Polynomial-Time Perfect Matchings in Dense Hypergraphs

Peter Keevash
School of Mathematical Sciences, Queen Mary, University of London
London E1 4NS
United Kingdom
p.keevash@qmul.ac.uk

Fiachra Knox
School of Mathematical Sciences, Queen Mary, University of London
London E1 4NS
United Kingdom
F.Knox@qmul.ac.uk

Richard Mycroft
School of Mathematics
University of Birmingham
Birmingham B31 4AF
United Kingdom
r.mycroft@bham.ac.uk

ABSTRACT

Let H be a k-graph on n vertices, with minimum codegree at least $n/k + cn$ for some fixed $c > 0$. In this paper we construct a polynomial-time algorithm which finds either a perfect matching in H or a certificate that none exists. This essentially solves a problem of Karpiński, Ruciński and Szymańska, who previously showed that this problem is NP-hard for a minimum codegree of $n/k - cn$. Our algorithm relies on a theoretical result of independent interest, in which we characterise any such hypergraph with no perfect matching using a family of lattice-based constructions.

Categories and Subject Descriptors

G.2.2 [**Discrete Mathematics**]: Graph Theory—*Hypergraphs, Graph Algorithms.*

Keywords

Perfect Matchings; Lattices.

1. INTRODUCTION

The question of whether a given k-uniform hypergraph (or k-graph) H contains a perfect matching (i.e. a partition of the vertex set into edges), while simple to state, is one of the key questions of combinatorics. In the graph case $k = 2$, Tutte's Theorem [18] gives necessary and sufficient conditions for H to contain a perfect matching, and Edmonds' Algorithm [5] finds such a matching in polynomial time. However, for $k \geq 3$ this problem was one of Karp's celebrated 21 NP-complete problems [6]. Results for perfect matchings have many potential practical applications; one example which has garnered interest in recent years is the 'Santa Claus' allocation problem (see [3]). Since the general problem is intractable provided P $\neq$ NP, it is natural to seek conditions on H which render the problem tractable or even guarantee that a perfect matching exists. In recent years a substantial amount of progress has been made in this direction. One well-studied class of such conditions are minimum degree conditions. In this paper we provide an algorithm that essentially eliminates the hardness gap between the sparse and dense cases for the most-studied of these conditions.

1.1 Minimum degree conditions

Suppose that H has n vertices and that k divides n (we assume this throughout, since it is a necessary condition for H to contain a perfect matching). In the graph case, a simple argument shows that a minimum degree of $n/2$ guarantees a perfect matching. Indeed, Dirac's theorem [4] states that this condition even guarantees that H contains a Hamilton cycle. For $k \geq 3$, there are several natural definitions of the minimum degree of H. Indeed, for any set $A \subseteq V(H)$, the *degree* $d(A)$ of A is the number of edges of H containing A. Then for any $1 \leq \ell \leq k-1$, the *minimum ℓ-degree* $\delta_\ell(H)$ of H is the minimum of $d(A)$ over all subsets $A \subseteq V(H)$ of size ℓ. Two cases have received particular attention: the minimum 1-degree $\delta_1(H)$ is also known as the *minimum vertex degree* of H, and the minimum $(k-1)$-degree $\delta_{k-1}(H)$ as the *minimum codegree* of H.

For sufficiently large n, Rödl, Ruciński and Szemerédi [14] determined the minimum codegree which guarantees a perfect matching in H to be exactly $n/2 - k + c$, where $c \in \{1.5, 2, 2.5, 3\}$ is an explicitly given function of n and k. They also showed that the condition $\delta_{k-1}(H) \geq n/k$ is sufficient to guarantee a matching covering all but k vertices of H (i.e. one edge away from a perfect matching). This provides a sharp contrast to the graph case, where a minimum degree of $\delta(G) \geq n/2 - \varepsilon n$ only guarantees the existence of a matching covering at least $n - 2\varepsilon n$ vertices. Many other minimum degree results for perfect matchings have also been proved; see the survey [12] for details, as well as recent papers [1], [2], [9], [10], [11] and [17].

Let $\mathbf{PM}(k, \delta)$ be the decision problem of determining whether a k-graph H with $\delta_{k-1}(H) \geq \delta n$ contains a perfect matching. Given the result of [14], a natural question to ask is the following: For which values of δ can $\mathbf{PM}(k, \delta)$ be decided in polynomial time? The main result of [14] implies that $\mathbf{PM}(k, 1/2)$ is in P. On the other hand, $\mathbf{PM}(k, 0)$ includes no degree restriction on H at all; as stated above, this was shown to be NP-complete by Karp [6]. Szymańska [16] proved that for $\delta < 1/k$ the problem $\mathbf{PM}(k, 0)$ admits a polynomial-time reduction to $\mathbf{PM}(k, \delta)$ and hence $\mathbf{PM}(k, \delta)$ is also NP-complete, while Karpiński, Ruciński and Szymańska [7] showed that there exists $\varepsilon > 0$ such that $\mathbf{PM}(k, 1/2 - \varepsilon)$ is in P. This left a hardness gap for $\mathbf{PM}(k, \delta)$ when $\delta \in [1/k, 1/2 - \varepsilon)$.

Main Result. In this paper we provide an algorithm which eliminates this hardness gap almost entirely. Moreover, it not only solves the decision problem, but also provides a perfect matching or a certificate that none exists.

THEOREM 1.1. *Fix $k \geq 3$ and $\gamma > 0$. Then there is an algorithm with running time $O(n^{2k^{k+3}+k(k-1)+1})$, which given any k-graph H on n vertices with $\delta_{k-1}(H) \geq (1/k+\gamma)n$, finds either a perfect matching or a certificate that no perfect matching exists.*

We remark that a more technical argument can reduce the running time of the algorithm to $O(n^{3k^2-7k+1})$. We give this argument in the full version of the paper.

1.2 Lattices and divisibility barriers

Theorem 1.1 relies on a result of Keevash and Mycroft [8] giving fairly general sufficient conditions which ensure a perfect matching in a k-graph. In this context, their result essentially states that if H is a k-graph on n vertices, and $\delta_{k-1}(H) \geq n/k + o(n)$, then H either contains a perfect matching or is close to one of a family of lattice-based constructions termed 'divisibility barriers'. The latter play a key role in this paper, so we now describe them in detail.

The simplest example of a divisibility barrier is the following construction, given as an extremal example in [14]: for a suitable choice of $|A|$, the k-graph formed by this construction has the highest minimum codegree of any k-graph on n vertices with no perfect matching.

CONSTRUCTION 1.2. *Let A and B be disjoint sets such that $|A|$ is odd and $|A \cup B| = n$, and let H be the k-graph on $A \cup B$ whose edges are all k-tuples which intersect A in an even number of vertices.*

To describe divisibility barriers in general, we make the following definition: for any k-graph H and any partition $\mathcal{P}$ of $V(H)$ into d parts, we define the *index vector* $\mathbf{i}_{\mathcal{P}}(S) \in \mathbb{Z}^d$ of a subset $S \subseteq V(H)$ with respect to $\mathcal{P}$ to be the vector whose coordinates are the sizes of the intersections of S with each part of $\mathcal{P}$ (note that we consider a partition to include an implicit order on its parts, so that $\mathbf{i}_{\mathcal{P}}(S)$ is well-defined). We then define $I_{\mathcal{P}}(H)$ to be the set of index vectors $\mathbf{i}_{\mathcal{P}}(e)$ of edges $e \in H$, and $L_{\mathcal{P}}(H)$ to be the lattice (i.e. additive subgroup) in $\mathbb{Z}^d$ generated by $I_{\mathcal{P}}(H)$.

A *divisibility barrier* is a k-graph H which admits a partition $\mathcal{P}$ of its vertex set V such that $\mathbf{i}_{\mathcal{P}}(V) \notin L_{\mathcal{P}}(H)$; the next proposition shows that such an H contains no perfect matching (we omit the easy proof). To see that this generalises the construction above, let $\mathcal{P}$ be the partition into parts A and B; then $L_{\mathcal{P}}(H)$ is the lattice of vectors (x, y) for which x is even, and $|A|$ being odd implies that $\mathbf{i}_{\mathcal{P}}(V) \notin L_{\mathcal{P}}(H)$.

PROPOSITION 1.3. *Let H be a k-graph with vertex set V, and let $\mathcal{P}$ partition V such that $\mathbf{i}_{\mathcal{P}}(V) \notin L_{\mathcal{P}}(H)$. Then H does not contain a perfect matching.*

A special case of the main theoretical result of this paper is the following theorem, which states that the converse of Proposition 1.3 holds for sufficiently large 3-graphs as in Theorem 1.1. Thus we obtain an essentially best possible strong stability version of the result of Rödl, Ruciński and Szemerédi [14] in the case $k = 3$.

THEOREM 1.4. *For any $\gamma > 0$ there exists $n_0 = n_0(\gamma)$ such that the following statement holds. Let H be a 3-graph on $n \geq n_0$ vertices, such that 3 divides n and $\delta_{k-1}(H) \geq (1/3+\gamma)n$, which does not contain a perfect matching. Then there is a subset $A \subseteq V(H)$ such that $|A|$ is odd but every edge of H intersects A in an even number of vertices.*

Theorem 1.4 can be used to decide $\mathbf{PM}(3, 1/3 + \gamma)$, as the existence of a subset A as in the theorem can be checked using (simpler versions of) the algorithms in Section 2. However, the case $k = 3$ is particularly simple because there is only one possible divisibility barrier; for $k \geq 4$, the next construction shows that the converse of Proposition 1.3 does not hold for general k-graphs as in Theorem 1.1.

CONSTRUCTION 1.5. *Let A, B and C be disjoint sets of vertices with $|A \cup B \cup C| = n$, $|A|, |B|, |C| = n/3 \pm 2$ and $|A| = |B| + 2$. Fix some vertex $x \in A$, and let H be the k-graph with vertex set $A \cup B \cup C$ whose edges are*

1. *any k-tuple e with $|e \cap A| = |e \cap B|$ modulo 3, and*
2. *any k-tuple $(x, z_1, \dots, z_{k-1})$ with $z_1, \dots, z_{k-1}$ in C.*

Construction 1.5 satisfies $\delta_{k-1}(H) \geq n/3-k-1$, so if $k \geq 4$ then H meets the degree condition of Theorem 1.1. Furthermore, it is not hard to see that $\mathbf{i}_{\mathcal{P}}(V(H)) \in L_{\mathcal{P}}(V(H))$ for any partition $\mathcal{P}$ of $V(H)$. In particular, if $\mathcal{P}$ is the partition of $V(H)$ into A, B and C, then $(0, 0, k), (1, 1, k-2)$ and $(1, 0, k-1)$ are all values of $\mathbf{i}_{\mathcal{P}}(e)$ for edges $e \in H$, and the above claim quickly follows in this case (recall that $k \mid n$). However, H does not contain a perfect matching. To see this, let M be a matching in H, and note that any edge $e \in M$ has $|e \cap A| = |e \cap B|$ modulo 3, except for at most one edge of M which has $|e \cap A| = |e \cap B| + 1$ modulo 3. So, letting $(i_1, i_2, i_3) = \mathbf{i}_{\mathcal{P}}(V(M))$, we have $i_1 - i_2 \in \{0, 1\}$ (modulo 3). Since this is not true of $\mathbf{i}_{\mathcal{P}}(V(H))$, we conclude that $V(M) \neq V(H)$, that is, that M is not perfect.

1.3 Approximate divisibility barriers

Our starting point will be (a special case of) a result of Keevash and Mycroft [8] on approximate divisibility barriers. First we introduce the following less restrictive degree assumption that we will use for the rest of the paper; it is not hard to see that for small $\gamma > 0$ it follows from the assumption in Theorem 1.1. (The reason for doing so will become clear at the end of Section 2.)

$$\delta_1(H) \geq \gamma n^{k-1}, \text{ and} \tag{1}$$

$$\begin{array}{l} d(A) \geq (1/k+\gamma)n \text{ for all but at most} \\ \varepsilon n^{k-1} \text{ sets } A \text{ of } k-1 \text{ vertices of } H. \end{array} \tag{2}$$

The result from [8] states that under our degree assumptions, if H does not contain a perfect matching then we can delete $o(n^k)$ edges from H to obtain a subgraph H' for which there exists a partition $\mathcal{P}$ of $V(H)$ such that $\mathbf{i}_{\mathcal{P}}(V(H)) \notin L_{\mathcal{P}}(H')$. Thus if H is far from a divisibility barrier then it has a perfect matching. On the other hand, if H is itself a divisibility barrier then H does not have a perfect matching.

New Result. The main theoretical contribution of this paper is to fill the gap between these cases, by giving a necessary and sufficient condition for the existence of a perfect matching under our degree assumptions.

For the statement we need the following definitions. First, we say that a lattice L is *transferral-free* if it does not contain any vector with $+1$ in one co-ordinate, -1 in another, and all remaining co-ordinates being zero. Second, we call a lattice L an *edge-lattice* if it is generated by vectors whose co-ordinates are non-negative and sum to k; thus $L_{\mathcal{P}}(H)$ is an edge-lattice for any k-graph H and partition $\mathcal{P}$.

THEOREM 1.6. *For any k, γ, C_0, there exist $\varepsilon = \varepsilon(k, \gamma, C_0)$ and $n_0 = n_0(k, \gamma, C_0)$ such that, for any C with $2k^{k+2} \leq C \leq C_0$, any $n \geq n_0$ divisible by k, and any k-graph H on n vertices satisfying (1) and (2), H has a perfect matching if and only if the following condition holds:*

(*) If $\mathcal{P}$ is a partition of $V(H)$ into d parts, where $1 \leq d < k$, and $L \subseteq \mathbb{Z}^d$ is a transferral-free edge-lattice such that any matching M in H formed of edges $e \in H$ with $\mathbf{i}_{\mathcal{P}}(e) \notin L$ has size less than C, then there exists a matching M' of size at most $k-2$ such that $\mathbf{i}_{\mathcal{P}}(V(H) \backslash V(M')) \in L$.

Our algorithm for the decision problem is essentially an exhaustive check of condition (*) in Theorem 1.6, although we also need to provide an algorithm to efficiently list the partitions $\mathcal{P}$ as in the theorem. Thus Theorem 1.6 is required to prove correctness of the algorithm; we also believe it to be of independent interest. Our proof of Theorem 1.6 relies on the main result of [8] as well as a substantial amount of geometric, probabilistic and combinatorial theory.

1.4 Contents and notation

In the next section we present the algorithmic details of the results in this introduction. That is, we assume Theorem 1.6 and deduce Theorem 1.1. In Section 3 we outline the proof of Theorem 1.6, presenting only the key ideas and steps, as the full proof is too long and technical to include here. In the final section we make some concluding remarks and note analogues of our main results pertaining to multipartite hypergraphs, which can be proved in a similar way.

We write $[r]$ to denote the set of integers from 1 to r, and $x \ll y$ to mean for any y there exists x_0 such that for any $x \leq x_0$ the following statement holds. Similar statements with more constants are defined similarly. Also, we write $a = b \pm c$ to mean $b - c \leq a \leq b + c$.

2. ALGORITHMS AND ANALYSIS

We start with the following theorem, which can be used to solve the decision problem of determining whether or not H has a perfect matching.

THEOREM 2.1. *Fix $k \geq 3$ and $\gamma > 0$. Then there exists $\varepsilon = \varepsilon(k, \gamma)$ such that for any k-graph H on n vertices which satisfies (1) and (2), Procedure DeterminePM determines correctly whether or not H contains a perfect matching. Furthermore, it will do so in time $O(n^{2k^{k+3}+k(k-2)})$.*

Procedure DeterminePM is essentially an exhaustive check of condition (*) in Theorem 1.6. It is clear that the ranges of M, d, L and M' in the procedure can be listed by brute force in polynomial time. However, brute force cannot be used for $\mathcal{P}$, as there would be exponentially many possibilities to consider, so first we provide an algorithm to construct all possibilities for $\mathcal{P}$.

We imagine each vertex class V_j to be a 'bin' to which vertices may be assigned, and keep track of a set U of vertices yet to be assigned to a vertex class. So initially we take each V_j to be empty and $U = V(H)$. The procedure operates as a *search tree*; at certain points the instruction is to branch over a range of possibilities. This means to select one of these possibilities and continue with this choice, then, when the algorithm halts, to return to the branch point, select the next possibility, and so forth. Each branch may produce an output partition; the output of the procedure consists of all output partitions. An informal statement of our procedure is that we generate partitions by repeatedly branching over all possible assignments of a vertex to a partition class, exploring all consequences of each assignment before branching again. Furthermore, we only branch over assignments of vertices which satisfy the following condition. Given a set of assigned vertices, we call an unassigned vertex x *reliable* if there exists a set B of $k-2$ assigned vertices such that $d(x \cup B) \geq (1/k + \gamma)n$.

Procedure DeterminePM

Data: An integer $k \geq 3$, a constant $\gamma > 0$ and a k-graph $H = (V, E)$.
Result: Determines correctly whether or not H has a perfect matching provided H satisfies (1) and (2) for $\varepsilon = \varepsilon(k, \gamma, 2k^{k+3})$.

if $|V| < n_0(k, \gamma, 2k^{k+3})$ **then**
  Test every possible perfect matching in H, and halt with appropriate output.
foreach *matching M in H of size at most $2k^{k+2}$, integer $1 \leq d < k$, transferral-free edge-lattice $L \subseteq \mathbb{Z}^d$ and partition $\mathcal{P}$ of V into d parts so that any edge $e \in H$ which does not intersect $V(M)$ has $\mathbf{i}_{\mathcal{P}}(e) \in L$* **do**
  if *there is no matching $M' \subseteq H$ of size at most $k-2$ such that $\mathbf{i}_{\mathcal{P}}(V \setminus V(M') \in L$* **then**
    Output "no perfect matching" and halt.
Output "perfect matching" and halt.

LEMMA 2.2. *Fix $0 < \varepsilon \ll \gamma$. Let H be a k-graph on a vertex set V of size n which satisfies (1) and (2), let $1 \leq d < k$ and let $L \subseteq \mathbb{Z}^d$ be a transferral-free edge-lattice. Then there are at most d^{2k-2} partitions $\mathcal{P}$ of V such that $\mathbf{i}_{\mathcal{P}}(e) \in L$ for every $e \in H$, and Procedure ListPartitions lists them in time $O(n^{k+1})$.*

PROOF. First we note that the instruction 'Assign x to V_j' in Procedure ListPartitions is well-defined. Indeed, since L is transferral-free, for any set $S \subseteq V(H)$ there is at most one $j \in [k]$ such that $\mathbf{i}_{\mathcal{P}}(S) + \mathbf{u}_j \in L$.

Next we show that if the number of assigned vertices is at least $(1/k + \gamma)n$ and at most $(1-\gamma)n$ then there is always a reliable unassigned vertex. To see this, note that the number of sets $x \cup B$, where x is unassigned and B is a set of $k-2$ assigned vertices, is at least $\gamma n \binom{n/k+\gamma n}{k-2} > \varepsilon n^{k-1}$. Hence some such $x \cup B$ has degree at least $(1/k + \gamma)n$, and so x is reliable.

The final line of the procedure ensures that any partition $\mathcal{P}$ of $V(H)$ which is output has that property that $\mathbf{i}_{\mathcal{P}}(e) \in L$ for every $e \in H$. The converse is also true: any partition $\mathcal{P}$ of $V(H)$ such that $\mathbf{i}_{\mathcal{P}}(e) \in L$ for every $e \in H$ will be output by some branch of the procedure. To see this, consider the branch of the procedure in which, at each branch point, the vertex x under consideration is assigned to the vertex class in which it lies in $\mathcal{P}$. By our initial remark, every other vertex of H must also be assigned to the vertex class in which

Procedure ListPartitions

Data: A k-graph H and a transferral-free edge-lattice $L \subseteq \mathbb{Z}^d$.

Result: Outputs all partitions of $V(H)$ with $\mathbf{i}_\mathcal{P}(e) \in L$ for every $e \in H$.

Set $U = V(H)$.
Choose $A \subseteq V(H)$ of size $k-1$ such that $d(A) \geq \delta n$.
Branch over all possible assignments of vertices in A.
while $U \neq \emptyset$ **do**
  if $xy_1 \dots y_{k-1} \in H$ *for some vertices* $x \in U$ *and* $y_1, \dots, y_{k-1} \notin U$ **then**
    if $\mathbf{i}_\mathcal{P}(y_1 \dots y_{k-1}) + \mathbf{u}_j \in L$ *for some* $j \in [k]$ **then**
      Assign x to V_j and remove x from U.
    else
      Halt with no output.
  else
    Choose $x \in U$ which is reliable;
    Branch over all possible assignments of x.
if $\mathbf{i}_\mathcal{P}(e) \in L$ *for every* $e \in H$ **then** halt with output $\mathcal{P}$.

it lies in $\mathcal{P}$. We conclude that Procedure ListPartitions indeed runs correctly, returning all partitions $\mathcal{P}$ of V such that $\mathbf{i}_\mathcal{P}(e) \in L$ for every $e \in H$.

It remains to bound the number of such partitions. Observe that after branching initially over all possible assignments of A, at least $(1/k+\gamma)n$ vertices will be assigned (all the neighbours of A) before the procedure branches again. Further, the procedure will no longer branch once there are fewer than γn unassigned vertices remaining. Indeed, in this case for any unassigned vertex x there are fewer than γn^{k-1} edges containing x and another unassigned vertex. Thus condition (1) implies that every unassigned vertex x is contained in some edge $xy_1 \dots y_{k-1}$ of H where $y_1, \dots, y_{k-1}$ are assigned. Hence when branching, we may always choose a reliable vertex as stated in the procedure.

Consider some x over which the procedure branches. Then there can be no edge $xy_1 \dots y_{k-1}$ of H where $y_1, \dots, y_{k-1}$ are assigned. Suppose that this is the case, and let $B = y_1 y_2 \dots y_{k-2}$ be a set of assigned vertices such that $d(x \cup B) \geq \delta n$. (Such a B must exist since we chose x to be reliable.) None of the $(1/k+\gamma)n$ vertices v such that $xvy_1 \dots y_{k-2}$ is an edge of H can have been assigned, and each will be assigned before the next branch of the procedure. We conclude that after any branch in the procedure and before the next branch, at least $(1/k+\gamma)n$ vertices are assigned.

Hence the search tree has depth at most k. Since the degree of the root of the search tree is at most d^{k-1} and since the degree of every other vertex is the number of vertex classes d, it follows that the search tree has at most d^{2k-2} leaf nodes. At most one partition is output at each leaf node, so at most d^{2k-2} partitions $\mathcal{P}$ will be output by the algorithm, as required. Furthermore, over all branches there will be at most $d^{2k-2}n$ iterations of the *while* loop, and the condition of the first *if* statement takes $O(n^k)$ operations to check, and so the overall running time is $O(n^{k+1})$. □

Proof of Theorem 2.1. Fix $k \geq 3$ and $0 < \varepsilon \ll \gamma$, and let H be a k-graph on a vertex set V of size n which satisfies (1) and (2). We first show that Procedure DeterminePM determines correctly whether or not H has a perfect matching. This is trivial if $n < n_0 = n_0(k, \gamma, 2k^{k+3})$, so assume without loss of generality that $n \geq n_0$.

Suppose first that H does not contain a perfect matching. Then by Theorem 1.6 applied with $C = 2k^{k+2}$ there exists a partition $\mathcal{P}$ of V into d parts, where $1 \leq d < k$, and a transferral-free edge-lattice $L \subseteq \mathbb{Z}^d$ such that

(i) any matching M in H formed of edges $e \in H$ with $\mathbf{i}_\mathcal{P}(e) \notin L$ has size less than $2k^{k+2}$, and

(ii) no matching M' in H of size at most $k-2$ has $\mathbf{i}_\mathcal{P}(V \setminus V(M')) \in L$.

Let M be a maximal matching in H formed of edges $e \in H$ with $\mathbf{i}_\mathcal{P}(e) \notin L$, so $|M| < 2k^{k+2}$. By maximality of M any edge $e \in H$ which does not intersect $V(M)$ has $\mathbf{i}_\mathcal{P}(e) \in L$. Therefore, Procedure DeterminePM will consider M, d, and L and $\mathcal{P}$ in some iteration, and will then find that no matching $M' \subseteq H$ of size at most $k-2$ has $\mathbf{i}_\mathcal{P}(V \setminus V(M')) \in L$. Hence the procedure will output that H does not contain a perfect matching, as required.

Now suppose that H does contain a perfect matching, and suppose for a contradiction that Procedure DeterminePM incorrectly claims that this is not the case. This can only arise if Procedure DeterminePM considers a matching M in H of size at most $2k^{k+2}$, an integer $1 \leq d < k$, a transferral-free edge-lattice $L \subseteq \mathbb{Z}^d$ and a partition $\mathcal{P}$ of V into d parts so that any edge $e \in H$ which does not intersect $V(M)$ has $\mathbf{i}_\mathcal{P}(e) \in L$, and finds that no matching $M' \subseteq H$ of size at most $k-2$ has $\mathbf{i}_\mathcal{P}(V \setminus V(M')) \in L$. Since any edge $e \in H$ with $\mathbf{i}_\mathcal{P}(e) \notin L$ must intersect $V(M)$, any matching in H formed of edges $e \in H$ with $\mathbf{i}_\mathcal{P}(e) \notin L$ has size at most $k|M| \leq 2k^{k+3}$. So $\mathcal{P}$, d and L are as in condition (*) of Theorem 1.6 with $C = 2k^{k+3}$; since H has a perfect matching this is a contradiction.

It remains only to show that Procedure DeterminePM must terminate in polynomial time. To see this, note that there are at most $n^{k \cdot 2k^{k+2}}$ choices of matchings M, and these can be generated in time $O(n^{2k^{k+3}})$ by considering each set of at most $2k^{k+2}$ edges in turn. Also, there are only a constant number of choices for d and L, and these can be generated in constant time. Indeed, since L is an edge-lattice, it is generated by vectors with non-negative coordinates which sum to k; the number of such generating sets is bounded by a function of k. Lemma 2.2 applied to $H \setminus V(M)$ shows that the number of choices for $\mathcal{P}$ is also constant, and the list can be generated in time $O(n^{k+1})$. Finally, it takes time $O(n^{k(k-2)})$ to evaluate the truth of the condition of the central *if* statement. We conclude that in any case this procedure will run in time $O(n^{2k^{k+3}+k(k-2)})$, as required. □

Now we will deduce our main result, Theorem 1.1. We start by using Procedure DeterminePM to check whether H contains a perfect matching. If H has no perfect matching then this is certified by some M, L and $\mathcal{P}$ as in the procedure. So suppose that H does contain a perfect matching. How can we find it? A naive attempt at a proof is the following well-known idea. We examine each edge e of H in turn and use the same procedure to test whether deleting the vertices of e would still leave a perfect matching in the remainder of H, in which case we say that e is *safe*. There must be some safe edge e, which we add to our matching, then repeat this process, until the number of vertices falls below n_0, at which point our procedure no longer works and we resort to a constant-time brute force search.

The problem with this naive attempt is that as we remove edges, the minimum codegree may become too low to apply Procedure DeterminePM, and then the process cannot continue. To motivate the solution to this problem, suppose that we have oracle access to a uniformly random edge from some perfect matching. Such an edge is safe, and if we repeatedly remove such random edges, standard large deviation estimates show that with high probability the minimum codegree condition is preserved (replacing γ by $\gamma/2$, say). As an aside, we note that since Linear Programming is in P, we can construct a distribution (p_e) on the safe edges such that $\sum_{e:x\in e} p_e = k/n$ for every vertex x; using this distribution instead of the oracle provides a randomised algorithm for finding a perfect matching.

Our actual algorithm is obtained by derandomising the oracle algorithm. Instead of a minimum codegree condition, we bound the sum of squares of codegree 'deficiencies', which is essentially the condition (2) considered above. We also need to introduce the vertex degree condition (1), otherwise we do not have an effective bound on the number of partitions $\mathcal{P}$ in Lemma 2.2. Conditions (i) and (ii) in the following lemma effectively serve as proxies for (2) and (1).

Lemma 2.3. *Suppose that $\varepsilon \le \varepsilon(k,\gamma)$ and $n \ge n_1(k,\gamma)$. Let V be a vertex set of size n and let H be a k-graph on V. Set $t_A = \max(0, (1/k+\gamma)n - d(A))$ for each set A of $k-1$ vertices of H. Suppose that*

(i) $\sum_{A\in\binom{V}{k-1}} t_A^2 < \varepsilon\gamma^2 n^{k+1}/4$,

(ii) $n\binom{n-1}{k-2}/3k - \delta_1(H) + \sum_{A\in\binom{V}{k-1}} t_A^2/\sqrt{\varepsilon}\gamma^2 n^2 < \sqrt{\varepsilon} n^{k-1}$,

(iii) H contains a perfect matching.

Then we can find, in time $O(n^{2k^{k+3}+k(k-1)})$, an edge $e \in H$ such that (i), (ii) and (iii) also hold for $H \setminus e$ with $n-k$ in place of n.

The proof of Lemma 2.3 involves messy calculations, so we just describe the idea. While $\delta_1(H) > n\binom{n-1}{k-2}/3k$ we only need to maintain condition (i); then an averaging argument shows that the required edge exists. On the other hand, if there is a vertex of small degree we can remove any edge containing it: this so greatly decreases $\sum_{A\in\binom{V}{k-1}} t_A^2$ as to compensate for any further decrease in $\delta_1(H)$.

Proof of Theorem 1.1. We begin by running Procedure DeterminePM to confirm that H contains a perfect matching; if it does not, then we obtain a matching M in H of size at most $2k^{k+2}$, a transferral-free lattice L and a partition $\mathcal{P}$ of $V(H)$ such that $L_{\mathcal{P}}(H \setminus V(M)) \subseteq L$, but $\mathbf{i}_{\mathcal{P}}(V \setminus V(M')) \notin L$ for any matching M' in H of size at most $k-2$, providing a certificate that no perfect matching exists. If H does contain a perfect matching, then we proceed by using repeated applications of Lemma 2.3 to delete edges of H (along with their vertices). Condition (iii) of Lemma 2.3 is satisfied by assumption and conditions (i) and (ii) follow easily from the codegree condition on H. Since Lemma 2.3 ensures that its conditions are preserved after an edge is deleted, we may repeat until $n < n_1(k,\gamma)$. At this point we use the brute-force algorithm to find a perfect matching in the remainder of H. Together with the deleted edges this forms a perfect matching in H. $\square$

3. OUTLINES OF THE PROOFS

In this section we sketch the proof of Theorem 1.6, which provides the theoretical basis for our algorithm.

3.1 Edge-detection lattices

We start by explaining and stating the theorem of Keevash and Mycroft on approximate divisibility barriers. Given a k-graph H and a partition $\mathcal{P}$ of $V(H)$, the lattice $L_{\mathcal{P}}(H)$ can be seen as 'detecting' where edges of H lie with respect to $\mathcal{P}$. However, the information conveyed by $L_{\mathcal{P}}(H)$ is by itself insufficient, as shown by the k-graph formed in Construction 1.5. Indeed, in that instance $L_{\mathcal{P}}(H)$ was complete, but some index vectors did not represent enough edges: specifically, H did not contain two disjoint edges with different numbers of vertices in A and B modulo 3.

Thus in the proof of Theorem 1.6 we will frequently want to know which index vectors are represented by many edges. This leads to two important definitions analogous to those of $I_{\mathcal{P}}(H)$ and $L_{\mathcal{P}}(H)$: we define $I^{\mu}_{\mathcal{P}}(H)$ to be the set of index vectors $\mathbf{i}$ for which at least μn^k edges $e \in H$ have $\mathbf{i}_{\mathcal{P}}(e) = \mathbf{i}$, and then we define $L^{\mu}_{\mathcal{P}}(H)$ to be the lattice in $\mathbb{Z}^d$ generated by this set (where d is the number of parts of $\mathcal{P}$). So $I^{\mu}_{\mathcal{P}}(H)$ and $L^{\mu}_{\mathcal{P}}(H)$ reflect how the edges of H lie with respect to $\mathcal{P}$ with 'weaker detection of edges'. Note that for any $\mu < \mu'$ we have $L^{\mu'}_{\mathcal{P}}(H) \subseteq L^{\mu}_{\mathcal{P}}(H) \subseteq L_{\mathcal{P}}(H)$.

We say that the lattice $L^{\mu}_{\mathcal{P}}(H)$ for a k-graph H is *complete* if it contains all vectors in $\mathbb{Z}^d$ whose co-ordinates sum to k. The point of this definition is that if $L^{\mu}_{\mathcal{P}}(H)$ is complete then we must have $\mathbf{i}_{\mathcal{P}}(V) \in L^{\mu}_{\mathcal{P}}(H)$ (recall that $k \mid n$), so to form a divisibility barrier we would have to delete at least μn^k edges. The following theorem is a consequence of a more general result of Keevash and Mycroft [8].

Theorem 3.1. *(Keevash and Mycroft [8]) Suppose that $1/n \ll \varepsilon, \mu \ll \gamma, 1/k$, and let H be a k-graph on a vertex set V of size n which satisfies the degree conditions (1) and (2). If $L^{\mu}_{\mathcal{P}}(H)$ is complete for every partition $\mathcal{P}$ of V into parts of size at least n/k then H contains a perfect matching.*

3.2 Fullness

Next we explain our earlier definition of transferral-free lattices and the associated concept of fullness. Let H be a k-graph with a partition $\mathcal{P}$ of $V(H)$ such that $L^{\mu}_{\mathcal{P}}(H)$ is incomplete. For any part $X \in \mathcal{P}$, write $\mathbf{u}_X$ for the index vector with respect to $\mathcal{P}$ which is 1 on X and zero on every other part of $\mathcal{P}$. Suppose that $L^{\mu}_{\mathcal{P}}(H)$ contains $\mathbf{u}_X - \mathbf{u}_Y$, for some $X \neq Y \in \mathcal{P}$, and let $\mathcal{P}'$ be the partition formed from $\mathcal{P}$ by merging the parts X and Y. Then it is not hard to see that $L^{2\mu}_{\mathcal{P}'}(H)$ is also incomplete. To say that $L^{\mu}_{\mathcal{P}}(H)$ is transferral-free is then to say that no such merging is possible; that is, $\mathbf{u}_X - \mathbf{u}_Y \notin L^{\mu}_{\mathcal{P}}(H)$ for any parts $X \neq Y$ of $\mathcal{P}$. So we can view a partition which gives rise to a transferral-free lattice as the simplest version of a given approximate divisibility barrier.

We say that an edge-lattice L with respect to $\mathcal{P}$ is *full* if it is transferral-free and for each index vector $\mathbf{i}$ of a $(k-1)$-set, there is a (unique) part X such that $\mathbf{i} + \mathbf{u}_X \in L$. It is not hard to show that if H satisfies (1) and (2) and $L^{\mu}_{\mathcal{P}}(H)$ is transferral-free, then $L^{\mu}_{\mathcal{P}}(H)$ is also full. The following lemma expresses an important property of full lattices, namely that they are subgroups of finite index in the *maximal lattice* $L^{max}_{\mathcal{P}}$ of vectors in $\mathbb{Z}^{\mathcal{P}}$ with co-ordinate sum divisible by k.

LEMMA 3.2. *Let $k \geq 3$ and suppose L is a full edge-lattice with respect to a partition $\mathcal{P}$ of a set V. Then*

(i) For any $\mathbf{i} \in L_{\mathcal{P}}^{max}$ and $X \in \mathcal{P}$ there is $X' \in \mathcal{P}$ such that $\mathbf{i} - \mathbf{u}_X + \mathbf{u}_{X'} \in L$,

(ii) $|L_{\mathcal{P}}^{max}/L| \leq |\mathcal{P}|$.

PROOF. We first show that:
(+) for any $X_1, X_1', X_2 \in \mathcal{P}$, there exists $X_2' \in \mathcal{P}$ such that $\mathbf{u}_{X_1} + \mathbf{u}_{X_2} - \mathbf{u}_{X_1'} - \mathbf{u}_{X_2'} \in L$.

To see this, fix any index vector $\mathbf{i}'$ of a $(k-3)$-set in V. Since L is full, we can find $Y \in \mathcal{P}$ such that $\mathbf{i}' + \mathbf{u}_{X_1} + \mathbf{u}_{X_2} + \mathbf{u}_Y \in L$. Similarly, we can find $X_2' \in \mathcal{P}$ such that $\mathbf{i}' + \mathbf{u}_{X_1'} + \mathbf{u}_Y + \mathbf{u}_{X_2'} \in L$. Then $\mathbf{u}_{X_1} + \mathbf{u}_{X_2} - \mathbf{u}_{X_1'} - \mathbf{u}_{X_2'}$ is the difference of these two index vectors and hence lies in L.

To prove (i), consider $\mathbf{i}' \in L$ with $\sum_{Z \in \mathcal{P}} i_Z' = \sum_{Z \in \mathcal{P}} i_Z$ that minimises $\sum_{Z \in \mathcal{P}} |i_Z' - i_Z|$ subject to this condition. Note that $\sum_{Z \in \mathcal{P}} |i_Z' - i_Z| \leq 2$. For suppose otherwise, and choose X_1, X_1', X_2 such that $i_{X_1} - i_{X_1}' > 0$, $i_{X_2} - i_{X_2}' > 0$ and $i_{X_1'} - i_{X_1'}' < 0$ (or $i_{X_1} - i_{X_1}' > 1$ and $i_{X_1'} - i_{X_1'}' < 0$ if $X_1 = X_2$). Using (+), let $X_2' \in \mathcal{P}$ be such that $\mathbf{i}^* = \mathbf{u}_{X_1} + \mathbf{u}_{X_2} - \mathbf{u}_{X_1'} - \mathbf{u}_{X_2'} \in L$. Then $\mathbf{i}' + \mathbf{i}^*$ contradicts our choice of $\mathbf{i}'$. It follows that $\mathbf{i}' = \mathbf{i} - \mathbf{u}_Y + \mathbf{u}_Y'$ for some $Y, Y' \in \mathcal{P}$. Using (+) again, we can choose X' such that $\mathbf{i}^{**} = \mathbf{u}_X + \mathbf{u}_Y - \mathbf{u}_{X'} - \mathbf{u}_{Y'} \in L$. Therefore $\mathbf{i} + \mathbf{u}_X - \mathbf{u}_{X'} = \mathbf{i}' + \mathbf{i}^{**} \in L$, as claimed.

To prove (ii), fix any index vector $\mathbf{i}'$ of a $(k-1)$-set in V. We claim that every coset $L + \mathbf{v}$ of L in $L_{\mathcal{P}}^{max}$ contains an index vector $\mathbf{i}' + \mathbf{u}_{X'}$ for some $X' \in \mathcal{P}$; this clearly suffices to prove the lemma. To see this, note that since I is full there exists $X \in \mathcal{P}$ such that $\mathbf{i}' + \mathbf{u}_X \in L$. By the previous claim, we obtain $X' \in \mathcal{P}$ such that $-\mathbf{v} - \mathbf{u}_X + \mathbf{u}_{X'} \in L$. But now $\mathbf{i}' + \mathbf{u}_{X'} = (\mathbf{i}' + \mathbf{u}_X) + (-\mathbf{v} - \mathbf{u}_X + \mathbf{u}_{X'}) + \mathbf{v} \in L + \mathbf{v}$, as required. $\square$

3.3 The key lemma and robust maximality

A key part of the proof of Theorem 1.6 is Lemma 3.4, which we shall state after some motivation and definitions. This lemma generalises Theorem 3.1 in two ways. First, instead of the condition of Theorem 3.1 that $L_{\mathcal{P}}^{\mu}(H)$ must be complete, we now only require that $\mathbf{i}_{\mathcal{P}}(V) \in L_{\mathcal{P}}^{\mu}(H)$ (as discussed earlier, this must be true if $L_{\mathcal{P}}^{\mu}(H)$ is complete). Furthermore, whereas the condition of Theorem 3.1 must hold for any partition $\mathcal{P}$ of $V(H)$ into sufficiently large parts, we now only require that $\mathbf{i}_{\mathcal{P}}(V) \in L_{\mathcal{P}}^{\mu}(H)$ for a single partition $\mathcal{P}$ which meets two additional requirements.

The first requirement is that every vertex $v \in V$ must lie in many edges $e \in H$ with $\mathbf{i}_{\mathcal{P}}(e) \in L_{\mathcal{P}}^{\mu}(H)$. This condition can be seen as ensuring that each vertex of H lies in the 'correct part' of $\mathcal{P}$. For example, if we fix $\mu > 0$ and take the k-graph from Construction 1.2 and move a single vertex from part A to part B (but do not change the edge set of H), then the only edges whose index changes are the fewer than n^{k-1} edges which contain v. So $L_{\mathcal{P}}^{\mu}(H)$ is unchanged, and H doesn't have a perfect matching (since H itself is unchanged), but we now have $\mathbf{i}_{\mathcal{P}}(V) \in L_{\mathcal{P}}^{\mu}(H)$, due to v being in the 'wrong part' of $\mathcal{P}$.

Secondly, we must assume that $L_{\mathcal{P}}^{\mu}(H)$ is transferral-free. However this requirement, while necessary, is not sufficient, as the following example will show. Fix $k \geq 5$ and suppose that $\mathcal{P}$ divides V into two parts, W_1 and W_2. Further, let $\mathcal{Q}$ be a refinement of $\mathcal{P}$ which divides W_1 into V_{11} and V_{12} and W_2 into V_{21} and V_{22}. Suppose further that $|W_1|$ is even, but that $|V_{11} \cup V_{21}|$ is odd. Now let H be the k-graph on V whose edges are precisely the k-subsets of V which contain an even number of vertices in W_1 and an even number of vertices in $V_{11} \cup V_{21}$. It is easy to see that $L_{\mathcal{P}}^{\mu}(H)$ is full, that $\delta_{k-1}(H) \geq (1/k + \gamma)n$ and that $\mathbf{i}_{\mathcal{P}}(V) \in L_{\mathcal{P}}^{\mu}(H)$. In this case, the conditions relating to $\mathcal{P}$ do not preclude the existence of a perfect matching, but the conditions relating to $\mathcal{Q}$ do. Indeed, H cannot contain a perfect matching since it is a subgraph of the k-graph described in Construction 1.2, where $A = V_{11} \cup V_{21}$ and $B = V_{12} \cup V_{22}$.

To avoid this kind of situation, we insist that there is no strict refinement of $\mathcal{P}$ into not-too-small parts such that $L_{\mathcal{P}}^{\mu}(H)$ is transferral-free. Note that the trivial partition into a single part satisfies this requirement if and only if H is not close to a divisibility barrier. However, being maximal transferral-free is a rather fragile property, that can be destroyed by even a small change in μ or alteration to H. In the course of our proof we will need to remove small matchings from H in order to cover vertices which are exceptional in various ways. We also need a property which is preserved (with high probability) when H is replaced by an induced subgraph on a randomly chosen set of vertices. Thus we require a stronger property, namely that for some $\mu' \gg \mu$ (i.e. even at a much weaker 'detection threshold') the lattice $L_{\mathcal{Q}}^{\mu'}(H)$ is not transferral-free for any strict refinement $\mathcal{Q}$ of $\mathcal{P}$ into not-too-small parts. Together these considerations lead to the following key definition of robust maximality.

DEFINITION 3.3. *Let H be a k-graph on a vertex set V of size n. We say that a partition $\mathcal{P}$ of V is* (c, μ, μ')-robustly maximal *with respect to H if*

(i) $L_{\mathcal{P}}^{\mu}(H)$ is transferral-free, and

(ii) for any partition $\mathcal{P}^+$ of V which strictly refines $\mathcal{P}$ into parts of size at least cn, the lattice $L_{\mathcal{P}^+}^{\mu'}(H)$ is not transferral-free.

We can now state the key lemma. We discuss its proof later, but first we will analyse the property of robust maximality in more detail, and describe how Theorem 1.6 follows from Lemma 3.4.

LEMMA 3.4. *Suppose that $1/n \ll \varepsilon \ll \mu \ll \mu' \ll c, d, \gamma \ll 1/k$, and let H be a k-graph whose vertex set V has size n and which satisfies the degree condition (2). Also let $\mathcal{P}$ be a partition of V with parts of size at least $(1/k + \gamma)n$ which is (c, μ, μ')-robustly maximal with respect to H. Suppose that*

(i) for any vertex $x \in V$ there are at least dn^{k-1} edges $e \in H$ with $x \in e$ and $\mathbf{i}_{\mathcal{P}}(e) \in L_{\mathcal{P}}^{\mu}(H)$, and

(ii) $\mathbf{i}_{\mathcal{P}}(V) \in L_{\mathcal{P}}^{\mu}(H)$.

Then H contains a perfect matching.

3.4 Properties of robust maximality

Lemma 3.4 will only be useful if we can actually find a (c, μ, μ')-robustly maximal partition of the vertex set of a k-graph H for some small constants μ, μ' and c with $\mu \ll \mu' \ll c$. This is achieved by the following proposition, which even allows us to refine any given partition $\mathcal{P}$ to a robustly maximal partition. The proof is straightforward; we repeatedly refine $\mathcal{P}$ into partitions with parts of size at least cn which give rise to transferral-free lattices. When this is no longer possible, the final refinement we obtain will be robustly maximal by definition.

Proposition 3.5. *Let $k \geq 2$ be an integer and $c > 0$ be a constant. Let $s = \lfloor 1/c \rfloor$ and fix constants $0 < \mu_1 < \cdots < \mu_{s+1}$. Suppose that H is a k-graph on a vertex set V of size n, and $\mathcal{P}$ is a partition of V with parts of size at least cn such that $L^{\mu_1}_{\mathcal{P}}(H)$ is transferral-free. Then there exists $t \in [s]$ and a partition $\mathcal{P}'$ of V with parts of size at least cn which refines $\mathcal{P}$ and is (c, μ_t, μ_{t+1})-robustly maximal with respect to H.*

The next proposition demonstrates two important 'inheritance' properties of robust maximality. Namely, if $\mathcal{P}$ is a robustly maximal partition of the vertex set V of a k-graph H, and we form a subgraph $H' \subseteq H$ either by deleting only a small number of vertices and edges from H, or by restricting H to a random subset of V, then (the restriction of) $\mathcal{P}$ will be robustly maximal with respect to H'. Actually, we prove a significantly stronger version of (ii), allowing us to specify how many vertices of S should be taken from each part of a partition $\mathcal{Q}$ of V.

Proposition 3.6. *Suppose that $1/n \ll 1/n' \ll \mu \ll \mu', \alpha \ll c, 1/k$. Let H be a k-graph on n vertices with m edges, and $\mathcal{P}$ be a (c, μ, μ')-robustly maximal partition of $V(H)$ with respect to H, with parts of size at least cn. Then:*

(i) If H' is a subgraph of H with at least $(1-\alpha)n$ vertices and at least $m - \alpha n^k$ edges, then the restriction of $\mathcal{P}$ to $V(H')$ is a $(c+2\alpha, \mu+2\alpha, \mu'-\alpha)$-robustly maximal partition with respect to H'.

(ii) Suppose that a set $S \subseteq V(H)$ of size n' is chosen uniformly at random. Then with probability $1 - o(1)$ the restriction of $\mathcal{P}$ to S is a $(2c, \mu/c, (\mu')^3)$-robustly maximal partition with respect to $H[S]$.

Proof sketch. For (i) we simply count how many edges of any given index can be removed in forming H', and the result follows. The first part of (ii) is also straightforward: a standard Chernoff bound shows that condition (i) of Definition 3.3 holds for the restriction of $\mathcal{P}$ with high probability. A similar argument shows that condition (ii) of Definition 3.3 holds for any specific partition $\mathcal{P}^+$ which refine this partition, but unfortunately there are too many possibilities for $\mathcal{P}^+$ to apply a union bound. Instead we proceed by a technical argument using weak hypergraph regularity. □

3.5 The proof of Theorem 1.6

Between Lemma 3.4 and Prop 1.3 we are moving towards a characterisation of k-graphs H which satisfy the degree conditions (1) and (2). Indeed, fix such an H and suppose that $\mathcal{P}$ is a (c, μ, μ')-robustly maximal partition of $V := V(H)$ with parts of size at least cn such that every vertex lies in many edges with $\mathbf{i}_{\mathcal{P}}(e) \in L^{\mu}_{\mathcal{P}}(H)$. Then by Proposition 1.5 H cannot contain a perfect matching if $\mathbf{i}_{\mathcal{P}}(e) \notin L_{\mathcal{P}}(H)$. On the other hand, if $\mathbf{i}_{\mathcal{P}}(e) \in L^{\mu}_{\mathcal{P}}(H)$ then H contains a perfect matching by Lemma 3.4.

The idea of the proof of Theorem 1.6 is that we now ask whether it is possible to delete a matching M of size at most $k-2$ from H so that $\mathbf{i}_{\mathcal{P}}(V \setminus V(M)) \in L^{\mu}_{\mathcal{P}}(H)$. If so, we may delete the vertices covered by M from H to form a k-graph H', and the restriction of $\mathcal{P}$ to $V(H')$ is $(2c, 2\mu, \mu'/2)$-robustly maximal with respect to H' by Proposition 3.6. So Lemma 3.4 implies that H' contains a perfect matching, and together with M this gives a perfect matching in H. On the other hand, if H contains a perfect matching M^* then $\mathbf{i}_{\mathcal{P}}(V \setminus V(M^*)) = \mathbf{0}$, so certainly $\mathbf{i}_{\mathcal{P}}(V \setminus V(M^*)) \in L^{\mu}_{\mathcal{P}}(H)$; we will see that there is a submatching $M \subseteq M^*$ of size at most $k-2$ such that $\mathbf{i}_{\mathcal{P}}(V \setminus V(M)) \in L^{\mu}_{\mathcal{P}}(H)$. Putting the two together gives the desired characterisation of whether a k-graph H which satisfies the degree conditions contains a perfect matching.

More precisely, let H be a k-graph whose vertex set V has size n, and which satisfies the degree conditions (1) and (2). Suppose first that H contains a perfect matching M^*. Let $\mathcal{P}$ be a partition of V into $1 \leq d < k$ parts, and $L \subseteq \mathbb{Z}^d$ be a transferral-free edge-lattice such that any matching M in H formed of edges $e \in H$ with $\mathbf{i}_{\mathcal{P}}(e) \notin L$ has size less than C. Certainly we must then have $L^{\mu}_{\mathcal{P}}(H) \subseteq L$, from which we deduce that L is full, and so has index at most $k-1$ in $L^{max}_{\mathcal{P}}$ by Lemma 3.2(ii). Next we require the following lemma, which is a simple application of the pigeonhole principle.

Lemma 3.7. *Let $G = (X, +)$ be an abelian group of order m, and suppose that elements $x_i \in X$ for $i \in [r]$ are such that $\sum_{i \in [r]} x_i = x'$. Then $\sum_{i \in I} x_i = x'$ for some $I \subseteq [r]$ with $|I| \leq m-1$.*

This implies that we can find a matching M' in H of size at most $k-2$ for which $\mathbf{i}_{\mathcal{P}}(M')$ lies in the same coset of L (in $L^{max}_{\mathcal{P}}$) as $\mathbf{i}_{\mathcal{P}}(M^*)$; since $\mathbf{i}_{\mathcal{P}}(V \setminus V(M^*)) = \mathbf{0} \in L$ we deduce that $\mathbf{i}_{\mathcal{P}}(V \setminus V(M')) \in L$, so M' satisfies the requirement of condition (*) of Theorem 1.6. This completes the proof of the simpler direction of Theorem 1.6.

To prove the other direction of Theorem 1.6, namely that condition (*) implies that H contains a perfect matching, we begin by applying Proposition 3.5 to obtain a partition $\mathcal{P}$ of V which is (c, μ, μ')-robustly maximal with respect to H. Since H has high vertex degree, all but a small number of vertices $v \in V$ lie in many edges $e \in H$ with $\mathbf{i}_{\mathcal{P}}(e) \in L^{\mu}_{\mathcal{P}}(H)$; we can ensure that this condition holds for all vertices $v \in V$ by moving these 'bad' vertices to a different part of $\mathcal{P}$. We continue writing $\mathcal{P}$ for the altered partition; Proposition 3.6 implies that $\mathcal{P}$ is still (c, μ, μ')-robustly universal for a slightly larger c and μ and a slightly smaller μ'.

Now let $I = I^{\mu}_{\mathcal{P}}(H)$ and $L = L^{\mu}_{\mathcal{P}}(H)$, form I' by adding to I any index vector $\mathbf{i}$ with respect to $\mathcal{P}$ for which H contains at least $2k^2$ disjoint edges $e \in H$ with $\mathbf{i}_{\mathcal{P}}(e) = \mathbf{i}$, and let L' be the lattice generated by I' (so $L \subseteq L'$). It may well be the case that L' is not transferral-free; but by merging any parts X, Y of $\mathcal{P}$ for which L' contains the vector $\mathbf{u}_X - \mathbf{u}_Y$ we obtain a new partition $\mathcal{P}^*$ of V into fewer than k parts and a new lattice L^* which is transferral-free. Now, since there are at most k^k possible values of $\mathbf{i}_{\mathcal{P}}(e)$ for an edge $e \in H$, it follows from our choice of I' that any matching of edges $e \in H$ such that $\mathbf{i}_{\mathcal{P}^*}(e) \notin L^*$ has size less than $2k^{k+2} \leq C$. So L^* and $\mathcal{P}^*$ satisfy the requirements of condition (*) of Theorem 1.6, from which we deduce that H contains a matching M' of size at most $k-2$ such that $\mathbf{i}_{\mathcal{P}^*}(V \setminus V(M')) \in L^*$. From this it follows that $\mathbf{i}_{\mathcal{P}}(V \setminus V(M')) \in L'$. By the definition of I', when a set of fewer than $2k^2$ vertices is deleted from H there will remain, for every $\mathbf{i}$ added to I to form I', an edge e such that $\mathbf{i}_{\mathcal{P}}(e) = \mathbf{i}$. Using this fact along with Lemma 3.7, we may greedily choose a small matching M'' in $H \setminus V(M')$ such that $\mathbf{i}_{\mathcal{P}}(V \setminus V(M' \cup M'')) \in L = L^{\mu}_{\mathcal{P}}(H)$.

Form H'' by deleting vertices covered by M or M' from H; then by Proposition 3.6 the restriction of $\mathcal{P}$ to $V(H'')$ is $(2c, \mu/c, (\mu')^3)$-robustly maximal with respect to H''. So H'' satisfies the conditions of Lemma 3.4 and so contains a perfect matching; together with M and M'' this gives a perfect matching in H.

3.6 Proof of the key lemma

Rather than prove Lemma 3.4 directly, we first prove the following version of Lemma 3.4 for k-partite k-graphs H. We say that a k-graph is *k-partite* if there is a partition of its vertex set into *vertex classes* $V_1, \dots, V_k$ such that every edge of H intersects each vertex class in a single vertex. Similarly, we say that a set S of $k-1$ vertices from such a graph is *k-partite* if it intersects each vertex class in at most one vertex.

LEMMA 3.8. *Suppose that $1/n \ll \varepsilon, \mu \ll \mu' \ll c, d \ll \gamma, 1/k$ and $\ell \leq k$, and let H be a k-partite k-graph each of whose k vertex classes has size n such that at most εn^{k-1} k-partite sets A of $k-1$ vertices have $d(A) < n/\ell + \gamma n$. Also let $\mathcal{P}$ be a partition of $V(H)$ which refines the partition into vertex classes into parts of size at least cn and which is (c, μ, μ')-robustly maximal with respect to H. Suppose that*

(i) for any vertex $x \in V$ there are at least dn^{k-1} edges $e \in H$ with $x \in e$ and $\mathbf{i}_{\mathcal{P}}(e) \in L^{\mu}_{\mathcal{P}}(H)$, and

(ii) $\mathbf{i}_{\mathcal{P}}(V) \in L^{\mu}_{\mathcal{P}}(H)$.

Then H contains a perfect matching.

To deduce Lemma 3.4 for a (non-partite) k-graph H, we take a random k-partition of $V(H)$ into vertex classes of size n/k. A technical argument using weak hypergraph regularity then shows that the conditions of Lemma 3.8 (with $\ell = k$) are satisfied with positive probability, and we apply Lemma 3.8 to deduce that H contains a perfect matching.

It remains to prove the k-partite form, Lemma 3.8. One key idea for this proof is the next proposition. For any k-graph H on n vertices, any partition $\mathcal{P}$ of $V(H)$ and any index vector $\mathbf{i} \in I^{\mu}_{\mathcal{P}}(H)$, there are at least μn^k edges $e \in H$ with $\mathbf{i}_{\mathcal{P}}(e) = \mathbf{i}$. However, for the k-graphs H and partitions $\mathcal{P}$ which we consider we can say much more. For each $\mathbf{i} \in I^{\mu}_{\mathcal{P}}(H)$, we let $H_{\mathbf{i}}$ be the k-partite subgraph of H on the vertex set $\bigcup_{W \in \mathcal{P}, i_W = 1} W$, whose edges are the edges in H of index $\mathbf{i}$.

PROPOSITION 3.9. *Suppose that $1/n \ll \mu, \varepsilon \ll \varepsilon' \ll \delta, c, 1/k$. Let H be a k-partite k-graph whose vertex classes $V_1, \dots, V_k$ each have size $cn \leq |V_i| \leq n$, and in which at most εn^{k-1} k-partite sets S of $k-1$ vertices have $d(S) < \delta n$. Also let $\mathcal{P}$ be a partition of $V(H)$ into parts of size at least cn which refines the partition into vertex classes and has the property that $L^{\mu}_{\mathcal{P}}(H)$ is transferral-free.*

(i) For any $\mathbf{i} \in I^{\mu}_{\mathcal{P}}(H)$ there are at least $\delta c^{k-1} n^k/2$ edges $e \in H$ with index $\mathbf{i}_{\mathcal{P}}(e) = \mathbf{i}$.

(ii) For any $\mathbf{i} \in I^{\mu}_{\mathcal{P}}(H)$, at most $\varepsilon' n^{k-1}$ k-partite sets S of $k-1$ vertices of $H_{\mathbf{i}}$ have $d_{H_{\mathbf{i}}}(S) < (\delta - \varepsilon')n$.

(iii) Each part of $\mathcal{P}$ has size at least $(\delta - \varepsilon')n$.

PROOF. For (ii), let $W_1 \subseteq V_1, \dots, W_k \subseteq V_k$ be the parts of $\mathcal{P}$ such that $i_W = 1$ (so $V(H_{\mathbf{i}}) = \bigcup_{j \in [k]} W_j$). Consider the k-partite sets $S = \{x_1, \dots, x_{k-1}\}$ with $x_j \in W_j$ for each j. By assumption, at most εn^{k-1} such sets S have $d_H(S) < \delta n$. Furthermore, if S satisfies $d_H(S) \geq \delta n$ but $d_{H_{\mathbf{i}}}(S) < (\delta - \varepsilon')n$, then S is contained in at least $\varepsilon' n$ edges $e \in H$ with $\mathbf{i}_{\mathcal{P}}(e) \notin I^{\mu}_{\mathcal{P}}(H)$ (since $L^{\mu}_{\mathcal{P}}(H)$ is transferral-free). Since there are at most $1/c^k$ possible index vectors $i_{\mathcal{P}}(e)$ for an edge $e \in H$, there are at most $\mu(kn)^k/c^k$ such edges of H. We therefore conclude that at most $\mu(kn)^k/c^k\varepsilon' n \leq \varepsilon' n^{k-1}$ sets S have this form, completing the proof of (ii). We immediately deduce (iii). Since there are at least $(cn)^{k-1}$ choices for S, (iii) implies in particular that $H_{\mathbf{i}}$ contains at least $((cn)^{k-1} - \varepsilon' n^{k-1})(\delta - \varepsilon')n \geq c^{k-1}\delta n^k/2$ edges, and so we also have (i). $\square$

What this proposition tells us is that if H and $\mathcal{P}$ satisfy the conditions of Lemma 3.8, then for any $\mathbf{i} \in I^{\mu}_{\mathcal{P}}(H)$ the subgraph $H_{\mathbf{i}}$ satisfies a similar degree condition to H. In fact, if $\mathcal{P}$ is non-trivial then the degree condition on $H_{\mathbf{i}}$ is in a sense stronger than that on H. Indeed, in this case Proposition 3.9(iii) applied with $\delta = 1/\ell + \gamma$ implies that each part of $V(H_{\mathbf{i}})$ has size at most $(1 - 1/\ell)n$. So when Proposition 3.9(ii) states that all but at most $\varepsilon' n^{k-1}$ k-partite sets S of $k-1$ vertices have $d(S) \geq (1/\ell + \gamma - \varepsilon')n$, this is out of a maximum of $(1 - 1/\ell)n$ vertices. Hence the proportion of possible neighbours of S which are in fact neighbours of S is at least $\frac{1/\ell + \gamma/2}{1 - 1/\ell} \geq \frac{1}{\ell - 1} + \frac{\gamma}{2}$. So expressed as a proportion, the degree of most sets of $k-1$ vertices in $H_{\mathbf{i}}$ is significantly greater than in H.

This suggests the method of proof of Lemma 3.8: we proceed by induction on ℓ. For the base case $\ell = 2$ Proposition 3.9(iii) implies that the partition $\mathcal{P}$ is simply the trivial partition of $V(H)$ into its vertex classes, whereupon (a k-partite version of) Theorem 3.1 implies that H contains a perfect matching. The main part of the proof is therefore the inductive step, for which we may assume that $\mathcal{P}$ is non-trivial by the same argument. Suppose then that H and $\mathcal{P}$ are as in Lemma 3.8 for some value of ℓ, and consider the subgraphs $H_{\mathbf{i}}$ for $\mathbf{i} \in I := I^{\mu}_{\mathcal{P}}(H)$. For each $\mathbf{i} \in I$ apply Proposition 3.5 to choose a partition $\mathcal{Q}_{\mathbf{i}}$ of the vertex set $V(H_{\mathbf{i}})$ of $H_{\mathbf{i}}$ which is (c, μ_1, μ_2)-robustly maximal with respect to $H_{\mathbf{i}}$, for some small $\mu_1 \ll \mu_2$. Our strategy is to randomly choose a subset $T_{\mathbf{i}}$ of $V(H_{\mathbf{i}})$ for each $\mathbf{i} \in I$, and then to use the inductive hypothesis to find perfect matchings in each of the induced k-graphs $H'_{\mathbf{i}} := H_{\mathbf{i}}[T_{\mathbf{i}}]$. To do this we stipulate that the sets $T_{\mathbf{i}}$ must partition V and that when each $T_{\mathbf{i}}$ is partitioned according to the k-partition of V, the parts of $T_{\mathbf{i}}$ have equal size $t_{\mathbf{i}} \geq \varepsilon n$. In addition, we need to show that for each $\mathbf{i} \in I$:

(i) At most $\varepsilon' n^{k-1}$ k-partite sets A of $k-1$ vertices of $T_{\mathbf{i}}$ have $d_{H'_{\mathbf{i}}}(A) < (1/(\ell - 1) + \gamma/3)t_{\mathbf{i}}$ (for a constant $\varepsilon' \gg \varepsilon$).

(ii) The restriction of $\mathcal{Q}_{\mathbf{i}}$ to $T_{\mathbf{i}}$ is $(2c, \mu_1/c, \mu_2^3)$-robustly maximal.

(iii) For any vertex $x \in T_{\mathbf{i}}$ there are at least $dt_{\mathbf{i}}^{k-1}$ edges $e \in H'_{\mathbf{i}}$ with $x \in e$ and $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}}}(e) \in L^{\mu_1/c}_{\mathcal{Q}_{\mathbf{i}}}(H'_{\mathbf{i}})$.

(iv) $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}}}(T_{\mathbf{i}}) \in L^{\mu_1/c}_{\mathcal{Q}_{\mathbf{i}}}(H'_{\mathbf{i}})$.

Condition (i) is a consequence of our observation above that for most k-partite sets of $k-1$ vertices of $H_{\mathbf{i}}$, the proportion of possible neighbours which are actually neighbours is at least $1/(\ell - 1) + \gamma/2$. With high probability this property is inherited by $H'_{\mathbf{i}}$, so we obtain (i). Also, since $\mathcal{Q}_{\mathbf{i}}$ was chosen to be (c, μ_1, μ_2)-robustly maximal with respect to $H_{\mathbf{i}}$, (ii) holds with high probability by Proposition 3.6(ii).

For (iii) we notice that there may actually be a number of vertices of $H_{\mathbf{i}}$ which lie in fewer than dn^{k-1} edges $e \in H_{\mathbf{i}}$ with $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}}}(e) \in L^{\mu_1}_{\mathcal{Q}_{\mathbf{i}}}(H_{\mathbf{i}})$. However, a short deduction from Proposition 3.9(ii) shows that the number of such 'bad' vertices is small. Since our assumption on H was that every vertex $v \in V$ lies in many edges $e \in H$ with $\mathbf{i}_{\mathcal{P}}(e) \in I$ we may delete (before choosing the sets $T_{\mathbf{i}}$) a small matching to remove all of the 'bad' vertices (however, for simplicity we continue to write V for the new vertex set). We may then assume that every vertex of $H_{\mathbf{i}}$ lies in at least dn^{k-1} edges $e \in H_{\mathbf{i}}$ with $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}}}(e) \in L^{\mu_1}_{\mathcal{Q}_{\mathbf{i}}}(H_{\mathbf{i}})$. With high probability the random selection will ensure that $L^{\mu_1}_{\mathcal{Q}_{\mathbf{i}}}(H_{\mathbf{i}}) \subseteq L^{\mu_1/c}_{\mathcal{Q}_{\mathbf{i}}}(H'_{\mathbf{i}})$, and so (iii) also holds with high probability.

Ensuring that (iv) is satisfied is therefore the principal difficulty in the proof of Lemma 3.8. As noted above we will have $L^{\mu_1}_{\mathcal{Q}_{\mathbf{i}}}(H_{\mathbf{i}}) \subseteq L^{\mu_1/c}_{\mathcal{Q}_{\mathbf{i}}}(H'_{\mathbf{i}})$, so it is enough to ensure that $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}}}(T_{\mathbf{i}}) \in L^{\mu_1}_{\mathcal{Q}_{\mathbf{i}}}(H_{\mathbf{i}})$. However, we must use other arguments to control $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}}}(T_{\mathbf{i}})$; in the remainder of the section we give a sketch of these arguments. We begin by observing that (iv) depends only on the values of $|T_{\mathbf{i}} \cap Y|$ for parts $Y \in \mathcal{Q}_{\mathbf{i}}$. Hence if two vertices x and y are contained in the same part of $\mathcal{Q}_{\mathbf{i}}$ for *every* $\mathcal{Q}_{\mathbf{i}}$ such that $x, y \in V(H_{\mathbf{i}})$, then they are effectively interchangeable for our purposes. With this in mind, we define a partition $\mathcal{Q}^\cap$ of V where vertices $x, y \in V$ lie in the same part of $\mathcal{Q}^\cap$ if and only if they lie in the same part of every partition $\mathcal{Q}_{\mathbf{i}}$, i.e., $\mathcal{Q}^\cap$ is the 'meet' of the partitions $\mathcal{Q}_{\mathbf{i}}$. Our strategy will be to fix the number of vertices $n(\mathbf{i}, Z)$ which the set $T_{\mathbf{i}}$ will take from each part Z of $\mathcal{Q}^\cap$ so that (iv) is satisfied, and then choose the sets $T_{\mathbf{i}}$ uniformly at random to satisfy these constraints.

Recall that in our example in Section 3.3, problems arose in relation to a divisibility barrier partition $\mathcal{Q}$ which was 'hidden' inside $\mathcal{P}$. In order to discover and deal with any such partitions, we will need to consider the 'join' $\mathcal{Q}^\cup$ of the partitions $\mathcal{Q}_{\mathbf{i}}$. We define an auxiliary graph G whose vertices are the parts of $\mathcal{Q}^\cap$, where $Z, Z' \in \mathcal{Q}^\cap$ are adjacent in G if they are contained in the same part of some $\mathcal{Q}_{\mathbf{i}}$. The parts of $\mathcal{Q}^\cup$ are then formed by taking the union of the parts of $\mathcal{Q}^\cap$ in each component of G.

Choosing the numbers $n(\mathbf{i}, Z)$ so that (iv) is satisfied progresses through three stages: we first choose rough targets for the number of vertices to be contained in each $T_{\mathbf{i}}$; then we choose how many vertices each $T_{\mathbf{i}}$ will take from each part of $\mathcal{Q}^\cup$, before finally refining this choice to obtain the numbers $n(\mathbf{i}, Z)$ as required. A *fractional perfect matching* in a k-graph H is a weighting p_e of its edges such that $\sum_{x \in e} p_e = 1$ for each $x \in V(H)$. The first stage relies on the following theorem, which is proved similarly to [13, Corollary 3.1]:

THEOREM 3.10. *Let H be a k-partite k-graph with vertex classes of size n such that every partite set of $k-1$ vertices has degree at least n/k. Then H contains a fractional perfect matching.*

To obtain our targets for the size of each $T_{\mathbf{i}}$, we apply Theorem 3.10 to find a fractional perfect matching p_e in an auxiliary k-graph on V whose edges are all k-tuples e with $\mathbf{i}_{\mathcal{P}}(e) \in I$. Now set $\lambda_{\mathbf{i}} = \sum_{\mathbf{i}_{\mathcal{P}}(e)=\mathbf{i}} p_e$ for each $\mathbf{i} \in I$; then $\sum_{\mathbf{i} \in I} \lambda_{\mathbf{i}} \mathbf{i} = \mathbf{i}_{\mathcal{P}}(V)$ by the definition of p_e. Next we adjust the $\lambda_{\mathbf{i}}$ slightly to obtain integers $\rho_{\mathbf{i}}$ with $\sum_{\mathbf{i} \in I} \rho_{\mathbf{i}} \mathbf{i} = \mathbf{i}_{\mathcal{P}}(V)$. So we will aim for each $T_{\mathbf{i}}$ to take approximately $\rho_{\mathbf{i}}$ vertices from each part of V (of size n). Actually, by using the error term in the degree condition of Lemma 3.8 we can ensure further that $\rho_{\mathbf{i}} \geq \varepsilon' n$ for each $\mathbf{i} \in I$.

In the second stage we choose provisional values for the quantities $|T_{\mathbf{i}} \cap X|$ for $X \in \mathcal{Q}^\cup$ in proportion to the integers $\rho_{\mathbf{i}}$. So if $X \subseteq W$ for some $W \in \mathcal{P}$ with $i_W = 1$ then our target for $|T_{\mathbf{i}} \cap X|$ will be $\rho_{\mathbf{i}}|X|/|W|$ (and if $X \subseteq W$ with $i_W = 0$ then our target will be 0). We then round these values to nearby integers and derive provisional values for $\mathbf{i}_{\mathcal{Q}^\cup}(T_{\mathbf{i}})$ from these integers in the natural way; that is, for each $X \in \mathcal{Q}^\cup$ our provisional vector will have our provisional value for $|T_{\mathbf{i}} \cap X|$ as its X-coordinate. For each $\mathbf{i} \in I$ we further adjust the provisional value for $\mathbf{i}_{\mathcal{Q}^\cup}(T_{\mathbf{i}})$ so that it lies in $L^{\mu}_{\mathcal{Q}^\cup}(H_{\mathbf{i}})$. Now the provisional values will no longer sum to $\mathbf{i}_{\mathcal{Q}^\cup}(V)$; instead they will sum to some $\mathbf{i}' \in L^{\mu}_{\mathcal{Q}^\cup}(H)$. Crucially, the robust maximality of $\mathcal{P}$ now implies that $\mathcal{Q}^\cup$ is not transferral-free, and it is not hard to show that further $\mathbf{i}_{\mathcal{Q}^\cup}(V)$ must lie in $L^{\mu}_{\mathcal{Q}^\cup}(H)$. It follows that $\mathbf{d} = \mathbf{i}_{\mathcal{Q}^\cup}(V) - \mathbf{i}'$ must also lie in $L^{\mu}_{\mathcal{Q}^\cup}(H)$. Using this fact, we can write $\mathbf{d} = \sum_{\mathbf{i} \in I} \mathbf{d}_{\mathbf{i}}$ in such a way that $\mathbf{d}_{\mathbf{i}} \in L^{\mu}_{\mathcal{Q}^\cup}(H_{\mathbf{i}})$ for each $\mathbf{i} \in I$. Further $\mathbf{d}$ is small (in absolute value), so we can ensure that each $\mathbf{d}_{\mathbf{i}}$ is small. Finally, we add to each provisional $\mathbf{i}_{\mathcal{Q}^\cup}(T_{\mathbf{i}})$ the vector $\mathbf{d}_{\mathbf{i}}$ and note that the property $\mathbf{i}_{\mathcal{Q}^\cup}(T_{\mathbf{i}}) \in L^{\mu}_{\mathcal{Q}^\cup}(H)$ is preserved. This yields our final values for $\mathbf{i}_{\mathcal{Q}^\cup}(T_{\mathbf{i}})$. Thus we fix the quantity $|T_{\mathbf{i}} \cap X|$ for each $X \in \mathcal{Q}^\cup$.

In the final stage we use Baranyai's Matrix Rounding Theorem, which is stated below.

THEOREM 3.11. *[19, Theorem 7.5] Let A be a real matrix. Then there exists an integer matrix B whose entries, row sums, column sums and the sum of all the entries are the entries, row sums, column sums and the sum of all the entries respectively of A, rounded either up or down to the nearest integer.*

For this last stage, we deal with each part X of $\mathcal{Q}^\cup$ in turn. Similarly to the previous step, we take the real numbers $|Z||T_{\mathbf{i}} \cap X|/|X|$ as provisional values for $|T_{\mathbf{i}} \cap Z| = n(\mathbf{i}, Z)$ for each part $Z \in \mathcal{Q}^\cap$. Let A be the matrix indexed by the parts $Z \subseteq X$ of $\mathcal{Q}^\cap$ whose entries are the provisional values of $n(\mathbf{i}, Z)$. We apply Theorem 3.11 to A to obtain a new matrix B and take new, integer provisional values from B. The fact that these values remain feasible is implied by Theorem 3.11, as the row and column sums of A are the part sizes $|Z|$ and the values $|T_{\mathbf{i}} \cap X|$; since these are already integers, the row and column sums remain unchanged. From these integers we can derive provisional values for $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}}}(T_{\mathbf{i}})$ for each $\mathbf{i}$ similarly as before; our goal now is to adjust these provisional values so that in addition (iv) holds.

Fix $\mathbf{i}_1, \mathbf{i}_2 \in I$, and suppose that Z and Z' are parts of $\mathcal{Q}^\cap$ which are subsets of distinct parts Y_1 and Y'_1 respectively of $\mathcal{Q}_{\mathbf{i}_1}$, but that Z and Z' are subsets of the same part Y_2 of $\mathcal{Q}_{\mathbf{i}_2}$. Define an $(\mathbf{i}_1, \mathbf{i}_2, Z, Z')$*-swap* as the operation of increasing the provisional values of $n(\mathbf{i}_1, Z)$ and $n(\mathbf{i}_2, Z')$ each by one, and decreasing those for $n(\mathbf{i}_2, Z)$ and $n(\mathbf{i}_1, Z')$ by one. An $(\mathbf{i}_1, \mathbf{i}_2, Z, Z')$-swap has no effect on the provisional values of $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}_2}}(T_{\mathbf{i}_2})$; however, it adds $\mathbf{u}_{Y_1} - \mathbf{u}_{Y'_1}$ to the provisional value of $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}_1}}(T_{\mathbf{i}_1})$.

We now make use of the auxiliary graph G, which was used to define $\mathcal{Q}^\cup$. Apply (a k-partite version of) Lemma 3.2(i) to $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}}}(T_{\mathbf{i}})$ to obtain parts Y and Y' of $\mathcal{Q}_{\mathbf{i}}$ such that $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}}}(T_{\mathbf{i}}) + \mathbf{u}_Y - \mathbf{u}_{Y'} \in L^{\mu}_{\mathcal{Q}_{\mathbf{i}}}(H_{\mathbf{i}})$. Then Y and Y' are both subsets of some part $X \in \mathcal{Q}^\cup$. We will show that we may make a sequence of modifications to our targets for $n(\mathbf{i}, Z)$ whose effect is to add $\mathbf{u}_Y - \mathbf{u}_{Y'}$ to $\mathbf{i}_{\mathcal{Q}_\mathbf{i}}(T_\mathbf{i})$, while leaving $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}'}}(T_{\mathbf{i}'})$ unchanged for any other $\mathbf{i}' \in I$. Indeed, choose parts Z and Z' of $\mathcal{Q}^\cap$ with $Z \subseteq Y$ and $Z' \subseteq Y'$, and choose a path $Z = Z_0, \dots, Z_s = Z'$ in G. We now perform a sequence of $(\mathbf{i}, \mathbf{i}_j, Z_{j-1}, Z_j)$-swaps, where each $\mathbf{i}_j$ is chosen so that Z_{j-1} and Z_j lie in the same part of $\mathcal{Q}_{\mathbf{i}_j}$. The net effect of these swaps is to add $\mathbf{u}_Y - \mathbf{u}_{Y'}$ to the provisional value of $\mathbf{i}_{\mathcal{Q}_\mathbf{i}}(T_\mathbf{i})$, while leaving $\mathbf{i}_{\mathcal{Q}_{\mathbf{i}'}}(T_{\mathbf{i}'})$ unaltered for every other $\mathbf{i}' \in I$. Applying this process for each $\mathbf{i} \in I$ in turn, we obtain (iv).

Together, properties (i)-(iv) above show that each of the k-graphs $H'_\mathbf{i}$ satisfies the conditions of Lemma 3.8 with $t_\mathbf{i}$ in place of n and other weaker constants, but crucially with $\ell-1$ in place of ℓ. So by our inductive hypothesis we deduce that each $H'_\mathbf{i}$ contains a perfect matching; since the vertex sets $T_\mathbf{i}$ partition V we conclude that H itself contains a perfect matching, completing the proof of Lemma 3.8.

4. CONCLUSIONS

Our main result shows that the decision problem $\mathbf{PM}(k,\delta)$ is in P for $\delta > 1/k$. Moreover, our algorithm provides either a perfect matching or a certificate that none exists. The correctness of our algorithm relies on a theoretical result of independent interest, giving a characterisation of perfect matchings in terms of approximate divisibility barriers. In the case $k = 3$ this characterisation takes a particularly nice form, which may be viewed as a strong stability version of a theorem of Rödl, Ruciński and Szemerédi [14].

The complexity status of $\mathbf{PM}(k,\delta)$ when $\delta = 1/k$ remains open, that is, deciding whether there is a perfect matching in a k-graph H with $\delta_{k-1}(H) \geq n/k$. We expect this will be difficult to resolve, as this is the minimum degree threshold at which a perfect fractional matching is guaranteed, so there is a clear behavioural change at this point.

Finally, by similar methods we were also able to prove similar results pertaining to k-partite k-graphs. The differences between the proofs of the standard and k-partite versions of these theorems are minor, so we state these results here without further comment.

Theorem 4.1. *For any k, γ and C_0, there exist $n_0 = n_0(k,\gamma,C_0)$ and $\varepsilon = \varepsilon(k,\gamma,C_0)$ such that for any C with $2k^{k+2} \leq C \leq C_0$ we have the following characterisation. Let H be a k-partite k-graph whose k vertex classes each have size $n \geq n_0$, such that $\delta_1(H) \geq \gamma n^{k-1}$ and at most εn^{k-1} k-partite sets S of $k-1$ vertices of H have $d(S) < (1/k+\gamma)n$. Then H has a perfect matching if and only if the following condition holds:*

If $\mathcal{P}$ is a partition of $V(H)$ which refines the partition into vertex classes and partitions each vertex class into d parts, where $1 \leq d < k$, and $L \subseteq \mathbb{Z}^{kd}$ is a transferral-free edge-lattice such that any matching M in H formed of edges $e \in H$ with $\mathbf{i}_\mathcal{P}(e) \notin L$ has size less than C, then there exists a matching M' in H of size at most $k-2$ such that $\mathbf{i}_\mathcal{P}(V(H) \setminus V(M')) \in L$.

Theorem 4.2. *Fix $k \geq 3$ and $\gamma > 0$. Then there is a polynomial-time algorithm, which given any k-partite k-graph H whose k vertex classes each have size n and in which every k-partite set S of $k-1$ vertices has $d(S) \geq (1/k+\gamma)n$, finds either a perfect matching in H or a certificate that no perfect matching exists.*

5. ACKNOWLEDGEMENTS

Research supported in part by ERC grant 239696 and EPSRC grant EP/G056730/1.

6. REFERENCES

[1] R. Aharoni, A. Georgakopoulos and P. Sprüssel, Perfect matchings in r-partite r-graphs, *European Journal of Combinatorics* **30** (2009), 39–42.

[2] N. Alon, P. Frankl, H. Huang, V. Rödl, A. Ruciński and B. Sudakov, Large matchings in uniform hypergraphs and the conjectures of Erdős and Samuels, *J. Combin. Theory A* **119** (2012), 1200–1215.

[3] A. Asadpour, U. Feige and A. Saberi, Santa Claus meets hypergraph matchings, *Proc. APPROX-RANDOM* (2008), 10–20.

[4] G. A. Dirac, Some theorems on abstract graphs, *Proc. London Math. Soc.* **2** (1952), 69–81.

[5] J. Edmonds, Paths, trees, and Flowers, *Canad. J. Math.* **17** (1965), 449–467.

[6] R. M. Karp, Reducibility among combinatorial problems, *Complexity of Computer Computations* (1972), 85–103.

[7] M. Karpiński, A. Ruciński and E. Szymańska, The complexity of perfect matching problems on dense hypergraphs, *Proc. ISAAC '09* (2009), 626–636.

[8] P. Keevash and R. Mycroft, A geometric theory for hypergraph matchings, arXiv:1108.1757.

[9] I. Khan, Perfect matching in 3-uniform hypergraphs with large vertex degree, arXiv:1101.5830.

[10] I. Khan, Perfect Matchings in 4-uniform hypergraphs, arXiv:1101.5675.

[11] A. Lo and K. Markström, Perfect matchings in 3-partite 3-uniform hypergraphs, arXiv:1103.5654.

[12] V. Rödl and A. Ruciński, Dirac-type questions for hypergraphs — a survey (or more problems for Endre to solve), *An Irregular Mind (Szemerédi is 70)* **21** (2010), 1–30.

[13] V. Rödl, A. Ruciński and E. Szemerédi, Perfect matchings in large uniform hypergraphs with large minimum degree, *Europ. J. Combin.* **27** (2006) 1333–1349.

[14] V. Rödl, A. Ruciński and E. Szemerédi, Perfect matchings in large uniform hypergraphs with large minimum collective degree, *J. Combin. Theory Ser. A* **113** (2009), 613–636.

[15] E. Szemerédi, Regular partitions of graphs, *Problèmes combinatoires et théorie des graphes* (1978), 399–401.

[16] E. Szymańska, The complexity of almost perfect matchings in uniform hypergraphs with high codegree, *Lecture Notes in Computer Science* **5874** (2009), 438–449.

[17] A. Treglown and Y. Zhao, Exact minimum degree thresholds for perfect matchings in uniform hypergraphs, *J. Combin. Theory A* **119** (2012), 1500–1522.

[18] W. T. Tutte, The factorisation of linear graphs, *J. London Math. Soc.* **22** (1947), 107–111.

[19] J.H. van Lint and R.M. Wilson, A course in combinatorics, Cambridge University Press, 2001.