discrete Mathematics* 26 (Feb. 2012), pp. 145–168.

[26] Frédéric Havet, Bruce Reed, and Jean-Sébastien Sereni. “L(2,1)-labelling of graphs”. In: *ACM-SIAM symposium on Discrete algorithms (SODA 2008)*. San Francisco, California, United States, Jan. 2008, pp. 621–630.

[27] Frédéric Havet et al. “Exact Algorithms for L(2,1)-Labeling of Graphs”. In: *Algorithmica* 59.2 (2011), pp. 169–194.

[28] Pranava K. Jha, Sandi Klavžar, and Aleksander Vesel. “L(2,1)-labeling of direct product of paths and cycles”. In: *Discrete Applied Mathematics* 145.2 (2005). Structural Decompositions, Width Parameters, and Graph Labelings, pp. 317–325.

[29] Konstanty Junosza-Szaniawski et al. "Fast exact algorithm for L(2,1)-labeling of graphs". English. In: *Theoretical Computer Science* 505.Complete (2013), pp. 42–54.

[30] R. Kalfakakou et al. "Graph radio coloring concepts". In: *Yugoslav Journal of Operations Research* 13 (Jan. 2003).

[31] Nathaniel Karst et al. "The minimum span of L(2,1)-labelings of generalized flowers". In: *Discrete Applied Mathematics* 181 (2015), pp. 139–151.

[32] Hal A. Kierstead et al. "A fast algorithm for equitable coloring". In: *Combinatorica* 30 (2010), pp. 217–224.

[33] Byeong Moon Kim et al. "NEW L(j,k)-LABELINGS FOR DIRECT PRODUCTS OF COMPLETE GRAPHS". In: *Taiwanese Journal of Mathematics* 18.3 (2014), pp. 793–807.

[34] Byeong Kim, Byung Song, and Yoomi Rho. "The λ-number of the Cartesian product of a complete graph and a cycle". In: *Korean Journal of Mathematics* 21 (June 2013).

[35] Daniel Kráĺ and Riste Skrekovski. "A Theorem about the Channel Assignment Problem". In: *SIAM Journal on Discrete Mathematics* 16.3 (2003), pp. 426–437. eprint: https://doi.org/10.1137/S0895480101399449.

[36] E.L. Lloyd and S. Ramanathan. "On the complexity of distance-2 coloring". In: *Proceedings ICCI '92: Fourth International Conference on Computing and Information*. 1992, pp. 71–74.

[37] Rudolf Mathar and Jürgen Mattfeldt. "Channel assignment in cellular radio networks". In: *IEEE Transactions on Vehicular Technology* 42 (1993), pp. 647–656.

[38] Bhawani Sankar Panda and Preeti Goel. "Heuristic Algorithms for the L(2, 1)-Labeling Problem". In: *SEMCCO*. 2010.

[39] F. S. Roberts. *From garbage to rainbows: generalizations of graph coloring and their applications*. RUTCOR Research Report. Rutgers University, 1988, pp. 36–88.

[40] Sattar B. Sadkhan. "A Proposed Genetic Based Method to Solve Frequency Assignment Problem for HF Band". In: *2019 4th Scientific International Conference Najaf (SICN)*. 2019, pp. 48–53.

[41] Z. Tabi et al. “Quantum Optimization for the Graph Coloring Problem with Space-Efficient Embedding”. In: *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*. Los Alamitos, CA, USA: IEEE Computer Society, Oct. 2020, pp. 56–62.

[42] Marshall A. Whittlesey, John P. Georges, and David W. Mauro. “On the λ-number of Q_n and Related Graphs”. In: *SIAM J. Discret. Math.* 8 (1995), pp. 499–506.

[43] Robin Wilson. *Four colours suffice: how the Map Problem was solved.* London, UK: Allen Lane, 2003.

[44] Roger K. Yeh. “The edge span of distance two labellings of graphs”. In: *Taiwanese Journal of Mathematics* 4.4 (2000), pp. 675–683.

A The Havet et al. construction

A worked example of the [26] labelling process, using the graph in Figure 19. The setup requires a decomposition $V = D_0, \ldots, D_n, S$. We take $D_0 = \{1, \ldots, 8\}$, $D_1 = \{9, \ldots, 13\}$, $D_2 = \{14, \ldots, 20\}$, $D_3 = \{21, 22, 23\}$ and leave S empty for this example. As discussed in the text, this decomposition does *not* satisfy the required properties; we merely wish to get a feel for the process.

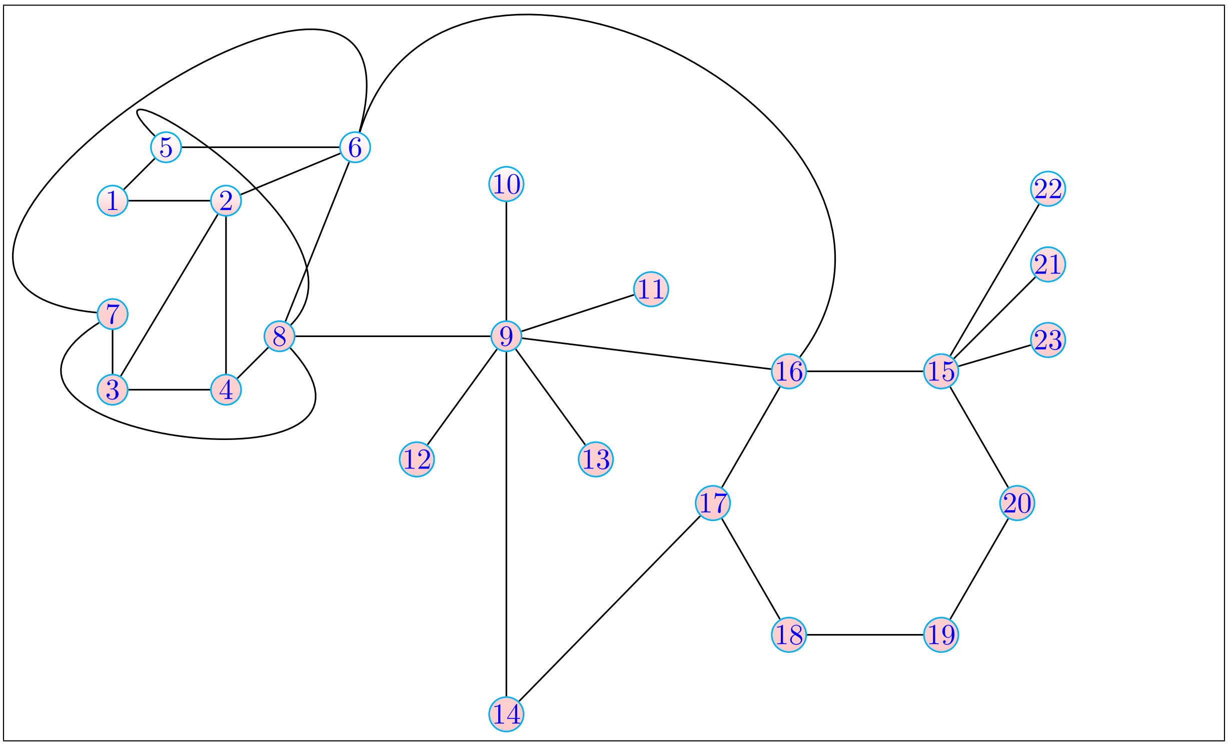


Figure 19: Extended example graph

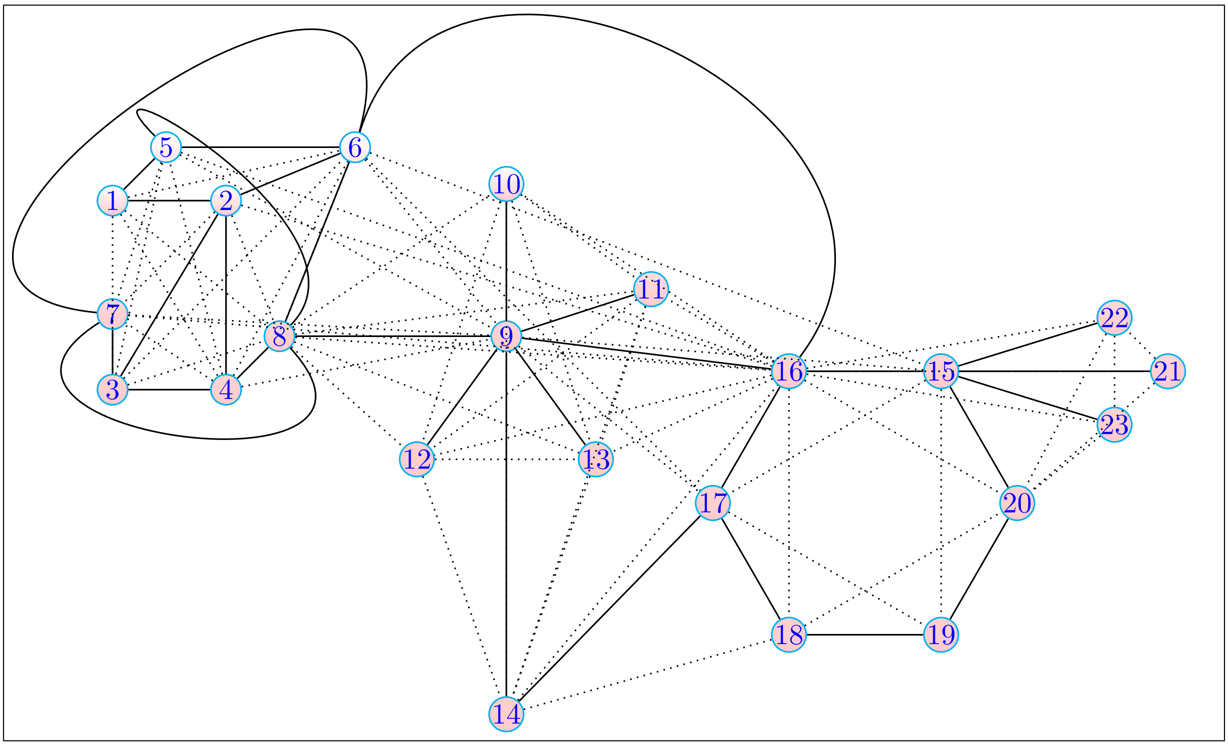


Figure 20: G^2 for our extended example: new edges are shown as dotted

---

Figure 21 Havet et al. construction, first stage

The first stage involves the individual D_i:

- Let H_i be the subgraph of G^2 induced by D_i (Figure 22a).
- Take $\overline{H_i}$, the complement of H_i (Figure 22b).
- Let M_i be a maximum matching of $\overline{H_i}$ (Figure 23a). *Make a note of the M_i, which will be used again later.*
- Let K_i be $D_i \setminus V(M_i)$ (Figure 23b) (these are likely to be well-connected vertices in H_i).
- Let B_i be vertices in K_i with more than $\Delta^{5/4}$ neighbours *outside* H_i in G^2.
- Let $A_i = K_i \setminus B_i$ (Figure 23b) (i.e. subtract vertices that are highly connected outside this subgraph).

Figure 21 Havet et al. construction, second stage

- Next, take the union of the A_i and construct two subgraphs with the following vertex sets:

$$V_1 = V(G) \setminus \bigcup_i A_i;$$

$$V_2 = \bigcup_i A_i$$

  V_1 is thus the $K_i \cup B_i$, i.e. the highly connected vertices, and will be labelled first, subject to some additional conditions to ensure that we can extend the labelling to V_2. Finally, we will label the remaining vertices S of the original decomposition.

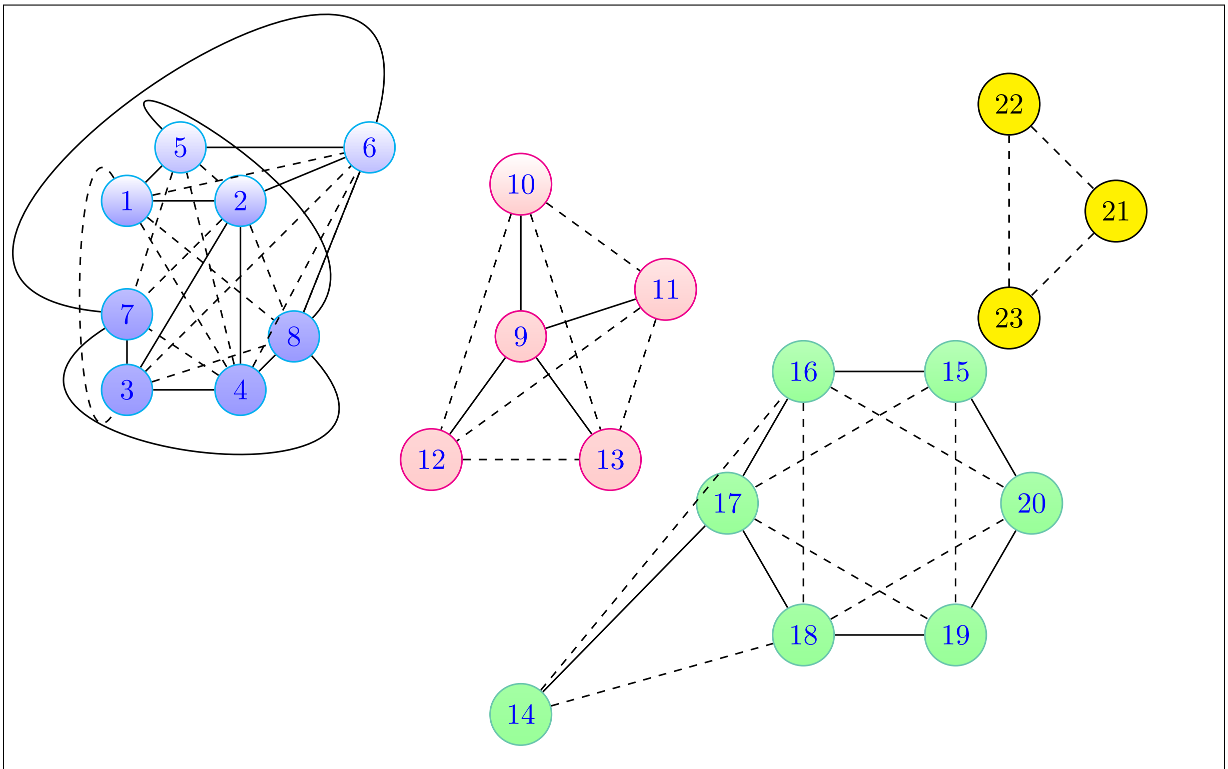
- To label V_1, derive new graphs G^* and G^{2*} as follows:

  - Construct new vertices from the M_i by replacing each matched pair of vertices v_j, v_k with a single new vertex C_{jk} (Figure 24) whose neighbours are the neighbours of v_j and of v_k, so the maximum degree of the new graph G^{2*} is $\leq 2(\Delta^2 - \Delta)$ (since G^2 may have degree $\Delta^2 - \Delta$).
  - Add additional edges between particularly highly connected vertices, subject to certain conditions (for G^{2*} only). The requirement for "highly connected" here is $> \Delta^{9/5}$, i.e. close to the maximum $\Delta^2 - \Delta$ (the C_i are not candidates for the additional edges).
  - Remove the vertices of the A_i – this leaves us with the C_{jk} and various highly connected vertices (Figure 25).

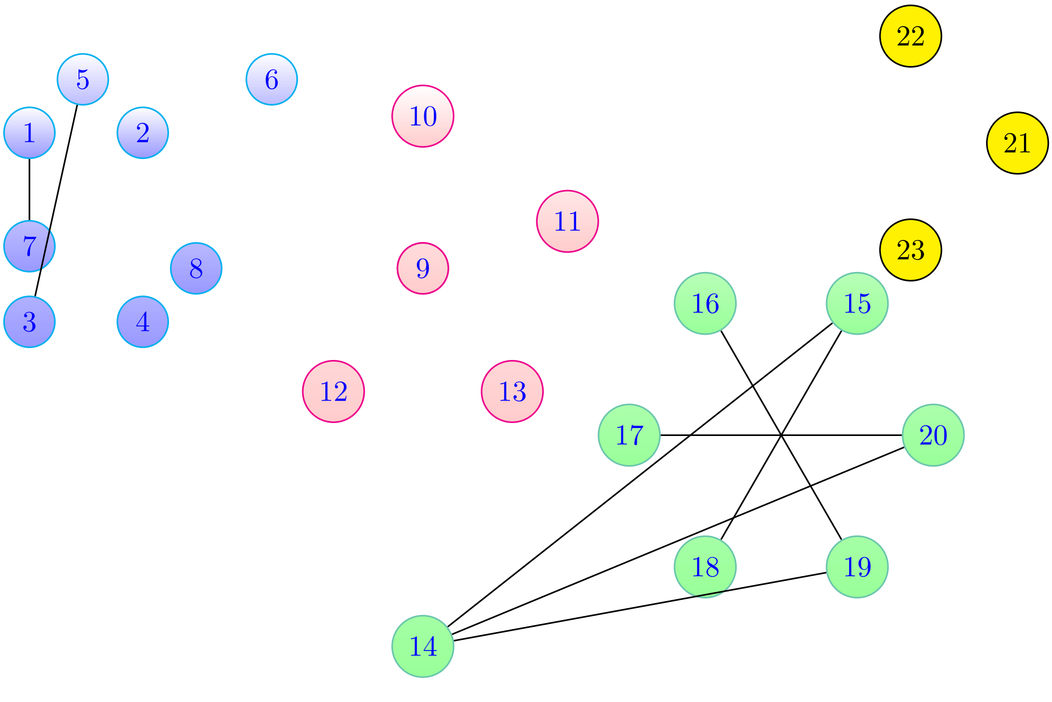
- Now find a labelling such that vertices adjacent in G^{2*} have distinct labels, and vertices adjacent in G have label distance ≥ 2. (Note that since we added "extra" edges to G^{2*}, this is nearly, but not precisely, an L(2,1)-labelling of G^*).

- Extend the labelling to the subgraph of G induced by V_1 by assigning each pair of matched vertices v_i, v_j the label of C_{ij} in G^*. Since these matched vertices correspond to edges in the complement of G^2, they cannot be neighbours in G^2, so the labelling will be valid.

- We omit details of further extension of the colouring to the full graph, which is subject to additional constraints required for the probabilistic element of the proof. Figures 27a and 27b show a labelling of V_1 for our example and an extension to the rest of G.

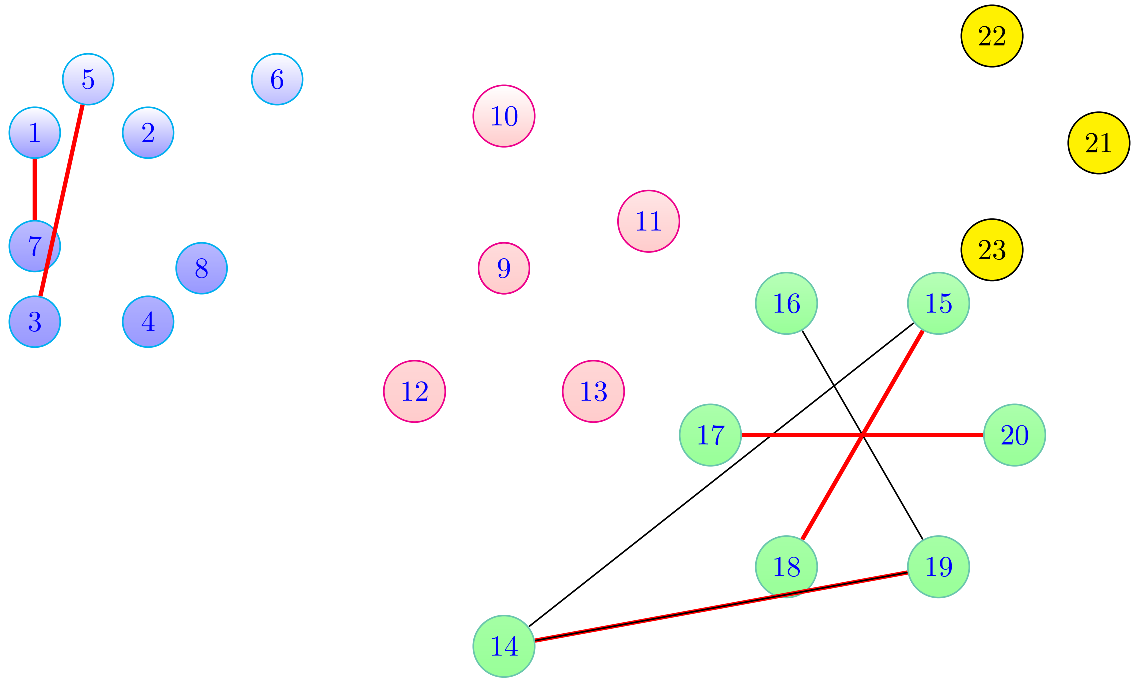


(a) Subgraphs H_0, H_1, H_2, H_3 (dashed lines are edges from G^2)



(b) Complements $\overline{H_0}, \overline{H_1}, \overline{H_2}, \overline{H_3}$

Figure 22: The H_i and $\overline{H_i}$



(a) Maximum matchings in the $\overline{H_i}$

$K_i := D_i - V(M_i)$

$B_i :=$ vertices in K_i with $> 7^{5/4} = 12$ or more neighbours in G_2 outside H_i.

$K_0 = \{2, 4, 6, 8\}$. None of these has more than 11 neighbours outside D_0. $\Rightarrow B_0 = \{\}$.

$K_1 = \{9, 10, 11, 12, 13\}$; of these, 9 has the whole of D_1 as neighbours, as well as 14, 15, 16, 17 $= 12 > 11$ neighbours outside D_2. $B_1 = \{9\}$.

$K_2 = \{16\}$. Vertex 16 has 8+5+3 = 15 neighbours. $\Rightarrow B_2 = \{16\}$.

$K_3 = \{21, 22, 23\}$; B_3 is empty.

Now we set $A_i = K_i \setminus B_i$: $A_0 = \{2, 4, 6, 8\}, A_1 = \{10, 11, 12, 13\}, A_2 = \{\}, A_3 = \{21, 22, 23\}$

$$\bigcup_{A_i} = \{2, 4, 6, 8, 10, 11, 12, 13, 21, 22, 23\}.$$

(b) Constructing the A_i

Figure 23: The Havet process: from H_i to $\bigcup_{A_i}$

- $V_1 = V \setminus \bigcup A_i = \{1, 3, 5, 7, 9, 14, 15, 16, 17, 18, 19, 20\}$

- $V_2 = \bigcup A_i = \{2, 4, 6, 8, 10, 11, 12, 13, 21, 22, 23\}$

$C_0 = \{v_1v_7, v_3v_5\}$;

$C_1 = \{\}$;

$C_2 = \{v_{18}v_{15}, v_{17}v_{20}, v_{14}v_{19}\}$; $C_3 = \{\}$

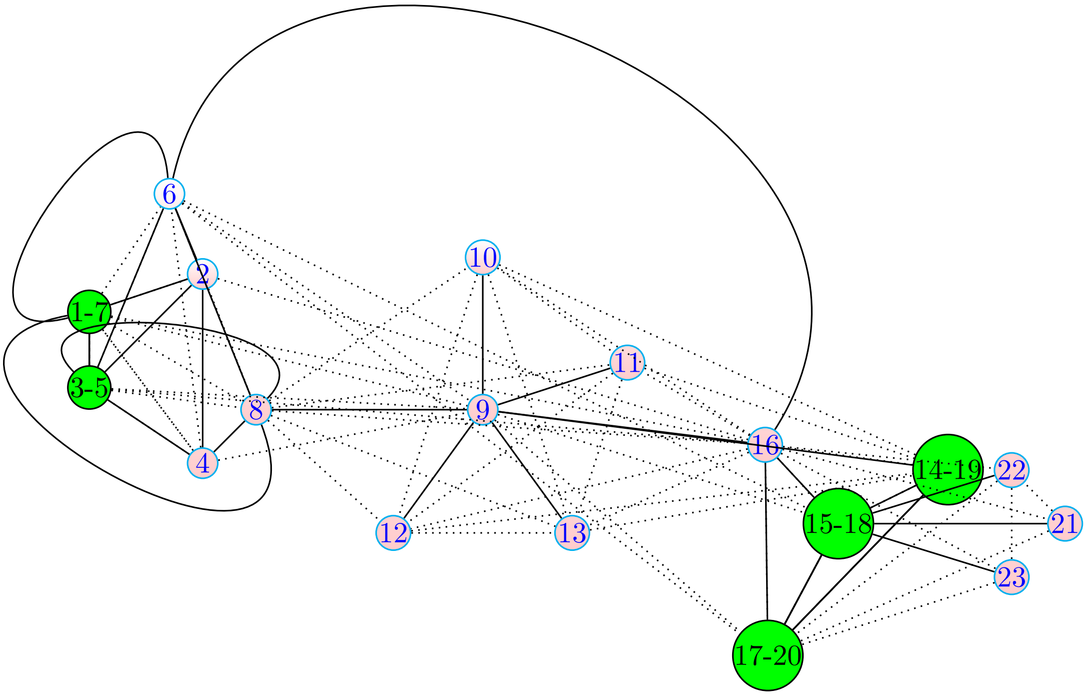
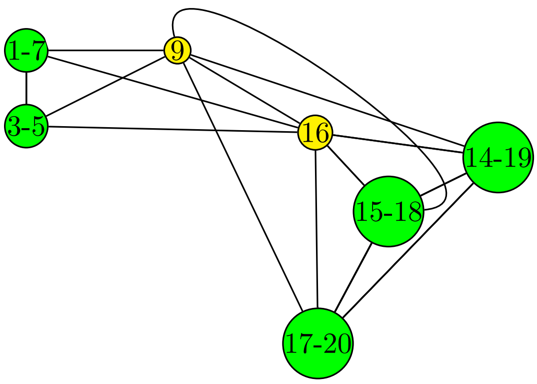


Figure 24: G^2 with the M_i contracted

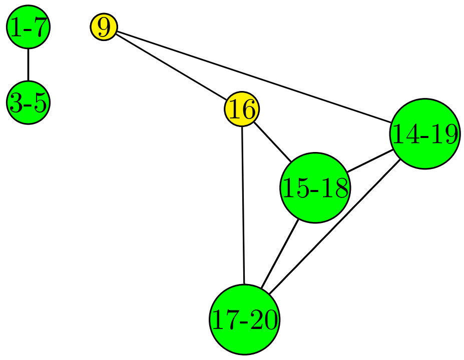
$$
\begin{aligned}
V^* =& V(G) - \{1,3,5,7,14,15,16,18,19,20\} + \bigcup_i C_i \\
=& \{2,4,6,8,9,10,11,12,13,16,21,22,23\} + \bigcup_i C_i
\end{aligned}
$$

(nothing has $> \Delta^{9/5} = 34$ neighbours $\rightarrow$ no new edges)

Now we subtract the vertices of $\bigcup_i A_i$, leaving $V^* = \{9, 17, v_1v_7, v_3v_5, v_{18}v_{15}, v_{17}v_{20}, v_{14}v_{19}\}$:

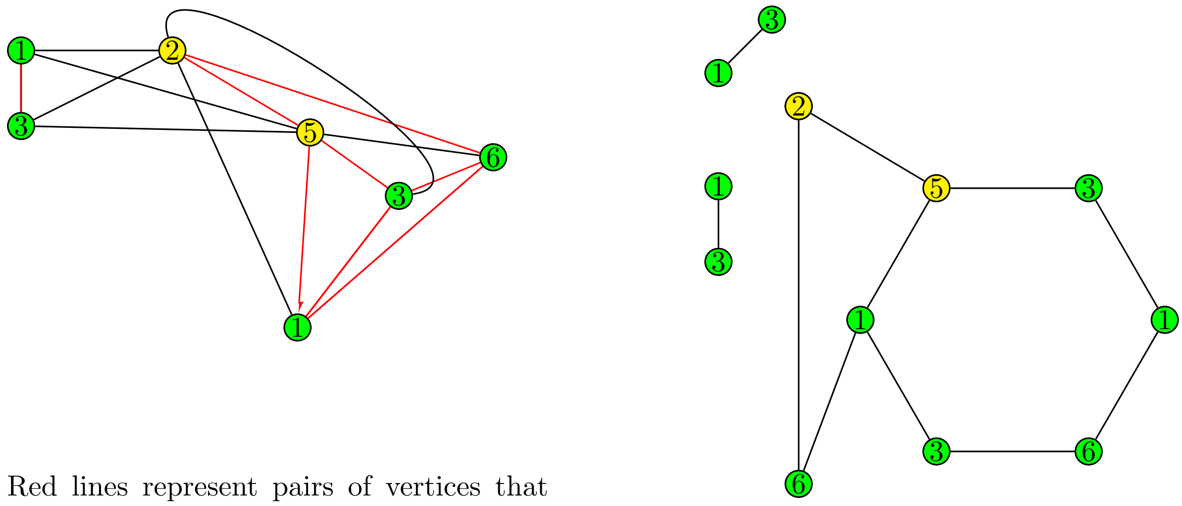


(a) G^{2^*}



(b) G^*

Figure 25: The derived graphs G^* and G^{2^*}

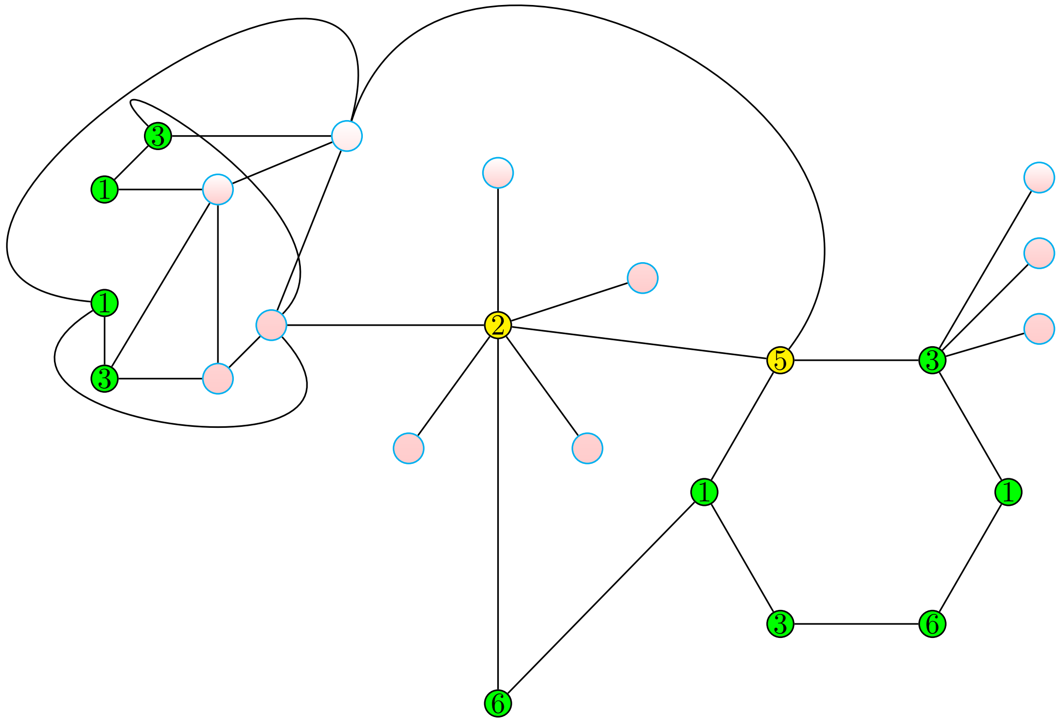


Red lines represent pairs of vertices that must be labelled with distance ≥ 2.

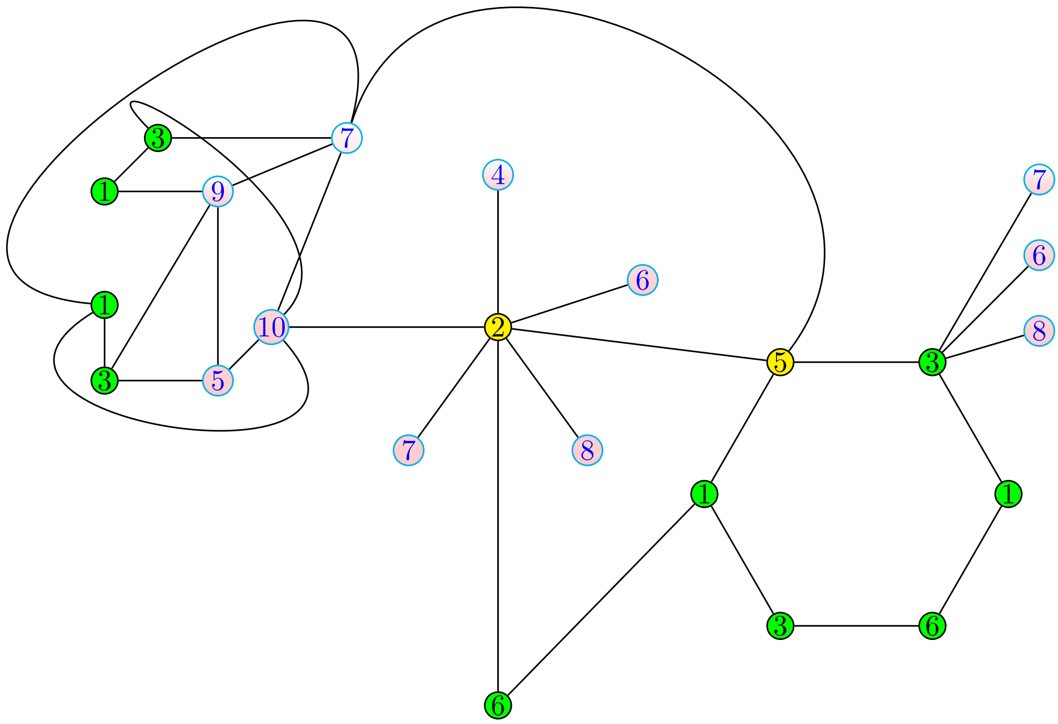
(a) Labelling of G^{2^*}

(b) Labelling expanded onto $G(V_1)$

Figure 26: Labelling the vertices V_1



(a) G with the labels from Figure 26b inserted



(b) Labelling of Figure 27a extended into an L(2,1)-labelling for G with labelling number 10

Figure 27: Final labelling for G