CHAPTER 1

Very large networks

1.1. Huge networks everywhere

In the last decade it became apparent that a large number of the most interesting structures and phenomena of the world can be described by networks: often the system consists of discrete, well separable elements, with connections (or interactions) between certain pairs of them. To understand the behavior of the whole system, one has to study the behavior of the individual elements as well as the structure of the underlying network. Let us see some examples.

- Among very large networks, probably the best known and the most studied is the *internet*. Moreover, the internet (as the physical underlying network) gives rise to many other networks: the network of hyperlinks (web, logical internet), internet based social networks, distributed data bases, etc. The size of the internet is growing fast: currently the number of web pages may be 30 billion ($3 \cdot 10^{10}$) or more, and the number of interconnected devices is probably more than a billion. The graph theoretic structure of the internet determines, to a large degree, how communication protocols should be designed, how likely certain parts get jammed, how fast computer viruses spread etc.
- Social networks are basic objects of many studies in the area of sociology, history, epidemiology and economics. They are not necessarily formally established, like Facebook and other internet networks: The largest social network is the acquaintance graph of all living people, with about 7 billion nodes. The structure of this acquaintance graph determines, among others, how fast news, inventions, religions, diseases spread over the world, now and during history.
- Biology contributes ecological networks, networks of interactions between proteins, and the human brain, just to mention a few. The human brain, a network of neurons, is really large for its mass, having about a hundred billion nodes. One of the greatest challenges is, of course, to understand ourselves.
- Statistical physics studies the interactions between large numbers of discrete particles, where the underlying structure is often described by a graph. For example, a crystal can be thought of as a graph whose nodes are the atoms and whose edges represent chemical bonds. A perfect crystal is a rather boring graph, but impurities and imperfections create interesting graph-theoretical digressions. 10 gram of a diamond has about 5×10^{23} nodes. The structure of a crystal influences important macroscopic properties like whether the material is magnetizable, or how it melts.
- Some of the largest networks in engineering occur in chip design. There can be more than a billion transistors on a chip nowadays. Even though these networks are man-made and carefully designed, many of their properties, like

the exact time they will need to perform some computation, are difficult to determine from their design, due to their huge size.

- To be pretentious, we can say that the whole universe is a single (really huge, possibly infinite) network, where the nodes are events (interactions between elementary particles), and the edges are the particles themselves. This is a network with perhaps 10^{80} nodes. It is an ongoing debate in physics how much additional structure the universe has, but perhaps understanding the graph-theoretical structure of this graph can help with understanding the global structure of the universe.

These huge networks pose exciting challenges for the mathematician. Graph Theory (the mathematical theory of networks) has been one of the fastest developing areas of mathematics in the last decades; with the appearance of the Internet, however, it faces fairly novel, unconventional problems. In traditional graph theoretical problems the whole graph is exactly given, and we are looking for relationships between its parameters or efficient algorithms for computing its parameters. On the other hand, very large networks (like the Internet) are never completely known, in most cases they are not even well defined. Data about them can be collected only by indirect means like random local sampling or by monitoring the behavior of various global processes.

Dense networks (in which a node is adjacent to a positive percent of other nodes) and very sparse networks (in which a node has a bounded number of neighbors) show a very different behavior. From a practical point of view, sparse networks are more important, but at present we have more complete theoretical results for dense networks. In this introduction, most of the discussion will focus on dense graphs; we will survey the additional challenges posed by sparse networks in Section 1.7.

1.2. What to ask about them?

Think of a really large graph, say the internet, and try to answer the following four simple questions about it.

Question 1. ***Does the graph have an odd or even number of nodes?***

This is a very basic property of a graph in the classical setting. For example, it is one of the first theorems or exercises in a graph theory course that every graph with an odd number of nodes must have a node with even degree.

But for the internet, this question is clearly nonsense. Not only does the number of nodes change all the time, with devices going online and offline, but even if we fix a specific time like 12:00am today, it is not well-defined: there will be computers just in the process of booting up, breaking down etc.

Question 2. ***What is the average degree of nodes?***

This, on the other hand, is a meaningful question. Of course, the average degree can only be determined with a certain error, and it will change as technology or the social composition of users change; but at a given time, a good approximation can be sought (I am not speaking now about how to find it).

Question 3. ***Is the graph connected?***

To this question, the answer is almost certainly no: somewhere in the world there will be a faulty router with some unhappy users on the wrong side of it. But this is not the interesting way to interpret the question: we should consider the

internet "disconnected" if, say, an earthquake combined with a sunflare severs all connections between the Old and New worlds. So we want to ignore small components that are negligible in comparison with the whole graph, and consider the graph "disconnected" only if it decomposes into two parts which are commeasurable with the whole. On the other hand, we may want to allow that the two large parts be connected by a very few edges, and still consider the graph "disconnected".

Question 4. ***Where is the largest cut in the graph?***

(This means to find the partition of the nodes into two classes so as to maximize the number of edges connecting the two classes.) This example shows that even if the question is meaningful, it is not clear in what form can we expect the answer. We can ask for the fraction of edges contained in the largest cut (depending on the model, this can be determined relatively easily, with an error that is small with high probability, although it is not easy to prove that the algorithm works). But suppose we want to "compute" the largest cut itself; how to return the result, i.e., how to specify the largest cut (or even an approximate version of it)? We cannot just list all nodes and tell on which side do they belong: this would be too much time and memory space. Is there a better way to answer the question?

1.3. How to obtain information about them?

If we face a large network (think of the internet) the first challenge is to obtain information about it. Often, we don't even know the number of nodes.

1.3.1. Sampling. Properties of very large graphs can be studied by randomly sampling small subgraphs. The theory of this, sort of a statistics where we work with graphs instead of numbers, is called *property testing* in computer science. Initiated by Goldreich, Goldwasser and Ron [1998], this theory emerged in the last 15-20 years, and will be one of the main areas of applications of the methods developed in this book.

In the case of dense graphs G, it is simple to describe a reasonably realistic sampling process: we select independently a fixed number k of random nodes, and determine the edges between them, to get a random induced subgraph (Figure 1.1). We have to assume, of course, that we have methods to select a uniformly distributed random node of the graph, and to determine whether two nodes are adjacent. We'll call this *subgraph sampling*. For each graph F on $[k] = \{1, 2, \ldots, k\}$, there is a certain probability of seeing F when k nodes are sampled, which we denote by $\sigma_{G,k}(F)$. So every graph G defines a probability distribution $\sigma_{G,k}$ on all graphs with k nodes. It turns out that this sample contains enough information to determine many properties and parameters of the graph, with some error of course. This error can be made arbitrarily small with high probability if we choose the sample size k sufficiently large, depending on the error bound (and only on the error bound, not on the graph!).

One may try to strengthen this and allow this sampling process to be repeated a bounded number of times. This would not give anything new, however: sampling k nodes r times gives less information than sampling kr nodes once. For clarity, it is sometimes better to describe algorithms saying that we repeat a certain sampling process, but this could always be replaced by taking a single sample (larger, but still of bounded size).

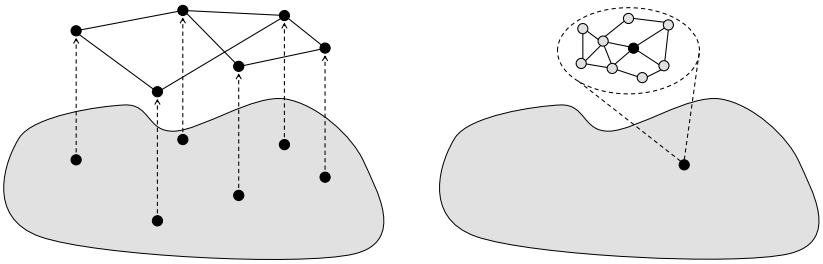


FIGURE 1.1. Sampling from a dense graph and from a graph with bounded degree.

1.3.2. Global observables. Instead of taking a random subset of the nodes (sampling) and studying the subgraph induced by them, we can take a random partition of the nodes into a small number of classes, and study the "quotient", the small graph obtained by merging the classes of the partition. (This will have very large edge multiplicities, which we have to normalize appropriately.) These quotients carry information about global measurements (like the number of stable sets, the maximum cut, various quantities in statistical physics, etc.). The remarkable fact is that under the right conditions, these "global" observables carry the same information as "local" sampling (see Sections 12.3 and 20.2.)

Another source of information about a very large network is the observation of the behavior of various processes on the graph, through a longer time interval. The observation can be global (measurement of some global parameter), or local (at one node, or a few neighboring nodes). Observing heat propagation through a material is an example of the first kind of approach; web crawlers can be considered as examples of the second, and in a sense so is our observation of the universe. There are some sporadic results about the local observation of simple random processes (Benjamini and Lovász [2002], Benjamini, Kozma, Lovász, Romik, and Tardos [2006], but a general theory of such local observation of global processes has not emerged yet.

1.3.3. Left and right homomorphisms. In theoretical studies, it is often more convenient to talk about homomorphisms (adjacency-preserving maps) between graphs, instead of looking at randomly chosen induced subgraphs. For two finite simple graphs F and G, let $\mathsf{hom}(F,G)$ denote the number of homomorphisms of F into G (adjacency-preserving maps from $V(F)$ to $V(G)$). We often normalize these homomorphism numbers, to get *homomorphism densities*:

$$t(F,G) = \frac{\mathsf{hom}(F,G)}{\mathsf{v}(G)^{\mathsf{v}(F)}}. \tag{1.1}$$

This number is the probability that a random map of $V(F)$ into $V(G)$ preserves adjacency. (We denote by $V(F)$ and $E(F)$ the sets of nodes and edges of the graph F, respectively, and their cardinalities by $\mathsf{v}(F) = |V(F)|$ and $\mathsf{e}(F) = |E(F)|$.)

Homomorphisms will be basic tools throughout the book. We introduce them in Chapter 5 (where we survey some of our knowledge about them), but use them all the time thereafter. There will be different versions like homomorphisms into weighted graphs, which play an important role in statistical physics (we will return to them at the end of the Introduction).

Homomorphism densities can be expressed in terms of the distribution of samples, and vice versa (at least asymptotically, as the size of G tends to infinity). For example, let us consider the homomorphism density of the quadrilateral C_4 in a large graph G. If we map the four nodes of C_4 into G, if may be that the images are different, and so the image is a quadrilateral. It could also happen that the images of two nodes coincide. This cannot happen to adjacent nodes, because the image of an edge must be an edge; but it may happen to two opposite nodes. In this case, the image is a "V" (path with two edges). Or it can happen that both pairs of opposite nodes have the same image, and the image is a single edge. If we know the numbers of edges, V's and quadrilaterals in G, then we can compute the number of homomorphisms of the quadrilateral into G. (Warning: the same quadrilateral in G can be the image in 8 different ways, the same V can be the image in 4 ways, and the same edge, in 2 ways!). The numbers of quadrilaterals, V's and edges can be estimated by sampling. In fact, the last two will not matter much for very large graphs G, since a random map of 4 elements into $\mathsf{v}(G)$ elements will be one-to-one with high probability.

So homomorphism densities and sampling distributions carry the same information, why bother to introduce both? Using homomorphisms has several advantages (and some disadvantages).

- Homomorphism numbers are better behaved algebraically, and they have been used before to study various algebraic questions concerning direct product of graphs, like cancellation laws (see Section 5.4.2). Furthermore, a lot is known about other issues concerning homomorphisms: existence, structure, etc.

- When looking at a (large) graph G, we may try to study its local structure by counting homomorphisms from various "small" graphs F into G; we can also study its global structure by counting homomorphisms from G into various small graphs H. The first type of information is closely related (in many cases, equivalent) to sampling, while the second is related to global observables. This way homomorphisms are pointing at a certain duality between sampling and global observation. We can sum up our framework for studying large graphs in the following formula:

$$F \longrightarrow G \longrightarrow H.$$

  We will informally talk about "left-homomorphisms" and "right-homomorphisms" to refer to these two kind of mappings.
- We will characterize which distributions come from sampling k nodes from a (large) graph G, and we will characterize homomorphism densities as well. It turns out that a characterization of sample distributions is simpler and more natural, but putting it in another way, the characterization of homomorphism densities is more surprising, and therefore has more interesting applications.

- Using homomorphisms leads us to looking at things through the spectacles of category theory, and this point of view is very fruitful. For example, sometimes one can simply "turn arrows around", and get new results almost for free. We will say more about this generalization to categories near the end of the book, in Section 23.4.

1.4. How to model them?

1.4.1. Random graphs. We celebrated the 50th birthday of random graphs recently: The simplest random graph model was developed by Erdős and Rényi [1959] and Gilbert [1959]. Given a positive integer n and a real number $0 \le p \le 1$, we generate a random graph $\mathbb{G}(n,p)$ by taking n nodes, say $[n] = \{1,\dots,n\}$, and connecting any two of them with probability p, making an independent decision about each pair.

There are alternate models, which are essentially equivalent from the point of view of many properties. Two of these were introduced in the early papers by Erdős–Rényi [1959, 1960]: We could fix the number of edges m, and then choose a random m-element subset of the set of pairs in $[n]$, uniformly from all such subsets. This random graph $\mathbb{G}(n,m)$ has very similar properties to $\mathbb{G}(n,p)$ when $m = p\binom{n}{2}$. Another model, closer to some of the more recent developments, is *evolving random graphs*, where edges are added one by one, always choosing uniformly from the set of unconnected pairs. Stopping this process after m steps, we get $\mathbb{G}(n,m)$.

Random graphs have many interesting, often surprising properties, and a huge literature, see Bollobás [2001], Janson, Łuczak and Ruczinski [2000], or Alon and Spencer [2000].

One conventional wisdom about random graphs with a given edge density is that they are all alike. Their basic parameters, like chromatic number, maximum clique, triangle density, spectra etc. are highly concentrated. This fact will be an important motivation when defining the right measure of global similarity of two graphs in Chapter 8.

Many generalizations of this random graph model have been studied. For example, we can consider a "template" for the random graph in the form of a weighted graph H on q nodes, with a weight $\alpha_i > 0$ associated with each node, and a weight $0 \le \beta_{ij} \le 1$ associated with each edge ij. We assume that the nodeweights sum to 1. We may also assume that H is complete with a loop at every node, since the missing edges can be added with weight 0. A *multitype random graph* $\mathbb{G}(n;H)$ with template H is generated as follows: We take $[n] = \{1,\dots,n\}$ as its node set, where we think of n as a much larger number than q. We partition $[n]$ into q sets $V_1,\dots,V_q$, by putting node u in V_i with probability α_i, and connecting each pair $u \in V_i$ and $v \in V_j$ with probability β_{ij} (all these decisions are made independently).

While multitype random graphs are too close to the original Erdős–Rényi model to be useful as, say, internet models, they play an extremely important role by serving as simple objects approximating arbitrarily large graphs (see Section 1.5.2 and Chapter 9).

More generally, one could have different probabilities assigned to different edges (this was suggested by Erdős and Rényi in their second paper [1960] already). The construction we will use a lot, namely constructing a random graph from a symmetric measurable function $[0,1]^2 \to [0,1]$ is a related idea, discovered independently several times, first time, probably, by Diaconis and Freedman [1981]. These random graphs, which we call W-random, will be discussed in Section 10.1 and will play an important role throughout the book.

1.4.2. Quasirandom graphs. Deterministic objects that look and behave like randomly generated ones are important in various branches of science. For

example, pseudorandom number generators are basic algorithms in computer science, with many applications in Monte-Carlo algorithms, computer security and elsewhere. Exact definitions are usually difficult to give, and they vary according to need. It is very remarkable that in graph theory it is possible to give a very robust definition of quasirandom graphs, where many related or even seemingly quite different formalizations of properties of random graphs capture the same notion. We know that random graphs have a variety of quite strict properties (with high probability); it turns out that for several of these basic properties, the exceptional graphs are the same. In other words, any of these properties implies the others, regardless of any stochastic consideration.

A measure of quasirandomness of a graph was introduced by Thomason [1987]; the theory of quasirandom graph sequences, which has been an important example for convergent graph sequences central to this book, was developed by Chung, Graham and Wilson [1989].

To make this idea precise, we consider a sequence of graphs (G_n) with $\mathsf{v}(G_n) \to \infty$. For simplicity of notation, assume that $\mathsf{v}(G_n) = n$. Let $0 < p < 1$ be a real number. Consider the following properties of the sequence of graphs:

(QR1) Almost all degrees are asymptotically pn and almost all codegrees (numbers of common neighbors of two nodes) are asymptotically $p^2 n$.

(QR2) For every fixed graph F, the number of homomorphisms of F into G_n is asymptotically $p^{\mathsf{e}(F)} n^{\mathsf{v}(F)}$.

(QR3) The number of edges is asymptotically $pn^2/2$ and the number of 4-cycles is asymptotically $p^4 n^4/8$. (We have to divide by 2 and 8, because we are counting unlabeled copies rather than homomorphisms.)

(QR4) The number of edges induced by any set of $n/2$ nodes is asymptotically $pn^2/8$.

(QR5) For any two disjoint sets X, Y of nodes, the number of edges between X and Y is $p|X||Y| + o(n^2)$.

All these properties hold with probability 1 if $G_n = \mathbb{G}(n,p)$. However, more is true: if a graph sequence satisfies either one of them, then it satisfies all. Perhaps the most surprising fact along these lines is the equivalence of the second and third: prescribing the right asymptotic number of copies in G_n for just two small graphs (the edge and the 4-cycle) forces every other simple graph to have (asymptotically) the right number of copies.

Such graph sequences are called *quasirandom*. The five properties above are only a sampler; there are many other properties of random graphs that are also equivalent to these (Chung, Graham and Wilson [1989], Simonovits and Sós [1991, 1997, 2003]).

Many interesting deterministic graph sequences are quasirandom. We mention an important example from number theory:

Example 1.1 **(Paley graphs)**. Let q be any prime congruent 1 modulo 4, and let us define a graph on $\{0, \dots, q-1\}$ by connecting i and j if and only if $i - j$ is a quadratic residue modulo q. We construct this graph for every such prime, and order them in a sequence.

This graph sequence is quasirandom with density 1/2. (To verify the first property above is perhaps the easiest; see Exercise 1.2). This example also illustrates how some of the equivalent conditions above may be much easier to verify than

others: in this case, the third would be as easy as the verification of the first, but the second and fourth would be quite difficult: How would you count the number of copies of, say, the Petersen graph in a Paley graph? How would you count the number of those differences that are quadratic residues between, say, square-free integers in $\{0, \dots, q-1\}$? When posed directly, these questions sound formidable; but the equivalence of the above conditions provides answers to them. ♦

We should emphasize that in this setting, quasirandomness is a property of a sequence of graphs, not of a single graph. Of course, one could introduce a measure of deviation from the "ideal" quasirandomness in each of the conditions (QR1)–(QR5), and prove explicit relationships between them. Since our interest is the limit theory, we will not go in this direction.

Sometimes we need to consider quasirandom bipartite graphs, which can be defined, *mutatis mutandis*, by any of the properties above. More generally, just as there are multitype random graphs, there are also *multitype quasirandom graph sequences*. Similarly as for random graphs, a multitype quasirandom graph sequence (G_n) is defined by a "template" weighted graph H on q nodes, with a nodeweights $\alpha_i > 0$ and edgeweights β_{ij}. The sequence is multitype quasirandom with template H, if the node set $V(G_n)$ can be partitioned into q sets $V_1, \dots, V_q$ such that $|V_i| \sim \alpha_i \mathsf{v}(G_n)$, the subgraphs $G_n[V_i]$ induced by V_i form a quasirandom sequence for every $i \in [q]$, and the bipartite subgraphs $G_n[V_i, V_j]$ between V_i and V_j form a quasirandom bipartite graph sequence for each pair $i \neq j \in [q]$.

The same remark applies as for multitype random graphs: they play an extremely important role by serving as simple objects approximating arbitrarily large graphs. The equivalence of conditions (Q1)–(Q5) can be generalized appropriately (with a larger, but finite set of graphs in (Q3) instead of just 2), as it will be discussed in Section 16.7.1.

The main topic of the book, the theory of convergent graph sequences, can be considered as a further, rather far-reaching generalization of quasirandom sequences.

1.4.3. Randomly growing graphs. Random graph models on a fixed set of nodes, discussed above, fail to reproduce important properties of real-life networks. For example, the degrees of Erdős–Rényi random graphs follow a binomial distribution, and so they are asymptotically normal if the edge probability p is a constant, and asymptotically Poisson if the expected degree is constant (i.e., $p = p(n) \sim c/n$). In either case, the degrees are highly concentrated around the mean, while the degrees of real life networks tend to obey the "Zipf phenomenon", which means that the tail of the distribution decreases according to a power law (unlike the most familiar distributions like Gaussian, geometric or Poisson, whose tail probability drops exponentially; Figure 1.2).

In 1999 Albert and Barabási [1999, 2002, 2002] created a new random network model. Perhaps the main new feature compared with the Erdős–Rényi graph evolution model is that not only edges, but also nodes are added by natural rules of growing. When a new node is added, it connects itself to a given number d of old nodes, where each neighbor is selected randomly, with probability proportional to its degree. (This random selection is called *preferential attachment*.) The Albert–Barabási graphs reproduce the "heavy tail" behavior of the degree sequences of real-life graphs. Since then a great variety of growing networks were introduced, reproducing this and other empirical properties of real-life networks.

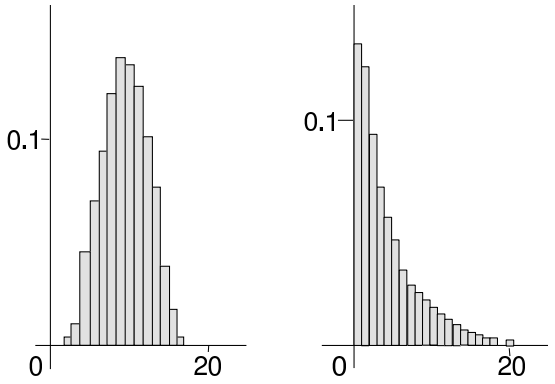


FIGURE 1.2. Degree distributions of an Erdős–Rényi random graph on 100 nodes with edge density .1 (left) and of a real life graph with similar parameters (right). The main feature to observe about the latter is not that the largest frequency is 1, but that it is much more stretched out.

This is perhaps the first point which suggests one of our main tools, namely assigning limits to sequences of graphs. Just as the Law of Large Numbers tells us that adding up more and more independent random variables we get an increasingly deterministically behaving number, these growing graph sequences tend to have a well-defined structure, for almost all of the possible random choices along the way. In the limit, the randomness disappears, and the asymptotic behavior of the sequence can be described by a well-defined limit object. We will return to this in this Introduction in Sections 1.5.3 and 11.3.

EXERCISE 1.2. Prove that the sequence of Paley graphs is quasirandom.

1.5. How to approximate them?

If we want to experiment with a large network (say, try out a new protocol for the internet), then it is good to have a "scaled down" version of it. In other words, we want a compact approximate description of a very large network, from which a network similar to the original, but of suitable size, can be generated. To make this mathematically precise, we need to define what we mean by two graphs to be "similar" or "close", and describe what kind of structures we use for approximation.

1.5.1. The distance of two graphs. There are many ways of defining the distance of two graphs G and G'. Suppose that the two graphs have a common node set $[n]$. Then a natural notion of distance is the *edit distance*, defined as the number of edges to be changed to get from one graph to the other. This could also be viewed as the *Hamming distance* $|E(G)\triangle E(G')|$ of the edge sets ($\triangle$ denotes symmetric difference). Since our graphs are very large, we want to normalize this. If the graphs are dense, then a natural normalization is

$$d_1(G,G') = \frac{|E(G)\triangle E(G')|}{n^2}.$$

While this distance plays an important role in the study of testable graph properties, it does not reflect structural similarity well. To raise one objection, consider two

random graphs on $[n]$ with edge density $1/2$. As mentioned in the introduction, these graphs are very similar from almost every aspect, but their normalized edit distance is large (about $1/2$ with high probability). One might try to decrease this by relabeling one of them to get the best overlay minimizing the edit distance; but the improvement would be marginal (tending to 0 if n tends to infinity).

Another trouble with the notion of edit distance is that it is defined only when the two graphs have the same set of nodes. We want to define a notion of distance for two graphs that are so large that we don't even know the number of their nodes, and these numbers might be very different. For example, we want to find that two large random graphs are "close" even if they have a different number of nodes.

One useful way to overcome these difficulties is to base the measurement of distance on sampling. Recall that for a graph G, $\sigma_{G,k}$ is the probability distribution on graphs on $[k] = \{1, 2, \dots, k\}$ obtained by selecting a random ordered k-subset of nodes and taking the subgraph induced by them. Strictly speaking, this is only defined when $k \le \mathsf{v}(G)$; but we are interested in taking a small sample from a large graph, not the other way around. To make the definition precise, let us say that the sampling returns the edgeless k-node graph if $k > \mathsf{v}(G)$. (In this case it would be a better solution to sample with repetition, but sampling without repetition is better in other cases, so let us stick to it.)

Now if we have two graphs G and G', we can compare the distributions of k-node samples for any fixed k. We use the *variation distance* between distributions α and β on the same set, defined by

$$d_{\mathrm{var}}(\alpha, \beta) = \sup_X |\alpha(X) - \beta(X)|,$$

where the supremum is taken over all measurable subsets (observable events). If we want to measure the distance of two graphs by a single number, we use a simple trick known from analysis: We define the *sampling distance* of two dense graphs G and G' by

$$\delta_{\mathrm{samp}}(G, G') = \sum_{k=1}^{\infty} \frac{1}{2^k} d_{\mathrm{var}}(\sigma_{G,k}, \sigma_{G',k}) \tag{1.2}$$

(Here the coefficients $1/2^k$ are quite arbitrary, they are there only to make the sum convergent; but the above is a convenient choice.) This distance notion is very suitable for our general goals, since two graphs are close in this distance if and only if random sampling of "small" induced subgraphs does not distinguish them reliably. However, sampling distance has one drawback: it does not directly reflect any structural similarity.

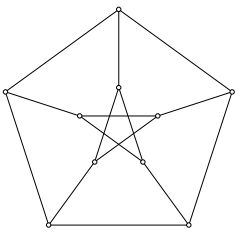
In Chapter 8 we will define a notion of distance, called *cut distance*, between graphs, which will be satisfactory from all these points of view: it will be defined for two graphs with possibly different number of nodes, the distance of two random graphs with the same edge density will be very small, and it will reflect global structural similarity. The definition involves too many technical details to be given here, unfortunately. But it will turn out (and this is one of the main results in this book) that the cut distance is equivalent to the sampling distance in a topological sense.

1.5.2. Approximation by smaller: Regularity Lemmas. Let us return to the question of "scaling down" a huge graph, first in the dense case. The main

tool for doing so is the "Szemerédi Partition" or "Regularity Lemma". Szemerédi developed the first version of the Regularity Lemma for his celebrated proof of the Erdős–Turán Conjecture on arithmetic progressions in dense sets of integers in 1975. Since then, the Lemma has emerged as a fundamental tool in graph theory, with many applications in extremal graph theory, combinatorial number theory, graph property testing etc., and became a true focus of research in the past years.

Informally, the Regularity Lemma says that every graph can be approximated by a multitype quasirandom graph, where the number of classes depends on the error of the approximation only. This lemma can be viewed as an archetypal example of dichotomy between randomness and structure, where we try to decompose a (large and complicated) object A into a more highly structured object A' with a (quasi)random perturbation (cf. Tao [2006c]). The highly structured part may be easier to handle because of the structure, and the quasirandom part will often be easier to handle due to Laws of Large Numbers.

Pixel pictures. In this introductory part, we want to illustrate the idea of a regularity partition visually. To this end, let us introduce a non-standard way of visualizing graphs. On the left of Figure 1.3 we see a graph (the Petersen graph). In the middle, we see its adjacency matrix. On the right, we see another version of its adjacency matrix, where the 0's are replaced by white squares and the 1's are replaced by black squares. We think of the whole picture as the unit square, so the little squares have side length $1/n$, where n is the number of nodes. The origin is in the upper left corner, following the convention of indexing matrix elements.



0	1	0	0	1	1	0	0	0	0
1	0	1	0	0	0	1	0	0	0
0	1	0	1	0	0	0	1	0	0
0	0	1	0	1	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
1	0	0	0	0	0	0	1	1	0
0	1	0	0	0	0	0	0	1	1
0	0	1	0	0	1	0	0	0	1
0	0	0	1	0	1	1	0	0	0
0	0	0	0	1	0	1	1	0	0

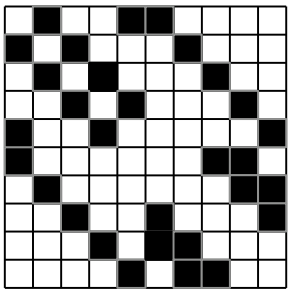


FIGURE 1.3. The Petersen graph, its adjacency matrix, and its pixel picture

It is not clear that this pixel picture reveals more about small graphs than the usual way of drawing them (probably less), but it can be suggestive for large graphs. Figure 1.4 shows the usual drawing and the pixel picture of a *half-graph*, a bipartite graph defined on the set $\{1, \dots, n, 1', \dots, n'\}$, where the edges are the pairs (i, j) with $i \leq j'$. For large n, the pixel picture of a half-graph may be more informative, as we will see in the next section.

The left square in Figure 1.5 is the pixel picture of a (reasonably large) random graph. We don't see much structure—and we shouldn't. From a distance, this picture is more-or-less uniformly grey, similar to the second square. The 100×100 chessboard in the third picture is also uniformly grey, or at least it would become so if we increased the number of pixels sufficiently. One might think that it represents

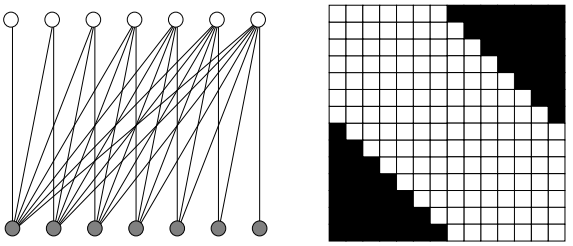


FIGURE 1.4. A half-graph and its pixel picture

a graph that is close to the random graph. But rearranging the rows and columns so that odd indexed columns come first, we get the 2×2 chessboard on the right! So wee see that both the middle and the right side pictures represent a complete bipartite graph. *The pixel picture of a graph depends on the ordering of the nodes.* We can be reassured, however, that a random graph remains random, no matter how we order the nodes, and so the picture on the left remains uniformly grey, no matter how the nodes are ordered.

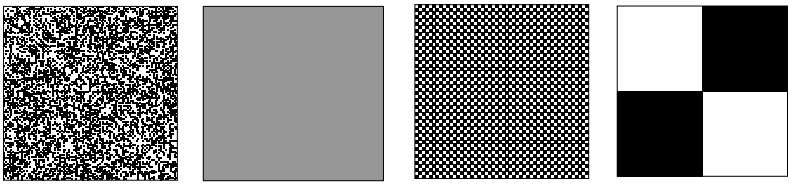


FIGURE 1.5. A random graph with 100 nodes and edge density 1/2, a random graph with very many nodes and edge density 1/2, a chessboard, and the pixel picture obtained by rearranging the rows and columns.

REMARK 1.3. Using pixel pictures to represent graphs, in particular random graphs, goes in a sense in the opposite direction to what was studied in the psychology of vision. Of course, processing images given by pixel pictures has been a fundamental issue in connection with computer graphics and related areas, and we are not going into this issue in this book. But we should mention the work of Julesz, who studied the question of how well the human eye can distinguish random noise (like Figure 1.5(a)) from images that are also uniformly grey but more structured (textured). The chessboard in Figure 1.5(b) would be a trivial example of such an image. Disproving some of his conjectures, Diaconis and Freedman [1981] constructed pixel pictures that are very closely related to our W-random graphs.

The Regularity Lemma. We illustrate the Regularity Lemma by Figure 1.6. The graph on the left side (given by its pixel picture) looks quite random. In the middle we see the same graph, with its nodes ordered differently. In this picture, we see some structure of the graph (even though it is not as clear-cut as in Figure 1.5); what we see is that the upper left corner is denser, and the lower right corner is sparser. If we cut the picture into four equal parts, and average the "blackness" in each, we get the picture on the right. Inside each of the four parts, the arrangement

is quite random-like, and further rearrangement would not reveal any additional structure.

Still informally, the Regularity Lemma says the following:

The nodes of every graph can be partitioned into a "small" number of "almost equal" parts in such a way that for "almost all" pairs of partition classes, the bipartite graph between them is "quasirandom".

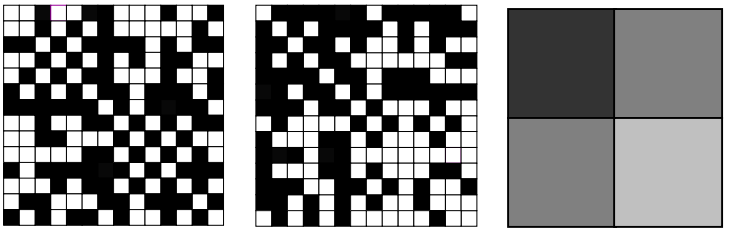


FIGURE 1.6. A random-looking pixel picture, an informative rearrangement, and its regularity partition

Some of the expressions in quotation marks are easy to explain. For the whole theorem, we have an error bound $0 < \varepsilon < 1$ specified in advance. The condition that the parts are "almost equal" means that their sizes differ by at most one: if the graph has n nodes partitioned into k classes, then the size of each class is either $\lfloor n/k \rfloor$ or $\lceil n/k \rceil$. The condition that the number of classes is "small" means that it can be bounded by an explicit function $f(\varepsilon)$ of ε; to exclude trivialities, we also assume that $k \geq 1/\varepsilon$. "Almost all" pairs of classes means that we allow $\varepsilon\binom{k}{2}$ exceptional pairs about which we don't claim anything (we can include the subgraphs induced by the classes among these exceptions). Finally, we need to define what it means to be "random-like": one way to put it is that this bipartite graph is quasirandom with some density p_{ij} (which may be different for different pairs of classes) and with error ε, in the sense introduced (informally) in Section 1.4.2.

Regularity partitions and quasirandomness have a lot to do with each other. Not only is quasirandomness part of the statement of the Regularity Lemma, but the regularity lemma can be used to characterize quasirandomness: Simonovits and Sós [1991] proved that a graph sequence is quasirandom with density p if and only if the graphs have regularity partitions for arbitrarily small $\varepsilon > 0$ such that the densities p_{ij} between the partition classes tend to p.

I have to come back to the "small" number of partition classes. The proof gives a tower $2^{2^{2^{\cdots}}}$ of height $1/\varepsilon^5$, which is a very large number, and which unfortunately cannot be improved too much, since Gowers [1997] constructed graphs for which the smallest number of classes in a Szemerédi partition was at least a tower of height $\log(1/\varepsilon)$. So the tower behavior is a sad fact of life. There are related partitions with a more decent number of classes, as we shall see in Chapter 9, where regularity partitions will be defined formally. We will also discuss situations when the regularity partitions have a very decent size, like $1/\varepsilon^{\text{const}}$ (Sections 13.4 and 16.7). Implicitly or explicitly, regularity partitions will be used throughout this book.

1.5.3. Approximation by infinite: convergence and limits. This idea can be motivated by how we look at a large piece of metal. This is a crystal, that is a really large graph consisting of atoms and bonds between them. But from many points of view (e.g., the use of the metal in building a bridge), it is more useful to consider it as a continuum with a few important parameters (density, elasticity etc.). Its behavior is governed by differential equations relating these parameters. Can we consider a more general very large graph as some kind of a continuum?

Our way to make this intuition precise is to consider a growing sequence (G_n) of graphs whose number of nodes tends to infinity, to define when such a sequence is convergent, and to assign a limit object to convergent graph sequences, which somehow incorporates all the properties we want to be remembered. (We have mentioned this idea in connection with randomly growing graphs, but now we don't assume anything about how the graphs in the sequence are obtained.) This plan is the backbone of this book: we will carry it out both for dense graphs and also for graphs with bounded degree. There will be a good collection of applications of this work.

Our discussion of sampling from a graph suggests a general principle leading to a definition: we consider samples of a fixed size k from G_n, and their distribution. We say that the sequence is *locally convergent* (with respect to the given sampling method) if this distribution tends to a limit as $n \to \infty$ for every fixed k.

For dense graphs, this notion of convergence was defined by Borgs, Chayes, Lovász, Sós, and Vesztergombi [2006, 2008]; some elements of this definition go back to Erdős, Lovász and Spencer [1979]. This notion has many useful properties. Perhaps most important of these is that it can be characterized in terms of the cut distance of graphs. It is not hard to see that the above notion of convergence is equivalent to saying that the graph sequence is a Cauchy sequence in the sampling distance. One of the main results presented in this book is Theorem 11.3, which can be stated informally as follows:

The same graph sequences are convergent (Cauchy sequences) for both the cut distance and the sampling distance.

If we have a notion of convergence, the question arises naturally: what does it converge to? Can we describe a limit object for every convergent graph sequence? The family of limiting sample distributions (one for each k) can be considered as a limit object of the sequence (we call this the "weak limit"). This is not always a helpful representation of the limit object, and a more explicit description is desirable.

A next step is to represent the family of distributions on finite graphs (the samples) by a single probability distribution on countable graphs: we get certain notion of random graphs on the countable set $\mathbb{N}^* = \{1, 2, 3, \dots\}$ (see Theorem 11.52).

More explicit descriptions of these limit objects can also be given, in the form of a two-variable measurable function $W : [0,1]^2 \to [0,1]$, called a *graphon* (Lovász and Szegedy [2006]; see Section 7). These limit objects can be considered as weighted graphs with an underlying set of continuum cardinality. (If you wish, you can also think of these graphons as unweighted graphs on a non-standard model of the unit interval, where $W(x, y)$ is the density of edges between an infinitesimal neighborhood of x and an infinitesimal neighborhood of y; this approach will be explained in Section 11.3.2). Random graphs with edge density $1/2$ converge to the

identically 1/2 function (have a look at the two squares on the left of Figure 1.5). Figure 1.7 illustrates that the sequence of half-graphs (discussed in Section 1.5.2) converges to a limit (the function $W(x, y) = \mathbb{1}(y \geq x + 1/2 \text{ or } x \geq y + 1/2)$. It has been observed and used before (see e.g. Sidorenko [1991]) that such functions can be used as generalizations of graphs, and this gives certain arguments a greater analytic flexibility.

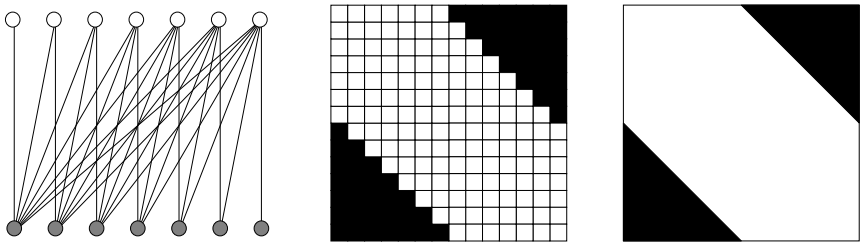


FIGURE 1.7. A half-graph, its pixel picture, and the limit function

Let us describe another example here (more to follow in Section 11.4.2). The picture on the left side of Figure 1.8 is the adjacency matrix of a graph G with 100 nodes, where the 1's are represented by black squares and the 0's, by white squares. The graph itself is constructed by a simple randomized growing rule: Starting with a single node, we create a new node, and connect every pair of nonadjacent nodes with probability $1/n$, where n is the current number of nodes. (This construction will be discussed in detail in Section 11.4.2.)

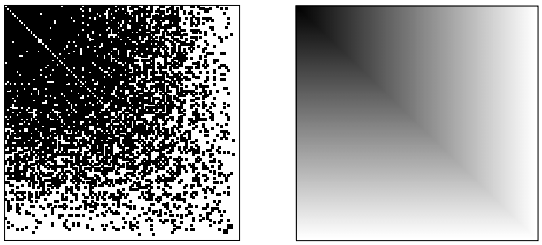


FIGURE 1.8. A randomly grown uniform attachment graph with 100 nodes, and a (continuous) function approximating it

The picture on the right side is a grayscale image of the function $U(x, y) = 1 - \max(x, y)$. (Recall that the origin is in the upper left corner!) The similarity with the picture on the left is apparent, and suggests that the limit of the graph sequence on the left is this function. This turns out to be the case in a well defined sense. It follows that to approximately compute various parameters of the graph on the left side, we can compute related parameters of the function on the right side. For example, the triangle density of the graph on the left tends (as $n \to \infty$) to the integral

$$\int_{[0,1]^3} U(x, y)U(y, z)U(z, x)\, dx\, dy\, dz \tag{1.3}$$

(the evaluation of this integral is a boring but easy task). It is easy to see how to generalize this formula to express the limiting density of any fixed graph F.

We hope that the examples above provide motivation for the following fact, which is one of the key results to be discussed in the book (Theorem 11.21):

The limit of any convergent graph sequence can be represented by a graphon, in the sense that the limiting density of any fixed simple graph F is given by an integral of the type (1.3).

Of course, a graphon can be infinitely complicated; but in many cases, limits of growing graph sequences have a limit graphon that is a continuous function described by a simple formula (see some further examples in Section 11.4.2). Such a limit graphon provides a very useful approximation of a large dense graph.

Graphons can be considered as generalizations of graphs, and this way of looking at them is very fruitful. In fact, many results can be stated and proved for graphons in a more natural and cleaner way. In particular, regularity lemmas can be extended to graphons, where we will see that they are statements about approximating general measurable functions by stepfunctions. Approximating graphs by multitype quasirandom graphs is as basic a tool in graph theory as approximating functions by stepfunctions is in analysis.

Remark 1.4. Much of this book is about finite, countable and uncountable graphs and connections between them. There are two technical limitations of measure theory that we have to work our way around. (a) *One cannot construct more than countably many independent random variables* (in a nontrivial way, neither of them concentrated on a single value). This is the reason while we cannot define a random graph on an uncountable set like $[0, 1]$, only on finite and countable subsets of it. (b) *There is no uniform distribution on a countable set* (while there is one on every finite set and then again on sets with continuum cardinality like $[0, 1]$). This limitation is connected to the fact that the limit objects for convergent graph sequences will be graphons (which could be considered as graphs defined on a continuum) rather than graphs on a countable set as one would first expect.

I want to emphasize that these difficulties are not just annoying technicalities: they reflect the fact, for example, that the limit object of a convergence graph sequence carries a lot more information than what could be squeezed into a countable graph. Both measure theory and combinatorics force us into the same realm.

1.6. How to run algorithms on them?

1.6.1. Parameter estimation. What can we learn about a huge graph G from sampling? There are several related questions here, depending on what we need as a result. The easiest setup is when we want to compute a numerical parameter of the graph; say, how large is the maximum cut, or what fraction of the triples induce a triangle. We call this problem *parameter estimation*. Most of the time we normalize the parameter to be between 0 and 1. Since, as discussed above, we get information about the graph through random sampling, any answer we can possibly compute will, with some probability, be in error. So we will have to specify an error parameter $\varepsilon > 0$, and will have to accept an answer which, with probability at least $1 - \varepsilon$, will be closer than ε to the true value of the parameter.

An easy example is to estimate the triangle density (number of triangles divided by $\binom{n}{3}$). A trivial algorithm is to pick many random triples of nodes independently,

and count how many of them form triangles in the graph. Elementary statistics tells us that if we sample $O(\varepsilon^{-2}|\log\varepsilon|)$ triples, then with probability at least $1-\varepsilon$, our estimate will be closer than ε to the truth.

A much more interesting and difficult example is that of estimating the density a of the maximum cut (its size divided by $\binom{n}{2}$) in a graph G. One thing we can try is to choose N random nodes (where N depends on the error bound ε), and compute the density X of the maximum cut in the subgraph H they induce. Is X a good estimate for a?

The inequality $X \geq a-\varepsilon$ (for every $\varepsilon > 0$ if N is large enough, with high probability) is relatively easy to prove. The graph G has a cut C with density a, and this cut provides a cut C' in the random induced subgraph H. It is easy to see that the density of C' is the same as the density a of C in expectation, and it takes some routine computation in probability theory to show that it is highly concentrated around this value. The density X of the largest cut in H is at least the density of C', and so with high probability it is at least $a-\varepsilon$ (Figure 1.9).

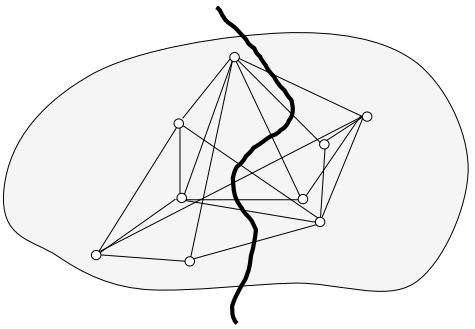


FIGURE 1.9. A dense cut in the large graph gives a dense cut in the sample.

The reverse inequality is much more difficult to prove, at least from scratch, and in fact it is rather surprising. We can phrase the question like this: Suppose that most random induced subgraphs H on N nodes have a cut that is denser than b. Does it follow that G has a cut that is denser than $b-\varepsilon$? It is not clear why this should be so: why should these cuts in these small subgraphs "line up" to give a dense cut in G?

We will see that it does follow that the estimate is correct, once N is large enough (about $\varepsilon^{-4}|\log\varepsilon|$). In fact, one can give general necessary and sufficient conditions under which parameters can be estimated by sampling, as we will see in Section 15.1.

1.6.2. Property testing. A more complicated issue is *property testing*: we want to determine whether the graph has some given property, for example, can it be decomposed into two connected components of equal size, is it planar, or does it contain any triangle. We could consider this as a 0-1 valued parameter, but computing this parameter approximately would not make sense (or rather, it would be requiring too much, since this would be equivalent to exact computation).

A good way of posing this problem was developed by Rubinfeld and Sudan [1996] and Goldreich, Goldwasser and Ron [1998]. In the slightly different context of "additive approximation", closely related problems were studied by Arora, Karger

and Karpinski [1995] (see e.g. Fischer [2001] for a survey and the volume edited by Goldreich [2010] for a collection of more recent surveys).

This approach acknowledges that any answer is only approximate. Suppose that we want to test for a property $\mathcal{P}$, and we get information about the graph by taking a bounded size random sample of the nodes, and inspecting the subgraph induced by them. We interpret the answer of the algorithm as follows: If it concludes that the graph has property $\mathcal{P}$, this means that we can change εn^2 edges so that we get a graph with property $\mathcal{P}$; if it concludes that the graph does not have property $\mathcal{P}$, this means that we can change εn^2 edges so that we get a graph without property $\mathcal{P}$.

Again, we have to specify an error parameter $\varepsilon > 0$ in advance, and will have to accept an answer which may be wrong with probability ε, and even if it is "right", it only means that we can change εn^2 edges in the graph so that the answer becomes correct.

Sometimes we can do better and eliminate either false positives or false negatives. As an example, let us try to test whether a given (dense) graph contains a triangle. We take a sample of size $f(\varepsilon)$ (the best function f which is known to work is outrageously large, but let's not worry about this), and check whether they contain a triangle. If they do, then we know that the graph has a triangle. If they don't, then one can prove (see Section 15.3) that with high probability, we can delete εn^2 edges from the graph so that no triangle remains.

Remark 1.5. We will not be concerned with the sample size as a function of the error bound ε. Sometimes it is polynomial (as in the examples above), but in other cases one uses the Regularity Lemma, which forces tower-size samples, making the algorithms of theoretical interest only. Goldreich [2010], in his survey of property testing, emphasizes the importance of testing with samples of manageable size, and I could not agree more; but this book, being about limit theory, does not address this issue.

Another caveat: Many extensions deal with testing models where we are allowed to sample more than a constant number of nodes of the large graph G. For this, we have to take the number of nodes into account, but usually it is enough to know the order of magnitude of the number of nodes, which in practical situations is easy to do. We do not discuss these important methods in our book.

1.6.3. Computation of a structure. Perhaps the most complex algorithmic task is the *computation of a structure*, where the structure is of size comparable with the graph itself. For example, we want to find a perfect matching in the graph, or a maximum cut (not just its density, but the cut itself), or a regularity partition in a huge dense graph. The conceptual difficulty is that the output of the algorithm is too large to be explicitly produced. What we can do is to carry out some preprocessing whose result can be stored (e.g., label a bounded number of nodes or edges), and give an algorithm which, for given input node or nodes, determines the local part of the structure we are looking for. Usually, this algorithm returns the "status" of a node or edge in the output structure (for example, whether the given edge belongs to matching, or which side of the cut the given node belongs to).

As an example, we will describe in Section 15.4.3 how to compute a maximum cut. We can access the graph by taking a bounded size sample of the nodes, and inspect the subgraph induced by them. For a given $\varepsilon > 0$, we precompute a

"representative set" (see next section) together with a bipartition of this set. In addition, we describe a "Placing Algorithm" which has an arbitrary node v as its input, and tells us on which side of the cut v is located. This Placing Algorithm can be called any number of times with different nodes v, and the answers it gives should be consistent with an approximately maximum cut. For example, calling this algorithm many times, we can estimate the density of the maximum cut (but this can be done in an easier way, as we have seen).

The parameter ε is an error bound: the cut computed may be off the true maximum cut by εn^2 edges, the precomputation may be wrong with probability at most ε, and for each query, the answer may be in error with probability at most ε.

1.6.4. Representative set. Szemerédi partitions are closely related to the main ingredient in these algorithms, namely a "representative set". We want to select a (fairly large, but bounded size) subset R of the nodes such that every node is "similar" to one of the nodes in R. To be economical, we don't want to include similar points in R.

We must start with defining what "similar" means; we will do so by defining a "similarity distance" between two nodes of a graph. A first idea would be to use their distance in the graph (the length of the shortest path connecting them). However, this measures something else (the prime minister and the doorman in his office know each other, but their positions in the society are certainly not similar).

We could try considering two nodes similar, if their neighborhoods differ by little. This is certainly a reasonable thing to do, but it is too restrictive for our purposes. For example, if we consider a random graph on n nodes with edge density $1/2$, then the neighborhoods of any two nodes are very different (they have about $n/2$ elements and overlap in about $n/4$), but all nodes of a random graph are alike, so we would like them to be close in the similarity distance.

It turns out (somewhat surprisingly) that it suffices to consider second neighborhoods: we consider two nodes s and t similar, if for most other nodes v, the number of paths of length two from s to v is about the same as the number of paths of length two from t to v. The similarity distance defined this way (for the exact definition, see Section 15.4.1) has many nice properties:

- The similarity distance can be computed by sampling.
- For every $\varepsilon > 0$, every graph has a "representative set" R of nodes, whose size depends on ε only; nodes in this set are at least ε apart, and almost every node is at a distance less than ε from the representative set.
- The representative set can be computed by sampling.
- Borrowing a phrase from geometry, we define the *Voronoi cell of a node* v of the representative set R as the set of all nodes in the whole graph that are closer to v than to any other node of R. The Voronoi cells of the representative set give a Weak Regularity Partition, and vice versa, every Weak Regularity Partition, after deletion of a fraction of ε of the nodes, consists of sets with small diameter in the similarity distance.

The key to many structural computational problems is that first a representative set is computed, and then the status of any node or edge can be computed using the representative set. For example, if we want to compute a Weak Regularity Partition, we compute a representative set, which we consider as a set of representative nodes of the partition classes, which are the Voronoi cells of the nodes. We

cannot compute all the Voronoi cells; but if we want to know which class (cell) does a given node belong to, all we need to do is to compute its distance to the nodes in R.

1.7. Bounded degree graphs

Let us discuss briefly how, and to what degree, the above considerations carry over to graphs with bounded degree. (We are doing injustice here to a rich and very active research area; I hope some of this will be rectified in Part 4 of the book. One of the reasons is that the technicalities in the bounded degree case are deeper, and so it is more difficult to state key results, even informally.)

Sampling. In the case of graphs with bounded degree, the subgraph sampling method gives a trivial result: the sampled subgraph will almost certainly be edgeless. Probably the most natural way to fix this is to consider *neighborhood sampling* (Figure 1.1). Let $\mathcal{G}_D$ denote the class of finite graphs with all degrees bounded by D. For $G \in \mathcal{G}_D$, select a random node and explore its neighborhood to a given depth r. This provides a probability distribution $\rho_{G,r}$ on graphs in $\mathcal{G}_D$, with a specified root node, such that all nodes are at distance at most r from the root. We will briefly refer to these rooted graphs as r-balls. Note that the number of possible r-balls is finite if D and r are fixed.

The situation for bounded degree graphs is, however, less satisfactory than for dense graphs, for two reasons. First, a full characterization of what distributions of r-balls the neighborhood sampling procedure can result in is not known (cf. Conjecture 19.8). Second, neighborhood sampling misses some important global properties of the graph, like expansion. In Section 19.2 we will introduce a notion of convergence, called local-global, which is better from this point of view, but it is not based on any implementable sampling procedure.

This suggests looking at further possibilities. Suppose, for example, that instead of exploring the neighborhood of a single random node, we could select two (or more) random nodes and determine simple quantities associated with pairs of nodes, like pairwise distances, maximum flow, electrical resistance, hitting times of random walks (studies of this nature have been performed, for example, on the internet, see e.g. Kallus, Hága, Mátray, Vattay and Laki [2011]). What information can be gained by such tests? Is there a "complete" set of tests that would give enough information to determine the global structure of the graph to a reasonable accuracy? Such questions could lead to different theories of large graphs and their limit objects; at this time, however, they are unexplored.

Remark 1.6. It is interesting to note that our two sampling methods correspond to the two basic data structures for graph algorithms, adjacency matrix and neighborhood lists. To be more specific, both methods assume that we can choose a uniformly distributed random node, and repeat this a constant number of times. In subgraph sampling, we must be able to determine whether two given nodes are adjacent or not. For a graph that is explicitly given, this is easy if the graph is given by its adjacency matrix. For neighborhood sampling, we have to be able to find all neighbors of a given node. This is easy if the graph is given by neighborhood lists. It would be very time consuming to perform these sampling operations on a graph given by the wrong data structure.

Sampling distance. The construction of the sampling distance can be carried over to graphs with bounded degree, by replacing in (1.2) the sampling distributions $\sigma_{G,k}$ by the neighborhood distributions $\rho_{G,k}$. We must point out, however, that it seems to be difficult to define a notion of distance between two graphs with bounded degree (in analogy with the cut distance) that would reflect global similarity.

Regularity Lemma. This is one of the big unsolved problems for graphs with bounded degree. If we consider regularity lemmas as providing "approximation by the smaller", then there is a simple non-constructive result (Proposition 19.10), which should be proved in a constructive way to be really useful. One can start at many other facets of the Regularity Lemma, but a satisfactory version of bounded degree graphs has turned out most elusive.

Convergence. The notion of a convergent sequence of bounded degree graphs was in fact the first among such convergence notions, introduced by Benjamini and Schramm [2001], motivated in part by earlier work of Aldous [1998]. Our discussion of local convergence of dense graphs above, based on the convergence of the distribution of samples, was modeled on the Benjamini–Schramm definition of convergence of bounded degree graphs.

There are, however, good reasons to try to strengthen this notion. Unlike in the dense case, neighborhood sampling cannot distinguish between bipartite graphs and graphs that are far from being bipartite, cannot estimate the maximum cut etc., which means that locally convergent graph sequences must lose this information in the limit. We will introduce and study a stronger notion of convergence, which we call *local-global*, which passes on these properties and parameters to the limit. However, we don't know if there is any natural and practical algorithmic setup that would correspond to local-global convergence.

Limit objects. For bounded degree graphs, Benjamini and Schramm provide a notion of a limit object (see Section 18). The Benjamini–Schramm limit object can be described as a distribution on rooted countable graphs with a special property called "involution invariance".

Another way of describing a limit object is a "graphing". In a sense, this latter object is what we expect: a bounded degree graph on an infinite (typically uncountable) set, with appropriate measurability and measure preserving conditions. This construction was folklore in an informal way; the first exact statements were published by Aldous and Lyons [2007] and Elek [2007a].

Graphings were invented by group theorists. The idea is to consider a finitely generated group acting on a probability space (for example, rotations of a circle by integer multiples of a given angle). One can construct a graph on the underlying space, by connecting each point to its images under the generators of the group. This construction gives a graph with bounded degree (the set of points is typically of continuum cardinality). It is a beautiful fact that

graphings, representing groups this way, are just right to describe the limit objects of convergent graph sequences with bounded degree.

Depending on personal taste, a graphing may be considered more complicated or less complicated than an involution-invariant random countable rooted graph. But graphings have an important advantage: they can express a richer structure, the limits of graph sequences convergent in the local-global sense.

Algorithms. Here is finally an area where the study of bounded degree graphs can be considered at least as advanced as the study of dense graphs. Let us discuss the task of computing a structure.

Selecting random nodes and exploring their neighborhoods, we see (with high probability) disjoint parts of the graph, and so there is no method to build up a global structure. Still, very nontrivial algorithms can be designed in this model. For example, in Section 22.3.1 we describe an algorithm due to Nguyen and Onak [2008], that constructs an almost maximum matching. The way the output can be described is similar to how the output of a maximum cut algorithm was described in the dense setting: for any node we can tell which other node it is matched to, inspecting a bounded neighborhood only; these assignments will be consistent throughout the graph; and the difference in size from the true maximum matching is only εn, where $\varepsilon > 0$ is an error bound and n is the number of nodes.

There is an equivalent way to describe such algorithms, which may be easier to follow, and this is the model of *distributed computing* (going back to the 1980's). In this case, an agent (or processor) is sitting at each node of the graph, and they cooperate in exploring various properties of it. They can only communicate along the edges. In the case we are interested in (which is in a sense extreme), they are restricted to exchange a bounded number of bits (where the bound may depend on the degree D, on an error bound ε, and of course on the task they are performing, but not on the number of nodes). In some other versions of the model (cellular automata), the amount of communication is not restricted, but the computing power of the agents is. Note that in our model communication between the agents is restricted to a bounded number of bits, and hence they may be assumed to be very stupid, even finite automata.

There is a large literature on distributed computing, both from the practical and theoretical aspect. We will not be able to cover this; we will restrict ourselves to the discussion of the strong connection of this computation model with our approach to large graphs and graph limits.

CHAPTER 2

Large graphs in mathematics and physics

The algorithmic treatment of very large networks is not the only area where the notions of very large graphs and their limits can be applied successfully. Many of the problems and methods in graph limit theory come from extremal graph theory or from statistical physics. Let us give s very brief introduction to these theories.

2.1. Extremal graph theory

Extremal graph theory is one of the oldest areas of graph theory; it has some elegant general results, but also many elementary extremal problems that are still unsolved. Graph limit theory (mostly the related theory of flag algebras by Razborov) has provided powerful tools for the solution of some of these problems. Furthermore, graph limits, along with the algebraic tools that will be introduced soon, will enable us to formulate and (at least partially) answer some very general questions in extremal graph theory (similarly to the general questions for very large graphs posed in the previous chapter).

2.1.1. Edges vs. triangles. Perhaps the first result in extremal graph theory was found by Mantel [1907]. This says that if a graph on n nodes has more than $n^2/4$ edges, then it contains a triangle. Another way of saying this is that if we want to squeeze in the largest number of edges without creating a triangle, then we should split the nodes into two equal classes (if n is odd, then their sizes differ by 1) and insert all edges between the two classes. As another early example, Erdős [1938] proved a bound on the number of edges in a C_4-free bipartite graph (see (2.9) below), as a lemma in a paper about number theory.

Mantel's result is a special case of Turán's Theorem [1941], which is often considered as the work that started the systematic development of extremal graph theory. Turán solved the generalization of Mantel's problem for any complete graph in place of the triangle. We define the *Turán graph* $T(n,r)$ $(1 \le r \le n)$ as follows: we partition $[n]$ into r classes as equitably as possible, and connect two nodes if and only if they belong to different classes. Since we are interested in large n and fixed r, the complication that the classes cannot be exactly equal in size (which causes the formula for the number of edges of $T(n,r)$ to be a bit ugly) should not worry us. It will be enough to know that the number of edges in a Turán graph is

$$\mathsf{e}\big(T(n,r)\big) \sim \binom{r}{2}\Big(\frac{n}{r}\Big)^2,$$

and in terms of the homomorphism densities defined in the previous chapter in (1.1), we have $t\big(K_2, T(n,r)\big) \sim 1 - \frac{1}{r}$. For the triangle density we have the similar formula $t\big(K_3, T(n,r)\big) \sim (1-\frac{1}{r})(1-\frac{2}{r})$.

THEOREM 2.1 **(Turán's Theorem)**. *Among all graphs on n nodes containing no complete k-graph, the Turán graph $T(n, k-1)$ has the maximum number of edges.*

Let us return to triangles, however, and ask for not just their existence, but for their number, when the number of edges is known. All of a sudden, we get to a rather difficult problem with some unexpected complications (which makes the subject fascinating). It is really difficult to think of a simpler question about small subgraphs of a large graph!

Since we are interested in large n, it is natural to normalize, and use homomorphism densities. The Mantel–Turán Theorem says, in this language, that

$$t(K_2, G) > 1/2 \Rightarrow t(K_3, G) > 0. \tag{2.1}$$

Every graph G produces a pair of numbers $\big(t(K_2, G), t(K_3, G)\big)$ this way, which we can consider as a point in the plane. If we plot this point for every graph G, we get a picture as in Figure 2.1(a). To be more precise, we get a countably infinite set of points; the figure shows its closure, which we denote by $D_{2,3}$. (Another motivation for introducing convergent graph sequences and their limit objects: they give a meaning to all points of this figure.)

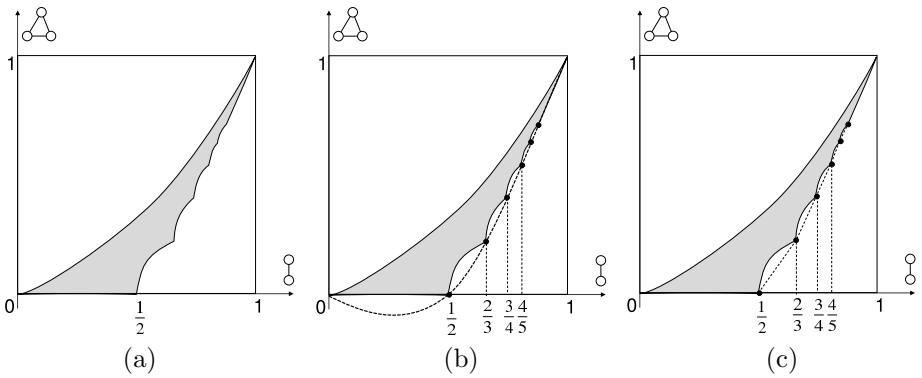


FIGURE 2.1. (a) The closure $D_{2,3}$ of the set of pairs of edge density and triangle density. (b) Goodman's bound. (c) Bollobás's' bound. The picture is a little distorted in order to show its special features better.

Some features of this picture are easy to explain. The lower edge means that there are triangle-free graphs with edge density up to 1/2, and the Mantel–Turán Theorem says that for larger edge density, the triangle density must be positive. A lower bound for the triangle density was proved by Goodman [1959],

$$t(K_3, G) \geq t(K_2, G)(2t(K_2, G) - 1), \tag{2.2}$$

which corresponds to the parabola shown in 2.1(b).

The upper boundary curve turns out to be given by the equation $y = x^{3/2}$, which is a very special case of the Kruskal–Katona Theorem in extremal hypergraph theory (the full theorem gives the precise value, not just asymptotics, and concerns uniform hypergraphs, not just graphs). In other words, this says that

$$t(K_3, G) \leq t(K_2, G)^{3/2}. \tag{2.3}$$

Both (2.2) and (2.3) are sharp in a sense: Goodman's Theorem is sharp if the edge density is of the form $1/2, 2/3, 3/4, \ldots$ (Turán graphs give equality). In this form of the Kruskal–Katona Theorem equality is not attained except at the points $(0,0)$ and $(1,1)$, but for every point $(x, x^{3/2}$ of the upper boundary curve there are points representing a graph arbitrarily close (just use graphs consisting of a complete graph and isolated nodes).

From our perspective, there is nothing to improve on the upper bound, but can we get arbitrarily close to the lower bound between two special edge density values $1 - 1/k$? Surprisingly, the answer is no. Bollobás [1976] proved in 1976 that the triangle density for a graph with edge density $x \in (1 - \frac{1}{k-1}, 1 - \frac{1}{k})$ is not only above the parabola, but also above the chord of the parabola connecting the special points corresponding to $T(n, k-1)$ and $T(n,k)$.

Lovász and Simonovits [1976, 1983] formulated a conjecture about the exact bounding curve, and proved it in very small neighborhoods of the special edge density values above. One way to state this is that the minimum number of triangles is attained by a complete k-partite graph with unequal color classes. The sizes of the color classes can be determined by solving an optimization problem, which leads to a cubic *concave* curve connecting the two special points. This conjecture turned out quite hard. Lovász and Simonovits proved it in the special case when the edge density x was close to one of the endpoints of the interval. Fisher [1989] proved the conjecture for the first interval $(1/2, 2/3)$. After quite a while, Razborov [2007, 2008] proved the general conjecture. His work was extended by Nikiforov [2011] to bounding the number of complete 4-graphs, and by Reiher [2012] to all complete graphs.

So we know what the lower and upper bounding curves are. Luckily, math plays no further tricks on us: it is easy to see that for every point between the two curves there are points representing graphs arbitrarily close.

I dwelt quite long on this very simple special problem not only to show how complicated it gets (and yet solvable), but also because Razborov's methods for the solution fit quite well in the framework developed in this book, and they will be presented in Chapter 16.

2.1.2. A sampler of classical results. Let us start with some remarks to simplify and to some degree unify the statements of these results. Every algebraic inequality between subgraph densities can be "linearized", using the following multiplicativity of $t(., G)$:

$$t(F_1F_2, G) = t(F_1, G)t(F_2, G), \tag{2.4}$$

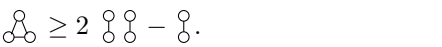
where F_1F_2 denotes the disjoint union of F_1 and F_2. (This property will play a very important role in the sequel, but right now it is just a convenient simplification.) For example, we can replace (2.2) by

$$t(K_3, G) \geq 2t(K_2K_2, G) - t(K_2, G). \tag{2.5}$$

We can make the statements (and their proofs, as we will see below) more transparent by two further tricks: first, if a linear inequality between the densities of certain subgraphs $F_1, \ldots, F_k$ holds for all graphs, then we write it as an inequality between $F_1, \ldots, F_k$; and for specific small graphs F_i, we use little pictograms. Goodman's Inequality (2.2) can be expressed as follows:

$$K_3 \geq 2{K_2}^2 - K_2 \tag{2.6}$$

or

(2.7) 

The Kruskal–Katona Theorem for triangles is:

(2.8) 

Let us describe some further classical results. Instead of counting complete graphs, we can consider the density of some other graph F in G. Erdős proved the inequality

$$t(C_4, G) \geq t(K_2, G)^4, \tag{2.9}$$

or in pictograms

(2.10) 

Graphs with asymptotic equality here are quasirandom graphs (Section 1.4.2).

Bounding from below the homomorphism density of paths is a more difficult question, but it turns out to be equivalent to theorems of Mulholland and Smith [1959], Blakley and Roy [1965], and London [1966] in matrix theory (applied to the adjacency matrix). If P_k denotes the path with k nodes, then for all $k \geq 2$,

$$t(P_k, G) \geq t(K_2, G)^{k-1}. \tag{2.11}$$

Regular graphs give equality here. The first nontrivial case of inequality (2.11) is

(2.12) 

Translating to homomorphisms, this means that

$$\mathsf{v}(G)\mathsf{hom}(P_3, G) \geq \mathsf{hom}(K_2, G)^2.$$

If we count the homomorphisms on the left side by the image of the middle node, we see that it is the sum of the squared degrees of G. Since $\mathsf{hom}(K_2, G) = 2\mathsf{e}(G)$ is the sum of the degrees, this inequality is just the inequality between arithmetic and quadratic means, applied to the sequence of degrees.

Bounding the P_3-density from above in terms of the edge density is more difficult, but it was solved by Ahlswede and Katona [1978]; we formulate this as Exercise 2.4 below.

The next case of inequality (2.11) is

(2.13) 

and this is already quite hard, although short proofs with a tricky application of the Cauchy–Schwarz inequality are known.

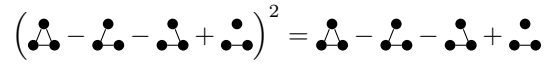
In Chapter 16 we will return to the question of how far the application of such elementary inequalities takes us in proving inequalities between subgraph densities.

2.1.3. An algebraic "proof" of an extremal theorem. We illustrate the use of the formalism with pictograms for an algebraic proof of Goodman's Inequality 2.2. This will motivate a basic tool to be introduced in Chapter 6, namely graph algebras.

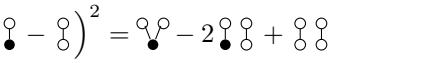
To describe this proof, we extend the pictogram formalism from Section 2.1.2. If we fill a node, this indicates that this node is *labeled* . We should write the label on the node, but to keep the picture simple, let us agree that the labels are $1, 2, \ldots,$

starting from the lower left corner, and going counterclockwise. (It does not really matter.)

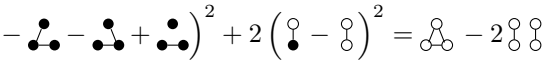
The role of the labels is that when taking a "product" of two graphs, we take the disjoint union, but identify nodes with the same label. With this convention, it is easy to check that



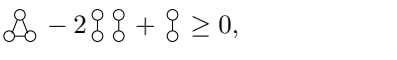
(this combination is "idempotent") and



Forgetting the labels, adding up, and deleting isolated nodes, we get



So the right side is a sum of squares, which implies that it is nonnegative:



which is just (2.7).

Is this a valid argument? It turns out that it is, and the method can be formalized using the notion of graph algebras. These will be very useful tools in the proofs of characterization theorems of homomorphism functions, and also in some other studies of graph parameters.

2.1.4. General results. Moving from special extremal graph problems to the more general, let us describe some quite general results about extremal graphs, which were obtained quite a long time ago in several papers of Erdős, Stone and Simonovits [1946, 1966, 1968]. We exclude an arbitrary graph L as subgraph of a simple graph G, and want to determine the maximum number of edges of G, given the number of nodes n. Turán's Theorem 2.1 is a special case when L is a complete graph. It turns out that the key quantity that governs the answer is the chromatic number $r = \chi(G)$.

The Turán graph $T(n, r-1)$ is certainly one of the candidates for the extremal graph, since it cannot contain any graph as a subgraph that has chromatic number r. For certain excluded graphs L it is easy to construct examples that have slightly more edges than this Turán graph; however, the gain is negligible: for every graph G on n nodes that does not contain L as a subgraph, we have

$$\mathsf{e}(G) \le (1+o(1))\mathsf{e}(T(n,r-1)) = \Bigl(1 - \frac{1}{r-1} + o(1)\Bigr)\binom{n}{2}. \tag{2.14}$$

There is also a "stability" result: For every $\varepsilon > 0$ there is an $\varepsilon' > 0$ (depending on L and ε, but not on G) such that if G is a graph not containing L with at least $\bigl(1 - 1/(r-1) - \varepsilon'\bigr)\binom{n}{2}$ edges, then we can change at most $\varepsilon\binom{n}{2}$ edges of G to get a Turán graph $T(n, r-1)$.

We will see that graph limit theory gives very short and elegant proofs for these facts. The idea that extremal graph problems have "continuous versions" (in a sense quite similar to our use of graphons), which are often cleaner and easier to handle, goes back to around 1980, when Katona [1978, 1980, 1985] and Sidorenko [1980, 1982] used this method to generalize graph and hypergraph problems, and also to give applications in probability theory.

REMARK 2.2. If $r = 2$ (which means that L is bipartite), then the main term in (2.14) disappears, and all we get is that the number of edges is $o(n^2)$. Of course, one would like to know the precise order of magnitude of the best upper bound. This is known in several cases (e.g., small complete bipartite graphs and cycles), but in general it seems to be a difficult unsolved problem. The extremal graphs in this case are sparse, and quite complex: for example, C_4-free graphs with maximum edge density are constructed from finite projective planes. Extremal problems for graphs with excluded bipartite graphs do not seem to fit in with the framework developed in this book, but perhaps they can serve as motivation for extending it to sparser graphs.

2.1.5. General questions. We have brought up the idea of introducing graphons (graph limits) in Section 1.5.3 motivated by the goal to approximate very large networks by simpler analytic objects. We have seen that graphons provide cleaner formulations, with no error terms, of some results in graph theory (for example, about quasirandom graphs). We will see in Section 16.7 that extremal graph theory provides another, also quite compelling motivation: Graphons provide a way to state, in an exact way, general questions about the nature of extremal graphs, and also help answering them, at least in some cases. (They have similar uses in the theory of computing; cf. Chapter 15).

Which inequalities between subgraph densities are valid? Given a linear inequality between subgraph densities (like (2.7) above), is it valid for all graphs G? Hatami and Norine [2011] proved recently that this question is algorithmically undecidable. We will describe the proof of this fundamental result in Section 16.6.1. On the other hand, it follows from the results of Lovász and Szegedy [2012a] that if we allow an arbitrarily small "slack", then it becomes decidable (see Section 16.6.2).

Can all linear inequalities between subgraph densities be proved using just Cauchy–Schwarz? We described above a proof of the simple inequality (2.12) using the inequality between arithmetic and quadratic means, or equivalently, the Cauchy–Schwarz Inequality. Many other extremal problems can be proved by using the Cauchy–Schwarz Inequality (often repeatedly and in nontrivial ways). Exercise 2.5 shows that Goodman's Inequality can also be proved by this method. How general a tool is the Cauchy–Schwarz Inequality in this context?

Using the notions of graphons and graph algebras we will be able to give an exact formulation of this question. It will turn out that the answer is negative (Hatami and Norine [2011], Section 16.6.1), but it becomes positive if we allow an arbitrarily small error (Lovász and Szegedy [2012a], Section 16.6.2).

Is there always an extremal graph? Let us consider extremal problems of the form "maximize a linear combination of subgraph densities, subject to fixing other such combinations". For example, "maximize the triangle density subject to a given edge density" (the answer is given by the first nontrivial case of the Kruskal–Katona Theorem (2.8)).

To motivate our approach, consider the following two optimization problems.

CLASSICAL OPTIMIZATION PROBLEM. Find the minimum of $x^3 - 6x$ over all numbers $x \geq 0$.

GRAPH OPTIMIZATION PROBLEM. Find the minimum of $t(C_4, G)$ over all graphs G with $t(K_2, G) \geq 1/2$.

The solution of the classical optimization problem is of course $x = \sqrt{2}$. This means that it has no solution in rationals, but we can find rational numbers that are arbitrarily close to being optimal. If we want an exact solution, we have to go to the completion of the rationals, i.e., to the reals.

The graph optimization problem may take a bit more effort to solve, but (2.9) shows that if the edge density is 1/2, then the 4-cycle density is at least 1/16. With a little effort one can show that equality is never attained here. Furthermore, the 4-cycle-density gets arbitrarily close to 1/16 for appropriate families of graphs: the simplest example is a random graph with edge density 1/2 (cf. also Section 1.4.2).

The analogy with the classical optimization problem above suggests that we should try to enlarge the set of (finite) graphs with new objects so that the appropriate extension of our optimization problem has a solution among the new objects. Furthermore, we want that these new objects should be approximable by graphs, just like real numbers are approximable by rationals. As it turns out, graphons are just the right objects for this.

One can prove that there is always an extremal graphon, which then gives a "template" for asymptotically extremal graphs. This follows from another fact that can be considered one of the basic results treated in this book:

The space of graphons is compact in the cut-distance metric.

(This notion of distance was mentioned in Section 1.5.1, and will be defined in Chapter 8; the compactness of the graphon space will be proved in Section 9.3).

Which graphs are extremal? This is not a good question (every graph is extremal for some sufficiently complicated extremal graph problem), but replacing "graph" by "graphon" makes it mathematically meaningful. Every extremal graphon gives a "template" for asymptotically extremal graphs.

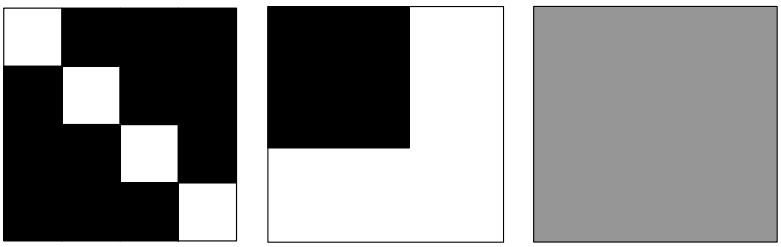


FIGURE 2.2. Templates for optimal solutions to some classical extremal graph results: (a) Turán's Theorem 2.1 and Goodman's Inequality (2.2); (b) the Kruskal–Katona Theorem (2.3); (c) Erdős's inequality (2.9)

In classical extremal graph results, these templates are quite simple (Figure 2.2). A natural guess would be that all templates have the form of a stepfunction, like the rightmost square in Figure 1.6. All of these are indeed templates for appropriate extremal problems, but they is not all the templates: we will see that the limit of half-graphs (the rightmost square in Figure 1.7) is also the template for the extremal graph of a quite simple extremal problem, and there are many other, more complicated, templates. We will prove several results about the structure of these extremal templates (Section 16.7), but no full characterization is known.

EXERCISE 2.3. Prove inequality (2.13)

EXERCISE 2.4. Let G be a simple graph with edge density $d = t(K_2, G)$. Prove that $t(P_3, G) \le \max(d^{3/2}, 1 - 2d + d^{3/2})$.

EXERCISE 2.5. Translate the "proof" of Goodman's Inequality 2.2 above into a valid proof using the Cauchy–Schwarz inequality twice.

2.2. Statistical physics

One area of research where graph homomorphisms play an important role, and the study of the asymptotic behavior of parameters when tending to infinity with the size of a graph is a main goal, is statistical physics. I am afraid this book will not do justice to this connection; my excuse is that statistical physics is such a large area, with so advanced special methods, that any reasonable treatment would double the size of the book. Nevertheless, I must give a very short introduction to the subject here.

To describe a basic model in statistical physics, suppose that we have a piece of a crystal, where the spin of every atom can point either up or down. We model this by an $n \times n$ grid $G = G_{n\times n}$ (for simplicity, in two dimensions). If we assign to every node of G (every "site") a "state", which can be UP or DOWN, we get a "configuration". The atoms are changing their spins randomly all the time, but not independently of each other: depending on the spins of adjacent atoms, one direction of the spin of an atom may be less likely then the other, or even entirely impossible. We would like to know how a typical configuration looks like: is it random-like as the first picture in Figure 2.3, is it homogeneous as the second (well, maybe with a few exceptions here and there), or is it structured in other ways, as the third?

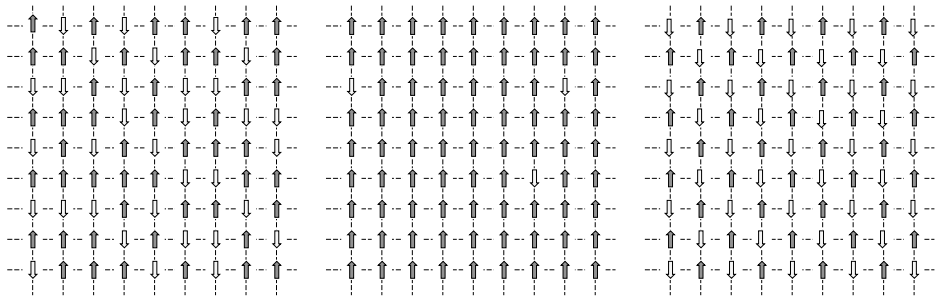


FIGURE 2.3. Three configurations of the Ising model.

Two atoms that are adjacent in the grid have an "interaction energy", which depends on their states. In the simplest version of the basic Ising model, the interaction energy is some number $-J$ if the atoms are in the same state, and J if they are not. The states of an atom can be described by the integers 1 and -1, and so a configuration is a mapping $\sigma : V(G) \to \{1, -1\}$. If σ_u denotes the state of atom u, then the total energy of a given configuration is

$$H(\sigma) = -\sum_{uv\in E(G)} J\sigma_u\sigma_v.$$

Basic physics (going back to Boltzmann) tells us that the system is more likely to be in states with low energy. In formula, the probability of a given configuration

is proportional to $e^{-H(\sigma)/T}$, where T is the temperature (from the point of view of the mathematician, just a parameter). Since probabilities must add up to 1, these values must be normalized:

$$\mathsf{P}(\sigma) = \frac{e^{-H(\sigma)/T}}{Z},$$

where the normalizing factor Z is called the *partition function* of the system (it is called a "function" because it depends on the temperature). This is perhaps the most important quantity to know, which contains implicitly many important physical parameters. The partition function is simple to describe:

$$Z = \sum_{\sigma} e^{-H(\sigma)/T} = \sum_{\sigma} \exp\Big(\frac{1}{T} \sum_{uv \in E(G)} J\sigma_u\sigma_v\Big),$$

but since the number of terms is enormous, partition functions can be very hard to compute or analyze.

The behavior of the system depends very much on the sign of J. If $J > 0$, then adjacent pairs that are in the same state contribute less to the total energy than those that are in different state, and so the configuration with the lowest energy is attained when all atoms are in the same state. The typical configuration of the system will be close to this, at least as long as the temperature T is small. This is called the *ferromagnetic* Ising model, because it gives an explanation how materials like iron get magnetized. If $J < 0$ (the antiferromagnetic case), then the behavior is different: the chessboard-like pattern minimizes the energy, and no magnetization occurs at any temperature.

One may notice that the temperature T emphasizes the difference between the energy of different configurations when $T \to 0$ (and de-emphasizes it when $T \to \infty$). In the limit when $T \to 0$, all the probability will be concentrated on the states with minimum energy, which are called *ground states*. In the simplest ferromagnetic Ising model, there are two ground states: either all atoms are in state UP, or all of them are in state DOWN. If the temperature increases, disordered states like the left picture in Figure 2.3 become more likely. The transition from the ordered state to the disordered may be gradual (in dimension 1), or it may happen suddenly at a given temperature (in dimensions 2 and higher, for large graphs G); this is called a *phase transition*. This leads us to one of the central problems in statistical physics; alas, we cannot go deeper into the discussion of this issue in our book.

To make the connection to graph homomorphisms, we generalize the Ising model a little. First, we replace the grid by an arbitrary graph G. (From the point of view of physics, other lattices, corresponding to crystals with other structure, are certainly natural. Other materials don't have a simple periodic crystal structure.) Second, we introduce a "magnetic field", which prefers one state over the other: in the simplest case it adds $-\sum_u h\sigma_u$ to the energy function, with some parameter h. Third, we consider not two, but q possible states for every atom, which we label by $1, 2, \ldots, q$ (unlike 1 and -1 before, these should not be considered as numbers: they are just labels). We have to specify an interaction energy J_{ij} for any two states i and j, and a magnetic field energy h_i for every state i. A configuration is now a map $\sigma : V(G) \to [q]$, and the energy of it is

$$H(\sigma) = -\sum_{v \in V(G)} h_{\sigma(v)} - \sum_{uv \in E(G)} J_{\sigma(u),\sigma(v)}.$$

The partition function is

$$Z = \sum_{\sigma\colon V(G)\to[q]} \exp\Big(-\frac{1}{T}\Big(\sum_{v\in V(G)} h_{\sigma(v)} + \sum_{uv\in E(G)} J_{\sigma(u),\sigma(v)}\Big)\Big).$$

We are almost at homomorphisms! For $i, j \in [q]$, let

$$\alpha_i = \exp\Big(-\frac{1}{T}h_i\Big), \qquad \text{and} \qquad \beta_{ij} = \exp\Big(-\frac{1}{T}J_{ij}\Big),$$

then the partition function can be expressed as

$$Z = \sum_{\sigma:V(G)\to[q]} \prod_{v\in V(G)} \alpha_{\sigma(v)} \prod_{uv\in E(G)} \beta_{\sigma(u)\sigma(v)}. \tag{2.15}$$

Consider the case when $\alpha_i = 1$ for all i, and β_{ij} is 0 or 1 (in the Ising model β_{ij} cannot be zero, but (2.15) allows this substitution). Then every term in (2.15) is either 0 or 1, and a term is 1 if and only if $\beta_{\sigma(u)\sigma(v)} = 1$ for every $uv \in E(G)$. Let us build a graph H with node set $V(H) = [q]$, in which $i, j \in [q]$ are adjacent if and only if $\beta_{ij} = 1$. Then a term in (2.15) is 1 if and only if σ is a homomorphism $G \to H$, and so the sum simply counts these homomorphisms, and gives the value $Z = \mathsf{hom}(G, H)$.

In the case of general values for the α and β, we can define a weighted graph H with nodeweights α_i and edgeweights β_{ij}. Formula (2.15) can then serve as the definition of $\mathsf{hom}(G, H)$, which will be very important for us.

We don't discuss the connections between statistical physics and graph theory (homomorphisms and limits) any further; for an introduction to the connections between statistical physics and graph theory, with more examples, see de la Harpe and Jones [1993].

EXERCISE 2.6. Define a model in statistical physics in which the ground state corresponds to the maximum cut of a graph.