2013-04-22

Practical and theoretical applications of the Regularity Lemma

Fei Song
Worcester Polytechnic Institute

Practical and theoretical applications of the Regularity Lemma

by

Fei Song

A Dissertation

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

in Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

in

Computer Science

by

April 2, 2013

APPROVED:

Professor Gábor N. Sárközy
Advisor

Professor Stanley M. Selkow
Committee Member

Professor Joshua D. Guttman
Committee Member

Professor András Gyárfás
External Committee Member

Dedicated to Miaoyuan Lu

This is the beginning of our adventure

Acknowledgments

I would like to sincerely thank to my advisers: Professor Gábor N. Sárközy and Professor Stanley M. Selkow, who have guided me through my Ph.D career. Special thanks to Professor Joshua D. Guttman and Professor András Gyárfás for serving on my committee.

I wish to thank all my teachers for their help during the five years of my studies at WPI, especially Professor Elke Rundensteiner and Professor Neil T. Heffernan.

I also would like to thank all my fellow students for their help and company, special thanks to Di Yang, Shubhendu Trivedi and Yutao Wang.

Finally, I would like to express my deepest gratitude to the most important people in my life: my parents, my pet and my sister. Without their constant support and continuous trust, this thesis can not be achieved.

Abstract

The Regularity Lemma of Szemerédi is a fundamental tool in extremal graph theory with a wide range of applications in theoretical computer science. Partly as a recognition of his work on the Regularity Lemma, Endre Szemerédi has won the Abel Prize in 2012 for his outstanding achievement. In this thesis we present both practical and theoretical applications of the Regularity Lemma. The practical applications are concerning the important problem of data clustering, the theoretical applications are concerning the monochromatic vertex partition problem.

In spite of its numerous applications to establish theoretical results, the Regularity Lemma has a drawback that it requires the graphs under consideration to be astronomically large, thus limiting its practical utility. As stated by Gowers, it has been "well beyond the realms of any practical applications" [28], the existing applications have been theoretical, mathematical.

In the first part of the thesis, we propose to change this and we propose some modifications to the constructive versions of the Regularity Lemma. While this affects the generality of the result, it also makes it more useful for much smaller graphs. We call this result the practical regularity partition-

ing algorithm and the resulting clustering technique Regularity Clustering. This is the first integrated attempt in order to make the Regularity Lemma applicable in practice. We present results on applying regularity clustering on a number of benchmark data-sets and compare the results with k-means clustering and spectral clustering. Finally we demonstrate its application in Educational Data Mining to improve the student performance prediction.

In the second part of the thesis, we study the monochromatic vertex partition problem. To begin we briefly review some related topics and several proof techniques that are central to our results, including the greedy and absorbing procedures. We also review some of the current best results before presenting ours, where the Regularity Lemma has played a critical role.

Before concluding we discuss some future research directions that appear particularly promising based on our work.

Contents

Chapter 1

Introduction

The Regularity Lemma of Szemerédi [75] has been proven to be a very useful tool in graph theory. It was initially developed as an auxiliary lemma to prove a long standing conjecture of Erdős and Turán [18] on arithmetic progressions, which stated that sequences of integers with positive upper density must contain arbitrarily long arithmetic progressions. Now the Regularity Lemma by itself has become an important tool and found numerous applications (see [48]). Based on the Regularity Lemma and the Blow-up Lemma [46], [47] the Regularity Method has been developed that has been quite successful in a number of applications in graph theory (e.g. [32], [33]).

The basic content of the Regularity Lemma could be described by saying that every graph can, in some sense, be partitioned into random graphs. Since random graphs of a given edge density are much easier to treat than all graphs of the same edge-density, the Regularity Lemma helps us to carry over results that are trivial for random graphs to the class of all graphs with a given number of edges.

In spite of its importance, most of the applications using the Regularity Lemma are theoretical in nature. The lack of practical applications is due to the requirement that the graphs under consideration have to be astronomically large. Specifically, the number of vertices need to be a tower of 2's with height proportional to ε^{-5} to ensure the existence of ε-regular partition in the Regularity Lemma, this has been demonstrated by Gowers in [27].

In the first part of this thesis, we present practical results using the Regularity Lemma: a modification to the Regularity Lemma that we call the "Practical Regularity Partitioning Algorithm" and show experimental results. First we demonstrate the constructive versions of the Regularity Lemma, then we discuss some possible modifications, these modifications lead to a general technique called the "Practical Regularity Partitioning Algorithm" by modifying the constructive procedure for getting the regular partition. This Practical Regularity Partitioning Algorithm is a general technique which can be used in various applications. After the description of the algorithm we will show how to use it for clustering (Regularity Clustering) with experimental results [72]. Furthermore we will demonstrate an application in educational data mining, namely, how to improve student performance prediction by using this technology [73].

The second part of this thesis contains applications of the Regularity Lemma in the monochromatic vertex partition problem. It is to ask how many monochromatic vertex disjoint subgraphs are needed to cover *all* the vertices of an r-colored complete graph. This is a problem in extremal graph theory which studies extremal (maximum or minimum) graphs that satisfy certain properties. To study the monochromatic vertex partition problem, we need to review several closely related research branches: Turán type questions, Ramsey theory and the largest monochromatic subgraph problem. They are all important branches of extremal graph theory and have a wide literature. After reviewing them, we note that one plausible way to attack the monochromatic vertex partition problem is to use the greedy procedure. We will then show that the greedy procedure does not give the optimal solution and we will introduce the absorbing technique as an alternative. After reviewing the research background and proof techniques, we list some current best results. At the end of this part, we present our research work on the monochromatic vertex partition problem using the Regularity Lemma. Our work has resulted in two papers [70], [71], we present the theorems and the detailed proofs.

The organization of this thesis follows this outline. We start with introducing the necessary preliminaries.

Chapter 2

Preliminaries

2.1 Graph Theory

For general graph definitions see the book of Diestel [12]. We list the notation and definitions that we need below:

1. A graph is a pair $G = (V, E)$ of sets satisfying $E \subseteq [V]^2$. The elements of V are the vertices of the graph G, the elements of E are its edges, denoted as $V(G)$ and $E(G)$.

2. A graph is finite if both its vertex set and edge set are finite. A graph is simple if it has no loops and no two of its edges join the same pair of vertices. Except otherwise noted, our thesis is concerned with the study of finite simple graphs only.

3. The complement $\overline{G}$ of G is the graph on V with edge set $[V]^2 \setminus E$.

4. Two vertices x, y of G are adjacent if xy is an edge of G. Two edges $e \neq f$ are adjacent if they have an endpoint in common. If all the

vertices of G are pairwise adjacent, then G is complete, denoted as K_n on n vertices.

5. The degree $deg_G(v) = deg(v)$ of a vertex v is the number of edges at v; $\Gamma(v)$ is the set of neighbors of $v \in V$, $|\Gamma(v)| = deg(v)$.

6. A vertex of degree 0 is isolated; the number

$$\delta(G) = min\{deg(v)|v \in V\}$$

is the minimum degree of G; the number

$$\Delta(G) = max\{deg(v)|v \in V\}$$

is the maximum degree.

7. If all the vertices of G have the same degree k, then G is k-regular.

8. A complete graph K_n is a graph with n vertices and an edge between every two vertices.

9. A graph is said to be k-connected if there does not exist a set of $k-1$ vertices whose removal disconnects the graph.

10. Let $G = (V, E)$ and $G' = (V', E')$ be two graphs, if $V' \subseteq V$ and $E' \subseteq E$ then G' is a subgraph of G. G' is a induced subgraph of G if $G' \subseteq G$ and G' contains all the edges $xy \in E$ with $x, y \in V'$; a spanning subgraph is a subgraph that contains all the vertices of the original graph.

11. A path is a non-empty graph $P = (V, E)$ of the form

$$V = \{x_0, x_1, \dots, x_n\},\ E = \{x_0x_1, x_1x_2, \dots, x_{n-1}x_n\},$$

where the x_i's are all distinct. The vertices x_0 and x_k are called its endpoints; the vertices $x_1, \dots, x_{n-1}$ are the inner vertices. A path with n vertices is denoted as P_n. The cycle is a closed path denoted by C_n, if it has n vertices.

12. A Hamiltonian path/cycle is a path/cycle which visits each vertex of the graph. If a graph has a Hamiltonian cycle it is called Hamiltonian.

13. Let $r \geq 2$ be an integer. A graph $G = (V, E)$ is called r-partite if V admits a partition into r classes such that every edge has its ends in different classes, i.e. vertices in the same partition class are never adjacent. An r-partite graph in which every two vertices from different partition classes are adjacent is called complete. $K(n_1, \dots, n_k)$ is the complete k-partite graph G with classes containing $n_1, \dots, n_k$ vertices. Instead of 2-partite, we say bipartite. (A, B, E) denotes a bipartite graph $G = (V, E)$, where $V = A \cup B$, and $E \subset A \times B$.

14. A star is a complete bipartite graph $K_{1,n}$ which is a tree formed by the central vertex and n leaves around it; a double star is the tree obtained from two vertex disjoint stars by connecting their centers.

15. A multi-coloring of a graph G is a coloring where each edge may receive more than one color.

16. An independent set is a set of vertices in a graph G such that no two of which are adjacent. The independent number $\alpha(G)$ of a graph G is the size of the largest independent set of G.

2.2 Algorithm Complexity

We list the notation and definitions for algorithm complexity analysis. It is more in a descriptive manner rather than strict definitions. Formal definitions can be found in the book [11].

1. For a given function $g(n)$, we denote $O(g(n))$ as:

$$O(g(n)) = \{f(n) : 0 \leq f(n) \leq cg(n) \text{ for all } n \geq n_0.\}$$

   for some positive constants c and n_0.

2. An algorithm solves a problem in time $O(T(n))$ if, when it is provided with a problem instance i of length $n = |i|$, the algorithm can produce the solution in $O(T(n))$ time.

3. A problem is polynomial-time solvable, if there exists an algorithm to solve it in time $O(n^k)$ for some constant k.

4. A decision problem is a problem to which the answer is simply “yes” or “no”.

5. The complexity class P is the set of decision problems that are polynomial-time solvable.

6. A verification algorithm is a two-argument algorithm A, such that A verifies an input string x if there exists a certificate y such that $A(x, y) = 1$.

7. The complexity class NP is the class of languages that can be verified by a polynomial-time algorithm.

8. A language L_1 is polynomial-time reducible to a language L_2, written $L_1 \leq_p L_2$, if there exists a polynomial-time computable function f such that
$$x \in L_1 \text{ if and only if } f(x) \in L_2.$$

9. A language $L \subseteq \{0, 1\}^*$ is NP-complete if
   (a) $L \in NP$
   (b) $L' \leq_p L$ for every $L' \in NP$.

10. If a language L satisfies "$L' \leq_p L$ for every $L' \in NP$.", but not necessarily "$L \in NP$", we say that L is NP-hard.

11. Define the complexity class co-NP as the set of languages L such that $\overline{L} \in NP$.

12. Define the complexity class co-NP-complete as the set of languages L such that $\overline{L} \in NP$-complete.

2.3 Parallel Programming

We introduce some complexity analysis notations for parallel programming. They will be revisited in Section 6.1 (Alon *et al.* algorithmic Regularity Lemma). More details can be found in [66].

A random access machine (RAM) (see [4]) is more similar to a high level computer than Turing machines. A RAM has its own local random-access memory, each cell of which can store an arbitrary large integer. The instructions for RAMs are multiplication, division, addition, subtraction, conditional branches based on predicates "=", "<", "and", "or" and "not" and reading and writing into its memory. A parallel random-access machine ([30], [23], [43]) is a collection of RAMs operating synchronously in parallel. The RAM's are communicating with one another through a global memory. All of the processors execute the same program in lock-step fashion, except that each processor knows its unique processor number, and this can be used in the instructions.

PRAMs can be classified according to restrictions on global memory access. Even though there is a variety of PRAM models, they do not differ very widely in their computational power. Therefore we choose the weakest possible model, the EREW, as our model. An Exclusive-Read Exclusive-Write (or EREW) PRAM is a PRAM for which simultaneous access to any memory location by different processors is forbidden for both reading and writing.

For the PRAM model we choose the time and the number of parallel processors to measure the complexity of a computation. The time of the computation is the total cost of instructions executed by the processors.

A PRAM algorithm is said to be efficient if it runs in time polynomial in the log of the input size and uses polynomially many processors. A problem solvable by such a PRAM algorithm is said to be in NC. We refer to the algorithm as an NC algorithm. When the running time is $O((\log n)^i)$, the algorithm is in NC^i.

A major goal in parallel computation is to prove that a given problem belongs to NC. Additional objectives are to minimize the number of processors used and to find the precise time bounds.

2.4 Matrix Theory

Here we introduce definitions that will be used in Section 5.3 in the spectral clustering algorithm.

Definition 2.1 *For any $n \times n$ real matrix A, if there exists a non-zero vector v and a real number λ such that*

$$\lambda v = Av,$$

then we say that v is an eigenvector of A, λ is said to be the eigenvalue corresponding to v.

Definition 2.2 *Let $G = (V, E)$ be an undirected graph with vertex set $V = \{v_i, \ldots, v_n\}$, assume that each edge between two vertices v_i and v_j carries a non-negative weight $w_{ij} \geq 0$. The weighted adjacency matrix of a given* *graph is the matrix*

$$W = (w_{ij})_{i,j=1,\dots,n}.$$

We require $w_{ij} = w_{ji}$, and $w_{ij} = 0$ means that there is no edge between v_i and v_j.

Recall the definition of the degree in Section 2.1, here in weighted graphs we define the weighted degree of a vertex as

$$d_i = \sum_{j=1}^{n} w_{ij}.$$

Now we define the degree matrix:

Definition 2.3 *The degree matrix D is defined as the diagonal matrix with the degrees $d_i, \dots, d_n$ in the diagonal.*

We will use normalized graph Laplacians defined as follows:

Definition 2.4 *L_{sym} is a symmetric matrix (called normalized graph Laplacian), defined by*

$$L_{sym} = D^{-\frac{1}{2}} L D^{-\frac{1}{2}} = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}.$$

Chapter 3

Regularity Lemma

The Regularity Lemma [75] is one of the most powerful tools of extremal graph theory. It was invented as an auxiliary lemma in the proof of a major result on the Ramsey properties of arithmetic progressions, its importance has been realized and has been used more and more in recent years.

Basically this lemma claims that all (dense) graphs can be approximated by random graphs in the following sense: every graph can be partitioned into a bounded number of equal parts, so that most of its edges run among different parts and the edges between any two parts are distributed fairly uniformly, just as they had been generated randomly. Since random graphs of a given edge density are much easier to treat than all graphs of the same edge-density, the Regularity Lemma helps us to translate results that are trivial for random graphs to the class of all graphs with a given number of edges.

To present the Regularity Lemma precisely, we need some definitions first:

Let $G = (V, E)$ denote a graph, where V is the set of vertices and E is the set of edges. When A, B are disjoint subsets of V, the number of edges with one endpoint in A and the other in B is denoted by $e(A, B)$. When A and B are nonempty, recall that the *density* of edges between A and B is

$$d(A, B) = \frac{e(A, B)}{|A||B|}.$$

The most important concept is the following.

Definition 3.1 *The bipartite graph $G = (A, B, E)$ is* ε-regular *if for every $X \subset A$, $Y \subset B$ satisfying: $|X| > \varepsilon|A|$, $|Y| > \varepsilon|B|$, we have $|d(X, Y) - d(A, B)| < \varepsilon$, otherwise it is* ε-irregular.

Roughly speaking this means that in an ε-regular bipartite graph the edge density between *any* two relatively large subsets is about the same as the original edge density. In effect this implies that all the edges are distributed almost uniformly.

The most important property of regular pairs is the following: let (A, B) be an ε-regular pair with density d. Then for any $Y \subset B, |Y| > \varepsilon|B|$ we have

$$\#\{x \in A : deg(x, Y) \leq (d - \varepsilon)|Y|\} \leq \varepsilon|A|.$$

Definition 3.2 *A partition P of the vertex set $V = V_0 \cup V_1 \cup \ldots \cup V_k$ of a graph $G = (V, E)$ is called an* equitable *partition if all the classes $V_i, 1 \leq i \leq k$, have the same cardinality. V_0 is called the exceptional class.*

Note that the exceptional class V_0 is there only for a technical reason, namely to guarantee that the other classes have the same cardinality.

Definition 3.3 *For an equitable partition P of the vertex set $V = V_0 \cup V_1 \cup \ldots \cup V_k$ of $G = (V, E)$, we associate a measure called the* index *of P (or the* potential*) which is defined by*

$$ind(P) = \frac{1}{k^2} \sum_{s=1}^{k} \sum_{t=s+1}^{k} d(V_s, V_t)^2.$$

This will measure the progress towards an ε-regular partition.

Definition 3.4 *An equitable partition P of the vertex set $V = V_0 \cup V_1 \cup \ldots \cup V_k$ of $G = (V, E)$ is called ε-*regular *if $|V_0| < \varepsilon|V|$ and all but εk^2 of the pairs (V_i, V_j) are ε-regular where $1 \leq i < j \leq k$.*

With these definitions we are now in a position to state the Regularity Lemma.

Theorem 3.5 (Szemerédi, 1976 [75]) *For every positive $\varepsilon > 0$ and positive integer t there is an integer $T = T(\varepsilon, t)$ such that every graph with $n > T$ vertices has an ε-regular partition into $k + 1$ classes, where $t \leq k \leq T$.*

Below is an r-color version of the Regularity Lemma:

Theorem 3.6 (Szemerédi, 1976 [75]) *For every positive ε and positive integer m there are positive integers M and n_0 such that for $n \geq n_0$ the following holds. For all graphs G_1, $G_2, \ldots, G_r$ with $V(G_1) = V(G_2) = \ldots = V(G_r) = V$, $r \geq 2$, $|V| = n$, there is a partition of V into $l + 1$ classes (clusters)*

$$V = V_0 + V_1 + V_2 + ... + V_l$$

such that

- $m \leq l \leq M$
- $|V_1| = |V_2| = ... = |V_l|$
- $|V_0| < \varepsilon n$
- *apart from at most* $\varepsilon\binom{l}{2}$ *exceptional pairs, the pairs* $\{V_i, V_j\}$ *are* (ε, G_s)*-regular for* $s = 1, 2, \ldots, r$.

There are a large number of applications using the Regularity Lemma. An important concept in these applications is the reduced graph.

Definition 3.7 *Given an arbitrary graph* $G = (V, E)$*, a partition of* V *into* k *clusters as in Theorem 3.6, and two parameters* ε*,* d*, we define the* **reduced** *graph* G^R *as the graph whose vertices are associated to the clusters and whose edges are associated to* ε*-regular pairs with density more than* d*. If we have a coloring on the edges of* G*, then the edges of the reduced graph will be colored with a color that appears on most of the edges between the two clusters.*

The most important property of the reduced graph is that many properties of G are inherited by G^R.

Part I

Practical applications of the Regularity Lemma

Chapter 4

Motivation

The Regularity Lemma has become an important tool and has numerous theoretical applications, not only in graph theory but also in theoretical computer science and number theory (see [48]). The original Regularity Lemma only claims the existence of a partition with certain properties. To apply the Regularity Lemma in practical settings, first we need a constructive version which describes a method to construct the partition. Alon *et al.* [3] were the first to give an algorithmic version. Since then a few other algorithmic versions have also been proposed [21], [44].

Although these algorithms are efficient and run in polynomial time (see section 6.1), but they are still not truly applicable. This is due to the fact that the graph under consideration has to be astronomically large. The number of vertices of the input graph must be of a tower of 2's with height proportional to ε^{-5}. Furthermore, Gowers demonstrated [24] that this tower bound is necessary.

To make the Regularity Lemma applicable to much smaller graphs, say with several thousand vertices, we have to make certain modifications. This is going to be the main theme of this part of the thesis. We start with introducing data clustering in next chapter.

Chapter 5

Clustering

5.1 Clustering

Clustering is one of the most important branches in data processing. Intuitively it means to divide the data points into meaningful groups, and then these groups can be used for feature extraction and summarizing, or for making data-driven inferences. A useful view of clustering is the following: Given a space X, clustering could be thought of as a partitioning of this space into k parts, i.e. $f : X \longmapsto \{1, \ldots, k\}$. Usually this partitioning is obtained by optimizing some internal criteria such as the inter-cluster distances, etc. However, which criteria will lead to an optimal clustering is still unclear.

There are variety of clustering algorithms. In this thesis we will use k-means clustering and spectral clustering, a brief description of both algorithms are given below.

5.2 k-means clustering algorithm

Define a set of n data points

$$X = \{x_1, \ldots, x_n\},$$

and a set of k centers

$$C = \{c_1, \ldots, c_n\},$$

as the clustering solution. k-means finds the clusters by minimizing the function :

$$\sum_{i=1}^{n} \sum_{j=1}^{k} \|x_i - c_j\|^2.$$

k-means Algorithm [41] :

1. **Initialize:** *Select the initial cluster centers.*

2. **Assign Center:** *For every data point find the nearest center.*

3. **Recompute the center:** *Recompute the center using the data points inside same cluster.*

4. **Iteration:** *If certain criteria meet then output the clustering result, otherwise iteration with the new centers.*

In spite of the great popularity of the k-means algorithm, very few theoretical guarantees on its performance are known.

5.3 Spectral clustering algorithm

Out of the various modern clustering techniques, spectral clustering has become one of the most popular. This has happened due to not only its superior performance over traditional clustering techniques, but also due to the strong theoretical underpinnings in spectral graph theory and its ease of implementation.

Spectral clustering is to approximately solve the balanced mincut problem. Attach a weight value to each edge, then the mincut problem can be formalized as following: find a partition $A_1, A_2, \ldots, A_k$ that minimizes the value

$$cut(A_1, A_2, \ldots, A_k) = \frac{1}{2}\sum_{i=1}^{k} W(A_i, \bar{A}_i).$$

In practice we want a balanced cut. For example when $k = 2$, the optimal solution often gives the answer such that a single vertex stands as a part [74], which is not a desired result. When dealing with balanced cuts, it is important to define the meaning of balance. There are several different definitions [40], [67]. Intuitively, a balanced mincut means a mincut with more or less the same size for each part.

Finding a balanced mincut is an NP-hard problem. Even further, there is no polynomial algorithm that can even approximate the optimal solution up to a constant factor, this approximation problem is NP-hard itself [6]. The advantage of spectral clustering is that it takes an approximation which can be translated into a standard linear algebra problem and has a standard yet simple solution.

Despite various advantages of spectral clustering, one major problem is that for large datasets it is very computationally intensive. Another interesting issue is that a balanced mincut might not be the best criteria to evaluate a partition. It is to minimize the inter-cluster distance, not even considering the uniform behavior inside the clusters.

For the sake of completeness we present the spectral clustering algorithm here, the detailed analysis and algorithms can be found in [55].

We first introduce the similarity graph.

A similarity graph is to model the local neighborhood relationships between the data points. Two popular constructions are the following:

1. *k-nearest neighbor graphs:* Here the goal is to connect vertex v_i with vertex v_j if v_j among the k-nearest neighbors of v_i.

2. *The fully connected graph:* Here we simply connect all points with positive similarity with each other, and we weight all edges by s_{ij}.

Both graphs mentioned above are regularly used in spectral clustering. For now we do not have any knowledge on how the choice of the similarity graph influences the spectral clustering result. We will use both methods for our sake of comparison.

Basic notation and definitions, such as the weighted adjacency matrix, eigenvectors and the normalized Laplacian can be found in Section 2.4,

Normalized spectral clustering according to Ng, Jordan and Weiss [57] :

1. Construct a similarity graph by one of the ways described above. Let W be its weighted adjacency matrix.

2. Compute the normalized Laplacian L_{sym}.

3. Compute the first k eigenvectors $\{u_1, \ldots, u_k\}$ of L_{sym}.

4. Let $U \in \mathbb{R}^{n \times k}$ be the matrix containing the vectors $\{u_1, \ldots, u_k\}$ as columns.

5. Form the matrix $T \in \mathbb{R}^{n \times k}$ from U by normalizing the rows to norm 1.

6. For $i = 1, \ldots, n$, let $y_i \in \mathbb{R}^k$ be the vector corresponding to the i-th row of T.

7. Cluster the points $(y_i)_{i=1,\ldots,n}$ with the k-means algorithm into clusters $\{C_1, \ldots, C_k\}$.

5.4 Our Methodology: Regularity Clustering

The Regularity Lemma, as we stated in Chapter 3 (Theorem 3.6), is to claim the existence of a regular partition, from which we can construct the reduced graph, hence decreasing the order of the input graph significantly. Also the criteria for a regular partition is quite different from spectral clustering, it takes into account the uniform distribution of the edge weights.

In the next chapter we will propose a general methodology to make the Regularity Lemma more useful in practice. To make it truly applicable, instead of constructing a provably regular partition we construct an *approximately* regular partition. This partition will be less accurate, yet it behaves just like a regular partition (especially for graphs appearing in practice) and it does not require the large number of vertices as mandated by the original Regularity Lemma. We use this approximately regular partition for performing clustering, and we call the resulting new clustering technique *Regularity Clustering*.

We will also present applications of Regularity clustering: first we present the accuracy comparisons with standard clustering methods such as k-means and spectral clustering on UCI datasets [72]; then we present an application within the Educational Data Mining realm to improve student performance prediction [73].

Chapter 6

A Practical regularity partitioning algorithm

The original Regularity Lemma is an existential, non-constructive result. It does not give a method to construct a regular partition but only shows that one must exist. To make it truly applicable we first need an algorithmic version. Alon *et al.* [3] were the first to give an algorithmic version. Below we present the details of the Alon *et al.* algorithm.

6.1 Alon *et al.* version

For the definition of NC^1 see Section 2.3.

Theorem 6.1 (Algorithmic Regularity Lemma, Alon *et al.*, 1994 [3]) *For every $\varepsilon > 0$ and every positive integer t there is an integer $T = T(\varepsilon, t)$ such that every graph with $n > T$ vertices has an ε-regular partition into $k + 1$ classes, where $t \leq k \leq T$. For every fixed $\varepsilon > 0$ and $t \geq 1$ such a* *partition can be found in $O(M(n))$ sequential time, where $M(n)$ is the time for multiplying two n by n matrices with $0, 1$ entries over the integers. The algorithm can be parallelized and implemented in NC^1.*

This result is somewhat surprising from a computational complexity point of view since it was proved in [3] that the corresponding decision problem (checking whether a given partition is ε-regular) is co-NP-complete (see Section 2.2). Thus the search problem is easier than the decision problem. To describe this algorithm, we need a couple of lemmas.

Lemma 6.2 (Alon *et al.*, 1994 [3]) *Let H be a bipartite graph with equally sized classes $|A| = |B| = n$. Let $2n^{-1/4} < \varepsilon < \frac{1}{16}$. There is an $O(M(n))$ algorithm that verifies that H is ε-regular or finds two subset $A' \subset A$, $B' \subset B$, $|A'| \geq \frac{\varepsilon^4}{16}n$, $|B'| \geq \frac{\varepsilon^4}{16}n$, such that $|d(A, B) - d(A', B')| \geq \varepsilon^4$. The algorithm can be parallelized and implemented in NC^1.*

This lemma basically says that we can either verify that the pair is ε-regular or we provide certificates that it is not. The certificates are the subsets A', B' and they help to proceed to the next step in the algorithm. The next lemma describes the procedure to do the refinement from these certificates.

Lemma 6.3 (Szemerédi, 1976 [75]) *Let $G = (V, E)$ be a graph with n vertices. Let P be an equitable partition of the vertex set $V = V_0 \cup V_1 \cup \ldots \cup V_k$. Let $\gamma > 0$ and let k be a positive integer such that $4^k > 600\gamma^{-5}$. If more than γk^2 pairs (V_s, V_t), $1 \leq s < t \leq k$, are γ-irregular then there is an equitable partition Q of V into $1 + k4^k$ classes, with the cardinality of the exceptional* *class being at most*

$$|V_0| + \frac{n}{4^k}$$

and such that

$$ind(Q) > ind(P) + \frac{\gamma^5}{20}.$$

See Definition 3.3 for *ind* function.

This lemma implies that whenever we have a partition that is not γ-regular, we can refine it into a new partition which has a better index (or potential) than the previous partition. The refinement procedure to do this is described below.

Refinement Algorithm: *Given a γ-irregular equitable partition P of the vertex set $V = V_0 \cup V_1 \cup \ldots \cup V_k$ with $\gamma = \frac{\varepsilon^4}{16}$, construct a new partition Q.*
For each pair (V_s, V_t), $1 \leq s, t \leq k, s \neq t$, we apply Lemma 6.2 with $A = V_s$, $B = V_t$ and ε. If (V_s, V_t) is found to be ε-regular we do nothing. Otherwise, the certificates partition V_s and V_t into two parts (namely the certificate and the complement). For a fixed s we do this for all $t \neq s$. In V_s, these sets define the obvious equivalence relation with at most 2^{k-1} classes, namely two elements are equivalent if they lie in the same partition set for every $t \neq s$. The equivalence classes will be called atoms. Set $m = \lfloor \frac{|V_i|}{4^k} \rfloor$, $1 \leq i \leq k$. Then we choose a collection Q of pairwise disjoint subsets of V such that every member of Q has cardinality m and every atom A contains exactly $\lfloor \frac{|A|}{m} \rfloor$ members of Q. The collection Q is an equitable partition of V into at most $1 + k4^k$ classes and the cardinality of its exceptional class is at most $|V_0| + \frac{n}{4^k}$.

Now we are ready to present the main algorithm.

Regular Partition Algorithm (Alon *et al.*): *Given a graph G and ε, construct a ε-regular partition.*

1. **Initial partition:** *Arbitrarily divide the vertices of G into an equitable partition P_1 with classes $V_0, V_1, \ldots, V_b$, where $|V_1| = \lfloor \frac{n}{b} \rfloor$ and hence $|V_0| < b$. Denote $k_1 = b$.*

2. **Check regularity:** *For every pair (V_s, V_t) of P_i, verify if it is ε-regular or find $X \subset V_s, Y \subset V_t, |X| \geq \frac{\varepsilon^4}{16}|V_s|, |Y| \geq \frac{\varepsilon^4}{16}|V_t|$, such that $|d(X,Y) - d(V_s, V_t)| \geq \varepsilon^4$.*

3. **Count regular pairs:** *If there are at most εk_i^2 pairs that are not verified as ε-regular, then halt. P_i is an ε-regular partition.*

4. **Refinement:** *Otherwise apply the Refinement Algorithm and Lemma 6.3, where $P = P_i, k = k_i, \gamma = \frac{\varepsilon^4}{16}$, and obtain a partition Q with $1 + k_i 4_i^k$ classes.*

5. **Iteration:** *Let $k_{i+1} = k_i 4_i^k, P_{i+1} = Q, i = i + 1$, and go to step 2.*

Since the index cannot exceed $1/2$, the algorithm must halt after at most $\lceil 10\gamma^{-5} \rceil$ iterations (see [3]). Unfortunately, in each iteration the number of classes increases exponentially to $k4^k$ from k. This implies that the graph G must be indeed astronomically large (a tower function) to ensure the completion of this procedure. As mentioned before, Gowers [27] proved that indeed this tower function is necessary in order to guarantee an ε-regular partition for *all* graphs. The size requirement of the algorithm above makes it impractical for real world situations where the number of vertices typically is a few thousand. We will show our modifications to make it applicable to small graphs in Section 6.3.

6.2 Frieze-Kannan version

The Frieze-Kannan constructive version is quite similar to the Alon *et al.* version, the only difference is how to check regularity of the pairs in Step 2. Instead of Lemma 6.2, another lemma is used based on the computation of singular values of matrices. For the sake of completeness we present the details below. More details can be found at [21].

First we need some definitions:

An $m \times n$ matrix A has a singular value decomposition into the sum of rank one matrices, The first singular value σ_1 is defined as

$$\sigma_1(A) = max_{|x|=|y|=1}|x^T Ay|.$$

This value can be computed with high accuracy in polynomial time. It is the square root of the largest eigenvalue of $A^T A$.

For the following lemma, W is a $p \times q$ matrix with rows indexed by R, columns indexed by C. We define

$$\|W\|_\infty = max_{i \in R, j \in C}|W(i,j)|.$$

Assume $\|W\|_\infty \leq 1$. For $S \subset R, U \subset C$ we define

$$W(S,T) = \sum_{i \in S} \sum_{j \in T} W(i,j) = x_S^T W x_U.$$

where x_S is the 0-1 indicator vector of S i.e. $(x_S)_i = 1$ iff $i \in S$.

Now we state the lemma.

Lemma 6.4 (Frieze, Kannan, 1999 [21]) *Let W be an $R \times C$ matrix with $|R| = p$, $|C| = q$ and $\|W\|_\infty \leq 1$ and let γ be a positive real.*

a If there exists $S \subseteq R, T \subseteq C$ such that $|S| \geq \gamma p, |T| \geq \gamma q$ and $|W(S,T)| \geq \gamma |S||T|$ then $\sigma_1(W) \geq \gamma^3 \sqrt{pq}$. Where σ_1 is the first singular value.

b If $\sigma_1(W) \geq \gamma\sqrt{pq}$ then there exist $S \subseteq R, T \subseteq C$ such that $|S| \geq \gamma' p, |T| \geq \gamma' q$ and $W(S,T) \geq \gamma' |S||T|$, where $\gamma' = \frac{\gamma^3}{108}$. Furthermore, S, T can be constructed in polynomial time.

Combining Lemmas 6.3 and 6.4, we get an algorithm for finding an ε-regular partition, quite similar to the Alon *et al.* version [3], which we present below:

Regular Partition Algorithm (Frieze-Kannan): *Given a graph G and ε, construct a ε-regular partition.*

1. **Initial partition:** *Arbitrarily divide the vertices of G into an equitable partition P_1 with classes $V_0, V_1, \ldots, V_b$, where $|V_1| = \lfloor \frac{n}{b} \rfloor$ and hence $|V_0| < b$. Denote $k_1 = b$.*

2. **Check regularity:** *For every pair (V_s, V_t) of P_i, compute $\sigma_1(W_{s,t})$. If a pair (V_s, V_t) is not ε-regular then by Lemma 6.4 we obtain a proof that it is not $\gamma = \varepsilon^9/108$-regular.*

3. **Count regular pairs:** *If there are at most εk_i^2 pairs that produce proofs of non γ-regularity, then halt. P_i is an ε-regular partition.*

4. **Refinement:** *Otherwise apply the Refinement Algorithm and Lemma 6.3, where* $P = P_i, k = k_i, \gamma = \frac{\varepsilon^9}{108}$*, and obtain a partition* P' *with* $1 + k_i 4_i^k$ *classes.*

5. **Iteration:** *Let* $k_{i+1} = k_i 4_i^k, P_{i+1} = P', i = i + 1$*, and go to step 2.*

This algorithm is guaranteed to finish in at most ε^{-45} steps with an ε-regular partition (see [21]).

6.3 The practical regularity partitioning algorithm

We see that even the constructive versions are not directly applicable to real world scenarios. We note that the above algorithms have such restrictions because their aim is to be applicable to *all* graphs. Thus, to make the Regularity Lemma truly applicable we would have to give up our goal that the lemma should work for *every* graph and should be content with the fact that it works for *most* graphs. To ensure that this happens, we modify the Regular Partition Algorithm(s) (6.1, 6.2) so that instead of constructing a regular partition, we find an *approximately* regular partition, which should be much easier to construct. We have the following 3 major modifications to the Regular Partition Algorithm (Alon *et al.* Version).

Modification 1: We want to decrease the cardinality of atoms in each iteration. In the Refinement Algorithm (6.1) the cardinality of the atoms in a V_s may be 2^{k-1}, where k is the number of classes in the current partition. This is because the algorithm tries to find all the possible ε-irregular pairs such that this information can then be embedded into the subsequent refinement procedure. Hence potentially each class may be involved with up to $(k-1)$ ε-irregular pairs. One way to avoid this problem is to bound this number. To do so, instead of using all the ε-irregular pairs, we only use some of them. Specifically, in this thesis, for each class we consider at most one ε-irregular pair that involves the given class. By doing this we reduce the number of atoms to at most 2. We observe that in spite of the crude approximation, this seems to work well in practice.

Modification 2: We want to bound the rate by which the class size decreases in each iteration. As we have at most 2 atoms for each class, we could significantly increase m used in the Refinement Algorithm as $m = \frac{|V_i|}{l}$, where a typical value of l could be 3 or 4, much smaller than 4^k. We call this user defined parameter l the refinement number.

Modification 3: Modification 2 might cause the size of the exceptional class to increase too fast. Indeed, by using a smaller l, we risk putting $\frac{1}{l}$ portion of all vertices into V_0 after each iteration. To overcome this drawback, we "recycle" most of V_0, i.e. we move back most of the vertices from V_0. Here is the modified Refinement Algorithm.

Modified Refinement Algorithm: *Given a γ-irregular equitable partition P of the vertex set $V = V_0 \cup V_1 \cup \ldots \cup V_k$ with $\gamma = \frac{\varepsilon^4}{16}$ and refinement number l, construct a new partition Q.*
For each pair (V_s, V_t), $1 \leq s < t \leq k$, we apply Lemma 6.2 with $A = V_s$, $B = V_t$ and ε. For a fixed s if (V_s, V_t) is found to be ε-regular for all $t \neq s$ we do nothing, i.e. V_s is one atom. Otherwise, we select one ε-irregular pair (V_s, V_t) randomly and the corresponding certificate partitions V_s into two atoms. Set $m = \lfloor \frac{|V_i|}{l} \rfloor$, $1 \leq i \leq k$. Then we choose a collection Q' of pairwise disjoint subsets of V such that every member of Q' has cardinality

m and every atom A contains exactly $\lfloor \frac{|A|}{m} \rfloor$ members of Q'. Then we unite the leftover vertices in each V_s, we select one more subset of size m from these vertices and add these sets to Q' resulting in the partition Q. The collection Q is an equitable partition of V into at most $1 + lk$ classes.

Now, we are ready to present our Practical Regular Partitioning Algorithm. There are three main parameters to be selected by the user: ε, refinement number l and h, the minimum class size when we must halt the refinement procedure. h is used to ensure that if the class size has gone too small then the procedure should not continue.

Practical Regular Partitioning Algorithm: *Given a graph G and parameters ε, l, h, construct an approx. ε-regular partition.*

1. **Initial partition:** *Arbitrarily divide the vertices of G into an equitable partition P_1 with classes $V_0, V_1, \ldots, V_l$, where $|V_1| = \lfloor \frac{n}{l} \rfloor$ and hence $|V_0| < l$. Denote $k_1 = l$.*

2. **Check size and regularity:** *If $|V_i| < h$, $1 \leq i \leq k$, then halt. Otherwise for every pair (V_s, V_t) of P_i, verify if it is ε-regular or find $X \subset V_s, Y \subset V_t, |X| \geq \frac{\varepsilon^4}{16}|V_s|, |Y| \geq \frac{\varepsilon^4}{16}|V_t|$, such that $|d(X,Y) - d(V_s, V_t)| \geq \varepsilon^4$.*

3. **Count regular pairs:** *If there are at most εk_i^2 pairs that are not verified as ε-regular, then halt. P_i is an ε-regular partition.*

4. **Refinement:** *Otherwise apply the Modified Refinement Algorithm, where $P = P_i, k = k_i, \gamma = \frac{\varepsilon^4}{16}$, and obtain a partition Q with $1 + lk_i$ classes.*

5. **Iteration:** *Let $k_{i+1} = lk_i, P_{i+1} = Q, i = i + 1$, and go to step 2.*

The Frieze-Kannan version is modified in a similar way.

6.4 Regularity clustering

To make the Regularity Lemma applicable in clustering settings, we adopt the following two phase strategy (as in [68] and illustrated in Figure 6.1):

1. **Application of the Practical Regularity Partitioning Algorithm:** In the first stage we apply the Practical Regularity Partitioning Algorithm as described in the previous section to obtain an approximately regular partition of the graph representing the data. Once such a partition has been obtained, the reduced graph as described in Definition 3.7 could be constructed from the partition.

2. **Clustering the Reduced Graph:** The reduced graph as constructed above would preserve most of the properties of the original graph (see [48]). This implies that any changes made in the reduced graph would also reflect in the original graph. Thus, clustering the reduced graph would also yield a clustering of the original graph. We apply spectral clustering (Section 5.3, though any other pairwise clustering technique could be used) on the reduced graph to get a partitioning and then project it back to the higher dimension. Recall that vertices in the exceptional set V_0 are leftovers from the refinement process and must be assigned to the clusters obtained. Thus in the end these leftover vertices are redistributed amongst the clusters using k-nearest neighbor

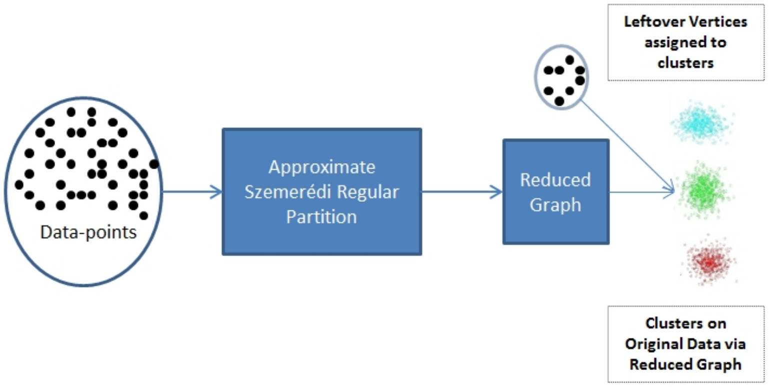
Data-points
Approximate Szemerédi Regular Partition
Reduced Graph
Leftover Vertices assigned to clusters
Clusters on Original Data via Reduced Graph

Figure 6.1: A Two Phase Strategy for Clustering

classifier to get the final grouping.

We call this method Regularity Clustering. We present our experimental results in the next Section.

6.5 Experimental results on UCI data sets

In this section we present extensive experimental results to indicate the efficacy of regularity clustering by employing it for clustering on a number of benchmark datasets. We compare the results with spectral clustering and k-means clustering in terms of accuracy. We also report results that indicate the amount of compression obtained by constructing the reduced graph. Results including some numbers on the increase in the index with each step of the algorithm (as defined earlier) and on the number of iterations to obtain a regular partition are also reported.

We first review the datasets considered and the metrics used for comparisons.

6.5.1 Datasets and metrics used

The datasets considered for empirical validation were taken from the University of California, Irvine machine learning repository [79]. A total of 12 datasets were used for validation. We considered datasets with real valued features and associated labels or ground truth. In some datasets that had a large number of real valued features, we removed categorical features to make it easier to cluster. Unless otherwise mentioned, the number of clusters was chosen so as to equal the number of classes in the dataset (i.e. if the number of classes in the ground truth is 4, then the clustering results are for k = 4 etc). An attempt was made to pick a wide variety of datasets i.e. with integer features, binary features, synthetic datasets and of course real world datasets with both very high and small dimensionality.

The following datasets were considered: (1) Red Wine (R-Wine) and (2) White Wine (W-Wine) are two datasets having 1599 and 4898 datapoints respectively, each having 11 features. The target measures wine quality on a scale of 0-10. Though both are ten class problems, they only contain labels for 6 and 7 classes respectively. (3) The Arcene dataset (Arcene) has data for the task of distinguishing cancer from normal patterns from mass spectroscopic data. Thus it is a 2-class problem and was used in the NIPS 2003 feature selection challenge. The data consists of a train set with 100 points, a validation set with 100 points and a test set with 700 points (the test set does not come with labels). However, since we are not making any prediction as such we can combine the train and validation sets here and use it as one dataset. Thus, this dataset has 200 datapoints with each data instance described by 10000 features. Given this very high dimensionality, this should be an interesting dataset to experiment on. (4) The Blood Transfusion Dataset (Blood-T) has 748 data-instances with 4 features each. The task is to predict whether a person donated blood in a certain month (March, 2007) and hence is a two-class problem. (5) The Ionosphere dataset (Ionos) has 351 data instances each of which is 34 dimensional feature vector having information about radar returns from the Ionosphere. The task is to classify the radar returns as "good" i.e. those showing some structure in the ionosphere and "bad" i.e. those returns that do not.

(6) The Wisconsin Breast cancer dataset (Cancer) has 699 datapoints and 9 attributes. The task is classifying a point as benign or malignant. Some rows having missing values were deleted so the actual number of datapoints considered is 683. All the features in this dataset are integer valued. (7) The Pima Indian diabetes dataset (Pima) is a standard dataset provided by the National Institute of Diabetes and Digestive and Kidney diseases. It has 8 attributes for 768 patients (all female of the Pima Indian heritage). The 2-class task for this dataset is to predict whether or not a patient has diabetes. (8) The Vertebral Column dataset (Vertebral-1) has data for 310 orthopaedic patients with 6 bio-mechanical features. The task is to classify patients into either normal, disk hernia or spondilolysthesis and alternately as normal and abnormal. The second task (9) (Vertebral-2) is considered as another dataset. (10) The Steel Plates Faults Dataset (Steel) is a 7-class dataset having 1941 instances and 27 attributes. The goal is to recognize faults of seven different types. (11) The Musk 2 (Musk) dataset has information about a set of 102 molecules of which 39 are judged by human

experts to be musks and the rest judged to be non-musks. Thus, this is a two class problem in which the goal is to predict whether a new molecule will be a musk or not. However, considering all the possible conformations, this dataset has 6598 examples,each with 168 features. Two of which are deleted. (12) Haberman's Survival (Haberman) has data from a study conducted on the survival of patients who had undergone surgery for breast cancer. It only has three features and 306 points, the task is to predict if the patient (described by three features each) survived for more than five years or not after surgery, thus being a two class problem.

Next we discuss the metric used for comparison with other clustering algorithms. For evaluating the quality of clustering, we follow the approach of [81] and use the cluster accuracy as a measure. This is an interesting combinatorial measure that relies on the confusion matrix. The measure is defined as:

$$Accuracy = 100 * \left(\frac{\sum_{i=1}^{n} \delta(y_i, map(c_i))}{n} \right)$$

Where, n is the number of data-points considered, y_i represents the true label (ground truth) while c_i is obtained cluster label of data-point x_i. The function $\delta(y, c)$ equals one if the true and the obtained labels match $(y = c)$ and 0 if they don't. The function *map* is basically a permutation function that maps each cluster label to the true label. An optimal match can be found by using the Hungarian Method for the assignment problem [50].

In the next section we report some experiments and results on one of the above datasets as a case study.

Table 6.1: Clustering Results on Red Wine Dataset by Other Methods

Clustering Method	k = 6	k = 3
Self Tuned Spectral (k-nearest neighbor graph)	26.0163	40.6504
Self Tuned Spectral (fully connected graph)	25.8286	37.3984
k-means	23.8899	37.0857

6.5.2 Case study

Before reporting comparative results on benchmark datasets, we first consider one dataset as a case study. While experiments reported in this case study were carried on all the benchmark datasets considered, the purpose here is to illustrate the investigations conducted at each stage of application of the Regularity Lemma. An auxiliary purpose is also to underline a set of guidelines on what changes to the practical regularity partitioning algorithm proved to be useful.

For this task we consider the Red Wine dataset which has 1599 instances with 11 attributes each. For the Red Wine dataset, the number of classes involved is six. It must be noted though that the class distribution in this dataset is pretty skewed (with the various classes having 10, 53, 681, 638, 199 and 18 datapoints respectively), this makes clustering this dataset quite difficult when k = 6. We however consider both k = 6 and k = 3 to compare results with spectral clustering.

Recall that our method has two meta-parameters that need to be user specified (or estimated by cross-validation) - ε and l. The first set of experiments thus explore the accuracy landscape of regularity clustering spanned over these two parameters. Care has to be taken that ε is not too large or small, so we consider 25 linearly spaced values of ε between 0.15 and

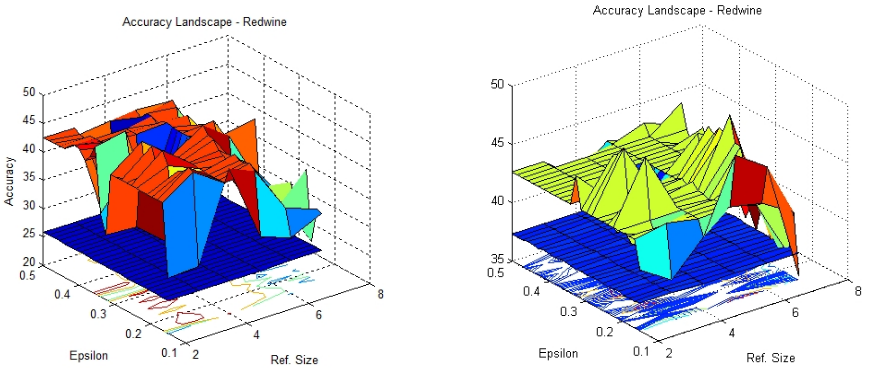

Figure 6.2: Accuracy Landscape for Regularity Clustering on the Red Wine Dataset for different values of ε and refinement size l (with k = 6 on the left and k = 3 on the right). The Plane cutting through in blue represents accuracy by running self-tuned spectral clustering using the fully connected similarity graph.

0.50. The "next refinement size", l as noted in Section 6.3 can not be too large. Since it can only take integer values, we consider six values from 2 to 7. For the sake of comparison, we also obtain clustering results on the same dataset with spectral clustering with self tuning [64] (both using all connected and k-nearest neighbor graph versions) and k-means clustering. We pick the variant of spectral clustering that is known to return the best results to make for a good comparison. Figure 6.2 gives the accuracy of the regularity clustering on a grid of ε and l. Even though this plot is only for exploratory purposes, it shows that the accuracy landscape is in general much better than the accuracy obtained by spectral clustering for this dataset. In this particular dataset it appears that the better performance of regularity clustering is not really too dependent on the choice of ε and l. We summarize results obtained by other methods on the Red Wine dataset in Table 6.1.

Table 6.2: Reduced Graph Sizes. Original Affinity Matrix size : 1599 × 1599

l ε	**2**	**3**	**4**	**5**	**6**	**7**
0.15	16 × 16	27 × 27	27 × 27	27 × 27	36 × 36	49 × 49
0.33	49 × 49	49 × 49	66 × 66	66 × 66	66 × 66	66 × 66
0.50	66 × 66	66 × 66	66 × 66	66 × 66	66 × 66	66 × 66

An important aspect of the regularity clustering method is that by using a modified constructive version of the Regularity Lemma we obtain a much reduced representation of the original data. The size of the reduced graph depends both on ε and l. However, in our observation it is more sensitive to changes to l and understandably so. From the grid for ε and l we take three rows to illustrate the obtained sizes of the reduced graph (more precisely, the dimensions of the affinity matrix of the reduced graph). We compare these numbers with the original dataset size. The compression obtained is quite striking. As we note in the results over the benchmark datasets in section 6.5.3, this compression is quite big in larger datasets.

The proof of the Regularity Lemma is using a potential function, the index of the partition defined earlier in Definition 3.3. In each refinement step the index increases significantly. Surprisingly this remains true in our modified refinement algorithm when the number of partition classes is not increasing as fast as in the original version, see Table 6.3. Another interesting observation is that if we take ε to be sufficiently high, we do get a ε-regular partition in just a few iterations. A few examples where this was noticed in the Red Wine dataset are mentioned in Table 6.4.

It is mentioned above that for refinement we only consider one ε-irregular

Table 6.3: Illustration of Increase in Potential

$ind(P)$ (ε,l)	$ind(P_1)$	$ind(P_2)$	$ind(P_3)$	$ind(P_4)$
0.15, 2	0.1966	0.2892	0.3321	0.3539
0.33, 2	0.1966	0.2883	0.3321	0.3683
0.50, 2	0.1965	0.2968	0.3411	0.3657

Table 6.4: Regular Partitions with required number of regular pairs and actual number present

$(\varepsilon$, $l)$	# for ε-regularity	# of Reg. Pairs	# Iterations
0.6, 2	1180	1293	6
0.7, 6	352	391	2
0.7, 7	506	671	2

pair for each class. Strategies for picking this irregular pair were also investigated and compared. Two natural strategies were tried: Picking a random irregular pair from the set of all irregular pairs and picking the most irregular pair. Intuitively, the second strategy should yield better results, but it was observed that this was rarely the case. It should be noted that the accuracy results reported earlier were based on choosing a random irregular pair.

Another aspect of the implementation that was investigated in detail was attempting to model the intra-cluster similarities. The practical regularity partitioning algorithm gives a method to model inter-cluster variations. However for clustering, modeling the intra-cluster variations are as important. One way of doing this is to sort the subsets in the refinement process by decreasing degree. By ordering subsets by degree it could be ensured that vertices with higher degrees remain in the same subset while the vertices with the lowest degree are put in the exceptional set. This seems intuitive as the vertices with the lowest degree would perhaps be leftovers. An unexpected advantage of ordering vertices is that the randomness in the algorithm is substantially reduced. Using the strategy outlined above (and with a random irregular pair and no ordering of vertices) causes some variations in the results on each run with the same meta-parameters. By ordering vertices this randomness is substantially reduced and the results are more stable. As for the most irregular pair, ordering vertices did not necessarily lead to better accuracy in all datasets. We consider exploring this aspect of the methodology an important aspect to fine-tune and refine. For now we only report results when the vertices are not ordered.

Finally, before reporting results we comment on constructing the reduced graph. The reduced graph was defined in Definition 3.7. But note that there is some ambiguity in our case when it comes to constructing the reduced graph. The reduced graph G^R is constructed such that the vertices correspond to the classes in the partition and the edges are associated to the ε-regular pairs between classes with density above d. However, in many cases the number of regular pairs is quite small (esp. when ε is small) making the matrix too sparse, making it difficult to find the eigenvectors. Thus for technical reasons we added all pairs to the reduced graph. We contend that this approach works well because the classes that we consider (and thus the densities between them) are obtained after the modified refinement procedure and thus enough information is already embedded in the reduced graph.

We now report clustering results on a number of benchmark datasets.

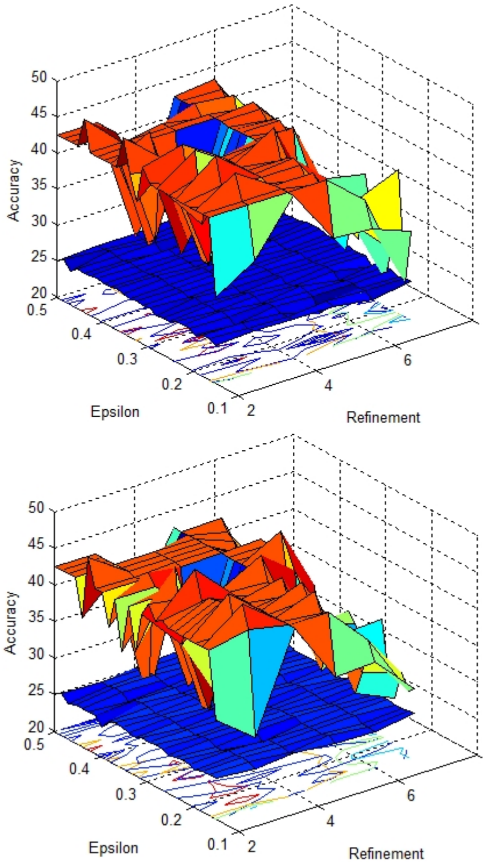

Figure 6.3: Accuracy Landscape on the Red Wine Dataset (with k = 6 on the left and k = 3 on the right) when the most irregular pair is considered in each refinement. The Plane cutting through in blue represents accuracy by running self-tuned spectral clustering using the fully connected similarity graph.

6.5.3 Clustering results on benchmark datasets

In this section we report results on a number of datasets described earlier in Section 6.5.1. We do a five fold cross-validation on each of the datasets, where a validation set is used to learn the meta parameters for the data. The accuracy reported is the average clustering quality on the rest of the data after using the learned parameters from the validation set. We use a grid-search to learn the meta-parameters. Initially a coarse grid is initialized with a set of 25 linearly spaced values for ε between 0.15 and 0.50 (we do not want ε to be outside this range). For l we simply pick values from 2 to 7 simply because that is the only practical range that we are looking at. This also justifies the use of grid-search in the following way: In the initial coarse grid search, because l can take only integer values, once a good value of l (with ε) has been identified the search becomes one dimensional (looking for the best ε given l) in the subsequent finer grid searches.

We compare our results with a fixed σ spectral clustering with both a fully connected graph (Spect2) and a k-nearest neighbour graph (Spect1). For the sake of comparison we also include results for k-means on the entire dataset. These results are reported in Table 6.5 (the best accuracy is indicated by bold-face). The results for the compression obtained on these datasets are reported in Table 6.6.

In these results we observe that the regularity clustering method, as indicated by the clustering accuracies is quite powerful; it gave significantly better results in 10 out of 12 datasets. It was also observed that the regularity clustering method did not appear to work very well in synthetic datasets.

Table 6.5: Clustering Results on UCI Datasets. Regular-A and Regular-FK represent the results obtained by the constructive versions due to Alon *et al.* and Frieze-Kannan, respectively. Spect1 and Spect2 give the results obtained by spectral clustering with a k-nearest neighbor graph and a fully connected graph, respectively. The best accuracy is indicated by bold-face. Follow the text for more details.

Dataset	**Regular - A**	**Regular - FK**	**Spect1**	**Spect2**	**k-means**
R-Wine	**47.0919**	**46.8342**	23.9525	23.9524	23.8899
W-Wine	**44.7509**	**44.9121**	23.1319	20.5798	23.8465
Arcene	**68**	**68**	61	62	59
Blood-T	**76.2032**	**75.1453**	65.1070	66.2331	72.3262
Ionos	**74.0741**	**74.6787**	70.0855	70.6553	71.2251
Cancer	93.5578	93.5578	**97.2182**	97.2173	96.0469
Pima	**65.1042**	**64.9691**	51.5625	60.8073	63.0156
Vertebral-1	67.7419	67.8030	**74.5161**	71.9355	67.0968
Vertebral-2	**70**	**69.9677**	49.3948	48.3871	65.4839
Steel	**42.5554**	**43.0006**	29.0057	34.7244	29.7785
Musk	**84.5862**	**81.4344**	53.9103	53.6072	53.9861
Haberman	**73.5294**	**70.6899**	52.2876	51.9608	52.2876

Table 6.6: Compression Obtained on the UCI Datasets

Dataset	**No. of Features**	**Original Dimension**	**Reduced Dimension**
R-Wine	11	1599×1599	49×49
W-Wine	11	4898×4898	125×125
Arcene	10000	200×200	9×9
Blood-T	4	748×748	49×49
Ionos	34	351×351	25×25
Cancer	9	683×683	52×52
Pima	8	768×768	52×52
Vertebral-1	6	310×310	25×25
Vertebral-2	6	310×310	25×25
Steel	27	1941×1941	54×54
Musk	166	6598×6598	126×126
Haberman	3	306×306	16×16

This seems understandable given the quasi-random aspect of the Regularity Method. We also report that the results obtained by the Alon *et al.* and by the Frieze-Kannan versions are virtually identical, which is not surprising.

Chapter 7

Prediction improvement using Regularity Clustering

We have also applied our new regularity clustering technique to an Educational Data Mining task: predicting student test result from features derived from tutors. (This work appears in *FLAIRS* 2013 [73]).

The data considered in this chapter comes from the ASSISTments system, a web-based tutoring system hosted by WPI, for 4th to 10th grade mathematics. The system is widely used in Northeastern United States by students in labs and for doing homework in the night.

7.1 Background

An important concept in student modeling is of “mastery learning” - that is, a student continues to learn a skill till mastery is achieved. Intuitively, whether a student will remember enough to answer a question after taking a break is a better definition of mastery as compared to a local measure based on next item response.

A recent work [80] drew our attention to the question whether such a near singular focus is important after all. That is, they found that features such as the number of distinct days that the student practiced a skill was more important than features that accounted for how many questions they got correct.

To attempt to improve upon Wang & Beck [80], we have used the technique of using clustering to generate an ensemble introduced by [77] to see if we can improve our predictions. The research question that we have is: Can we employ this technique to increase accuracy in predicting long term retention? In [78] it was found that spectral clustering was more effective than k-means for this type of work. It is natural to ask: “How does regularity clustering compare in performance with spectral and k-means?” In the next section we review a technique that uses (general) clustering for bootstrapping.

7.2 Clustering students and strategy for bootstrapping

The idea that students are perhaps quite different when it comes to forgetting makes it quite apparent that it is perhaps not a good idea to fit a global model on all of the data. In spite of individual differences, we hypothesize that broadly the patterns and underlying reasons of forgetting would fall into several coarse groups, with each such group having students more "similar" to each other in regard to forgetting. Honing on this intuition, it might make more sense to cluster students into somewhat homogeneous groups and then train a predictor separately on each such group, which considers only the points from that cluster as the training set for itself. It is clear that each such predictor would be a better representative for that group of students as compared to a single global predictor trained on all the students at one time. While this idea sounds compelling, there is a major issue with it. While it is useful to model students as belonging to different groups, it is perhaps not a good idea to simply divide them into clusters. This is because the groupings are usually not very clear. For example, a student might be extremely good at retaining information about certain aspects of Trigonometry but not other aspects, while at the same time might be strong with retaining algebra. Such complex characteristics can not be modeled by a simplistic solution as only clustering the data to some upper limit and then training predictors on each cluster. The "fuzzy" nature of such a process, which is like a spread of features across groups needs to be captured to make a distributive model such as the above more meaningful. This issue can be fixed by varying the granularity of the clustering and training separate models each time so the such features can be accounted for. A simple strategy to do so was proposed recently and was found quite useful in various tasks in student modeling [77], [78].

The technique is actually a simple ensemble method. The basic idea behind ensemble methods is that they involve running a "base learning algorithm" multiple times, each time with some change in the representation of the input (e.g. considering only a subset of the training examples or a subset of features etc) so that a number of diverse predictions can be obtained. This process also gives a rich representation of the input, which is one of the reasons why they work so well. In the particular case of our method, unlike many other ensemble methods that use a random subset to bootstrap, we use clustering to bootstrap. The training set is first clustered into k disjoint clusters and then a logistic regression model is trained on each of the clusters only based on the training points that were assigned to that cluster. Each such model, being a representative of a cluster is referred to as a *cluster model*. Thus for a given value of k there would be k cluster models. Note that since all the clusters are mutually exclusive, the training set is represented by all the k *cluster models* taken together. We refer to this as a *Prediction Model*, PM_k. For an incoming test point, we first figure out the cluster that point belongs to and then use the concerned cluster model alone to make a prediction on that point. Now also note that we don't specify the number of clusters above. Hence, we can change the granularity of the clustering from 1 (PM_1, which is the entire dataset as one cluster) to some high value K. In each such instance we would get a different *Prediction Model*, thus obtaining a set of k *Prediction Models*. Since the granularity of the clustering is varied, the predictions obtained would be diverse and hence could be combined together by some method such as averaging them together to get a single prediction.

Note that the clustering algorithm above is not specified and hence could be any clustering technique, as long as there is a straightforward way to map test points to clusters. In particular we clustered students using three algorithms: k-means [41], spectral clustering [55] and our regularity clustering

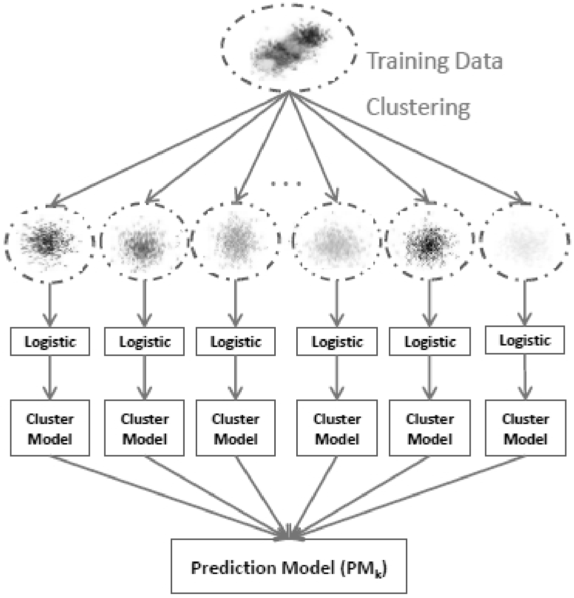

Figure 7.1: Construction of a Prediction Model for a given K. See text for details

[72], then we compare the accuracy using different algorithms.

7.3 Dataset description and experimental results

The dataset used is the same as used in [80]. The only exception being that we considered the data for a unique 1969 students and did not consider multiple data points of the same student attempting something from a different skill. This was only done because we were interested in clustering students according to *user_id*. The following features were used. The goal was to predict whether a response was correct i.e. 1 or incorrect 0.

1. *n_correct:* the number of prior student correct responses on this skill; This feature along with *n_incorrect*, the number of prior incorrect re-

sponses on this skill are both used in PFA models.

2. *n_day_seen:* the number of distinct days on which students practiced this skill. This feature distinguishes the students who practiced more days with fewer opportunities each day from those who practiced fewer days but more intensely, and allow us to evaluate the difference between these two situations. This feature was designed to capture certain spaced practice effect in students data.

3. *g_mean_performance:* the geometric mean of students previous performances, using a decay of 0.7. For a given student and a given skill, use opp to represent the opportunity count the student has on this skill, we compute the geometric mean of students previous performance using formula: $g\_mean\_performance(opp) = g\_mean\_performance(opp - 1) \times 0.7 + correctness(opp) \times 0.3$. The geometric mean method allows us to examine current status with a decaying memory of history data. The number 0.7 was selected based on experimenting with different values.

4. *g_mean_time:* the geometric mean of students previous response time, using a decay of 0.7. Similar with g_mean_performance, for a given student and a given skill, the formula of the geometric mean of students previous response time is: $g\_mean\_time(opp) = g\_mean\_time(opp - 1) \times 0.7 + response\_time(opp) \times 0.3$.

5. *slope_3:* the slope of students most recent three performances. The slope information helps capture the influence of recent trends of stu-

dent performance.

6. *delay_since_last:* the number of days since the student last saw the skill. This feature was designed to account for a gradual forgetting of information by the student.

7. *problem_difficulty:* the difficulty of the problem. The problem_difficulty term is actually the problem easiness in our model, since it is represented using the percent correct for this problem across all students. The higher this value is, the more likely the problem can be answered correctly.

Out of these features it was reported that features such as *n_correct* and *n_incorrect* had very little influence on the prediction performance while the features *g_mean_performance* and *n_day_seen* appear to be reliable predictors of student retention. This observation is consistent with the spaced practice effect in cognitive science. Hence, in our experiments we don't consider *n_correct* and *n_incorrect* while training the model. As mentioned before, we used *k*-means, Spectral and Regularity Clustering in conjunction with the ensemble technique described. It must also be noted that the features were normalized to values between -1 and 1 to avoid undue dominance of performance by a specific feature. The results obtained were rather surprising. The use of *k*-means clustering and Spectral Clustering, that has been reported useful in other tasks does not seem to help in the case of predicting long term retention (at least on this data). The baseline model used by Wang & Beck is represented in Figure 7.2 by PM_1, the starting point on the x-axis. The other values on the x-axis represent how many *Prediction*

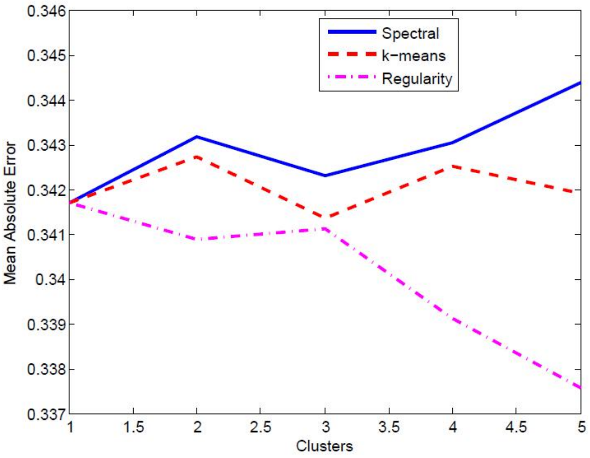

Figure 7.2: Mean Absolute Errors on Using the three Clustering Techniques for Bagging

Models were averaged. The errors reported are the mean absolute errors. As reported in Table 7.1, the ensemble used in conjunction with Regularity Clustering is significantly better than the baseline with strong p-values.

The Paired t-test compares the means of two variables. The p-value is the probability of the differences in the variables generated from the same population by chance. It is calculated using the outcome of the t-test. In our case, the less p-value is, the more reliable of our test result. A convention in statistics is to accept the result with p-value less than 0.05.

Table 7.1: Paired t-tests on the predictions obtained with the baseline (PM_1) and regularity clustering

Pred. Models	Baseline & Regularity
1	-
2	0.00531
3	**0.0401**
4	**0.0018**
5	**0.0044**

Table 7.2: Paired t-tests on the predictions obtained with spectral and with regularity clustering at different k

Pred. Models	Spectral & Regularity
1	-
2	0.1086
3	**0.0818**
4	**0.0045**
5	$\ll$ **0.005**

Part II

Theoretical applications of the Regularity Lemma

Chapter 8

Background

In [36] Gyárfás conjectured the following:

Conjecture 8.1 (Gyárfás, 1989 [36]) *If the edges of a finite undirected complete graph K are colored with r colors, then the vertex set of K can be covered by at most $f(r)$ vertex disjoint monochromatic paths.*

The key part of this conjecture is that this partition number depends only on r. It means that no matter how large the graph is, we are able to cover all vertices by monochromatic paths, and the number of these paths is only determined by the number of colors to color the edges and it does not depend on n.

A natural generalization is the following. Assume that K_n is a complete graph with n vertices, its edges are colored with r colors and $\mathcal{H}$ is a family of graphs, then how many monochromatic subgraphs from $\mathcal{H}$ are needed to cover all the vertices of K_n.

We call questions of this type the monochromatic vertex partition problem and denote the number of subgraphs needed as

$$p(r, \mathcal{H}).$$

The study of this quantity is the main goal of this part of this thesis. This problem is in extremal graph theory. Extremal graph theory studies extremal (maximum or minimum) graphs which satisfy certain properties. It has several important branches which are closely related to our topic, in this chapter we will review three of them: Turán-type questions, Ramsey theory and the largest monochromatic subgraph problem. Several of the results reviewed here will be used later.

We begin our discussion with Turán-type questions.

8.1 Turán-type questions

A vitally important question in extremal graph theory is to determine the size of the largest subgraph given some properties of the original graph, especially how many edges a graph must contain to ensure the existence of a certain subgraph. Here, the most important result is Turán's Theorem. To address it formally, we need to define the Turán graph first.

Below is the definition of the Turán graph. For simplicity we will assume n is divisible by r. The definition of r-partite graph can be found at Section 2.1.

Definition 8.2 *the Turán graph $T_r(n)$ is a complete r-partite graph on n*

vertices such that each partition class has exactly $\frac{n}{r}$ *vertices.*

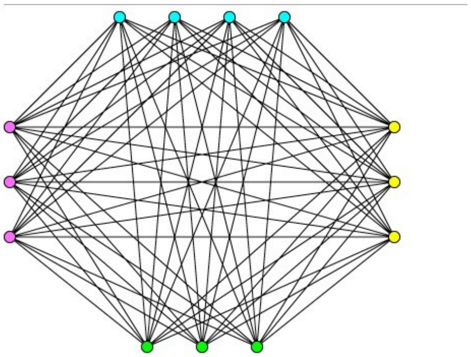

Figure 8.1: Turán graph

The Turán graph satisfies the following: choose any $r+1$ vertices, there must be at least 2 vertices x and y from the same partition class; and by the definition of r-partite graphs, there is no edge xy, so the Turán graph does not contain a K_{r+1} (see Section 2.1) as a subgraph. The importance of the Turán graph is that it gives an optimal construction for a K_{r+1}-free graph.

Each vertex of a Turán graph has degree $n - \frac{n}{r}$, thus the number of edges of a Turán graph is

$$\frac{(n-\frac{n}{r})n}{2} = \frac{r-1}{2r}n^2$$

and if n is not divisible by r the edges will be slightly less than this. Turán proved that this is an upper bound for the number of edges of a K_{r+1}-free graph.

Theorem 8.3 (Turán, 1941 [76]) *Let G be a graph on n vertices and contains no K_{r+1} as a subgraph, then G has at most $\frac{r-1}{2r}n^2$ edges.*

Here generally, a Turán-type question is the following: for a graph G_n, if it contains no subgraph from a family $\mathcal{H}$, what is the maximum number of edges G_n can have. Formally we define the maximum number of edges as $ex(n, H)$:

Definition 8.4 *Given a fixed graph H,*

$$ex(n, H) = max\{|E(G)||H \not\subset G, |V(G)| = n\}.$$

So Turán theorem says

$$ex(n, K_{r+1}) = \frac{r-1}{2r}n^2.$$

The special case $r = 2$ is one of the earliest Turán-type results, Mantel's Theorem.

Theorem 8.5 (Mantel, 1907) *If a simple graph on n vertices has more than $\lfloor \frac{n^2}{4} \rfloor$ edges, then it contains a triangle.*

A famous generalization of Turán's theorem is the Erdős-Stone theorem, where instead of a complete subgraph, they find a complete r-partite graph with equal size t of each partition class.

Theorem 8.6 (Erdős, Stone, 1946 [16]) *For $r \geq 2$,*

$$en(n, K_{r+1}(t, \ldots, t)) = (\frac{r-1}{2r})\binom{n}{2} + o(n^2).$$

Notice that t does not show up in the formula. To understand the importance of the Erdős-Stone Theorem, first let us look at the definition of the chromatic number of a graph:

Definition 8.7 *A proper coloring of a graph G is a function from the vertices to a set C of colors such that the end points of every edge have distinct colors. The chromatic number $\chi(G)$ of a graph G is the minimal number of colors for which a proper coloring exists.*

We get a generalization of Theorem 8.6 when we look for subgraphs of a given chromatic number.

Theorem 8.8 (Erdős, Stone, 1946 [16]; Erdős-Simonovits, 1966 [17]) *Let H be a fixed graph with $\chi(H) = r + 1$ then*

$$ex(n, H) = (\frac{r-1}{2r})\binom{n}{2} + o(n^2)$$

Note that $\chi(K_{r+1}(t, \ldots, t)) = r + 1$, so Theorem 8.6 is a special case of Theorem 8.8.

For a bipartite graph H, $r = 1$, this theorem just says that $ex(n, H) = o(n^2)$; for a non-bipartite graph it provides the general asymptotic solution. Refer to [13] for generalizations for different kinds of H.

In [14], Erdős and Gallai have proved a classical Turán-type result about the occurrence of a cycle.

Theorem 8.9 (Erdős, Gallai, 1959 [14]) *Every graph with n nodes and more than $\frac{(n-1)l}{2}$ edges ($l \geq 2$) contains a cycle with more than l edges.*

In an r-colored complete graph, select the most frequent color, say red; by the pigeon hole principle, the number of red edges is at least $\frac{n(n-1)}{2r}$, so by Theorem 8.9, we can claim that the largest monochromatic path or cycle contains at least $\frac{n}{r}$ vertices. We will formalize this theorem later.

Thus we can use the above Turán-type Theorem 8.9 to give a lower bound on the size of the largest monochromatic cycle. In addition, it is shown in the same paper [14] that this bound is best possible for general graphs.

8.2 Ramsey Theory

Turán-type questions try to identify the conditions to ensure the existence of certain subgraphs. In Ramsey theory, we have an r-coloring of the *complete* graph, and we try to find the conditions to ensure the existence of certain monochromatic subgraphs.

First let us take a look at the well-known pigeonhole principle:

Theorem 8.10 (Pigeonhole Principle) *If $n > r$ and n items are put into r pigeonholes, then there must be at least one pigeonhole containing at least 2 items.*

Now try to rephrase it in another way: suppose we have n vertices and r colors, $n > r$, color all these vertices, then there must be at least 2 vertices with the same color.

This is the simplest case of Ramsey theory which colors the 1-subsets (vertices) of the vertex set, and $r + 1$ is the Ramsey Number. It can be formalized as

$$R_1(2, \ldots, 2) = r + 1$$

The 2-subset version of Ramsey theory colors the edges. The simplest case of this version is a 2-coloring: define $R_2(m_1, m_2)$ to be the minimum number n such that if each edge of K_n is colored red of blue, then there must exist either a red K_{m_1} or a blue K_{m_2} subgraph. An example is the well-known fact (also known as the 6-party theorem):

$$R_2(3,3) = 6$$

This means that we have a 2-coloring of K_5 with no monochromatic triangles, but for any 2-coloring of K_6 there must be a monochromatic triangle.

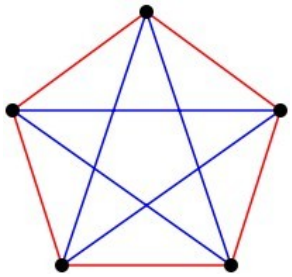

Figure 8.2: A 2-coloring of K_5 with no monochromatic triangle

Now we generalize this to r-colorings of the edges:

Definition 8.11 *$R_2(m_1, \ldots, m_r)$ is the minimum n such that if the edges of K_n are colored by r colors, then there must exist a monochromatic K_{m_i} subgraph in color i.*

Finally, we come to the general version of Ramsey Theorem:

Theorem 8.12 (Ramsey, 1930 [61]) *Let $u \geq 1$ and $m_i \geq u$, $i = 1, 2, \ldots, r$ be given. There exists a minimal positive integer $R_u(m_1, m_2, \ldots, m_r)$ with the following property. Let S be a set with n elements. Suppose that all $\binom{n}{u}$*

u-subsets of S are divided into r mutually exclusive families $T_1, \ldots, T_r$. Then if $n \geq R_u(m_1, m_2, \ldots, m_r)$ there is an i, $1 \leq i \leq r$, and some m_i-subset of S for which every u-subset is in T_i.

Definition 8.13 *This minimal positive integer $R_u(m_1, m_2, \ldots, m_r)$ in Ramsey's Theorem is called the Ramsey number.*

In the above for $r > 2$ we get hypergraphs, in this thesis we only consider graphs $n = 2$, so we drop the index 2.

Below we list some important results from Ramsey theory.

Theorem 8.14 (Greenwood, Gleason, 1955 [26]) *$R(k, l) \leq R(k-1, l) + R(k, l-1)$.*

The inequality is strict when both terms on the right hand side are even.

Theorem 8.15 (Harary, 1972 [38]) *$R(K_{1,n}, K_{1,m}) = n + m - \varepsilon$, where $\varepsilon = 1$ for even n and m, and $\varepsilon = 0$ otherwise.*

For a 2-colored complete graph we have:

Theorem 8.16 (Gerencseŕ, Gyárfás, 1967 [25]) *$R(P_n, P_m) = n + \lfloor \frac{m}{2} \rfloor - 1$ for all $n \geq m \geq 2$.*

For a 3-colored complete graph, Gyárfás, Ruszinkó, Sárközy and Szemerédi [32] established that for sufficiently large n:

Theorem 8.17 (Gyárfás, Ruszinkó, Sárközy, Szemerédi, 2007 [32])

$$R(P_n, P_n, P_n) = \begin{cases} 2n - 1: & \textit{for odd } n \\ 2n - 2: & \textit{for even } n \end{cases}$$

8.3 Largest monochromatic subgraphs

Ramsey theory studies how large the graph is to ensure the existence of a certain monochromatic subgraph. The opposite direction is to ask, given an r-coloring of the complete graph, what is the size of the largest monochromatic subgraph. In this section we will list some best results for the largest monochromatic subgraph problem in an r-colored graph. We will start with 2-colorings, then give some results for r-colorings.

The first important result in this area is that every 2-colored complete graph has a monochromatic spanning tree, it is a remark of Erdős and Rado:

Theorem 8.18 (Erdős, Rado) *Every 2-colored complete graph has a monochromatic spanning tree.*

Since a tree is a connected component, a natural generalization is to ask for the largest monochromatic k-connected subgraph in a 2-coloring of K_n. This was studied in [5] by Bollobás and Gyárfás:

Theorem 8.19 (Bollobás, Gyárfás, 2008 [5]) *For $n \geq 5$ there is a monochromatic 2-connected subgraph with at least $n - 2$ vertices in every 2-coloring of K_n.*

There is a conjectures about this question in the same paper [5]:

Conjecture 8.20 (Bollobás, Gyárfás, 2008 [5]) *For $n > 4(k-1)$, every 2-colored K_n has a k-connected monochromatic subgraph with at least $n - 2(k-1)$ vertices.*

In the same paper the authors proved that this conjecture is true for $k \leq 2$. Liu, Morris and Prince [53] showed the conjecture holds for $k = 3$. Fujita and Magnant have proved a weaker version of this conjecture:

Theorem 8.21 (Fujita, Magnan, 2011 [22]) *For $n > 6.5(k-1)$, every 2-colored K_n has a k-connected monochromatic subgraph with at least $n - 2(k-1)$ vertices.*

For the double star (a special tree), Gyárfás and Sárközy [31] has proved:

Theorem 8.22 (Gyárfás, Sárközy, 2008 [31]) *In every 2-coloring of K_n there is a monochromatic double star with at least $\frac{3n+1}{4}$ vertices.*

Now we take a look at the generalizations to r-colorings.

Theorem 8.18 has been generalized to r-colorings by Gyárfás [24]:

Theorem 8.23 (Gyárfás, 1971 [24]) *In every r-coloring of K_n there is a monochromatic component with at least $\frac{n}{r-1}$ vertices.*

This result is sharp if $r-1$ is a prime power and $(r-1)^2$ divides n. The proof is based on the following lemma:

Lemma 8.24 *In every r-coloring of a complete bipartite graph on n vertices there is a monochromatic subtree with at least $\frac{n}{r}$ vertices.*

A similar lemma for a double star is given in [53] and [56] :

Lemma 8.25 (Liu, Morris, Prince, 2009 [53]; Mubayi, 2002 [56]) *In every r-coloring of a complete bipartite graph on n vertices there is a monochromatic double star with at least $\frac{n}{r}$ vertices.*

A corollary of Lemma 8.25 is the following:

Corollary 8.26 *Suppose that the edges of K_n are colored with r colors. Then either all color classes have monochromatic spanning trees or there is a monochromatic double star with at least $\frac{n}{r-1}$ vertices.*

This raises the question to find the largest monochromatic double star in an r-coloring. Gyárfás and Sárközy have investigated this problem. Their conclusion [31] is:

Theorem 8.27 (Gyárfás, Sárközy, 2008 [31]) *For $r \geq 2$ there is a monochromatic double star with at least $\frac{n(r+1)+r-1}{r^2}$ vertices in any r-coloring of the edges of K_n.*

The bound in this theorem is close to best possible for $r = 2$, the existence of such a 2-coloring is proved by the random method. However, for $r \geq 3$ the random method seems to fail to provide good bounds for such a function and it is conceivable that it is $\frac{n}{r-1}$, a good test case would be $r = 3$.

For paths and cycles, as in the discussion followed by Theorem 8.9, the largest monochromatic path or cycle has size at least $\frac{n}{r}$.

Theorem 8.28 (Erdős, Gallai, 1959 [14]) *In an r-colored complete graph, there exists a monochromatic cycle (path) of length at least $\frac{n}{r}$.*

Furthermore, our recent result gives a bound on the largest connected monochromatic k-regular subgraph in an r-colored complete graph:

Theorem 8.29 (Sárközy, Selkow, Song, 2013 [71]) *For every positive ε and integers $r, k \geq 2$ there exists a constant $n_0 = n_0(\varepsilon, r, k)$ such that* *for any r-coloring of the edges of a complete graph on $n \geq n_0$ vertices, we can find a connected monochromatic k-regular subgraph spanning at least $(1-\varepsilon)n/r$ vertices.*

Chapter 9

Monochromatic Vertex partitions

Let us recall the monochromatic vertex partition problem; i.e. the study of $P(r, \mathcal{H})$. In the last section of the last chapter, we have listed some results for the size of the largest monochromatic subgraph in an r-colored complete graph. Based on this information, we discuss potential ways to solve the monochromatic vertex partition problem.

9.1 First idea: the greedy procedure

A natural approach is to use a greedy procedure. Take the largest monochromatic substructure, then remove the vertices belonging to it, consider the leftover vertices to form a subgraph $G' \subset G$, and continuously keep removing monochromatic substructures in this manner.

Let us analyze the greedy procedure for cycles (so $\mathcal{H}$ is the family of cycle). G is a complete graph on n vertices and its edges are r edge colored, the question is how many monochromatic vertex disjoint cycles are needed to cover all the vertices. Here single vertices and edges are considered to be (degenerate) cycles. By Theorem 8.28, the largest monochromatic cycle contains at least $\frac{n}{r}$ vertices. Then applying the greedy procedure described above, after t iterations, the number of leftover vertices is at most:

$$u = n(1 - \frac{1}{r})^t$$

In order to cover all the vertices by the greedy procedure only, we need to make sure that the number of leftover vertices is a constant. Then we can cover it by constant monochromatic cycles (isolated vertices). Suppose this constant number of vertices is c. Then we need to ensure:

$$n(1 - \frac{1}{r})^t \leq c \tag{9.1}$$

Since

$$1 - x \leq e^{-x},$$

(9.1) is true, if the following is true:

$$ne^{-\frac{t}{r}} \leq c.$$

From this we get

$$t \geq r \log \frac{n}{c}.$$

Thus we get the conclusion:

$$t \geq c_1 r \log n.$$

This gives an answer to the vertex partition problem, which is to cover all the vertices of an r-colored complete graph with vertex-disjoint monochromatic subgraphs $\mathcal{H}$, in our case, by cycles. Later we will show that this answer $O(r \log n)$ is far from optimal since it depends on n and the optimal answer will not. To improve, let us analyze the greedy procedure first.

The greedy procedure has two steps. The first step is to remove the largest monochromatic subgraphs greedily; after t steps there would be only a constant number of vertices left. The second step is to cover the leftovers by u subgraphs. Since there are only a constant number of vertices left, we can treat them as single vertices. We got the conclusion that for the cycle partition, $t = O(r \log n)$ and $u = O(1)$.

To improve on t, we need to enlarge the leftover set as a function $l(r, n)$. By the same analysis as before, we need to ensure:

$$n(1 - \frac{1}{r})^t \leq l(r, n).$$

Then we get the conclusion:

$$t \geq r \log \frac{n}{l(r, n)}.$$

This way we improve on t, but now we have $l(r, n)$ vertices left uncovered, u is not a constant anymore. If we still treat these vertices as single vertices, then $u = O(l(r,n))$, which will make our bound even worse. Therefore we need to find another way to cover the leftover vertices, the greedy procedure alone is not sufficient.

9.2 The Absorbing procedure

In a landmark paper [15], Erdős, Gyárfás and Pyber proved that $p(r, cycles) \leq cr^2 \log r$ with some constant c, thus they also proved Conjecture 8.1 with $f(r) = cr^2 \log r$. Their approach has become a standard proof technique in this research area.

Theorem 9.1 (Erdős, Gyárfás, Pyber, 1991 [15]) *If the edges of a finite complete graph K are colored with r colors then the vertex set of K can be covered by at most $cr^2 \log r$ vertex disjoint monochromatic cycles.*

We will discuss this absorbing idea in general first and present their proof details later.

The absorbing procedure has three steps. In the first step, instead of just finding the largest monochromatic subgraph, we try to seek a smaller monochromatic subgraph, with the following additional property: removing some portion from this subgraph, we can still find a spanning monochromatic subgraph. This property will be used to cover the leftovers after the greedy procedure.

The second step is the greedy procedure until $l(r,n)$ vertices left uncovered.

The third step is to cover the leftovers by using the subgraph found in the first step. This step involves the vertex partition for unbalanced complete bipartite graphs which is interesting on its own.

The special subgraph mentioned above is a triangle cycle in [15]:

Definition 9.2 *A triangle cycle of length k, T_k, is a cycle $a_1, a_2, \ldots, a_k$ of length k and k further vertices $b_1, b_2, \ldots, b_k$ such that b_i is adjacent to a_i and to a_{i+1} for $i = 1, 2, \ldots, k (a_{k+1} = a_1)$.*

The property of T_k that is important to us is that T_k has a Hamiltonian cycle (see Section 2.1 after the deletion of any subset of $\{b_1, b_2, \ldots, b_k\}$.

In the same paper the authors also proved a lemma on the Ramsey number of a triangle cycle:

Lemma 9.3 (Erdős-Gyárfás-Pyber, 1991 [15]) *If the edges of K_n are colored with r colors then there exists a monochromatic T_k with $k \geq \frac{cn}{r(r!)^3}$.*

To prove their main theorem, the authors also proved the following:

Theorem 9.4 (Erdős, Gyárfás, Pyber, 1991 [15]) *Assume that the edges of the complete bipartite graph (A, B) are colored with r colors. If $|B| \leq \frac{|A|}{r^3}$ then B can be covered by at most r^2 vertex disjoint monochromatic cycles.*

Now we demonstrate the proof of Theorem 9.1. It follows the general absorbing proof technique described above.

Proof:

- Step 1: By Lemma 9.3 we can find a sufficiently large monochromatic, say red, triangle cycle T_k. More specifically the size of this triangle cycle is at least $k \geq \frac{cn}{r(r!)^3}$; let X denote the set $\{b_1, \ldots, b_k\}$.

- Step 2: By Theorem 8.28, K_n contains a monochromatic cycle of legth at least $\frac{n}{r}$. Apply repeatedly this fact to $K_n - T_k$ until the leftover vertices are small. How many times do we need to apply this? To use Theorem 9.4, we need to repeat t times until the leftover is smaller than $\frac{k}{r^3}$, which means

$$(n-2k)(1-\frac{1}{r})^t \leq \frac{k}{r^3}$$

  Calculation shows that $t = \lfloor cr^2 \log r \rfloor$ is good enough with some constant c; denote the set of leftover vertices by Y.

- Step 3: Using Theorem 9.4, for the unbalanced r-colored complete bipartite graph (X, Y), cover Y by at most r^2 vertex disjoint monochromatic cycles. By the above property of the triangle cycle, after removing the vertices to cover Y, it still can be covered by one red cycle.

□

9.3 Apply the Regularity Lemma and the Blow-up Lemma

Many of the up-to-date results for the monochromatic vertex partition problem are using the Regularity Lemma (Chapter 3) as a central tool. Here we use the proof of Theorem 8.29 as an example to show how to use the Regularity Lemma to prove a theoretical result (and we will need Theorem 8.29 later).

First, we need a definition:

Definition 9.5 *(A, B, E) is (ε, δ, G)-super-regular if it is (ε, G)-regular and*

$$deg_G(a) > \delta|B| \ \ \forall \, a \in A, \qquad deg_G(b) > \delta|A| \ \ \forall \, b \in B.$$

Then we need to state a lemma:

Lemma 9.6 *Given $|A| = |B|$, density $\varepsilon \ll d$, for any (ε, G)-regular graph (A, B) that has density $\geq d$, by removing no more than ε portion vertices from each part we can get a induced $(\frac{\varepsilon}{1-\varepsilon}, \frac{d-2\varepsilon}{1-\varepsilon}, G_1)$-super-regular subgraph $G_1 = (A_1, B_1)$, $|A_1| = |B_1| = (1-\varepsilon)|A|$.*

Now we present the proof for Theorem 8.29, which is to find the largest monochromatic k-regular subgraph in an r-colored complete graph.

Proof: We will assume that n is sufficiently large, and

$$0 < \varepsilon \ll \delta \ll 1.$$

- Step 1: Construct the reduced graph G^R (Definition 3.7). For an r-colored complete graph $G = (G_1, \ldots, G_r)$, apply the r-color version of Regularity Lemma, get a partition of $V(G) = \cup_{0 \leq i \leq l} V_i$, where $|V_i| = m, 1 \leq i \leq l$. We define the reduced graph G^R: The vertices of G^R are $p_1, \ldots, p_l$ corresponding to $V_1, \ldots, V_l$, and we have an edge between vertices p_i and p_j if the pair $\{V_i, V_j\}$ is (ε, G_s)-regular for $s = 1, 2, \ldots, r$. Then,

$$|E(G^R)| \geq (1-\varepsilon)\binom{l}{2},$$

and thus G^R is a $(1-\varepsilon)$-dense graph on l vertices. Define an edge-coloring $(G_1^R, G_2^R, \ldots, G_r^R)$ of G^R by r colors in the following way. The edge $p_i p_j$ is colored with a color s that contains the most edges from $K(V_i, V_j)$. Let us take the color class in this coloring that has the most edges, for simplicity assume that this is G_1^R and call this color red. Clearly, we have

$$|E(G_1^R)| \geq (1-\varepsilon)\frac{1}{r}\binom{l}{2},$$

- Step 2: Find a "fat" cycle. More precisely, in G_1^R, apply Theorem 8.9 to find a cycle C of length at least $(1-\varepsilon)\frac{1}{r}l$. According to lemma 9.6, by removing at most 2ε portion vertices we can make all pairs along the cycle super-regular.

- Step 3: By using the Blow-up lemma [46] (see below), we are able to find a red connected spanning k-regular subgraph within the remainder of C.

□

The Regularity Lemma is a powerful tool for embedding subgraphs into dense graphs. However, as we have seen in the example above, to embed spanning subgraphs, all degrees of the host graph are required to be large. That is why solely using regular pairs is not sufficient, we need super-regular pairs. The Blow-up Lemma plays an important role for embedding spanning subgraphs :

Theorem 9.7 (Komlós, Sárközy, Szemerédi, 1997 [46], 1998 [47]) *Given a graph R of order r and positive parameters δ, Δ, there exists a positive $\varepsilon = \varepsilon(\delta, \Delta, r)$ such that the following holds. Let $n_1, n_2, \ldots, n_r$ be arbitrary positive integers and let us replace the vertices $v_1, v_2, \ldots, v_r$ of R with pairwise disjoint sets $V_1, V_2, \ldots, V_r$ of sizes $n_1, n_2, \ldots, n_r$ (blowing up). We consturct two graphs on the same vertex-set $V = \cup V_i$. The first graph R is obtained by replacing each edge $\{v_i, v_j\}$ of R with the complete bipartite graph between the corresponding vertex-sets V_i and V_j. A sparser graph G is constructed by replaing each edge $\{v_i, v_j\}$ arbitrarily with an (ε, δ)-super-regular pair between V_i and V_j. If a graph H with $\Delta(H) \leq \Delta$ is embeddable into R then it is already embeddable into G.*

To make it short, the Blow-up Lemma states that regular pairs behave as complete bipartite graphs from the point of view of embedding bounded degree subgraphs.

Now we list some important results on the monochromatic vertex partition problem.

9.4 Best known results

9.4.1 Unbalanced complete bipartite graphs

In [33] the authors made a significant improvement on Theorem 9.4:

Theorem 9.8 (Gyárfás, Ruszinkó, Sárközy, Szemerédi, 2006 [33]) *There exists a constant $n_0(r)$ such that the following is true. Assume that the edges of the complete bipartite graph $K(A, B)$ are colored with r colors.*

If $|A| \geq n_0$, $|B| \leq \frac{|A|}{r^2}$, then B can be covered by at most $(6r\lceil \log r\rceil + 2r)$ vertex disjoint monochromatic cycles.

Then, the same authors improved this even further [35]:

Theorem 9.9 (Gyárfás, Ruszinkó, Sárközy, Szemerédi, 2006 [35]) *For every fixed r there exists a $n_0 = n_0(r)$ such that the following is true. Assume that the edges of the complete bipartite graph $K(A, B)$ are colored with r colors. If $|A| \geq n_0$, $|A| \geq 2r|B|$, then B can be covered by at most $3r$ vertex disjoint monochromatic cycles.*

A similar result has been established by Sárközy and Selkow in [69] for k-regular subgraphs.

Theorem 9.10 (Sárközy, Selkow, 2000 [69]) *If the edges of the complete bipartite graph (S, Y) are colored with r colors, $|S| = m$ and $|Y| < \frac{m}{x^2}$ (where x is defined as $x = 2r^2(2er)^{\lceil \frac{k}{2}\rceil}$), then the vertices of Y can be covered by at most $rx(1 + \lceil \frac{k}{2}\rceil) + 2r^2\lceil \frac{k}{2}\rceil$ vertex-disjoint connected monochromatic k-regular graphs and vertices.*

9.4.2 Monochromatic Cycles, Trees and k-regular subgraphs

In [15] (see also [36]) the authors construct an example to show that the path (and cycle) partition number is at least r:

Consider pairwise disjoint sets $A_1, A_2, \ldots, A_r$ and for $x \in A_i, y \in A_j, i \leq j$, color the edge xy with color i. If the sequence $|A_i|$ grows fast enough then the vertex set of this r-colored complete graph cannot be covered by less than r monochromatic paths. Motivated by this example they refined Conjecture 8.1 to the following remarkable conjecture.

Conjecture 9.11 (Erdős, Gyárfás, Pyber, 1991 [15]) *$p(r) = r$, where $p(r)$ is the cycle partition number.*

Unfortunately, a counterexample has been found by Pokrovskiy [59] recently. However, the counterexample is quite "weak", in it all but one vertex can be covered by r vertex disjoint monochromatic cycles. Perhaps, a weakening of the conjecture is true : apart from a constant number of vertices all vertices can be covered by r monochromatic vertex disjoint cycles.

The current best result is due to Gyárfás, Ruszinkó, Sárközy and Szemerédi [33]. They follow the same proof methodology as Erdős, Gyárfás and Pyber in Theorem 9.1, by making improvements on Step 1 and Step 3 to achieve a better bound:

Theorem 9.12 (Gyárfás, Ruszinkó, Sárközy, Szemerédi, 2006 [33]) *For every integer $r \geq 2$ there exists a constant $n_0 = n_0(r)$ such that if $n \geq n_0$ and the edges of the complete graph K_n are colored with r colors then the vertex set of K_n can be partitioned into at most $100r \log r$ vertex disjoint monochromatic cycles.*

To present the proof we need the following definition and lemmas:

Definition 9.13 *A matching in a graph G is a set of edges without common vertices. A matching in a graph G is called k-half dense if one can label its edges as $x_1y_1, \ldots, x_{|M|}y_{|M|}$ so that each vertex of $X = \{x_1, \ldots, x_{|M|}\}$*

(called the strong end points) is adjacent in G to at least k vertices of $Y = \{y_1, \ldots, y_{|M|}\}$.

Lemma 9.14 *For every $\delta > 0$ there exist an $\varepsilon > 0$ and m_0 such that the following holds. Let G be a bipartite graph with bipartition $V(G) = V_1 \cup V_2$ such that $|V_1| = |V_2| = m \geq m_0$, and let the pair (V_1, V_2) be (ε, δ, G)-super-regular. Then for every pair of vertices $v_1 \in V_1, v_2 \in V_2$, G contains a Hamiltonian path connecting v_1 and v_2.*

Lemma 9.15 *Every graph G of average degree at least $8k$ has a connected k-half dense matching.*

Now we demonstrate the proof:

Proof:

- Step 1: By applying the Regularity Lemma we construct the Reduced Graph G^R, by the same argument as in the proof of Theorem 8.29, we take the color class with most edges, say red, and denote it as G_1^R. It satisfies the requirements of Lemma 9.15, hence we find the large, red, half-dense, connected matching M in G_1^R. Some preparations need to be done on M. First we find the connecting paths among the edges of M within G_1, then remove some vertices to achieve super-regularity between the edges of M, this is guaranteed by Lemma 9.6. By Lemma 9.14, we could have a red cycle spanning the remaining all vertices of M.

- Step 2: Greedily remove cycles until the leftover is small enough. This is almost identical to the proof of Theorem 9.1.

- Step 3: Cover the leftover with the help of vertices from M, this is guaranteed by Theorem 9.8. Notice that we need to make sure that for any cluster only a small portion of it has been used.
- Step 4: We make the matching M balanced again and cover it by one red cycle.

□

Another special case is to cover the vertex set by vertex-disjoint monochromatic trees.

In the classical paper [15], Erdős, Gyárfás and Pyber remarked that the tree cover number is at most r since monochromatic stars at any vertex give a good covering (Note that in a covering, unlike a partition we can reuse the vertices.). And they give an example that shows that the tree cover number is at least $r-1$: Consider a complete graph with vertex set identified with the points of an affine plane of order $r-1$. Color the edge pq with color i $(1 \leq i \leq r)$ if the line through p and q is in the ith parallel class. This example demonstrated that the following conjecture, if true, is best possible.

Conjecture 9.16 (Erdős, Gyárfás, Pyber, 1991 [15]) *The tree partition number is $r-1$.*

Furthermore, they proved the following result for the case $r=3$ in the same paper.

Theorem 9.17 (Erdős, Gyárfás, Pyber, 1991 [15]) *For $r=3$, the tree partition number is 3.*

The latest result for tree partition is due to Haxell and Kohayakawa [39]:

Theorem 9.18 (Haxell, Kohayakawa, 1996 [39]) *Let $r \geq 1$ and $n \geq 3r^4 r!(1-\frac{1}{r})^{3(1-r)} \log r$ be integers, and suppose the edges of K^n are colored with r colors. Then K^n contains $t \leq r$ monochromatic trees of radius at most 2, each of a different color, such that their vertex sets $V(T_i)(1 \leq i \leq t)$ partition the vertex set of K^n.*

Sárközy and Selkow generalized the problem for k-regular graphs and in [69] proved the following.

Theorem 9.19 (Sárközy, Selkow, 2000 [69]) *There exists a constant c such that $f(r,k) \leq r^{c(r \log r + k)}$, i.e. for any $r, k \geq 2$ and for any coloring of the edges of a complete graph with r colors, its vertices can be partitioned into at most $r^{c(r \log r + k)}$ connected monochromatic k-regular subgraphs and vertices.*

One of the main results of this thesis is an improvement on Theorem 9.19. The new result will be presented in Chapter 6.

9.4.3 2-colorings and 3-colorings

A special case of Conjecture 9.11 is when r is equal to a constant. The case $r = 2$ was asked earlier by Lehel and for $n \geq n_0$ was first proved by Łuczak, Rödl and Szemerédi [54]:

Theorem 9.20 (Łuczak, Rödl, Szemerédi, 1998 [54]) *There exists n_0 such that, for every $n \geq n_0$, and every 2-coloring of the edges of K_n, there* *exists a partition of the vertices of K_n into two monochromatic cycles of different colors.*

However, again the Regularity Lemma [75] was used in the proof, which means it applies only to large n. Later Allen [1] offered a proof without the Regularity Lemma and recently Bessy and Thomassé [7] found an elementary argument that works for every n.

For $r = 3$, the current best known result is given by Gyárfás, Ruszinkó, Sárközy and Szemerédi [34]:

Theorem 9.21 (Gyárfás, Ruszinkó, Sárközy, Szemerédi, 2011 [34]) *In every 3-coloring of the edges of K_n the vertices can be partitioned into at most 17 monochromatic cycles.*

They first proved:

Theorem 9.22 (Gyárfás, Ruszinkó, Sárközy, Szemerédi, 2011 [34]) *In every 3-coloring of the edges of K_n all but $o(n)$ of its vertices can be partitioned into three monochromatic cycles.*

Then they use Theorem 9.22 to prove the main theorem (Theorem 9.21). This proof methodology provided a possible way to get a linear bound for general r. As they stated: “in the same way for a general r if one could prove the corresponding asymptotic result as in Theorem 9.22 (even with a weaker linear bound on the number of cycles needed; unfortunately we are not there yet), then we would obtain a linear bound overall.”

9.4.4 Non-complete graphs

To generalize the vertex partition problem, we can cover other graphs instead of a complete graph. Here we discuss two cases: a bipartite complete graph and a graph with independence number $\alpha(G) = \alpha$ (see Section 2.1).

In [37] Haxell generalized Conjecture 9.11 to complete bipartite graphs and she gives the following upper bound:

Theorem 9.23 (Haxell, 1997 [37]) *Let a positive integer r be given. Let ε be such that*

$$\frac{1}{16r} < \varepsilon < \frac{1}{7r}(1 - \frac{1}{r^3})(\frac{4}{5} - \frac{1}{r^2}),$$

and let $s \geq 10$ be such that $f_s(\varepsilon) > 0$. Then for every positive integer n and for every coloring of the edges of $K(n,n)$ with r colors, there exists a set of at most $2r(s+3)\log r + 3r^2$ vertex-disjoint monochromatic cycles whose vertex sets partition the vertex set of $K(n,n)$. Here $f_s(\varepsilon)$ stands for

$$f_s(\varepsilon) = \frac{1}{1-\varepsilon} - (1-\varepsilon)^{1-\frac{1}{s}} - 2\varepsilon^{1-\frac{1}{s}}.$$

For graphs with independence number $\alpha(G) = \alpha$, Sárközy conjectured that $f(\alpha, r) = \alpha r$ and proved the following theorem:

Theorem 9.24 (Sárközy, 2011 [65]) *If the edges of a graph G with $\alpha(G) = \alpha$ are colored with r colors then the vertex set of G can be partitioned into at most $25(\alpha r)^2 \log(\alpha r)$ vertex disjoint monochromatic cycles.*

We may combine the two types of generalizations: we can cover non-complete graphs by using structures other than cycles/paths. One example is to ask for a graph with independence number $\alpha(G) = \alpha$ and its edges are colored with r colors, how many vertex disjoint connected monochromatic k-regular subgraphs and vertices are needed to cover its vertices. In the next chapter we present a new result in this direction.

Chapter 10

Vertex partitions of non-complete graphs by connected monochromatic k-regular graphs

The material of this chapter is from [70].

Let $p(\alpha, r, k)$ denote the minimum number of connected monochromatic k-regular subgraphs needed to partition the vertex set of any r-colored graph G with $\alpha(G) = \alpha$.

Theorem 10.1 (Sárközy, Selkow, Song, 2011 [70]) *There exists a constant c such that for a graph with independence number $\alpha(G) = \alpha$ and its edges colored with r colors, its vertices can be partitioned into at most $(\alpha r)^{c(\alpha r \log(\alpha r)+k)}$ vertex disjoint connected monochromatic k-regular sub-*

graphs and vertices.

In the other direction we have the following bound.

Claim 10.2

$$p(\alpha, r, k) \geq \alpha((r-1)(k-1)+1).$$

Indeed, to see this let us take α cliques of roughly equal size and r-edge coloring inside each clique which requires at least $(r-1)(k-1)+1$ vertex disjoint connected monochromatic k-regular subgraphs and vertices to cover. This can be obtained in the following way. Let S_1 be a set of size $k-1$ and let all edges incident to a vertex of S_1 be colored with color 1. Let S_2 be a set of size $k-1$ disjoint from S_1 and let all edges incident to a vertex of S_2 (that are not colored yet) be colored with color 2. We continue in this fashion; finally S_{r-1} is a set of size $k-1$ disjoint from $\cup_{i=1}^{r-2} S_i$ and all edges incident to a vertex of S_{r-1} (that are not colored yet) are colored with color $r-1$. All remaining edges are colored with color r. Then in this construction we cannot have a non-trivial connected monochromatic k-regular subgraph in color i, $1 \leq i \leq r-1$. Indeed, we cannot have a vertex from outside of S_i (since the degree of any vertex outside in color i is less than k), but we have only $k-1$ vertices inside S_i. Thus all vertices in $\cup_{i=1}^{r-1} S_i$ must be single vertices in the partition, giving the claimed lower bound.

The rest of this chapter is devoted to the proof of Theorem 10.1.

10.1 Sketch of the proof

We follow a similar absorbing technique as before :

- Step 1: Greedily find and remove a series of monochromatic super-regular pairs until the number of leftover vertices is small enough; all but the first pair will be covered by a spanning connected monochromatic k-regular subgraph. The first pair (denoted by (A_1, B_1)) will be combined in Step 2 with some of the leftover vertices to form monochromatic k-regular subgraphs.

- Step 2: Divide the leftover vertices Y into three sets $Y = Y' \cup Y'' \cup Y'''$. We will use a bipartite lemma (Lemma 10.7) to cover the vertices of Y' and some vertices of A_1 and to cover the vertices of Y'' and some vertices of B_1 by vertex disjoint connected monochromatic k-regular subgraphs. After balancing the sizes of the two color classes in the remainder of A_1 and B_1, we will find a spanning connected monochromatic k-regular subgraph in the remainder of (A_1, B_1).

- Step 3: In Y''' we will have $\alpha(G|_{Y'''}) \leq \alpha - 1$, so we can use induction on α to partition the vertices in Y''' into vertex disjoint connected monochromatic k-regular subgraphs.

10.2 Tools

Our first tool will be a lemma of Komlós ([45], see also [37]) claiming that whenever a graph is sufficiently dense, it contains a super-regular pair. The size of this super-regular pair depends on the density.

Lemma 10.3 *There exists a constant* ε_0 *such that if* $\varepsilon \leq \varepsilon_0$*,* $t = (3/\varepsilon)\log(1/\varepsilon)$ *and* G_n *is a graph with* n *vertices and* cn^2 *edges, then* G_n *contains an* (ε, δ)

super-regular subgraph (A_1, B_1) *with*

$$|A_1| = |B_1| = m \geq (2c)^t \lfloor \frac{n}{2} \rfloor \ \text{ and } \ \delta \geq c.$$

We will also use the following lemma from [69] (Lemma 6 in [69]). Note that this lemma is a very special case of the Blow-up Lemma [46]. It says that we can always find a spanning connected k-regular subgraph inside a super-regular pair.

Lemma 10.4 *Given an* $\varepsilon > 0$ *and an integer* $k \geq 2$*, if* (A, B) *is an* (ε, δ) *super-regular pair with* $|A| = |B| = m \geq \frac{k}{\varepsilon^2}$ *and* $\delta > 9\varepsilon$*, then* (A, B) *contains a connected* k*-regular spanning subgraph.*

We will also need a simple consequence of the complementary form of Turán's theorem.

Lemma 10.5 *In a graph* G *on* n *vertices we have*

$$e(G) \geq \frac{n}{2}\left(\frac{n}{\alpha(G)} - 1\right).$$

Proof: Indeed, Turán's theorem applied to the complement of G yields the fact (see e.g. inequality (10.1) on page 150 in [60]) that

$$\alpha(G) \geq \frac{n^2}{2e(G) + n}.$$

From this we get

$$e(G) \geq \frac{n^2}{2\alpha(G) + \frac{n\alpha(G)}{e(G)}} = \frac{n}{2}\left(\frac{2ne(G)}{\alpha(G)(2e(G) + n)}\right) =$$

$$= \frac{n}{2}\left(\frac{n}{\alpha(G)} - \frac{n^2}{\alpha(G)(2e(G)+n)}\right) \geq \frac{n}{2}\left(\frac{n}{\alpha(G)} - 1\right),$$

as desired. □

Finally we will need the following lemma of Pósa ([58], see also Exercise 8.3 in [51]).

Lemma 10.6 *The vertices of a graph G can be covered by not more than $\alpha(G)$ vertex disjoint cycles, edges and vertices.*

10.3 Proof of Theorem 10.1

10.3.1 Step 1

Let G be a graph on n vertices with $\alpha(G) = \alpha$. Let H_i be the subgraph of G with all edges of color i. Let i_1 be a color for which $e(H_{i_1}) \geq e(G)/r$. Using this and Lemma 10.5, for the number of edges of H_{i_1} we get the following.

$$e(H_{i_1}) \geq e(G)/r \geq \frac{n}{2r}(\frac{n}{\alpha} - 1) \geq \frac{n^2}{4\alpha r}.$$

Let ε_0 be as in Lemma 10.3 and $\varepsilon = \frac{\varepsilon_0}{50\alpha r}$. Applying Lemma 10.3 to H_{i_1} there is a $\delta_1 \geq \frac{1}{4\alpha r}$ and a pair (A_1, B_1) in color i_1 such that

- $|A_1| = |B_1| = m_1 \geq \left(\frac{1}{4\alpha r}\right)^t n$ where $t = \left(\frac{3}{\varepsilon}\right) \log\left(\frac{1}{\varepsilon}\right)$, and
- (A_1, B_1) is (ε, δ_1) super-regular.

Let us remove the vertices in the pair (A_1, B_1) and denote the result by G_1. With a similar procedure we find a super-regular pair (A_2, B_2) in color i_2 (possibly different from i_1). Removing (A_2, B_2) and continuing in this fashion, after p steps the number of remaining vertices is at most

$$n\left(1-2\left(\frac{1}{4\alpha r}\right)^t\right)^p. \tag{10.1}$$

Defining

$$x = 2(\alpha r)^2(2e\alpha r)^{\lceil \frac{k}{2}\rceil} \text{ and } x' = \max\left(\frac{m_1}{x^2}, \frac{(4\alpha r)^t k}{\varepsilon^2}\right), \tag{10.2}$$

we stop with the procedure when no more than x' vertices remain. Denote the last chosen super-regular pair by $(A_{p'}, B_{p'})$. Note that we may apply Lemma 10.4 for a pair $(A_i, B_i), 1 \leq i \leq p'$, since $|A_i| = |B_i| \geq \frac{k}{\varepsilon^2}$ and $\delta_i \geq \frac{1}{4\alpha r} > 9\varepsilon$.

In the case $x' = \frac{(4\alpha r)^t k}{\varepsilon^2}$, we are done; we do not even need Step 2 and Step 3. The remaining vertices are just going to be single vertices in the partition (the fact that their number is small enough is checked in the final computation in (10.6)), and by using Lemma 10.4 in $(A_i, B_i), 1 \leq i \leq p'$, the rest of G is partitioned by p' connected monochromatic k-regular graphs.

In the other case when $x' = \frac{m_1}{x^2}$ holds, we apply Lemma 10.4 only in $(A_i, B_i), 2 \leq i \leq p'$, so G consists of (A_1, B_1), a set of $p' - 1$ connected monochromatic k-regular graphs, plus a set Y of fewer than $\frac{m_1}{x^2}$ vertices and we go to Step 2.

Next let us estimate p'. Let us consider a p' for which

$$n\left(1-\frac{2}{(4\alpha r)^t}\right)^{p'} \leq \frac{m_1}{x^2}.$$

This inequality is certainly true (using the lower bound on m_1) if

$$\left(1 - \frac{2}{(4\alpha r)^t}\right)^{p'} \leq \frac{1}{(4\alpha r)^t x^2},$$

which in turn is true using $1 - x \leq e^{-x}$ if

$$e^{-\frac{2p'}{(4\alpha r)^t}} \leq \frac{1}{(4\alpha r)^t x^2}.$$

Thus it follows from the above and (10.1) that in either case we have

$$p' \leq \lceil \frac{(4\alpha r)^t}{2} \left(2 \log x + t \log (4\alpha r)\right) \rceil. \tag{10.3}$$

10.3.2 Step 2

Divide the remaining vertices Y into three sets $Y = Y' \cup Y'' \cup Y'''$ in the following way. If a vertex $y \in Y$ satisfies

$$deg(y, A_1) < m_1/\alpha \ \text{ and } \ deg(y, B_1) < m_1/\alpha,$$

we put it into Y''', and we will deal with this set later in Step 3 by using induction on α.

Next we consider the vertices $y \in Y$ satisfying

$$deg(y, A_1) \geq m_1/\alpha \ \text{ and } \ deg(y, B_1) \geq m_1/\alpha. \tag{10.4}$$

We may assume that the number of vertices satisfying (10.4) is even by removing a single vertex (a vertex that is going to be a singleton in the final partition). Then we put half of these vertices into Y' and the other half into Y''.

Then the vertices $y \in Y$ satisfying

$$deg(y, A_1) \geq m_1/\alpha \text{ and } deg(y, B_1) < m_1/\alpha$$

are also put into Y', and the vertices $y \in Y$ satisfying

$$deg(y, A_1) < m_1/\alpha \text{ and } deg(y, B_1) \geq m_1/\alpha \tag{10.5}$$

are put into Y''.

Without loss of generality, assume that $|Y'| \leq |Y''|$. Take $|Y''| - |Y'|$ vertices from Y'' satisfying (10.5) and put them into Y''' (note that there must be $|Y''| - |Y'|$ such vertices). Thus now $|Y'| = |Y''|$, for every $y \in Y'$ we have $deg(y, A_1) \geq m_1/\alpha$, for every $y \in Y''$ we have $deg(y, B_1) \geq m_1/\alpha$ and finally for every $y \in Y'''$ we have $deg(y, A_1) < m_1/\alpha$.

Then the following lemma will help to cover the vertices in Y' and some vertices in A_1 and the vertices in Y'' and some vertices in B_1. We will apply the lemma twice: once with the choices $S = A_1$ and $Y = Y'$, then again with the choices $S = B_1$ and $Y = Y''$.

Lemma 10.7 *If the edges of a bipartite graph (S, Y) are colored with r colors, $|S| = m$, $|Y| < \frac{m}{x^2}$ (where x is given by (10.2)), and for every $y \in Y$ we have $deg(y, S) \geq m/\alpha$, then the vertices of Y can be covered by at most $rx(1 + \lceil \frac{k}{2} \rceil) + 2\alpha r^2 \lceil \frac{k}{2} \rceil$ vertex disjoint connected monochromatic k-regular graphs and vertices.*

Proof: For each $y \in Y$ and $1 \leq i \leq r$, we define

$$N_i(y) = \{s \in S \ : \ (s, y) \text{ has color } i\},$$

and for $Y' \subset Y$ we define $N_i(Y') = \cap_{y \in Y'} N_i(y)$. Clearly Y can be partitioned into classes $Y_1, Y_2, \ldots, Y_r$ such that $|N_i(y)| \geq \frac{m}{\alpha r}$ for each $y \in Y_i$. In the proof of Lemma 10.7 we will need two claims.

Claim 10.8 *For each Y_i, there is an a_i such that Y_i can be partitioned into classes $Y_{i0}, Y_{i1}, \ldots, Y_{ia_i}$ where*

- $|Y_{i0}| < 2\alpha r \lceil \frac{k}{2} \rceil$,
- $|Y_{ij}| = \lceil \frac{k}{2} \rceil$ *for* $1 \leq j \leq a_i$, *and*
- $|N_i(Y_{ij})| \geq \frac{\alpha r m}{x}$ *for* $1 \leq j \leq a_i$.

Proof: If $|Y_i| < 2\alpha r \lceil \frac{k}{2} \rceil$, the proof is trivial. Let H_i be the subgraph of (S, Y_i) with all edges of color i. If $|Y_i| \geq 2\alpha r \lceil \frac{k}{2} \rceil$, then we have

$$\sum_{\substack{s \in S \\ deg_{H_i}(s) \geq \lceil \frac{k}{2} \rceil}} deg_{H_i}(s) \geq \frac{m}{\alpha r}|Y_i| - \lceil \frac{k}{2} \rceil m \geq \frac{m}{2\alpha r}|Y_i|.$$

We are going to count with multiplicity the number of subsets of Y_i of size $\lceil \frac{k}{2} \rceil$ with a common neighbor $s \in S$ (meaning that if a particular subset has l common neighbors in S, then it is counted l times). Using Jensen's inequality,

$$\sum_{\substack{s \in S \\ deg_{H_i}(s) \geq \lceil \frac{k}{2} \rceil}} \binom{deg_{H_i}(s)}{\lceil \frac{k}{2} \rceil} \geq \frac{m}{2\alpha r} \binom{\frac{|Y_i|}{2\alpha r}}{\lceil \frac{k}{2} \rceil} \geq \frac{m}{2\alpha r} \left(\frac{|Y_i|}{2\alpha r \lceil \frac{k}{2} \rceil} \right)^{\lceil \frac{k}{2} \rceil}.$$

But there are only

$$\binom{|Y_i|}{\lceil \frac{k}{2} \rceil} \leq \left(\frac{e|Y_i|}{\lceil \frac{k}{2} \rceil} \right)^{\lceil \frac{k}{2} \rceil}$$

subsets of Y_i of size $\lceil \frac{k}{2} \rceil$. Thus there must be a $Y_{i1} \subset Y_i$ such that

$$|Y_{i1}| = \lceil \frac{k}{2} \rceil \text{ and } |N_i(Y_{i1})| \geq \frac{m}{2\alpha r} \frac{\left(\frac{|Y_i|}{2\alpha r \lceil \frac{k}{2} \rceil} \right)^{\lceil \frac{k}{2} \rceil}}{\left(\frac{e|Y_i|}{\lceil \frac{k}{2} \rceil} \right)^{\lceil \frac{k}{2} \rceil}} = \frac{m}{2\alpha r (2\alpha e r)^{\lceil \frac{k}{2} \rceil}} = \frac{\alpha r m}{x}.$$

Replacing Y_i by $Y_i \backslash Y_{i1}$ we repeat the procedure until for the leftover we have $|Y_{i0}| < 2\alpha r \lceil \frac{k}{2} \rceil$. We denote the number of repetitions by a_i. This completes the proof of Claim 10.8. □

For each Y_i we define an auxiliary graph G_i with vertices $\{Y_{i1}, Y_{i2}, \ldots, Y_{ia_i}\}$ and edges

$$\left\{ (Y_{ij}, Y_{il}) \ : \ |N_i(Y_{ij}) \cap N_i(Y_{il})| \geq \frac{m}{x^2} > |Y| \right\}.$$

The second claim we need in the proof of Lemma 10.7 is the following.

Claim 10.9 *The size of a maximum independent set of G_i is less than x.*

Proof: Assume indirectly that $\{w_1, w_2, \ldots, w_x\} \subset \{Y_{i1}, Y_{i2}, \ldots, Y_{ia_i}\}$ is an independent set of vertices of G_i. If $w_j = Y_{ij}$, then we define $N_i(w_j) =$

$N_i(Y_{ij})$. Hence we have $|N_i(w_j)| \geq \frac{\alpha rm}{x}$ for $1 \leq j \leq x$. But then

$$m \geq |\cup_{1\leq j\leq x} N_i(w_j)| \geq \alpha rm - \sum_{1\leq j<l\leq x} |N_i(w_j) \cap N_i(w_l)| \geq$$

$$\geq \alpha rm - \frac{x^2}{2}\frac{m}{x^2} = (\alpha r - \frac{1}{2})m > m.$$

By contradiction, G_i can not have an independent set of x vertices, finishing the proof of Claim 10.9. □

Now we are ready to prove Lemma 10.7. By Claim 10.9 and Lemma 10.6, the vertices of G_i can be partitioned into at most x cycles (and edges and vertices), and thus the vertices of $\cup_{1\leq i\leq r} G_i$ can be partitioned into at most rx cycles (and edges and vertices). The single vertices in this partition will correspond to single vertices ($\lceil \frac{k}{2} \rceil$ vertices of Y for each) in the final partition. Between every adjacent pair of vertices on these cycles, we insert disjoint sets of S. Between adjacent vertices Y_{ij} and Y_{il}, we insert $S_{ij} \subset S$ such that $|S_{ij}| = \lceil \frac{k}{2} \rceil$ and $S_{ij} \times (Y_{ij} \cup Y_{il})$ is monochromatic in color i. Inserting these sets (from S) between the corresponding pairs of sets (from Y) on a cycle yields a new, blown-up "cycle", $Z_1, Z_2, \ldots, Z_{2p}$ of sets of vertices of size $\lceil \frac{k}{2} \rceil$, where we have complete bipartite graphs between adjacent sets. The graph with vertices $\cup_{1\leq j\leq 2p} Z_j$ and edges $\cup_{1\leq j<2p}(Z_j \times Z_{j+1}) \cup (Z_1 \times Z_{2p})$ is a connected monochromatic $k + (k \ mod\ 2)$-regular subgraph of G. For odd k, removing a perfect matching in each of $Z_{2j+1} \times Z_{2j+2}$ for $0 \leq j < p$ yields a connected monochromatic k-regular graph. Hence the vertices of $S \times Y$ can be partitioned into at most rx connected monochromatic k-regular graphs plus at most $rx\lceil \frac{k}{2} \rceil + 2\alpha r^2 \lceil \frac{k}{2} \rceil$ single vertices resulting from the single vertices in the cover of G_i and the vertices in Y_{i0}. This finishes the proof of Lemma 10.7. □

Applying Lemma 10.7 for $S = A_1$ and Y', we obtain a set of at most $rx(1+\lceil\frac{k}{2}\rceil)+2\alpha r^2\lceil\frac{k}{2}\rceil$ connected monochromatic k-regular graphs and vertices that partition the vertices in Y' and a subset A' of A_1. Similarly we have a set of at most $rx(1+\lceil\frac{k}{2}\rceil)+2\alpha r^2\lceil\frac{k}{2}\rceil$ connected monochromatic k-regular graphs and vertices that partition the vertices in Y'' and a subset B' of B_1. Assuming $|A'|<|B'|$, we add $|B'|-|A'|$ additional single vertices from A_1 to A'; thus now $|A_1\setminus A'| = |B_1\setminus B'|$. Finally we apply Lemma 10.4 for $H_{i_1}|_{(A_1\setminus A')\cup(B_1\setminus B')}$. It is not hard to check that the conditions of Lemma 10.4 are still satisfied.

Thus, using (10.2) and (10.3), we get the conclusion that the number of vertex disjoint monochromatic k-regular graphs and vertices needed to cover G except vertices in Y''' is at most

$$p'+3\left(rx(1+\lceil\frac{k}{2}\rceil)+2\alpha r^2\lceil\frac{k}{2}\rceil\right)+\frac{(4\alpha r)^t k}{\varepsilon^2}+1\le(\alpha r)^{c(\alpha r\log(\alpha r)+k)} \qquad (10.6)$$

with some constant c. Indeed, here the p' comes from the super-regular pairs, in the factor 3, one is for the application of Lemma 10.7 for (A_1, Y'), one is for the application of Lemma 10.7 for (B_1, Y'') and one is for the balancing of the remainder of (A_1, B_1) with single vertices. The $\frac{(4\alpha r)^t k}{\varepsilon^2}$ term is for the remaining single vertices when we had the case $x' = \frac{(4\alpha r)^t k}{\varepsilon^2}$ in Step 1 and finally the plus 1 is the potential single vertex needed to make $|Y'\cup Y''|$ even.

10.3.3 Step 3

In the graph $G|_{Y'''}$ we claim that $\alpha(G|_{Y'''}) \leq \alpha - 1$.

Indeed, otherwise let us take an independent set $\{y_1, y_2, \ldots, y_\alpha\}$ in $G|_{Y'''}$. By the definition of Y''', we have

$$deg(y_j, A_1) < m_1/\alpha \text{ for every } 1 \leq j \leq \alpha.$$

But then we can choose a vertex $a \in A_1$ that is not adjacent to any of the vertices $y_j, 1 \leq j \leq \alpha$, giving an independent set of size $\alpha + 1$ in G, a contradiction.

But then, we can iterate our whole procedure with $\alpha - 1$ inside $G|_{Y'''}$. Hence for $p(\alpha, r, k)$, the minimum number of connected monochromatic k-regular subgraphs needed to partition the vertex set of any r-colored graph G with $\alpha(G) = \alpha$, we get the following bound.

$$p(\alpha, r, k) \leq (\alpha r)^{c(\alpha r \log(\alpha r) + k)} + p(\alpha - 1, r, k).$$

Repeating this for all $1 < j < \alpha$ and finally using the bound $p(1, r, k) \leq r^{c(r \log r + k)}$ from [69], we get the bound

$$p(\alpha, r, k) \leq (\alpha r)^{c(\alpha r \log(\alpha r) + k)} + ((\alpha - 1) r)^{c((\alpha - 1) r \log((\alpha - 1) r) + k)} + p(\alpha - 2, r, k) \leq \ldots \leq$$

$$\leq \alpha(\alpha r)^{c(\alpha r \log(\alpha r) + k)} \leq (\alpha r)^{(c+1)(\alpha r \log(\alpha r) + k)},$$

and the proof is finished. □

Chapter 11

Vertex partitions by connected monochromatic k-regular graphs

This chapter presents the results from [71].

Theorem 11.1 (Sárközy, Selkow, Song, 2013 [71]) *For every integer $r \geq 2$ and $k \geq 2$ there exists a constant $n_0 = n_0(r,k)$ such that if $n \geq n_0$ and the edges of the complete graph K_n are colored with r colors then the vertex set of K_n can be partitioned into at most $f(r,k)$ connected monochromatic k-regular subgraphs and vertices such that*

$$f(r,k) \leq cr\log r + r(k-1).$$

We note that this is not far from being best possible (especially if r is small compared to k), as we have the following lower bound.

Theorem 11.2

$$f(r,k) \geq (r-1)(k-1)+1.$$

One of our tools in the proof of Theorem 11.1 is Theorem 8.29, a Ramsey-type result for the existence of connected monochromatic k-regular subgraphs that may be of independent interest. For the completeness we restate Theorem 8.29 here.

Theorem 11.3 *For every positive ε and integers $r, k \geq 2$ there exists a constant $n_0 = n_0(\varepsilon, r, k)$ such that for any r-coloring of the edges of a complete graph on $n \geq n_0$ vertices, we can find a connected monochromatic k-regular subgraph spanning at least $(1-\varepsilon)n/r$ vertices.*

Thus perhaps surprisingly we can guarantee a connected monochromatic k-regular subgraph almost as large as the largest monochromatic cycle we can guarantee.

The next three sections of this chapter are devoted to the proof of Theorem 11.1.

11.1 Sketch of the proof

To prove Theorem 11.1 we apply the edge-colored version of the Regularity Lemma to an r-colored K_n. Again we introduce the reduced graph G^R, the graph whose vertices are associated with the clusters and whose edges are associated with dense ε-regular pairs. The edges of the reduced graph will be colored with a color that appears most often on the edges between the two clusters. Then we study large monochromatic connected matchings in the reduced graph. That was initiated in [52] and for example it played an important role in [32] where the three-color Ramsey numbers of paths for large n have been determined (see Theorem 8.17).

We follow the absorbing proof technique as before. We establish the bound on $f(r,k)$ in the following steps.

- Step 1: We find a sufficiently large monochromatic (say red), dense (more precisely half-dense as defined earlier in Definition 9.13), connected matching M in G^R.
- Step 2: We remove the vertices of M from G^R and greedily remove a number (depending on r) of vertex disjoint connected monochromatic k-regular subgraphs from the remainder in K_n until the number of leftover vertices is much smaller than the number of vertices associated with M. For this purpose we will use the Ramsey-type result (Theorem 8.29) for the existence of connected monochromatic k-regular subgraphs.
- Step 3: Using a lemma about k-regular subgraph covers of r-colored unbalanced complete bipartite graphs we combine the leftover vertices with some vertices of the clusters associated with vertices of M. (M absorbs the leftover vertices.)
- Step 4: Finally after some adjustments through alternating paths with

respect to M, we find a red k-regular subgraph spanning the remaining vertices of M.

The proof of Theorem 11.1 in Section 11.3 will follow this outline. Since some steps in the proof are straightforward adaptations of the corresponding steps from [33] to k-regular graphs, at some places we will omit the details. First we discuss the necessary tools. Then the easy construction for Theorem 11.2 is given in Section 11.5.

11.2 Tools

As stated in Lemma 9.6, a well-known property of ε-regular pairs is that they contain large super-regular subgraphs.

Lemma 10.4 is a special case of the Blow-up Lemma, [46], [47], claiming that a balanced super-regular pair can be spanned by a k-regular subgraph.

We will also need a lemma of Gyárfás, Ruszinkó, Sárközy and Szemerédi from [33].

Lemma 11.4 (Lemma 5 in [33]) *Let $\vec{G} = \vec{G}(V, E)$ be a directed graph with $|V| = n$ sufficiently large and minimum out-degree $d_+(x) \geq cn$ for some constant $0 < c \leq .001$. Then there are subsets $X, Y \subseteq V$ such that*

- *$|X|, |Y| \geq cn/2$;*
- *From every $x \in X$ there are at least $c^6 n$ internally vertex disjoint paths of length at most c^{-3} to every $y \in Y$ (denoted by $x \hookrightarrow y$).*

11.3 Proof of Theorem 11.1

11.3.1 Step 1

We will assume that n is sufficiently large and that $k \geq 3$. In fact for $k = 2$ Theorem 11.1 follows from the main result of [33] (actually the proof there gives a $98r \log r$ bound). We will use the following main parameters

$$0 < \varepsilon \ll \delta \ll 1, \tag{11.1}$$

where $a \ll b$ means that a is sufficiently small compared to b. In order to present the results transparently we do not compute the actual dependencies, although it could be done.

Consider an r-edge coloring $(G_1, G_2, \ldots, G_r)$ of K_n. Apply the r-color version of the Regularity Lemma (Theorem 3.6), with ε as in (11.1) and get a partition of $V(K_n) = V = \cup_{0 \leq i \leq l} V_i$, where $|V_i| = m, 1 \leq i \leq l$. We define the reduced graph G^R: The vertices of G^R are $p_1, \ldots, p_l$, and we have an edge between vertices p_i and p_j if the pair $\{V_i, V_j\}$ is (ε, G_s)-regular for $s = 1, 2, \ldots, r$. Thus we have a one-to-one correspondence $f : p_i \to V_i$ between the vertices of G^R and the clusters of the partition. Then,

$$|E(G^R)| \geq (1 - \varepsilon)\binom{l}{2},$$

and thus G^R is a $(1 - \varepsilon)$-dense graph on l vertices.

Define an edge-coloring $(G_1^R, G_2^R, \ldots, G_r^R)$ of G^R by r colors in the following way. The edge $p_i p_j$ is colored with a color s that contains the most edges from $K(V_i, V_j)$, thus clearly $e_{G_s}(V_i, V_j) \geq \frac{1}{r}|V_i||V_j|$. Let us take the color class in this coloring of G^R that has the most edges. For simplicity assume that this is G_1^R and call this color red. Clearly, we have

$$\left|E(G_1^R)\right| \geq (1-\varepsilon)\frac{1}{r}\binom{l}{2}, \tag{11.2}$$

and thus using (11.1) the average degree in G_1^R is at least $(1-\varepsilon)(l-1)/r \geq l/2r$. Using Lemma 9.14 we can find a connected $l/16r$-half dense matching M in G_1^R. Say M has size

$$|M| = l_1 \geq \frac{l}{16r}, \tag{11.3}$$

and the matching $M = \{e_1, e_2, \ldots, e_{l_1}\}$ is between the two sets of end points U_1 and U_2, where U_1 contains the strong end points, i.e. the points in U_1 have at least $l/16r$ neighbors in U_2. Furthermore, define $f(e_i) = (V_1^i, V_2^i)$ for $1 \leq i \leq l_1$ where V_1^i is the cluster assigned to the strong end point of e_i, and V_2^i is the cluster assigned to the other end point. Hence we have our large, red, half-dense, connected matching M as desired in Step 1.

However, we need to do some preparations on the matching M. We will need the following lemma (this will be used later again).

Lemma 11.5 *Assume that for some positive constant c we find a monochromatic connected matching M (say in G_1^R) saturating at least $c|V(G^R)|$ vertices of G^R. Then in the original r-edge colored K_n we find a connected monochromatic k-regular subgraph in G_1 covering at least $c(1-3\varepsilon)n$ vertices.*

Proof: Note that for $k = 2$ this lemma is well-known and has been used extensively (e.g. in [33], [32]). Let us use the same notation as above, the matching $M = \{e_1, e_2, \dots, e_{l_1}\}$, $f(e_i) = (V_1^i, V_2^i)$ for $1 \leq i \leq l_1$ and $2l_1 \geq cl$.

First we make the matching edges super-regular by applying Lemma 9.6. Then we find connecting paths between the edges of the matching M. Since M is a connected matching in G_1^R we can find a connecting path P_i^R in G_1^R from $f^{-1}(V_2^i)$ to $f^{-1}(V_1^{i+1})$ for every $1 \leq i \leq l_1$ (for $i = l_1$ we have $i + 1 = 1$). Note that these paths in G_1^R may not be internally vertex disjoint. From these paths P_i^R in G_1^R we can construct l_1 vertex disjoint connecting (almost) k-regular subgraphs H_i in G_1 connecting V_2^i and V_1^{i+1}. More precisely we construct H_1 with the following simple greedy strategy. Denote $P_1^R = (p_1, \dots, p_t), 2 \leq t \leq l$, where according to the definition $f(p_1) = V_2^1$ and $f(p_t) = V_1^2$. First let us take a set C^1 of $2k$ "typical" vertices in $f(p_1) = V_2^1$, more precisely we have $|C^1| = 2k$ and $N_{G_1}(C^1, f(p_2)) \geq (1/r - \varepsilon)^{2k} m$. By (ε, G_1)-regularity most of the vertices in V_2^1 satisfy this. We halve C^1 arbitrarily: $C^1 = C_1^1 \cup C_2^1$, $|C_1^1| = |C_2^1| = k$. Next we take a set C^2 of $2k$ typical vertices in $N_{G_1}(C^1, f(p_2))$, more precisely we have $|C^2| = 2k$ and $N_{G_1}(C^2, f(p_3)) \geq (1/r - \varepsilon)^{2k} m$. By (ε, G_1)-regularity most of the vertices satisfy this in $N_{G_1}(C^1, f(p_2))$. Note that between C^1 and C^2 we have a complete bipartite graph $K(2k, 2k)$. Again halve C^2 arbitrarily: $C^2 = C_1^2 \cup C_2^2$, $|C_1^2| = |C_2^2| = k$. We continue in this fashion. Finally for the last C^t we take $2k$ typical vertices in $N_{G_1}(C^{t-1}, f(p_t))$.

To define the connecting subgraph H_1, we do the following. First from each $K(2k, 2k)$ between C^i and C^{i+1}, $1 \leq i \leq t - 1$ we take a $\lfloor k/2 \rfloor$-regular subgraph (clearly this can be done). Then if k is odd, we add perfect matchings between C_1^i and C_2^{i+1}, $1 \leq i \leq t - 1$. Then for the resulting connecting subgraph H_1, all interior vertices (vertices in $\cup_{i=2}^{t-1}C^i$) have degree k, the degrees in C_1^1 and C_2^t are $\lceil k/2 \rceil$ and the degrees in C_2^1 and C_1^t are $\lfloor k/2 \rfloor$.

Then we move on to the next connecting subgraph H_2. We follow the same greedy procedure, we always take the next subset from the next cluster in P_2^R. However, if the cluster has occurred already on the path P_1^R, then we just have to make sure that we pick vertices that have not been used yet on H_1.

We continue in this fashion and construct the vertex disjoint connecting subgraphs H_i in G_1, $1 \leq i \leq l_1$. Note that for $k = 3$ these connecting subgraphs may not be connected. However, the final k-regular subgraph will be connected. These will be parts of the final connected k-regular subgraph in G_1. We remove the internal vertices of these subgraphs from G_1. At this point we might have some discrepancies in the cardinalities of the clusters of a matching edge. We remove some more vertices from some clusters V_j^i of the matching to assure that now we have the same number of vertices left in both clusters of a matching edge. For simplicity we still keep the notation $f(e_i) = (V_1^i, V_2^i)$ for the modified clusters. Note that from each cluster V_j^i we removed altogether at most $2\varepsilon m$ vertices.

Finally by applying Lemma 10.4 we close the connected k-regular subgraph in G_1 within each super-regular matching edge in such a way that we span all the remaining vertices in (V_1^i, V_2^i). Indeed, let us take a balanced super-regular matching edge. In both clusters in this subgraph there must be k vertices with degree $\lfloor k/2 \rfloor$, k vertices with degree $\lceil k/2 \rceil$ and all other vertices must have degree k (so here these are the missing degrees in the

k-regular subgraph we are constructing). First remove the vertices with degree $\lfloor k/2 \rfloor$, and by applying Lemma 10.4 with $\lceil k/2 \rceil$ (note that $\lceil k/2 \rceil \geq 2$) we find a connected $\lceil k/2 \rceil$-regular subgraph in the remainder. Remove the edges of this subgraph and those vertices that only need degree $\lceil k/2 \rceil$ and add back the vertices with degree $\lfloor k/2 \rfloor$. By applying Lemma 10.4 again with $\lfloor k/2 \rfloor$ we find a connected $\lfloor k/2 \rfloor$-regular subgraph in the resulting pair (if $k = 3$ we just find a perfect matching) in such a way that we are not using any edges from the bipartite graph between the two sets of vertices with degree $\lfloor k/2 \rfloor$ (since these sets have a constant size this is not a significant restriction). Then from the construction it follows that the resulting subgraph is a connected k-regular subgraph. □

Returning to Step 1, for our matching $M = \{e_1, e_2, \ldots, e_{l_1}\}$ satisfying (11.3) we follow the same procedure as in Lemma 11.5 (so in Lemma 11.5 we have $c = 1/8r$). However, for technical reasons we postpone the last step, the closing of the k-regular subgraph within each (V_1^i, V_2^i), until the end of Step 4, since in Step 3 we will use some of the vertices in $f(M)$, and we will have to make some adjustments first in Step 4.

11.3.2 Step 2

We go back from the reduced graph to the original graph and we remove the vertices assigned to the matching M, i.e. $f(M)$. We apply repeatedly Theorem 11.3 to the r-colored complete graph induced by $K_n \setminus f(M)$. This way we choose t vertex disjoint connected monochromatic k-regular subgraphs in $K_n \setminus f(M)$. Define the constant $c = 1/500r$ (thus note $c \leq 0.001$ what is needed in Lemma 11.4). We wish to choose t such that the remaining set

B of vertices in $K_n \setminus f(M)$ not covered by these t cycles has cardinality at most $c^{11}n$. Since after t steps at most

$$(n - |f(M)|)\left(1 - \frac{1-\varepsilon}{r}\right)^t$$

vertices are left uncovered, we have to choose t to satisfy

$$(n - |f(M)|)\left(1 - \frac{1-\varepsilon}{r}\right)^t \leq c^{11}n.$$

This inequality is certainly true if

$$\left(1 - \frac{1-\varepsilon}{r}\right)^t \leq c^{11},$$

which in turn is true using $1 - x \leq e^{-x}$ if

$$e^{-\frac{(1-\varepsilon)t}{r}} \leq c^{11}.$$

This shows that we can choose $t = 12r\lceil \log 500r \rceil$ (assuming that ε is small enough).

We may assume that the number of remaining vertices in B is even by removing one more vertex (a degenerate cycle) if necessary.

11.3.3 Step 3

This step is similar to the corresponding step in [33]. The key to this step is the following lemma about r-colored complete unbalanced bipartite graphs.

Lemma 11.6 *There exists a constant n_0 such that the following is true.*

Assume that the edges of the complete bipartite graph $K(A,B)$ are colored with r colors. If $|A| \geq n_0$, $|B| \leq |A|/2r$, then B can be covered by at most $(k+1)r$ vertex disjoint connected monochromatic k-regular subgraphs.

The proof of this lemma is postponed until Section 11.4. We have the connected, red matching M of size l_1 between U_1 and U_2. Define the auxiliary directed graph $\vec{G}$ on the vertex set U_1 as follows. We have the directed edge from V_1^i to V_1^j, $1 \leq i,j \leq l_1$ if and only if $(V_1^i, V_2^j) \in G_1^R$. The fact that M is $l/16r$-half dense implies that in $\vec{G}$ for the minimum outdegree we have

$$\min_{x \in U_1} d_+(x) \geq \frac{l}{16r} \geq \frac{|U_1|}{16r} \left(\geq \frac{|U_1|}{500r} \right).$$

Thus applying Lemma 11.4 for $\vec{G}$ with $c = \frac{1}{500r}(< .001)$, there are subsets $X_1, Y_1 \subset U_1$ such that

- $|X_1|, |Y_1| \geq c|U_1|/2$;
- From every $x \in X_1$ there are at least $c^6|U_1|$ internally vertex disjoint paths of length at most c^{-3} to every $y \in Y_1$ $(x \hookrightarrow y)$.

Let X_2, Y_2 denote the set of the other endpoints of the edges of M incident to X_1, X_2, respectively. Note that a path in $\vec{G}$ corresponds to an alternating path with respect to M in G_1^R.

In each cluster $V_1^i \in Y_1$ let us consider an arbitrary subset of $c^8|V_1^i|$ vertices. Denote by A_1 the union of all of these subsets. Similarly we denote by A_2 the union of arbitrary subsets of $V_2^j \in X_2$ of size $c^8|V_2^j|$. Then we have

$$|A_1|, |A_2| \geq c^8 |f(Y_1)| \geq c^8 \frac{c}{2} |f(U_1)| \geq c^8 \frac{c}{2} \frac{n}{16r} \geq c^{10} n.$$

Let us divide the remaining vertices in B (B was defined in Step 2) into two equal sets B_1 and B_2. Thus we have $|B_1|, |B_2| \leq |B| \leq c^{11} n$. We apply Lemma 11.6 in $K(A_1, B_1)$ and in $K(A_2, B_2)$. The conditions of the lemma are satisfied by the above since $|B_i| \leq |A_i|/2r$ for $i = 1, 2$. Let us remove the at most $(k+1)r$ vertex disjoint connected monochromatic k-regular subgraphs covering B_1 in $K(A_1, B_1)$ and the at most $(k+1)r$ k-regular subgraphs covering B_2 in $K(A_2, B_2)$. By doing this we may create discrepancies in the number of remaining vertices in the two clusters of a matching edge. In the next step we have to eliminate these discrepancies with the use of the many alternating paths.

11.3.4 Step 4

Again similar to Step 4 in [33]. By removing the vertex disjoint connected monochromatic k-regular subgraphs covering B_1 in $K(A_1, B_1)$ we have created a "surplus" of $|B_1|$ vertices in the clusters of Y_2 compared to the remaining number of vertices in the corresponding clusters of Y_1. Similarly by removing the k-regular subgraphs covering B_2 in $K(A_2, B_2)$ we have created a "deficit" of $|B_2| (= |B_1|)$ vertices in the clusters of X_2 compared to the number of vertices in the corresponding clusters of X_1. The natural idea is to "move" the surplus from Y_2 through an alternating path to cover the deficit in X_2. The details can be found in [33]. The only difference is the way we extend the connecting subgraphs (see page 864 in [33]); the adaptation for k-regular subgraphs is straightforward. Assume that the surplus $s \leq 2k$ (we move at most $2k$ vertices at a time). Instead of extending the connecting path P_{j-1} by a path of length $2s+2$ as in [33], we have to extend the connecting subgraph H_{j-1} by a 4-partite subgraph. The partite sets (of size $2k$) in this extension come from the following sets (in this order):

$$V_2^j, V_1^j, V_2^j \cup V_2^{j_1}, V_1^j,$$

where we make sure that the third partite set includes exactly s vertices from $V_2^{j_1}$. Otherwise the construction of this extension is the same as in the proof of Lemma 11.5. All other details are as in [33].

After this process the remaining vertices in a matching edge $f(e_i) = (V_1^i, V_2^i)$ will form a balanced super-regular pair where the parameters are somewhat weaker (say $(2\varepsilon, 1/2r, G_1)$-super-regular). Then as we mentioned at the end of Step 1 we can close the k-regular subgraph to span all the remaining vertices of $f(M)$.

Thus the total number of vertex disjoint connected monochromatic k-regular subgraphs we used to partition the vertex set of K_n is at most

$$12r\lceil\log(500r)\rceil + 2(k+1)r + 2 \leq 100r\lceil\log r\rceil + 2kr,$$

finishing the proof of Theorem 11.1. □

11.4 Proof of Lemma 11.6

Again similar to the corresponding Lemma 6 in [33] (so at some places we will omit the details) but we will use a more recent, improved lemma from [35]. Lemma 11.6 clearly follows from the following two lemmas (corresponding to Lemmas 7 and 8 in [33]).

Lemma 11.7 *For every positive ε there exists a constant $n_0 = n_0(\varepsilon)$ such that the following is true. Assume that the edges of the complete bipartite graph $K(A, B)$ are colored with r colors. If $|A| \geq n_0$, $|B| \leq |A|/2r$, then apart from at most $\varepsilon|B|$ vertices B can be covered by at most r vertex disjoint connected monochromatic k-regular subgraphs.*

Lemma 11.8 *There exists a constant n_0 such that the following is true. Assume that the edges of the complete bipartite graph $K(A, B)$ are colored with r colors. If $|A| \geq n_0$, $|B| \leq |A|/(8r)^{8(r+1)}$, then B can be covered by at most kr vertex disjoint connected monochromatic k-regular subgraphs.*

Lemma 11.7 follows easily from Lemma 11.5 and the following lemma from [35].

Lemma 11.9 (Gyárfás, Ruszinkó, Sárközy, Szemerédi, 2006 [35]) *For some $0 < \varepsilon < 1/9$ assume that the edges of a $(1-\varepsilon)$-dense bipartite graph $G(A, B)$ are colored with r colors, $|B| \leq 2|A|/3r$. Then there are vertex disjoint monochromatic connected matchings, each of a different color, such that their union covers at least $(1-\sqrt{\varepsilon})$-fraction of the vertices of B.*

Indeed, we apply the bipartite, colored version of the Regularity Lemma for $K(A,B)$, define the bipartite reduced graph G^R, apply Lemma 11.9 in G^R and then return to $K(A,B)$ by Lemma 11.5. See [35] for the details.

The proof of Lemma 11.8 will use the following simple lemma (corresponding to Lemma 9 in [33]). Note that this is the only place in the proof of our main theorem where the bound depends on k.

Lemma 11.10 *Assume that the edges of the complete bipartite graph $K(A,B)$ are colored with r colors. If $(|B|-1)r^{|B|} < |A|$, then B can be covered by at most $(k-1)r$ vertex disjoint connected monochromatic k-regular subgraphs.*

Proof of Lemma 11.10: Denote the vertices of B by $\{b_1, b_2, \ldots, b_{|B|}\}$. To each vertex $v \in A$ we assign a vector $(v_1, v_2, \ldots, v_{|B|})$ of colors, where v_i is the color of the edge (v, b_i). The total number of distinct color vectors possible is $r^{|B|}$. Since we have $|A| > (|B|-1)r^{|B|}$ vectors, by the pigeon-hole principle we must have a vector that is repeated at least

$$\frac{|A|}{r^{|B|}} \geq |B|$$

times. In other words, there are at least $|B|$ vertices in A for which the colorings of the edges going to $\{b_1, b_2, \ldots, b_{|B|}\}$ are exactly the same. Now if for these vertices in A the number of edges in one color is at least k, then we can clearly cover the other endpoints of these edges in B with one connected k-regular subgraph in this color. However, if the number of edges is less than k for a certain color, then the corresponding endpoints in B will be isolated vertices in our cover. Thus altogether in the worst case we need $(k-1)r$

vertex disjoint connected monochromatic k-regular subgraphs to cover B. □

Proof of Lemma 11.8: This is almost identical to the proof of the corresponding lemma (Lemma 8) in [33]. Therefore we omit the details and highlight only the differences. Of course, one difference is that whenever we have a monochromatic connected matching in the reduced graph we saturate it with a connected k-regular subgraph instead of a cycle by applying Lemma 11.5. The second difference is again in the way we handle the vertices v satisfying (16) in [33] (see the top of page 869 in [33]). The adaptation is again straightforward (similar to the adaptation in Step 3); instead of extending the connecting path P_{i-1} by a path of length 6, we have to extend the connecting subgraph H_{i-1} by a 6-partite subgraph. The partite sets in this extension come from the following sets (in this order):

$$V_B^i, f(p_A^j), v \cup V_B^i, f(p_A^j), V_B^i, V_A^i,$$

where we make sure that the third partite set includes v. Otherwise the construction of this extension is the same as in the proof of Lemma 11.5. The rest of the proof is identical to the proof of Lemma 8 in [33] but of course we finish with Lemma 11.10 resulting in at most $(k-1)r + r = kr$ connected monochromatic k-regular subgraphs in the cover. This finishes the proof of Lemma 11.6. □

11.5 Proof of Theorem 11.2

In this section we present the easy construction for Theorem 11.2. Let $A_1, \ldots, A_{r-1}$ be disjoint vertex sets of size $k-1$, and A_r is the set of remaining vertices (assuming $n > (r-1)(k-1)$). The r-coloring is defined in the following way: color 1 is all the edges containing a vertex from A_1, color 2 is all the edges containing a vertex from A_2 and not in color 1, etc. we continue in this fashion. Color $r-1$ is all the edges containing a vertex from A_{r-1} and not in color $1, \ldots, r-2$. Finally color r is all the edges within A_r.

To show the lower bound let us assume that we have a covering by vertex disjoint connected monochromatic k-regular subgraphs. It is not hard to see that in this covering the vertices in $A_1 \cup \ldots \cup A_{r-1}$ must be isolated vertices. Indeed, to cover any vertex in $A_i, 1 \leq i \leq r-1$ by a non-trivial connected monochromatic k-regular subgraph, the only possible color is color i. However, we have to include at least one vertex from the outside of A_i. But then this vertex must have k neighbors in A_i, a contradiction. The vertices in $A_1 \cup \ldots \cup A_{r-1}$ must be indeed isolated vertices. Counting one more subgraph to cover A_r, altogether we need at least $(r-1)(k-1)+1$ connected monochromatic k-regular subgraphs to cover all the vertices. □

Chapter 12

Future directions

The Regularity Lemma states that there is a regular partition for every dense graph; from this regular partition one can construct a reduced graph of much smaller size which is an essence of the original graph. However, the size requirement of the Regularity Lemma makes it impractical for real world situations where the number of vertices typically is a few thousand only. Our practical regularity partitioning algorithm is a tradeoff between a (almost) perfect representation of the original graph and the requirement of the large graph size. Our strategy is one possible way to make this tradeoff. Based on our work, below we list some possible future extensions.

12.1 Different algorithms

In our work we modify the algorithmic version of the Regularity Lemma due to Alon *et al.* [3] and Frieze and Kannan [21] for constructing a reduced graph. As we note earlier, there is another constructive version of the Regularity Lemma due to Fischer *et al.* [19]. They give a new approach for finding a regular partition which is quite different from the previous approaches. All the previous ones try to find partitions of the tower type, while this paper gives a method to find a smaller regular partition if one exists in the graph. Employing this methodology for refinement instead of using an approximate version of the algorithmic Regularity Lemma could also be a fruitful direction of work.

All the algorithms described above are designed to find the (perfect) regular partition, they are balanced algorithms in which each iteration generates the same amount of information. On the contrary, in practice we might not need the perfect regular partition. This fact could be used to make the practical regularity partitioning algorithm more efficient. Specifically, we believe that a greedy strategy based on local optimization in which the first several iterations give as much information as possible might be useful.

12.2 Refinement strategy

Currently our strategy in the practical regularity partitioning algorithm is to use only one certificate for each class while doing the refinement, there are several possible ways to make an improvement. Theoretically, the more certificates we use, the more information we preserve, so a straightforward way is to use two or more certificates and compare the result with current one.

Another possibility is to use all the certificates just as in the original algorithm, but instead of going till the end we stop after some iterations to make it applicable on small graphs. Here how to define the stopping condition will be the main issue.

In the Regularity Lemma, the clusters found in each iteration are to be of equal size. In practical problems, we might not need this constraint. For example, in clustering it is more natural that different clusters have different sizes. This could lead to another possible modification to the practical regularity partitioning algorithm. Notice in the constructive version due to Alon *et al.* [3] it does require equal sized clusters, but in Frieze and Kannan [21] there is no such constraint and the clusters can be of different size.

12.3 Sparse graph

Our practical regularity partitioning algorithm is only applicable when the graph is dense. However, there are sparse versions of the Regularity Lemma that work with, as the name indicates, sparse graphs. Implementation of the sparse Regularity Lemma for refinement has important meaning in solving practical problems. For example, the spectral clustering pipeline involves two stages: Construction of the pairwise affinity matrix (and hence the graph Laplacian) and eigendecomposition of the output of this stage for dimensionality reduction. It is on this reduced dimension that we run a traditional clustering method such as k-means to obtain the final clustering. As we note earlier, both of these stages require significant computation and have inspired research to get around these bottlenecks. In our current work (the practical regularity partitioning algorithm on dense graph) we give a method to substantially ease the second bottleneck. To make the entire method far more powerful with a very wide range of applicability we need to make changes to the first stage of the bottleneck. Utilizing the sparse Regularity Lemma for refinement in our method could be used to get around the first bottleneck in the framework above as well (this would be possible as it work allow us to work with k-nearest neighbor graphs). And thus together, the regularity clustering method could be made really powerful.

12.4 Extensions to hypergraphs

One of the most attractive notions of pairwise clustering methods is that they give a more "global" view of the data. Given enough number of data-points we could at least approximately get an idea about the geometry of the data, thus significantly improving its performance over traditional methods which are more "local". As pointed out by Fowkles *et al.* [20],when seen through the lens of computer vision this makes such "global" clustering methods (for segmentation) closer to the original views on form and perception that for a human an image is much more than a mere collection of objects. However, while pairwise affinities capture a more global view of the data, it is not necessary that the relationship between data-points in most domains has to be dyadic and thus restricting it to being dyadic might lead to loss of information. Indeed, it might be the case that the relationship between data-points is triadic or even higher. Thus, this natural extension has led to work on clustering methods for such problems, which can be naturally formulated as a hypergraph partitioning problem [2], [82], [9]. There are a number of important results that extend the Regularity Lemma to hypergraphs [62], [63], [29], [10]. It is thus natural that our methodology could be extended to hypergraphs and then used for hypergraph clustering. This seems to be a particularly promising direction.

Bibliography

[1] P. Allen, Covering two-edge-coloured complete graphs with two disjoint monochromatic cycles, *Probability, Combinatorics and Computing*, 17(4), (2008), 471-486.

[2] S. Agarwal, L. Zelnik-Manor J. Lim, P. Perona, D. Kriegman, S. Belongie, Beyond pairwise clustering, In *IEEE Conf. on Computer Vision and Pattern Recognition*, (2005).

[3] N. Alon, R. A. Duke, H. Lefmann, V. Rödl, R. Yuster, The Algorithmic Aspects of the Regularity Lemma, *Journal of Algorithms*, 16, (1994), 80-109.

[4] A. Aho, J. Hopcroft, J. Ullman, The design and analysis of computer algorithms, *Addison-Wesley*, Menlo Park, CA, (1974).

[5] B. Bollobás, A. Gyárfás, Highly connected monochromatic subgraphs, *Discrete Mathematics*, 308, (2008) 1722-1725.

[6] T. N. Bui, C. Jones, Finding good approximate vertex and edge partitions is NP-hard, *Inf. Process. Lett.* 42(3), 153-159.

[7] S. Bessy, S. Thomassé, Partitioning a graph into a cycle and an anticycle: a proof of Lehel's conjecture, *Journal of Combinatorial Theory*, Ser. B 100(2), (2010), 176-180.

[8] S. A. Burr, J. A. Roberts, On Ramsey Numbers for Stars, *Utilitas Mathematica*, 4, (1973), 217-220.

[9] S. Bulo and M. Pelillo, A game-theoretic approach to hypergraph clustering, In *Advances in Neural Information Processing Systems*, (2009).

[10] F. Chung, Regularity lemmas for hypergraphs and quasi-randomness, In *Random Struct. Alg.* , 2, (1991), 241-52.

[11] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, Introduction to Algorithms, third edition, *The MIT Press.*

[12] R. Diestel, Graph Theory, *Springer-Verlag*, New York, (1997).

[13] P. Erdős, Some recent results on extremal problems in graph theory, *International Symposium, Rome*, (1966), 118-123.

[14] P. Erdős, T. Gallai, On maximal paths and circuits of graphs, *Acta Math. Sci. Hungar.*, 10, (1959), 337-356.

[15] P. Erdős, A. Gyárfás, L. Pyber, Vertex coverings by monochromatic cycles and trees, *Journal of Combinatorial Theory*, Ser. B 51, (1991), 90-95.

[16] P. Erdős, A. H. Stone, On the structure of linear graphs, *Bull. Amer. Math. Soc*, 52, (1946), 1089-1091.

[17] P. Erdős, M. Simonovits, A limit theorem in graph theory, *Studia Sci. Math. Hung*, 1, (1966), 51-57.

[18] P. Erdős, P. Turán, On some sequences of integers, *J. London Math. Soc*, 11, (1936), 261-264.

[19] E. Fischer, A. Matsliah, A. Shapira, Approximate hypergraph partitioning and applications, In *Proceedings of the 48th annual IEEE Symposium on Foundations of Computer Science (FOCS)* , (2007), 579-589.

[20] C. Fowlkes, S. Belongie, F. Chung, J. Malik, Spectral grouping using the Nyström method, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26, (2004), 214-225.

[21] A. M. Frieze, R. Kannan, A simple algorithm for constructing Szemerédi's regularity partition, *Electron. J. Comb*, 6, (1999).

[22] S. Fujita, C. Magnant, Note on Highly Connected Monochromatic Subgraphs in 2-Colored Complete Graphs, *The Electronic Journal of Combinatorics*, Vol 18, Issue 1, (2011).

[23] S. Fortune, J. Wyllie, Parallelism in random access machines, *Proc. 10th ACM STOC*, (1978), 114-118.

[24] A. Gyárfás, Partition coverings and blocks sets in hypergraphs (in Hungarian), *Communications of the Computer and Automation Institute of the Hungarian Academy of Science*, 71 (1971), 66.

[25] L. Gerencseŕ, A. Gyárfás, On Ramsey-Type Problems *Annales Uni-*

versitatis Scientiarum Budapestinensis, Eotvos Sect. Math., 10, (1967), 167-170.

[26] R. E. Greenwood, A. M. Gleason, Combinatorial Relations and Chromatic Graphs, *Canadian Journal of Mathematics*, 7, (1955), 1-7.

[27] W. T. Gowers, Lower bounds of tower type for Szemerédi's uniformly lemma, Geom. Funct. Anal, 7, (1997), 322-337.

[28] W. T. Gowers, The Work of Endre Szemerédi. Exposition on Endre Szemerédi's work for the Abel Prize 2012, Online at http://www.abelprize.no/c54147/binfil/download.php?tid=54060.

[29] W. T. Gowers, Hypergraph regularity and the multidimensional Szemerédi theorem, In *Annals of Mathematics*, (2) 166 (2007), no.3, 897-946.

[30] L. M. Goldschlager, Synchronous parallel computation, Ph.D. Thesis, University of Toronto, (1977), see also, *J. ACM* 29, (1982), 1073-1086.

[31] A. Gyárfás, G. N. Sárközy, Size of monochromatic double stars in edge colorings *Graphs and Combinatorics*, 24, (2008), 531-536.

[32] A. Gyárfás, M. Ruszinkó, G. N. Sárközy and E. Szemerédi, Three-color Ramsey Numbers for Paths, *Combinatorica*, 27, (2007), 35-69.

[33] A. Gyárfás, M. Ruszinkó, G. N. Sárközy and E. Szemerédi, An improved bound for the monochromatic cycle partition number, *Journal of Combinatorial Theory, Ser. B* 96, (2006), 855-873.

[34] A. Gyárfás, M. Ruszinkó, G. N. Sárközy and E. Szemerédi, Partitioning 3-colored complete graphs into three monochromatic cycles, *Electronic Journal of Combinatorics* 18, (2011), #P53.

[35] A. Gyárfás, M. Ruszinkó, G. N. Sárközy and E. Szemerédi, One-sided coverings of colored complete bipartite graphs, *Topics in Discrete Mathematics (dedicated to J. Nesetril on his 60th birthday), Algorithms and Combinatorics*, 26, Springer, Berlin, (2006), 133-154.

[36] A. Gyárfás, Covering complete graphs by monochromatic paths, in *Irregularities of Partitions*, Algorithms and Combinatorics, Vol. 8, Springer-Verlag, (1989), 89-91.

[37] P. Haxell, Partitioning complete bipartite graphs by monochromatic cycles, *Journal of Combinatorial Theory*, Ser. B 69, (1997), 210-218.

[38] F. Harary, Recent Results on Generalized Ramsey Theory for Graphs, *Graph Theory and Applications*, Springer, Berlin, (1972), 125-138.

[39] P. Haxell and Y.Kohayakawa, Partitioning by monochromatic trees, *Journal of Combinatorial Theory, Ser. B* 68, (1996), 218-222.

[40] L. Hagen and A. Kahng, New spectral methods for ration cut partitions and clustering, *IEEE trans. Computer-Aided Design*, 11(9), 1074-1085.

[41] J. A. Hartigan, M. A. Wong, A K-Means Clustering Algorithm, In *J Royal Stat. Soc. Series C (App. Stat.)*, 28 (1), 100-108.

[42] D. Kühn, D. Osthus, Packings in Dense Regular Graphs, *Combinatorics, Probability and Computing*, 14, (2005), 325-337.

[43] R. Karp, V. Ramachandran, Parallel algorithms for shared memory machines, in *Handbook of Theoretical Computer Science*, J. Van Leeuven, ed, North Holland, (1990), 869-941.

[44] Y. Kohayakawa, V. Rödl, L. Thoma, An optimal algorithm for checking regularity, *SIAM J. Comput*, 32(5), (2003), 1210-1235.

[45] J. Komlós and M. Simonovits, Szemerédi's Regularity Lemma and its applications in graph theory, in *Combinatorics, Paul Erdős is Eighty* (D. Miklós, V.T. Sós, and T. Szőnyi, Eds.), 295-352, Bolyai Society Mathematical Studies, Vol. 2, János Bolyai Mathematical Society, Budapest, (1996).

[46] J. Komlós, G. N. Sárközy, E. Szemerédi, Blow-up Lemma, *Combinatorica*, 17 (1), (1997), 109-123.

[47] J. Komlós, G. N. Sárközy, E. Szemerédi, An algorithmic version of the Blow-up Lemma, *Random Structures and Algorithms* 12, (1998), 297-312.

[48] J. Komlós, A. Shokoufandeh, M. Simonovits, E. Szemerédi, The Regularity Lemma and Its Applications in Graph Theory, *Theoretical Aspects of Computer Science*, LNCS 2292, (2002), 84-112.

[49] A. T. Corbett, J. R. Anderson, Knowledge Tracing: Modeling the acquisition of procedural knowledge, *User Modeling and User-Adapted Interaction*, 4, (1995), 253-278.

[50] H. W. Kuhn, The Hungarian method for the Assignment Problem,

Naval Research Logistics, 52(1), (2005). Originally appeared in Naval Research Logistics Quarterly, 2, (1955), 83-97.

[51] L. Lovász, *Combinatorial Problems and Exercises*, North-Holland, Amsterdam, (1979).

[52] T. Łuczak, $R(C_n, C_n, C_n) \leq (4+o(1))n$, *Journal of Combinatorial Theory, Ser. B* 75, (1999), 174-187.

[53] H. Liu, R. Morris, N. Prince, Highly connected monochromatic subgraphs of multicoloured graphs, *Journal of Graph Theory*, 61(1), (2009), 22-44.

[54] T. Łuczak, V. Rödl, E. Szemerédi, Partitioning two-colored complete graphs into two monochromatic cycles, *Probability, Combinatorics and Computing*, 7, (1998), 423-436.

[55] U. Luxburg, A Tutorial on Spectral Clustering, In *Statistics and Computing*, Kluwer Academic Publishers, Hingham, MA, USA. Vol 17, Issue 4, (2007).

[56] D. Mubayi, Generalizing the Ramsey problem through diameter, *Electronic Journal of Combinatorics*, 9 (2002).

[57] A. Ng, M. Jordan and Y. Weiss, On Spectral Clustering: Analysis and an algorithm. In T. Dietterich, S. Becker, and Z. Ghahramani (Eds.), *Advances in Neural Information Processing Systems*, MIT Press, 14, (2002), 849-856.

[58] L. Pósa, On the circuits of finite graphs, *MTA Mat. Kut. Int. Közl.*, 8, (1963), 355-361.

[59] A. Pokrovskiy, Partitioning edge-coloured complete graphs into monochromatic cycles and paths, ArXiv:1205.5492v1.

[60] J. Pach, P. Agarwal, *Combinatorial Geometry*, Wiley & Sons, New York, (1995).

[61] F. P. Ramsey, On a Problem of Formal Logic, *Proceedings of the London Mathematical Society*, 30, (1930), 264-286.

[62] V. Rödl, M. Schacht, Regular partitions of hypergraphs: regularity lemmas, In *Combinatorics, Probability and Computing*, 16(6), 833-885.

[63] V. Rödl, B. Nagle, J. Skokan, M. Schacht, Y. Kohayakawa, The hypergraph regularity method and its applications, In *Proceedings of the National Academy of Sciences* USA, 102, (2005), 8109-8113.

[64] L. Zelnik-Manor, P. Perona, Self-tuning Spectral Clustering. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems* 17, MIT Press, Cambridge, MA, (2005), 1601-1608.

[65] G. N. Sárközy, Monochromatic cycle partitions of edge-colored graphs, *Journal of Graph Theory* 66, (2011), 57-64.

[66] G. N. Sárközy, Finding trees and cycles in graphs; existence theorems and fast parallel algorithms, Doctoral Dissertation, Rutgers University New Brunswick, NJ, USA (1994).

[67] J. Shi and J. Malik, Normalized cuts and image segmentation *IEEE trans. Pattern Analysis and Machine Intelligence*, 22(8), 888-905.

[68] A. Sperotto, M. Pelilo, Szemerédi Regularity Lemma and its Applications to Pairwise Clustering and Segmentation, In: *EMMCVPR, LNCS*, 4679. Springer, (2007).

[69] G. N. Sárközy, S. Selkow, Vertex partitions by connected monochromatic k-regular graphs, *Journal of Combinatorial Theory, Ser. B* 78, (2000), 115-122.

[70] G. N. Sárközy, S. Selkow, F. Song, Vertex partitions of non-complete graphs by connected monochromatic k-regular graphs, *Discrete Mathematics*, 311, (2011), 279-284.

[71] G. N. Sárközy, S. Selkow, F. Song, An improved bound for vertex partitions by connected monochromatic k-regular graphs, *Journal of Graph Theory* 72, (2013).

[72] G. N. Sárközy, F. Song, E. Szemerédi and S. Trivedi, A Practical Regularity Partitioning Algorithm and its Applications in Clustering, Submitted for publication.

[73] F. Song, S. Trivedi, Y. Wang, G. N. Sárközy, N. T. Heffernan, Applying Clustering to the Problem of Predicting Retention within an ITS: Comparing Regularity Clustering with Traditional Methods, Accepted for publication in *FLAIRS*.

[74] M. Stoer, F. Wagner, A simple min-cut algorithm, *J. ACM*, 44(4), 585-591.

[75] E. Szemerédi, Regular partitions of graphs, Colloques Internationaux C.N.R.S. *Problèmes Combinatoires et Théorie des Graphes*, Orsay, (1976), 399-401.

[76] Pál Turán, On an extremal problem in graph theory (in Hungarian), *Matematikai és Fizikai Lapok*, 48, (1941), 436-452.

[77] S. Trivedi, Z. A. Pardos, N. T. Heffernan, Clustering Students to Generate an Ensemble to Improve Standard Test Predictions, The fifteenth international Conference on Artificial Intelligence in Education, (2011).

[78] S. Trivedi, Z. A. Pardos, G. Sarkozy, N. T. Heffernan, Spectral Clustering in Educational Data Mining, Proceedings of *the 4th International Conference on Educational Data Mining*, (2011), 129-138.

[79] A. Frank, A. Asuncion, UCI Machine Learning Repository, Irvine, CA: University of California, School of Information and Computer Science, (2010).

[80] Y. Wang, J. E. Beck, Incorporating Factors Influencing Knowledge Retention into a Student Model, In the Proceedings of *the 5th International Conference on Educational Data Mining*, (2012), 201-203.

[81] M. Wu, B. Schölkopf, A Local Learning Approach for Clustering, In *Proceedings of the Neural Information Processing Systems*, (2007), 1529-1536.

[82] D. Zhou, J. Huang, B. Schölkopf, Learning with Hypergraphs: Clustering, Classification, and Embedding, In *Advances in Neural Information Processing Systems* , 19, (2007), 1601-1608.