# Combinatorial and computational aspects of graph packing and graph decomposition

*Raphael Yuster*
Department of Mathematics
University of Haifa
Haifa 31905, Israel *

**Abstract**

Packing and decomposition of combinatorial objects such as graphs, digraphs, and hypergraphs by smaller objects are central problems in combinatorics and combinatorial optimization. Their study combines probabilistic, combinatorial, and algebraic methods. In addition to being among the most fascinating purely combinatorial problems, they are often motivated by algorithmic applications. There are a considerable number of intriguing fundamental problems and results in this area, and the goal of this paper is to survey the state of the art.

## 1 Introduction

### 1.1 Background

A *vertex-packing* of a graph $G$ with a smaller (usually fixed) graph $H$ is a collection of pairwise vertex-disjoint (not necessarily induced) subgraphs of $G$ that are isomorphic to $H$. Vertex packing is sometimes called *$H$-matching*, generalizing the usual notion of matching which is just the case $H = K_2$. Similarly, an *edge-packing* of a graph $G$ with $H$ is a collection of pairwise edge-disjoint subgraphs of $G$ that are isomorphic to $H$. Edge packing will be called *$H$-packing* throughout this paper. The definitions of $H$-matching and $H$-packing are naturally extended to hypergraphs, where, in most cases, the hypergraphs are assumed to be $r$-uniform for some fixed $r$. The definitions of $H$-matching and $H$-packing are also naturally extended to vertex-colored graphs and edge-colored graphs, where the subgraphs must also be color-isomorphic to $H$.

Naturally, the goal in combinatorial packing theory is to find a packing with maximum cardinality. We therefore denote by $p_H(G)$ the maximum cardinality of an $H$-packing and by $m_H(G)$ the maximum cardinality of an $H$-matching. Notice that $m_{K_2}(G) = m(G)$ is just the matching number

*email: raphy@math.haifa.ac.il

of $G$, and that, trivially, $p_{K_2}(G) = e(G)$. It is easy to see that $m_H(G) \leq v(G)/v(H)$, and that $p_H(G) \leq e(G)/e(H)$. When equality holds for $m_H(G)$ we say that $G$ has an *$H$-factor* and when equality holds for $p_H(G)$ we say that $G$ has an *$H$-decomposition*.

The combinatorial study of packing problems has a long and rich history. In fact, already in 1847 Kirkman [49] gave nontrivial constructions showing that $K_n$ has a $K_3$-decomposition if and only if $n \equiv 1, 3 \bmod 6$. Some 100 year later, Tutte [77] gave nontrivial conditions that guarantee that a graph $G$ has a $K_2$-factor (a perfect matching; also sometimes called a 1-factor). But it was much later, dating back to the 70's, when modern tools from combinatorics and theoretical computer science were applied to these problems, resulting in a significant deepening of our understanding the complex world of graph packing problems. To date, there is already a vast amount of literature on these topics (see, e.g., [63]), and it would be hopeless to cover all of it. The goal of this survey is to focus on the major results and methods, as well as to exhibit some challenging open problems and conjectures. Whenever applicable, we will focus on both the combinatorial and computational aspects of the problems.

Packing and decomposition have many applications ranging from information theory to the design of efficient statistical experiments, as well as many others. Consider the following obvious, but illuminating, example. A $K_k$-packing of $K_n$ is simply an error-correcting code where each codeword has length $n$ and Hamming weight $k$, and the Hamming distance between any two codewords is at least $2(k-1)$. Thus, the code can correct $k-2$ errors. As a binary code with length $n$ and Hamming weight $k$ that corrects $2(k-1)$ errors can contain at most $(n(n-1))/(k(k-1))$ codewords, a $K_k$-decomposition of $K_n$ corresponds to an optimal code with these properties.

### 1.2 Overview of the survey

The survey is organized according to different topics in combinatorial packing theory, rather than by the tools used to prove the results, although, in many case, there is a correlation. In Section 2 we consider $H$-factors and, more generally, $H$-matchings. An important tool used in this case is the Regularity Lemma of Szemerédi [76]. In Section 3 we consider $H$-decompositions and, more generally, $H$-packings. Important tools used in this case are Wilson's Theorem [79], variants of the Rödl Nibble, as well as the Regularity Lemma and probabilistic methods. In Section 4 we consider packing problems in related combinatorial structures that include, in particular, random graphs, hypergraphs, and directed graphs. Open problems and conjectures are stated throughout the paper, in their appropriate sections.

### 1.3 Notations and conventions

As noted earlier, we use $m_H(G)$ and $p_H(G)$ for the $H$-matching and $H$-packing number of a graph $G$, respectively. We let $V(G)$ and $E(G)$ (or, simply, $V$ and $E$ when this is not ambiguous) denote the set

of vertices and edges of a graph $G$, respectively. We also denote $v(G) = |V(G)|$ and $e(G) = |E(G)|$. In many cases we will use $n$ and $m$ to denote $v(G)$ and $e(G)$, respectively, when this is not ambiguous. The *minimum degree* is denoted by $\delta(G)$ and the maximum degree by $\Delta(G)$. For additional standard graph theoretic notations we refer to [13] and to [27].

We say that a proof that shows that existence of a mathematical structure is *algorithmic* if the proof yields a polynomial time algorithm for constructing that structure.

# 2 $H$-factors and $H$-matchings

In this section we focus on $H$-factors and, more generally, on $H$-matchings. We start with the computational complexity of $H$-matching and $H$-factor recognition. We then turn to a few important combinatorial results that guarantee the existence of $K_k$-factors, $H$-factors, and $H$-matchings that are close to $H$-factors, in sufficiently dense graphs, as well as their algorithmic aspects.

## 2.1 Finding a large $H$-matching

An algorithm that finds a maximum matching in polynomial time was discovered by Edmonds [29]. His algorithm, and consequently most of the other known algorithms for this problem, are based on augmenting path techniques and are quite complicated. Currently, the fastest known algorithm for this problem, obtained by Micali and Vazirani, runs in $O(\sqrt{n} \cdot m)$ time [64]. For dense graphs, a slightly faster randomized algorithm was recently obtained by Mucha and Sankowski [66]. The running time of their algorithm is $O(n^\omega)$ where $\omega < 2.376$ is the *matrix multiplication exponent* [22]. The algorithm is based on a relatively simple and beautiful algorithm of Lovász [62], that computes, with high probability, the size of a maximum matching (without producing the edges, though). Lovász's algorithm is based on the observation that the rank of the symbolic skew-symmetric adjacency matrix (Tutte Matrix) of a graph is just twice the size of a maximum matching, and the fact that Gaussian elimination over a large (but not too large) finite field requires $O(n^\omega)$ time. Lovász's algorithm can also be implemented in RNC. It is still not known whether maximum matching can be solved in NC.

Things change dramatically when switching from $K_2$ to $K_3$. It is well-known that deciding whether a graph has a $K_3$-factor is NP-complete, as a special case of this problem is 3-Dimensional Matching (3DM). In fact, it is well-known [50] that:

**Theorem 2.1 ([50])** *Computing $m_H(G)$ is NP-hard, and the $H$-factor decision problem is NP-complete, for every graph $H$ that contains a connected component with at least three vertices.*

Can we reasonably approximate $m_H(G)$ quickly? A simple strategy is the greedy algorithm: simply find a maximal (with respect to containment) set of copies of $H$ in $G$. As $H$ is fixed, this can be done in polynomial time. The greedy algorithm has approximation ratio $v(H)$ and this is tight.

Indeed, every copy in the proposed solution intersects at most $v(H)$ copies of an optimal solution. Already for $H = K_3$ this yields an approximation ratio of 3. Can we do better? The following Theorem, which is a special case of an approximation algorithm of Hurkens and Schrijver [44] shows that we can do significantly better.

**Theorem 2.2 ([44])** *For every $\epsilon > 0$, there exists a polynomial time approximation algorithm for $m_{K_k}(G)$ whose approximation ratio is $(k/2 + \epsilon)$.*

**Proof (sketch).** The algorithm works as follows: given a collection of vertex-disjoint $K_k$ constructed so far, check whether there are $p \leq 1/\epsilon$ disjoint $k$-cliques that are not in the current collection that intersect at most $p - 1$ cliques that are in it. If this happens, swap the cliques to form a larger collection, and repeat; otherwise, terminate the search. ■

Kann [45] proved that computing $m_H(G)$ is APX-hard if $H$ has a component which contains at least three vertices. In other words, it is impossible to approximate the optimal solution within an arbitrary factor unless P = NP. More recently, it was shown that computing $m_{K_3}(G)$ is APX-complete already for graphs with bounded degree [17].

It is possible to decide if a graph has an $H$-factor in case $G$ has bounded tree-width (we refer the reader to [27] for the definition of this important class of graphs):

**Theorem 2.3** *For every fixed graph $H$ and every fixed positive integer $t$, there is a linear time algorithm for deciding whether a graph with tree-width at most $t$ has an $H$-factor.*

**Proof.** It is well-known that for every graph property that can be expressed in *monadic second-order logic* (MSOL), it is possible to decide if a given graph $G$ with bounded tree width has this property in polynomial (in fact, linear) time [24, 8]. Having an $H$-factor is expressible in MSOL. Here is an MSOL expression describing a $K_3$-factor: "There are sets of vertices $V_1$, $V_2$, $V_3$ and sets of edges $E_1$, $E_2$, $E_3$, so that $V = V_1 \cup V_2 \cup V_3$, $V_1 \cap V_2 = V_1 \cap V_3 = V_2 \cap V_3 = \emptyset$, $E_1 \cap E_2 = E_1 \cap E_3 = E_2 \cap E_3 = \emptyset$, for each $x \in V_1$ there exists unique $y \in V_2$ and unique $z \in V_3$ so that $(x, y) \in E_2$, $(x, z) \in E_3$ and $(y, z) \in E_1$, for each $x \in V_2$ there exists unique $y \in V_3$ and unique $z \in V_1$ so that $(x, y) \in E_1$, $(x, z) \in E_3$ and $(y, z) \in E_2$, for each $x \in V_3$ there exists unique $y \in V_1$ and unique $z \in V_2$ so that $(x, y) \in E_2$, $(x, z) \in E_1$ and $(y, z) \in E_3$". ■

Another class of graphs for which positive $H$-matching results were obtained are *planar* graphs. Berman et al. [10] proved that, although the $H$-factor problem is NP-complete for any connected outerplanar graph $H$ with three or more vertices and input planar graph $G$, it is solvable in linear time for triangulated $H$ with four or more vertices. Baker [9] showed that finding an $H$-matching of size $m_H(G)$ admits a polynomial time approximation scheme (PTAS) for any planar graph $H$ Namely, the solution can be approximated within any given constant factor in polynomial time.

## 2.2 $H$-factors in graphs with sufficiently large minimum degree

A classical result of Dirac [26] asserts that if a graph has minimum degree at least $n/2$, then it is Hamiltonian. In particular, minimum degree $n/2$ guarantees a perfect matching, or, in other words, a $K_2$-factor (when discussing $H$-factors we will always assume that $n$ is a multiple of $v(H)$). Dirac's proof is algorithmic. Alternatively, Tutte's Theorem [77] also shows that minimum degree $n/2$ guarantees a perfect matching.

Trying to extend these results to a condition guaranteeing a $K_k$-factor is highly non-trivial. Corrádi and Hajnal [23] proved that a $K_3$-factor exists already when $\delta(G) \geq 2n/3$, and this is tight, as can be seen by taking a complete three-partite graph with vertex class sizes $n/3-1$, $n/3$, $n/3+1$. Finally, in a seminal result, Hajnal and Szemerédi [42] proved:

**Theorem 2.4 ([42])** *If $n$ is a multiple of $k$ then every graph $G$ with $n$ vertices and with $\delta(G) \geq (k-1)n/k$ has a $K_k$-factor.*

Theorem 2.4 is tight as can be seen by taking a complete $k$-partite graph with each vertex class having $n/k$ vertices, except two which have $n/k-1$ and $n/k+1$ vertices. Somewhat surprising is the fact that the proof of Hajnal and Szemerédi is *not* algorithmic. Namely, given a graph with minimum degree, say, $3n/4$, we *know* it has a $K_4$-factor, but the proof does not yield a way to find such a factor in polynomial time (the proof uses "improvement steps", until finally reaching a $K_k$-factor; the number of these steps may be exponential). To appreciate the difficulties, the reader may try the following: Given a graph with 16 vertices and minimum degree 12, prove that it contains four vertex-disjoint $K_4$'s. Only recently, an alternative *algorithmic* proof of Theorem 2.4 has been obtained by Mydlarz and Szemerédi [67]. Subsequently, an alternative simple and algorithmic proof has been obtained by Kirstead and Kostochka [47]. Their algorithm employs a clever discharging technique and runs in $O(n^5)$ time.

The situation concerning $K_k$-factors is well understood combinatorially and computationally. What about other graphs $H$? Clearly, if $H$ has $k$ vertices, we can use Theorem 2.4 to obtain an $H$-factor if $\delta(G) \geq (k-1)n/k$. But is this really necessary? For example, if $H$ is bipartite, already very sparse graphs contain many vertex-disjoint copies of $H$, which suggests that a much smaller minimum degree requirement is needed. As an extreme example, conside the case where $H = P_k$ is a path with $k$ vertices. Dirac's Theorem shows that $\delta(G) \geq n/2$ guarantees an $H$-factor (and this is tight, by obvious divisibility restrictions). In fact, Erdős and Faudree conjectured in [31] that $\delta(G) \geq n/2$ guarantees a $C_4$-factor. Recently, Wang [78] announced a solution to this conjecture. A much stronger conjecture of El-Zahar [30] is the following.

**Conjecture 2.5 ([30])** *If $G$ is a graph with $n = n_1 + \ldots + n_k$ vertices and $\delta(G) \geq \lceil \frac{1}{2}n_1 \rceil + \cdots + \lceil \frac{1}{2}n_k \rceil$ then $G$ has a spanning subgraph consisting of cycles of lengths $n_1, \ldots, n_k$.*

It was first proved by Alon and Yuster in [7] that, in general, the correct parameter to look at,

when considering $H$-factors, is not the number of vertices of $H$; it is the *chromatic number* of $H$ (denote usually by $\chi(H)$).

**Theorem 2.6 ([7])** *For every $\epsilon > 0$ and for every positive integer $h$, there exists an $n_0 = n_0(\epsilon, h)$ such that for every graph $H$ with $h$ vertices and for every $n > n_0$, any graph $G$ with $hn$ vertices and with minimum degree $\delta(G) \geq ((\chi(H) - 1)/\chi(H) + \epsilon)nh$ has an $H$-factor.*

Theorem 2.6 is asymptotically tight, as can be easily seen by letting $G$ be a complete $k$-partite graph with non-equal color classes where $H$ is any uniquely-colorable $k$-partite graph with equal color classes. In fact, it is shown in [7] that, in the full generality, some error term is needed in Theorem 2.6; one cannot prove the same result with $\epsilon = 0$.

Theorem 2.6 *is* algorithmic (the running time is polynomial in $n$, for $\epsilon$ and $h$ fixed). This may seem quite surprising given the fact that Theorem 2.6 *uses* Theorem 2.4, and cannot be proved without it, while, as mentioned earlier, the original proof of Theorem 2.4, which was the only one known when Theorem 2.6 was proved, is not algorithmic. In order to understand this phenomenon, we need to introduce an important tool used in the proof of Theorem 2.6, as well as in many other applications, some of which will be presented in this paper. To describe this tool, the *Regularity Lemma* of Szemerédi [76], we need several definitions.

Let $G = (V, E)$ be a graph, and let $A$ and $B$ be two disjoint subsets of $V$. If $A$ and $B$ are non-empty, let $E(A, B)$ denote the set of edges between them, and put $e(A, B) = |E(A, B)|$. The *density of edges* between $A$ and $B$ is defined as

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

For $\gamma > 0$ the pair $(A, B)$ is called *$\gamma$-regular* if for every $X \subset A$ and $Y \subset B$ satisfying $|X| > \gamma|A|$ and $|Y| > \gamma|B|$ we have

$$|d(X, Y) - d(A, B)| < \gamma.$$

An *equitable partition* of a set $V$ is a partition of $V$ into pairwise disjoint classes $V_1, \ldots, V_t$ whose sizes are as equal as possible. An equitable partition of the set of vertices $V$ of a graph $G$ into the classes $V_1, \ldots, V_t$ is called *$\gamma$-regular* if $|V_i| < \gamma|V|$ for every $i$ and all but at most $\gamma\binom{t}{2}$ of the pairs $(V_i, V_j)$ are $\gamma$-regular. The Regularity Lemma states the following:

**Lemma 2.7 ([76])** *For every $\gamma > 0$, there is an integer $T(\gamma) > 0$ such that for every graph $G$ of order $n > T$ there is a $\gamma$-regular partition of the vertex set of $G$ into $t$ classes, for some $1/\gamma < t < T$.*

What the Regularity Lemma essentially says is that, for a fixed parameter $\gamma$, the set of vertices of every graph can be partitioned into constantly many (almost) equal sized parts, so that (after throwing away at most $\gamma n^2$ edges, those inside parts and those belonging to a non $\gamma$-regular pair; a negligible quantity for dense graphs) the bipartite graph defined between any two parts behaves

like a random bipartite graph. This is because $\gamma$-regularity guarantees a pseudo-random behavior. The regular partition guaranteed by Lemma 2.7 can be associated with a complete weighted *cluster graph* $R$ having $t$ vertices, where the weight of each edge $(i, j)$ is just the density of the corresponding pair (for non-regular pairs we can set this density to zero, since we just delete the edges between non-regular pairs). For more details on the Regularity Lemma we refer the reader to the excellent survey of Komlós and Simonovits [57], which discusses various applications of this powerful result.

**Proof of Theorem 2.6 (sketch).** Lemma 2.7 is used to prove Theorem 2.6 as follows. We obtain a $\gamma$-regular partition where $\gamma$ is chosen sufficiently small with respect to $\epsilon$. Looking at the cluster graph, which has constantly many vertices, we throw away the edges whose weight is too small (again, this weight threshold is a function of $\epsilon$). What remains of the cluster graph is a graph with $t - f(k)$ vertices which satisfies the conditions of Lemma 2.4 for $k = \chi(H)$ (if not, this would violate the minimum degree assumption on $G$). Thus, the remaining cluster graph has a $K_k$-factor. Each element in this $K_k$-factor corresponds to $k$ vertex classes of $G$ every pair of which is $\gamma$-regular and sufficiently dense. We can greedily pack almost all of the vertices in these $k$ vertex classes with vertex-disjoint copies of $H$, just as we can do in a random $k$-partite graph (we don't claim, and don't need, that all $\binom{k}{2}$ densities are the same). We do this for every $K_k$-factor in the cluster graph. This shows that we can obtain an *almost* $H$-factor; there are only $f(\epsilon)n$ unpacked vertices. It is shown in [7] that there is ample freedom to match up the remaining unpacked vertices so that there are no leftovers at all. One way to do this is to actually find some $H'$-factor for $\chi(H') = \chi(H)$ and match the leftovers with vertices of copies of $H'$, where, in turn, each $H'$ or what's left of it after being used to match leftover vertices, has, in turn, an $H$-factor. ■

As can be seen from the proof above, Theorem 2.4 is applied on a constant size graph, the cluster graph. Thus, we do not care whether it is algorithmic. What we do care about is to find a $\gamma$-regular partition in polynomial time (all other proof ingredients; most notably the greedy embedding part, are polynomial). This is not so obvious, especially since deciding, for an input bipartite graph $G$ and a parameter $\gamma$, whether the two vertex classes of $G$ are a $\gamma$-regular pair, is co-NP Complete, as proved in [3]. In particular, computing, for a given bipartite graph, the smallest $\gamma$ for which the graph is $\gamma$-regular, is NP-hard. Fortunately, however, there is an efficient approximation algorithm; for every constant $\gamma$, there is another constant $\gamma' < \gamma$, depending only on $\gamma$, so that given a bipartite graph $G$, if $G$ is not $\gamma$-regular, the algorithm correctly reports this is the case, and provides evidence showing that $G$ is not $\gamma'$-regular. In case $G$ is $\gamma'$ regular the algorithm decides it is $\gamma$-regular. In any other case (i.e., if $G$ is $\gamma$-regular but not $\gamma'$-regular) the algorithm behaves according to one of the above two possibilities and we have no control on the possibility it chooses. The approximation algorithm runs in time $O(n^\omega) = o(n^{2.376})$, as its bottleneck procedure needs to compute, for each pair of vertices, the number of common neighbors they have. The latter can be found by squaring the adjacency matrix. This approximation algorithm, combined with several additional ideas, yields an algorithmic version of the Regularity Lemma, proved by Alon et al. in [3].

**Lemma 2.8 ([3])** *For every fixed $\gamma > 0$, there is an integer $T(\gamma) > 0$ such that for every graph $G$ of order $n > T$ there is a $\gamma$-regular partition of the vertex set of $G$ into $t$ classes, for some $1/\gamma < t < T$, and such a partition can be found in $O(n^\omega)$ time.*

As a consequence, Theorem 2.6 can be implemented to find the desired $H$-factor in $O(n^\omega)$ time (the other algorithmic parts of the proof require less time). In fact, the algorithm is a *fixed parameter algorithm* with respect to $\epsilon$ and $h$. That is, these constants only affect the constant in the runtime expression, and not the exponent. It should be noted that, naturally, the constant $T$ in Lemma 2.8 is larger than the constant $T$ in Lemma 2.7. This is not an issue, however, since already the latter is a really huge constant. In fact, $\log^*(T)$ is a polynomial in $1/\gamma$, and this horrible dependency is, in fact, necessary, as proved by Gowers [37]. Needless to say that algorithms that use Lemma 2.8 are impractical. We mention also that Frieze and Kannan [35] obtained another algorithmic version of the Regularity Lemma, which is somewhat simpler to prove, and which is based on computation of singular values of matrices. A faster, and essentially optimal algorithmic version is given in [51].

Theorem 2.6, although very general, and algorithmic, still leaves much to be desired. The error term in Theorem 2.6 is $o(n)$, but can it be made not to depend on $n$ at all? We know this is true for some graphs, e.g. cliques and paths, for which the error term is 0. Can we characterize those graphs for which the error term is zero? What about graphs $H$ which are locally very sparse; can we guarantee an $H$-factor already for a smaller minimum degree? Conjecture 2.5 suggests that minimum degree $0.6n$ guarantees a $C_5$-factor. All of these questions have been, at least partially, answered.

Komlós et al. [56] sharpened Theorem 2.6 and proved that the error term is only a constant, depending on $H$.

**Theorem 2.9 ([56])** *Let $H$ be a graph with $h$ vertices and $\chi(H) = k$. There exist constants $c(H)$ an $n_0(H)$ such that if $n > n_0(H)$ and $G$ is a graph with $hn$ vertices and minimum degree at least $(1 - 1/k)hn + c(H)$, then $G$ contains an $H$-factor. In fact, if $H$ has a $k$-coloring with color-class sizes $h_1 \le h_2 \le \cdots \le h_k$, then the theorem holds for $c(H) = h_k + h_{k-1} - 1$.*

A crucial tool in the proof of Theorem 2.9, in addition to the Regularity Lemma, is the following important embedding lemma, the *Blow-up Lemma*, that they have developed, and that turned out to be an invaluable tool in extremal $H$-matching theory. In order to describe this lemma we need a definition. We say that a $\gamma$-regular pair $(A, B)$ is $(\gamma, \delta)$-*super regular* if every vertex of $A$ has at least $\delta|B|$ neighbors in $B$ and every vertex of $B$ has at least $\delta|A|$ neighbors in $A$. The Blow-up Lemma essentially says that regular pairs in the Regularity Lemma behave like complete bipartite graphs from the point of view of embedding bounded degree subgraphs.

**Lemma 2.10 ([53])** *Given a fixed graph $R$ with vertices $v_1, \ldots, v_r$ and positive parameters $\delta, \Delta$ there exists a $\gamma = \gamma(\delta, \Delta, r) > 0$ so that the following holds. Let $V_1, \ldots, V_r$ be pairwise disjoint sets and replace $v_i$ with $V_i$. We construct two graphs on $V = \cup V_i$. The first graph, $R_b$ is obtained by*

*replacing each edge $(v_i, v_j)$ of $R$ with the complete bipartite graph between $V_i$ and $V_j$. A sparser graph $G$ is constructed by replacing each edge $(v_i, v_j)$ arbitrarily with some $(\gamma, \delta)$-super-regular pair between $V_i$ and $V_j$. Then: if a graph $H$ with $\Delta(H) \leq \Delta$ is a subgraph of $R_b$ then it is also a subgraph of $G$.*

Notice that the subgraph $H$ in Lemma 2.10 is allowed to be a *spanning subgraph* of $G$. This is a crucial fact, that explains the importance of Lemma 2.10 in obtaining extremal $H$-factor type results.

The original proof of the Blow-up Lemma is not algorithmic, and applies probabilistic methods. An algorithmic version of the Blow-up Lemma is provided in [55]. The desired subgraph $H$, guaranteed to exist in $G$, can be found in time $O(n^{\omega+1}) = o(n^{3.376})$ time where $n = v(G)$. The algorithm can be parallelized and implemented in $\mathrm{NC}^5$. Perhaps the most striking application combining the Blow-up Lemma and the Regularity Lemma is the proof of a conjecture of Seymour, for sufficiently large graphs. It is a rare example where an *exact* result can be proved via tools that are asymptotic in nature. Seymour's conjecture asserts that every graph of order $n$ and with minimum degree at least $kn/(k+1)$ contains the $k$-th power of a Hamiltonian cycle. Pósa conjectured this first for the case $k = 2$ of a square of a Hamiltonian cycle. The case $k = 1$ is simply Dirac's Theorem.

**Theorem 2.11 ([54])** *For any positive integer $k$, there is an $N(k)$ such that, if $G$ has $n > N(k)$ vertices and $\delta(G) \geq kn/(k+1)$ then $G$ contains the $k$-th power of a Hamiltonian cycle.*

Notice that Seymour's conjecture is a significant extension of Theorem 2.4, the Hajnal-Szemerédi Theorem, as the $k$-th power of a Hamiltonian cycle contains a $K_{k+1}$-factor (assuming $k+1$ divides $n$).

While Theorem 2.9 supplies a satisfactory answer to the behavior of the error term in Theorem 2.6, it is still insensitive to the actual structure of the graph $H$. Both theorems yield asymptotically the same bound, depending only on the chromatic number of $H$. An important result that does take the structure of $H$ into account (though, at the price of leaving a negligible fraction of vertices of $G$ unpacked) was given by Komlós [52]. In order to describe his result, we need a definition. For an $r$-chromatic graph $H$ on $h$ vertices, let $u = u(H)$ denote the smallest possible size of a color class in an $r$-coloring of $H$. The *critical chromatic number* of $H$ is $\chi_{cr}(H) = (r-1)h/(h-u)$. It is straightforward to verify that $\chi(H) - 1 < \chi_{cr}(H) \leq \chi(H)$ where $\chi_{cr}(H) = \chi(H) = r$ if and only if every $r$-coloring of $H$ has equal color class sizes. For example, The 5-cycle has $\chi_{cr}(C_5) = 2.5$, and generally, $\chi_{cr}(C_{2k+1}) = 2 + 1/k$.

Unbalanced Turán graphs show that, for any graph $H$, one cannot improve Theorem 2.6 by replacing $\chi(H)$ with a constant smaller than $\chi_c(H)$ [4]. Komlós proved that in Theorem 2.6 one *can* replace $\chi(H)$ with $\chi_{cr}(H)$ at a price of obtaining an *almost* $H$-factor.

**Theorem 2.12 ([52])** *For any graph $H$ and $\epsilon > 0$ there is an $n_0 = n_0(H, \epsilon)$ so that if $G$ is a graph with $n > n_0$ vertices and minimum degree at least $(1 - 1/\chi_{cr}(H))n$ then $G$ contains an $H$-matching that covers all but at most $\epsilon n$ vertices.*

Once again, the Blow-up Lemma and the Regularity Lemma play a crucial role in the proof of Theorem 2.12. Komlós conjectured that the $\epsilon n$ error term in his result can be replaced with a constant that depends only on $H$. He also showed, as did his predecessors, that such a result, if true, is best possible in the sense that one cannot, in general, replace $\epsilon n$ with 0. Three year later, Shokoufandeh and Zhao [74] confirmed the conjecture of Komlós. They have shown that the error term in Theorem 2.12 can be replaced with the constant $5(r-2)(h-u)^2/(u(r-1))$ where $h$ is the number of vertices of $H$, $r = \chi(H)$, and $u = u(H)$.

Still somewhat unsatisfactory, is the fact that we cannot, in general, replace $\chi(H)$ with $\chi_{cr}(H)$ in Theorem 2.9. Can we characterize the cases when we can? Is there a dichotomy? Namely, is it true that for each graph $H$, either $\chi(H)$ or $\chi_{cr}(H)$ is the "right" constant to place in Theorem 2.9, guaranteeing an $H$-factor? Recently, Kühn and Osthus [59] gave a positive answer to this question. They prove that for any graph $H$, either its critical chromatic number or its chromatic number is the relevant parameter which guarantees the existence of $H$-factors in graphs of large minimum degree. The exact classification depends on a parameter which they call the *highest common factor* of $H$ and which is defined as follows.

Given an optimal vertex coloring $c$ of $H$ with $\chi(H) = r$ colors, let $x_1 \leq x_2 \leq \cdots \leq x_r$ denote the sizes of the color classes and put $D(c) = \{x_{i+1} - x_i \mid i = 1 \ldots, r-1\}$. Let $D(H)$ denote the union of all the sets $D(c)$ taken over all optimal colorings $c$. Denote by $hcf_\chi(H)$ the highest common factor of all integers in $D(H)$. If $D(H) = \{0\}$ define $hcf_\chi(H) = \infty$. Write $hcf_c(H)$ for the highest common factor of all the orders of components of $H$. If $\chi(H) \neq 2$ put $hcf(H) = 1$ if $hcf_\chi(H) = 1$. If $\chi(H) = 2$ then we say that $hcf(H) = 1$ if both $hcf_c(H) = 1$ and $hcf_\chi(H) \leq 2$. Note that if all the optimal colorings of $H$ have the property that all color classes have equal size (namely, if $\chi(H) = \chi_{cr}(H)$), then $D(H) = \{0\}$ and so $hcf(H) \neq 1$ in this case. Moreover, there are graphs $H$ with $hcf_\chi(H) = 1$ but such that for all optimal colorings $c$ of $H$ the highest common factor of all integers in $D(c)$ is bigger than one. For such graphs $H$ one needs to consider all optimal colorings of $H$.

Define $\chi^F(H) = \chi_{cr}(H)$ if $hcf(H) = 1$ and $\chi^F(H) = \chi(H)$ otherwise. Also, let $\delta(H, n)$ denote the smallest integer $k$ such that every graph $G$ whose order $n$ is divisible by $v(H)$ and with $\delta(G) \geq k$ contains an $H$-factor.

**Theorem 2.13 ([59])**

$$\left(1 - \frac{1}{\chi^F(H)}\right) - 1 \leq \delta(H, n) \leq \left(1 - \frac{1}{\chi^F(H)}\right) + C.$$

It is not difficult to show that when $\chi(H) > 2$ the $-1$ on the l.h.s. can be omitted.

We end this section by showing that the minimum degree condition in the Hajnal-Szemerédi Theorem is not only a threshold for $K_k$-factor existence, but also a complexity threshold. In other words:

**Theorem 2.14** *Let $k \geq 3$ be an integer, and let $1 - 1/k > \gamma > 0$. The problem of deciding if a given graph with $n$ vertices and minimum degree at least $n(1 - 1/k - \gamma)$ contains a $K_k$-factor is NP-complete.*

**Proof.** We use the simple reduction from the general $K_k$-factor problem given in [60]. Given an input $G$ for the general $K_k$-factor problem, we construct an instance graph $G'$ for the problem appearing in the lemma as follows. Add to $G$ a complete $k$-partite graph with vertex classes $Z_1, \ldots, Z_k$ of size $t := \lceil n/\gamma \rceil$ each. Connect each vertex of $G$ to each vertex of $Z_i$ for $i = 1, \ldots, k-1$. It is easy to see that $G'$ has $N = n + k\lceil n/\gamma \rceil$ vertices and minimum degree at least $N(1 - 1/k - \gamma)$. Clearly, if $G$ has a $K_k$-factor, so does $G'$. Now suppose that $G$ does not have a $K_k$-factor, and that $G'$ has some $K_k$-factor, denoted $L$. Let $L' \subset L$ be the elements of $L$ containing a vertex of $G$. If $S \in L'$ then $S$ contains no vertex from $Z_k$. Moreover, since $G$ does not have a $K_k$-factor, at least one vertex of $Z_1 \cup \cdots \cup Z_{k-1}$ is contained in some element of $L'$. Consider the subgraph $G_0$ of $G'$ obtained by deleting the vertices in all the elements of $L'$. Then the $k$'th vertex class of $G_0$ still has size $\lceil n/\gamma \rceil$ whereas the union of all the other vertex classes of $G'$ has size less than $(k-1)\lceil n/\gamma \rceil$. Thus, $G_0$ does not contain a $K_k$-factor, contradicting the fact that $L \setminus L'$ is a $K_k$-factor of $G_0$. ■

Although the extremal graph-theoretic aspects of $H$-factor and $H$-matching are now quite well understood, there are still many interesting open problems that remain in this area. Let us mention just two of these. Firstly, can Theorem 2.14 be extended to all graph $H$?

**Problem 2.15** *Is it true that for every fixed graph $H$, and for every $1 - 1/\chi^F(H) > \gamma > 0$ it is NP-complete to decide if a given $n$-vertex graph with minimum degree $n(1 - 1/\chi^F(H) - \gamma)$ contains an $H$-factor?*

Secondly, as we know that for complete graphs no error terms are needed in Theorem 2.13 it is interesting to ask:

**Problem 2.16** *Characterize those graphs $H$ for which no error terms are needed in Theorem 2.13.*

Kawarabayashi conjectured [46] that if $H = K_k^-$ (i.e. a complete graph with one edge removed) then no error terms are needed. Notice that $\chi^F(K_k^-) = (k-2)k/(k-1)$. He proved this for $k = 4$. His conjecture was confirmed for all sufficiently large $n$ (as a function of $k$) [25].

# 3 $H$-decomposition and $H$-packing

In this section we focus on $H$-decompositions and, more generally, on $H$-packings. We start with the computational complexity of these problems. Unlike the $H$-factor case, the combinatorial nature of $H$-decompositions turns out to be considerably more notorious. In fact, we devote a separate subsection just to the case of $H$-decompositions and $H$-packings of complete graphs, which is already far from trivial. We then proceed to survey the state of the art in the extremal graph-theoretic aspects of $H$-decompositions.

### 3.1 Finding a large $H$-packing

Computing $p_H(G)$ is, of course, trivial when $H = K_2$. One can also quite easily see how to compute $p_{K_{1,2}}(G)$ in linear time. In fact, $2p_{K_{1,2}}(G) = e(G) - odd(G)$ where $odd(G)$ is the number of odd components of $G$. Several researchers considered other graphs $H$ for which $p_H(G)$ can be computed in polynomial time. It turned out that in all of these results (some of which are quite difficult) each connected component of $H$ is either $K_2$ or $K_{1,2}$. The case $H = 3K_2$ was proved in [12], the case $H = tK_2$ in [1, 15], the case $K_{1,2} \cup tK_2$ in [69], the case $tK_{1,2}$ in [61] and finally the case $sK_2 + tK_{1,2}$ in [16].

Holyer [43] conjectured that the $H$-decomposition problem is NP-complete for every graph $H$ with a connected component consisting of at least three edges. He proved this for complete graphs $H$, and for a few other specific types of $H$. Holyer's conjecture turned out to be a difficult one (unlike Theorem 2.1, the analogous result for $H$-factors) and it was more than a decade later until it was resolved by Dor and Tarsi [28].

**Theorem 3.1 ([28])** *The $H$-decomposition decision problem is NP-complete for every graph $H$ that contains a connected component with at least three edges. In particular, computing $p_H(G)$ is NP-hard for such $H$.*

Their proof reduces the exact $k$-cover problem to $H$-decomposition and is considered one of the more complicated NP-completeness reduction proofs known. The general complexity status of $H$-decomposition is therefore now completely settled.

The greedy algorithm clearly approximates $p_H(G)$ to within an $e(H)$ factor, and it is easily seen to be tight for some $H$, including all cliques. Kann [45] proved that already for $H = K_3$, computing $p_H(G)$ is APX-hard. As in the case of Theorem 2.2 for $H$-matchings, the general result of Hurkens and Schrijver [44] yields a $k/2+\epsilon$ approximation for $p_{K_k}(G)$. Baker's polynomial time approximation scheme for the $H$-matching problem in planar graphs can be extended also to the $H$-packing problem [9]. Caprara and Rizzi [17] prove that maximum triangle packing remains APX-hard even for planar graphs with maximum degree 5 or more. On the positive side, they prove:

**Theorem 3.2 ([17])** *Maximum triangle packing can be solved in polynomial time for graphs with maximum degree* 4.

**Proof.** Clearly we can assume that the input graph is connected and that every edge belongs to some triangle. Furthermore, we can assume that for every vertex, and for every partition of its incident edges into two parts there is a triangle consisting of edges from the two parts (otherwise we can simply split the vertex). Suppose now that $G$ is such a "reduced" graph. If $\Delta(G) = 2$ then $G$ is a triangle. If $\Delta(G) = 3$ then it is easy to check that $G$ is a $K_4$ or a $K_4^-$. Thus, we assume $\Delta(G) = 4$. We define an auxiliary graph $A_G = (\mathcal{T}, \mathcal{F})$ where $\mathcal{T}$ are the triangles of $G$ and $\mathcal{F}$ connects two triangles that share an edge. Our goal is, therefore, to find a maximum independent set in $A_G$. We

show that $A_G$ is either claw-free (does not contain an induced $K_{1,3}$) or else has a constant number of vertices. Since maximum independent set can be solved in polynomial time for claw-free graphs [65], the result follows. Assume that there is a claw in $A_G$. That is, a triangle $T$ of $G$ and three triangles $T_1, T_2, T_3$ sharing an edge with $T$. These four triangles induce 6 vertices in $G$, and it is easily checked that these must be the only vertices of $G$, by our assumption on the reduced structure of $G$. ■

Interestingly, the complexity status of $H$-decomposition for bounded tree-width graphs is not yet settled. This is unlike Theorem 2.3 for the case of $H$-factors for bounded tree-width graphs. The main obstacle lies in the fact that $H$-decomposition is not known to (and probably cannot) be expressed in monadic second-order logic (MSOL). Still, for a large class of graphs $H$, we do know that $H$-decomposition is in P. A graph $H$ is *chromatically $k$-connected* if every vertex cutset induces a subgraph with chromatic number at least $k$ [36]. We consider $K_k$ as being chromatically $(k-1)$-connected. For example, complete $r$-partite graphs with $r \geq 3$ are (at least) chromatically 2-connected. A wheel on $n \geq 6$ vertices and $n-1$ spokes is chromatically 2-connected although it is not complete $r$-partite for any $r$. Bipartite graphs are only chromatically 1-connected. The property of being chromatically $k$-connected is monotone-increasing with respect to edge addition. The following result was proved by Caro and Yuster in [20]:

**Theorem 3.3 ([20])** *For every fixed chromatically 2-connected graph $H$, and every fixed positive integer $t$, there is a polynomial time algorithm for deciding whether a graph with tree-width at most $t$ has an $H$-decomposition.*

**Proof (sketch).** The idea of the proof is to use divide and conquer on the tree-width decomposition of the input graph. In each stage we obtain a $(t+1)$-separator $C$ of the current graph $G$, separating it into parts $A \cup C$ and $B \cup C$ each of size at most $2v(G)/3$. The crucial point is that any copy of $H$ that crosses the cut $C$ *must* contain an edge with both endpoints in $C$. Indeed, if this were not the case then the vertices of such a copy that belong to $C$ form an independent disconnecting set of $H$, contradicting the assumption that $H$ is chromatically 2-connected. Now, since there are at most $\binom{t+1}{2}$ edges in $C$ there are only a constant (although a large one) number of possible configurations that describe the possible partial $H$-decompositions of $A \cup C$ (resp. $B \cup C$) when performing the recursion for $B \cup C$ (resp. $A \cup C$). ■

We note that Theorem 3.3, whose full proof is quite involved, does not yield a fixed parameter tractable algorithm with respect to the parameter $t$. In addition to chromatically 2-connected graphs, $H$-decomposition can be solved for bounded tree-width graphs whenever $H$ is a star [20]. We do suggest the following open problem.

**Problem 3.4** *Show that for every fixed graph $H$ and every fixed positive integer $t$, there is a polynomial time algorithm for deciding whether a graph with tree-width at most $t$ has an $H$-decomposition.*

In spite of the APX-hardness of $H$-packing, there is an interesting and important technique that yields a polynomial time approximation scheme (PTAS) for computing $p_H(G)$ if $G$ is a sufficiently

dense graph. A *fractional $H$-packing* of a graph $G$ is an assignment of nonnegative real weights to the copies of $H$ in $G$ such that for each $e \in E(G)$, the sum of the weights of copies of $H$ containing $e$ is at most one. The value of a fractional packing is the sum of the weights of all copies of $H$. The *fractional packing number* $p_H^*(G)$ is the maximum value of a fractional $H$-packing of a graph $G$. We say that $G$ has a *fractional $H$-decomposition* if $p_H^*(G) = e(G)/e(H)$. Notice that this occurs if and only if the sum of the weights of copies of $H$ containing each edge $e$ is precisely one. Trivially, $p_H^*(G) \geq p_H(G)$ as a fractional packing is a fractional relaxation of an integral one. Also notice that there are cases where a strict inequality holds. Furthermore, a fractional decomposition can exist even though an integral one cannot. For example, $p_{K_3}(K_4) = 1$ while $p_{K_3}^*(K_4) = 2$ since we can assign $1/2$ to each triangle in $K_4$. In fact, a trivial symmetry argument shows that $K_n$ has a fractional $H$-decomposition (assuming $v(H) \leq n$).

Computing $p_H^*(G)$ amounts to solving a linear programming problem with polynomially many variables and constraints. Hence, $p_H^*(G)$ can be computed in polynomial time, and deciding fractional $H$-decomposability is in P. It is therefore interesting to examine those cases where $p_H^*(G)$ is close to $p_H(G)$, since this immediately yields an approximation algorithm for $p_H(G)$. An important result in this direction was obtained by Haxell and Rödl in [41].

**Theorem 3.5 ([41])** *For every fixed graph $H$, if $G$ has $n$ vertices then $p_H^*(G) \leq p_H(G) + o(n^2)$. Furthermore, there is a deterministic polynomial time algorithm that finds an $H$-packing of $G$ of size $p_H^*(G) - o(n^2)$.*

The proof of Theorem 3.5 is quite difficult. A simpler and more general *randomized* algorithm is given in [84]. The main idea in both proofs is to use the Regularity Lemma and compute an optimal fractional $H$-packing on the corresponding cluster graph of the regular partition (recall that this graph has a constant number of vertices). Then, each copy of $H$ in the cluster graph with some weight $\gamma$ is blown up, using Rödl nibble techniques, to obtain a set of edge-disjoint copies of $H$ in $G$ that occupy an almost $\gamma$ fraction of the edges belonging to the cluster pairs that correspond to the edges of $H$ in the cluster graph. Theorem 3.5 obviously implies an approximation algorithm for $H$-packing with an $o(n^2)$ additive error. Since Turán's Theorem asserts that graphs with $n^2(1-1/(\chi(H)-1))+\epsilon n^2$ have $\Theta(n^2)$ edge-disjoint copies of $H$ we have that Theorem 3.5 is a PTAS for $p_H(G)$ if $G$ is a sufficiently dense graph.

Finally, we note that the error term $o(n^2)$ appearing in Theorem 3.5 cannot essentially be improved. It is shown in [84] that for every $\epsilon > 0$ there exist $k = k(\epsilon)$ and $N = N(\epsilon)$ such that for all $n > N$ there exists a graph $G$ with $n$ vertices for which $p_{K_k}^*(G) - p_{K_k}(G) > n^{2-\epsilon}$.

## 3.2 $H$-decompositions and $H$-packings of complete graphs

The $H$-decomposition problem for complete graphs has a long and rich history. The problem of constructing $K_k$-decompositions of $K_n$ is, in fact, at the heart of a major branch of combinatorics

known as *design theory*. A $t$-$(n,k,\lambda)$ *design* is a family $D$ of $k$-element subsets of $[n]$ (called *blocks*) so that any $t$-element subset appears in precisely $\lambda$ elements of $D$. A $t$-$(n,k,1)$ design is also called a $S(t,k,n)$-*Steiner system*. Notice that a $K_k$-decomposition of $K_n$ is simply a $S(2,k,n)$-Steiner system. For more information on block designs we refer to [11].

Clearly, for an $S(2,k,n)$-Steiner system to exist we must have that $k-1$ divides $n-1$ and that $\binom{k}{2}$ divides $\binom{n}{2}$. Notice that these two conditions hold if $n \equiv 1, k \bmod k(k-1)$. More generally, for an $H$-decomposition of $G$ to exist we must have that $e(H)$ divides $e(G)$ and that $gcd(H)$ divides $gcd(G)$ where the *gcd* of a graph is the largest integer dividing the degree of each vertex of the graph. If these two necessary conditions hold we say that $G$ is $H$-*divisible*. Obviously, $H$-divisibility does not guarantee an $H$-decomposition. For example, $K_4$ is $K_{1,3}$-divisible, but there is no $K_{1,3}$-decomposition of $K_4$. A more complicated analysis shows that $K_{16}$, as well as $K_{21}$ and $K_{36}$ do not have a $K_6$-decomposition.

The first non-trivial problem is to construct an $S(2,3,n)$-Steiner system, also known as a *Steiner triple system* $STS(n)$. Is it true that $K_3$-divisibility of $K_n$ guarantees a $K_3$-decomposition? This *non-trivial* question was solved affirmatively by Kirkman in 1847 [49]. Indeed:

**Theorem 3.6 ([49])** *An $STS(n)$ exists if and only if $n \equiv 1, 3 \bmod 6$.*

**Proof.** We will show the case $n \equiv 3 \bmod 6$ using a construction known as the *Bose Construction*. The case $n \equiv 1 \bmod 6$ is constructed using similar techniques, such as *Skolem Constuctions*. Let $n = 6m+3$, and consider any Latin square $L$ of order $2m+1$ which is symmetric and idempotent. That is, $L$ can be thought of as a square symmetric matrix of order $2m+1$, where each row and each column constitutes a permutation of $Q = \{1, \ldots, 2m+1\}$, and, furthermore, $L[i,i] = i$. Such Latin squares are easily obtained by taking the addition table of $Z_{2m+1}$ and renaming the elements so that $L[i,i] = i$ for $i = 1, \ldots, 2m+1$. Our $n$-element ground set $S$ is $Q \times \{0,1,2\}$. The set of triples consists of two types. The first type is $\{(i,0),(i,1),(i,2)\}$ for each $i \in Q$. The second type is $\{(i,k),(j,k),(L[i,j],k+1 \pmod 3)\}$ for $i,j \in Q$ with $i < j$ and for $k = 0,1,2$. Notice that there are $2m+1$ triples of the first type and $3m(2m+1)$ triples of the second type, altogether there are $n(n-1)/6$ triples as necessarily required. It remains to show that each pair of distinct elements of $S$ appears in a triple. Let $(a,b)$ and $(c,d)$ be distinct elements of $S$. If $a = c$ then this pair is in a triple of the first type. Assume now that $a \neq c$. If $b = d$, the pair is in a triple of the second type. We now also assume that $b \neq d$. If $d = b+1 \pmod 3$, let $x$ be the unique index satisfying $L[a,x] = c$. The triple containing the pair is thus $\{(a,b),(x,b),(c,d)\}$. If $d = b-1 \pmod 3$, let $y$ be the unique index satisfying $L[y,c] = a$. The triple containing the pair is thus $\{(y,d),(c,d),(a,b)\}$. ■

The smallest nontrivial Steiner triple system, $STS(7)$, is unique and is isomorphic to the *Fano plane*, the smallest non-trivial finite projective geometry. For general $n$, it is not yet known how many non-isomorphic $STS(n)$ exist, nor is it known how to sample such an $STS(n)$ uniformly. Some successful randomized heuristics, such as *hill climbing*, are useful for generating an arbitrary

$STS(n)$ [75].

Although Theorem 3.6 settles the $K_3$-decomposition problem of $K_n$, we still do not have a complete theory for $H$-decompositions of $K_n$, even when $H = K_k$. We do know that $K_4$-divisibility of $K_n$ (that is, $n \equiv 1, 4 \bmod 12$) guarantees a $K_4$-decomposition, but the constructions are much more involved (see, for example, [14]). Some other small sporadic cases have also been fully analyzed ($H = K_k$, for $k \leq 7$, $H = C_4$, $H = C_5$, trees on at most 7 vertices, the $d$-cube when $d$ is even or $d \leq 5$, as well as a few other specific graphs, see [11]). It was therefore a major breakthrough when Wilson proved in [79], building upon his earlier work on pairwise balanced designs, that $H$-divisibility is *sufficient* in order to guarantee an $H$-decomposition of $K_n$, for all sufficiently large $n$.

**Theorem 3.7 ([79])** *For every fixed graph $H$, there exists $N = N(H)$ so that for all $n > N$, if $K_n$ is $H$-divisible then $K_n$ has an $H$-decomposition.*

Wilson's proof is deep. It uses, among other things, tools from linear algebra, finite fields, and design theory. In a qualitative sense, the theorem is best possible, since, as demonstrated earlier, $H$-divisibility does not generally guarantee $H$-decomposition for all $n$. The value $N(H)$ in Wilson's proof is quite large, and probably far from optimal, but an exact determination of the smallest possible $N(H)$ seems to be beyond reach, in general. Theorem 3.7 does settle the $H$-decomposition decision problem for $K_n$, which can now be trivially solved in polynomial time. However, it does not supply an efficient algorithm for producing an $H$-decomposition. We do note that if we only wish to find an asymptotically optimal $H$-packing of $K_n$, that is, a set of $\binom{n}{2}/e(H) - o(n^2)$ edge-disjoint copies of $H$ in $K_n$, the problem becomes more tractable using Rödl's nibble method [71, 33] and an algorithmic version of it [38]. We shall elaborate more on these important techniques in Section 4 as these result are most striking in the hypergraph setting.

If $K_n$ is $H$-divisible, and $n > N(H)$, we obviously have from theorem 3.7 that $p_H(K_n) = \binom{n}{2}/e(H)$. In particular, $p_H(K_n)$ can be computed in polynomial time when $K_n$ is $H$-divisible (checking if a graph $G$ is $H$-divisible can trivially be done in linear time). Can we also compute $p_H(K_n)$ efficiently when $K_n$ is *not* $H$-divisible? This may not seem like a difficult question at first glance. After all, there is a constant difference between any two consecutive values of $n$ for which $K_n$ is $H$-divisible, which immediately implies that $p_H(K_n)$ can be approximated within an additive $O(n)$ error. Still, computing $p_H(K_n)$ precisely turns out to be a much more demanding task. As an illuminating example, suppose that $H$ is a tree with $h$ edges. We know that $p_H(K_n) \leq \binom{n}{2}/h$ and we know that equality holds if $h$ divides $\binom{n}{2}$ (and $n$ is sufficiently large). Can we show that $p_H(K_n) = \lfloor \binom{n}{2}/h \rfloor$ for all sufficiently large $n$? Although this is known to be true, there is no known simple direct proof of this fact. If we knew that $H$-divisibility of a very dense graph $G$ (say, with very high minimum degree) guarantees $H$-decomposition, then $p_H(K_n) = \lfloor \binom{n}{2}/h \rfloor$ would be a trivial consequence. Indeed, given $K_n$, delete at most $h - 1$ independent edges so that the resulting graph would be $H$-divisible and with minimum degree at least $n - 2$. Fortunately, it was proved in

1991 by Gustavsson [39] that $H$-divisible graphs with sufficiently many vertices and with a (very) high minimum degree, are $H$-decomposable (the exact statement of his result is given in the next subsection). Using Gustavsson's result, together with several additional ideas, including a theorem of Erdős and Gallai on realizable degree sequences, Caro and Yuster [19] were able to obtain a closed formula for $p_H(K_n)$, for all sufficiently large $n$.

**Theorem 3.8 ([19])** *Let $H$ be a fixed graph with $h$ edges and $d = gcd(H)$. There exists $N = N(H)$ such that for all $n > N$,*

$$p_H(K_n) = \left\lfloor \frac{dn}{2h} \left\lfloor \frac{n-1}{d} \right\rfloor \right\rfloor,$$

*unless $n \equiv 1 \bmod d$ and $n(n-1)/d \equiv b \bmod (2h/d)$ where $1 \leq b \leq d$, in which case*

$$p_H(K_n) = \left\lfloor \frac{dn}{2h} \left\lfloor \frac{n-1}{d} \right\rfloor \right\rfloor - 1.$$

*In particular, $p_H(K_n)$ can be computed in polynomial time.*

**Proof (sketch).** It is first shown that the claimed value of $p_H(K_n)$ is an upper bound. It is next shown how to construct an $H$-divisible spanning subgraph $G$ with very high minimum degree having the property that $e(G)/h$ equals the claimed value. The construction of $G$ is done by first determining the degree sequence required from $G$, and then using a result of Erdős and Gallai in order to realize this degree sequence. After obtaining $G$, one uses Gustavsson's Theorem to show that $G$ has an $H$-decomposition. ■

### 3.3 $H$-decompositions in graphs with sufficiently large minimum degree

We begin this subsection by showing that if one searches for sufficient extremal conditions that guarantee $H$-decomposition (in addition to $H$-divisibility), the "correct" parameter to look at is the minimum degree.

**Proposition 3.9** *For each $k \geq 3$ there are infinitely many $n$ for which there exist $K_k$-divisible graphs with $n$ vertices and minimum degree at least $\frac{k}{k+1}n - 1$ that are not $K_k$-decomposable.*

**Proof.** We show a construction of Nash-Williams for the case $k = 3$. It can be quite easily generalized to arbitrary $k$. Let $n = 24m + 12$ and consider four vertex-disjoint cliques on $6m + 3$ vertices each. Denote their vertex sets by $V_i$ for $i = 1, \ldots, 4$. Construct a complete bipartite graph between $V_i$ and $V_{i+1}$ (indices modulo 4). The resulting graph $G$ has $n$ vertices, it is regular of degree $18m + 8$ and it has $(9m + 4)(24m + 12)$ edges. Thus, it is $K_3$-divisible. Any $K_3$-decomposition of $G$ consists of $(9m+4)(8m+4)$ triangles. However, there are $4(6m+3)^2$ edges with endpoints in distinct cliques, and since each triangle of $G$ contains at least one edge both of whose endpoints are in the same clique, we need at least $2(6m+3)^2$ triangles in any $K_3$-decomposition of $G$. The contradiction follows since $2(6m + 3)^2 > (9m + 4)(8m + 4)$. ■

Nash-Williams conjectured [68] that the construction for triangles is best possible. That is, any $K_3$-divisible graph with minimum degree at least $3n/4$ is $K_3$-decomposable. This conjecture is still wide open. For general $k$ we have the following conjecture (see [39]).

**Conjecture 3.10** *For each $k \geq 3$ there exists $N = N(k)$ so that for all $n > N$, if an $n$-vertex graph $G$ is $K_k$-divisible and $\delta(G) \geq \frac{k}{k+1}n$ then $G$ is $K_k$-decomposable.*

In light of Conjecture 3.10 one can wonder whether *any* minimum degree condition guarantees a $K_k$-decomposition, or more generally, an $H$-decomposition. Let us denote by $\rho(H)$ the largest constant for which $\delta(G) \geq (1-\rho)(n-1) + o(n)$ and $H$-divisibility of $G$ guarantee $H$-decomposition, for $n$ sufficiently large. The above conjecture, if true, asserts that $\rho(K_k) = 1/(k+1)$. Notice that $\rho(H) \geq 0$ is just Wilson's Theorem. Can we show that $\rho(H) > 0$ for every graph $H$? This (extremely difficult) question was solved by Gustavsson [39].

**Theorem 3.11 ([39])** *For every fixed graph $H$ there exists $c = c(H) > 0$ and $N = N(H)$ so that every graph $G$ with $n > N$ vertices that is $H$-divisible and with $\delta(G) \geq (1-c)n$, is also $H$-decomposable.*

**Proof (ultra sketch).** Gustavsson's proof is long and difficult, and no short proof of this result is known (and probably no such proof exists). Before sketching some of his arguments, we note that the proof is only existential. Furthermore, the constant $c(H)$ obtained is extremely small. Already $c(K_3) = 10^{-24}$, while Conjecture 3.10 asserts that taking $c(K_3) = 1/4$ suffices. That is, the obtained $c(H)$ is probably much (much) smaller than $\rho(H)$.

The first observation is that it suffices to prove Theorem 3.11 for $H = K_k$. Indeed, if $H$ is not a complete graph, let $k$ be the smallest integer for which $K_k$ is $H$-decomposable. Such a $k$ exists by Theorem 3.7. Now, given a graph $G$ with very high minimum degree, which is $H$-divisible, it is not difficult to show that we can always greedily delete a small number of edge-disjoint copies of $H$ from $G$ (in fact, each vertex of $G$ appears in at most constantly many deleted copies), and obtain a spanning subgraph that is $K_k$-divisible. The minimum degree is only reduced by a constant. Another, more subtle important argument is that it suffices to prove Theorem 3.11 for $k$-partite graphs with $n$ vertices in each vertex class, and that have minimum degree $n(k-1)(1-c'(K_k))$; we call such graphs $G(k,n)$ graphs. Indeed, if $G$ is a graph with $nk$ vertices with very high minimum degree we can arbitrarily split its vertex set to $k$ parts of size $n$ each. Each vertex class $V_i$ induces an $n$-vertex graph $G[V_i]$ with very high minimum degree. We can think of each $G[V_i]$ as if it were a complete graph and $K_k$-decompose it using Theorem 3.7. Now, the "illegal" copies of $H$ in these $k$ decompositions (those that contain a missing edge) can be ignored, and the other "true" edges in those ignored copies can be repacked with edge-disjoint partite copies of $K_k$ ones that use a single vertex from each partite graph. After doing this, and taking care about maintaining $K_k$-divisibility, which is not straightforward, but manageable, one remains with the need to $K_k$-decompose a $G(k,n)$

graph whose minimum degree fraction $1 - c'(K_k)$ is still very high, although slightly smaller than the original minimum degree fraction $1 - c(K_k)$.

Proving $K_k$-decomposability of a $G(k, n)$ graph transforms the problem to a significantly more manageable setting. Two Latin squares $L_1$ and $L_2$ are *orthogonal* if for any symbol, then $n$ cells containing that symbol in one of the squares contain all $n$ symbols in the other square. One can interpret a system of $k-2$ mutually orthogonal Latin squares $L_1, \ldots, L_{k-2}$ of order $n$ as $n^2$ ordered $k$-tuples of $[n]$ where each tuple is of the form $(i, j, L_1(i, j), \ldots, L_{k-2}(i, j))$. Notice that the orthogonality implies that no two coordinates have the same pair of values in two distinct $k$-tuples. In other words, the $n^2$ ordered $k$-tuples form a $K_k$-decomposition of a *complete* $G(k, n)$. Notice that the converse is equally true. Any $K_k$-decomposition of a complete $G(k, n)$ defines a set of $k-2$ mutually orthogonal Latin squares of order $n$. Consider for example the case $k = 3$. This case simply states that a Latin square is isomorphic to a $K_3$-decomposition of a complete $G(3, n)$. Now assume that $G(3, n)$ is not complete, but has very high minimum degree. The tripartite complement of $G(3, n)$ is another $G(3, n)$ with very small minimum degree. Clearly, it corresponds to a partially filled Latin square, where each row and column contain a very small number of occupied cells, and each symbol appears only a small number of times. If we can *fill* this partial Latin square, and obtain a complete Latin square, this filling corresponds to a triangle decomposition of the original dense $G(3, n)$. Similarly, if we can fill a partially filled system of $k-2$ orthogonal Latin squares, the filling corresponds to a $K_k$-decomposition of a $G(k, n)$ with very high minimum degree. Gustavson's main technical work focuses, therefore, on completion of systems of orthogonal, partially filled, Latin squares. It uses clever applications of the marriage theorem, chain rotations, and case analysis. ■

Theorem 3.11 immediately yields a trivial polynomial time algorithm for computing $p_H(G)$ whenever $G$ is $H$-divisible and has sufficiently high minimum degree. It turns out that significantly more effort in required if one drops the $H$-divisibility requirement. A polynomial time algorithm for computing $p_H(G)$ in graphs with sufficiently high minimum degree was given by Alon et al. [2].

**Theorem 3.12 ([2])** *For every fixed graph $H$ there exist positive constants $\epsilon(H)$ and $N(H)$ such that if $G$ is a graph with $n > N(H)$ vertices and has minimum degree at least $(1 - \epsilon(H))n$, then $p_H(G)$ can be computed in polynomial time.*

The idea of the proof is to find a minimum cardinality set of edges whose deletion from $G$ results in an $H$-divisible spanning subgraph with (still) very high minimum degree, and use Theorem 3.11 to $H$-decompose it. Finding such a minimum cardinality set of edges is non-trivial, but can be reduced to a maximum weighted matching problem.

As we do not know $\rho(H)$ even for very small graphs such as $K_3$, it is interesting to see for which graphs we *can* determine $\rho(H)$. We first notice that for any (say, connected) graph $H$ other than $K_2$, we have $\rho(G) \leq 1/2$. Indeed, this is so already for $K_{1,2}$. Let $n$ be even and such that $\binom{n/2}{2}$ is odd. Then, a graph composed of two vertex-disjoint $K_{n/2}$ is $K_{1,2}$-divisible, but not $K_{1,2}$-decomposable.

Similar examples hold for any other graph. In fact, it is not difficult to show that $\rho(G) > 1/2$ if $G$ is not bipartite. As the Turán number of bipartite graphs is $o(n^2)$, it is tempting to conjecture:

**Conjecture 3.13** *For every bipartite graph $H$ we have $\rho(H) = 1/2$.*

Even this conjecture is far from being solved. In fact, it is open even for $H = C_4$. It has been proved by the author in a very precise form for trees in [80, 81]:

**Theorem 3.14 ([81])** *Let $H$ be any tree with $h$ edges. Let $G$ be a graph on $n \geq (12h)^{10}$ vertices with $\delta(G) \geq \lfloor n/2 \rfloor$. Then $p_H(G) = \lfloor e(G)/h \rfloor$, and, in particular, if $h$ divides $e(G)$, then $G$ has an $H$-decomposition.*

**Proof (sketch).** The first step (and the most difficult one) is to prove that graphs which are relatively good expanders satisfy the theorem. It is proved in [80] that if $G$ has the property that for every set $X \subset V$ with $|X| \leq n/2$ there are at least $|X|10h^4\sqrt{n \log n}$ edges between $X$ and $V = \setminus X$, then $p_H(G) = \lfloor e(G)/h \rfloor$ (provided $n$ is sufficiently large). The proof uses several probabilistic arguments and applications of the marriage theorem. Notice that, in particular, graphs with minimum degree $n/2+10h^4\sqrt{n \log n}$ satisfy the condition. This already gives $\rho(H) = 1/2$ for trees. To prove the sharp result in the statement of the theorem one now needs only to consider graphs with minimum degree at least $n/2$ that are *not* good expanders. However, it is not difficult to show that the structure of such graphs is very similar to two large almost cliques that are almost disjoint. One therefore needs to carefully embed trees that intersect both of these almost cliques (there are not too many edges in all of these trees together) so as to be able to decompose the remaining two (now disjoint, and still very dense) sets of unused edges. ■

The largest class of bipartite graphs for which Conjecture 3.13 has been verified is given in [82].

**Theorem 3.15 ([82])** *For every bipartite graph $H$ with $\delta(H) = 1$ we have $\rho(H) = 1/2$.*

Notice that the class of bipartite graphs with $\delta(H) = 1$ contains all trees, as well as many other interesting bipartite graphs, including many dense ones. Both Theorem 3.14 and 3.15 are algorithmic. They yield a randomized algorithm that, given as input a graph $G$ satisfying the requirements, finds an $H$-decomposition of $G$ in polynomial time.

A conjecture of Thomassen suggests that sufficient connectivity suffices in order to guarantee an $H$-decomposition when $H$ is a tree.

**Conjecture 3.16** *For every tree $H$, there is a constant $c = c(H)$ so that every $c$-edge connected graph whose number of edges is divisible by $e(H)$ has an $H$-decomposition.*

Thomassen's conjecture allows for the decomposed graph $G$ to be sparse, as opposed to the graphs $G$ in Theorem 3.14 that are always dense.

Finally, recalling the definition of fractional $H$-decomposition and Theorem 3.5, let us denote by $\rho^*(H)$ the largest constant for which $\delta(G) \geq (1-\rho)(n-1) + o(n)$ guarantees a fractional $H$-decomposition. Clearly we have $\rho^*(H) \geq \rho(H)$. Interestingly, it is shown in [85] that a somewhat more complicated construction to the one used in Proposition 3.9 gives $\rho^*(K_k) \leq 1/(k+1)$ for $k \geq 3$. We therefore conjecture:

**Conjecture 3.17** *$\rho^*(H) = \rho(H)$ for every non-bipartite graph $H$.*

We mention that, unlike the horribly small lower bound for $\rho(K_k)$ that follows from Gustavsson's Theorem (this bound is roughly $10^{-37}k^{-94}$), a "reasonable" lower bound of $1/9k^{10}$ for $\rho^*(K_k)$ is given in [85]. That result also yields a polynomial time algorithm for finding a fractional $K_k$-decomposition, and an asymptotically optimal $K_k$-packing in graphs with minimum degree at least $n(1-1/9k^{10}) + o(n)$.

# 4 Packing problems in other combinatorial structures

## 4.1 Hypergraphs

The $H$-packing and $H$-matching problems are naturally extended to the hypergraph settings, and all definitions remain intact. However, unlike the graph-theoretic case, obtaining $H$-factor and $H$-decomposition results that are based upon the density of the input hypergraph is generally wide open. In fact, for all but a few specific hypergraphs we still do not even know the Turán number. Namely we cannot even tell, in general, how many edges guarantee $p_H(G) > 0$.

Still, an important set of results that turned out to be extremely useful in *asymptotic* hypergraph $H$-packing is based upon the Rödl *nibble* method. Let $K_k^r$ denote the complete $r$-uniform hypergraph with $k$ vertices. Clearly, $p_{K_k^r}(K_n^r) \leq \binom{n}{r}/\binom{k}{r}$ and equality holds if and only if $K_r^n$ has a $K_k^r$-decomposition. Solving a longstanding conjecture of Erdős and Hanani, Rödl proved in [71]:

**Theorem 4.1 ([71])**

$$p_{K_k^r}(K_n^r) = \frac{\binom{n}{r}}{\binom{k}{r}}(1 + o_n(1)).$$

Notice that the case $r = 2$ (the case of graphs) is just an asymptotic version of Theorem 3.8 that gives exact values for $p_{K_k}(K_n)$. However, already for $r = 3$, no result analogous to Theorem 3.8 is known, and even Wilson's theorem has no hypergraph equivalent. In this respect, Theorem 4.1 is very strong. Moreover, it also has an algorithmic version (see [38]). That is, an almost complete $K_k^r$ packing of $K_k^n$ can be produced in polynomial time. This algorithmic result is also interesting in the case $r = 2$ of graphs. Several researchers noted that Rödl's result can be expressed as a result on *matchings* of hypergraph. Indeed, consider an $r'$-uniform hypergraph $X$ with $r' = \binom{k}{r}$ whose vertices are all the $r$-subsets of $[n]$ and whose edges are all the $r'$ $r$-tuples that lie in a $k$-subset of $[n]$. Then,

$p_{K_k}(K_n)$ is just the cardinality of a maximum matching of $X$. Thus, Theorem 4.1 can be interpreted as saying that $X$ has an almost perfect matching. Notice that $X$ is regular of degree $D = \binom{n-r}{k-r}$ and every pair of vertices of $X$ lie together in at most $\binom{n-r-1}{k-r-1} = o(D)$ common edges. It turns out that *every* hypergraph with such properties has an almost perfect matching, as proved by Frankl and Rödl [33]. Recall that if $x, y$ are two vertices of a hypergraph then $deg(x)$ denotes the degree of $x$ and $deg(x, y)$ denotes the number of edges that contain both $x$ and $y$ (their *co-degree*).

**Theorem 4.2 ([33])** *For an integer $r \geq 2$ and a real $\beta > 0$ there exists a real $\mu > 0$ so that: If the $r$-uniform hypergraph $L$ on $q$ vertices has the following properties for some $D$:*
*(i) $(1-\mu)D < deg(x) < (1+\mu)D$ holds for all vertices,*
*(ii) $deg(x, y) < \mu D$ for all distinct $x$ and $y$,*
*then $L$ has a matching of size at least $(q/r)(1-\beta)$.*

The proof of Theorem 4.2, which immediately implies Theorem 4.1, is based on probabilistic arguments, in particular repeated applications of the second moment method. We refer the reader to [5] for a proof of Theorem 4.2 due to Pippenger, which is slightly stronger and simpler than the original proof.

When the hypergraph $G$ is not complete, one can resort again to fractional decompositions. This turns out to be beneficial also in the hypergraph setting. An analogue of Theorem 3.5 also holds for $r$-uniform hypergraphs. The following has been proved by Rödl et al. in [72].

**Theorem 4.3 ([72])** *For a fixed $r$-uniform hypergraph $H$, if $G$ is an $n$-vertex $r$-uniform hypergraph then $p^*_H(G) - p_H(G) = o(n^r)$.*

The proof is a non-trivial generalization of Theorem 3.5. In particular, a strong hypergraph version of the Regularity Lemma that has only recently been obtained is a crucial ingredient in the proof.

Let $G = (V, E)$ be an $r$-uniform hypergraph. For $S \subset V$ with $1 \leq |S| \leq r-1$, let $deg(S)$ be the number of edges of $H$ that contain $S$. For $1 \leq d \leq r-1$ let $\delta_d(H) = \min_{S \subset V, |S|=d} deg(S)$ be the *minimum $d$-degree* of $H$. For $0 \leq \beta \leq 1$ we say that $H$ is *$\beta$-dense* if $\delta_d(H) \geq \beta\binom{n-d}{r-d}$ for all $1 \leq d \leq r-1$. Notice that $K^r_n$ is 1-dense and that $H$ is $\beta$-dense if and only if $\delta_{r-1}(H) \geq \beta(n-r+1)$. The following result has been proved by the author in [86].

**Theorem 4.4** *Let $k$ and $r$ be integers with $k > r \geq 3$. There exists a positive $\beta = \beta(k, r)$ and an integer $N = N(k, r)$ such that if $H$ is a $(1-\beta)$-dense $r$-uniform hypergraph with more than $N$ vertices then $H$ has a fractional $K^r_k$-decomposition.*

We note that the constant $\beta = \beta(k, r)$ obtain in the proof is larger than $6^{-kr}$. Combining Theorem 4.4 and 4.3 we immediately obtain:

**Corollary 4.5** *Let $H$ be a fixed $r$-uniform hypergraph. There exists a positive constant $\beta = \beta(H)$ so that if $G = (V, E)$ is a $(1-\beta)$-dense $r$-uniform hypergraph with $n$ vertices then $G$ has an $H$-packing that covers $|E|(1 - o_n(1))$ edges.*

## 4.2 Random graphs

Let $G = G(n,p)$ denote, as usual, the random graph with $n$ vertices and edge probability $p$. In the extensive study of the properties of random graphs many researchers observed that there are sharp *threshold functions* for various natural graph properties. For a graph property $A$ and for a function $p = p(n)$, we say that $G(n,p)$ satisfies $A$ *almost surely* if the probability that $G(n,p(n))$ satisfies $A$ tends to 1 as $n$ tends to infinity. We say that a function $f(n)$ is a *sharp threshold function* for the property $A$ if there are two positive constants $c$ and $C$ so that $G(n,cf(n))$ almost surely does not satisfy $A$ and $G(n,Cf(n))$ satisfies $A$ almost surely.

Problems concerning sharp threshold functions for $H$-factors and $H$-decompositions of $G(n,p)$ have been studied by several researchers. Already Erdős and Rényi proved [32] that $\log n/n$ is a sharp threshold function for $K_2$-factor. Many subsequent papers by various authors that supply more detailed information regarding perfect matchings in random graphs. The following problem is attributed to Erdős.

**Problem 4.6** *For every fixed graph $H$, determine a threshold function $f_H(n)$ for the property that $G(n,p)$ has an $H$-factor (assuming $v(H)$ divides $n$).*

It is not difficult to prove that a threshold function for $H$-factors *exists* (this follows, in particular from the fact that $H$-factor existence is a monotone graph property [34]). However, already for $H = K_3$ we still do not know what this threshold is. Erdős conjectured that a $K_k$-factor occurs already when $G(n,p)$ has the property that every vertex lies on *some* $K_k$. The latter property, denoted $EP(k)$, is certainly a necessary condition for having an $H$-factor. Sharp thresholds for $EP(k)$ are known. It is an easy exercise to show that $\log n/n$ is a threshold for $EP(2)$. The threshold for $EP(3)$ (every point on some triangle) is also not too difficult to compute. It is $(\log n)^{1/3}n^{-2/3}$. Thus, it is conjectured that $(\log n)^{1/3}n^{-2/3}$ is a threshold for $K_3$-factor. At present, the best upper bound that guarantees a $K_3$-factor follows from the result of Kim [48] who proved that $p = \Omega(n^{-2/3+1/18})$ already guarantees a $K_3$-factor, improving an earlier $n^{-2/3+1/15}$ bound of Krivelevich [58]. Recently, Johansson obtained a bound which is very close to the conjectured bound, and this was subsequently further improved by Kahn and Vu, but these results are still not published. We also note that if one settles for an *almost* $K_k$-factor, namely a factor that misses a small constant proportion of the vertices, it is not difficult to show that the conjectured threshold of $EP(k)$ *is* sufficient. This follows by applying Theorem 4.2 on the hypergraph whose edges are the $K_k$-copies.

There is, however, a wide class of graphs $H$ for which problem 4.6 has been solved. Define the *fractional arboricity* of $H$ as

$$a(H) = max\{\frac{|E'|}{|V'|-1}\},$$

where the maximum is taken over all subgraphs $(V',E')$ of $H$ with $|V'| > 1$. The reason for name fractional arboricity is that $\lceil a(H) \rceil$, the *arboricity of* $H$, is the minimum number of forests whose union covers all edges of $H$. The following theorem has been proved in [73] and independently in [6].

**Theorem 4.7 ([73, 6])** *Let $H$ be a graph with $\delta(H) < a(H)$. Then $n^{-1/a(H)}$ is a sharp threshold function for $H$-factors.*

**Proof (sketch).** It is an easy large deviation argument to show that if $p = cn^{-1/a(H)}$ and $c$ is a small constant then almost surely $G(n,p)$ does not even have $n/h$ copies of the dense subgraph of $H$ that yields its arboricity. In particular $G(n,p)$ almost surely cannot have an $H$-decomposition. This part does not assume that $\delta(H) < a(H)$. For the other direction, one can show using Janson's inequality that almost surely $G(n, Cn^{-1/a(H)})$ has $n/h$ vertex-disjoint copies of the subgraph $H'$ obtained from $H$ by deleting a vertex of minimum degree. Notice that by our assumption, $a(H') = a(H)$. What now remains is to match the $n/h$ remaining uncovered vertices with the $n/h$ copies of $H'$ so that a matched vertex plays the role of the deleted minimum degree vertex. This can be done by constructing an appropriate bipartite matching problem and using the above-mentioned result of Erdős and Rényi [32]. One just needs the neighborhoods of the $n/h$ remaining vertices to still be random. This is easy to achieve using a standard two-stage uncovering technique of the edges of $G(n,p)$. ■

The class of graphs with $\delta(H) < a(H)$ contains many interesting graphs. For example, it contains all graphs that have at least one cycle and that have a vertex with degree 1.

It is interesting to note that the *fractional* version of $K_k$-factor thresholds is completely settled. In a fractional $H$-factor one assigns weights to the copies of $H$ so that for each vertex, the sum of the weights is at most one, and the sum of all weights of all $H$-copies is $n/v(H)$ (notice that in this version we do not require than $v(H)$ divides $n$). The lower bound of a threshold for $EP(k)$ clearly also applies to thresholds for fractional $K_k$-factors. Haber and Krivelevich [40] proved that the threshold for $EP(k)$ is, in fact, also a threshold for fractional $K_k$-factors (they also showed it for a larger class of graphs $H$).

We now turn to $H$-decompositions. Threshold functions for $H$-decompositions in the $G(n,p)$ model are not well defined because of the enforced $H$-divisibility requirements. There are several ways to overcome this obstacle. One can ask for the threshold in the random *regular* model, where all $r$-regular graphs with $n$ vertices are equally likely. For example, if $H = K_k$ one can ask for the minimum $r_k = r_k(n)$ so that if $r \geq r_k$ and $k-1$ divides $r$ then a random $r$-regular $n$-vertex graph has $\lfloor rn/(k(k-1)) \rfloor$ edge-disjoint $K_k$, almost surely. This problem is wide open already for $H = K_3$. An easy lower bound of $r_3 = \Omega(\sqrt{n \log n})$ follows from the requirement that every edge be on a triangle. It is therefore conjectured that $r_3 = \Theta(\sqrt{n \log n})$. Another way to overcome the $H$-divisibility obstacle is to ask for an $H$-packing of $G(n,p)$ that leaves at most, say, $nv(H)$ unpacked edges. It is not even clear that a sharp threshold function for this property exists, in particular since this property is not monotone. Still, it is conjectured that $p = (\log n/n)^{1/2}$ is a sharp threshold function for containing a $K_3$-packing that leaves at most $3n$ edges unpacked.

Whenever $gcd(H) = 1$ it is well-defined to ask for a threshold function for an $H$-packing of $G(n,p)$ that leaves less than $e(H)$ edges unpacked (in particular, such a packing is an $H$-decomposition if the number of edges of $G(n,p)$ divides $e(H)$, and this occurs with probability approaching $1/e(H)$

as $n$ tends to infinity). Although this problem is generally wide open, it has been settled for trees [83].

**Theorem 4.8 ([83])** *if $H$ is a tree with more than one edge, then $\log n/n$ is a sharp threshold function for the property that $G(n,p)$ contains $\lfloor e(G)/e(H) \rfloor$ edge-disjoint copies of $H$.*

The proof of Theorem 4.8, as well as the proofs of the other theorems in this subsection, yield a corresponding polynomial time randomized algorithm.

**Problem 4.9** *Find a sharp threshold function for even one single graph $H$ that is not a tree.*

Finally, it is well-defined to ask for the threshold of a *fractional* $H$-decomposition. Here there are no $H$-divisibility obstacles. In particular, we state the following open problem.

**Problem 4.10** *Find a sharp threshold function for fractional $K_3$-decomposition of $G(n,p)$.*

## 4.3 Directed and oriented graphs

For a directed graph $H$, let us denote by $gcd(H)$ the smallest positive integer so that there exist integers $x_v$ for $v \in V$ so that $\sum_{v \in V} x_v d^+(v) = \sum_{v \in V} x_v d^-(v) = gcd(H)$. For example, the $gcd$ of a directed triangle is 1 while the gcd of $D_3$, the complete directed graph with 3 vertices, is 2. Clearly, in order for $D_n$, the complete directed graph with $n$ vertices, to be $H$-decomposable, we must have that $gcd(H)$ divides $n-1$ and that $e(H)$ divides $n(n-1)$.

Wilson's Theorem [79] also applies to directed graph.

**Theorem 4.11** *For every fixed directed graph $H$, there exists $N = N(H)$ so that for all $n > N$, if $e(H)$ divides $n(n-1)$ and $gcd(H)$ divides $n-1$ then $D_n$ has an $H$-decomposition.*

Theorem 4.11 essentially settles the $H$-decomposition problem for $D_n$, but the situation becomes significantly more complicated for tournaments. Suppose $H$ is a fixed oriented graph (orientation, for short). When is it true that a tournament with $n$ vertices is $H$-decomposable?

**Problem 4.12** *Characterize those fixed orientations $H$ for which it is NP-complete to decide if a given tournament is $H$-decomposable.*

For some nontrivial orientations $H$ it is plausible that *each* tournament $T_n$ with $n$ vertices has a packing with edge-disjoint $H$ that leaves at most $o(n^2)$ edges of $T_n$ unpacked. In particular, this means that for such $H$, there is a trivial PTAS for $p_H(T_n)$. Let $TT_k$ denote the transitive tournament with $k$ vertices. We propose the following exact conjecture.

**Conjecture 4.13** *For every tournament $T_n$ with $n$ vertices, $p_{TT_3}(T_n) \geq \lceil n(n-1)/6 - n/3 \rceil$.*

The bound in the conjecture is tight. To see this consider a tournament constructed from three disjoint sets of $n/3$ vertices $V_1, V_2, V_3$. All edges are oriented from $V_i$ to $V_{i+1}$ (indices modulo 3). Edges within each set are oriented arbitrarily. It is easy to see that any $TT_3$ packing of such a tournament has size at most $\lceil n(n-1)/6 - n/3 \rceil$.

Interestingly, already for $TT_4$ there are tournaments with the property that any $TT_4$-packing misses $\Theta(n^2)$ edges. There is a well-known construction of a tournament with seven vertices that has *no* copy of $TT_4$. Blowing up this tournament into 7 vertex sets of size $n/7$ each and orienting the edges within each set arbitrarily yields a tournament $T_n$ in which every $TT_4$-packing misses at least $n^2/14$ edges.

For every fixed orientation $H$, a *random* tournament almost surely has an $H$-packing that misses only $o(n^2)$ edges. This is an easy consequence from Theorem 4.2 applied to the hypergraph whose vertices are the edges of $T_n$ and whose edges are the copies of $H$ in $T_n$. This hypergraph is $e(H)$-uniform, and the conditions in Theorem 4.2 hold almost surely if $T_n$ is a random tournament.

Turning to $H$-factors, we note that it is very easy to show that any $T_n$ (with $n > 3$) has $\lfloor n/3 \rfloor$ vertex-disjoint $TT_3$. This follows from the easy observation that any $T_6$ has a $TT_3$-factor. Thus, the greedy algorithm easily produces $\lfloor n/3 \rfloor$ transitive triangles in a $T_n$. Is it true that any $T_n$ has $\lfloor n/k \rfloor$ vertex-disjoint $TT_k$, assuming $n$ is sufficiently large? Let $g(k)$ denote the minimum integer $n$ that guarantees that every $T_n$ has a $TT_k$-factor, assuming $k$ divides $n$. The (nontrivial) existence of $g(k)$ is attributed to Erdős in [70] and currently the best upper bound is $g(k) < 4^k$ obtained by Caro [18], but is probably still far from being optimal. Chen et al. [21] proved that every $T_{4m^2-6m}$ has an $H$-factor where $H$ is the star with $m$ vertices where all $m-1$ edges either all emanate from or enter the root.

We end with the following conjecture:

**Conjecture 4.14** *Every regular tournament with $6n+3$ vertices has a directed triangle factor.*

# References


[1] N. Alon, *A note on the decomposition of graphs into isomorphic matchings*, Acta Math. Acad. Sci. Hung. 42 (1983), 221–223.

[2] N. Alon, Y. Caro, and R. Yuster, *Packing and covering dense graphs*, Journal of Combinatorial Designs 6 (1998), 451–472.

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

[4] N. Alon and E. Fischer, *Refining the graph density condition for the existence of almost $K$-factors*, Ars Combinatoria 52 (1999), 296–308.

[5] N. Alon and J. H. Spencer, *The Probabilistic Method*, Second Edition, Wiley, New York, 2000.

[6] N. Alon and R. Yuster, *Threshold functions for $H$-factors*, Combinatorics, Probability, and Computing 2 (1993), 137–144.

[7] N. Alon and R. Yuster, *$H$-factors in dense graphs*, Journal of Combinatorial Theory, Series B 66 (1996), 269–282.

[8] S. Arnborg, J. Lagergren, and D. Seese, *Easy problems for tree-decomposable graphs*, Journal of Algorithms 12 (1991), 308–340.

[9] B.S. Baker, *Approximation algorithms for NP-complete problems on planar graphs*, Proceedings of the $24^{th}$ IEEE Symposium on Foundations of Computer Science (FOCS), 265–273, 1983.

[10] F. Berman, D. Johnson, T. Leighton, P.W. Shor, and L. Snyder, *Generalized planar matching*, Journal of Algorithms 11 (1990), 153–184.

[11] T. Beth, D. Jungnickel and H. Lenz, *Design Theory (2 volumes)*, Cambridge University Press, Cambridge, 1999.

[12] A. Bialostocki and Y. Roditty, *$3K_2$-decomposition of a graph*, Acta Math. Acad. Sci. Hung. 40 (1982), 201–208.

[13] B. Bollobás, *Extremal Graph Theory*, Academic Press, New York 1978.

[14] A.E. Brouwer, *Optimal packing of $K_4$'s into a $K_n$*, Journal of Combinatorial Theory, Series A 26 (1979), 278–297.

[15] A. E. Brouwer and R. M. Wilson, *The decomposition of graphs into ladder graphs*, Stiching Mathematisch Centrum (zn 97/80), Amsterdam, 1980.

[16] K. Bryś and Z. Lonc, *A complete solution of a Holyer problem*, Fourth Twente Workshop on Graph and Combinatorial Optimization, University of Twente, Enschede, The Netherlands, June 1995.

[17] A. Caprara and R. Rizzi, *Packing triangles in bounded degree graphs*, Information Processing Letters 84 (2002), 175–180.

[18] Y. Caro, *Decomposition of large combinatorial structures*, Arch. Math. 52 (1989), 289–297.

[19] Y. Caro and R. Yuster, *Packing Graphs: The packing problem solved*, Electronic Journal of Combinatorics 4 (1997), #R1.

[20] Y. Caro and R. Yuster, *Graph decomposition of slim graphs*, Graphs and Combinatorics 15 (1999), 5–19.

[21] G. Chen, X. Lu, and D.B. West, *Star-factors of tournaments*, Journal of Graph Theory 28 (1998), 141–145.

[22] D. Coppersmith and S. Winograd, *Matrix multiplication via arithmetic progressions*, Journal of Symbolic Computation 9 (1990), 251–280.

[23] K. Corrádi and A. Hajnal, *On the maximal number of independent circuits in a graph*, Acta Math. Acad. Sci. Hungar. 14 (1963), 423–439.

[24] B. Courcelle, *The monadic second order logic of graphs III: tree-decompositions, forbidden minors and complexity issues*, Informatique Théorique et Applications 26 (1992), 257–286.

[25] O. Cooley, D. Kuhn, and D. Osthus, *Perfect packings with complete graphs minus an edge*, European Journal of Combinatorics, to appear.

[26] G.A. Dirac, *Some theorems on abstract graphs*, Proceedings of the London Mathematical Society 2 (1952), 69–81.

[27] R. Diestel, *Graph Theory*, Graduate Texts in Mathematics volume 173, Springer-Verlag, Heidelberg, third edition, 2005.

[28] D. Dor and M. Tarsi, *Graph decomposition is NP-complete: A complete proof of Holyer's Conjecture*, SIAM Journal on Computing 26 (1997), 1166–1187.

[29] J. Edmonds, *Paths, trees, and flowers*, Canadian Journal of Mathematics 17 (1965), 449–467.

[30] M.H. El-Zahar, *On Circuits in Graphs*, Discrete Mathematics 50 (1984), 227–230.

[31] P. Erdős, *Some recent combinatorial problems*, preprint, November 1990.

[32] P. Erdős and A. Rényi, *On the existence of a factor of degree one of a connected random graph*, Acta Math. Acad. Sci. Hungary., 17 (1966), 359–368.

[33] P. Frankl and V. Rödl, *Near perfect coverings in graphs and hypergraphs*, European Journal of Combinatorics 6 (1985), 317–326.

[34] E. Friedgut and G. Kalai, *Every monotone graph property has a sharp threshold*, Proceedings of the American Mathematical Society 124(10) (1996), 2993–3002.

[35] A. Frieze and R. Kannan, *A simple algorithm for constructing Szemerédi's regularity partition*, Electronic Journal of Combinatorics 6 (1999), #R17.

[36] C. Godsil, J. Nešetřil, and R. Nowakowski, *The chromatic connectivity of graphs*, Graphs and Combinatorics 4 (1988), 229–233.

[37] W.T. Gowers, *Lower bounds of tower type for Szemerédi's uniformity lemma*, Geometric and Functional Analysis 7 (1997), 322–337.

[38] D. Grable, *Nearly-perfect hypergraph packing is in NC*, Information Processing Letters 60 (1996), 295–299.

[39] T. Gustavsson, *Decompositions of large graphs and digraphs with high minimum degree*, Doctoral Dissertation, Dept. of Mathematics, Univ. of Stockholm, 1991.

[40] S. Haber and M. Krivelevich, *On fractional $K$-factors of random graphs*, Random Structures and Algorithms 30 (2007), 441–463.

[41] P. E. Haxell and V. Rödl, *Integer and fractional packings in dense graphs*, Combinatorica 21 (2001), 13–38.

[42] A. Hajnal and E. Szemerédi, *Proof of a conjecture of Erdős*, in: *Combinatorial Theory and its Applications*, Vol. II (P. Erdös, A. Renyi and V. T. Sós eds.), Colloq. Math. Soc. J. Bolyai 4, North Holland, Amsterdam 1970, 601–623.

[43] I. Holyer, *The NP-completeness of some edge partition problems*, Siam Journal on Computing 10 (1981), 713–717.

[44] C.A.J. Hurkens and A. Schrijver, *On the size of systems of sets every $t$ of which have an SDR, with an application to the worst-case ratio of heuristics for packing problems*, SIAM Journal on Discrete Mathematics 2 (1989), 68–72.

[45] V. Kann, *Maximum bounded $H$-matching is MAX SNP-complete*, Information Processing Letters 49 (1994), 309–318.

[46] K. Kawarabayashi, *$K_4^-$-factors in a graph*, Journal of Graph Theory 39 (2002), 111–128.

[47] H.A. Kierstead and A.V. Kostochka, *A Short proof of the Hajnal-Szemerédi Theorem on equitable coloring*, manuscipt (2006).

[48] J.H. Kim, *Perfect matchings in random uniform hypergraphs*, Random Structures and Algorithms, 23 (2003), 111–132.

[49] T.P. Kirkman, *On a Problem in Combinatorics*, Cambridge Dublin Mathematical Journal 2 (1847), 191–204.

[50] D.G. Kirkpatrick and P. Hell, *On the complexity of a generalized matching problem*, Proceedings of the $10^{st}$ ACM Symposium on the Theory of Computing (STOC), 240–245, 1978.

[51] Y. Kohayakawa, V. Rödl, and L. Thoma. *An optimal algorithm for checking regularity*, SIAM Journal on Computing 32 (2003), 1210-1235.

[52] J. Komlós, *Tiling Turán Theorems*, Combinatorica 20 (2000), 203–218.

[53] J. Komlós, G.N. Sárkőzy, and E. Szemerédi, *Blow-up Lemma*, Combinatorica 17 (1997), 109–123.

[54] J. Komlós, G.N. Sárkőzy, and E. Szemerédi, *Proof of the Seymour conjecture for large graphs*, Annals of Combinatorics 2 (1998), 43–60.

[55] J. Komlós, G.N. Sárkőzy, and E. Szemerédi, *An algorithmic version of the Blow-up Lemma*, Random Structures and Algorithms 12 (1998), 297–310.

[56] J. Komlós, G.N. Sárkőzy, and E. Szemerédi, *Proof of the Alon-Yuster conjecture*, Discrete Mathematics 235 (2001) 255-269.

[57] J. Komlós and M. Simonovits, *Szemerédi regularity lemma and its application in Graph Theory*, in: *Paul Erdős is 80*, Proc. Coll. Bolyai Math. Soc. Vol 2. (1996), 295–352.

[58] M. Krivelevich, *Triangle factors in random graphs*, Combinatorics, Probability, and Computing 6 (1997), 337–347.

[59] D. Kühn and D. Osthus, *The minimum degree threshold for perfect graph packings*, manuscript.

[60] D. Kühn and D. Osthus, *Critical chromatic number and the complexity of perfect packings in graphs*, Proceedings of the $17^{th}$ ACM-SIAM Symposium on Discrete Algorithms (SODA), 851–859, 2006.

[61] Z. Lonc, *Edge decomposition into isomorphic copies of* $sK_{1,2}$, Journal of Combinatorial Theory, Series B 69 (1997), 164–182.

[62] L. Lovász, *On determinants, matchings, and random algorithms*, In: *Fundamentals of computation theory*, Vol. 2 pages 565–574, Akademie-Verlag, Berlin, 1979.

[63] L. Lovász and M.D. Plummer, *Matching Theory*, Annals of Discrete Mathematics 29, North-Holland, 1986.

[64] S. Micali and V.V. Vazirani, *An* $O(\sqrt{|V|} \cdot |E|)$ *algorithm for finding maximum matching in general graphs*, Proceedings of the $21^{st}$ IEEE Symposium on Foundations of Computer Science (FOCS), 17–27, 1980.

[65] G.J. Minty, *On maximal independent sets of vertices in claw-free graphs*, Journal of Combinatorial Theory, Series B 28 (1980), 284–304.

[66] M. Mucha and P. Sankowski, *Maximum Matchings via Gaussian Elimination*, Proceedings of the $45^{th}$ IEEE Symposium on Foundations of Computer Science (FOCS), 248–255, 2004.

[67] M. Mydlarz and E. Szemerédi, *Balanced coloring in Polynomial time*, European Journal of Combinatorics, to appear.

[68] C.St.J.A. Nash-Williams, *An unsolved problem concerning decomposition of graphs into triangles*, Combinatorial Theory and its Applications III. ed. P. Erdös, P. Rényi and V.T. Sós. North Holland (1970), 1179–1183.

[69] M. Priesler and M. Tarsi, *On the decomposition of the graphs into copies of* $P3 \cup tK_2$, Ars Combinatoria 35 (1993), 325–333.

[70] K.B. Reid, *Three problems on tournaments*, Graph Theory and its Applications: East and West (Proc. 2nd China-USA Intl. Conf. Graph Th., San Francisco 1989), New York Academy of Sciences Annals 576 (1989), 466–473.

[71] V. Rödl, *On a packing and covering problem*, European Journal of Combinatorics 6 (1985), 69–78.

[72] V. Rödl, M. Schacht, M.H. Siggers, and N. Tokushige, *Integer and fractional packings of hypergraphs*, Journal of Combinatorial Theory, Series B 97 (2007), 245–268.

[73] A. Rucinski, *Matching and covering the vertices of a random graph by copies of a given graph*, Discrete Mathematics 105 (1992), 185–197.

[74] A. Shokoufandeh and Y. Zhao, *On a tiling conjecture of Komlós*, Random Structures and Algorithms 23 (2003), 180–205.

[75] D.R. Stinson and H. Ferch, 2000000 *Steiner Triple Systems of order* 19, Mathematics of Computation 44 (1985), 533–535.

[76] E. Szemerédi, Regular partitions of graphs, in: *Proc. Colloque Inter. CNRS* 260, CNRS, Paris, 1978, 399–401.

[77] W.T. Tutte, *The factorization of linear graphs*, Journal of the London Mathematical Society 22 (1947), 107–111.

[78] H. Wang, *Proof of the Erdős-Faudree conjecture on quadrilaterals*, manuscript.

[79] R.M. Wilson, *Decomposition of complete graphs into subgraphs isomorphic to a given graph*, Congressus Numerantium XV (1975), 647–659.

[80] R. Yuster, *Tree decomposition of graphs*, Random Structures and Algorithms 12 (1998), 237–251.

[81] R. Yuster, *Packing and decomposition of graphs with trees*, Journal of Combinatorial Theory, Series B 78 (2000), 123–140.

[82] R. Yuster, *The decomposition threshold of bipartite graphs with minimum degree one*, Random Structures and Algorithms 21 (2002), 121–134.

[83] R. Yuster, *Families of trees decompose the random graph in an arbitrary way*, Combinatorics, Probability and Computing 13 (2004), 893–910.

[84] R. Yuster, *Integer and fractional packing of families of graphs*, Random Structures and Algorithms 26 (2005), 110–118.

[85] R. Yuster, *Asymptotically optimal $K_k$-packing of dense graphs via fractional $K_k$-decomposition*, Journal of Combinatorial Theory, Series B 95 (2005), 1–11.

[86] R. Yuster, *Fractional decompositions of dense hypergraphs*, Bulleting of the London Mathematical Society 39 (2007), 156-166.