FACULTY OF SCIENCES

Algorithms for computing Ramsey numbers

Steven Van Overberghe

Student number: 01504990

Supervisor: Dr. Jan Goedgebeur

Department of Applied Mathematics, Computer Science and Statistics

A dissertation submitted to Ghent University in fulfillment of the requirements for the degree of Master of Science in Mathematics

Academic year: 2019 - 2020

GHENT UNIVERSITY

Voorwoord

In deze thesis heb ik mij een jaar lang kunnen uitleven in een onderwerp dat mij al heel lang boeit: Ramsey theorie. Vorig academiejaar maakte ik hierover reeds een literatuurstudie, dit jaar ging ik er zelf mee aan de slag.
Ik wil dan ook graag mijn promotor Jan Goedgebeur bedanken om me de kans te geven hierin mijn eigen weg te gaan en me te ondersteunen en bij te sturen waar nodig.

Graag bedank ik ook mijn vriend Pieter Vyncke, om me vijf jaar geleden in te leiden in deze mooie tak van de combinatoriek, maar ook voor zijn hulp en interesse in deze thesis. Ook bedankt aan alle medestudenten die mijn opleiding tijdens en na de lessen heel aangenaam hebben gemaakt. Een dankjewel ook aan iedereen die delen van dit manuscript heeft nagelezen of feedback gaf.

Verder ben ik ook de Universiteit Gent erkentelijk, voor hun kwaliteitsvol onderwijs dat mij in staat heeft gesteld dit werk te maken, alsook voor het gebruik van de supercomputer-infrastructuur om bepaalde resultaten hier mogelijk te maken.

My thanks also go to professor Stanisław Radziszowski for his help and advice, and for his excellent survey.

Als laatste bedank ik ook mijn familie en mijn vriendin, voor hun emotionele ondersteuning doorheen het jaar.

Toelating tot bruikleen

De auteur geeft de toelating deze masterproef voor consultatie beschikbaar te stellen en delen van de masterproef te kopiëren voor persoonlijk gebruik. Elk ander gebruik valt onder de beperkingen van het auteursrecht, in het bijzonder met betrekking tot de verplichting de bron uitdrukkelijk te vermelden bij het aanhalen van resultaten uit deze masterproef.

Steven Van Overberghe
1 juni 2020

Contents

Introduction **xi**

1 Preliminaries **1**

2 Ramsey theory **3**

2.1 Graphs 4

2.2 Hypergraphs 7

2.3 Current status on small Ramsey numbers 8

3 Lower bounds on Ramsey numbers **11**

3.1 Circulant graphs 11

3.1.1 Enumeration 13

3.1.2 Avoiding different kinds of graphs 15

3.1.3 Isomorphism 18

3.2 Distance graphs 20

3.3 Block-circulant graphs 22

3.3.1 Avoiding isomorphisms 23

3.3.2 Search order 27

3.4 Searching between known interesting graph classes 27
3.4.1 Vertex-transitive graphs 28
3.4.2 Strongly regular graphs 29
3.5 Results 30
3.5.1 Classical numbers 30
3.5.2 Almost-cliques 31
3.5.3 Wheels 32

4 Upper bounds on Ramsey numbers **35**
4.1 Direct Enumeration 35
4.2 Graph splitting 36
4.3 Gluing Ramsey graphs 37
4.3.1 Edge-counting constraints 39
4.3.2 Restricting the search 41
4.4 R(K_3,J_4,J_4) 42
4.4.1 Graph splitting 42
4.4.2 Graph gluing 42
4.5 R(J_5,J_6) 44
4.5.1 R(J_5,J_5) 45
4.5.2 Final steps 49

5 Conclusion and future work **51**

A Nederlandstalige Samenvatting **55**

A.1 Ondergrenzen op Ramseygetallen 56

A.2 Bovengrenzen op Ramseygetallen 56

B Graph counting tables 59

C Graphs and results 63

Bibliography 69

Introduction

In this Master's thesis, we will investigate Ramsey theory, a research field that studies conditions under which order must always show up in chaos. For graphs there is the famous theorem of Ramsey which states that a large enough graph always contains either a large clique or a large co-clique. The transition point from 'not large enough' to 'large enough' is called a Ramsey number. We will see that finding the exact value of these Ramsey numbers is both a theoretical and an algorithmic challenge.

The necessary graph-theoretic notations and definitions are explained in chapter 1. We will then give a general introduction to Ramsey theory for graphs in chapter 2. Except for some basic mathematics this chapter should be self-contained.

In chapter 3 we considered some ways to improve lower bounds on Ramsey numbers. For this we implemented several enumeration-algorithms, each time building further upon our experiences. We succeeded in proving several new lower bounds and even exact values of Ramsey numbers.

Proving upper bounds to Ramsey numbers is in general much harder, and is discussed in chapter 4. We focused on two particular cases of which one succeeded and one did not, resulting in another exact value for a Ramsey number.

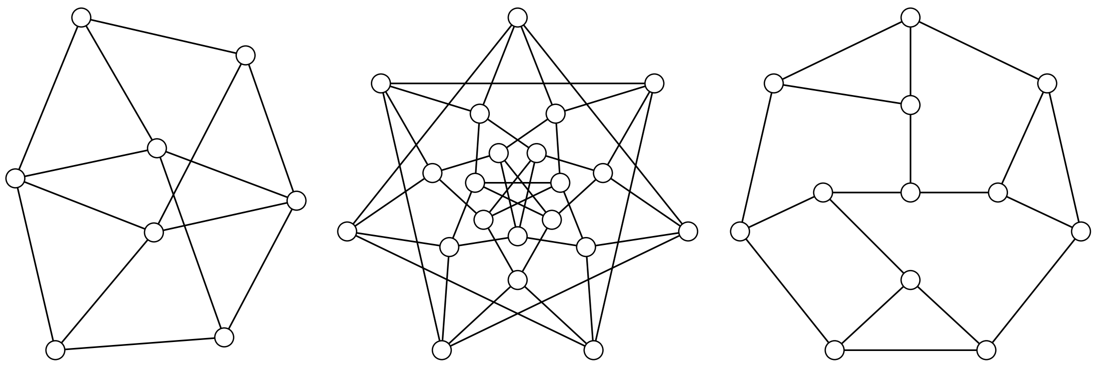
The thesis ends with chapter 5 where we look back at the results and give possible paths to continue the research.

Chapter 1

Preliminaries

A (simple) **graph** G is set equipped with a symmetrical binary relation. More specifically we say $G = (V, E)$, with V a set and $E \subseteq \binom{V}{2}$. In this document, we will always assume V to be finite. Elements of V are called **vertices** or *nodes* and will mostly be denoted by the letters $v, w, \ldots$ while elements of E are called **edges**. We say that two vertices v, w are *adjacent* if $\{v, w\} \in E$, which will be denoted by $v \sim w$. The number of vertices (i.e. the cardinality of V) will be called the **order** of G, and the number of edges is known as the *size* of G.

In visual representations of graphs, every vertex will be a small circle and an edge will be a line connecting two circles.



Graphs have many real-word applications, being models of road maps, molecule structures, social networks, etc. Many definitions on graphs therefore have recognizable names, such as: walk, distance, cycle, clique... For a full list of definitions I will redirect the reader to [41].

The set of vertices which are adjacent to v is called the **neighbourhood** of v: $N_G(v) := \{w \in V \mid w \sim v\}$ (=: $N(v)$ if G is clear from the context). If $G = (V, E)$ then $\overline{G} := (V, \binom{V}{2} \setminus E)$

is called the **complementary graph** and consists of all edges which are not present in G.

The following concepts on graphs will be very important in this work: $H = (V', E')$ is called a **subgraph** of $G = (V, E)$ if $V' \subseteq V$ and $E' \subseteq E \cap \binom{V'}{2}$. If the latter inclusion is an equality then H is called an induced subgraph. $K \subseteq V$ is called a **clique** if the subgraph of G induced by K is a complete graph, i.e. $(\forall v, w \in K)(v \sim w)$. The **clique-number** of a graph G is equal to the size of the largest clique in G: $\omega(G) := \max\{|K| \mid K \subseteq V \text{ is a clique}\}$ A clique of size q is called a q-clique, and when considered as a graph on its own, is denoted by K_q.

The dual notion of a clique is an **independent set** (or co-clique): a subset of V containing no edges between its vertices. This is equivalent to being a clique in the complementary graph.

We will also frequently talk about **colourings** of graphs. In general, this could either denote an edge-colouring or a vertex-colouring. In both cases it simply means assigning a single colour to all of the objects considered. This can also be seen as defining an ordered partition on these objects. Unless otherwise stated, we will say *colouring* for an edge colouring. Technically, this will be a function from the edges to a set of colours (which will here be finite): $\chi : E \to C$. When the number of colours is low, we will denote them by 1, 2, 3, 4,... or blue, red, green, yellow,... (in this order, so the first colour is always blue). For brevity, we will say k-colouring (or k-edge-colouring) for a colouring which uses at most k colours. If a certain edge-colouring on G is present, we will speak of the 'blue neighbourhood' of v etc. to denote $N^1_G(v) := \{w \in V \mid \chi(\{v, w\}) = 1\}$.

Very often we will identify a two-coloured complete graph G with a regular (non-complete) graph H by considering the blue-coloured edges in G to be the present edges in H. This identification will mostly happen implicitly.

Vertex- and edge-colourings have important applications in scheduling problems, where vertices may for example represent events and edges model the restrictions on planning two events at the same time. In such examples one often looks for *proper* colourings, that is: no two adjacent vertices have the same colour. In this work we will however not desire this, so a colouring here is never assumed to be proper.

Chapter 2

Ramsey theory

Ramsey theory is based around the idea that for many kinds of '*structures*' it is true that, if we choose the structure to be large enough, we can always assure ourselves it will have a large *substructure* having certain properties. When formulated in such abstract way, one can detect it in many areas of mathematics; but to capture the idea behind Ramsey theory, it will help to look at a simple example. We will take the structures to be 'coloured sets', and the property of substructures we desire is that all elements have the same colour (a property we will call **monochromatic**).

Theorem 2.1 (Generalized pigeon hole principle)**.** *For all $k, m \in \mathbb{N}$, if $n = (m-1) \cdot k + 1$ items are coloured with k different colours, at least m items will have the same colour.*

Thus, the theorem says that for whatever number of colours, and for whichever monochromatic size we desire, there is a size n for which *every* colouring of n objects will have such a monochromatic subset.

Of course, if this property holds for n, it will also hold for all $n' > n$, because one can always 'think away' the last items in the set. Note also that $n = (m-1) \cdot k$ items are not enough to be certain of finding a monochromatic set of size m. Simply consider the partition of n items in k colour classes of size $m-1$. Thus, for this Ramsey property, the transition from 'not large enough' to 'large enough' always occurs at $n = (m-1) \cdot k + 1$.

We were however very lucky here that this problem is so simple that we can give the exact and general value of the transition. For other Ramsey problems we will have to settle with an interval in which the transition point must lie, meaning that theorems of the following style will be common:

Theorem 2.2 (Weakened generalized pigeon hole principle). $(\forall k, m \in \mathbb{N})(\exists n \in \mathbb{N})$*: if* n *items are coloured with* k *different colours, at least* m *items will have the same colour.*

2.1 Graphs

Instead of colouring single objects, we could also assign a colour to every *pair* of objects. A way to visualise this, is by taking a collection of people, and check for every two people if they know each other or not (at least, once a formal and symmetrical notion of *knowing* has been agreed). Then we could say a pair of people will be coloured blue if they know each other and red otherwise. In the language of graph theory, every person would be considered a vertex of a complete graph, and the *knowing*-relation would be a colouring of the edges.

This context can now give rise to new Ramsey properties, if we demand the right conditions on the subgraphs. A natural example would be to look for *monochromatic* subsets: a collection W of vertices such that every edge between two vertices in W has the same colour (i.e. a monochromatic *clique*). Or to say it in terms of people: a group of people such that they are all mutual acquaintances or all mutual strangers. In this context it is also known as the 'party problem'.

It might not be immediately clear that one can ever be certain of the existence of such a clique or co-clique of people, but this was proven by Frank Ramsey in 1930 [35]. His motivation was to solve a problem in logic, but soon it was discovered that the ideas behind it were also helpful in other combinatorial problems. Over time, the idea of order showing up in chaos turned into a full-fledged research domain called *Ramsey theory*, of which we will only see a tiny piece here. A very nice overview of this research area is given in [19].

Ramsey's theorem for graphs (which will be formally stated as theorem 2.5) holds for any number of colours, but for clarity, we will first focus on the 2-colour case. We want to prove that for any k, there is a number of vertices n such that any 2-colouring of a K_n contains a monochromatic k-clique. It will however turn out to be easier to prove if we generalise this theorem just a little to 'off-diagonal' cases. Namely we will not always demand the same size of clique in all of the colours. Some additional notation and terminology will be practical.

Definition 2.3. *A 2-edge-coloured graph G is called a **Ramsey-$(\mathbf{k_1}, \mathbf{k_2})$-graph** (or simply a **$(\mathbf{k_1}, \mathbf{k_2})$-graph**) if it does not contain a monochromatic k_1-clique in the first colour nor a k_2-clique in the second colour. If k_1 and k_2 are clear from the context G can also be called a Ramsey graph. $\mathcal{R}(k_1, k_2; n)$ denotes the set of all Ramsey (k_1, k_2)-graphs of order n. Similarly the notation $(k_1, k_2; n)$-graph specifies the order.*

Definition 2.4. *The property that no 2-edge-colouring of K_n is a Ramsey-(k_1, k_2)-graph will be expressed by the 'arrowing'-relation:*

$$n \to (k_1, k_2)$$

Equivalently, it expresses that every blue/red-colouring of a K_n either contains a blue K_{k_1} or a red K_{k_2} as a subgraph.

If $G = (V, E, c)$ is a Ramsey-(k_1, k_2)-graph, then $G' = (V \setminus v, E', c')$ (obtained by removing an arbitrary vertex v and retaining the colour of all edges not containing v), will also be a Ramsey graph: any blue k_1-clique or red k_2-clique in G' would still be present in G if we added v back again. Hence for $n > 0$ we have that

$$n \nrightarrow (k_1, k_2) \Rightarrow n - 1 \nrightarrow (k_1, k_2)$$

As a direct consequence:

$$n \to (k_1, k_2) \Rightarrow (\forall m > n)\ m \to (k_1, k_2)$$

Since a 1-clique is just a vertex (and hence independent of the edge-colouring), we can immediately see that $(\forall k_2)\ 1 \to (1, k_2)$. A 2-clique is simply an edge, so a 2-edge-colouring containing no blue K_2 means that the blue colour was never used. Therefore all edges are coloured red, implying that we have a red k_2-clique precisely when $n \geq k_2$, thus:

$$(\forall k_2 \in \mathbb{N})\ k_2 \to (2, k_2)$$

Notice also that $n \to (k_1, k_2)$ implies $n \to (k_2, k_1)$ since swapping the colours of a (k_2, k_1)-graph would lead to a (k_1, k_2)-graph on the same number of vertices.

Now we can prove the full (two-coloured) version of this Ramsey property on graphs.

Theorem 2.5 (Ramsey, 1930 [35])**.**

$$(\forall k_1, k_2 \in \mathbb{N})(\exists n \in \mathbb{N})\ n \to (k_1, k_2)$$

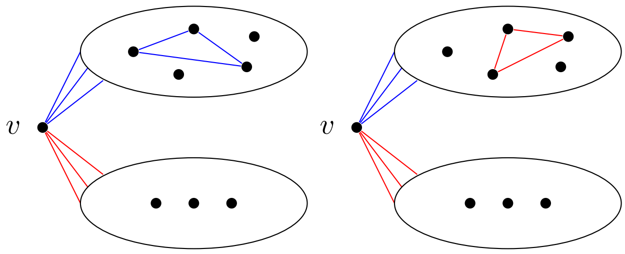


Figure 2.1: Schematic drawing of the two cases in the proof. On the left we have a blue 4-clique, on the right a red 3-clique.

Proof. With induction on k_1. For $k_1 = 1, 2$ the arguments were given above. Now suppose $(\forall k_2)(\exists n)\ n \to (k_1 - 1, k_2)$. We will show with induction on k_2 that the induction step holds for k_1. Therefore, we can assume that $n_1 \to (k_1 - 1, k_2)$ and $n_2 \to (k_1, k_2 - 1)$.
Consider a complete red/blue-coloured graph G on $n_1 + n_2$ vertices. Every vertex v has $n_1 + n_2 - 1 = (n_1 - 1) + (n_2 - 1) + 1$ neighbours. By a slight generalisation of theorem 2.1, it follows that at least n_1 edges containing v are coloured blue, or at least n_2 are coloured red. Assume without loss of generality that the first case holds.

Now, by the induction hypothesis, we have that either $N^1_G(v)$ contains a blue $(k_1 - 1)$-clique or a red k_2-clique. In the first case we find a blue k_1-clique by adding v to the $(k_1 - 1)$-clique. Together with the second case, and since the colouring on G was arbitrary, we can conclude that $n_1 + n_2 \to (k_1, k_2)$. (See figure 2.1.) □

This theorem tells us that for each k_1 and k_2, there really is some graph order which is large enough to ensure us of one of the desired cliques. Since by an argument above 'being large enough' is upward closed, it follows that there must be a unique transition point. This is called the **Ramsey number** for k_1, k_2:

$$R(k_1, k_2) := \min\{n \in \mathbb{N} \mid n \to (k_1, k_2)\}$$

If in the above proof we choose n_1 and n_2 to be minimal, i.e. $n_1 = R(k_1 - 1, k_2)$ and $n_2 = R(k_1, k_2 - 1)$, then we get a general upper bound on the Ramsey numbers, which turns out to be very useful in spite of its simplicity:

Theorem 2.6 ([10])**.**

$$(\forall k_1, k_2 \in \mathbb{N})\ R(k_1, k_2) \leq R(k_1 - 1, k_2) + R(k_1, k_2 - 1)$$

Finding the value of these Ramsey numbers is very hard, and no exact formula nor tight (general) asymptotics are known. Even evaluating *very* small instances appears to be an immense challenge. For example, even after at least 50 years of research the value of $R(5,5)$ is not known. The best known bounds are $43 \leq R(5,5) \leq 46$ [12] [3].

Of course, we are not limited to using two colours. Colouring every edge of a K_n with more colours leads to obvious generalisations theorem 2.5. The proof now requires the full version of the pigeon hole principle (theorem 2.1). If we also generalise the arrow relation and $R(\cdot)$-notation, the analogue to theorem 2.6 becomes:

$$R(k_1, k_2, \ldots, k_t) \leq \sum_{i=1}^{t} (R(k_1, \ldots k_{i-1}, k_{i-1} - 1, k_{i+1}, \ldots, k_t) - 1) + 1$$

We will also frequently use another generalisation of these ideas: we are not limited to avoiding monochromatic cliques. It is also possible (and is often done) to look at maximal graphs which do not contain a special monochromatic subgraph, for example a cycle or a path. In general, we will write

$$n \to (H_1, H_2)$$

to denote that any blue/red-colouring of a K_n contains either a (not necessarily induced) subgraph isomorphic to H_1 which is coloured blue, or contains a 'red H_2'.

The Ramsey numbers are then extended in the same way to $R(H_1, H_2)$. This number certainly exists because if H_1 has order m, a monochromatic K_m includes a monochromatic H_1. Therefore we have that for all graphs H_1 and H_2

$$R(H_1, H_2) \leq R(|H_1|, |H_2|)$$

Ramsey numbers which only avoid cliques will be called *classical* Ramsey numbers.

2.2 Hypergraphs

We saw that Ramsey's theorem was a two-dimensional generalisation of the pigeon hole principle. This was strengthened by the fact that we used the latter principle in the proof of theorem 2.5. The exact relationship between these two theorems becomes more clear if we further extend the ideas to higher dimensions.

Imagine we have a collection of vertices and colour now triples of vertices instead of pairs. The corresponding combinatorial object would be a *3-uniform hypergraph*, but we will not use its terminology here. It will be possible to prove that if we take a large enough collection, there will always be a large subset of vertices such that every triple inside the subset has the same colour. This 3-dimensional Ramsey number will be denoted by $R(k_1, k_2)^3$.

Theorem 2.7 (Ramsey's theorem for hypergraphs [35])**.**

$$(\forall k_1, k_2 \in \mathbb{N})(\exists n \in \mathbb{N})(n = R(k_1, k_2)^3)$$

Proof. Like in the two-dimensional case, we will prove this by induction, but will not state the details. Consider a hypergraph G on n vertices for which every triple is coloured blue or red, and an arbitrary $v \in V$.

v induces a colouring of pairs in the following way: $\forall w, z \in (V \setminus v)$ we colour $\{w, z\}$ blue if $\{v, w, z\}$ is blue, and red otherwise. Hence this gives an edge-colouring of $G \setminus \{v\}$ with the property that a blue k-clique K leads to a set of triples around v which are all coloured blue. If K now were large enough te ensure ourselves that each colouring of its triples contains either a blue set of size $k_1 - 1$ or a red set of size k_2, then together with v we would find the desired monochromatic set in G. By induction, this is reached by $R(k_1 - 1, k_2)^3$. For the red triples around v we analogously need a clique of size $R(k_1, k_2 - 1)^3$.

Now we fall back to theorem 2.5 to ensure us of such cliques and find:

$$R(k_1, k_2)^3 \leq R[R(k_1 - 1, k_2)^3, R(k_1, k_2 - 1)^3]^2 + 1$$

□

For example $R(3, 4)^3 = 4$ (trivially), so $R(4, 4)^3 \leq R(4, 4)^2 + 1 = 19$. By McKay and Radziszowski [29] it is known that $R(4, 4)^3 = 13$, and this is the only classical (non-trivial) Ramsey number for hypergraphs for which the exact value is calculated [34].

From here on it should be clear that Ramsey's theorem extends to any (constant) subset-size that is being coloured.

2.3 Current status on small Ramsey numbers

The classical Ramsey numbers are widely studied and, as discussed before, it appears to be a real challenge to determine them exactly even for small parameters.

The smallest 'non-trivial' Ramsey number is $R(3, 3) = 6$. The upper bound follows from Ramsey's original bound $R(n, n) \leq n!$ [35]. The lower bound is given by a pentagon-graph.

It might be interesting to note that proving $R(3,3) \leq 6$ was a question in both the Eötvös-Kürschák Competition and the William Lowell Putnam Mathematical Competition held in 1947 and 1953 respectively.

The first few 'real' Ramsey numbers were computed in 1955 by Gleason and Greenwood, completely by hand [20]: $R(3,4) \leq 9$ follows from theorem 2.6, with one slight improvement. Namely if $R(k_1-1, k_2)$ and $R(k_1, k_2-1)$ are both even, the presumed critical graphs should be regular of odd degree and have an odd number of vertices. Since this is impossible, the inequality becomes strict. The lower bound follows from the circulant graph in figure 3.1 (defined later).
$R(3,5) = 14$ and $R(4,4) = 17$ were proven in precisely the same way. Their (circulant) extremal graphs later turned out to be unique.

Kalbfleisch found several lower bounds for higher Ramsey numbers in 1965 [23], all by circulant graphs. A computer was already used to establish exhaustive results about the best-possible circulant Ramsey-graphs. He was able to prove that $R(3,9) \geq 36$ and $R(4,5) \geq 25$.
Proving that $R(4,5) \leq 25$ already turned out to be much harder (the standard inequality gives 31) and was only completed in 1995 by McKay and Radziszowski [30]. It used very sophisticated gluing algorithms (of which a simple version will be explained in section 4.3), and required massive CPU-computations. Up to 2020, this is also the last classical two-colour Ramsey number which was determined exactly. Work is still being done to learn more about specific properties of $(4,5)$-graphs because this is crucial in proving bounds on higher Ramsey numbers.

$R(3,10)$ is the smallest open case for triangle-avoiding numbers, with the best bounds being $40 \leq R(3,10) \leq 42$. The lower bound was found by Exoo in 1989, but the upper bound was established only in 2013 by Goedgebeur and Radziszowski [17]. They did this by proving bounds on the minimal number of edges in $(3,k;n)$-colourings, requiring years of CPU-time. They also showed that the circulant $(3,9;35)$-graph found by Kalbfleisch is unique as an extremal graph.

$R(5,5)$ is now perhaps the most intriguing open case, withstanding evaluation for a very long time already. After a first lower bound of 38 was established in 1965 by Abbott, it took more than 20 years to find the current lower bound of 43, achieved by Exoo with simulated annealing techniques [12]. In the 90's, consecutive work was done by McKay and Radziszowski (for example [31]), ending in $R(5,5) \leq 49$.
Twenty years later, Angeltveit and McKay improved this bound by one, with a novel gluing technique [2]. At the end of 2019, this result was further improved to $R(5,5) \leq 46$ [3], requiring decades of CPU-time, but this was not yet published.

For classical multi-colour Ramsey numbers, even less is known.
$R(3,3,3) = 17$ was proved by Greenwood and Gleason, by partitioning a K_{16} into three copies of the Clebsch graph. The upper bound was again trivial [20].
The only other known value is $R(3,3,4) = 30$. The lower bound is given by a circulant graph found by Kalbfleisch, the upper bound was proven in 2016 by means of symmetry breaking [7].

An overview of almost all bounds that are known for small Ramsey numbers of different kinds is presented in the dynamic survey of Stanisław Radziszowski [34], and all claimed 'best-known bounds' in this work are due to this survey (except for some papers which are more recent than its latest revision).

Chapter 3

Lower bounds on Ramsey numbers

Lower bounds on Ramsey numbers are mostly proved by giving a witness that doesn't have the desired Ramsey property. Such a witness (called a *Ramsey graph*) could be part of a general construction, or found 'at random' by a heuristic algorithm. It is also possible to prove a lower bound by non-constructive means like Erdős did in [9].
In search for new lower bounds, my intention was to follow an algorithmic path which would not need the experience and finesse of heuristic approaches, and which would have no randomised elements in it. Because it is generally not possible to generate all Ramsey graphs except for very small parameters, the idea here was to focus on Ramsey graphs which are part of certain families of graphs.

In this chapter, enumeration of circulant graphs will first be explained. We will then generalise this concept into two other classes: distance graphs and block-circulant graphs. Lower bounds found (or not found) by other graphs will be considered in section 3.4, and the chapter ends with an overview of all newly achieved lower bounds (section 3.5).

3.1 Circulant graphs

When looking at what specific graphs are critical for the exactly known classical Ramsey numbers, very few can be said in general. Some believe that for every Ramsey number there should be critical graphs with significant structure on them, but this is not the case with the conjecture about $R(5,5) = 43$, where all 656 presumed critical graphs have nearly no automorphisms ([31]). Yet it appears that circulant graphs arise frequently as lower bounds for Ramsey numbers. Many

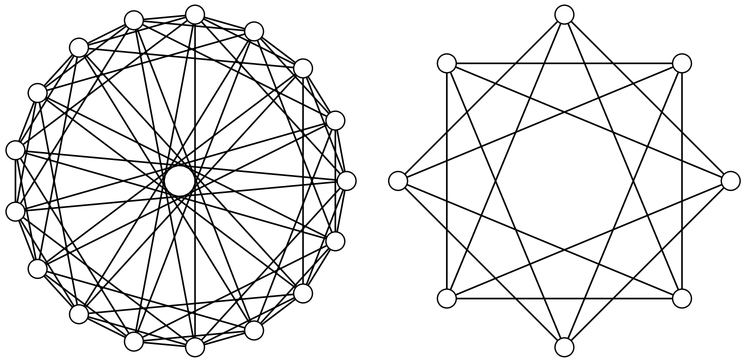


Figure 3.1: Two circulant graphs: the extremal Ramsey graphs $(17, [1, 2, 4, 8])$ and $(8, [2, 3])$

of the currently best-known bounds are reached by these graphs. For example the lower bounds on $R(4, 4)$, $R(3, 9)$ and $R(K_3, J_7)$ are all reached by unique circulant graphs. Also many of the currently best-known lower bounds on diagonal numbers $R(k, k)$ are given by such graphs.

Definition 3.1. *A graph G is called **circulant** if there exists an ordering of the vertices such that every circular rotation is an automorphism of G. Alternatively, if $G = (V, E)$ with $V = \{0, \dots, n-1\}$, then there must be an ordering of V such that $v \sim w$ only depends on $(v - w) \mod n$.*

In this section we will always assume $V = \{0, \dots, n-1\}$ to ease the notation, and if we speak about circulant graphs, the order of V is assumed to be correct. With *distance between v and w* we will mean here the natural rotational distance, being $(v - w) \mod n$. This is of course not a real distance measure because it is not symmetrical, so sometimes 'distance' could also mean $\min\{(v - w) \mod n, (w - v) \mod n\}$, but this should be clear from the context.

A circulant graph of order n is uniquely determined by the set of *distances* between adjacent vertices. By the assumption of the properties of V, this is equivalent to giving the neighbourhood of 0, which we will call D, and can be represented by a bitset of length n. This duality between graph and set will be used throughout the entire process. We will denote such circulant graph by $G = (n, D)$. Sometimes only the lower half of D is used to describe this graph.

Conversely, a set of distances $D \subseteq \{0, \dots, n-1\}$ corresponds to a circulant graph if it does not contain 0 (because we don't want loops), and is closed under negation modulo n. This is because we must have that

$$v \sim w \Leftrightarrow (v - w) \mod n \in D \Leftrightarrow (w - v) \mod n \in D \Leftrightarrow w \sim v$$

That means there are only $\lfloor n/2 \rfloor$ independent choices for the present distances. Therefore there are $2^{\lfloor n/2 \rfloor}$ circulant graphs on n vertices, which is relatively few to the $2^{n(n-1)/2}$ simple graphs (ignoring isomorphism). Together with the easy representation as bitstrings, this makes them a friendly class for enumeration.

3.1.1 Enumeration

In order to use these circulant graphs for Ramsey purposes, finding them heuristically could suffice, but given the discussion before, the goal here was to write a program which enumerates all circulant Ramsey graphs for certain Ramsey-properties.
Because this is equivalent to generating all relevant sets of distances $D \subseteq \{1, \ldots, \lfloor n/2 \rfloor\}$, a backtracking procedure seemed the right approach. That means, we first consider all distances to be uncoloured. Then iteratively we pick a distance and for every allowed colour we proceed recursively. This is explained in pseudo-code in algorithm 1.

Algorithm 1 Enumerating circulant graphs

```
function ENUMERATE_CIRCULANTS(graph-size n, k colours, avoid-sizes: (k_i)_i)
    D_i = ∅,  1 ≤ i ≤ k                                ▷ Distance-sets for each colour
    GENERATE_RECURSIVELY(n, (k_i)_i, 0, (D_i)_i )
end function
function GENERATE_RECURSIVELY(n, (k_i)_i, distance d, (D_i)_i )
    if d = ⌊n/2⌋ + 1  then
        Found circulant Ramsey graph                    ▷ Save (D_i)_i
        return
    end if
    for each  colour c ∈ {1 . . . k} do
        if  (n, D_c ∪ {d}) contains no K_{k_c}  then
            GENERATE_RECURSIVELY(n, (k_i)_i, d + 1, (D_1, . . . D_c ∪ {d} . . . , D_k) )
        end if
    end for
end function
```

Clique detection

Since the algorithm ensures (by pruning) that at every point in time, no forbidden cliques are formed in the coloured graphs, we know that only the newly added distance can cause such cliques to appear. We will take this into consideration when checking if we can proceed the recursion.

Now it is possible to exploit the fact that we are dealing with circulant graphs. Imagine that after adding distance d to the blue set, there is a forbidden blue k-clique. By the above reasoning, we know it must contain two vertices at distance d, say v and w. Because circular rotation is an automorphism, we now know that there must also be a blue k-clique containing the vertices 0 and d by rotating either v or w to 0.

This means that to check if it is allowed to add the n (or sometimes $n/2$) edges of length d, we only need to verify that the common neighbours of 0 and d contain no $(k-2)$-clique. This gives a depth gain of 2 orders over an uninformed clique-algorithm. For example, if we are avoiding triangles, we only need to check in every stage that $N(0) \cap N(d)$ is empty.

For higher cliques, we proceed with a very simple recursive algorithm which iterates through a set $W \subseteq V$, calculates intersections of neighbourhoods with W and goes to a deeper level. This is explained in algorithm 2.

Algorithm 2 Finding K_n

```
function CONTAINS_KN(G, (clique-)size, set S)          ▷ set-informed search
    if size=1 then
        return |S| > 0
    end if
    if |S| < size then
        return False
    end if
    for v ∈ S do
        if CONTAINS_KN(G, size − 1, S ∩ {w ∈ N(v) | w > v}) then
            return True
        end if
    end for
    return False
end function
```

Clique prediction

Instead of checking in each step if adding a distance to D is allowed, it is also possible to work the other way around.
In the beginning every distance can be added to D (with one exception explained below). Each time we add a distance d, we could look for every distance d' which, when added to D, would now give rise to a forbidden clique containing d. This has the benefit that we can now keep track of how many colours are still available for every leftover distance. Especially in a 2-colour case this is very handy because if one colour is impossible, we know it must get the second colour, hence we can proceed one step without branching.

If we would be colouring one edge at a time, then after colouring $e = \{v, w\}$, we should look for every monochromatic K_n-e which contains e, giving us information about the missing edge. In this situation however, colouring a distance assigns a colour to many edges at the same time, meaning that the gaps in a monochromatic K_n-$n \cdot e$ could in theory be filled with only 1 distance. This makes it hard to predict the forbidden distances without actually testing them out one by one.

Only for the case of avoiding triangles I managed to work this out by hand and implement it in the generator. In that case we have that after adding distance d to D, also adding $x + d$ would lead to a triangle if $x \in D$. This is the triangle consisting of vertices 0, d and $x + d$. If $n - d$ is even, one must notice that there is also the possibility of making a triangle with d and *twice* the new distance, i.e. the triangle consisting of vertices 0, d and $\frac{n-d}{2}$, so this distance must be avoided as well.
If n is divisible by three, it is even possible to create a triangle consisting of one unique distance, $n/3$. This extra case must be taken into consideration at the start of the algorithm.

3.1.2 Avoiding different kinds of graphs

Since it was not possible to improve bounds on classical Ramsey numbers (i.e.: avoiding cliques) with circulant graphs, the program was soon extended so it could also avoid other interesting graphs (discussed in the dynamic survey [34]).

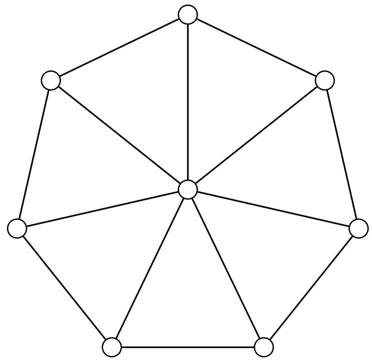


Figure 3.2: W_8

Almost-cliques

A complete graph or order n with one edge removed is called K_n-e, or J_n. Checking if a graph contains a K_n-e was quite different in case we were uninformed compared to the case where we know which edge it must contain.
If we know nothing, or a set S of vertices in which the entire K_n-e must lie, we can iterate every non-edge $\{v, w\}$ in S and check if $N(v) \cap N(w)$ contains a K_{n-2}. This will go through every possible K_n-e at most once, because the missing edge is unique in such subgraph.

In the generation of circulant graphs we will however always know an edge which the $K_n - e$ should contain. That is because just like with cliques, we can rotate a $K_n - e$ in G such that it contains the newly added edge $\{0, d\}$. There are however two kinds of edges in a $K_n - e$: edges which are incident with a vertex of degree $n - 2$ and those who are not.
If $\{0, d\}$ is of the former kind, we can look for the other vertex of degree $n - 2$ and check the intersection of the neighbourhoods. In the latter case, $N(0) \cap N(d)$ must contain a $K_{n-2} - e$, but now we are uninformed about its position. This is explained in algorithm 3. Notice that when implemented like this, we will only find induced subgraphs, so we should check for a K_n first.

Wheels

The **wheel-graph** W_n is a graph consisting of 1 vertex together with its neighbourhood which is exactly a cycle of length $n - 1$ (figure 3.2).

Finding a W_n which must contain an edge e in a circulant graph G turned out to be cumbersome because e could either be a 'spoke' or an edge in the cycle.

Algorithm 3 Finding $K_n - e$

```
function CONTAINS_KN-E_EDGE(G, size, e={v, w})          ▷ Edge-informed search
    if CONTAINS_KN-E(G, size − 2, N(v) ∩ N(w)) then
        return True
    end if
    for each z ∈ (N(v) △ N(w)) do                       ▷ △ = symmetric difference
        if CONTAINS_KN(G, size − 3, N(v) ∩ N(w) ∩ N(z)) then
            return True
        end if
    end for
    return False
end function

function CONTAINS_KN-E(G, size, set S)                  ▷ Vertex-set-informed
    for each e = {v, w} ∈ Ḡ[S] do
        if CONTAINS_KN(G, size − 2, S ∩ N(v) ∩ N(w)) then
            return True
        end if
    end for
    return False
end function
```

Since I did not intend to use wheels on very large instances, I settled with the following way of exploiting circularity: If G contains a W_n as a subgraph, then by rotating the graph, it also contains a W_n with vertex 0 as the centre. Hence we only have to check if $N(0)$ contains a cycle of length $n-1$. This is described in algorithm 4. It is clear that roughly the same idea can be used to avoid cycles.

Algorithm 4 Finding W_n

---

function CONTAINS_WN(G, size)
  if size ≤ 4 **then return** CONTAINS_KN(G, size)
  end if
  for each $v \in V \setminus \{0\}$ **do** ▷ Find smallest v in the cycle
    if CYCLE_NEAR_0($size - 1$, $\{0, v\}$, $N(0) \cap \{w \in V | w > v\}$, v) **then return** True
    end if
  end for
  return False
end function
function CYCLE_NEAR_0(*length*, *used*, *available*, *target*)
  if $length = 2$ **then**
    return $|N(target) \cap available| > 0$
  end if
  for each $v \in available$ **do**
    if CYCLE_IN_SET($length - 1$, $used \cup \{v\}$, $(N(v) \cap N(0)) \setminus used$, $target$) **then**
      return True
    end if
  end for
end function

3.1.3 Isomorphism

When enumerating instances of a class of combinatorial objects, it is a desirable property to generate every object only once up to isomorphism. The graph-isomorphism problem is is quite complicated in general, and its theoretical complexity is not yet known. There exist however some practical algorithms such as *nauty* ([28]) which can solve this problem in reasonable time.

For special classes of graphs, the isomorphism problem can nevertheless be solved in polynomial time. For example checking if two trees are isomorphic can be done in linear time. For circu-

lant graphs however, the problem seems to be more difficult. The recognition and canonisation problem was solved in polynomial time in [11], but the algorithms needed are complicated.
In case we already have a representation of the circulant graph G as set of distances D, things get a little more easier. Ádam conjectured that two circulant graphs (n, D) and (n, D') are isomorphic precisely when $D = q \cdot D'$ for some invertible element $q \in \mathbb{Z}_n$ (i.e.: q and n are co-prime). This conjecture was proven to be wrong in general, but for some orders n it does hold [1]. For example: it holds for all prime numbers n or products of two distinct primes.

No simple algorithm was found to do the effective isomorphism testing for all orders, but this is not strictly necessary for our purposes. If we have sufficient conditions on isomorphism of circulant graphs we could already avoid unnecessary generations.

We have that one direction of Ádams conjecture still holds.

Lemma 3.2. *If $G = (n, D)$, $G' = (n, D')$ and $(\exists q \in \mathbb{Z}_n^*)(D = qD')$, then $G \cong G'$.*

Proof. All arithmetic in this proof is done modulo n. The isomorphism between G and G' is established by $\phi : V \to V : v \mapsto q \cdot v$. Since $q \in \mathbb{Z}_n^*$, this is a bijection. Now we have

$$v \sim_G w \Leftrightarrow v - w \in D \Rightarrow q(v - w) \in qD \Leftrightarrow (qv) - (qw) \in qD \Leftrightarrow (qv) \sim_{G'} (qw)$$

where the second implication becomes an equivalence because q is invertible. That is because, if $q(v - w) = qd$, then $v - w = q^{-1}q(v - w) = q^{-1}qd = d$. □

To use this lemma, we should first compute the set $K := \mathbb{Z}_n^*$, which is of course constant for each n. For every distance-set D, $\mathcal{D}' = \{q \cdot D \mid q \in K\}$ corresponds with isomorphic graphs to the current graph (but they could be equal as well). In theory we could save all partial graphs we visited so far and check if $\mathcal{D}'$ contains one of them. There would however soon be very much graphs to compare so that his would become very slow.

Notice that $D_1 \sim D_2 \Leftrightarrow (\exists q \in \mathbb{Z}_n^*)(D_1 = qD_2)$ is an equivalence relation: symmetry follows from the existence of q^{-1} and transitivity is due to associativity. On this relation we can use a technique called canonisation and choose one representative out of each equivalence class. Often this is equivalence of labelled graphs under isomorphism, but as explained before, our classes based on modular arithmetic do not fully represent isomorphism. Since D is just a set over $\{1, \ldots, n-1\}$, we could choose the lexicographic smallest set as representative and reject all others. This has the additional benefit that, if we are consistently building D from smaller to larger distances (as in algorithm 1), we can be sure that for each final graph G there is a construction path consisting entirely of canonical graphs.

Lemma 3.3. *If $D_2 = [x_1, x_2, \ldots, x_k, x_{k+1}]$ with $x_1 < x_2 < \cdots < x_{k+1}$ is canonical, then $D_1 = [x_1, \ldots, x_k]$ is as well (for the same graph size n).*

Proof. Suppose $\exists q \in \mathbb{Z}_n^*$ such that $[x_1, \ldots x_k]$ is transformed to the strictly smaller (increasing) sequence $E = [y_1, \ldots y_k]$, where $y_i = q \cdot x_{f(i)}$ for some permutation f. We now have $E < D_1 < D_2$. Because $E < D_1$, $(\exists j > 0)(\forall i < j : y_i = x_i$ and $y_j < x_j)$. qD_2 is equal to E with $y' = q \cdot x_{k+1}$ inserted in the right place. It now suffices to see that the resulting E' is still smaller than D_1: if y' is inserted at the right end, $y_{\leq j}$ remain unchanged, and if y' is inserted somewhere else, $E' < E$. From $E' = qD_2 < D_1 < D_2$ the statement follows, because D_2 was assumed to be canonical. □

This means that to enumerate all circulant graphs of specified order, at each step we are allowed to stop the search if the current graph is not canonical, yet the set of equivalence classes in which we will end won't change.

For our Ramsey purposes we are always building multiple graphs at the same time (one for each colour). Since at each step all distances up to d must be coloured, it is of course not possible to ensure that all of these graphs are canonical (because if a change makes one graph smaller then another must have become bigger). So for two colours, only the first graph should be checked for such equivalences. For more colours, it is theoretically possible to make sure the second graph is minimal considering multiplication with all $q \in \mathbb{Z}_n^*$ which leave the first graph unchanged, etc.

The above pseudo-codes were implemented into an enumeration algorithm. It uses bitsets and bit-operations to build the neighbourhoods from the sets of distances and calculate very quickly the intersections of sets. These ideas about isomorphism avoidance are however not (yet) implemented in the generator because they were considered too late in time.

3.2 Distance graphs

Circulant graphs do not behave well with respect to adding new vertices: if G is a circulant graph of order n, there might not be a way to extend it by one vertex such that the result is also circulant. Conversely, removing a vertex from G will almost never lead to a circulant graph. This is because they are represented by bitstrings which are necessarily symmetric. For our Ramsey-purposes, this causes that we cannot (independently) claim that there is no circulant Ramsey

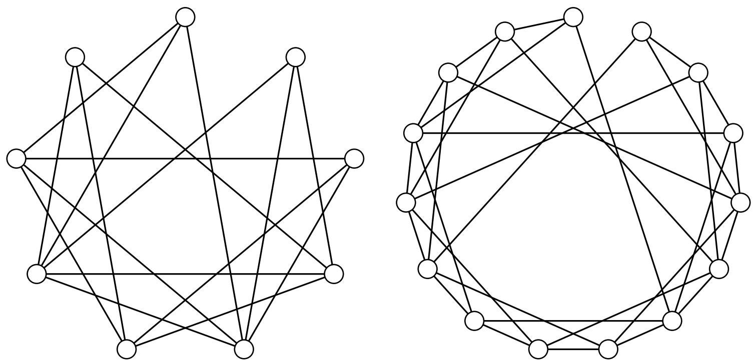


Figure 3.3: Two distance graphs: $[9, [2, 3, 5]]$ and $[15, [1, 3, 9]]$

graph of order $m > n$ if we only have information about those of order $\leq n$. In practice it can happen sometimes that there is a gap of some orders between which no circulant Ramsey-graphs exist.

There is however a generalisation to the notion of circulant graph which can be built one vertex at a time. They are called distance graphs and are identified by a not necessarily symmetric bitstring, or a set of distances.

Definition 3.4. *For any $n \in \mathbb{N}$ and $D \subseteq \{1, \ldots, n-1\}$, the **distance graph** $[n, D]$ is defined as (V, E) where $V = \{0, \ldots, n-1\}$ and $E = \{\{v, w\} \mid v > w \wedge v - w \in D\}$.*

From an algorithmic point of view, this means that $N(v) \cap \{w \mid w > v\}$ is computed by a bit-shift of D. Most of what was derived earlier of where and how certain subgraphs are located in circulant graphs remains true for distance graphs. For example: if $[n, D]$ does not contain a k-clique, then $[n, D \cup \{d\}]$ contains a k-clique if and only if $N(0) \cap N(d)$ contains a $(k-2)$-clique, because we can always rotate cliques so that they contain 0.

The good thing about distance graphs is that they can be built up from smaller distance graphs. From the definition it is obvious that every $[n, D]$ can be extended to $[n+1, D]$, and that the former is a subgraph of the latter. This extension behalves well towards classical Ramsey numbers.

Lemma 3.5. *If $[n, D]$ contains no k-clique, then also $[n+1, D]$ does not.*

Proof. Observe first by definition that the induced subgraph of $[n+1, D]$ on the first n vertices is precisely $[n, D]$. Now imagine that $[n+1, D]$ contains a k-clique K. Since $n \notin D$ either $0 \notin K$ or $n \notin K$. The former case clearly is impossible given the hypothesis, and in the latter case we can strictly rotate K until it does not contain n, also contradicting the hypothesis. $\square$

Therefore, if we have a clique-avoiding Ramsey colouring on n vertices composed of distance graphs, we can add a vertex and proceed the colouring algorithm. Because removing the last vertex also does not create cliques and leaves a distance graph, we have that if there exists no Ramsey-$[n, D]$-graph, there are also no Ramsey-$[m, D]$-graphs, for all $m > n$. Such a bound can give good estimates of the actual size of the Ramsey number. Since every circulant graph is also a distance graph, this gives also an independent bound on circulant Ramsey graphs.

Note that in lemma 3.5 it was crucial for the proof that the clique could not contain both vertex 0 and n. This is not necessarily true when avoiding J_k's, and indeed: one can find examples of distance graphs $[n, D]$ not containing a J_k but where $[n+1, D]$ does.

Kuznetsov [25] already created an enumeration algorithm for distance graphs, but only for classical two-colour Ramsey numbers. He found that this is only computationally feasible up to $R(6, 7)$, $R(4, 13)$ and $R(3, 21)$. Nevertheless an implementation was made here which was capable of handling more colours and different kinds of graphs.

3.3 Block-circulant graphs

Another generalisation of circulant graphs are the so-called *block-circulant graphs*. They are the union of several circulant graphs which are connected to each other in a circulant fashion. This means that the adjacency matrix of such a graph can be seen as a block matrix:

$$\begin{bmatrix} C_{11} & C_{12} & \dots & C_{1k} \\ C_{12}^T & C_{22} & \dots & C_{2k} \\ \vdots & \vdots & & \vdots \\ C_{1k}^T & C_{2k}^T & \dots & C_{kk} \end{bmatrix}$$

where all $C_{i,j}$'s are circulant matrices (the diagonal ones also being symmetric). It makes most sense (and will make the programming easier) to consider these graphs only when all blocks have the same size. In theory it is also possible to choose diagonal blocks such that for each two sizes, one is a multiple of the other, but we will not consider such graphs here.

Where circulant graphs had the property that circular rotations were automorphisms, here we have that rotating every block by the same amount leads to an automorphism. (See figure 3.4)

The block-circulant graphs form a much larger class than the circulant graphs, capable of expressing many interesting graphs while at the same time being small enough to be enumerated.

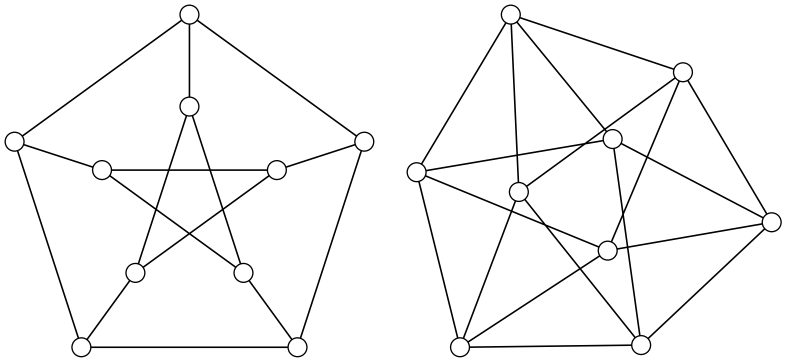


Figure 3.4: Two block-circulant graphs: the Petersen graph (2 blocks of size 5) and the Paley graph of order 9 (3 blocks of size 3)

It will also turn out that for many bounds on Ramsey numbers extremal graphs can be found which are block-circulant.

These graphs were already used by Exoo in search for Ramsey graphs [13], but these were all heuristic searches. Here we will go on in the same fashion as before and try to do full enumeration.

The main idea behind the algorithms remains unchanged: we perform a backtracking search over colour assignments to edge-classes and prune based on their Ramsey properties. These block-circulant graphs required however more care in the algorithmic details.

3.3.1 Avoiding isomorphisms

We will again do an effort to reduce the number of isomorphic graphs that are generated. Because block-circulant graphs have many degrees of freedom, we can't hope to eliminate all of them without using specialised algorithms. Yet it appears that there already are many 'internal isomorphisms' which can be avoided.

- Permutation of the blocks
- Relative rotation of the blocks
- Isomorphisms on a single block
- Permutation on the colours

Permutation of the blocks

In general, every permutation of the rows and columns of the adjacency matrix of a graph G leads to an isomorphic graph. But since we will only generate block-circulant graphs, we are only interested in isomorphisms which preserve the block-structure.

If $A = (C_{ij})_{1\leq i,j\leq k}$ is a block-circulant matrix, then every permutation of the diagonal blocks results in an isomorphic block-graph, which technically means that

$$\text{for all } \pi \in S_k : \; A = (C_{ij}) \mapsto A' := (C_{\pi(i)\pi(j)})_{1\leq i,j\leq k} \text{ is an isomorphism.}$$

An easy way to avoid these isomorphisms is to define a total ordering on all possible blocks and then only accept block-matrices for which the diagonal blocks are in increasing order. I.e.: $C_{11} \leq C_{22} \leq \cdots \leq C_{kk}$, for the chosen relation $\leq$. This has the extra advantage that it is possible to stop te search even if the whole matrix is not filled. Whenever the first violation in ordering is found this can never be restored.

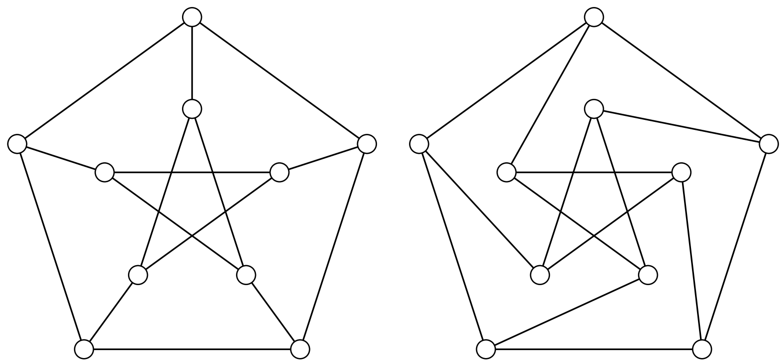
In our case, all blocks are circulant matrices which are represented by one bitstring. If we interpret these as natural numbers we can use the ordering on these numbers for our blocks. When using multiple colours, it is sufficient only to check the ordering in the first colour.

Relative rotation of the blocks

Since cyclical rotations are automorphisms on the diagonal blocks and all off-diagonal blocks are circulant it is possible to change the relative rotation of the components without actually changing the graph. It can either be thought of as rotating the blocks or rotating the connections between the blocks. Figure 3.5 illustrates this idea.

The larger the number of blocks k, the more important it becomes to be aware of this isomorphism, because all these rotations can then be combined. In the (really) worst case, this could result in $(\frac{n}{k})^{k-1}$ isomorphic graphs.

If for each component, we fix its rotation towards the first component, then we will avoid all isomorphisms of this kind. This means the bitstrings generating $C_{1,i}, 1 < i \leq k$, should be canonical in some sense. Since relative rotation of components is caused by rotation of these generating bitstring, we want to fix a specific rotation for each string.
If we call two bitstrings of length m equivalent precisely when the first can be obtained by a



$$\left[\begin{array}{c|c} (01001) & (10000) \\ \hline (10000)^T & (00110) \end{array}\right] \left[\begin{array}{c|c} (01001) & (01000) \\ \hline (01000)^T & (00110) \end{array}\right]$$

Figure 3.5: Two isomorphic block-circulant graphs obtained by changing the relative rotation of the blocks. The blocks in the adjacency matrices are depicted by their generating bitstring.

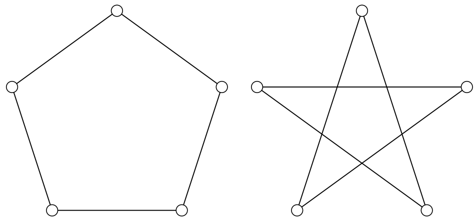
cyclical rotation of the second, then this forms an equivalence relation. The equivalence classes are known as *necklaces*. Now for our purposes we need to choose one representative from each class and make sure all generating strings of $C_{1,i}, 1 < i \leq k$ are such representative strings.

A well known set of representatives is the set of strings which are the lexicographically smallest of their class. These strings are called *Lyndon words*. This is how it is implemented in the eventual algorithm: every time a $C_{1,i}$ is generated, it is checked if we have a Lyndon word and backtrack in case it isn't. Some computationally cheap checks can be done on partial words to see if they could ever be extended to a Lyndon word, but it is perhaps an interesting question if other classes of representatives for necklaces exist which behave better towards being incrementally built.

Isomorphism on a single block

In section 3.1.3 we saw that different bitstrings of the same length can correspond with isomorphic circulant graphs and that the generation can be made canonical to some degree. One might therefore be tempted to believe that this can directly be applied to all diagonal blocks $C_{i,i}$, but this is not correct.

For example: the two graphs below are circulant and isomorphic, generated by `01001` and `00110` respectively.



There is however no way to build the Petersen graph as a block-circulant graph on two equal diagonal blocks (see figure 3.6). The problem is that the isomorphism between the two circulant graphs of order 5 is not circular with respect to $\mathbb{Z}_5$, which was actually the reason why we were interested in it in the first place. It is therefore not guaranteed that we can build a certain block-circulant graph using only canonical diagonal blocks.

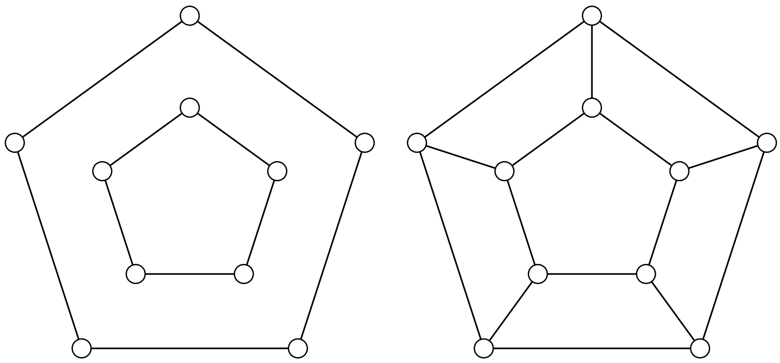


Figure 3.6: The graph on the left cannot be extended in circulant fashion to the Petersen graph. The only 3-regular possibility is up to isomorphism the graph on the right.

Permutation on the colours

When the Ramsey number considered avoids the same subgraph in different colours, then permuting these colours in a Ramsey graph does not change the fact that it is Ramsey. Therefore it is mostly desired to consider these graphs as the same. Two coloured graphs which are isomorphic but for the name of the colours are called *weakly isomorphic*.

We can avoid such weak isomorphisms by forcing the first edges used in each colour to be in the right order. That is, if colours k and l with $k < l$ avoid the same subgraph, then during generation we are only allowed to use colour l once k has been used. This solves these colour-permutation problems completely.

3.3.2 Search order

All occurring blocks are circulant matrices, and they are built using the same idea behind the generator for circulant graphs: a backtracking search over all possible bitstrings representing the block. Here we have the additional freedom to build the blocks in any particular order (but of course, we only fill in the upper triangle). While in theory the ordering used will not affect the resulting set of graphs, some orderings will be substantially better than others. For example: consider building all k diagonal blocks first. Since all connections between the blocks are yet to be assigned a colour this means that all blocks are chosen independently from each other (apart maybe from the ordering discussed in section 3.3.1). This is of course highly ineffective for a backtracking algorithm, since for example the first two blocks might already be impossible to connect, yet we try *every* possible assignment of the other diagonal blocks first.

It was therefore opted to use an ordering which ensures maximal connectivity between all coloured edges. The most logical approach seemed going down to the diagonal, i.e.:

$$C_{11} \to C_{12} \to C_{22} \to C_{13} \to C_{23} \to \dots$$

This starts with building the first diagonal block. Then the connections to the (unknown) second diagonal block are made, which can already cause cliques to appear. Only afterwards this second block is filled in, etc.

3.4 Searching between known interesting graph classes

For some Ramsey numbers, the extremal graphs show a nice structure, and were sometimes already known in other areas of graph theory. For example the unique (extremal) graph in $\mathcal{R}(J_7, J_4; 27)$ is also known as the *Schläfli graph*. It is a strongly regular graph (see definition below) which can also be encountered in finite geometry, for example as the 1-skeleton of the 2_{21}-polytope [1]. The *Clebsch graph* is another example which arises a number of times in Ramsey theory. That is why it might be interesting to search in large collections to find graphs which might not have been known to be extremal Ramsey graphs. For example, simply by looking for the Schläfli graph on Wikipedia, I found by accident the *Gosset graph*, which after some examination turned out to be a (J_8, J_5)-graph, which improved the current lower bound by 6. (This bound was later improved again, see section 3.5.) The Gosset graph is the 1-skeleton of

[1] Polytopes are multi-dimensional polygons and their 1-skeleton is the structure formed by the 1-dimensional edges.

the 3_{21}-polytope, so it seemed there might be a pattern here. Because the definition was rather obscure to me, the skeleton of the 4_{21}-polytope was hard to generate, but it turned out to be a (J_9, J_{17})-graph, being of seemingly little value here. Nevertheless it feels like graphs coming from polytopes might be interesting for Ramsey-purposes.

Many of the largest block-circulant graphs found with the generator appeared to be vertex-transitive. Therefore this class of graphs was further examined.

3.4.1 Vertex-transitive graphs

A graph is called *vertex-transitive* if for every $v, w \in V$ there exists an automorphism which maps v to w. This means that all vertices have the same role in the graph. It is easy to see that every circulant graph is vertex-transitive. On the other hand, every vertex-transitive graph with a prime number of vertices is circulant [26]. Many interesting block-circulant graphs also turned out to be vertex-transitive, so these classes are all somewhat related.

All vertex-transitive graphs up to 47 vertices have already been computed by Derek Holt and Gordon Royle [22] and this data was publicly available for download. (`https://staffhome.ecm.uwa.edu.au/~00013890/`) Even in compressed format, these were 12 gigabytes of graphs, so some care was required in processing them.

First of all, the set of vertex-transitive graphs is closed under complements, and complementary graphs have the same Ramsey properties, so we will only process the graphs with fewer edges than their complement. To verify the presence of certain cliques it is possible to limit the search to the neighbourhood of only one vertex because of the transitivity.

For each vertex-transitive graph G, the clique number $\omega(G)$ is computed exactly, which is a cheap calculation if there are few edges in G. However, if there are large independent sets in G it could be quite hard to calculate $\alpha(G)$ exactly so we will not do this automatically. Instead we search a k such that:

$$\underline{\mathrm{R}}(\omega(G)+1, k) \leq n < \underline{\mathrm{R}}(\omega(G)+1, k+1)$$

where $\underline{\mathrm{R}}$ denotes the current best-known lower bound on the Ramsey number R. Then all we need to check is if G contains a co-clique of size k, which is easier than computing the exact independence number. If the answer is no, we have found a new lower bound for $R(\omega(G)+1, k)$. If the answer is yes (which will most likely always be the case) it could still be that $\overline{G}$ does not contain a K_{k+1}-e. This is only relevant if $\underline{\mathrm{R}}(K_{\omega(G)+1}, K_{k+1}\text{-}e) \leq n$, hence this is only tested for such cases.

Some improvements over the lower bounds in [34] were found, including $37 \leq R(J_5, J_6)$, but only one which I did not find with earlier methods: $41 \leq R(J_5, K_6)$. It turned out however, that I simply had not yet investigated all of the possible parameter values for this case, and a block-circulant graph proving $42 \leq R(J_5, K_6)$ was found not much later.

I believe that vertex-transitive graphs certainly are very interesting to prove lower bounds on Ramsey numbers, but there are perhaps too many to enumerate them for the higher relevant orders.

3.4.2 Strongly regular graphs

A graph G is called **strongly regular** if it is regular of degree k and there exist $\lambda, \mu \in \mathbb{N}$ such that every two adjacent vertices share λ common neighbours, and every two non-adjacent vertices share μ common neighbours. If G has v vertices, then it is called an $srg(v, k, \lambda, \mu)$. Although many are, certainly not all of these graphs are vertex-transitive.

E. Spence and others computed complete lists of strongly regular graphs for some parameter-sets with $v \leq 64$ [8]. (See `http://www.maths.gla.ac.uk/~es/srgraphs.php`) All these graphs were downloaded and checked, but none of them (further) improved bounds on Ramsey numbers concerning K_n and J_n.

An upcoming book by Brouwer and Van Maldeghem gives an overview of many interesting strongly regular graphs coming from various families [6]. (The first author also maintains an online overview at `https://www.win.tue.nl/~aeb/graphs/srg/srgtab.html`.) It contains for example the creature called $NO_6^-(2)$, which is not a chemical molecule, but rather a graph defined by a quadratic form over a vector space. It is vertex-transitive, an $srg(36, 15, 6, 6)$ and is isomorphic to the only block-circulant graph found for $R(J_5, J_6; 36)$.
Many other interesting graphs in this book were also tested, some improving bounds on higher Ramsey numbers. (See table 3.1) It also contained a description of a graph denoted by $VO_6^-(2)$. It is vertex-transitive, an $srg(64, 27, 10, 12)$, and turns out to be a (J_5, J_7)-graph. Currently it does not appear to be isomorphic to a block-circulant graph. The neighbourhood of every vertex is isomorphic to the complement of the Schläfli graph, and for its complement this is $NO_6^-(2)$. For (J_5, J_7)-graphs of this order, the former property was necessary because of the unicity of the Schäfli graph. The latter could be seen as an argument that also $NO_6^-(2)$ is unique as extremal graph (see section 4.5), but this is certainly not a proof. Nevertheless, we find:

Theorem 3.6. $R(J_5, J_7) = 65$

Proof. The upper bound was proven by Pfender and Lidický with semi-definite programming [27]. The lower bound is achieved by the graph $VO_6^-(2)$ described above, which can also be found in appendix C. □

The existence of this $VO_6^-(2)$ also improves the lower bound on six other Ramsey-numbers in a trivial way: a (J_k, J_r)-graph is automatically a (J_{k+1}, J_r)-graph and so on. This can be seen in table C.2.

3.5 Results

Most of the above-mentioned algorithms and ideas were implemented into `C`-programs, capable of avoiding all kinds of interesting graphs. The source code can be found at `https://github.com/Steven-VO/circulant-Ramsey`.

3.5.1 Classical numbers

No bounds on the classical (clique-avoiding) Ramsey numbers were found, which was not so surprising given the many efforts already put into them. The best-possible bounds using circulant graphs up to 105 vertices had already been computed by Harborth and Krause [21], and for distance graphs exhaustive work has been done by Kuznetsov [25]. In 2015, Exoo and Tatarevic found many improvements on lower bounds based on circulant and block-circulant graphs, in combination with other metaheuristical techniques. [16]

For circulant graphs, mostly exhaustive searches were done going to the current upper bound. If this was unfeasible, then the search was stopped after reaching some consecutive non-existence results. For block-circulant graphs, not many cases could be *fully* checked exhaustively because of the sometimes many possible block-sizes (divisors of n). For example: if we want to prove that there are no block-circulant $(4, 8; 60)$-graphs, we have to check for 11 different block-sizes, including graphs consisting of 30 blocks of size 2. This last case is so close to a general (non-constrained) graph that it certainly cannot be exhaustively enumerated in this way. In some sense these aren't really block-circulant graphs any more.
On the other hand, if n is prime or has very few divisors, chances are also low to find a block-

circulant Ramsey graph. For example $R(5,5) \geq 43$, a prime number, so no strict block-circulant graph can higher this bound by 1.

Some test were done for multi-colour Ramsey numbers, mostly with circulant graphs, but no new bounds were found. With block-circulant graphs this was not tested extensively.

3.5.2 Almost-cliques

The most effort in improving bounds was put in Ramsey numbers concerning cliques with one edge missing, i.e. $K_n - e$ $(= J_n)$. It appeared that our methods worked well on these cases, for at least 15 improvements on lower bounds were found.

For most cases, we searched for the best-possible circulant graphs. Where necessary and feasible, distance-graphs were used to validate their maximality. It was surprising to see for how many values block-circulant graphs could reach the best-know lower bounds. For the strict (J_n, J_m)-cases, they seemed almost always to suffice, but for the mixed (J_n, K_m)-cases they sometimes fell just short. It was also observed that the extension algorithm could in many cases add an extra vertex except for one edge to be coloured.
Table 3.1 contains the improved bounds for these cases. They were all found with techniques described earlier, except for $30 \leq R(4\text{-e}, 8\text{-e})$. This was found by starting from the Schläfli graph and considering all ways in which it could be extended to a (J_4, J_8)-graph. This process could be repeated no more than twice.
It was also encountered a number of times that there was quite a large gap between consecutive existing block-circulant Ramsey graphs. This caused that sometimes I discovered some graphs only very late, and this leaves it hard to ever be certain that no larger block-circulant graphs exist (unless of course, if it was possible to reach the upper bound).

An overview of all achieved results and non-results for circulant and block-circulant graphs (within the shown ranges) can be found in table C.1. Results achieved with all techniques (including special graphs and extension-methods) are shown in table C.2. Since there are many cases, no more than a couple of CPU-days was put in each of them. This work was done for the most part on the Stevin high performance computing infrastructure hosted at Ghent University. The correctness of the witnessing graphs was largely verified with Sage (where this was feasible), and with a subgraph-detecting program written by Gunnar Brinkmann.

Old bounds	New LB	Old LB reference	Method
$47 \leq R(3, 12\text{-}e) \leq 53$	49	Exoo (2013), implied by $R(3, 11)$ [15]	Block-circulant
$60 \leq R(3, 14\text{-}e) \leq 71$	61	[Kol] (2015), implied by $R(3, 13)$ [24]	Block-circulant
$29 \leq R(4\text{-}e, 8\text{-}e) \leq 32$	30	implied by $R(4\text{-}e, 7\text{-}e) = 28$	extension$\times$2
$29 \leq R(4\text{-}e, 8) \leq 39$	36	implied by $R(4\text{-}e, 7) \geq 28$	Block-circulant
$34 \leq R(4\text{-}e, 9\text{-}e) \leq 46$	36	Exoo (2000) [14]	Block-circulant
$42 \leq R(4\text{-}e, 11\text{-}e) \leq 86$	44	Wang e.a. (1998), by $R(3, 11\text{-}e)$ [40]	Block-circulant + extension
$74 \leq R(4\text{-}e, 16\text{-}e)$	82	Implied by $R(3, 15)$	Strongly regular graph
$59 \leq R(4, 9\text{-}e) \leq 105$	62	Ex. & Tat. (2015), by $R(4, 8)$ [16]	Circulant
$31 \leq R(5\text{-}e, 6\text{-}e) \leq 38$	37	Exoo (2000) [14]	Block-circulant
$37 \leq R(5\text{-}e, 6) \leq 53$	42	Exoo (2000) [14]	Block-circulant + extension
$40 \leq R(5\text{-}e, 7\text{-}e) \leq 65$	**65**	Exoo (2000) [14]	Strongly regular graph
$59 \leq R(5\text{-}e, 8) \leq 123$	69	Implied by $R(4, 8)$	Extension$\times$4
$101 \leq R(5, 9\text{-}e) \leq 275$	121	Implied by $R(5, 8)$	Strongly regular graph
$80 \leq R(6\text{-}e, 8\text{-}e) \leq 218$	83	Implied by $R(5, 7)$	Circulant

Table 3.1: Most (non-trivially) improved lower bounds for almost-clique cases

3.5.3 Wheels

In the paper of Pfender and Lidický, some cases about wheel-graphs are considered, of which most of them didn't have bounds already [27]. Using circulant graphs, a first lower bound can be established very fast, and sometimes it could be improved using block-circulant graphs. Some cases were exactly determined in this way. This was in some sense a little unexpected because wheels consist for the most part out of a cycle, and one can easily see that circulant graphs are probably not so good at avoiding monochromatic cycles.

Old bounds	New LB	Method
$R(W_7, W_4) \leq 21$	**21**	Block-circulant
$R(W_7, W_5) \leq 16$	13	Circulant
$R(W_7, W_6) \leq 19$	**19**	Circulant
$R(W_7, W_7) \leq 19$	**19**	Block-circulant
$R(W_8, W_4) \leq 26$	22	Circulant
$R(W_8, W_5) \leq 17$	**17**	Circulant
$R(W_8, W_7) \leq 21$	19	Circulant
$R(W_8, W_8) \leq 25$	22	Circulant
$R(K_6, W_6) \leq 40$	33	Circulant
$R(K_7, W_6) \leq 55$	43	Block-circulant
$43 \leq R(K_7, W_5) \leq 50$	45	Block-circulant
$39 \leq R(K_{11}, C_4) \leq 44$	40	Block-circulant
$30 \leq R(K_3, K_4\text{-}e, K_4) \leq 40$	31	Block-circulant
$28 \leq R(K_4, K_4\text{-}e, C_4) \leq 36$	29	Block-circulant

Table 3.2: Established lower bounds for cases on wheels and other graphs. The upper bounds are from [27].

Chapter 4

Upper bounds on Ramsey numbers

While lower bounds on Ramsey numbers can be established by giving one graph which does not have the desired property, to prove an upper bound one must give an argument implying that *all* graphs of a certain order have the desired property.
Mostly, this is done by using general or specific theorems to vastly reduce the number of possible counter-example graphs. The remaining graphs must then be enumerated by a computer to verify that none of them is a Ramsey graph. In general, this is much harder than proving a lower bound, since a correctly implemented program designed to prove no graphs with certain properties exists, should be able to find one if there would be.

In this chapter we will first talk about general methods to derive upper bounds, both algorithmic and theoretical. These will then be applied to two cases: $R(K_3, J_4, J_4)$ and $R(J_5, J_6)$.

4.1 Direct Enumeration

Proving that $R(k_1, k_2) \leq n$ is equivalent to showing $\mathcal{R}(k_1, k_2; n) = \emptyset$. Yet sometimes (as we will see) it is very handy to know the exact sets of Ramsey graphs for smaller orders, meaning $\mathcal{R}(k_1, k_2; m)$ for all $m < n$.
If a graph G is a Ramsey-(k_1, k_2)-graph, then $\forall v \in V\colon\ G \setminus \{v\}$ is also a (k_1, k_2)-graph, because no illegal subgraphs can be formed by removing a vertex. Therefore, we could in theory find all Ramsey graphs with the following algorithm that constructs larger Ramsey graphs from smaller ones:

Starting from $\mathcal{R}(k_1, k_2; 1) = \{K_1\}$, we recursively compute $\mathcal{R}(k_1, k_2; n)$ by considering every

Ramsey-proper way a graph $G \in \mathcal{R}(k_1, k_2; n-1)$ can be extended by one vertex. That means, we add a vertex and look for all colour assignments to the newly-added edges such that the result does not contain forbidden cliques. Then, all isomorphic copies are removed from $\mathcal{R}(k_1, k_2; n)$. Repeat this process until we find an n with $\mathcal{R}(k_1, k_2; n) = \emptyset$.

Since the number of Ramsey-graphs for a specific case typically tends to increase very rapidly with increasing size of graphs, and only starts dropping close to the exact value of the Ramsey number, this method is only suitable for the smallest cases.

One of the quickest ways to implement the above algorithm with reasonable efficiency is perhaps with *geng*, a program from the isomorphism-testing package *nauty* [28]. This program is able to build graphs having certain properties without generating isomorphic copies. With *geng*, $\mathcal{R}(K_4, K_4)$ can be computed in a matter of minutes, while experiments with modulo parameters show that $\mathcal{R}(J_5, J_5)$ would probably already take years. ($R(4,4) = 18$ and $R(J_5, J_5) = 22$)

4.2 Graph splitting

For multi-colour Ramsey numbers, another way to determine (both upper- and lower-)bounds is to use results for two-colour numbers. A straightforward way to do this is by being 'colour-blind' for some colours, resulting in a two-colour problem. More specifically, if $R(k_2, k_3) = n$, then $R(k_1, k_2, k_3) \leq R(k_1, n)$, because if we combine the red and green edges of a Ramsey (k_1, k_2, k_3)-colouring into one new colour, this colour cannot contain a monochromatic K_n.

For example: $R(4,5) = 25$. Therefore, every Ramsey $(3,4,5)$-graph has the property that merging the second and third colour into one common colour will lead to a Ramsey $(3,25)$-graph, hence $R(3,4,5) \leq R(3,25)$. Of course, this value could be far too high, because the implication in the above reasoning certainly does not necessarily hold in the opposite direction. Namely if G is a Ramsey $(3,25)$-colouring, then it may well be that there is no possible way to split the second colour in two new colours without making a forbidden K_4 or K_5.

Definition 4.1. *We call a two-coloured graph G **splittable** (for a certain Ramsey property (k_1, k_2, k_3)) if there is a way to recolour the red edges of G into red and green edges without making a red K_{k_2} or a green K_{k_3} .*

Now, if we verify that no graph in $\mathcal{R}(k_1, R(k_2, k_3); n)$ is splittable, we can further improve the upper bound to n.

This technique was used with success for the case of $R(K_3, K_3, J_4) = 17$ by Shetler, Wurtz and Radziszowski [36]. Since $R(K_3, J_4) = 7$, they needed to check every $(3, 7; 17)$-graph for splittability. Since this set was not known at the time, and would perhaps have been too large to process, they reduced the number of cases by closer examining the arrowing relation for $R(K_3, J_4)$. They found that $J_7 \rightarrow (K_3, J_4)$, meaning that only all $(K_3, J_7; 17)$-graphs needed to be tested for splittability.

4.3 Gluing Ramsey graphs

Another common way to establish upper bounds is by using the idea behind theorem 2.6. It looks at a (hypothetical) Ramsey (k, r)-graph and observes that the neighbourhoods of vertices must be Ramsey graphs for smaller parameters: Suppose G is a Ramsey-(k, r)-graph. Then for every $v \in V$, $N_G^1(v)$ is a Ramsey-$(k-1, r)$-graph and $N_G^2(v)$ a Ramsey-$(k, r-1)$-graph. For non-complete avoided graphs H, the sub-cases are less unambiguous, but we will write them symbolically as $H{-}v$. (This has however no relation to any specific vertex v being in the graph or not.) For this work it is sufficient to know that $J_k{-}v = J_{k-1}$.

To know if a Ramsey-$(k, r; n)$-graph exists, we could now consider a vertex v and connect it with blue edges to every possible graph $H_1 \in \mathcal{R}(k-1, r)$ and with red edges to every possible $H_2 \in \mathcal{R}(k, r-1)$, so we enumerate all suitable neighbourhood-graphs of v. Of course, to obtain a graph on n vertices we need to select H_1 and H_2 such that $|H_1| + |H_2| = n - 1$. There are obviously at most n pairs of orders for which this is true, and upper bounds on $R(k-1, r)$ and $R(k, r-1)$ will further reduce this number.

For example: imagine we want to prove that $\mathcal{R}(4, 5; 28) = \emptyset$ so that $R(4, 5) \leq 28$. Suppose that we already know that $R(4, 4) = 18$ and $R(3, 5) = 14$, and that we have all Ramsey-graphs for these cases. Then the possible ways to pick a $(3, 5)$-graph of order n_1 and a $(4, 4)$-graph of order n_2 are:

n_1	$\lvert\mathcal{R}(3, 5; n_1)\rvert$	n_2	$\lvert\mathcal{R}(4, 4; n_2)\rvert$	combinations
13	1	14	130816	130816
12	12	15	640	7680
11	105	16	2	210
10	313	17	1	313

Table 4.1: All cases to be checked for $R(4, 5) \leq 28$

This gives a total of 139 019 combinations for both subgraphs. For each of these cases we need to check if the partially coloured graph which corresponds with it can be completed to a true Ramsey $(4, 5)$-graph. This is equivalent to asking if the edges between H_1 and H_2 can be assigned a colour without constructing a blue K_4 or a red K_5. We will call this process **gluing** the graphs H_1 and H_2.
Now, how could we check if such gluings exist?

A first idea would be to do a backtracking search over the edges: one by one you give them a colour, and go back if you reach a dead end. This approach will however risk to be computationally infeasible even for very small Ramsey numbers because it does not use enough of the structure that is present in the subgraphs. For the example cases in table 4.1, between 170 and 182 edges need to be assigned a colour, thus requiring a very deep recursion tree.

We constructed our partially coloured graph by observing that $N^1(v)$ is a $(k-1, r)$-graph and $N^2(v)$ is a $(k, r-1)$-graph. This must of course be true for every vertex w in the eventual graph obtained after gluing. We can use this information to speed up the gluing process itself: Because we allow no blue k-cliques, we must have that $\forall x \in H_1 : N^1_G(x) \cap H_2$ contains no blue $(k-1)$-clique, and similar for the red neighbourhood of x. Such a partition of H_2 induced by this colouring is called a **(feasible) cone**. The problem is now reduced to assigning to each $x \in H_1$ a feasible cone such that no illegal graphs are formed.

This has a lot of advantages over a naive backtracking procedure:

- The search tree has much smaller depth (but a higher branching factor).
- Out of all pairs of subgraphs (H_1,H_2) needed to be checked many share the same H_2 (see for example Table 4.1). Of course, the set of feasible cones only depends on H_2 and not on H_1, meaning that we only need to compute them once for every H_2.
- Since the cones are now objects on their own, they can be taken together into sets of cones, relations can be defined over them, etc.
- The problem is made more abstract, so other optimisations (for example regarding automorphisms) are easier to find.

Such cone searches have already been used many times before to derive upper bounds on Ramsey numbers, or to enumerate full Ramsey sets. ([17], [30], [33],. . .) An example of a gluing algorithm with just some of the parameters filled can be found in algorithm 5, but this will be explained later in more detail.

4.3.1 Edge-counting constraints

While such an approach with cones is already better than simple backtracking, it is still far from being fast enough to tackle the actual open bounds on classical Ramsey numbers. Some additional framework is needed to further reduce the amount of computations done.

In [39], a method is described to eliminate the existence of certain gluings using mathematical properties. This is done by counting edges and triangles in a hypothetical Ramsey-graph. The idea is that both $N_G^1(v)$ and $N_G^2(v)$ in such a Ramsey graph must have a nearly-extremal number of edges in order to be possibly glued together.

Definition 4.2. *For any coloured graph $G = (V, E, c)$, the number of edges coloured i is denoted by $e^i(G) = |\{e \in E \mid c(e) = i\}|$. The maximal number of edges in a Ramsey colouring will be written as: $E^i(H_1, H_2; n) = \max\{e^i(G) \mid G \in \mathcal{R}(H_1, H_2; n)\}$*
$n_i := |\{v \in V| \ |N_G^1(v)| = i\}|$ is the number of vertices having degree i.

In the best case (as we will see), for each $v \in V$, $N_G^1(v)$ has an extremal number of blue edges and $N_G^2(v)$ an extremal number of red edges. This is expressed in the *deficiency* function δ.

$$\delta_G(v) := E^1(H_1{-}v, H_2; \deg(v)) - e^1(N_G^1(v)) + E^2(H_1, H_2{-}v; n-1-\deg(v)) - e^2(N_G^2(v))$$

Since this deficiency is the sum of two positive differences, it is clear that for every $G \in \mathcal{R}(H_1, H_2; n)$, and every $v \in V$: $\delta_G(v) \geq 0$ and hence also $\sum_{v \in V} \delta_G(v) \geq 0$.

This counting of edges in neighbourhoods can also be seen as counting monochromatic triangles. Namely if $w, z \in N_G^1(v)$ and $\{w, z\}$ is coloured blue, then $\{v, w, z\}$ forms a blue triangle. Conversely every blue triangle containing v will be of this kind. Of course, the same holds for red edges in $N_G^2(v)$. If we sum this over all vertices, every triangle will be counted exactly three times. Therefore, we can compute the number of monochromatic triangles $T(G)$ in G via:

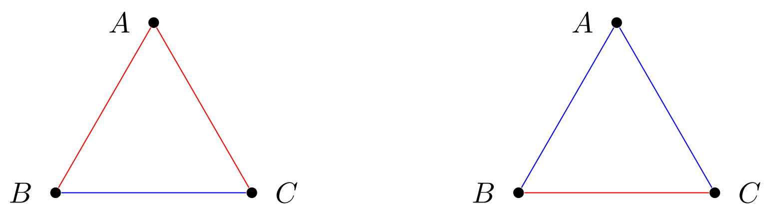
$$\begin{aligned}
3 \cdot T(G) &= \sum_{v \in V} [e^1(N_G^1(v)) + e^2(N_G^2(v))] \\
&= \sum_{v \in V} [E^1(H_1{-}v, H_2; \deg(v)) + E^2(H_1, H_2{-}v; n-1-\deg(v)) - \delta_G(v)] \\
&= \sum_{i=0}^{n-1} n_i \cdot [E^1(H_1{-}v, H_2; i) + E^2(H_1, H_2{-}v; n-1-i)] - \sum_{v \in V} (\delta_G(v)) \\
\Rightarrow \sum_{v \in V} \delta_G(v) &= \sum_{i=0}^{n-1} n_i \cdot [E^1(H_1{-}v, H_2; i) + E^2(H_1, H_2{-}v; n-1-i)] - 3 \cdot T(G) \qquad (4.1)
\end{aligned}$$

This means that if we have the information about the number of edges in the subcases, the sum of the deficiencies can be computed from the triangle count if ones knows the degree sequence of the graph. Surprisingly, also the triangle count can be fully determined by the degree sequence alone. This was first proved by Goodman in 1959. [18]

Theorem 4.3 (Monochromatic triangle count)**.**

$$T(G) = \binom{n}{3} - \frac{1}{2}\sum_{i=0}^{n-1}[n_i \cdot i \cdot (n-1-i)]$$

Proof. Let G be a 2-coloured K_n. We will use double counting to enumerate all triples of vertices which do not form a monochromatic triangle. This can only happen in one of two cases:



Hence in every non-triangle there is a blue edge adjacent to a red edge. Conversely, every blue edge adjacent to a red edge extends to a single non-triangle.

Now for each $v \in V$, there are $\deg(v) \cdot (n-1-\deg(v))$ pairs of edges containing v which can be made consisting of a blue and red edge. Summing over V will count every non-triangle exactly twice: once in B and once in C in the picture. $\square$

For a moment, I thought this framework could be extended to more colours: the counting of i-edges in $N^i(v)$ still counts monochromatic triangles. However, starting from 3 colours, the degree sequence no longer determines the number of triangles, because in the proof of theorem 4.3 some non-triangles will be counted three times and others two times, depending on the amount of colours in the non-triangle.

The smallest counter-example I could find were the two graphs in figure 4.1, both with degree sequence:

$$\begin{pmatrix}1\\1\\2\end{pmatrix}, \begin{pmatrix}1\\1\\2\end{pmatrix}, \begin{pmatrix}2\\0\\2\end{pmatrix}, \begin{pmatrix}2\\0\\2\end{pmatrix}, \begin{pmatrix}2\\2\\0\end{pmatrix}$$

the first containing a blue triangle and the second containing no triangles.

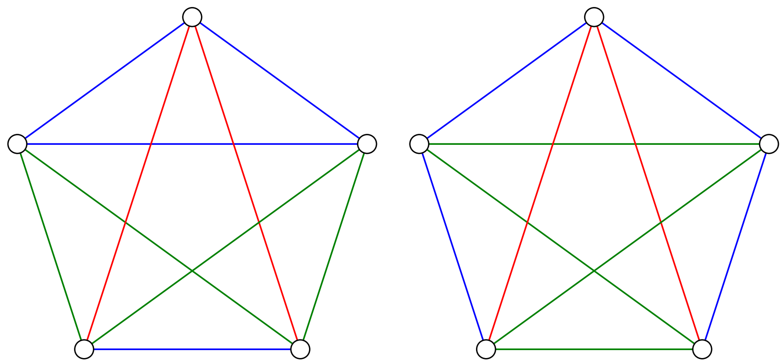


Figure 4.1: Two graphs with the same degree sequence but a different number of monochromatic triangles.

4.3.2 Restricting the search

With the knowledge of the maximum number of edges in the underlying Ramsey problems, it is now possible to eliminate certain degree sequences for hypothetical Ramsey graphs. We already knew that all degrees must lie within a range defined by the Ramsey number of the underlying problems. More specifically for a $(k_1, k_2; n)$-graph G, all degrees must satisfy

$$n - R(k_1, k_2 - 1) \leq \deg(v) < R(k_1 - 1, k_2)$$

If we are looking for graphs with specified degree sequences (for example regular graphs), there is a possibility that we can skip the search altogether because the sum of the deficiencies (as computed in (4.1)) is negative.

Even if the sum is positive, we can still get valuable information about possible graphs. For example: a 10-regular (J_5, J_5)-graph G of order 22 has $\sum_{v\in V} \delta_G(v) = 22$. Since $(\forall v \in V)(\delta_G(v) \geq 0)$ we know $(\exists v \in V)(\delta_G(v) \leq 1)$. That means that there is at least one vertex for which the neighbourhoods have a nearly extremal number of edges. We will exploit this information by only building G from this special vertex and therefore only try to glue H_1 and H_2 if together they lack at most one edge.

Now, in general we won't know the exact degree sequence of the possible Ramsey graphs, but once we see that the deficiency of G is actually a linear function over the degree sequence, it is quite easy to find the extrema. Nevertheless a simple program was written to double-check the reasoning. It enumerates every degree sequence which can effectively be realised by a graph and for which the degrees lie within the correct range, and calculates the deficiencies.

4.4 R(K_3,J_4,J_4)

In the paper of Pfender and Lidicky [27] many new upper bounds for all kinds of Ramsey numbers were obtained by semi-definite programming, with some of them only leaving very small gaps between lower and upper bound. One case seemed particularly interesting to me, $21 \leq R(K_3, J_4, J_4) \leq 22$, because not only are the forbidden subgraphs very small, the Ramsey number itself is not too large as well.

It should also be mentioned that efforts to increase the lower bound failed: no circulant, block-circulant or distance-graphs were found. A local search algorithm which drops and adds vertices produced 70 non-isomorphic $(K_3, J_4, J_4; 20)$-graphs, but none of them could be extended further.

4.4.1 Graph splitting

The first technique I considered was the splitting approach. For the case of $R(K_3, J_4, J_4)$ we could merge colours in two different ways. The most logical one would be to merge the last two colours. Since $K_{10} \rightarrow (J_4, J_4)$ it would be sufficient to generate all Ramsey-$(3, 10)$-colourings on 21 vertices and verify that no such one is splittable. It is also easy to see that we can restrict this search to *maximal-triangle-free* graphs: graphs without triangles such that the insertion of any new edge would create a triangle. These graphs can be generated efficiently by the program *triangleramsey* made in [5]. However, the number of such graphs was calculated to be 2 815 960 999, which is far too much to handle.

As is done in [36], we could make the arrowing more precise in order to lower the amount of graphs to be verified. We have that $K_9 \nrightarrow (J_4, J_4)$, and with a simple vertex-add method it was found that the unique such Ramsey-colouring could only be extended by at most 5 edges. This implies that $K_9 + 6e \rightarrow (J_4, J_4)$. Based on computations using *triangleramsey*, we estimate however that the number of *mtf*-graphs of this latter kind are about $2 \cdot 10^9$, which is still too much to process. Therefore it seems that graph splitting is not a viable approach to calculate this Ramsey number.

4.4.2 Graph gluing

To find Ramsey (K_3, J_4, J_4)-graphs $G = (V, E, c)$ by gluing, we first need to compute the sets of Ramsey graphs for the subcases. Those are $\mathcal{R}(K_2, J_4, J_4)$ and $\mathcal{R}(K_3, J_3, J_4)$, and are small enough to be enumerated completely with the *vertex-add* method from section 4.1 (even in a language like Python). More specifically, I computed all necessary subgraphs in Sage [37], which has useful built-in tools to work with graphs and isomorphisms of edge-colourings. The generation didn't take longer than 10 seconds.

Now, if there were a (K_3, J_4, J_4)-graph on 21 vertices and we composed the graph into (v, G_v, H_v, I_v), these subgraphs would all be in Table B.1. This means that we could work backwards and try every combination of suitable graphs in the table which sum to 20. That is: we are looking for all triples (n_1, n_2, n_3) such that $n_1 \leq 9$, $n_2, n_3 \leq 10$ and $n_1 + n_2 + n_3 = 20$. This would lead to 55 triples which, when multiplied with the number of graphs on the given orders, leads to 1 160 407 cases to be checked.

We can however do a little bit better than this. Currently we chose the vertex v at random, but we could pick a more specific one to avoid ending in isomorphic graphs. For example: we could choose v to have the maximum degree in the blue (K_3-avoiding) colour. This already allows us to eliminate some cases.

Theorem 4.4. *For every $G \in \mathcal{R}(K_3, J_4, J_4; 21)$, the maximum degree in the first colour is at least 2.*

Proof. Turan's theorem states that a graph G on n vertices with more than $(1 - \frac{1}{r-1}) \cdot \frac{n^2}{2}$ edges must have clique number at least r.[38] As discussed before, the union of the red and green edges cannot contain a K_{10}. This implies that there are at least $\frac{21 \cdot 20}{2} - \frac{8}{9}\frac{21^2}{2} = 14$ blue edges.

This is impossible if the maximum degree in the blue colour equals 1. Note that this approach is already not strong enough to even prove that the maximum degree is bigger than 2. $\square$

Further, it is easy to see that, if a Ramsey $(K_3, J_4, J_4; 21)$-graph exists, there also exists one where $|N_G^2(v)| \geq |N_G^3(v)|$, because we can always swap all red and green edges.
These two results together leave about 739 771 cases to be checked for gluing.

Naive backtracking

Since the Ramsey number considered here is rather low and the avoided subgraphs are very small, I thought at first that the gluing would be doable with a quite naive backtracking approach.
A program was written which considers a graph composed out of three blocks. One edge at a time, it then tries to colour all remaining edges. A good ordering was computed for every

instance and then fixed for the entire backtracking procedure. The bounds on the degrees in the different colours were also respected. For quite some cases, this could quickly determine that no gluing was possible. However, for far too many configurations the backtracking reached a recursion depth of more than 100, and since we are in a three-colour case, branch pruning is scarce. Together with the large number of cases, this caused the approach to be infeasible.

Cones

An effort was done to devise a cone-search algorithm for this problem. Since we now have three colours it is not straightforward what these cones should be.
When building the graphs from (v, G_v, H_v, I_v) three interconnections must be specified. It was opted to first glue G_v to H_v and consider the result as a partially coloured graph. Then, the new set of cones should be computed and a second gluing would be needed. Due to the small graph sizes, these first gluings could be found at a relatively fast pace. However, because the result is largely uncoloured, this resulted in far too many graphs to start the second phase of the gluing.

Here are some ideas to solve these problems: The first gluing should probably already incorporate the fact that another gluing has to be done. (For example: that enough cones should still be feasible, that the degrees are not too large, etc.)
It would also help to avoid isomorphisms already during the backtracking phase. If we consider each feasible cone as a colour, then the gluing algorithm in fact becomes a heavily-constrained vertex-colouring problem. For such problems there already exist methods to avoid isomorphic colourings.

With these two and perhaps more optimisations, I believe there is a chance to solve this case. However, before I could try them and completely make out if it were possible or not, another opportunity came at hand.

4.5 R(J_5,J_6)

With the block-circulant graph generator a (J_5, J_6)-graph on 36 vertices was found. (See table 3.1) This brought the lower bound from 31 to 37, very close to the best-known upper bound of 38 from [27]. Therefore an effort was done to lower it.

Full enumeration of $\mathcal{R}(J_5, J_6)$ is impossible, because this set is certain to be larger than $\mathcal{R}(K_4, K_5)$ for which only very specialized algorithms could determine this set for some orders. [2] Thus, the main idea was to use the gluing method as described before. For this case the necessary subgraphs are (J_4, J_6)- and (J_5, J_5)-graphs. Normally gluing procedures on subgraphs of this complexity could already risk being computationally infeasible, but here we have the advantage that the lower bound is very close to the simple theoretical upper bound derived from theorem 2.6.

$$R(J_5, J_6) \leq R(J_4, J_6) + R(J_5, J_5) = 17 + 22 = 39$$

so we only need to lower the bound by two. For this we need the sets $\mathcal{R}(J_4, J_6; 15, 16)$ and $\mathcal{R}(J_5, J_5; 20, 21)$. These sets have both already been computed, the first in [32], and the second in [33]. This was however a very long time ago, and the second set was lost after all this time. Therefore it was necessary to compute $\mathcal{R}(J_5, J_5; 20, 21)$ myself, but this also served as an independent verification of the results in [33], and as a warm-up exercise.

4.5.1 R(J_5,J_5)

The case of $\mathcal{R}(J_5, J_5)$ might already seem more tempting to solve by brute force, but it still is computationally infeasible. Quick estimates showed that there most likely are billions of (J_5, J_5)-graphs, so full enumeration is impossible here. (J_4, J_5), the only subcase, could however be done by the simple vertex-extend algorithm of 4.1 in Sage in less than a minute. The counts matched completely to the ones reported in [33], giving 1623 graphs in total (Table B.2). This already indicates the vast speed with which Ramsey problems tend to get more difficult.

To start with (J_5, J_5), a general gluing algorithm based on cone search was implemented. As before, this means that we have a vertex v, a (J_4, J_5)-graph G_v as $N^1_G(v)$ and a (J_5, J_4)-graph H_v as $N^2_G(v)$. Here, a cone is a subset of vertices of H_v which does not contain a blue J_4 (or 3 vertices of a K_4) and for which the complement does not contain a red K_3. This is drawn symbolically in figure 4.2.
We take advantage of the independence between G_v and H_v and compute the list of feasible cones $\mathcal{C}$ for each H_v only once. Then for each G_v, the gluing process is started with this information. That is, a backtracking algorithm tries to assign a cone C_x to every vertex $x \in G_v$. In each step it is checked if C_x forms a J_5 with some of the $C_y, y < x$. (See algorithm 5)

These constraints on pairs of cones are already a bit complicated because in contrast to K_5, there are many cases in which a J_5 could arise from gluing: the five vertices can be distributed among G_v and H_v, the missing edge could either be in G_v, in H_v or in the interconnection…

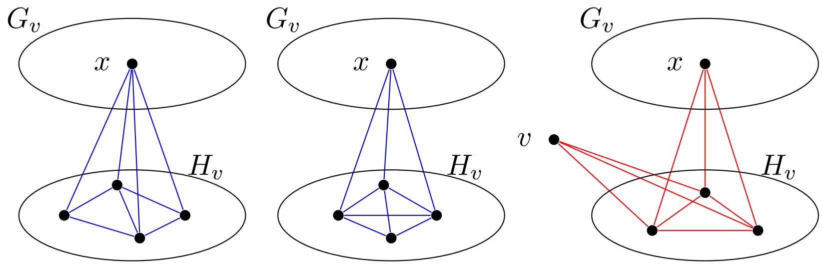


Figure 4.2: The three restrictions on the cones

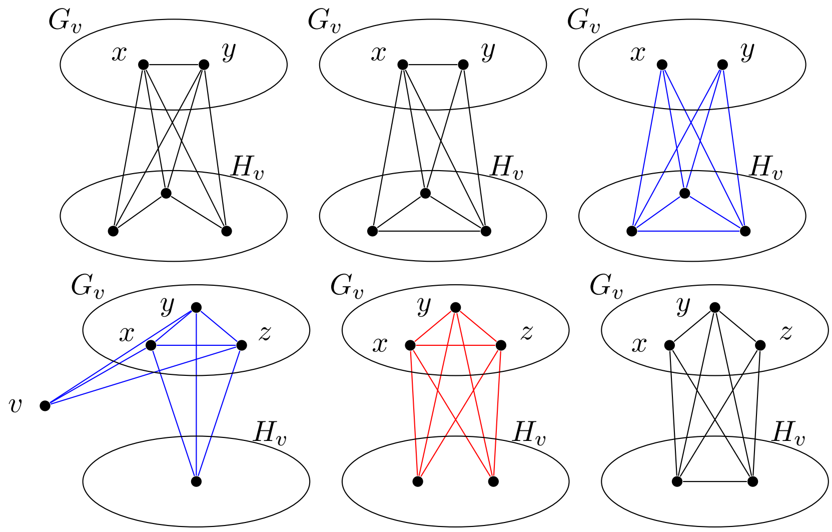


Figure 4.3: Some illegal cases for combinations of cones

Also in contrast to K_5, v can be a part of an illegal gluing since J_5 is non-complete. Some of these cases are shown in figure 4.3, but the more complicated ones are left behind (and those were mostly not tested *as such* in the implementation).

Although the number of possible cones mostly was quite moderate (ca. 100-300), the vast number of case-checks caused this approach to be far too slow. The most time-consuming part of the algorithm was in this case the continuous testing whether the intersection of two cones contained a J_3 . This was solved by precomputing this binary relation over all possible cones: $C_x \sim C_y \Leftrightarrow C_x \cap C_y$ contains no J_3. This relation is also independent of G_v, so it can be calculated right after the cones and be re-used.

To gain even more speedup, this relation was implemented as a list of bitsets. Then, for every $x \in G_v$, one needs only to consider those cones C_x for which there exist no $y \in G_v, y \sim x$ such that $C_y \nsim C_x$. This is done efficiently by starting from a bitset $S = \{1, \ldots, |\mathcal{C}|\}$ and then iteratively taking intersections with the precomputed relation.

Algorithm 5 Gluing Ramsey graphs

```
function FIND_ALL_RAMSEY_GRAPHS(size)
    for n_2 ∈ {10, 11, 12} do
        for each Graph H_v ∈ R(J_5, J_4; n_2) do
            cones = CALCULATE_CONES(H_v)
            for each Graph G_v ∈ R(J_4, J_5; size − 1 − n_2)  do
                GLUE_GRAPHS(G_v,H_v, cones)
            end for
        end for
    end for
end function

function GLUE_GRAPHS(G_v, H_v, cones)
    ASSIGN_CONE(G_v, H_v, cones, 0)
end function

function ASSIGN_CONE(G_v, H_v, cones, x)
    if x = |G_v| then
        Check if cones form Ramsey graph and save in that case.
        return
    end if
    for each Cone C ∈ cones do
        C_x ← C
        for all y < x do                          ▷ Check if this assignment is allowed
            if C_y and C_x form illegal subgraph then
                NEXT(C)
            end if
        end for
        GLUE_GRAPHS(G_v, H_v, cones, x + 1)       ▷ No conflicts: go deeper in recursion
    end for
end function
```

Also, it was precomputed which subsets of H_v contain no edge, since this was a check which had to be done many times.

Changing the vertex ordering

For backtracking procedures, the order in which the search tree is traversed is in general very important. For example the *degree heuristic* proposes to always do the vertex with the least amount of leftover options first. Here this would mean the vertex $x \in G_v$ with the smallest amount of non-conflicting cones. If there would be such a vertex with 0 options, we can backtrack right away.

However, the complexity of the constraints in this problem make it hard to estimate the number of leftover cones, let alone compute them exactly during runtime. There are however very general heuristics which could help here. Namely, vertices $x \in G_v$ lying in a triangle generally have more constraints on them than vertices who don't. That means that during runtime, it will be *likely* that they have fewer leftover cones. Also, vertices lying in a red K_4 or being the centre of many J_3's will be more constrained.

It was tried to compute for each G_v, before the start of the gluing algorithm, a good ordering of the vertices based on the previous ideas. This was done with a heuristic algorithm assigning values to the number and heaviness of the constraints in the consecutive partial graphs $G[\{1, \ldots, x\}]$.

The details of the algorithm will not be explained here, because it did not work out as expected. For some cases the runtime became longer while for other it went shorter, giving on average no significant speed-up. This could either mean the above ideas are somewhat *too* general, or there perhaps were some technicalities I was missing.

Edge-counting constraints

Luckily, using the framework around edge-counting explained in section 4.3.1 did result in a huge speed gain. With the information about the number of edges in (J_4, J_5)-graphs, the possible degree sequences and their deficiencies were computed with the simple program mentioned earlier. The results are shown in Table 4.2.

The search was then carried out as described in section 4.3.2: the gluing process was only

n	18	19	20	21	22	23
maximal deficiency	108	85	60	42	22	11
$\Rightarrow \exists v : \delta(v) =$	6	4	3	2	1	0

Table 4.2: Maximal deficiency for (J_5, J_5)-graphs of specified order.

started if G_v and H_v had low combined deficiency. The symmetry of the problem was also used by forcing $|G_v| \leq |H_v|$. For $n = 19$ and $|G_v| = 9 = |H_v|$ the symmetry was used again to cut in half the search time by only starting the gluing if $e^1(G_v) \leq e^2(H_v)$. Afterwards, the set of discovered Ramsey graphs was enlarged with their complements.

Results and verification

All of the above allowed to prove very quickly that $R(J_5, J_5) = 22$. For $n = 21$ the enumeration could be completed in a couple of minutes and the case of $n = 20$ took less than two CPU-hours. $\mathcal{R}(J_5, J_5; 19)$ turned out to be a greater challenge for which roughly 1-2 weeks of computations were required. This resulted in millions of graphs for which only 12422 were non-isomorphic (see table B.3). For $n = 21$ and $n = 20$, their characteristics matched completely to the ones described in [33]. $\mathcal{R}(J_5, J_5; 19)$, had not been enumerated before. As an extra verification the following procedure was run:
For every $G \in \mathcal{R}(J_5, J_5; 19)$ delete every vertex v once to obtain a graph of order 18 (resulting in 141 604 graphs having between 69 and 84 edges). For each of those graphs, find all possible ways to add again a vertex without creating illegal subgraphs. After removing isomorphisms this resulted in exactly the same number of Ramsey graphs of order 19.
The search was extended by deleting two vertices from each $G \in \mathcal{R}(J_5, J_5; 19)$, then going back in the opposite direction as before. Now 1 271 686 $(J_5, J_5; 18)$-graphs were produced (with the same edge range), yet no new Ramsey graphs of order 19 appeared.

4.5.2 Final steps

With the knowledge of all necessary subgraphs, the edge-counting techniques were now applied to $R(J_5, J_6)$.

Theorem 4.5. $R(J_5, J_6) = 37$

Proof. The degrees of a $(J_5, J_6; 37)$-graph must all be either 15 or 16, so $n_{15} + n_{16} = 37$. The values for $E^1(J_4, J_6)$ can be read off from table B.4, which was built with the complete set of graphs I received from professor Radziszowski:

$$E^1(J_4, J_6; 15) = 45,\ E^1(J_4, J_6; 16) = 50$$

From table B.3 we have:

$$E^2(J_5, J_5; 20) = 100,\ E^2(J_5, J_5; 21) = 105$$

Every vertex of degree 15 highers the sum of the deficiencies with $E^1(J_4, J_5; 15)+E^2(J_5, J_5; 21)+ 3 \cdot \frac{15 \cdot 20}{2} = 622,5$ and a vertex of degree 16 adds 630. The only negative component in (4.1) is $3 \cdot \binom{37}{3} = 23310$. This means that

$$\sum_{v \in V} \delta_G(v) = -23310 + n_{15} \cdot 622,5 + n_{16} \cdot 630$$

Since choosing $n_{16} = 37$ and $n_{15} = 0$ leads to $\sum_{v \in V} \delta_G(v) = 0$, all other combinations will give negative sums, meaning that such Ramsey graphs cannot exist.

Therefore, the only remaining possibility for $G \in \mathcal{R}(J_5, J_6; 37)$ are 16-regular graphs with $\delta_G(v) = 0$ for each $v \in V$. This means that $\forall v, N_G^1(v)$ contains exactly 50 blue edges. Again every such edge leads to a blue triangle with v and summing over all vertices counts every triangle three times. This would imply that G has exactly $\frac{37 \cdot 50}{3}$ blue triangles, but this is not a natural number, so such a G does not exist.

$37 \leq R(J_5, J_6)$ is established by the graph in appendix C, also known as $NO_6^-(2)$. □

New upper bounds on a Ramsey number often lead to improved bounds on higher Ramsey numbers as well. A straightforward way is to use the inequality from theorem 2.6. It is also possible to use the properties of smaller Ramsey graphs in counting arguments as was done here before, but this technique can still be brought to a much higher level.

The newly-obtained Ramsey number appears in the inequality:

$$R(J_5, J_7) \leq R(J_4, J_7) + R(J_5, J_6) = 28 + 37 = 65$$

This bound was already found in 2017 ([27]), but can now be proven without calculations.
Some fruitless efforts were done to lower it by using information about the unique critical graph for $R(J_4, J_7)$, the Schläfli graph, and the critical (J_5, J_6)-graph found with the block-circulant generator. It later turned out however, that this was mathematically impossible, because $R(J_5, J_7) = 65$ (see theorem 3.6).

Chapter 5

Conclusion and future work

In this thesis, we investigated methods to improve the bounds on Ramsey numbers in an algorithmic fashion and put them into practice.

First, we made enumeration algorithms for circulant-, distance-, and block-circulant Ramsey graphs. Many lower bounds could be improved, especially for cases avoiding J_n's. For some of these numbers, the earlier lower bound was also given by a block-circulant graph, but was found with heuristic searches. This shows that enumeration methods can certainly have added-value over heuristic algorithms, be it for existence or for non-existence results.
There are however also downsides, of course. Although it should definitely be possible to further improve the implementations, we already touched the edges of enumerability for block-circulant graphs here. Especially for large block-sizes, 24 hours were often not enough to prove that no circulant-block graphs exist.

Next, we looked at more general and well-known methods to improve upper bounds. Because of that, we only applied them to two smaller but unsolved cases. The first, $R(K_3, J_4, J_4)$, failed because computations turned out to be heavier than expected. Also, the fact that this is a multi-colour problem made that we could not use the edge-counting techniques. Yet I believe that with better theoretic and algorithmic frameworks it should be doable.

For the second case, $R(J_5, J_6)$, it was possible to improve the upper bound using only theoretical methods. They required some knowledge about properties of (J_5, J_5)-graphs, which we constructed algorithmically. Together with the improved lower bound, this allowed to completely solve the former case.

It was interesting to see how good block-circulant graphs could reach or improve (J_k, J_r)-cases, compared with how bad they were at even reaching the bounds on the classical Ramsey numbers. Either there could be a pattern here, where critical (J_k, J_r)-graphs tend to show more structure, or it could also mean that these bounds could still be improved a little with more random-like graphs.

In the last week of this thesis, a special strongly-regular graph was discovered, $VO_6^-(2)$, which improved the previous lower bound on $R(J_5, J_7)$ by 25. Surprisingly, this worked together very well with the earlier result about $R(J_5, J_6) = 37$: it is now immediately clear that $R(J_5, J_7) = 37 + 28$ reaches the theoretical upper bound, even if one (or maybe two) of its subcases have unique extremal graphs.
$R(J_5, J_7) = 65$ is now the largest exactly solved (non-trivial) Ramsey number for cliques and almost-cliques, the second-largest being $R(K_3, J_{10}) = 37 = R(J_5, J_6)$. We now have the rather strange phenomenon that $R(J_5, J_7)$ seems to be larger than the more diagonal $45 \leq R(J_6, J_6) \leq 70$. It is possible that the lower bound for this case could also be increased by a large amount.

Efforts were done to look for $VO_6^-(2)$ with the block-circulant generator but they found nothing within the dedicated time-frame. If it were truly not block-circular, then it would be interesting to find out if the block-circulant concept can be further generalised so it also describes this graph. Accordingly, writing a heuristic for finding strongly-regular Ramsey graphs could be useful (but might be impossible).
If it were block-circulant, then it might be interesting to search for better heuristics in the generator. Even though it is an exhaustive algorithm, it could still explore the more likely paths first. That way, the generator could perhaps be useful for just slightly larger parameters.

Observe how now these two new Ramsey numbers are solved, but several strictly smaller numbers are not. Most notably, these are $R(J_5, K_5)$ and $R(K_4, J_6)$, both with best-known bounds being $30 \leq R(.,.) \leq 33$. Since these deviate much more from their theoretical upper bounds, they probably are harder to determine exactly. In fact even the unsolved classical number $R(K_4, K_6)$ is strictly smaller than $R(J_5, J_7)$, but it seems unlikely that they will influence each other.

Several of things that were done in this work could still be improved further. For example, the partial isomorphism avoidance in circulant graphs was not actually implemented since I considered it too late. Also, the implemented gluing algorithm is certainly not state-of-the-art. Although it was sufficient for some our purposes, I am convinced that it can be made several orders of magnitude faster.

For $R(J_5, J_6)$, we 'succeeded' in staying away from algorithms, but there certainly are opportunities here. Given the observations with $R(J_5, J_7)$, I believe there is a good chance that the critical graph for $R(J_5, J_6)$ is unique. Also, no other block-circulant graphs were found and removing and adding vertices didn't yield any others.
As a nice side-effect, the unicity of this $NO_6^-(2)$-graph would almost immediately imply the unicity of $VO_6^-(2)$ as an extremal graph for $R(J_5, J_7)$.

The results shown are mainly for almost-cliques and wheels. The first kind I investigated because I believe they are very interesting. The implementation for wheels was rather to demonstrate the versatility of the frameworks discussed here and their use in cases which are less-studied. Of course, other kinds of graphs could easily be added to the generators, provided that they behave at least decent towards circulant graphs. It could be interesting to seek applications in such directions. Because the code is open source, any researcher could implement the necessary methods himself/herself.

Other quite easy non-heuristic ways of improving bounds were not fully exploited in this thesis. An *extension* algorithm was used to check if a (block-)circulant graph could be extended by one vertex, sometimes with success. This technique can also be used on its own, and together with deletion of vertices and recolouring of edges can be a powerful tool. This was not implemented into a stand-alone program here. Only smaller tests were run, resulting in $R(J_4, J_8) > 29$, but it seems likely that more is possible here.

Near the very end of my thesis, the strength of strongly-regular graphs was discovered. The ones that were interesting mostly appeared to come from structures in finite geometry or algebra. It might be possible to find deeper mathematical relations between these fields and Ramsey theory. It namely is very reasonable to assume that the exact value of small Ramsey numbers is heavily influenced by the existence of certain algebraic structures.

The contradiction between the absolute harmony in the unique critical graph for $R(4, 4)$, against the total disorder in the 656 (presumed) critical graphs for $R(5, 5)$, makes all of this even more interesting to me.

Appendix A

Nederlandstalige Samenvatting

Een (enkelvoudige) graaf $G = (V, E)$ is een verzameling **toppen** V samen met een verzameling van paren van toppen $E \subseteq \binom{V}{2}$ die **bogen** worden genoemd. Grafen hebben belangrijke toepassingen, onder andere als modellen voor wegennetten, molecuulstructuren, etc.

Voor een graaf $G = (V, E)$ heet $K \subseteq V$ een k-**kliek** als $|K| = k$ en $(\forall v, w \in K)(v \sim w)$. Een cokliek is een kliek in de complementaire graaf $\overline{G}$. Het **Ramseygetal** $R(k, r)$ is het kleinste getal n waarvoor geldt dat elke graaf op n toppen ofwel een k-kliek bevat, ofwel een r-cokliek. Het bestaan van dergelijke getallen wordt de Stelling van Ramsey genoemd. (Stelling 2.5) Het bepalen van de exacte waarde van deze Ramseygetallen blijkt zelfs voor zeer kleine parameterwaarden een moeilijk probleem, dus over het algemeen zijn enkel onder- en bovengrenzen gekend. Zo geldt bijvoorbeeld dat $R(4, 4) = 18$, maar over $R(5, 5)$ is enkel geweten dat $43 \leq R(5, 5) \leq 46$ [3].

In dit werk worden algoritmes besproken en geïmplementeerd, waarmee deze grenzen kunnen worden verbeterd. Ondergrenzen kunnen bewezen worden aan de hand van één getuigende graaf die noch een k-kliek, noch een r-cokliek bevat. Zo'n graaf wordt een **Ramsey-**(k, r)**-graaf** genoemd, en als deze n toppen heeft bewijst zijn bestaan de ongelijkheid $n < R(k, r)$.
Voor een bovengrens $R(k, r) \leq n$ is het nodig om aan te tonen dat álle grafen op n toppen de gevraagde eigenschap hebben. Dit gebeurt meestal aan de hand van wiskundige argumenten samen met exhaustieve generatie van een zekere klasse grafen.

A.1 Ondergrenzen op Ramseygetallen

Als eerste werden **circulante grafen** bekeken om ondergrenzen mee aan te tonen. Dat zijn grafen waarvan je de toppen zo kan ordenen dat circulaire rotaties automorfismen zijn. (Zie bijvoorbeeld figuur 3.1). Deze grafen komen vaak voor als grootst-gekende (k, r)-grafen.
Er werd een enumeratie-algoritme ontwikkeld dat alle circulante Ramsey-grafen genereert voor willekeurig grote klieken. Dit maakt gebruik van eigenschappen van circulante grafen om snel de Ramsey-eigenschappen na te gaan. Het werd verder aangepast zodat het ook andere grafen dan klieken kan vermijden, met name cykels ($= C_n$), wielen ($= W_n$), klieken met één boog te kort ($= K_n - e = J_n$), etc.

Aangezien deze grafen geen verbeteringen voor ondergrenzen op kliek-vermijdende Ramsey-getallen opleverden, werd eveneens een veralgemening van circulante grafen onderzocht: de **blok-circulante grafen**. Dat zijn grafen wiens adjacentiematrix bestaat uit circulante blokken (van dezelfde grootte):

$$\begin{bmatrix} C_{11} & C_{12} & \dots & C_{1k} \\ C_{12}^T & C_{22} & \dots & C_{2k} \\ \vdots & \vdots & & \vdots \\ C_{1k}^T & C_{2k}^T & \dots & C_{kk} \end{bmatrix}$$

Deze grafen hebben veel meer vrijheidsgraden dan circulante en kunnen dus meer algemene grafen voorstellen. Er werd opnieuw een algoritme ontwikkeld dat al deze Ramseygrafen op een efficiënte wijze enumereert, waarbij zo goed mogelijk geprobeerd werd om ervoor te zorgen dat elke graaf slechts eenmaal wordt gegenereerd.
Met deze blok-circulante grafen bleek het wel mogelijk een significant aantal ondergrenzen te verbeteren, met name op de Ramseygetallen die K_n-e vermijden. Een overzicht daarvan is te vinden in tabel 3.1. Andere verbeteringen staan in tabel 3.2.

A.2 Bovengrenzen op Ramseygetallen

Er werden enkele algemene technieken besproken om bovengrenzen te bewijzen, met name: directe enumeratie, graafsplitsing en graaflijmen.

Bij de eerste techniek worden alle Ramseygrafen inductief gegenereerd, startend bij de ene graaf van orde 1. Door het grote aantal Ramseygrafen is deze techniek alleen bruikbaar voor gevallen met zeer lage parameters.

Het graafsplitsen is enkel mogelijk voor Ramseygetallen op minstens drie kleuren. Om de bovengrens $R(k_1, k_2, k_3) \leq n$ te bewijzen gebruikt deze techniek alle Ramsey-$(k_1, R(k_2, k_3))$-grafen van orde n. Hierbij wordt de tweede kleur opgesplitst in twee nieuwe kleuren, zonder daarbij verboden k_2-klieken of k_3-klieken te maken. Indien dit onmogelijk is, is de bovengrens bewezen.

Het lijmen van Ramsey grafen is een vaakgebruikte techniek die erop steunt dat in een Ramsey-(k, r)-graaf, het nabuurschap $N(v)$ van elke top v een $(k-1, r)$-graaf moet zijn en $V \setminus (N(v) \cup v)$ een $(k, r-1)$-graaf. Als men dus deze Ramseygrafen voor kleinere parameters kent, kan men de omgekeerde weg bewandelen en deze kleine grafen aan elkaar 'lijmen' tot grotere Ramseygrafen. Het efficiënt aan elkaar lijmen van grafen is zowel theoretisch als algoritmisch een uitdaging. Voor beide aspecten werden hier methoden gegeven.

Met combinaties van alle bovenstaande technieken waren we in staat de exacte waarde van twee Ramseygetallen op bijna-klieken te bepalen, naast enkele andere gevallen op wiel-grafen.

Theorem A.1. *$R(J_5, J_6) = 37$, $R(J_5, J_7) = 65$*

Een overzicht van alle gevonden resultaten is te vinden in tabel C.1 en C.2.

Appendix B

Graph counting tables

n	0	1	2	3	4	5	6	7	8	9	10	11
$R(K_2, J_4, J_4)$	1	1	2	4	7	10	12	4	1	1	0	0
$R(K_3, J_3, J_4)$	1	1	3	6	21	52	144	143	90	3	1	0

Table B.1: Multicolor graph counts up to (strong) isomorphism.

$e\backslash n$	1	2	3	4	5	6	7	8	9	10	11	12	total
0	1	1	1	1									4
1		1	1	1									3
2			1	2	2								5
3			1	3	4	1							9
4				2	6	5							13
5					5	11	1						17
6					3	16	8						27
7						12	21	1					34
8						6	39	5					50
9						2	39	18	1				60
10							20	62	1				83
11							6	102	3				111
12							1	92	18				111
13								37	70				107
14								9	173				182
15								1	176	3			180
16								1	81	18			100
17									16	74			90
18									4	153			157
19										116			116
20										37	5		42
21										6	19		25
22											39		39
23											32		32
24											10	2	12
25											2	2	4
26												3	3
27												3	3
28												2	2
29												1	1
30												1	1
total	1	2	4	9	20	53	135	328	543	407	107	14	1623

Table B.2: Counts for $\mathcal{R}(J_4, J_5)$ by number of vertices and edges.

$e \backslash n$	19	20	21
78	1		
79	3		
80	11		
81	42		
82	158		
83	630		
84	1926		
85	3440		
86	3440		
87	1926		
88	630		
89	158		
90	42	3	
91	11	3	
92	3	5	
93	1	11	
94		23	
95		35	
96		23	
97		11	
98		5	
99		3	
100		3	
101			
102			
103			
104			
105			1
total	12422	125	1

Table B.3: Counts for $\mathcal{R}(J_5, J_5; 19-21)$ by number of vertices and edges.

$e \backslash n$	5	6	7	8	9	10	11	12	13	14	15	16	total
6	3	17	34	15	1								70
7		12	49	49	4								114
8		6	55	122	25	1							209
9		2	45	210	101	5							363
10			22	260	355	23	1						661
11			6	223	853	104	3						1189
12			1	136	1399	529	12	1					2078
13				49	1537	2066	49	1					3702
14				12	1163	5567	230	4					6976
15				2	582	9713	1305	14					11616
16				1	187	11072	6876	45					18181
17					38	8261	24508	168					32975
18					9	4020	54803	912					59744
19					1	1238	76567	6341					84147
20					1	252	67697	36852	2				104804
21						41	37915	133255	26				171237
22						7	13360	288749	447				302563
23						2	2940	379164	5498				387604
24						1	420	306638	43510				350569
25						1	47	153238	196804				350090
26							4	47177	513057	40			560278
27							1	8832	786913	605			796351
28								1025	725109	6327			732461
29								78	405097	37163			442338
30								7	137389	128853	7		266256
31									28005	268857	24		296886
32									3420	343724	151		347295
33									250	269634	589		270473
34									16	129676	1645		131337
35										38220	3063		41283
36										6999	4105		11104
37										831	4030		4861
38										71	3156		3227
39										7	1979		1986
40											979	1	980
41											374		374
42											121		121
43											33		33
44											7		7
45											3		3
46													0
47													0
48												1	1
49												1	1
50												1	1
total	22	67	235	1082	6256	42903	286738	1362501	2845543	1231007	20266	4	5796640

Table B.4: Counts for $\mathcal{R}(J_4, J_6)$ by number of vertices and edges.

Appendix C

Graphs and results

Graph6-string of the $(J_6, J_5; 36)$-graph:

```
c?KpxxTUajjUXhYUKrZRMdT}Yjsin{tVye|Xi^hdLyltNtMrueUrxXhX{lJ]]slM
{rKrNrrKovw]W]Fx`feBrK{rN{{orrvw]F`}^eF`fw
```

Graph6-string of the $(J_5, J_7; 64)$-graph:

```
~?@?XKNnxdAYHPLkHGtb_os}HRjV[vOYPl@VoWkOlXsGZ@Rb?yVL`qjhEFLpR?ROf
BuiNWXGmyqoEHdgBoCIZNsoxQn`tsqj`obUY_FOR[xT`_dESiC`?fktIhGNoeGl`OK
tcD[A]FRengAtM?iGGeSKM?|I_AXl^RHOipamESTE]i?QsbsrI_LcWSe[?{CzUg?Ir
Ryieb_LLQC{Q{[BdCMUqmZ?E`KVYrqSqEaPepCb_sgrcHXEpW?zEk@owQf?su`H_Pcr
GFqpbMJlRI?AlppQQ[?xisI\EBQDRKVaB`rHOT]XPDm@YOmeRSDHKW{PK??Xjnaiwz_KKISIEcn}
```

These graphs can also be found in the House of Graphs [4]:

$(J_5, J_6; 36)$ $(J_5, J_7; 64)$ $(W_7, W_7; 18)$

Witnessing graphs for all other claimed bounds are available at the GitHub page:

https://github.com/Steven-VO/circulant-Ramsey

Legend to table C.1

Table C.1 gives a summary of all results found with the circulant and block-circulant generator. It is a combination of Table I and Table III in the dynamic survey [34]. All entries contain at most 4 values:

- On the right: exact value of the Ramsey number, or best-known lower and upper bound previous to this work (according to [3],[27] and [34]).
- Upper left: Lower bound achieved by circulant graphs, nearly always exhaustively checked to be the best-possible (with exceptions where the upper bound was too high). The number is blue if it reaches the current bound and orange otherwise.
- Lower left: <u>Lower</u> bound reached by block-circulant graph. Mostly only pseudo-exhaustive. Green if it reaches the current bound, in bold if it highers it. An asterisk denotes a one-vertex extension.

There are far too many possible outcomes to show which results are exhaustive, which are only partially checked, etc... So if some particular case is of interest to any reader, I will gladly give detailed information on request.

	K_3	$K_4 - e$	K_4	$K_5 - e$	K_5	$K_6 - e$
K_3	6 6					
$K_4 - e$	7 7	7 10 10				
K_4	9 9	10 11 11	18 18			
$K_5 - e$	11 11	11 13 13	18 19 19	19 22 22∗		
K_5	14 14	14 16 16	25 25	27 30 28 33	42 43 46	
$K_6 - e$	14 17 17	14 17 17	30 30 33	30 31 **37** 38	42 43 62	42 45 45 70
K_6	17 18 18∗	17 21 21	34 36 40	34 37 **42*** 53	57 58 85	57 58 110
$K_7 - e$	21 21	21 28 28	37 37 49	37 40 **50*** 65	57 58 102	59 59 124
K_7	22 23 23	22 28 28 29	47 49 58	49 51 **53** 83	80 80 133	80 80 192
$K_8 - e$	25 25	25 29 29 32	47 49 74	49 51 **61** 100	**81** 80? 180	**83** 80? 218
K_8	27 28 27	27 29 **36** 39	52 59 79	57 59? **61** 123		
$K_9 - e$	31 31	31 34 **36** 46	**62** 59 105			
K_9	36 36	36 36? **41** 46				
$K_{10} - e$	36 37 37	36 41 41 67				
K_{10}	39 40 42	39 41 **44***				

Table C.1: Existing and discovered bounds on Ramsey numbers found with circulant and block-circulant graphs. See the section C above for details.

	K_3	K_4-e	K_4	K_5-e	K_5	K_6-e
K_5-e	11	13	19	22		
K_5	14	16	25	30 33	43 46	
K_6-e	17	17	30 33	**37** 31 38	43 62	45 70
K_6	18	21	36 40	**42** 37 53	58 85	58 110
K_7-e	21	28	37 49	**65** 40 65	**65** 58 102	**66** 59 124
K_7	23	28 29	49 58	**65** 51 82 83	80 133	80 192
K_8-e	25	**30** 29 32	49 74	**66** 51 97 100	**81** 80 180	**83** 80 218
K_8	28	**36** 29 39	59 79	**69** 59 122 123	101 194	
K_9-e	31	**36** 34 46	**62** 59 105	59 145 152	**121** 101 275	
K_9	36	**41** 36 46				

	$K_{10}-e$	K_{10}	$K_{11}-e$	$K_{12}-e$	$K_{14}-e$	$K_{16}-e$
K_3	37	40 42	42 45	**49** 47 53	**61** 60 71	74 91
K_4-e	41 63 67	**46** 41 65	**46** 42 84 86	47 105	60 153?	**82** 74 209?

Table C.2: Existing and improved lower bounds with all techniques. Some updated upper bounds are also shown in the lower left.

Bibliography

[1] B. Alspach and T.D. Parsons. Isomorphism of circulant graphs and digraphs. *Discrete Mathematics*, 25(2):97 – 108, 1979.

[2] V. Angeltveit and B.D. McKay. $R(5,5) \leq 48$. *Journal of Graph Theory*, 89(1):5–13, 2018.

[3] V. Angeltveit and B.D. McKay. Personal communication (unpublished work), 2019.

[4] G. Brinkmann, K. Coolsaet, J. Goedgebeur, and H. Melot. House of graphs: a database of interesting graphs. *Discrete Applied Mathematics*, 161(1-2):311–314, 2013. Available at `http://hog.grinvin.org`.

[5] G. Brinkmann, J. Goedgebeur, and J.C. Schlage-Puchta. Ramsey numbers R(K3,G) for graphs of order 10. *Electronic Journal of Combinatorics*, 19(4), 2012.

[6] A.E. Brouwer and H. Van Maldeghem. *Strongly regular graphs*. Springer, (to appear).

[7] M. Codish, M. Frank, A. Itzhakov, and A. Miller. Computing the Ramsey number R(4,3,3) using abstraction and symmetry breaking. *Constraints*, 21:375–393, 2016.

[8] K. Coolsaet, J. Degraer, and E. Spence. The strongly regular (45,12,3,3) graphs. *Electronic Journal of Combinatorics*, Volume 13, 2016.

[9] P. Erdős. Some remarks on the theory of graphs. *Bull. Amer. Math. Soc.*, 53(4):292–294, 1947.

[10] P. Erdős and G. Szekeres. A combinatorial problem in geometry. *Compositio Mathematica*, 2:463–470, 1935.

[11] S. Evdokimov and I. Ponomarenko. Circulant graphs: Recognizing and isomorphism testing in polynomial time. *St. Petersbg. Math. J.*, 15:813–835, 01 2003.

[12] G. Exoo. A lower bound for r(5, 5). *Journal of Graph Theory*, 13(1):97–98, 1989.

[13] G. Exoo. Some new Ramsey colorings. *Electronic Journal of Combinatorics*, 5:5, 1998.

[14] G. Exoo. New lower bounds for table III. `http://ginger.indstate.edu/ge/RAMSEY/oneedge.pdf`, 2000.

[15] G. Exoo. On some small classical Ramsey numbers. *Electronic Journal of Combinatorics*, Volume 20(1), 2013.

[16] G. Exoo and M. Tatarevic. New lower bounds for 28 classical Ramsey numbers. *Electronic Journal of Combinatorics*, Volume 22, 2015.

[17] J. Goedgebeur and S.P. Radziszowski. New computational upper bounds for Ramsey numbers R(3,k). *Electronic Journal of Combinatorics*, Volume 20, 2013.

[18] A.W. Goodman. On sets of acquaintances and strangers at any party. *The American Mathematical Monthly*, 66(9):778–783, 1959.

[19] R. Graham, B. Rothschild, and J. Spencer. *Ramsey Theory.* John Wiley & Sons, 1990.

[20] R. E. Greenwood and A.M. Gleason. Combinatorial relations and chromatic graphs. *Canadian Journal of Mathematics*, 7:1–7, 1955.

[21] H. Harborth and S. Krause. Ramsey numbers for circulant colorings. *Congressus Numerantium*, 161:139–150, 2003.

[22] D. Holt and G. Royle. A census of small transitive groups and vertex-transitive graphs. *Journal of Symbolic Computation*, 2019.

[23] J.G. Kalbfleisch. Construction of special edge-chromatic graphs. *Canadian Mathematical Bulletin*, 8(5):575–584, 1965.

[24] M. Kolodyazhny. Novye nizhnie granitsy chisel ramseya R(3,12) i R(3,13). *Matematicheskoye i Informacionnoe Modelirovanie*, 14, 2015.

[25] E. Kuznetsov. Computational lower limits on small Ramsey numbers. *arXiv e-prints*, page arXiv:1505.07186, 2015.

[26] F.T. Leighton. Circulants and the characterization of vertex-transitive graphs. *Journal of Research of the National Bureau of Standards*, 88, 1983.

[27] B. Lidický and F. Pfender. Semidefinite Programming and Ramsey Numbers. *arXiv e-prints*, page arXiv:1704.03592, April 2017.

[28] B.D. McKay and A. Piperno. Practical graph isomorphism, II. *Journal of Symbolic Computation*, 60(0):94 – 112, 2014.

[29] B.D. McKay and S. Radziszowski. The first classical Ramsey number for hypergraphs is computed. *Proceedings of the Second Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 304–308, 1991.

[30] B.D. McKay and S.P. Radziszowski. R(4, 5) = 25. *Journal of Graph Theory*, 19(3):309–322, 1995.

[31] B.D. McKay and S.P. Radziszowski. Subgraph counting identities and Ramsey numbers. *Journal of Combinatorial Theory, Series B*, 69(2):193 – 209, 1997.

[32] J.N. McNamara and S.P. Radziszowski. The Ramsey numbers $R(K_4$-e, K_6-e) and $R(K_4$-e, K_7-e). *Congressus Numerantium*, 81:89–96, 1991.

[33] S. Radziszowski. On the Ramsey number R(K_5 - e,K_5 - e). *Ars Combinatoria*, 36:225–232, 1993.

[34] S. Radziszowski. Small Ramsey numbers, dynamic survey. *Electronic Journal of Combinatorics*, revision #15:104, 2017.

[35] F.P. Ramsey. On a problem of formal logic. *Proceedings of the London Mathematical Society*, s2-30(1):264–286, 1930.

[36] D.S. Shetler, M.A. Wurtz, and S.P. Radziszowski. On some multicolor Ramsey numbers involving $K_3{+}e$ and K_4-e. *SIAM J. Discrete Math.*, 26:1256–1264, 2012.

[37] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 8.6)*, 2019. `https://www.sagemath.org`.

[38] P. Turán. On an extremal problem in graph theory. *Matematikai és Fizikai Lapok*, 48, 1941.

[39] K. Walker. Dichromatic graphs and Ramsey numbers. *Journal of Combinatorial Theory*, 5(3):238 – 243, 1968.

[40] Q. Wang, G. Wang, and S. Yan. The Ramsey numbers R(K_3, K_q-e). *Beijing Daxue Xuebao Ziran Kexue Ban*, 34, 1998.

[41] R.J. Wilson. *Introduction to Graph Theory.* John Wiley & Sons, Inc., 1986.