Hypergraph Decompositions

Jackson J. I. Gatenby

Supervisor:
Dr. Barbara Maenhaut

A thesis submitted this eleventh day of November 2014, in partial fulfilment of the requirements for the degree of Bachelor of Science (Honours) at The University of Queensland.

Abstract

The central question in combinatorial design theory is that of the existence of designs. We use the existence question of t-wise balanced designs to motivate the study of decompositions of complete t-uniform hypergraphs into edge-disjoint copies of a given t-uniform hypergraph. Building off work in [2, 6, 8, 10], we solve the existence question of such decompositions for all 3-uniform hypergraphs with at most three edges, and all regular 3-uniform hypergraphs with four edges.

In approaching this question, we describe a general method of constructing decompositions of complete 3-uniform hypergraphs, and present algorithms to generate solutions to small cases that are used by the construction.

We also discuss known results and conjectures in decomposing hypergraphs into Hamiltonian cycles.

Contents

Chapter 1

Introduction

1.1 Definitions

A *graph* G is a pair (V, E), where $V = V(H)$ is a finite set of *vertices*, and $E = E(H)$ is a set of 2-element subsets of V, called *edges*. A *subgraph* of G is a graph H where $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$.

Two graphs G and H are *isomorphic* if there is a bijection $\varphi : V(G) \to V(H)$ such that for every edge $\{u, v\} \in E(G)$, $\{\varphi(u), \varphi(v)\} \in E(H)$, and for every edge $\{x, y\} \in E(H)$, $\{\varphi^{-1}(x), \varphi^{-1}(y)\} \in E(G)$. We write $G \simeq H$ to indicate that G is isomorphic to H. The map φ is called an *isomorphism* from G to H.

A *multigraph* G is a pair (V, E) where $V = V(G)$ is a finite set of vertices, and $E = E(G)$ is a finite multiset of 2-element subsets of V, called edges. That is, we allow for edges to occur multiple times in the edge set, but only finitely many copies of each edge.

A *decomposition* of a graph or multigraph G is a set $\mathcal{D} = \{G_1, G_2, \ldots, G_d\}$ of subgraphs of G whose edge sets partition $E(H)$. If Γ is a family of graphs such that each $G_i \in \mathcal{D}$ is isomorphic to some member of Γ, then $\mathcal{D}$ is said to be a Γ-decomposition of G. If Γ contains one element H, then $\mathcal{D}$ is said to be an *H-decomposition* of G.

The reader is referred to [7] for a discussion on known results in graph decompositions.

If we observe that each edge of a graph is merely a 2-element subset of the vertex set, it leads to a natural generalisation. A *hypergraph* H is a pair (V, E), where $V = V(H)$ is a finite set of *vertices*, and $E = E(H)$ is a set of subsets of V, called *hyperedges* or *edges*. The *order* of H is $|V|$, and the *size* of H is $|E|$. H is said to be *t-uniform* if each edge has size t.

Let $S \subseteq V(H)$ be a set of vertices in H. The *degree* of S, denoted

$\deg_H(S)$, is $|\{e \in E(H) : S \subseteq e\}|$, that is, the number of edges which contain S. For any vertex $x \in V(H)$, the degree of x is $\deg_H(x) = \deg_H(\{x\})$. If every vertex $x \in V(H)$ has degree d, then H is said to be *d-regular*, or *regular of degree d*.

For example, a 2-uniform hypergraph is a graph, and in this case the notions of the degree of a vertex x coincide. Note that $\deg_H(\emptyset) = |E(H)|$.

A *subhypergraph* of H is a hypergraph K where $V(K) \subseteq V(H)$ and $E(K) \subseteq E(H)$. The subhypergraph K is *spanning* if $V(K) = V(H)$.

Two hypergraphs H and K are *isomorphic* if there is a bijection $\varphi : V(H) \to V(K)$ such that for every edge $e \in E(H)$, $\varphi(e) = \{\varphi(x) : x \in e\} \in E(K)$, and for every $f \in E(K)$, $\varphi^{-1}(f) \in E(H)$. We write $H \simeq K$ to indicate that H is isomorphic to K. The map φ is called an *isomorphism* from H to K.

The *union* of hypergraphs H and K, denoted $H \cup K$, is the hypergraph with $V(H \cup K) = V(H) \cup V(K)$ and $E(H \cup K) = E(H) \cup E(K)$.

If K is a subhypergraph of H, then their *difference* is the hypergraph $H \setminus K$ with $V(H \setminus K) = V(H)$ and $E(H \setminus K) = E(H) \setminus E(K)$.

A *multihypergraph* H is a pair (V, E) where $V = V(H)$ is a finite set of vertices, and $E = E(H)$ is a finite multiset of subsets of V, called hyperedges or edges.

A *decomposition* of a hypergraph or multihypergraph K is a set $\mathcal{D} = \{K_1, K_2, \ldots, K_d\}$ of subhypergraphs of K, called *blocks*, whose edge sets partition $E(K)$. If each block $K_i \in \mathcal{D}$ is isomorphic to some member of a family Γ of hypergraphs, then the decomposition is a *Γ-decomposition* of K. If Γ contains one member H, then the decomposition is an *H-decomposition* of K.

1.1.1 Examples of Hypergraphs

In this section, we will describe some useful families of hypergraphs.

For a finite non-empty set V, the *complete t-uniform hypergraph on V* is the hypergraph $K_V^{(t)} = (V, E)$ with $E = \{e \in \mathcal{P}(V) : |e| = t\}$. For a positive integer v, the notation $K_v^{(t)}$ refers to any complete t-uniform hypergraph of order v. For example, we identify $K_v^{(2)}$ with the complete graph K_v.

For a finite non-empty set V and a natural number λ, the *complete t-uniform multihypergraph*, denoted $\lambda K_V^{(t)}$, is the multihypergraph with vertex set V consisting of λ copies of every t-element subset of V. For positive $v \in \mathbb{Z}$, the notation $\lambda K_v^{(t)}$ refers to any complete t-uniform multihypergraph $\lambda K_V^{(t)}$ where $|V| = v$.

Given a collection of m finite, non-empty, pairwise disjoint sets $V_1, \ldots, V_m$, let $V = \bigcup_{i=1}^m V_i$. The *complete t-uniform m-partite hypergraph* on V, denoted

$K^{(t)}_{V_1,V_2,\ldots,V_m}$, is the t-uniform hypergraph (V,E) having all edges which contain at most one element from each of $V_1,V_2,\ldots,V_m$. Formally, $e \in E$ iff $|e| = t$ and for all $i = 1,\ldots,m$, $|e \cap V_i| \le 1$. In such a hypergraph, the sets $V_1,V_2,\ldots,V_m$ are called the *partite sets* of the hypergraph.

For positive integers $v_1,v_2,\ldots,v_m$, the notation $K^{(t)}_{v_1,v_2,\ldots,v_m}$ is used to denote any complete t-uniform m-partite hypergraph where the partite sets have order $v_1,v_2,\ldots,v_m$ respectively.

For example, we identify $K^{(2)}_{v_1,v_2}$ with the complete bipartite graph K_{v_1,v_2}.

Given a collection of $m+1$ finite pairwise disjoint sets $V_1,V_2,\ldots,V_m,W$, where $V_i \ne \emptyset$ for each $i = 1,2,\ldots,m$, we define

$$L^{(t)}_{V_1,V_2,\ldots,V_m,[W]} = K^{(t)}_{V_1\cup V_2\cup\cdots\cup V_m\cup W} \setminus (K^{(t)}_{V_1\cup W} \cup K^{(t)}_{V_2\cup W} \cup \cdots \cup K^{(t)}_{V_m\cup W}).$$

That is, $L^{(t)}_{V_1,V_2,\ldots,V_m,[W]}$ is the t-uniform hypergraph with vertex set $V = V_1 \cup V_2 \cup \cdots \cup V_m \cup W$, and where $e \subseteq V$ is an edge if and only if $|e| = t$ and e has non-empty intersection with at least two of $V_1,\ldots,V_m$.

For integers $v_1,v_2,\ldots,v_m \ge 1$ and $w \ge 0$, the notation $L^{(t)}_{v_1,v_2,\ldots,v_m,[w]}$ is used to denote any hypergraph $L^{(t)}_{V_1,V_2,\ldots,V_m,[W]}$ where $|V_i| = v_i$ for each $i = 1,2,\ldots,m$, and $|W| = w$.

If $W = \emptyset$, we may use the notation $L^{(t)}_{V_1,V_2,\ldots,V_m} = L^{(t)}_{V_1,V_2,\ldots,V_m,[W]}$, and if $w = 0$, we may use the notation $L^{(t)}_{v_1,v_2,\ldots,v_m} = L^{(t)}_{v_1,v_2,\ldots,v_m,[0]}$.

Given a t-uniform hypergraph H, a *d-factor* of H is a spanning d-regular subhypergraph K. For example, a 1-factor consists of $\frac{|V(H)|}{t}$ pairwise disjoint edges from H. An obvious necessary condition then for the existence of a 1-factor of H is that t divides $|V(H)|$.

1.2 Hamiltonian decompositions

Various attempts have been made to generalise the well-known result of Walecki from the 1890's that K_v admits a decomposition into Hamiltonian cycles if and only if v is odd, and a decomposition into $\frac{v-1}{2}$ Hamiltonian cycles and one 1-factor if and only if v is even; we shall discuss them here.

First, it must be made clear what is meant by a Hamiltonian cycle of a hypergraph, and particularly, of $K^{(t)}_v$. The following definition, given by Berge in [3], extends naturally from the definition of a Hamiltonian cycle on a graph.

Definition 1.2.1. Let H be a hypergraph containing v vertices, with vertex set $V(H) = \{x_0,x_1,\ldots,x_{v-1}\}$. A *Berge type Hamiltonian cycle* in H is a sequence of alternating vertices and edges $(x_0,e_0,x_1,e_1,\ldots,x_{v-1},e_{v-1},x_0)$

where $x_i \neq x_j$ if $i \neq j$, and such that $e_i \in E(H)$ contains x_i and x_{i+1} for each $i = 0, 1, \ldots, v-1$ (modulo v). □

A decomposition of a hypergraph H into Berge type Hamiltonian cycles is called a *Berge type Hamiltonian decomposition.*

In 2-uniform graphs, one can drop the edges from the sequence and consider only the sequence of vertices, note that this cannot be done here.

The existence problem for Berge type Hamiltonian decompositions of $K_v^{(3)}$ was solved in [19], completing work in [5]:

Theorem 1.2.2 ([19])**.** *A Berge type Hamiltonian decomposition of $K_v^{(3)}$ exists iff $v \equiv 1$ or 2 (mod 3); and if $v \equiv 0$ (mod 3), there exists a Berge type Hamiltonian decomposition of $K_v^{(3)} \setminus T$, where T is a 1-factor of $K_v^{(3)}$.*

Since $K_v^{(t)}$ has $\binom{v}{t}$ edges, and a Berge type Hamiltonian cycle contains v edges, we must have $v \mid \binom{v}{t}$ for a Berge type Hamiltonian decomposition of $K_v^{(t)}$ to exist. In [4], the authors conjecture that this obvious necessary condition is sufficient for all v and t, and in [15], the authors prove this for $v \geq 30$:

Theorem 1.2.3 ([15])**.** *Suppose that $v \geq 30$, $4 \leq t < v$, and $v \mid \binom{v}{t}$. Then $K_v^{(t)}$ has a Berge type Hamiltonian decomposition.*

This is a clear analogue of Walecki's result for 2-uniform (hyper)graphs. Hence, the problem remains open for $4 \leq t < v < 30$.

A stronger definition for a Hamiltonian cycle of a t-uniform hypergraph H of order v is given by Katona and Kierstead in [11]:

Definition 1.2.4. Let H be a t-uniform hypergraph of order v, and suppose that $V(H) = \{x_0, x_1, \ldots, x_{v-1}\}$. A *Katona-Kierstead type Hamiltonian cycle* in H is a cyclic ordering $(x_0, x_1, \ldots, x_{v-1})$ of $V(H)$ such that $\{x_i, x_{i+1}, \ldots, x_{i+t-1}\} \in E(H)$ for each $i = 0, 1, \ldots, v-1$ (modulo v). □

The authors define paths and cycles on a t-uniform hypergraph in the same manner. A *Katona-Kierstead type Hamiltonian decomposition* of H is a decomposition of H into Katona-Kierstead type Hamiltonian cycles.

In [1], the authors conjecture that the obvious necessary condition is also sufficient for Katona-Kierstead Hamiltonian decompositions:

Conjecture 1.2.5 ([1])**.** *For $v \geq 5$ and $2 \leq t \leq v-2$, there exists a Katona-Kierstead type Hamiltonian decomposition of $K_v^{(t)}$ iff $v \mid \binom{v}{t}$.*

The conjecture has been solved for $t = 2$ by Walecki, for $t = 3$ with $v \leq 32$, and for $(v, t) \in \{(9, 4), (13, 4)\}$, in [1] and [16], as well as the equivalent cases with $t' = v - t$, but the conjecture otherwise remains open.

The following are also open conjectures about Katona-Kierstead type Hamiltonian decompositions, which are discussed in [16] and [14]:

Conjecture 1.2.6 ([16])**.** *For all $v \equiv 0 \pmod 3$, there exists a Katona-Kierstead type Hamiltonian decomposition of $K_v^{(3)} \setminus T$, where T is a 1-factor.*

The authors also establish this conjecture for $v \leq 12$. This conjecture is a clear analogue of Theorem 1.2.2, and of Walecki's result. The following conjecture extends Conjecture 1.2.5 to complete 3-uniform multihypergraphs $\lambda K_v^{(3)}$.

Conjecture 1.2.7 ([16])**.** *For $v \geq 4$, a (Katona-Kierstead type) Hamiltonian decomposition of $\lambda K_v^{(3)}$ exists iff $\lambda \equiv 0 \pmod 3$ or $v \equiv 1, 2 \pmod 3$ when $\lambda \equiv 1, 2 \pmod 3$.*

The following conjecture considers an analogue of Hamiltonian decompositions of bipartite graphs, whereby $K_{v,v}$ has a Hamiltonian decomposition iff v is even, and if T is a 1-factor, $K_{v,v} \setminus T$ has a decomposition iff v is odd:

Conjecture 1.2.8 ([14])**.** *For $v, t \geq 2$, the complete t-uniform t-partite hypergraph $K_{v,v,\ldots,v}^{(t)}$ has a (Katona-Kierstead type) Hamiltonian decomposition iff $t \mid v^{t-1}$.*

The authors of [14] show this conjecture holds for $t = 4$ and $t \mid v$, and also show that $K_{v,v,v}^{(3)} \setminus T$ has a Katona-Kierstead type Hamiltonian decomposition whenever $3 \nmid v$ and $v \neq 4$.

Chapter 2

Hypergraph Designs

2.1 Definitions

The study of hypergraph decompositions is related to the study of t-wise balanced designs, so there are shared definitions:

Definition 2.1.1. A t-(v, K, λ) design (also called a *t-wise balanced design*) is a pair $(X, \mathcal{B})$ where X is a set of v *points* and $\mathcal{B}$ is a collection of subsets of X, called *blocks*, such that for each $B \in \mathcal{B}$, $|B| \in K$, and every t-subset of X is contained in exactly λ blocks. If K has only one element k, then we write t-(v, k, λ), instead of t-$(v, \{k\}, \lambda)$. $\square$

A t-$(v, k, 1)$ design can also be referred to as a *Steiner system* $S(t, k, v)$. A design is *simple* if no two of its blocks are identical.

There is a natural correspondence between simple t-designs and $K_k^{(t)}$-decompositions of $\lambda K_v^{(t)}$: given a t-(v, k, λ) design $(X, \mathcal{B})$, consider the collection of hypergraphs $\mathcal{D} = \{K_B^{(t)} \mid B \in \mathcal{B}\}$ formed by taking the complete t-uniform hypergraph on each block B. Since each edge of $K_v^{(t)}$ occurs λ times as an edge in some hypergraph of $\mathcal{D}$, it follows that $\mathcal{D}$ is a $K_k^{(t)}$-decomposition of $\lambda K_v^{(t)}$. In the same manner, we can construct a t-(v, k, λ) design from a $K_k^{(t)}$-decomposition of $\lambda K_v^{(t)}$, in which the vertex set of each copy of $K_k^{(t)}$ forms a block of the design.

We can then generalise this concept to other decompositions of $\lambda K_v^{(t)}$:

Definition 2.1.2. If H is a t-uniform hypergraph, an *H-design* of order v and index λ is an H-decomposition of $\lambda K_v^{(t)}$. Each hypergraph in the decomposition is called an *H-block*. Such a design is sometimes denoted a (v, H, λ)-design. $\square$

Unless otherwise stated, it is standard to assume that an H-design has index $\lambda = 1$. An H-design is *simple* if no two of its H-blocks are identical.

If H is a t-uniform hypergraph, the *complete H-design of order v* is defined to be the set of all subhypergraphs of $K_v^{(t)}$ which are isomorphic to H.

The following definitions of group-divisible t-designs and candelabra t-systems are given by Moháscy and Ray-Chaudhuri in [17], where they are used to construct $S(3, k, v)$ Steiner systems. Both of these can be considered generalisations of group divisible designs (as defined in [18]) to $t \geq 3$.

Definition 2.1.3 ([17])**.** A *group-divisible t-design*, or t-GDD of order v, index λ, and block sizes from K is a triple $(X, \mathcal{G}, \mathcal{B})$ where X is a set of v *points*, $\mathcal{G}$ is a partition of X into *groups*, and $\mathcal{B}$ is a collection of subsets of X, called *blocks*, such that

(1) for each $B \in \mathcal{B}$, $|B| \in K$,

(2) for each block $B \in \mathcal{B}$ and each group $G \in \mathcal{G}$, $|B \cap G| \leq 1$, and

(3) every t-subset of X taken from t distinct groups is contained in exactly λ blocks.

□

Let $(X, \mathcal{G}, \mathcal{B})$ be a group-divisible t-design of index 1 where every block has size k, and denote the groups of the design by $\mathcal{G} = \{G_1, G_2, \ldots, G_m\}$. This design corresponds to a $K_k^{(t)}$-decomposition of $K_{G_1,G_2,\ldots,G_m}^{(t)}$, where each block $B \in \mathcal{B}$ corresponds to the block $K_B^{(t)}$ of the decomposition. Hence, decompositions of $K_{v_1,v_2,\ldots,v_m}^{(t)}$ can be considered generalisations of group divisible t-designs.

Definition 2.1.4 ([17])**.** A *candelabra t-system*, or t-CS of order v, index λ and block sizes from K is an ordered tuple $(X, S, \mathcal{G}, \mathcal{B})$ where X is a set of v *points*, $S \subseteq X$ is the *stem* of the candelabra, $\mathcal{G}$ is a partition of $X \setminus S$ into *groups*, and $\mathcal{B}$ is a collection of subsets of X, called *blocks* such that

(1) for each $B \in \mathcal{B}$, $|B| \in K$,

(2) every t-subset of X which is not a subset of $S \cup G$ for any $G \in \mathcal{G}$ is contained in some block, and

(3) for every $G \in \mathcal{G}$, no t-subset of $S \cup G$ is contained in any block.

□

Let $(X, S, \mathcal{G}, \mathcal{B})$ be a candelabra t-system of index 1 where every block has size k, and denote the groups of the system by $\mathcal{G} = \{G_1, \ldots, G_m\}$. This design corresponds to a $K_k^{(t)}$-decomposition of $L^{(t)}_{G_1,\ldots,G_m,[S]}$, where each block $B \in \mathcal{B}$ corresponds to the block $K_B^{(t)}$ of the decomposition. Hence, decompositions of $L^{(t)}_{v_1,\ldots,v_m,[w]}$ can be considered generalisations of candelabra t-systems.

2.2 Initial results

If there exists a (v, H, λ)-design, then there are certain necessary divisiblity conditions to be satisfied. The following lemma is from [6]; if necessary, we define $\gcd(\{0\}) = 0$.

Lemma 2.2.1 ([6]). *Let H and K be hypergraphs. Suppose there exists an H-decomposition of K. Then, for any subset R of $V(K)$ with $|R| \leq |V(H)|$, $\gcd(\{\deg_H(S) : S \subseteq V(H), |S| = |R|\})$ divides $\deg_K(R)$.*

Proof. Let $\mathcal{D}$ be an H-decomposition of K, $R \subseteq V(K)$, and suppose that $|R| \leq |V(H)|$. The H-blocks in $\mathcal{D}$ whose vertex sets contain R partition the edges of K which contain R, so

$$\deg_K(R) = \sum_{\substack{G \in \mathcal{D} \\ R \subseteq V(G)}} \deg_G(R). \tag{2.2.1}$$

For each $G \in \mathcal{D}$ with $R \subseteq V(G)$, we have

$$\deg_G(R) \in \{\deg_H(S) : S \subseteq V(H), |S| = |R|\}.$$

Therefore $\gcd(\{\deg_H(S) : S \subseteq V(H), |S| = |R|\})$ divides $\deg_K(R)$. □

Informally, for any set $R \subseteq V(K)$, there are $\deg_K(R)$ edges incident with R, but each of these edges must occur precisely once in the design $\mathcal{D}$. Hence, for every $G \in \mathcal{D}$ which contains R, the isomorphism $G \simeq H$ carries R into some $S \subseteq V(H)$ with $|S| = |R|$, and so G contains $\deg_H(S)$ of the $\deg_K(R)$ edges incident with R. Therefore, $\deg_K(R)$ must be given by a sum of elements in $\{\deg_H(S) : S \subseteq V(H), |S| = |R|\}$. So, if $\deg_H(S)$ is divisible by some d for all such S, then $\deg_K(R)$ must also be divisible by d.

Note that if $|R| > |V(H)|$, then there are no blocks $G \in \mathcal{D}$ with $R \subseteq V(G)$, so (2.2.1) is the empty sum (and so $\deg_K(R) = 0$), and the statement following holds vacuously. However, $\gcd(\{\deg_H(S) : S \subseteq V(H), |S| = |R|\}) = \gcd(\emptyset)$ is not defined, so the conclusion does not follow.

If we take $K = \lambda K_v^{(t)}$, then we have:

Lemma 2.2.2. *Let H be a t-uniform hypergraph. If there exists a (v, H, λ)-design, then for each $0 \leq i < t$,* $\gcd(\{\deg_H(S) : S \subseteq V(H), |S| = i\})$ *divides* $\lambda\binom{v-i}{t-i}$.

Proof. Apply Lemma 2.2.1 with $K = \lambda K_v^{(t)}$; note that for any $R \subseteq V(K)$ with $|R| = i$, $\deg_K(R) = \lambda\binom{v-i}{t-i}$. □

For example, if we take $H = K_k^{(t)}$, then we have the usual necessary divisibility conditions for t-designs: for each $0 \leq i < t$, $\binom{k-i}{t-i}$ divides $\lambda\binom{v-i}{t-i}$.

If $v < t$, then $\lambda K_v^{(t)}$ contains no edges, and we will consider the empty set to be an H-decomposition of $\lambda K_v^{(t)}$ for any H and λ. Otherwise, if $v \geq t$ and $\mathcal{D} \neq \emptyset$ is a H-design, then it is necessary that there is a subgraph of $\lambda K_v^{(t)}$ isomorphic to H, so $|V(H)| \leq |V(\lambda K_v^{(t)})|$, hence:

Lemma 2.2.3. *Let H be a t-uniform hypergraph. If $\mathcal{D}$ is a (v, H, λ)-design, then either $v < t$ and $\mathcal{D} = \emptyset$ or $v \geq |V(H)|$ and $\mathcal{D} \neq \emptyset$.*

Since $v < t$ is trivial, we shall assume $v \geq t$ unless otherwise stated.

The conditions in Lemmas 2.2.2 and 2.2.3 shall be referred to as the *obvious necessary conditions* for the existence of a (v, H, λ)-design. If H and λ are given, we shall say that an integer $v \geq t$ is *admissible* if the obvious necessary conditions hold.

The following result is an immediate corollary of Baranyai's Theorem in [2], which shows that the obvious necessary conditions are sufficient when H consists of m pairwise disjoint edges.

Theorem 2.2.4 ([2])**.** *Let H be the simple t-uniform hypergraph consisting of m pairwise disjoint edges. There is an H-design of order v if and only if m divides $\binom{v}{t}$ and $mt \leq v$.*

2.3 Small 3-uniform hypergraphs

One key problem in the study of hypergraph decompositions is the existence question for H-designs, namely: given a t-uniform hypergraph H, for which positive integers v does there exist a (v, H, λ)-design?

In this section, we will consider this question for 3-uniform hypergraphs H of bounded size and index 1. If we apply the obvious necessary conditions (Lemmas 2.2.2 and 2.2.3) with $t = 3$ and $\lambda = 1$, then we have:

Lemma 2.3.1 ([6])**.** *Let H be a 3-uniform hypergraph. An integer $v \geq 3$ is admissible if and only if the following conditions are satisfied:*

(1) $v \geq |V(H)|$,

(2) $|E(H)|$ *divides* $\binom{v}{3}$,

(3) $\gcd(\{\deg_H(x) : x \in V(H)\})$ *divides* $\binom{v-1}{2}$, *and*

(4) $\gcd(\{\deg_H(\{x, y\}) : x, y \in V(H), x \neq y\})$ *divides* $v - 2$.

2.3.1 3-uniform hypergraphs of size at most 3

First, we shall consider simple 3-uniform hypergraphs containing at most three edges, by considering every possible case. In combination with results from [2, 6, 8, 10], we shall prove the following result:

Theorem 2.3.2. *Let H be a simple 3-uniform hypergraph containing at most three edges and no isolated vertices. There exists an H-design of order $v \geq 3$ iff v is admissible.*

Up to isomorphism, there are fifteen simple 3-uniform hypergraphs with size at most 3 containing no isolated vertices. These are listed in Table 2.1 below, using the following notation: in the first row, $[1, 2, 3]_{H_{1,1}}$ is used to denote the hypergraph (V, E) with $V = \{1, 2, 3\}$ and $E = \{\{1, 2, 3\}\}$, and $H_{1,1}$ is used to denote any hypergraph isomorphic to $[1, 2, 3]_{H_{1,1}}$. The final column indicates that the existence of $H_{1,1}$-designs is proved in Theorem 2.2.4. The notation for the other rows is similar.

$H_{1,1}$	$[1,2,3]_{H_{1,1}}$	$V = \{1,2,3\}$	$E = \{\{1,2,3\}\}$	2.2.4
$H_{2,1}$	$[1,2,3,4,5,6]_{H_{2,1}}$	$V = \{1,2,3,4,5,6\}$	$E = \{\{1,2,3\},\{4,5,6\}\}$	2.2.4
$H_{2,2}$	$[1,2,3,4,5]_{H_{2,2}}$	$V = \{1,2,3,4,5\}$	$E = \{\{1,2,3\},\{1,4,5\}\}$	2.3.4
$H_{2,3}$	$[1,2,3,4]_{H_{2,3}}$	$V = \{1,2,3,4\}$	$E = \{\{1,2,3\},\{1,2,4\}\}$	2.3.4
$H_{3,1}$	$[1,2,3,4,5,6,7,8,9]_{H_{3,1}}$	$V = \{1,2,3,4,5,6,7,8,9\}$	$E = \{\{1,2,3\},\{4,5,6\},\{7,8,9\}\}$	2.2.4
$H_{3,2}$	$[1,2,3,4,5,6,7,8]_{H_{3,2}}$	$V = \{1,2,3,4,5,6,7,8\}$	$E = \{\{1,2,3\},\{1,4,5\},\{6,7,8\}\}$	2.3.5
$H_{3,3}$	$[1,2,3,4,5,6,7]_{H_{3,3}}$	$V = \{1,2,3,4,5,6,7\}$	$E = \{\{1,2,3\},\{1,2,4\},\{5,6,7\}\}$	2.3.5
$H_{3,4}$	$[1,2,3,4,5,6,7]_{H_{3,4}}$	$V = \{1,2,3,4,5,6,7\}$	$E = \{\{1,2,3\},\{1,4,5\},\{1,6,7\}\}$	2.3.5
$H_{3,5}$	$[1,2,3,4,5,6,7]_{H_{3,5}}$	$V = \{1,2,3,4,5,6,7\}$	$E = \{\{1,2,3\},\{1,4,5\},\{4,6,7\}\}$	2.3.5
$H_{3,6}$	$[1,2,3,4,5,6]_{H_{3,6}}$	$V = \{1,2,3,4,5,6\}$	$E = \{\{1,2,3\},\{1,2,4\},\{4,5,6\}\}$	2.3.5
$H_{3,7}$	$[1,2,3,4,5,6]_{H_{3,7}}$	$V = \{1,2,3,4,5,6\}$	$E = \{\{1,2,3\},\{1,2,4\},\{2,5,6\}\}$	2.3.5
$H_{3,8}$	$[1,2,3,4,5,6]_{H_{3,8}}$	$V = \{1,2,3,4,5,6\}$	$E = \{\{1,2,3\},\{1,4,5\},\{2,4,6\}\}$	2.3.5
$H_{3,9}$	$[1,2,3,4,5]_{H_{3,9}}$	$V = \{1,2,3,4,5\}$	$E = \{\{1,2,3\},\{1,2,4\},\{1,2,5\}\}$	2.3.5
$H_{3,10}$	$[1,2,3,4,5]_{H_{3,10}}$	$V = \{1,2,3,4,5\}$	$E = \{\{1,2,3\},\{1,2,4\},\{1,4,5\}\}$	2.3.5
$H_{3,11}$	$[1,2,3,4,5]_{H_{3,11}}$	$V = \{1,2,3,4,5\}$	$E = \{\{1,2,3\},\{1,2,4\},\{3,4,5\}\}$	2.3.5
$H_{3,12}$	$[1,2,3,4]_{H_{3,12}}$	$V = \{1,2,3,4\}$	$E = \{\{1,2,3\},\{1,2,4\},\{1,3,4\}\}$	2.3.3

Table 2.1: All simple 3-uniform hypergraphs of size at most 3

It can be verified exhaustively that these are indeed all the simple 3-uniform hypergraphs of size at most 3.

If $H \simeq H_{k,1}$ for $k \in \{1, 2, 3\}$, then H consists of k pairwise disjoint edges, so H-designs exist iff $v \geq 3k$ and k divides $\binom{v}{3}$, by Theorem 2.2.4.

In [8], the authors consider a copy of $K_4^{(3)}$ with an edge removed, denoted by $K_4^{(3)} - e$; this is isomorphic to $H_{3,12}$. They determined the following result:

Theorem 2.3.3 ([8]). *A $(K_4^{(3)} - e)$-design of order v exists iff $v \equiv 0$, 1, or 2 (mod 9).*

When H has size 2, the existence question was solved in [6], and we describe the proof technique below.

Theorem 2.3.4 ([6]). *Let $H \simeq H_{2,2}$ or $H_{2,3}$. An H-design of order v exists if and only if $v \equiv 0$, 1, or 2 (mod 4) and $v \geq |V(H)|$.*

Since each of $H_{2,2}$ and $H_{2,3}$ contains at least one vertex of degree 1, the obvious necessary conditions reduce to $v \geq |V(H)|$ and $|E(H)| = 2$ divides $\binom{v}{3}$. This second condition is equivalent to $v \equiv 0$, 1, or 2 (mod 4).

To construct an H-design of order $v = 4n + \epsilon$, where $\epsilon \in \{0, 1, 2\}$ and $n \geq 1$ (excluding the case $H \simeq H_{2,2}$ and $v = 4$), construct a collection of pairwise disjoint sets $V_1, V_2, \ldots, V_n$, each of size 4. Let ∞_1, ∞_2 be two points not contained in $\bigcup_{i=1}^n V_i$, and let $I_0 = \emptyset$, $I_1 = \{\infty_1\}$, $I_2 = \{\infty_1, \infty_2\}$. We then aim to construct an H-decomposition of $K = K^{(3)}_{V_1 \cup V_2 \cup \cdots \cup V_n \cup I_\epsilon} \simeq K_v^{(3)}$.

For each edge $e \in E(K)$, consider the sets V_i which have non-empty intersection with e:

- If $e \subseteq V_i \cup I_\epsilon$ for some $i \in \{1, \ldots, n\}$, then e is an edge of $K^{(3)}_{V_i \cup I_\epsilon} \simeq K^{(3)}_{4+\epsilon}$.
- If $e \subseteq V_i \cup V_j \cup I_\epsilon$ for distinct $i, j \in \{1, \ldots, n\}$, and e has at least one vertex in each of V_i and V_j, then e is an edge of $L^{(3)}_{V_i, V_j, [I_\epsilon]} \simeq L^{(3)}_{4,4,[\epsilon]}$.
- If $e \subseteq V_i \cup V_j \cup V_k$ for distinct $i, j, k \in \{1, \ldots, n\}$, and e has at least one vertex in each of V_i, V_j, and V_k, then e is an edge of $K^{(3)}_{V_i, V_j, V_k} \simeq K^{(3)}_{4,4,4}$.

Therefore,

$$K = \left(\bigcup_{1 \leq i \leq n} K^{(3)}_{V_i \cup I_\epsilon} \right) \cup \left(\bigcup_{1 \leq i < j \leq n} L^{(3)}_{V_i, V_j, [I_\epsilon]} \right) \cup \left(\bigcup_{1 \leq i < j < k \leq n} K^{(3)}_{V_i, V_j, V_k} \right)$$

describes a Γ_ϵ-decomposition of K, where

$$\Gamma_\epsilon = \left\{ K^{(3)}_{4+\epsilon}, L^{(3)}_{4,4,[\epsilon]}, K^{(3)}_{4,4,4} \right\}.$$

Hence, if we can find an H-decomposition of each $G \in \Gamma_\epsilon$, this can be extended to an H-decomposition of K, as required.

Examples of H-decompositions of G for each $H \in \{H_{2,2}, H_{2,3}\}$ and $G \in \left\{K_4^{(3)}, K_5^{(3)}, K_6^{(3)}, L_{4,4}^{(3)}, L_{4,4,[1]}^{(3)}, L_{4,4,[2]}^{(3)}, K_{4,4,4}^{(3)}\right\}$ are given in [6], except when $H \simeq H_{2,2}$ and $G \simeq K_4^{(3)}$, so this completes the proof aside from the case $H \simeq H_{2,2}$ and $v \equiv 0 \pmod 4$, $v \geq 8$.

In this remaining case,

$$K = K_{V_{n-1}\cup V_n}^{(3)} \cup \left(\bigcup_{1\leq i\leq n-2} K_{V_i\cup V_n}^{(3)} \setminus K_{V_n}^{(3)} \right) \cup \left(\bigcup_{1\leq i<j\leq n-1} L_{V_i,V_j}^{(3)} \right) \cup \left(\bigcup_{1\leq i<j<k\leq n} K_{V_i,V_j,V_k}^{(3)} \right)$$

describes a Γ'-decomposition of K, where

$$\Gamma' = \left\{K_8^{(3)}, K_8^{(3)} \setminus K_4^{(3)}, L_{4,4}^{(3)}, K_{4,4,4}^{(3)}\right\}.$$

It can be shown that $H_{2,3}$-decompositions of G exist for each $G \in \Gamma'$, so this can be extended to an $H_{2,3}$-decomposition of $v = 4n$.

We use a similar method to prove Theorems 2.3.5 and 2.3.8 below.

Theorem 2.3.5. *Let H be a simple 3-uniform hypergraph with three edges. There exists an H-design of order v if and only if $v \equiv 0$, 1, or 2 (mod 9).*

The case where H has five or six vertices has been solved in [6], so it remains to prove the result when $H \simeq H_{3,2}$, $H_{3,3}$, $H_{3,4}$, or $H_{3,5}$.

Proof. Suppose that H is isomorphic to one of $H_{3,2}$, $H_{3,3}$, $H_{3,4}$, or $H_{3,5}$. From Lemma 2.3.1, v is admissible if and only if $v \equiv 0$, 1, or 2 (mod 9), it remains to show this is sufficient.

Suppose that $v = 9n + \epsilon$ for $n \geq 1$ and $\epsilon \in \{0, 1, 2\}$. Construct n pairwise disjoint sets $V_1, \ldots, V_n$ of size 9, let ∞_1, ∞_2 be two distinct points, neither of which is in $\bigcup_{i=1}^n V_i$, and let $I_0 = \emptyset$, $I_1 = \{\infty_1\}$ and $I_2 = \{\infty_1, \infty_2\}$.

For each $1 \leq i \leq n$ and $\epsilon \in \{0, 1, 2\}$, let $\mathcal{D}_i^\epsilon$ be an H-decomposition of $K_{V_i\cup I_\epsilon}^{(3)}$, which exists by Examples 2.A.1, 2.A.2, and 2.A.3 for $\epsilon = 0$, 1, and 2 respectively. For each $1 \leq i < j \leq n$ and each $\epsilon \in \{0, 1, 2\}$, let $\mathcal{D}_{i,j}^\epsilon$ be an H-decomposition of $L_{V_i,V_j,[I_\epsilon]}^{(3)}$, which exists by Examples 2.A.5, 2.A.6, and 2.A.7 for $\epsilon = 0$, 1, and 2 respectively. For each $1 \leq i < j < k \leq n$, let $\mathcal{D}_{i,j,k}$ be an H-decomposition of $K_{V_i,V_j,V_k}^{(3)}$, which exists by Example 2.A.4.

Then,

$$\mathcal{D} = \left(\bigcup_{1\leq i\leq n} \mathcal{D}_i^\epsilon \right) \cup \left(\bigcup_{1\leq i<j\leq n} \mathcal{D}_{i,j}^\epsilon \right) \cup \left(\bigcup_{1\leq i<j<k\leq n} \mathcal{D}_{i,j,k}^\epsilon \right)$$

is an H-design of order $v = 9n + \epsilon$, for each $n \geq 1$ and $\epsilon \in \{0, 1, 2\}$. $\square$

When H is isomorphic to one of $H_{3,6}$, $H_{3,7}$, $H_{3,8}$, $H_{3,9}$ or $H_{3,10}$, the proof is essentially identical, since it has been shown in [6] that there exist H-decompositions of $K_v^{(3)}$ for each $v \in \{9, 10, 11\}$, $L^{(3)}_{9,9,[\epsilon]}$ for each $\epsilon \in \{0, 1, 2\}$, and $K^{(3)}_{9,9,9}$.

The case $H \simeq H_{3,11}$ must be treated differently, since there is no subgraph of $K^{(3)}_{9,9,9}$ isomorphic to $H_{3,11}$ (and hence there does not exist an $H_{3,11}$-decomposition of $K^{(3)}_{9,9,9}$).

Instead, the authors of [6] aim to find a Γ_ϵ-decomposition of the complete 3-uniform hypergraph $K = K^{(3)}_{V_1 \cup \cdots \cup V_n \cup I_\epsilon}$, where

$$\Gamma_\epsilon = \left\{ K_9^{(3)}, L^{(3)}_{9,9,[\epsilon]}, L^{(3)}_{U,V,[I_\epsilon]} \cup K^{(3)}_{U,V,W}, K^{(3)}_{9,9,9,9} \right\},$$

where U, V, W, I_ϵ are pairwise disjoint sets with $|U| = |V| = |W| = 9$ and $|I_\epsilon| = \epsilon$.

To do this, define $K^{(2,3)}_{\{1,\dots,n\}} = K^{(2)}_{\{1,\dots,n\}} \cup K^{(3)}_{\{1,\dots,n\}}$; that is, $K^{(2,3)}_{\{1,\dots,n\}}$ has vertex set $V = \{1, 2, \dots, n\}$, and edge set consisting of all 2-element and 3-element subsets of V. It can be shown that there exists a Λ-decomposition $\mathcal{D}'$ of $K^{(2,3)}_{\{1,\dots,n\}}$, where $\Lambda = \left\{ K_2^{(2)}, [1, 2, 3]_A, K_4^{(3)} \right\}$, and where $[1, 2, 3]_A$ is the hypergraph $(\{1, 2, 3\}, \{\{1, 2\}, \{1, 2, 3\}\})$. So, $\mathcal{D}'$ gives rise to a Γ_ϵ-decomposition of K given by

$$\begin{aligned} K = &\left(\bigcup_{1 \leq i \leq n} K^{(3)}_{V_i \cup I_\epsilon} \right) \cup \left(\bigcup_{K^{(2)}_{\{i,j\}} \in \mathcal{D}'} L^{(3)}_{V_i, V_j, [I_\epsilon]} \right) \\ &\cup \left(\bigcup_{[i,j,k]_A \in \mathcal{D}'} L^{(3)}_{V_i, V_j, [I_\epsilon]} \cup K^{(3)}_{V_i, V_j, V_k} \right) \cup \left(\bigcup_{K^{(3)}_{\{i,j,k,l\}} \in \mathcal{D}'} K^{(3)}_{V_i, V_j, V_k, V_l} \right), \end{aligned}$$

Informally, $\mathcal{D}'$ describes a pattern of arranging copies of $L \in \Lambda$ to give every pair and triple of points from the set $\{1, 2, \dots, n\}$, so we can therefore cover every $L^{(3)}_{V_i, V_j, [I_\epsilon]}$ and every $K^{(3)}_{V_i, V_j, V_k}$ using this pattern.

For each $\epsilon \in \{0, 1, 2\}$ and each $G \in \Gamma_\epsilon$, it can be shown that there exists an $H_{3,11}$-decomposition of G, so we can construct an $H_{3,11}$-decomposition of K.

This completes the proof of Theorem 2.3.2. $\square$

2.3.2 Regular 3-uniform hypergraphs

Next, we shall consider simple 3-uniform hypergraphs of larger size, but impose the restriction that the hypergraph is regular. Recall that a hypergraph H is d-regular if every vertex $x \in V(H)$ has degree d.

There are nine non-isomorphic regular 3-uniform hypergraphs of size at most 5, three of which are $H_{1,1}$, $H_{2,1}$ and $H_{3,1}$ from Table 2.1. The remaining six are listed in Table 2.2, using the same notation as Table 2.1.

$H_{4,1}$	$[1,2,\ldots,12]_{H_{4,1}}$	$V=\{1,2,\ldots,12\}$	$E=\{\{1,2,3\},\{4,5,6\},\{7,8,9\},\{10,11,12\}\}$	2.2.4
$H_{4,2}$	$[1,2,3,4,5,6]_{H_{4,2}}$	$V=\{1,2,3,4,5,6\}$	$E=\{\{1,2,3\},\{1,5,6\},\{2,4,6\},\{3,4,5\}\}$	2.3.8
$H_{4,3}$	$[1,2,3,4,5,6]_{H_{4,3}}$	$V=\{1,2,3,4,5,6\}$	$E=\{\{1,2,3\},\{1,5,6\},\{2,3,4\},\{4,5,6\}\}$	2.3.8
$H_{4,4}$	$[1,2,3,4]_{H_{4,4}}$	$V=\{1,2,3,4\}$	$E=\{\{1,2,3\},\{1,2,4\},\{1,3,4\},\{2,3,4\}\}$	2.3.7
$H_{5,1}$	$[1,2,\ldots,15]_{H_{5,1}}$	$V=\{1,2,\ldots,15\}$	$\{\{1,2,3\},\{4,5,6\},\{7,8,9\},\{10,11,12\},\{13,14,15\}\}$	2.2.4
$H_{5,2}$	$[1,2,3,4,5]_{H_{5,2}}$	$V=\{1,2,3,4,5\}$	$E=\{\{1,2,3\},\{1,2,4\},\{1,3,5\},\{2,4,5\},\{3,4,5\}\}$	Open

Table 2.2: All simple d-regular 3-uniform hypergraphs with size $\in \{4,5\}$

It can be easily verified that these are indeed all the regular 3-uniform hypergraphs with size at most 5. In conjunction with existing results, we shall prove the following:

Theorem 2.3.6. *Let H be a simple regular 3-uniform hypergraph containing four edges. There exists an H-design of order $v \geq 3$ iff v is admissible, except that there is no $H_{4,2}$-design of order 6.*

If $H \simeq H_{4,1}$ or $H_{5,1}$, the necessary conditions are sufficient by Theorem 2.2.4.

If $H \simeq H_{4,4} \simeq K_4^{(3)}$, then an H-design of order v is an $S(3,4,v)$ Steiner system (also called a *Steiner quadruple system*), and it can be seen that v is admissible if and only if $v \equiv 2$ or 4 (mod 6). The following result by Hanani establishes that these conditions are sufficient:

Theorem 2.3.7 ([10])**.** *An $S(3,4,v)$ exists iff $v \equiv 2$ or 4 (mod 6).*

For $H \simeq H_{4,2}$ or $H_{4,3}$, we have the following theorem. The case $H \simeq H_{4,2}$ was solved by Bryant et al. in [6], and the proof for $H \simeq H_{4,3}$ is similar.

Theorem 2.3.8. *Let $H \simeq H_{4,2}$ or $H_{4,3}$. An H-design of order v exists iff $v \equiv 1$, 2 or 6 (mod 8), except that there is no $H_{4,2}$-design of order 6.*

Proof. Let H be a hypergraph isomorphic to either $H_{4,2}$ or $H_{4,3}$.

From Lemma 2.3.1, it follows that $v \equiv 1$, 2 or 6 (mod 8) is a necessary condition, it remains to show that this is sufficient, and that there is no $(6, H_{4,2}, 1)$-design.

For $v = 6$, there exists an $H_{4,3}$-design of order 6 by Example 2.A.8, and it was shown in [6] that there does not exist an $H_{4,2}$-design of order 6. So, it remains to consider $v \geq 9$.

Let O denote the hypergraph with vertex set $\{0, 1, 2, 3, 4, 5\}$ and edge set

$$\{\{0, 1, 2\}, \{0, 1, 5\}, \{0, 2, 4\}, \{0, 4, 5\}, \{1, 2, 3\}, \{1, 3, 5\}, \{2, 3, 4\}, \{3, 4, 5\}\}.$$

The edges of O describe the faces of an octahedron. It is clear that there exists an $H_{4,2}$-decomposition of O given by

$$\left\{[0, 1, 2, 3, 4, 5]_{H_{4,2}}, [0, 1, 5, 3, 4, 2]_{H_{4,2}}\right\},$$

and an $H_{4,3}$-decomposition of O given by

$$\left\{[0, 1, 2, 3, 4, 5]_{H_{4,3}}, [0, 1, 5, 3, 2, 4]_{H_{4,3}}\right\}.$$

In [10], it was shown that there exists an O-design of order v whenever $v \equiv 2 \pmod 8$, so there exists an H-design of order v whenever $v \equiv 2 \pmod 8$.

In the remaining cases, we have $v = 8n + \epsilon$ for $n \geq 1$ and $\epsilon \in \{1, 6\}$. Construct a collection of n pairwise disjoint sets $V_1, \ldots, V_n$ of size 8, and let $I = \{\infty_1, \infty_2, \infty_3, \infty_4, \infty_5, \infty_6\}$ be a set of six points, none of which is in $\bigcup_{i=1}^{n} V_i$.

For each $1 \leq i \leq n$:

- let $\mathcal{D}_i$ be an H-decomposition of $K_{V_i \cup \{\infty_1\}}$, which exists for $H_{4,2}$ by [6], and for $H_{4,3}$ by Example 2.A.9; and
- let $\mathcal{D}'_i$ be an H-decomposition of $K^{(3)}_{V_i \cup I} \setminus K^{(3)}_I$, which exists for $H_{4,2}$ by [6], and for $H_{4,3}$ by Example 2.A.10.

Let $\mathcal{D}_\infty$ be an H-decomposition of $K^{(3)}_{V_n \cup I}$, which exists for $H_{4,2}$ by [6], and for $H_{4,3}$ is given by the union of $\mathcal{D}'_n$ and an $H_{4,3}$-decomposition of $K^{(3)}_I$ (which exists by example 2.A.8).

For each $1 \leq i < j \leq n$:

- let $\mathcal{D}_{i,j}$ be an H-decomposition of $L^{(3)}_{V_i, V_j, [\{\infty_1\}]}$, which exists for $H_{4,2}$ by [6], and for $H_{4,3}$ by Example 2.A.12; and
- let $\mathcal{D}'_{i,j}$ be an H-decomposition of $L^{(3)}_{V_i, V_j, [I]}$, which exists for $H_{4,2}$ by [6], and for $H_{4,3}$ by Example 2.A.11.

It was shown in [10] that there exists an O-decomposition of $K^{(3)}_{8,8,8}$, so there exists an H-decomposition of $K^{(3)}_{8,8,8}$. Then, for each $1 \leq i < j < k \leq n$, let $\mathcal{D}_{i,j,k}$ be an H-decomposition of $K^{(3)}_{V_i, V_j, V_k}$.

Then, if $v \equiv 1 \pmod 8$,

$$\left(\bigcup_{1\le i\le n} \mathcal{D}_i\right) \cup \left(\bigcup_{1\le i<j\le n} \mathcal{D}_{i,j}\right) \cup \left(\bigcup_{1\le i<j<k\le n} \mathcal{D}_{i,j,k}\right)$$

is an H-design of order $v = 8n+1$, and if $v \equiv 6 \pmod 8$,

$$\mathcal{D}_\infty \cup \left(\bigcup_{1\le i\le n-1} \mathcal{D}'_i\right) \cup \left(\bigcup_{1\le i<j\le n} \mathcal{D}'_{i,j}\right) \cup \left(\bigcup_{1\le i<j<k\le n} \mathcal{D}_{i,j,k}\right)$$

is an H-design of order $v = 8n+6$. □

This concludes the proof of Theorem 2.3.6. □

Even though there is no $H_{4,2}$-design of order 6, we can still consider the existence of $H_{4,2}$-designs of order 6 with index $\lambda \ge 2$. We have the following results:

Theorem 2.3.9. *There exists a simple* $(6, H_{4,2}, \lambda)$*-design if and only if* $\lambda \in \{2, 3, 4, 6\}$.

Proof. Let $V = \{0, 1, 2, 3, 4, \infty\}$, and let π be the permutation $(0\ 1\ 2\ 3\ 4)(\infty)$. Define six $H_{4,2}$-blocks on V by:

$$\begin{aligned} B_1 &= [\infty, 0, 1, 3, 2, 4]_{H_{4,2}}, \quad & B_2 &= [\infty, 0, 1, 3, 4, 2]_{H_{4,2}}, \\ B_3 &= [\infty, 0, 1, 4, 2, 3]_{H_{4,2}}, \quad & B_4 &= [\infty, 0, 1, 4, 3, 2]_{H_{4,2}}, \\ B_5 &= [\infty, 0, 2, 4, 1, 3]_{H_{4,2}}, \quad & B_6 &= [\infty, 0, 2, 4, 3, 1]_{H_{4,2}}. \end{aligned}$$

Then, let

$$\begin{aligned} \mathcal{D}_2 &= \mathcal{O}_\pi(\{B_1, B_2\}) \\ \mathcal{D}_3 &= \mathcal{O}_\pi(\{B_1, B_4, B_5\}) \\ \mathcal{D}_4 &= \mathcal{O}_\pi(\{B_1, B_2, B_3, B_5\}) \\ \mathcal{D}_6 &= \mathcal{O}_\pi(\{B_1, B_2, B_3, B_4, B_5, B_6\}), \end{aligned}$$

where $\mathcal{O}_\pi(X)$ denotes the orbit of the set X under π. Then, $\mathcal{D}_\lambda$ is a simple $(6, H_{4,2}, \lambda)$-design for each $\lambda \in \{2, 3, 4, 6\}$. It remains to show $\lambda \in \{2, 3, 4, 6\}$ is necessary.

If we can show that $\mathcal{D}_6$ is complete, then there cannot exist a simple $(6, H_{4,2}, \lambda)$-design for $\lambda > 6$. The following three permutations are automorphims of $[1, 2, 3, 4, 5, 6]_{H_{4,2}}$:

$$(1\,2\,3)(4\,5\,6), \quad (1\,2)(4\,5), \quad (1\,4)(2\,5).$$

The group generated by these three elements has order 24, so $|\operatorname{Aut}(H_{4,2})| \geq 24$. But, $\mathcal{D}_6$ contains $30 = \binom{6}{6}\frac{6!}{24}$ distinct $H_{4,2}$-blocks, so we conclude that $|\operatorname{Aut}(H_{4,2})| = 24$, and $\mathcal{D}_6$ is complete.[1]

If there existed a simple $(6, H_{4,2}, 5)$-design $\mathcal{D}_5$, then its complement $\mathcal{D}_6 \backslash \mathcal{D}_5$ would be a $(6, H_{4,2}, 1)$-design, which does not exist by Theorem 2.3.8. This completes the proof. □

Theorem 2.3.10. *There exists a (not necessarily simple)* $(6, H_{4,2}, \lambda)$*-design if and only if* $\lambda \geq 2$.

Proof. A $(6, H_{4,2}, 1)$-design does not exist by Theorem 2.3.8, so $\lambda \geq 2$ is necessary.

By Theorem 2.3.9, there exists a $(6, H_{4,2}, 2)$-design $\mathcal{D}_2$ and a $(6, H_{4,2}, 3)$-design $\mathcal{D}_3$.

For any $\lambda \geq 4$, let x, y be positive integers such that $\lambda = 2x + 3y$ (it is straightforward to see why such integers must exist). Then, the collection containing x copies of $\mathcal{D}_2$ and y copies of $\mathcal{D}_3$ is a $(6, H_{4,2}, \lambda)$-design. □

The case $H \simeq H_{5,2}$ remains an open problem, but is discussed in [16]. The necessary conditions require that $v \equiv 1, 2, 5, 7, 10$ or $11 \pmod{15}$, and we have the following results:

Theorem 2.3.11 ([16])**.** *An* $H_{5,2}$*-design of order* v *exists for all admissible* $v \leq 17$*, and for all* $v = 4^m + 1$ *for* m *a positive integer.*

Proof. There exists an $H_{5,2}$-design of order 5 on the vertex set $\{0, 1, 2, 3, 4\}$ given by $\big\{[0, 1, 2, 3, 4]_{H_{5,2}}, [4, 0, 1, 3, 2]_{H_{5,2}}\big\}$.

Hence, if there exists an $S(3, 5, v)$, then there exists an $H_{5,2}$-design of order v, given by replacing each block with an $H_{5,2}$-decomposition of $K_5^{(3)}$. It is known that there exists an $S(3, 5, v)$ for all $v = 4^m + 1$, given by spherical geometries (see [13]), so there is also an $H_{5,2}$-design of order $4^m + 1$.

Examples of $H_{5,2}$-designs of order 7, 10, and 11 are given in [16]. □

2.4 Further results

In [8], the authors consider a copy of $K_4^{(3)}$ with an edge removed, denoted by $K_4^{(3)} - e$, and in [9], the authors consider the graph denoted $K_4^{(3)} + e$, with vertex set $V = \{1, 2, 3, 4, 5\}$ and edges

$$E = \{\{1, 2, 3\}, \{1, 2, 4\}, \{1, 3, 4\}, \{2, 3, 4\}, \{3, 4, 5\}\} :$$

[1] Alternatively, one can enumerate all $H_{4,2}$-blocks on 6 vertices (by computer search or other means), and check that there are precisely 30.

Theorem 2.4.1 ([9]). *A $(K_4^{(3)}+e)$-design of order v exists iff $v \equiv 0$, 1, or 2 (mod 5) and $v \neq 5$.*

In [12], Keevash considers $K_k^{(t)}$-decompositions of an arbitrary t-uniform hypergraph H, and gives the following result.

Theorem 2.4.2 ([12]). *Let H be a t-uniform hypergraph on n vertices, and let $k \geq t$ be an integer such that $K_k^{(t)}$ satisfies the conditions of Lemma 2.2.1 for the existence of a $K_k^{(t)}$-decomposition of H.*

Suppose that c is a positive real number, and d, h are positive integers such that $\frac{1}{n} \ll c \ll d$, $\frac{1}{h} \ll \frac{1}{k}$, and $|E(H)| > dn^t$. Suppose also that there exists some real number $p > 0$ such that for any collection $\mathcal{A}$ of $(t-1)$-element subsets of $V(H)$ with $|\mathcal{A}| \leq h$,

$$|\{x \in V(H) : \forall S \in \mathcal{A}, S \cup \{x\} \in E(H)\}| = (1 \pm c)p^{|\mathcal{A}|}n.$$

Then there exists an $K_k^{(t)}$-decomposition of H.

Informally, this result states that if H contains sufficiently many edges and satisfies certain connectedness conditions, then the conditions of Lemma 2.2.1 are also sufficient for the existence of a $K_k^{(t)}$-decomposition of H.

If we take $H = K_v^{(t)}$, then we have as an immediate corollary that for sufficiently large v, there exists an $S(t, k, v)$ Steiner system.

2.A Examples

Example 2.A.1. There exists an $H_{3,i}$-design of order 9 for each $i \in \{2, 3, 4, 5\}$.

Proof. For each of the following hypergraphs, the orbits of the following blocks under the permutation $(0\,1\,2\,3\,4\,5\,6)(\infty_1)(\infty_2)$ give the required designs.

$H_{3,2}$	$[\infty_1, 0, 1, 2, \infty_2, 3, 4, 5]_{H_{3,2}}$, $[\infty_1, 0, 2, 1, 4, 5, 6, \infty_2]_{H_{3,2}}$, $[0, 1, 4, 2, 6, 3, 5, \infty_2]_{H_{3,2}}$, $[0, 1, 5, 2, 4, 3, 6, \infty_2]_{H_{3,2}}$
$H_{3,3}$	$[0, \infty_1, 1, \infty_2, 2, 3, 4]_{H_{3,3}}$, $[0, \infty_1, 2, 3, 4, 5, \infty_2]_{H_{3,3}}$, $[0, \infty_2, 2, 3, 1, 5, 6]_{H_{3,3}}$, $[0, 3, 4, 5, 1, 2, 6]_{H_{3,3}}$
$H_{3,4}$	$[\infty_1, 0, 1, 2, 4, 3, \infty_2]_{H_{3,4}}$, $[0, 1, 2, 3, \infty_1, 6, \infty_2]_{H_{3,4}}$, $[0, 1, 4, 2, 6, 5, \infty_2]_{H_{3,4}}$, $[0, 1, 5, 2, 4, 3, \infty_2]_{H_{3,4}}$
$H_{3,5}$	$[\infty_1, 0, 1, \infty_2, 2, 3, 4]_{H_{3,5}}$, $[0, 2, \infty_2, \infty_1, 3, 1, 6]_{H_{3,5}}$, $[0, 1, 2, 3, \infty_2, 4, 6]_{H_{3,5}}$, $[0, 1, 4, 5, 2, 3, 6]_{H_{3,5}}$

□

Example 2.A.2. There exists an $H_{3,i}$-design of order 10 for each $i \in \{2, 3, 4, 5\}$.

Proof. For each of the following hypergraphs, the orbits of the following blocks under the action of $\mathbb{Z}_{10}$ give the required designs.

$H_{3,2}$	$[0,1,2,3,9,4,5,7]_{H_{3,2}}$, $[0,1,5,2,3,4,8,9]_{H_{3,2}}$, $[0,2,4,3,8,1,5,7]_{H_{3,2}}$, $[0,1,7,3,5,2,6,9]_{H_{3,2}}$
$H_{3,3}$	$[0,1,2,3,4,5,8]_{H_{3,3}}$, $[0,1,5,8,2,6,7]_{H_{3,3}}$, $[0,2,4,5,1,3,7]_{H_{3,3}}$, $[0,3,4,5,2,6,9]_{H_{3,3}}$
$H_{3,4}$	$[0,1,2,3,9,7,8]_{H_{3,4}}$, $[0,1,6,2,3,4,9]_{H_{3,4}}$, $[0,2,8,4,6,5,7]_{H_{3,4}}$, $[0,1,7,3,6,5,8]_{H_{3,4}}$
$H_{3,5}$	$[0,1,2,3,9,4,6]_{H_{3,5}}$, $[0,1,5,2,3,6,7]_{H_{3,5}}$, $[0,2,4,3,8,1,7]_{H_{3,5}}$, $[0,1,7,3,5,6,9]_{H_{3,5}}$

□

Example 2.A.3. There exists an $H_{3,i}$-design of order 11 for each $i \in \{2,3,4,5\}$.

Proof. For each of the following hypergraphs, the orbits of the following blocks under the action of $\mathbb{Z}_{11}$ give the required designs.

$H_{3,2}$	$[0,1,2,3,10,4,5,7]_{H_{3,2}}$, $[0,1,5,2,3,4,9,10]_{H_{3,2}}$, $[0,2,4,3,9,1,5,10]_{H_{3,2}}$, $[0,1,8,3,5,2,4,9]_{H_{3,2}}$, $[0,1,7,3,6,2,5,9]_{H_{3,2}}$
$H_{3,3}$	$[0,1,2,3,4,5,8]_{H_{3,3}}$, $[0,1,5,6,7,9,10]_{H_{3,3}}$, $[0,2,4,5,1,3,7]_{H_{3,3}}$, $[0,2,7,8,1,4,5]_{H_{3,3}}$, $[0,3,6,7,1,2,8]_{H_{3,3}}$
$H_{3,4}$	$[0,1,2,3,10,8,9]_{H_{3,4}}$, $[0,1,6,2,3,4,10]_{H_{3,4}}$, $[0,2,4,3,9,5,7]_{H_{3,4}}$, $[0,1,8,2,7,3,5]_{H_{3,4}}$, $[0,1,7,3,6,4,8]_{H_{3,4}}$
$H_{3,5}$	$[0,1,2,3,10,4,6]_{H_{3,5}}$, $[0,1,5,2,3,7,8]_{H_{3,5}}$, $[0,2,4,3,9,1,7]_{H_{3,5}}$, $[0,1,8,2,7,4,10]_{H_{3,5}}$, $[0,1,7,3,8,6,10]_{H_{3,5}}$

□

Example 2.A.4. For each $i \in \{2,3,4,5\}$, there exists an $H_{3,i}$-decomposition of $K^{(3)}_{9,9,9}$.

Proof. Let

$$\begin{aligned} U &= \{u_0,u_1,u_2,u_3,u_4,u_5,u_6,u_7,u_8\},\\ V &= \{v_0,v_1,v_2,v_3,v_4,v_5,v_6,v_7,v_8\},\\ W &= \{w_0,w_1,w_2,w_3,w_4,w_5,w_6,w_7,w_8\} \end{aligned}$$

be three pairwise disjoint sets of size 9. Let π be the permutation which maps $u_i \mapsto u_{i+1}$, $v_i \mapsto v_{i+1}$, $w_i \mapsto w_{i+1}$, where addition is performed in $\mathbb{Z}_9$. For each of the following hypergraphs, the union of the orbits of the following blocks under π forms an $H_{3,i}$-decomposition of $K^{(3)}_{U,V,W}$.

$H_{3,2}$	$[w_0, u_0, v_0, u_7, v_7, u_1, v_1, w_2]_{H_{3,2}}$, $[w_0, u_4, v_4, u_5, v_5, u_0, v_0, w_3]_{H_{3,2}}$, $[w_0, u_1, v_1, u_2, v_2, u_0, v_0, w_6]_{H_{3,2}}$, $[w_0, u_0, v_1, u_7, v_8, u_1, v_2, w_2]_{H_{3,2}}$, $[w_0, u_4, v_5, u_5, v_6, u_0, v_1, w_3]_{H_{3,2}}$, $[w_0, u_1, v_2, u_2, v_3, u_0, v_1, w_6]_{H_{3,2}}$, $[w_0, u_0, v_2, u_7, v_0, u_1, v_3, w_2]_{H_{3,2}}$, $[w_0, u_4, v_6, u_5, v_7, u_0, v_2, w_3]_{H_{3,2}}$, $[w_0, u_1, v_3, u_2, v_4, u_0, v_2, w_6]_{H_{3,2}}$, $[w_0, u_0, v_3, u_7, v_1, u_1, v_4, w_2]_{H_{3,2}}$, $[w_0, u_4, v_7, u_5, v_8, u_0, v_3, w_3]_{H_{3,2}}$, $[w_0, u_1, v_4, u_2, v_5, u_0, v_3, w_6]_{H_{3,2}}$, $[w_0, u_0, v_4, u_7, v_2, u_1, v_5, w_2]_{H_{3,2}}$, $[w_0, u_4, v_8, u_5, v_0, u_0, v_4, w_3]_{H_{3,2}}$, $[w_0, u_1, v_5, u_2, v_6, u_0, v_4, w_6]_{H_{3,2}}$, $[w_0, u_0, v_5, u_7, v_3, u_1, v_6, w_2]_{H_{3,2}}$, $[w_0, u_4, v_0, u_5, v_1, u_0, v_5, w_3]_{H_{3,2}}$, $[w_0, u_1, v_6, u_2, v_7, u_0, v_5, w_6]_{H_{3,2}}$, $[w_0, u_0, v_6, u_7, v_4, u_1, v_7, w_2]_{H_{3,2}}$, $[w_0, u_4, v_1, u_5, v_2, u_0, v_6, w_3]_{H_{3,2}}$, $[w_0, u_1, v_7, u_2, v_8, u_0, v_6, w_6]_{H_{3,2}}$, $[w_0, u_0, v_7, u_7, v_5, u_1, v_8, w_2]_{H_{3,2}}$, $[w_0, u_4, v_2, u_5, v_3, u_0, v_7, w_3]_{H_{3,2}}$, $[w_0, u_1, v_8, u_2, v_0, u_0, v_7, w_6]_{H_{3,2}}$, $[w_0, u_0, v_8, u_7, v_6, u_1, v_0, w_2]_{H_{3,2}}$, $[w_0, u_4, v_3, u_5, v_4, u_0, v_8, w_3]_{H_{3,2}}$, $[w_0, u_1, v_0, u_2, v_1, u_0, v_8, w_6]_{H_{3,2}}$
$H_{3,3}$	$[u_0, v_0, w_0, w_1, u_1, v_1, w_3]_{H_{3,3}}$, $[u_0, v_0, w_3, w_4, u_1, v_1, w_6]_{H_{3,3}}$, $[u_0, v_0, w_6, w_7, u_1, v_1, w_0]_{H_{3,3}}$, $[u_0, v_1, w_0, w_1, u_1, v_2, w_3]_{H_{3,3}}$, $[u_0, v_1, w_3, w_4, u_1, v_2, w_6]_{H_{3,3}}$, $[u_0, v_1, w_6, w_7, u_1, v_2, w_0]_{H_{3,3}}$, $[u_0, v_2, w_0, w_1, u_1, v_3, w_3]_{H_{3,3}}$, $[u_0, v_2, w_3, w_4, u_1, v_3, w_6]_{H_{3,3}}$, $[u_0, v_2, w_6, w_7, u_1, v_3, w_0]_{H_{3,3}}$, $[u_0, v_3, w_0, w_1, u_1, v_4, w_3]_{H_{3,3}}$, $[u_0, v_3, w_3, w_4, u_1, v_4, w_6]_{H_{3,3}}$, $[u_0, v_3, w_6, w_7, u_1, v_4, w_0]_{H_{3,3}}$, $[u_0, v_4, w_0, w_1, u_1, v_5, w_3]_{H_{3,3}}$, $[u_0, v_4, w_3, w_4, u_1, v_5, w_6]_{H_{3,3}}$, $[u_0, v_4, w_6, w_7, u_1, v_5, w_0]_{H_{3,3}}$, $[u_0, v_5, w_0, w_1, u_1, v_6, w_3]_{H_{3,3}}$, $[u_0, v_5, w_3, w_4, u_1, v_6, w_6]_{H_{3,3}}$, $[u_0, v_5, w_6, w_7, u_1, v_6, w_0]_{H_{3,3}}$, $[u_0, v_6, w_0, w_1, u_1, v_7, w_3]_{H_{3,3}}$, $[u_0, v_6, w_3, w_4, u_1, v_7, w_6]_{H_{3,3}}$, $[u_0, v_6, w_6, w_7, u_1, v_7, w_0]_{H_{3,3}}$, $[u_0, v_7, w_0, w_1, u_1, v_8, w_3]_{H_{3,3}}$, $[u_0, v_7, w_3, w_4, u_1, v_8, w_6]_{H_{3,3}}$, $[u_0, v_7, w_6, w_7, u_1, v_8, w_0]_{H_{3,3}}$, $[u_0, v_8, w_0, w_1, u_1, v_0, w_3]_{H_{3,3}}$, $[u_0, v_8, w_3, w_4, u_1, v_0, w_6]_{H_{3,3}}$, $[u_0, v_8, w_6, w_7, u_1, v_0, w_0]_{H_{3,3}}$
$H_{3,4}$	$[w_0, u_0, v_0, u_7, v_7, u_8, v_8]_{H_{3,4}}$, $[w_3, u_0, v_0, u_7, v_7, u_8, v_8]_{H_{3,4}}$, $[w_6, u_0, v_0, u_7, v_7, u_8, v_8]_{H_{3,4}}$, $[w_0, u_0, v_1, u_7, v_8, u_8, v_0]_{H_{3,4}}$, $[w_3, u_0, v_1, u_7, v_8, u_8, v_0]_{H_{3,4}}$, $[w_6, u_0, v_1, u_7, v_8, u_8, v_0]_{H_{3,4}}$, $[w_0, u_0, v_2, u_7, v_0, u_8, v_1]_{H_{3,4}}$, $[w_3, u_0, v_2, u_7, v_0, u_8, v_1]_{H_{3,4}}$, $[w_6, u_0, v_2, u_7, v_0, u_8, v_1]_{H_{3,4}}$, $[w_0, u_0, v_3, u_7, v_1, u_8, v_2]_{H_{3,4}}$, $[w_3, u_0, v_3, u_7, v_1, u_8, v_2]_{H_{3,4}}$, $[w_6, u_0, v_3, u_7, v_1, u_8, v_2]_{H_{3,4}}$, $[w_0, u_0, v_4, u_7, v_2, u_8, v_3]_{H_{3,4}}$, $[w_3, u_0, v_4, u_7, v_2, u_8, v_3]_{H_{3,4}}$, $[w_6, u_0, v_4, u_7, v_2, u_8, v_3]_{H_{3,4}}$, $[w_0, u_0, v_5, u_7, v_3, u_8, v_4]_{H_{3,4}}$, $[w_3, u_0, v_5, u_7, v_3, u_8, v_4]_{H_{3,4}}$, $[w_6, u_0, v_5, u_7, v_3, u_8, v_4]_{H_{3,4}}$, $[w_0, u_0, v_6, u_7, v_4, u_8, v_5]_{H_{3,4}}$, $[w_3, u_0, v_6, u_7, v_4, u_8, v_5]_{H_{3,4}}$, $[w_6, u_0, v_6, u_7, v_4, u_8, v_5]_{H_{3,4}}$, $[w_0, u_0, v_7, u_7, v_5, u_8, v_6]_{H_{3,4}}$, $[w_3, u_0, v_7, u_7, v_5, u_8, v_6]_{H_{3,4}}$, $[w_6, u_0, v_7, u_7, v_5, u_8, v_6]_{H_{3,4}}$, $[w_0, u_0, v_8, u_7, v_6, u_8, v_7]_{H_{3,4}}$, $[w_3, u_0, v_8, u_7, v_6, u_8, v_7]_{H_{3,4}}$, $[w_6, u_0, v_8, u_7, v_6, u_8, v_7]_{H_{3,4}}$

$H_{3,5}$	$[v_0, u_0, w_0, w_8, u_8, u_7, v_7]_{H_{3,5}}$, $[u_0, v_0, w_2, v_1, w_1, u_1, w_4]_{H_{3,5}}$, $[u_0, v_0, w_4, v_1, w_2, u_1, w_6]_{H_{3,5}}$, $[u_0, v_0, w_6, v_1, w_3, u_1, w_8]_{H_{3,5}}$, $[u_0, v_0, w_8, w_5, v_1, u_1, v_2]_{H_{3,5}}$, $[u_0, v_1, w_6, v_2, w_0, u_1, w_8]_{H_{3,5}}$, $[u_0, v_1, w_8, w_2, v_2, u_1, v_3]_{H_{3,5}}$, $[u_0, v_2, w_3, v_3, w_0, u_1, w_5]_{H_{3,5}}$, $[u_0, v_2, w_5, v_3, w_1, u_1, w_7]_{H_{3,5}}$, $[u_0, v_2, w_7, v_3, w_2, u_1, w_0]_{H_{3,5}}$, $[u_0, v_3, w_3, v_4, w_0, u_1, w_5]_{H_{3,5}}$, $[u_0, v_3, w_5, v_4, w_1, u_1, w_7]_{H_{3,5}}$, $[u_0, v_3, w_7, v_4, w_2, u_1, w_0]_{H_{3,5}}$, $[u_0, v_4, w_3, v_5, w_0, u_1, w_5]_{H_{3,5}}$, $[u_0, v_4, w_5, v_5, w_1, u_1, w_7]_{H_{3,5}}$, $[u_0, v_4, w_7, v_5, w_2, u_1, w_0]_{H_{3,5}}$, $[u_0, v_5, w_3, v_6, w_0, u_1, w_5]_{H_{3,5}}$, $[u_0, v_5, w_5, v_6, w_1, u_1, w_7]_{H_{3,5}}$, $[u_0, v_5, w_7, v_6, w_2, u_1, w_0]_{H_{3,5}}$, $[u_0, v_6, w_3, v_7, w_0, u_1, w_5]_{H_{3,5}}$, $[u_0, v_6, w_5, v_7, w_1, u_1, w_7]_{H_{3,5}}$, $[u_0, v_6, w_7, v_7, w_2, u_1, w_0]_{H_{3,5}}$, $[u_0, v_7, w_3, v_8, w_0, u_1, w_5]_{H_{3,5}}$, $[u_0, v_7, w_5, w_2, v_8, u_1, v_0]_{H_{3,5}}$, $[u_0, v_7, w_6, w_4, v_8, u_1, v_0]_{H_{3,5}}$, $[u_0, v_7, w_7, w_6, v_8, u_1, v_0]_{H_{3,5}}$, $[v_0, u_2, w_1, w_8, u_1, u_0, v_8]_{H_{3,5}}$

□

Example 2.A.5. For each $i \in \{2,3,4,5\}$, there exists an $H_{3,i}$-decomposition of $L^{(3)}_{9,9}$.

Proof. Let

$$U = \{u_0, u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\} \text{ and } V = \{v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$$

be two disjoint sets of size 9. Let π be the permutation which maps $u_i \mapsto u_{i+1}$, $v_i \mapsto v_{i+1}$, where addition is performed in $\mathbb{Z}_9$. For each of the following hypergraphs, the union of the orbits of the following blocks under π forms an $H_{3,i}$-decomposition of $L^{(3)}_{U,V}$.

$H_{3,2}$	$[u_0, u_1, v_0, u_6, v_6, u_2, u_4, v_2]_{H_{3,2}}$, $[u_0, u_3, v_3, u_4, v_0, u_1, u_5, v_5]_{H_{3,2}}$, $[u_0, u_1, v_1, u_2, v_2, u_4, u_5, v_3]_{H_{3,2}}$, $[u_0, u_2, v_8, u_5, v_4, u_1, u_4, v_0]_{H_{3,2}}$, $[u_0, u_2, v_1, u_3, v_2, u_1, u_5, v_4]_{H_{3,2}}$, $[u_0, u_1, v_7, u_6, v_4, u_2, u_4, v_0]_{H_{3,2}}$, $[u_0, u_3, v_1, u_4, v_2, u_1, u_5, v_8]_{H_{3,2}}$, $[u_0, u_1, v_6, u_6, v_3, u_2, u_4, v_8]_{H_{3,2}}$, $[u_0, u_1, v_5, u_4, v_1, u_2, u_6, v_8]_{H_{3,2}}$, $[u_0, u_2, v_5, u_5, v_1, u_1, u_4, v_6]_{H_{3,2}}$, $[u_0, u_1, v_4, u_6, v_1, u_2, u_4, v_6]_{H_{3,2}}$, $[u_0, u_1, v_2, u_2, v_3, u_3, u_4, v_6]_{H_{3,2}}$, $[v_0, u_0, v_1, u_6, v_6, u_2, v_2, v_4]_{H_{3,2}}$, $[v_0, u_0, v_4, u_3, v_3, u_1, v_1, v_6]_{H_{3,2}}$, $[u_0, v_0, v_7, v_1, v_2, u_4, v_3, v_4]_{H_{3,2}}$, $[v_0, u_4, v_5, u_5, v_6, u_0, v_1, v_3]_{H_{3,2}}$, $[v_0, u_1, v_2, u_2, v_3, u_0, v_1, v_6]_{H_{3,2}}$, $[v_0, u_4, v_6, u_5, v_7, u_0, v_2, v_3]_{H_{3,2}}$, $[v_0, u_1, v_3, u_2, v_4, u_0, v_2, v_6]_{H_{3,2}}$, $[v_0, u_3, v_6, u_4, v_7, u_0, v_3, v_4]_{H_{3,2}}$, $[u_0, v_3, v_7, v_4, v_5, u_3, v_2, v_6]_{H_{3,2}}$, $[v_0, u_1, v_5, u_5, v_2, u_0, v_4, v_7]_{H_{3,2}}$, $[v_0, u_1, v_6, u_3, v_8, u_0, v_5, v_7]_{H_{3,2}}$, $[v_0, u_1, v_8, u_3, v_2, u_0, v_6, v_7]_{H_{3,2}}$

$H_{3,3}$	$[u_0,v_0,u_1,u_2,u_3,u_6,v_3]_{H_{3,3}}$, $[u_0,u_4,v_0,v_4,u_1,u_7,v_1]_{H_{3,3}}$, $[u_0,u_1,v_1,v_8,u_2,u_4,v_4]_{H_{3,3}}$, $[u_0,v_8,u_2,u_3,u_1,u_5,v_0]_{H_{3,3}}$, $[u_0,v_8,u_5,u_6,u_1,u_3,v_2]_{H_{3,3}}$, $[u_0,v_7,u_1,u_2,u_3,u_6,v_1]_{H_{3,3}}$, $[u_0,u_4,v_2,v_7,u_1,u_7,v_8]_{H_{3,3}}$, $[u_0,v_6,u_1,u_2,u_3,u_6,v_0]_{H_{3,3}}$, $[u_0,u_4,v_1,v_6,u_2,u_3,v_7]_{H_{3,3}}$, $[u_0,v_5,u_2,u_3,u_1,u_5,v_6]_{H_{3,3}}$, $[u_0,v_4,u_1,u_2,u_3,u_6,v_7]_{H_{3,3}}$, $[u_0,u_1,v_2,v_3,u_2,u_4,v_5]_{H_{3,3}}$, $[u_0,v_0,v_1,v_2,u_3,v_3,v_6]_{H_{3,3}}$, $[u_0,v_0,v_4,v_5,u_1,v_1,v_7]_{H_{3,3}}$, $[u_0,v_0,v_7,v_8,u_1,v_2,v_3]_{H_{3,3}}$, $[u_0,v_1,v_3,v_4,u_1,v_2,v_6]_{H_{3,3}}$, $[u_0,v_1,v_6,v_7,u_1,v_0,v_2]_{H_{3,3}}$, $[u_0,v_2,v_3,v_4,u_3,v_5,v_8]_{H_{3,3}}$, $[u_0,v_2,v_6,v_7,u_1,v_0,v_3]_{H_{3,3}}$, $[u_0,v_3,v_4,v_5,u_3,v_0,v_6]_{H_{3,3}}$, $[u_0,v_3,v_7,v_8,u_1,v_5,v_6]_{H_{3,3}}$, $[u_0,v_4,v_6,v_7,u_1,v_0,v_5]_{H_{3,3}}$, $[u_0,v_5,v_6,v_7,u_3,v_2,v_8]_{H_{3,3}}$, $[u_0,v_6,v_7,v_8,u_2,v_0,v_1]_{H_{3,3}}$
$H_{3,4}$	$[u_0,u_1,v_0,u_6,v_6,u_7,v_7]_{H_{3,4}}$, $[u_0,u_3,v_3,u_4,v_4,u_5,v_5]_{H_{3,4}}$, $[u_0,u_1,v_1,u_2,v_2,u_8,v_7]_{H_{3,4}}$, $[u_0,u_2,v_8,u_5,v_4,u_6,v_5]_{H_{3,4}}$, $[u_0,u_2,v_1,u_3,v_2,u_4,v_3]_{H_{3,4}}$, $[u_0,u_1,v_7,u_6,v_4,u_7,v_5]_{H_{3,4}}$, $[u_0,u_3,v_1,u_4,v_2,u_5,v_3]_{H_{3,4}}$, $[u_0,u_1,v_6,u_6,v_3,u_7,v_4]_{H_{3,4}}$, $[u_0,u_1,v_5,u_4,v_1,u_5,v_2]_{H_{3,4}}$, $[u_0,u_2,v_5,u_5,v_1,u_6,v_2]_{H_{3,4}}$, $[u_0,u_1,v_4,u_6,v_1,u_7,v_2]_{H_{3,4}}$, $[u_0,u_2,v_3,u_8,v_2,v_0,v_1]_{H_{3,4}}$, $[v_0,u_0,v_2,u_6,v_6,u_7,u_8]_{H_{3,4}}$, $[v_0,u_0,v_6,u_4,v_4,u_5,v_5]_{H_{3,4}}$, $[v_0,u_0,v_7,u_1,v_1,u_7,v_8]_{H_{3,4}}$, $[v_1,u_0,v_3,u_5,v_6,u_6,v_7]_{H_{3,4}}$, $[v_1,u_0,v_6,u_2,v_3,u_3,v_4]_{H_{3,4}}$, $[v_2,u_0,v_3,u_6,v_8,u_7,v_0]_{H_{3,4}}$, $[v_2,u_0,v_8,u_4,v_6,u_5,v_7]_{H_{3,4}}$, $[v_3,u_0,v_4,u_6,v_0,u_7,v_1]_{H_{3,4}}$, $[v_4,u_0,v_5,u_5,v_8,u_6,v_0]_{H_{3,4}}$, $[v_4,u_0,v_6,u_5,v_0,u_6,v_1]_{H_{3,4}}$, $[v_5,u_0,v_6,u_7,v_3,u_8,v_7]_{H_{3,4}}$, $[v_5,u_0,v_8,u_6,v_4,u_8,v_6]_{H_{3,4}}$
$H_{3,5}$	$[u_0,u_1,v_0,u_6,v_6,u_4,v_4]_{H_{3,5}}$, $[u_0,u_3,v_3,u_4,v_0,u_8,v_8]_{H_{3,5}}$, $[u_0,u_1,v_1,u_7,v_0,u_6,v_5]_{H_{3,5}}$, $[u_0,u_5,v_4,u_2,v_8,u_8,v_7]_{H_{3,5}}$, $[u_0,u_2,v_1,u_3,v_2,u_7,v_6]_{H_{3,5}}$, $[u_0,u_1,v_7,u_6,v_4,u_4,v_2]_{H_{3,5}}$, $[u_0,u_3,v_1,u_4,v_7,u_8,v_6]_{H_{3,5}}$, $[u_0,u_1,v_6,u_6,v_3,u_4,v_1]_{H_{3,5}}$, $[u_0,u_4,v_1,u_1,v_5,u_5,v_7]_{H_{3,5}}$, $[u_0,u_5,v_1,u_2,v_5,u_8,v_4]_{H_{3,5}}$, $[u_0,u_1,v_4,u_6,v_1,u_4,v_8]_{H_{3,5}}$, $[u_0,u_1,v_2,u_2,v_3,u_3,v_5]_{H_{3,5}}$, $[v_0,u_0,v_1,v_6,u_6,u_4,v_4]_{H_{3,5}}$, $[v_0,u_0,v_4,v_3,u_3,u_7,v_7]_{H_{3,5}}$, $[u_0,v_0,v_7,v_2,v_1,u_3,v_3]_{H_{3,5}}$, $[v_0,u_4,v_5,v_3,u_8,u_0,v_1]_{H_{3,5}}$, $[v_0,u_1,v_2,v_6,u_8,u_0,v_1]_{H_{3,5}}$, $[v_0,u_4,v_6,v_2,u_7,u_0,v_3]_{H_{3,5}}$, $[v_0,u_2,v_4,v_6,u_7,u_0,v_2]_{H_{3,5}}$, $[v_0,u_4,v_7,v_3,u_6,u_0,v_4]_{H_{3,5}}$, $[u_0,v_3,v_7,v_4,v_5,u_1,v_0]_{H_{3,5}}$, $[v_0,u_1,v_5,v_6,u_2,u_0,v_4]_{H_{3,5}}$, $[v_0,u_2,v_7,v_6,u_1,u_0,v_5]_{H_{3,5}}$, $[v_0,u_2,v_1,v_7,u_1,u_0,v_6]_{H_{3,5}}$

□

Example 2.A.6. For each $i \in \{2,3,4,5\}$, there exists an $H_{3,i}$-decomposition of $L^{(3)}_{9,9,[1]}$.

Proof. Let

$$U = \{u_0, u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\} \text{ and } V = \{v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$$

be two disjoint sets of size 9, and let ∞ be a point not in $U \cup V$.

Let π be the permutation which maps $u_i \mapsto u_{i+1}$, $v_i \mapsto v_{i+1}$, where addition is performed in $\mathbb{Z}_9$, and $\infty \mapsto \infty$. For each of the following hypergraphs, the union of the orbits of the following blocks under π forms an $H_{3,i}$-decomposition of $L^{(3)}_{U,V,[\{\infty\}]}$.

$H_{3,2}$	$[\infty, u_0, v_0, u_1, v_2, u_3, u_4, v_3]_{H_{3,2}}$, $[\infty, u_0, v_2, u_1, v_4, u_3, u_5, v_3]_{H_{3,2}}$, $[\infty, u_0, v_4, u_1, v_6, u_2, u_5, v_2]_{H_{3,2}}$, $[\infty, u_0, v_6, u_1, v_8, u_2, u_6, v_2]_{H_{3,2}}$, $[u_0, u_3, v_3, u_4, v_4, u_1, v_0, \infty]_{H_{3,2}}$, $[u_0, u_1, v_1, u_2, v_2, u_4, u_5, v_3]_{H_{3,2}}$, $[u_0, u_2, v_8, u_5, v_4, u_1, u_4, v_0]_{H_{3,2}}$, $[u_0, u_2, v_1, u_3, v_2, u_1, u_5, v_4]_{H_{3,2}}$, $[u_0, u_1, v_7, u_6, v_4, u_2, u_4, v_0]_{H_{3,2}}$, $[u_0, u_3, v_1, u_4, v_2, u_1, u_5, v_8]_{H_{3,2}}$, $[u_0, u_1, v_6, u_6, v_3, u_2, u_4, v_8]_{H_{3,2}}$, $[u_0, u_1, v_5, u_4, v_1, u_2, u_6, v_8]_{H_{3,2}}$, $[u_0, u_2, v_5, u_5, v_1, u_1, u_4, v_6]_{H_{3,2}}$, $[u_0, u_1, v_4, u_6, v_1, u_2, u_4, v_6]_{H_{3,2}}$, $[u_0, u_1, v_2, u_2, v_3, u_3, u_4, v_6]_{H_{3,2}}$, $[v_0, u_0, v_1, u_6, v_6, u_2, v_2, v_4]_{H_{3,2}}$, $[v_0, u_0, v_4, u_3, v_3, u_1, v_1, v_6]_{H_{3,2}}$, $[u_0, v_0, v_7, v_1, v_2, u_4, v_3, v_4]_{H_{3,2}}$, $[v_0, u_4, v_5, u_5, v_6, u_0, v_1, v_3]_{H_{3,2}}$, $[v_0, u_1, v_2, u_2, v_3, u_0, v_1, v_6]_{H_{3,2}}$, $[v_0, u_4, v_6, u_5, v_7, u_0, v_2, v_3]_{H_{3,2}}$, $[v_0, u_1, v_3, u_2, v_4, u_0, v_2, v_6]_{H_{3,2}}$, $[v_0, u_3, v_6, u_4, v_7, u_0, v_3, v_4]_{H_{3,2}}$, $[u_0, v_3, v_7, v_4, v_5, u_3, v_2, v_6]_{H_{3,2}}$, $[v_0, u_1, v_5, u_5, v_2, u_0, v_4, v_7]_{H_{3,2}}$, $[v_0, u_1, v_6, u_3, v_8, u_0, v_5, v_7]_{H_{3,2}}$, $[v_0, u_1, v_8, u_3, v_2, u_0, v_6, v_7]_{H_{3,2}}$
$H_{3,3}$	$[u_0, \infty, v_0, v_1, u_2, u_3, v_2]_{H_{3,3}}$, $[u_0, \infty, v_2, v_3, u_1, u_3, v_1]_{H_{3,3}}$, $[u_0, \infty, v_4, v_5, u_1, u_4, v_1]_{H_{3,3}}$, $[u_0, \infty, v_6, v_7, u_1, u_5, v_1]_{H_{3,3}}$, $[u_0, v_0, u_5, u_6, u_2, v_1, \infty]_{H_{3,3}}$, $[u_0, u_1, v_1, v_8, u_2, u_4, v_4]_{H_{3,3}}$, $[u_0, v_8, u_2, u_3, u_1, u_5, v_0]_{H_{3,3}}$, $[u_0, v_8, u_5, u_6, u_1, u_3, v_2]_{H_{3,3}}$, $[u_0, v_7, u_1, u_2, u_3, u_6, v_1]_{H_{3,3}}$, $[u_0, u_4, v_2, v_7, u_1, u_7, v_8]_{H_{3,3}}$, $[u_0, v_6, u_1, u_2, u_3, u_6, v_0]_{H_{3,3}}$, $[u_0, u_4, v_1, v_6, u_2, u_3, v_7]_{H_{3,3}}$, $[u_0, v_5, u_2, u_3, u_1, u_5, v_6]_{H_{3,3}}$, $[u_0, v_4, u_1, u_2, u_3, u_6, v_7]_{H_{3,3}}$, $[u_0, u_1, v_2, v_3, u_2, u_4, v_5]_{H_{3,3}}$, $[u_0, v_0, v_1, v_2, u_3, v_3, v_6]_{H_{3,3}}$, $[u_0, v_0, v_4, v_5, u_1, v_1, v_7]_{H_{3,3}}$, $[u_0, v_0, v_7, v_8, u_1, v_2, v_3]_{H_{3,3}}$, $[u_0, v_1, v_3, v_4, u_1, v_2, v_6]_{H_{3,3}}$, $[u_0, v_1, v_6, v_7, u_1, v_0, v_2]_{H_{3,3}}$, $[u_0, v_2, v_3, v_4, u_3, v_5, v_8]_{H_{3,3}}$, $[u_0, v_2, v_6, v_7, u_1, v_0, v_3]_{H_{3,3}}$, $[u_0, v_3, v_4, v_5, u_3, v_0, v_6]_{H_{3,3}}$, $[u_0, v_3, v_7, v_8, u_1, v_5, v_6]_{H_{3,3}}$, $[u_0, v_4, v_6, v_7, u_1, v_0, v_5]_{H_{3,3}}$, $[u_0, v_5, v_6, v_7, u_3, v_2, v_8]_{H_{3,3}}$, $[u_0, v_6, v_7, v_8, u_2, v_0, v_1]_{H_{3,3}}$

$H_{3,4}$	$[\infty, u_0, v_0, u_1, v_2, u_2, v_4]_{H_{3,4}}$, $[\infty, u_0, v_3, u_1, v_5, u_2, v_7]_{H_{3,4}}$, $[\infty, u_0, v_6, u_1, v_0, u_3, v_1]_{H_{3,4}}$, $[u_0, u_1, v_0, u_6, v_6, u_7, v_7]_{H_{3,4}}$, $[u_0, u_3, v_3, u_4, v_4, u_5, v_5]_{H_{3,4}}$, $[u_0, u_1, v_1, u_2, v_2, u_8, v_7]_{H_{3,4}}$, $[u_0, u_2, v_8, u_5, v_4, u_6, v_5]_{H_{3,4}}$, $[u_0, u_2, v_1, u_3, v_2, u_4, v_3]_{H_{3,4}}$, $[u_0, u_1, v_7, u_6, v_4, u_7, v_5]_{H_{3,4}}$, $[u_0, u_3, v_1, u_4, v_2, u_5, v_3]_{H_{3,4}}$, $[u_0, u_1, v_6, u_6, v_3, u_7, v_4]_{H_{3,4}}$, $[u_0, u_1, v_5, u_4, v_1, u_5, v_2]_{H_{3,4}}$, $[u_0, u_2, v_5, u_5, v_1, u_6, v_2]_{H_{3,4}}$, $[u_0, u_1, v_4, u_6, v_1, u_7, v_2]_{H_{3,4}}$, $[u_0, u_2, v_3, u_8, v_2, v_0, v_1]_{H_{3,4}}$, $[v_0, u_0, v_2, u_6, v_6, u_7, u_8]_{H_{3,4}}$, $[v_0, u_0, v_6, u_4, v_4, u_5, v_5]_{H_{3,4}}$, $[v_0, u_0, v_7, u_1, v_1, u_7, v_8]_{H_{3,4}}$, $[v_1, u_0, v_3, u_5, v_6, u_6, v_7]_{H_{3,4}}$, $[v_1, u_0, v_6, u_2, v_3, u_3, v_4]_{H_{3,4}}$, $[v_2, u_0, v_3, u_6, v_8, u_7, v_0]_{H_{3,4}}$, $[v_2, u_0, v_8, u_4, v_6, u_5, v_7]_{H_{3,4}}$, $[v_3, u_0, v_4, u_6, v_0, u_7, v_1]_{H_{3,4}}$, $[v_4, u_0, v_5, u_5, v_8, u_6, v_0]_{H_{3,4}}$, $[v_4, u_0, v_6, u_5, v_0, u_6, v_1]_{H_{3,4}}$, $[v_5, u_0, v_6, u_7, v_3, u_8, v_7]_{H_{3,4}}$, $[v_5, u_0, v_8, u_6, v_4, u_8, v_6]_{H_{3,4}}$
$H_{3,5}$	$[u_0, u_1, v_0, \infty, v_1, u_2, v_2]_{H_{3,5}}$, $[u_0, u_2, v_0, \infty, v_2, u_1, v_4]_{H_{3,5}}$, $[u_0, u_3, v_0, \infty, v_4, u_1, v_6]_{H_{3,5}}$, $[u_0, u_4, v_0, \infty, v_6, u_1, v_8]_{H_{3,5}}$, $[u_0, u_3, v_3, u_5, v_0, v_4, \infty]_{H_{3,5}}$, $[u_0, u_1, v_1, u_7, v_0, u_6, v_5]_{H_{3,5}}$, $[u_0, u_5, v_4, u_2, v_8, u_8, v_7]_{H_{3,5}}$, $[u_0, u_2, v_1, u_3, v_2, u_7, v_6]_{H_{3,5}}$, $[u_0, u_1, v_7, u_6, v_4, u_4, v_2]_{H_{3,5}}$, $[u_0, u_3, v_1, u_4, v_7, u_8, v_6]_{H_{3,5}}$, $[u_0, u_1, v_6, u_6, v_3, u_4, v_1]_{H_{3,5}}$, $[u_0, u_4, v_1, u_1, v_5, u_5, v_7]_{H_{3,5}}$, $[u_0, u_5, v_1, u_2, v_5, u_8, v_4]_{H_{3,5}}$, $[u_0, u_1, v_4, u_6, v_1, u_4, v_8]_{H_{3,5}}$, $[u_0, u_1, v_2, u_2, v_3, u_3, v_5]_{H_{3,5}}$, $[v_0, u_0, v_1, v_6, u_6, u_4, v_4]_{H_{3,5}}$, $[v_0, u_0, v_4, v_3, u_3, u_7, v_7]_{H_{3,5}}$, $[u_0, v_0, v_7, v_2, v_1, u_3, v_3]_{H_{3,5}}$, $[v_0, u_4, v_5, v_3, u_8, u_0, v_1]_{H_{3,5}}$, $[v_0, u_1, v_2, v_6, u_8, u_0, v_1]_{H_{3,5}}$, $[v_0, u_4, v_6, v_2, u_7, u_0, v_3]_{H_{3,5}}$, $[v_0, u_2, v_4, v_6, u_7, u_0, v_2]_{H_{3,5}}$, $[v_0, u_4, v_7, v_3, u_6, u_0, v_4]_{H_{3,5}}$, $[u_0, v_3, v_7, v_4, v_5, u_1, v_0]_{H_{3,5}}$, $[v_0, u_1, v_5, v_6, u_2, u_0, v_4]_{H_{3,5}}$, $[v_0, u_2, v_7, v_6, u_1, u_0, v_5]_{H_{3,5}}$, $[v_0, u_2, v_1, v_7, u_1, u_0, v_6]_{H_{3,5}}$

□

Example 2.A.7. For each $i \in \{2,3,4,5\}$, there exists an $H_{3,i}$-decomposition of $L^{(3)}_{9,9,[2]}$.

Proof. Let

$$U = \{u_0, u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\} \text{ and } V = \{v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$$

be two disjoint sets of size 9, and let ∞_1, ∞_2 be two points not in $U \cup V$.

Let π be the permutation which maps $u_i \mapsto u_{i+1}$, $v_i \mapsto v_{i+1}$, where addition is performed in $\mathbb{Z}_9$, and $\infty_k \mapsto \infty_k$ for $k \in \{1,2\}$. For each of the following hypergraphs, the union of the orbits of the following blocks under π forms an $H_{3,i}$-decomposition of $L^{(3)}_{U,V,[\{\infty_1,\infty_2\}]}$.

$H_{3,2}$	$[\infty_1, u_0, v_0, u_1, v_2, u_3, v_3, \infty_2]_{H_{3,2}}$, $[\infty_1, u_0, v_2, u_1, v_4, u_2, v_3, \infty_2]_{H_{3,2}}$, $[\infty_1, u_0, v_4, u_1, v_6, u_3, v_5, \infty_2]_{H_{3,2}}$, $[\infty_1, u_0, v_6, u_1, v_8, u_2, v_5, \infty_2]_{H_{3,2}}$, $[\infty_2, u_0, v_4, u_1, v_6, u_2, v_1, \infty_1]_{H_{3,2}}$, $[\infty_2, u_0, v_6, u_1, v_8, u_2, u_3, v_2]_{H_{3,2}}$, $[u_0, u_2, v_0, u_6, v_6, u_3, v_2, \infty_2]_{H_{3,2}}$, $[u_0, u_3, v_3, u_4, v_0, u_1, u_5, v_5]_{H_{3,2}}$, $[u_0, u_1, v_1, u_2, v_2, u_4, u_5, v_3]_{H_{3,2}}$, $[u_0, u_2, v_8, u_5, v_4, u_1, u_4, v_0]_{H_{3,2}}$, $[u_0, u_2, v_1, u_3, v_2, u_1, u_5, v_4]_{H_{3,2}}$, $[u_0, u_1, v_7, u_6, v_4, u_2, u_4, v_0]_{H_{3,2}}$, $[u_0, u_3, v_1, u_4, v_2, u_1, u_5, v_8]_{H_{3,2}}$, $[u_0, u_1, v_6, u_6, v_3, u_2, u_4, v_8]_{H_{3,2}}$, $[u_0, u_1, v_5, u_4, v_1, u_2, u_6, v_8]_{H_{3,2}}$, $[u_0, u_2, v_5, u_5, v_1, u_1, u_4, v_6]_{H_{3,2}}$, $[u_0, u_1, v_4, u_6, v_1, u_2, u_4, v_6]_{H_{3,2}}$, $[u_0, u_1, v_2, u_2, v_3, u_3, u_4, v_6]_{H_{3,2}}$, $[v_0, u_0, v_1, u_6, v_6, u_2, v_2, v_4]_{H_{3,2}}$, $[v_0, u_0, v_4, u_3, v_3, u_1, v_1, v_6]_{H_{3,2}}$, $[u_0, v_0, v_7, v_1, v_2, u_4, v_3, v_4]_{H_{3,2}}$, $[v_0, u_4, v_5, u_5, v_6, u_0, v_1, v_3]_{H_{3,2}}$, $[v_0, u_1, v_2, u_2, v_3, u_0, v_1, v_6]_{H_{3,2}}$, $[v_0, u_4, v_6, u_5, v_7, u_0, v_2, v_3]_{H_{3,2}}$, $[v_0, u_1, v_3, u_2, v_4, u_0, v_2, v_6]_{H_{3,2}}$, $[v_0, u_3, v_6, u_4, v_7, u_0, v_3, v_4]_{H_{3,2}}$, $[u_0, v_3, v_7, v_4, v_5, u_3, v_2, v_6]_{H_{3,2}}$, $[v_0, u_1, v_5, u_5, v_2, u_0, v_4, v_7]_{H_{3,2}}$, $[v_0, u_1, v_6, u_3, v_8, u_0, v_5, v_7]_{H_{3,2}}$, $[v_0, u_1, v_8, u_3, v_2, u_0, v_6, v_7]_{H_{3,2}}$
$H_{3,3}$	$[u_0, \infty_1, v_0, v_1, u_2, v_2, \infty_2]_{H_{3,3}}$, $[u_0, \infty_1, v_2, v_3, u_3, v_4, \infty_2]_{H_{3,3}}$, $[u_0, \infty_1, v_4, v_5, u_1, v_3, \infty_2]_{H_{3,3}}$, $[u_0, \infty_1, v_6, v_7, u_1, v_4, \infty_2]_{H_{3,3}}$, $[u_0, \infty_2, v_4, v_5, u_1, v_0, \infty_1]_{H_{3,3}}$, $[u_0, \infty_2, v_6, v_7, u_1, u_2, v_1]_{H_{3,3}}$, $[u_0, v_0, u_2, u_3, u_4, v_3, \infty_2]_{H_{3,3}}$, $[u_0, u_4, v_0, v_4, u_1, u_7, v_1]_{H_{3,3}}$, $[u_0, u_1, v_1, v_8, u_2, u_4, v_4]_{H_{3,3}}$, $[u_0, v_8, u_2, u_3, u_1, u_5, v_0]_{H_{3,3}}$, $[u_0, v_8, u_5, u_6, u_1, u_3, v_2]_{H_{3,3}}$, $[u_0, v_7, u_1, u_2, u_3, u_6, v_1]_{H_{3,3}}$, $[u_0, u_4, v_2, v_7, u_1, u_7, v_8]_{H_{3,3}}$, $[u_0, v_6, u_1, u_2, u_3, u_6, v_0]_{H_{3,3}}$, $[u_0, u_4, v_1, v_6, u_2, u_3, v_7]_{H_{3,3}}$, $[u_0, v_5, u_2, u_3, u_1, u_5, v_6]_{H_{3,3}}$, $[u_0, v_4, u_1, u_2, u_3, u_6, v_7]_{H_{3,3}}$, $[u_0, u_1, v_2, v_3, u_2, u_4, v_5]_{H_{3,3}}$, $[u_0, v_0, v_1, v_2, u_3, v_3, v_6]_{H_{3,3}}$, $[u_0, v_0, v_4, v_5, u_1, v_1, v_7]_{H_{3,3}}$, $[u_0, v_0, v_7, v_8, u_1, v_2, v_3]_{H_{3,3}}$, $[u_0, v_1, v_3, v_4, u_1, v_2, v_6]_{H_{3,3}}$, $[u_0, v_1, v_6, v_7, u_1, v_0, v_2]_{H_{3,3}}$, $[u_0, v_2, v_3, v_4, u_3, v_5, v_8]_{H_{3,3}}$, $[u_0, v_2, v_6, v_7, u_1, v_0, v_3]_{H_{3,3}}$, $[u_0, v_3, v_4, v_5, u_3, v_0, v_6]_{H_{3,3}}$, $[u_0, v_3, v_7, v_8, u_1, v_5, v_6]_{H_{3,3}}$, $[u_0, v_4, v_6, v_7, u_1, v_0, v_5]_{H_{3,3}}$, $[u_0, v_5, v_6, v_7, u_3, v_2, v_8]_{H_{3,3}}$, $[u_0, v_6, v_7, v_8, u_2, v_0, v_1]_{H_{3,3}}$

$H_{3,4}$	$[\infty_1, u_0, v_0, u_1, v_2, u_2, v_4]_{H_{3,4}}$, $[\infty_1, u_0, v_3, u_1, v_5, u_2, v_7]_{H_{3,4}}$, $[\infty_1, u_0, v_6, u_1, v_0, u_3, v_1]_{H_{3,4}}$, $[\infty_2, u_0, v_0, u_1, v_2, u_2, v_4]_{H_{3,4}}$, $[\infty_2, u_0, v_3, u_1, v_5, u_2, v_7]_{H_{3,4}}$, $[\infty_2, u_0, v_6, u_1, v_0, u_3, v_1]_{H_{3,4}}$, $[u_0, u_1, v_0, u_6, v_6, u_7, v_7]_{H_{3,4}}$, $[u_0, u_3, v_3, u_4, v_4, u_5, v_5]_{H_{3,4}}$, $[u_0, u_1, v_1, u_2, v_2, u_8, v_7]_{H_{3,4}}$, $[u_0, u_2, v_8, u_5, v_4, u_6, v_5]_{H_{3,4}}$, $[u_0, u_2, v_1, u_3, v_2, u_4, v_3]_{H_{3,4}}$, $[u_0, u_1, v_7, u_6, v_4, u_7, v_5]_{H_{3,4}}$, $[u_0, u_3, v_1, u_4, v_2, u_5, v_3]_{H_{3,4}}$, $[u_0, u_1, v_6, u_6, v_3, u_7, v_4]_{H_{3,4}}$, $[u_0, u_1, v_5, u_4, v_1, u_5, v_2]_{H_{3,4}}$, $[u_0, u_2, v_5, u_5, v_1, u_6, v_2]_{H_{3,4}}$, $[u_0, u_1, v_4, u_6, v_1, u_7, v_2]_{H_{3,4}}$, $[u_0, u_2, v_3, u_8, v_2, v_0, v_1]_{H_{3,4}}$, $[v_0, u_0, v_2, u_6, v_6, u_7, u_8]_{H_{3,4}}$, $[v_0, u_0, v_6, u_4, v_4, u_5, v_5]_{H_{3,4}}$, $[v_0, u_0, v_7, u_1, v_1, u_7, v_8]_{H_{3,4}}$, $[v_1, u_0, v_3, u_5, v_6, u_6, v_7]_{H_{3,4}}$, $[v_1, u_0, v_6, u_2, v_3, u_3, v_4]_{H_{3,4}}$, $[v_2, u_0, v_3, u_6, v_8, u_7, v_0]_{H_{3,4}}$, $[v_2, u_0, v_8, u_4, v_6, u_5, v_7]_{H_{3,4}}$, $[v_3, u_0, v_4, u_6, v_0, u_7, v_1]_{H_{3,4}}$, $[v_4, u_0, v_5, u_5, v_8, u_6, v_0]_{H_{3,4}}$, $[v_4, u_0, v_6, u_5, v_0, u_6, v_1]_{H_{3,4}}$, $[v_5, u_0, v_6, u_7, v_3, u_8, v_7]_{H_{3,4}}$, $[v_5, u_0, v_8, u_6, v_4, u_8, v_6]_{H_{3,4}}$
$H_{3,5}$	$[u_0, v_0, \infty_2, \infty_1, v_1, u_2, v_2]_{H_{3,5}}$, $[u_0, v_1, \infty_2, \infty_1, v_2, u_1, v_4]_{H_{3,5}}$, $[u_0, v_2, \infty_2, \infty_1, v_4, u_1, v_6]_{H_{3,5}}$, $[u_0, v_3, \infty_2, \infty_1, v_6, u_1, v_8]_{H_{3,5}}$, $[u_0, v_8, \infty_1, \infty_2, v_4, u_1, v_6]_{H_{3,5}}$, $[u_0, u_1, v_0, \infty_2, v_6, u_3, v_1]_{H_{3,5}}$, $[u_0, u_2, v_0, u_6, v_6, v_5, \infty_2]_{H_{3,5}}$, $[u_0, u_3, v_3, u_4, v_0, u_8, v_8]_{H_{3,5}}$, $[u_0, u_1, v_1, u_7, v_0, u_6, v_5]_{H_{3,5}}$, $[u_0, u_5, v_4, u_2, v_8, u_8, v_7]_{H_{3,5}}$, $[u_0, u_2, v_1, u_3, v_2, u_7, v_6]_{H_{3,5}}$, $[u_0, u_1, v_7, u_6, v_4, u_4, v_2]_{H_{3,5}}$, $[u_0, u_3, v_1, u_4, v_7, u_8, v_6]_{H_{3,5}}$, $[u_0, u_1, v_6, u_6, v_3, u_4, v_1]_{H_{3,5}}$, $[u_0, u_4, v_1, u_1, v_5, u_5, v_7]_{H_{3,5}}$, $[u_0, u_5, v_1, u_2, v_5, u_8, v_4]_{H_{3,5}}$, $[u_0, u_1, v_4, u_6, v_1, u_4, v_8]_{H_{3,5}}$, $[u_0, u_1, v_2, u_2, v_3, u_3, v_5]_{H_{3,5}}$, $[v_0, u_0, v_1, v_6, u_6, u_4, v_4]_{H_{3,5}}$, $[v_0, u_0, v_4, v_3, u_3, u_7, v_7]_{H_{3,5}}$, $[u_0, v_0, v_7, v_2, v_1, u_3, v_3]_{H_{3,5}}$, $[v_0, u_4, v_5, v_3, u_8, u_0, v_1]_{H_{3,5}}$, $[v_0, u_1, v_2, v_6, u_8, u_0, v_1]_{H_{3,5}}$, $[v_0, u_4, v_6, v_2, u_7, u_0, v_3]_{H_{3,5}}$, $[v_0, u_2, v_4, v_6, u_7, u_0, v_2]_{H_{3,5}}$, $[v_0, u_4, v_7, v_3, u_6, u_0, v_4]_{H_{3,5}}$, $[u_0, v_3, v_7, v_4, v_5, u_1, v_0]_{H_{3,5}}$, $[v_0, u_1, v_5, v_6, u_2, u_0, v_4]_{H_{3,5}}$, $[v_0, u_2, v_7, v_6, u_1, u_0, v_5]_{H_{3,5}}$, $[v_0, u_2, v_1, v_7, u_1, u_0, v_6]_{H_{3,5}}$

□

Example 2.A.8. There exists an $H_{4,3}$-design of order 6.

Proof. Let $V = \{0, 1, 2, 3, 4, \infty\}$. The orbit of $[\infty, 0, 2, 1, 3, 4]_{H_{4,3}}$ under the permutation $(0\ 1\ 2\ 3\ 4)(\infty)$ gives an $H_{4,3}$-decomposition of $K_V^{(3)}$. □

Example 2.A.9. There exists an $H_{4,3}$-design of order 9.

Proof. Let $V = \{0, 1, 2, 3, 4, 5, 6, \infty_1, \infty_2\}$. The orbit of the three H-blocks

$$B_1 = [0, 1, 2, 6, 3, 4]_{H_{4,3}} \quad B_2 = [\infty_1, 0, 1, \infty_2, 2, 5]_{H_{4,3}}$$
$$B_3 = [\infty_1, 0, \infty_2, 2, 4, 6]_{H_{4,3}}$$

under the permutation $(0\ 1\ 2\ 3\ 4\ 5\ 6)(\infty_1)(\infty_2)$ gives an $H_{4,3}$-decomposition of on $K_V^{(3)}$:

The orbit of B_1 gives all edges of the form $\{i, i+1, i+2\}$, $\{i, i+1, i+3\}$, $\{i, i+2, i+3\}$, and $\{i, i+1, i+4\}$, for each $i \in \mathbb{Z}_6$.

The orbit of B_2 gives all edges of the form $\{\infty_k, i, i+1\}$ and $\{\infty_k, i, i+3\}$, for each $k \in \{1, 2\}$ and $i \in \mathbb{Z}_6$.

The orbit of B_3 gives all edges of the form $\{\infty_k, i, i+2\}$, $\{\infty_1, \infty_2, i\}$, and $\{i, i+2, i+4\}$, for each $k \in \{1, 2\}$ and $i \in \mathbb{Z}_6$. $\square$

Example 2.A.10. There exists an $H_{4,3}$-decomposition of $K_{14}^{(3)} \setminus K_6^{(3)}$.

Proof. Let $V = \{v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$ and $U = \{u_0, u_1, u_2, u_3, u_4, u_5\}$ be two disjoint sets of size 8, 6 respectively. Consider first, the orbits of the $H_{4,3}$-blocks

$$B_1 = [v_0, v_2, v_7, v_1, v_4, v_6]_{H_{4,3}} \quad B_2 = [v_0, v_3, v_4, u_0, v_1, u_3]_{H_{4,3}}$$
$$B_3 = [v_0, v_4, v_5, u_1, v_1, u_4]_{H_{4,3}} \quad B_4 = [v_0, v_1, v_2, u_2, v_7, u_5]_{H_{4,3}}$$

under the permutation $(v_0\, v_1\, v_2\, v_3\, v_4\, v_5\, v_6\, v_7)$. This gives all edges of the form $\{v_i, v_j, v_k\}$, $\{v_i, v_{i+1}, u_j\}$ and $\{v_i, u_j, u_{j+3}\}$. In particular, for all $i \in \mathbb{Z}_8$, we can see that the orbit of B_1 contains all edges of the form $\{v_i, v_{i+1}, v_{i+3}\}$, $\{v_i, v_{i+2}, v_{i+3}\}$, $\{v_i, v_{i+2}, v_{i+4}\}$, and $\{v_i, v_{i+2}, v_{i+5}\}$; B_2 contains all edges of the form $\{v_i, v_{i+3}, v_{i+4}\}$; B_3 contains all edges of the form $\{v_i, v_{i+1}, v_{i+4}\}$; and B_4 contains all edges of the form $\{v_i, v_{i+1}, v_{i+2}\}$.

Next, take the orbit of $[u_0, v_0, v_2, u_1, v_1, v_4]_{H_{4,3}}$ under the group generated by the permutations

$$G_1 = \langle (v_0\, v_1\, v_2\, v_3\, v_4\, v_5\, v_6\, v_7),\ (u_0\, u_2\, u_4)(u_1\, u_3\, u_5) \rangle;$$

this gives all edges of the form $\{v_i, v_{i+2}, u_j\}$ and $\{v_i, v_{i+3}, u_j\}$. Next, take the orbit of $[u_0, v_0, v_4, u_1, v_1, v_5]_{H_{4,3}}$ under the group

$$G_2 = \langle (v_0\, v_2)(v_1\, v_3)(v_4\, v_6)(v_5\, v_7),\ (u_0\, u_2\, u_4)(u_1\, u_3\, u_5) \rangle;$$

this gives all edges of the form $\{v_i, v_{i+4}, u_j\}$. Finally, take the orbit of $[v_0, u_0, u_1, v_1, u_2, u_4]_{H_{4,3}}$ under the group

$$G_3 = \langle (v_0\, v_2\, v_4\, v_6)(v_1\, v_3\, v_5\, v_7),\ (u_0\, u_1\, u_2\, u_3\, u_4\, u_5) \rangle;$$

This gives all edges of the form $\{v_i, u_j, u_{j+1}\}$ and $\{v_i, u_j, u_{j+2}\}$. The union of all of these orbits forms an $H_{4,3}$-decomposition of $K_{U \cup V}^{(3)} \setminus K_U^{(3)}$. $\square$

Example 2.A.11. There exists an $H_{4,3}$-decomposition of $L_{8,8,[6]}^{(3)}$.

Proof. Let U, V, W be three pairwise disjoint sets of size 8, 8, and 6 respectively, and denote their elements by $U = \{u_0, u_1, u_2, u_3, u_4, u_5, u_6, u_7\}$, $V = \{v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$, and $W = \{\infty_1, \infty_2, \infty_3, \infty_4, \infty_5, \infty_6\}$.

First, consider the orbits of the seven $H_{4,3}$-blocks

$$\begin{array}{ll} B_{1,1} = [v_0, u_0, u_1, v_1, u_4, u_5]_{H_{4,3}} & B_{1,2} = [v_0, u_1, u_2, v_1, u_5, u_6]_{H_{4,3}} \\ B_{2,1} = [v_0, u_0, u_2, v_1, u_4, u_6]_{H_{4,3}} & B_{2,2} = [v_0, u_1, u_3, v_1, u_5, u_7]_{H_{4,3}} \\ B_{3,1} = [v_0, u_0, u_3, v_1, u_4, u_7]_{H_{4,3}} & B_{3,2} = [v_0, u_1, u_4, v_1, u_5, u_0]_{H_{4,3}} \\ B_{4,1} = [v_0, u_0, u_4, v_1, u_1, u_5]_{H_{4,3}} & \end{array}$$

under the permutation group

$$\begin{aligned} G_1 = \langle &(v_0\, v_2\, v_4\, v_6)(v_1\, v_3\, v_5\, v_7),\ (u_0\, u_2\, u_4\, u_6)(u_1\, u_3\, u_5\, u_7), \\ &(u_0\, v_0)(u_1\, v_1)(u_2\, v_2)(u_3\, v_3)(u_4\, v_4)(u_5\, v_5)(u_6\, v_6)(u_7\, v_7)\rangle. \end{aligned}$$

Note that each of these orbits is a short orbit, with a stabiliser of order 2. The orbits of the blocks $B_{\alpha,\beta}$ give all edges of the form $\{v_i, u_j, u_{j+\alpha}\}$ and $\{v_i, v_{i+\alpha}, u_j\}$.

Next, consider the orbit of $[\infty_1, v_0, u_0, \infty_2, v_1, u_2]_{H_{4,3}}$ under the group

$$\begin{aligned} G_2 = \langle &(v_0\, v_1\, v_2\, v_3\, v_4\, v_5\, v_6\, v_7)(u_0\, u_1\, u_2\, u_3\, u_4\, u_5\, u_6\, u_7), \\ &(u_0\, u_2\, u_4\, u_6)(u_1\, u_3\, u_5\, u_7), \\ &(\infty_1\, \infty_3\, \infty_5)(\infty_2\, \infty_4\, \infty_6)\rangle \end{aligned}$$

This orbit covers all edges of the form $\{\infty_i, v_j, u_k\}$. Hence, the union of these orbits give an $H_{4,3}$-decomposition of $L^{(3)}_{U,V,[W]}$. □

Example 2.A.12. There exists an $H_{4,3}$-decomposition of $L^{(3)}_{8,8,[1]}$.

Proof. Let U and V be two disjoint sets of size 8, and denote their elements by $U = \{u_0, u_1, u_2, u_3, u_4, u_5, u_6, u_7\}$ and $V = \{v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$. Let ∞ be a point not in $U \cup V$. Consider first the orbits of the five $H_{4,3}$-blocks

$$\begin{array}{ll} B_{2,1} = [v_0, u_0, u_2, v_1, u_4, u_6]_{H_{4,3}} & B_{2,2} = [v_0, u_1, u_3, v_1, u_5, u_7]_{H_{4,3}} \\ B_{3,1} = [v_0, u_0, u_3, v_1, u_4, u_7]_{H_{4,3}} & B_{3,2} = [v_0, u_1, u_4, v_1, u_5, u_0]_{H_{4,3}} \\ B_{4,1} = [v_0, u_0, u_4, v_4, u_1, u_5]_{H_{4,3}} & \end{array}$$

under the permutation group

$$\begin{aligned} G_1 = \langle &(v_0\, v_2\, v_4\, v_6)(v_1\, v_3\, v_5\, v_7),\ (u_0\, u_2\, u_4\, u_6)(u_1\, u_3\, u_5\, u_7), \\ &(u_0\, v_0)(u_1\, v_1)(u_2\, v_2)(u_3\, v_3)(u_4\, v_4)(u_5\, v_5)(u_6\, v_6)(u_7\, v_7)\rangle; \end{aligned}$$

as with Example 2.A.11, these five blocks each have a stabiliser of order 2, and the orbits give all edges of the form $\{v_i, u_j, u_{j+\alpha}\}$ and $\{v_i, v_{i+\alpha}, u_j\}$ for $\alpha \in \{2, 3, 4\}$.

Next, take the orbit of $[v_0, u_0, u_1, v_1, u_4, u_5]_{H_{4,3}}$ under the group

$$G_2 = \langle (u_0\, u_1\, u_2\, u_3\, u_4\, u_5\, u_6\, u_7),\ (v_0\, v_2\, v_4\, v_6)(v_1\, v_3\, v_5\, v_7)\rangle;$$

this gives all edges of the form $\{v_i, u_j, u_{j+1}\}$; note that this is a short orbit with a stabiliser of order 2.

Finally, take the orbit of $[u_0, v_0, \infty, u_1, v_4, v_5]_{H_{4,3}}$ under the group

$$G_3 = \langle (v_0\, v_1\, v_2\, v_3\, v_4\, v_5\, v_6\, v_7),\ (u_0\, u_2\, u_4\, u_6)(u_1\, u_3\, u_5\, u_7)\rangle;$$

this gives all edges of the form $\{v_i, v_{i+1}, u_j\}$ and $\{v_i, u_j, \infty\}$. The union of these orbits therefore gives an $H_{4,3}$-decomposition of $L^{(3)}_{U,V,[\{\infty\}]}$. □

Chapter 3

Generating Hypergraph Decompositions

3.1 Construction Algorithms

In this chapter, we discuss the methods used to construct the hypergraph decompositions given in Section 2.A, and present computer programs which were used to generate many of those examples.

Theorems 2.3.4, 2.3.5, and 2.3.8 give a construction of H-designs of order v for certain 3-uniform hypergraphs H and all admissible v. The general construction method can be summarised as follows:

Theorem 3.1.1. *Let $m, n_0 \geq 1$ and $\epsilon \geq 0$ be integers, and let H be a 3-uniform hypergraph. Suppose that there exist H-decompositions of the following hypergraphs:*

(1) $K^{(3)}_{mn_0+\epsilon}$,

(2) if $m(n_0-1)+\epsilon \geq 1$, $K^{(3)}_{mn_0+\epsilon} \setminus K^{(3)}_{m(n_0-1)+\epsilon}$,[1]

(3) $L^{(3)}_{m,m,[\epsilon]}$, *and*

(4) $K^{(3)}_{m,m,m}$.

Then there exists an H-design of order $v = mn + \epsilon$ for all $n \geq n_0$.

For example:

- In Theorem 2.3.4, we consider $m = 4$, $\epsilon \in \{0, 1, 2\}$, and $n_0 = 1$, except when $H \simeq H_{2,2}$ and $\epsilon = 0$, in which case we take $n_0 = 2$.

[1] If $1 \leq m(n_0-1)+\epsilon < 3$, then $K^{(3)}_{m(n_0-1)+\epsilon}$ has no edges, so this is equal to $K^{(3)}_{mn_0+\epsilon}$.

- In Theorem 2.3.5, we consider $m = 9$, $\epsilon \in \{0, 1, 2\}$, and $n_0 = 1$.
- In Theorem 2.3.8, we consider $m = 8$, $\epsilon \in \{1, 6\}$, and $n_0 = 1$.

Proof. If $n = n_0$, the result holds by assumption, so consider $n > n_0$. Let $V_1, V_2, \ldots, V_n$ be n pairwise disjoint sets of size m, and let I be a set of size ϵ, disjoint from each of $V_1, V_2, \ldots, V_n$. Let $K = K^{(3)}_{V_1 \cup V_2 \cup \cdots \cup V_n \cup I} \simeq K^{(3)}_{mn+\epsilon}$, and let Γ be the set of hypergraphs listed in the statement of the theorem. We show there exists a Γ-decomposition of K:

If $n_0 = 1$ and $\epsilon = 0$, then $m(n_0 - 1) + \epsilon = 0$, and

$$K = \left(\bigcup_{1 \le i \le n} K^{(3)}_{V_i} \right) \cup \left(\bigcup_{1 \le i < j \le n} L^{(3)}_{V_i, V_j} \right) \cup \left(\bigcup_{1 \le i < j < k \le n} K^{(3)}_{V_i, V_j, V_k} \right)$$

describes a Γ-decomposition of K. Otherwise, if $n = n_0 + 1$, let $W = I$, or if $n > n_0 + 1$, let $W = V_{n-n_0+2} \cup \cdots \cup V_n \cup I$. Then $|W| = m(n_0 - 1) + \epsilon \ge 1$, so $W \neq \emptyset$. Therefore,

$$\begin{aligned} K = K^{(3)}_{V_{n-n_0+1} \cup W} &\cup \left(\bigcup_{1 \le i \le n-n_0} K^{(3)}_{V_i \cup W} \setminus K^{(3)}_W \right) \\ &\cup \left(\bigcup_{1 \le i < j \le n-n_0+1} L^{(3)}_{V_i, V_j, [I]} \right) \cup \left(\bigcup_{1 \le i < j < k \le n} K^{(3)}_{V_i, V_j, V_k} \right) \end{aligned}$$

describes a Γ-decomposition of K.

If $\mathcal{D}'$ is a Γ-decomposition of K, let $\mathcal{D}_G$ be an H-decomposition of G for each $G \in \mathcal{D}'$. Then $\bigcup_{G \in \mathcal{D}'} \mathcal{D}_G$ is an H-decomposition of K. $\square$

Note that this method cannot always be applied (for example, see the discussion of Theorem 2.3.5 in the case $H \simeq H_{3,11}$).

With Theorem 3.1.1 as motivation, in particular the case that H has three edges and at least seven vertices, we shall present methods of finding H-decompositions of $K^{(3)}_v$, $L^{(3)}_{m,m,[\epsilon]}$, and $K^{(3)}_{m,m,m}$, for given v, w. These methods were used to construct Examples 2.A.1 through 2.A.7, which are used to prove Theorem 2.3.5.

Note that we omit consideration of methods to find H-decompositions of $K^{(3)}_v \setminus K^{(3)}_w$ for $w < v$, since these are not required to prove Theorem 2.3.5. In principle, these methods could be generalised to t-uniform hypergraphs with $t > 3$, and similarly Theorem 3.1.1 could be generalised. However, this would be a cumbersome task in practice, and would require significant computation time.

Let K be isomorphic to one of $K_v^{(3)}$, $L_{m,m,[\epsilon]}^{(3)}$, or $K_{m,m,m}^{(3)}$. In constructing H-decompositions of K, we first fix some automorphism π on K; that is, fix a permutation π acting on $V(K)$ such that for all $e \in E(K)$, $\pi(e) \in E(K)$. We then aim to find a set X of H-blocks such that the union of orbits of X under π,

$$\mathcal{D} = \bigcup_{H' \in X} \mathcal{O}_\pi(H'), \tag{3.1.1}$$

is an H-decomposition of K, where $\mathcal{O}_\pi(H')$ denotes the orbit of H' under π. Moreover, we shall require that π acts freely on X (that is, all stabilisers are trivial). We shall call X a set of *base blocks* of $\mathcal{D}$ for π.

Note the following immediate result:

Lemma 3.1.2. *Suppose $\mathcal{D}$ is an H-decomposition of K satisfying the conditions above, and let $|\pi|$ denote the order of the permutation π. Then $|E(K)| = |E(H)| \cdot |X| \cdot |\pi|$.*

Proof. Since π acts freely on X, the number of H-blocks in $\mathcal{D}$ is $|X| \cdot |\pi|$. The edge sets of these H-blocks partition $E(K)$, so $|E(K)| = |E(H)| \cdot |X| \cdot |\pi|$. $\square$

Hence, when we fix π, we must ensure that $|\pi|$ divides $|E(K)|/|E(H)|$, and then we will know in advance the size of the set X which we are required to find.

It should be noted that such a method is somewhat restrictive, in particular, there may exist H-decompositions $\mathcal{D}$ of a hypergraph K such that π is not an automorphism of $\mathcal{D}$, in which case the method will be incapable of finding $\mathcal{D}$. Moreover, it is possible that a decomposition can be expressed more succinctly or elegantly by giving the orbits of a smaller number of H-blocks under a larger permutation group, as with Examples 2.A.10, 2.A.11, and 2.A.12.

Each of the algorithms presented in this chapter have been implemented in Appendix A.

3.2 Decompositions of $K_v^{(3)}$

Let H be a 3-uniform hypergraph, and let $v \geq 3$ be an admissible parameter for the existence of an H-design of order v. We would like to fix a permutation π, and then aim to find an H-decomposition $\mathcal{D}$ of $K_v^{(3)}$ given by (3.1.1) (if one exists).

We shall choose π to be a cyclic permutation (possibly containing fixed points). The following result restricts π to having at most two fixed points:

Lemma 3.2.1. *Suppose $\mathcal{D}$ is an H-decomposition of $K_v^{(3)}$ given by* (3.1.1), *where π is a non-trivial cyclic permutation. Then $v-2 \leq |\pi| \leq v$.*

Proof. Since π acts on $|V(K_v^{(3)})| = v$ elements, $|\pi| \leq v$. To show that $|\pi| \geq v-2$, it suffices to show that π has at most two fixed points.

Suppose for contradiction that π has at least three fixed points, and let $\infty_1, \infty_2, \infty_3$ be distinct fixed points of π. Let H_0 be the unique H-block of $\mathcal{D}$ which contains the edge $e = \{\infty_1, \infty_2, \infty_3\}$. Since $\pi(e) = e$, e is an edge of $\pi(H_0)$. But H_0 is unique, so $\pi(H_0) = H_0$, and since π acts freely on X, π is the identity permutation. Since π was assumed to be non-trivial, we have a contradiction, so π has at most two fixed points. □

Given H and v, let $n = |\pi| \in \{v-2, v-1, v\}$ such that n divides $\binom{v}{3}/|E(H)|$. WLOG, we can label the vertices of $K_v^{(3)}$ so that

$$V(K_v^{(3)}) = \{0, 1, 2, \ldots, n-1\} \cup I, \tag{3.2.1}$$

where $I = \emptyset$ if $n = v$, $I = \{\infty_1\}$ if $n = v-1$, and $I = \{\infty_1, \infty_2\}$ if $n = v-2$. Then, let

$$\pi = \begin{cases} (0\ 1\ 2\ \cdots\ n-1) & \text{if } n = v \\ (0\ 1\ 2\ \cdots\ n-1)(\infty_1) & \text{if } n = v-1 \\ (0\ 1\ 2\ \cdots\ n-1)(\infty_1)(\infty_2) & \text{if } n = v-2 \end{cases} \tag{3.2.2}$$

So it only remains to find a suitable set X of base blocks, where $|X| = \binom{v}{3}/(|E(H)| \cdot n)$.

Definition 3.2.2. Let H and K be 3-uniform hypergraphs, and let π be an automorphism of K. Consider the orbit of each edge of K under π.

For a set B containing $|E(H)|$ of these orbits, say that B is *admissible* if there exists a subhypergraph H' of K such that $H' \simeq H$ and $E(H')$ is a system of distinct representatives of B. Equivalently, B is admissible if there exists a hypergraph H' isomorphic to H such that $B = \{\mathcal{O}_\pi(e) \mid e \in E(H')\}$.

If B is an admissible set of edge orbits, then any hypergraph H' satisfying the above conditions is said to be a hypergraph *corresponding to B*. Note that H' is not necessarily unique. □

Hence, if we can partition $E(K_v^{(3)})$ into parts of size $|E(H)|$ such that every part is admissible, then such a partition will give rise to a set X such that $\mathcal{D}$ given by (3.1.1) is an H-decomposition of $K_v^{(3)}$. In particular, X can be described as a set of hypergraphs corresponding to each part in the partition.

It then remains to describe the orbits of edges in $K_v^{(3)}$; we shall introduce the following definitions:

Definition 3.2.3. Given a positive integer $n \geq 3$, let A be the set of ordered triples (a, b, c), where $1 \leq a, b, c \leq \left\lfloor \frac{n}{2} \right\rfloor$, that are solutions to either

$$a + b = c \quad \text{or} \quad a + b + c = n.$$

(Note that these conditions will both hold if n is even and $c = \frac{n}{2}$.)

Define an equivalence relation $\sim$ on A where $(a, b, c) \sim (x, y, z)$ if and only if $(a, b, c) = (x, y, z)$, or $(a, b, c) = (y, z, x)$, or $(a, b, c) = (z, x, y)$. That is, $\sim$ is equivalence up to a cyclic rearrangement of (a, b, c).

A *difference triple* modulo n is defined to be an equivalence class of A under $\sim$, and we shall use the notation (a, b, c) to refer to the equivalence class containing (a, b, c). □

Definition 3.2.4. Let π be given by (3.2.2), and consider the orbit of each edge of $K_v^{(3)}$ under π. We define *types* of orbits as follows:

- The orbit containing all edges of the form $\{\infty_1, \infty_2, i\}$ for $i \in \mathbb{Z}_n$ is of type (∞_1, ∞_2);
- For each $1 \leq d \leq \left\lfloor \frac{n}{2} \right\rfloor$ and $k \in \{1, 2\}$, the orbit containing all edges of the form $\{\infty_k, i, i + d\}$ for $i \in \mathbb{Z}_n$ is of type (∞_k, d);
- For each difference triple (a, b, c) modulo n, the orbit containing all edges of the form $\{i, i + a, i + a + b\}$ for $i \in \mathbb{Z}_n$ is of type (a, b, c).

□

It is clear that the orbit of every edge of $K_v^{(3)}$ falls into precisely one of the types of Definition 3.2.4. In order to generate difference triples modulo n, we use the following lemma:

Lemma 3.2.5. *Let $1 \leq a, b \leq \left\lfloor \frac{n}{2} \right\rfloor$ be integers. There exists a unique (up to equivalence) difference triple of the form (a, b, c), where $a + b = c$ if and only if $a + b \leq \left\lfloor \frac{n}{2} \right\rfloor$, and $a + b + c = n$ if and only if $a + b \geq \left\lceil \frac{n}{2} \right\rceil$. Moreover, (b, c, a) and (c, a, b) are difference triples equivalent to (a, b, c) if and only if $a + b + c = n$.*

Proof. If (a, b, c) is a difference triple, recall that $1 \leq a, b, c \leq \left\lfloor \frac{n}{2} \right\rfloor$.

Let $1 \leq a, b \leq \left\lfloor \frac{n}{2} \right\rfloor$ be integers. It is then immediate that $(a, b, a + b)$ is a difference triple if and only if $a + b \leq \left\lfloor \frac{n}{2} \right\rfloor$. Let $c = n - a - b$, then (a, b, c) is a difference triple with $a + b + c = n$ if and only if $c = n - a - b \leq \left\lfloor \frac{n}{2} \right\rfloor$, which holds if and only if $a + b \geq n - \left\lfloor \frac{n}{2} \right\rfloor = \left\lceil \frac{n}{2} \right\rceil$.

If $a + b + c = n$, then clearly (b, c, a) and (c, a, b) are difference triples equivalent to (a, b, c). Conversely, if $a + b + c \neq n$, then $a + b = c$, and since $1 \leq a, b$, $b + c = a + 2b \neq a$ and $c + a = 2a + b \neq b$, therefore (b, c, a) and (c, a, b) are not difference triples. □

Given a 3-uniform hypergraph H, and $v \geq 3$, we can now describe a method of generating an H-decomposition of $K_v^{(3)}$ of the form (3.1.1).

First, given v and $n = |\pi| \in \{v-2, v-1, v\}$, Algorithm 3.1 below computes the set $\mathcal{O}$ of all orbit types of π acting on $K_v^{(3)}$, as given in Definition 3.2.4. Lemma 3.2.5 ensures that we have indeed generated all orbit types.

Next, recall that in order to generate the set of base blocks X, we require a mapping from admissible sets of orbit types B to their corresponding hypergraphs H'. Algorithm 3.3 below constructs a function c such that for every admissible set B, $c(B)$ is a hypergraph corresponding to B.

To achieve this, we begin by letting c be the 'empty function', i.e. the function with domain $\mathrm{Dom}(c) = \emptyset$. We consider every subhypergraph H' of $K_v^{(3)}$ isomorphic to H,[2] and find the set B consisting of the orbit types of edges in H', using Algorithm 3.2. If each edge of H' has a different orbit, then B is admissible; and if B is not in the domain of c, then we add B to the domain and set $c(B) = H'$.

Finally, Algorithm 3.4 describes a method for finding an H-decomposition of $K_v^{(3)}$ of the form (3.1.1), if one exists. It does so by employing a recursive procedure `find_partition`, which takes a set of orbit types $\mathcal{O}$, the function c above, and $e = |E(H)|$.

Note that we will reuse Algorithms 3.3 and 3.4 in Sections 3.3 and 3.4 to find decompositions of $L^{(3)}_{m,m,[\epsilon]}$ and $K^{(3)}_{m,m,m}$ respectively, so the algorithms also accept a parameter K, the hypergraph we wish to find a decomposition of.

[2]To find a subhypergraph of $K_v^{(3)}$ isomorphic to H, fix an ordering of $V(H)$, then let an V be an ordered selection of $|V(H)|$ elements from $V(K_v^{(3)})$. Since $V(H)$ and V are ordered sets of equal size, there is a natural map $\varphi : V(H) \to V$, then $\varphi(H)$ is a subhypergraph of $K_v^{(3)}$. Repeating this for all ordered selections $V \subseteq V(K_v^{(3)})$ of size $|V(H)|$ gives all subhypergraphs of $K_v^{(3)}$.

```
Input: K_v^(3) with vertex set given by (3.2.1), where v ≥ 3,
       n ∈ {v − 2, v − 1, v}
Output: The set of all orbit types for a cyclic permutation of length
        n on K_v^(3), as given by Definition 3.2.4
O ← ∅;
for a = 1 to ⌊n/2⌋ do
    for b = 1 to ⌊n/2⌋ do
        if a + b < ⌈n/2⌉ then
            /* We have a + b + c ≠ n.                              */
            c ← a + b;
            O ← O ∪ {(a, b, c)};
        else
            /* We have a + b + c = n.                              */
            c ← n − (a + b);
            /* We have (a, b, c) ∼ (b, c, a) ∼ (c, a, b).  WLOG, take
               the lexicographically smallest one.                 */
            if (a, b, c) ≤ (b, c, a) and (a, b, c) ≤ (c, a, b) then
                O ← O ∪ {(a, b, c)};
            end
        end
    end
end
if n ≤ v − 1 then
    for d = 1 to ⌊n/2⌋ do
        for k = 1 to v − n do
            O ← O ∪ {(∞_k, d)};
        end
    end
end
if n = v − 2 then
    O ← O ∪ {(∞_1, ∞_2)};
end
return O;
```

Algorithm 3.1: Find orbit types of $K_v^{(3)}$ under a cyclic permutation.

```
Input: A subhypergraph H' of K_v^(3), where V(K_v^(3)) is given by (3.2.1)
Output: A set B consisting of the types of orbits of edges in H'
B ← ∅;
for e ∈ E(H') do
    if e = {u, v, w}: u, v, w ∈ Z_n, u < v < w then
        t ← (a, b, c) such that e = {x, x + a, x + a + b} for some x ∈ Z_n;
        B ← B ∪ {t};
    else if e = {∞_k, v, w}: k ∈ {1, 2}, v, w ∈ Z_n, v < w then
        d ← min{w − v, n − (w − v)};
        B ← B ∪ {(∞_k, d)} ;
    else if e = {∞_1, ∞_2, w}: w ∈ Z_n then
        B ← B ∪ {(∞_1, ∞_2)};
    end
end
return B;
```

Algorithm 3.2: Find the set of orbit types described by a subhypergraph H' of $K_v^{(3)}$.

```
Input: 3-uniform hypergraphs H and K, where K is isomorphic to
       one of K_v^(3), L_{m,m,[ε]}^(3), or K_{m,m,m}^(3)
Output: A function c which maps admissible sets of orbits to one of
        their corresponding hypergraphs
c ← empty mapping;
for each ordered selection V of |V(H)| elements from V(K) do
    H' ← copy(H, V) ;          /* Isomorphic copy of H on V */
    if H' is a subhypergraph of K then
        B ← get_orbits(H') ;    /* Algorithms 3.2, 3.6, 3.8 */
        if |B| = |E(H)| and B ∉ Dom(c) then
            Define c(B) = H';
        end
    end
end
return c;
```

Algorithm 3.3: Find a mapping between admissible sets of orbits and corresponding H-blocks.

Input: 3-uniform hypergraphs H and K, where K is isomorphic to one of $K_v^{(3)}$, $L_{m,m,[\epsilon]}^{(3)}$, or $K_{m,m,m}^{(3)}$
Output: The order n of the cyclic permutation π, and a set X of base blocks

```
𝒪 ← get_types(K) ;                  /* Algorithms 3.1, 3.5, 3.7 */
c ← get_admissible(H, K) ;                      /* Algorithm 3.3 */
X ← find_partition(𝒪, c, e|E(H)|) ;                /* See below */
return n, X
Procedure find_partition(𝒪, c, e)
    /* This recursive procedure will attempt to find a
       partition of 𝒪 into admissible parts.                  */
    if 𝒪 = ∅ then
        return ∅;
    end
    for each B ⊆ 𝒪, |B| = e do
        if B ∈ Dom(c) then
            X ← find_partition(𝒪 \ B, c, e) ;
            if X ≠ "No Solution" then
                X ← X ∪ {c(B)} ;
                return X;
            end
        end
    end
    return "No Solution";
```

Algorithm 3.4: Find H-decompositions of a hypergraph K

3.3 Decompositions of $L^{(3)}_{m,m,[\epsilon]}$

Let H be a 3-uniform hypergraph, and let $m \geq 1$, $\epsilon \geq 0$ be integers such that $L^{(3)}_{m,m,[\epsilon]}$ satisfies the conditions of Lemma 2.2.2 for the existence of an H-decomposition of $L^{(3)}_{m,m,[\epsilon]}$. We would like to fix an automorphism π of $L^{(3)}_{m,m,[\epsilon]}$, and then find an H-decomposition of $L^{(3)}_{m,m,[\epsilon]}$ of the form (3.1.1), if one exists.

WLOG, we can identify $L^{(3)}_{m,m,[\epsilon]}$ with $L^{(3)}_{U,V,[W]}$ where U, V, and W are three pairwise disjoint sets given by

$$\begin{aligned}U &= \{u_0, u_1, \ldots, u_{m-1}\},\\ V &= \{v_0, v_1, \ldots, v_{m-1}\},\\ W &= \{\infty_1, \infty_2, \ldots, \infty_\epsilon\},\end{aligned}$$

where the verticesa in U and V are indexed by $\mathbb{Z}_m$, and W is possibly empty. Then,

$$\begin{aligned}E(L^{(3)}_{m,m,[\epsilon]}) = &\big\{\{u_i, u_j, v_k\}, \{u_i, v_j, v_k\} : i, j, k \in \mathbb{Z}_m\big\} \\ &\cup \big\{\{u_i, u_j, \infty_k\} : i, j \in \mathbb{Z}_m; 1 \leq k < \epsilon\big\}.\end{aligned} \tag{3.3.1}$$

Let π to be the automorphism which maps $u_i \mapsto u_{i+1}$ for each $u_i \in U$, $v_i \mapsto v_{i+1}$ for each $v_i \in V$ (with addition being performed in $\mathbb{Z}_m$), and $\infty_k \mapsto \infty_k$ for each $\infty_k \in W$. We shall use π to find an H-decomposition of $L^{(3)}_{m,m,[\epsilon]}$ the form (3.1.1).

It remains to classify the orbits of edges in $L^{(3)}_{m,m,[\epsilon]}$ under π (cf. Definition 3.2.4), and present methods analagous to Algorithms 3.1 and 3.2. Then, Algorithms 3.3 and 3.4 can be applied to find H-decompositions of $L^{(3)}_{m,m,[\epsilon]}$.

Definition 3.3.1. Let π be the automorphism of $L^{(3)}_{m,m,[\epsilon]}$ given above, and consider the orbit of each edge of $L^{(3)}_{m,m,[\epsilon]}$ under π. We define *types* of orbits as follows:

- For every pair of integers $a, b \in \mathbb{Z}_m$ with $a < b$, the orbit containing all edges of the form $\{v_i, u_{i+a}, u_{i+b}\}$ for $i \in \mathbb{Z}_m$ is of type $(\mathtt{u}, a, b)$, where $\mathtt{u}$ is an arbitrary symbol. The orbit containing all edges of the form $\{u_i, v_{i+a}, v_{i+b}\}$ for $i \in \mathbb{Z}_m$ is of type $(\mathtt{v}, a, b)$, where $\mathtt{v}$ is an arbitrary symbol.[3]

[3] Note that we distinguish the symbols $\mathtt{u}$ and $\mathtt{v}$ from u and v, which are used to denote elements of U and V.

- If $\epsilon > 0$, then for every integer $d \in \mathbb{Z}_m$ and every integer $1 \leq k \leq \epsilon$, the orbit containing every edge of the form $\{u_i, v_{i+d}, \infty_k\}$ for $i \in \mathbb{Z}_m$ is of type (∞_k, d).

$\square$

It is clear that every edge of $L^{(3)}_{m,m,[\epsilon]}$, given by (3.3.1), falls into precisely one of the orbit types given in Definition 3.3.1.

For integers m, ϵ, Algorithm 3.5 below will generate the set $\mathcal{O}$ of all orbit types of $L^{(3)}_{m,m,[\epsilon]}$, similar to Algorithm 3.1. Given a subhypergraph H' of $L^{(3)}_{m,m,[\epsilon]}$, Algorithm 3.6 below will generate the set B consisting of orbit types for each edge in H' under the permutation π, similar to Algorithm 3.2.

With these two procedures defined, Algorithm 3.4 can be used to construct an H-decomposition of $L^{(3)}_{m,m,[\epsilon]}$, if one exists.

Input: $L^{(3)}_{m,m,[\epsilon]}$ with edge set given by (3.3.1)
Output: The set of all orbit types of $L^{(3)}_{m,m,[\epsilon]}$, as given by Definition 3.3.1

```
𝒪 ← ∅;
for a = 0 to m − 2 do
    for b = a + 1 to m − 1 do
        𝒪 ← 𝒪 ∪ {(u, a, b)};
        𝒪 ← 𝒪 ∪ {(v, a, b)};
    end
end
if ε ≥ 1 then
    for d = 0 to m − 1 do
        for k = 1 to ε do
            𝒪 ← 𝒪 ∪ {(∞_k, d)};
        end
    end
end
return 𝒪;
```

Algorithm 3.5: Find orbit types of $L^{(3)}_{m,m,[\epsilon]}$.

Input: A subhypergraph H' of $L^{(3)}_{m,m,[\epsilon]}$, where $E(L^{(3)}_{m,m,[\epsilon]})$ is given by (3.3.1)
Output: A set B consisting of the types of orbits of edges in H'
$B \leftarrow \emptyset$;
for $e \in E(H')$ **do**
  if $e = \{u_i, u_j, v_k\}$: $i, j, k \in \mathbb{Z}_m$ **then**
    $a \leftarrow \min\{i-k, j-k\}$ (with subtraction performed in $\mathbb{Z}_m$);
    $b \leftarrow \max\{i-k, j-k\}$ (subtraction in $\mathbb{Z}_m$);
    $B \leftarrow B \cup \{(\mathtt{u}, a, b)\}$;
  else if $e = \{u_i, v_j, v_k\}$: $i, j, k \in \mathbb{Z}_m$ **then**
    $a \leftarrow \min\{j-i, k-i\}$ (subtraction in $\mathbb{Z}_m$);
    $b \leftarrow \max\{j-i, k-i\}$ (subtraction in $\mathbb{Z}_m$);
    $B \leftarrow B \cup \{(\mathtt{u}, a, b)\}$;
  else if $e = \{u_i, v_j, \infty_k\}$: $i, j \in \mathbb{Z}_m$, $k \in \{1, 2, \ldots, \epsilon\}$ **then**
    $d \leftarrow j - i \pmod{v}$;
    $B \leftarrow B \cup \{(\infty_k, d)\}$;
  end
end
return B;

Algorithm 3.6: Find the set of orbit types described by a subhypergraph H' of $L^{(3)}_{m,m,[\epsilon]}$.

3.4 Decompositions of $K^{(3)}_{m,m,m}$

Let H be a 3-uniform hypergraph, and let $m \geq 1$ be an integer such that $K^{(3)}_{m,m,m}$ satisfies the conditions of Lemma 2.2.2 for the existence of an H-decomposition of $K^{(3)}_{m,m,m}$. We would like to fix an automorphism π of $K^{(3)}_{m,m,m}$, and then find an H-decomposition of $K^{(3)}_{m,m,m}$ of the form (3.1.1).

WLOG, we can identify $K^{(3)}_{m,m,m}$ with $K^{(3)}_{U,V,W}$ where U, V, and W are pairwise disjoint sets given by

$$\begin{aligned} U &= \{u_0, u_1, \ldots, u_{m-1}\}, \\ V &= \{v_0, v_1, \ldots, v_{m-1}\}, \\ W &= \{w_0, w_1, \ldots, w_{m-1}\}, \end{aligned}$$

where the vertices in each of U, V, W are indexed by $\mathbb{Z}_m$. Then,

$$E(K^{(3)}_{m,m,m}) = \big\{\{u_i, v_j, w_k\} : i, j, k \in \mathbb{Z}_m\big\}. \tag{3.4.1}$$

Let π be the automorphism of $K^{(3)}_{m,m,m}$ which maps $u_i \mapsto u_{i+1}$, $v_i \mapsto v_{i+1}$, and $w_i \mapsto w_{i+1}$ for each $u_i \in U$, $v_i \in V$, $w_i \in W$ (where addition is performed in $\mathbb{Z}_m$). We shall use π to find an H-decomposition of $K^{(3)}_{m,m,m}$ the form (3.1.1).

It remains to classify the orbits of edges in $K^{(3)}_{m,m,m}$ under π (cf. Definition 3.2.4), and present methods analagous to Algorithms 3.1 and 3.2.

Definition 3.4.1. Let π be the automorphism of $K^{(3)}_{m,m,m}$ defined above, and consider the orbit of each edge of $K^{(3)}_{m,m,m}$ under π.

For every pair of integers $a, b \in \mathbb{Z}_m$, we say that the orbit containing edges of the form $\{u_i, v_{i+a}, w_{i+b}\}$ for $i \in \mathbb{Z}_m$ is of *type* (a, b). □

It is clear that Definition 3.4.1 classifies every edge of $K^{(3)}_{m,m,m}$ into precisely one type.

For an integer m, Algorithm 3.7 below will generate the set $\mathcal{O}$ of all orbit types of $K^{(3)}_{m,m,m}$, similar to Algorithm 3.1. Given a subhypergraph H' of $K^{(3)}_{m,m,m}$, Algorithm 3.8 below will generate the set B consisting of orbit types for each edge in H' under the permutation π, similar to Algorithm 3.2.

With these two procedures defined, Algorithm 3.4 can be used to construct an H-decomposition of $K^{(3)}_{m,m,m}$, if one exists.

Input: $K^{(3)}_{m,m,m}$ with edge set given by (3.4.1)
Output: The set of all orbit types of $K^{(3)}_{m,m,m}$, as given by Definition 3.4.1

```
𝒪 ← ∅;
for a = 0 to m − 1 do
    for b = 0 to m − 1 do
        𝒪 ← 𝒪 ∪ {(a, b)};
    end
end
return 𝒪;
```

Algorithm 3.7: Find orbit types of $K^{(3)}_{m,m,m}$.

Input: A subhypergraph H' of $K^{(3)}_{m,m,m}$, where $V(K^{(3)}_{m,m,m})$ is given by (3.4.1)
Output: A set B consisting of the types of orbits of edges in H'

```
B ← ∅;
for e = {(u_i, v_j, w_k} ∈ E(H') do
    a ← j − i (mod m);
    b ← k − i (mod m);
    B ← B ∪ {(a, b)};
end
return B;
```

Algorithm 3.8: Find the set of orbit types described by a subhypergraph H' of $K^{(3)}_{m,m,m}$.

3.5 Checking Decompositions

Many of the example hypergraph decompositions given in Chapter 2 would be tedious and time-consuming to verify by hand, so we present here an algorithm which takes as input a hypergraph K and a family $\mathcal{D}$ of subgraphs of K, and determines whether or not $\mathcal{D}$ is a decomposition of K.

Most of the decompositions listed in Section 2.A are specified by giving a set X of blocks, and a permutation group G such that the union of orbits of blocks under G,

$$\bigcup_{H \in X} \mathcal{O}_G(H), \tag{3.5.1}$$

gives the decomposition. We say that X is a set of *base blocks* of the decomposition, with corresponding permutation group G. The permutation group G is specified by giving a set S of generators, so that $G = \langle S \rangle$.

Recall that Examples 2.A.10, 2.A.11, and 2.A.12 specify the decomposition by giving multiple sets of base blocks $X_1, X_2, \ldots, X_n$, each with a corresponding permutation group $G_1 = \langle S_1 \rangle, G_2 = \langle S_2 \rangle, \ldots, G_n = \langle S_n \rangle$. The decomposition is then given by

$$\mathcal{D} = \bigcup_{i=1}^{n} \bigcup_{H \in X_i} \mathcal{O}_{\langle S_i \rangle}(H). \tag{3.5.2}$$

Note that (3.5.1) is simply a special case of this, in particular with $n = 1$. Given the sets of base blocks and permutation groups, we would like to determine if $\mathcal{D}$ is a decomposition of a given hypergraph K.

Let K be a hypergraph. Let $X_1, X_2, \ldots, X_n$ be sets of subhypergraphs of K, and let $\mathcal{X} = \{X_1, X_2, \ldots, X_n\}$. Let $S_1, S_2, \ldots, S_n$ be sets of automorphisms of K, and let $\mathcal{S} = \{S_1, S_2, \ldots, S_n\}$. Given K, $\mathcal{X}$, and $\mathcal{S}$, Algorithm 3.9 below computes $\mathcal{D}$ given by (3.5.2), and determines if $\mathcal{D}$ is a decomposition of K.

Input: A hypergraph K; collections $\mathcal{X} = \{X_i\}$ and $\mathcal{S} = \{S_i\}$
Output: $True$ if $\mathcal{D}$ given by (3.5.2) is a decomposition of K, otherwise $False$.

```
D ← ∅;
for i = 1 to n do
    D_i ← X_i;
    /* Note that as more elements get added to D_i, the loop
       'for H ∈ D_i' below will then iterate over those
       elements also.  This causes products of permutations
       in S_i to be included in D_i, therefore D_i will consist
       of O_⟨S_i⟩(H) for each H ∈ X_i.                        */
    for H ∈ D_i do
        for π ∈ S_i do
            D_i ← D_i ∪ {π(H)};
        end
    end
    D ← D ∪ D_i;
end
E = ∅;
for H ∈ D do
    E ← E ∪ E(H);
end
if E = E(K) then
    return True;
else
    return False;
```

Algorithm 3.9: Check if a union of orbits of hypergraphs gives a decomposition of a given hypergraph K

Appendix A

Implementation of Methods

The attached computer program gives an implementation of the methods described in Chapter 3. These programs are implemented in the Python 3 programming language, which is available at `http://python.org`.

A soft copy of this implementation is attached, and is also available from the author's webpage `http://jgat.github.io/honours/`. For posterity, a hard copy is included here.

A.1 Shared Resources

The following file, `hypergraphs.py`, defines data structures and methods shared across multiple programs.

```
"""A model for 3-uniform Hypergraphs"""
import itertools as _itertools


class Hypergraph(object):
    def __init__(self, vertices, edges):
        self._vertices = list(vertices)
        self._edges = [sorted(e) for e in edges]
        self._edge_set = set(tuple(e) for e in self._edges)

    vertices = property(lambda self: list(self._vertices))
    edges = property(lambda self: [list(e) for e in self._edges])

    @classmethod
    def from_edges(self, edges):
        vertices = []
        for e in edges:
            for v in e:
                if v not in vertices:
```

```
                vertices.append(v)
        return Hypergraph(vertices, edges)

    def copy(self, vertices):
        """Return an isomorphic copy of `self` on the given vertices."""
        # Construct the isomorphism, then apply it to each edge.
        isom = dict(zip(self._vertices, vertices))
        edges = [[isom[_] for _ in e] for e in self._edges]
        return Hypergraph(vertices, edges)

    def copy_edges(self, new_edges):
        """Return a copy of the graph onto the new edge set"""
        new_vertices = list(set(sum(new_edges, [])))
        new_edges = list(map(sorted, new_edges))
        # Try and find the appropriate vertex mapping
        for f in _itertools.permutations(new_vertices):
            isom = dict(zip(self._vertices, f))
            edges = [sorted(map(isom.get, e)) for e in self._edges]
            if edges == new_edges:
                return Hypergraph(f, edges)
        raise ValueError("Edge sets do not correspond")

    def __str__(self):
        return "(V = {}, E = {})".format(self.vertices, self.edges)

    def __repr__(self):
        return "Hypergraph({!r}, {!r})".format(self.vertices,
                                               self.edges)

    def __eq__(self, other):
        return (sorted(self._vertices) == sorted(other._vertices) and
                sorted(self._edges) == sorted(other._edges))

    def regular(self):
        """Return True if all vertices have the same degree."""
        degs = {len([e for e in self._edges if x in e])
                for x in self._vertices}
        return len(degs) == 1

    def isomorphic(self, other):
        """Return True if the two hypergraphs are isomorphic."""
        # This is not at all a smart way of doing it.
        if (len(self._vertices) != len(other._vertices) or
            len(self._edges) != len(other._edges)):
            return False
        for f in _itertools.permutations(other.vertices):
            # Does f describe an isomorphism onto `self`?
            if self.copy(f) == other:
                return True
```

```
        return False

    def contains(self, H):
        """Return True if H is a sub-hypergraph of this graph"""
        return all(tuple(e) in self._edge_set for e in H._edges)

    def subgraphs(self, H, copy=None):
        """Yield subgraphs isomorphic to H"""
        length = len(H._edges)
        if copy is None:
            copy = [None for i in range(length)]

        def agree(e, f):
            """Return True if there is a maping  e -> f  such that
            for each edge e' in E(H) and corresponding f' in `copy`,
            the mapping extends to a mapping e+e' -> f+f'."""
            for f_order in _itertools.permutations(f):
                e_to_f = {a: b for a, b in zip(e, f_order)}
                f_to_e = {b: a for a, b in zip(e, f_order)}
                for e2, f2 in zip(H._edges, copy):
                    if f2 is None:
                        return True
                    if any(v in e and e_to_f[v] not in f2 for v in e2):
                        break
                    if any(v in f and f_to_e[v] not in e2 for v in f2):
                        break
                else:
                    return True
            return False

        # Try and construct a copy of H out of edges of `self`:
        i = copy.index(None)
        for e in self._edges:
            # Check whether e can be used as edge number i of H:
            if agree(H._edges[i], e):
                copy[i] = e
                if i == length - 1:
                    yield H.copy_edges(copy)
                else:
                    yield from self.subgraphs(H, copy)
            copy[i] = None

    def __sub__(self, other, keep_isolated=False):
        assert all(e in self._edges for e in other._edges)
        edges = [e for e in self._edges if e not in other._edges]
        vertices = [v for v in self._vertices
                    if keep_isolated or any(v in e for e in edges)]
        return Hypergraph(vertices, edges)
```

```
    def __add__(self, other):
        vertices = (self._vertices +
                    [v for v in other._vertices
                     if v not in self._vertices])
        return Hypergraph(vertices, self._edges + other._edges)

    def __mul__(self, factor):
        return Hypergraph(self._vertices, self._edges * factor)

    def __rmul__(self, factor):
        return Hypergraph(self._vertices, self._edges * factor)


def complete_hypergraph(vertices):
    """Construct a complete 3-uniform hypergraph."""
    return Hypergraph(vertices,
                      _itertools.combinations(vertices, 3))


def multipartite(parts):
    """Construct a complete 3-uniform 3-partite hypergraph."""
    vertices = sum(parts, [])
    assert len(vertices) == len(set(vertices))
    edges = _itertools.product(*parts)
    return Hypergraph(vertices, edges)


def candelabra(parts, stem=None):
    """Construct a 3-uniform hypergraph L_{V1,...,Vm,[W]}."""
    if stem is None:
        stem = []

    vertices = sum(parts, []) + stem
    assert len(vertices) == len(set(vertices))
    H = complete_hypergraph(vertices)
    rest = Hypergraph([],[])
    for p in parts:
        rest += complete_hypergraph(p + stem)
    return H - rest


class FixedPoint(object):
    "An element which is not moved by cycles"
    def __init__(self, num):
        self.num = num

    def __str__(self):
        return "\\infty_{{{}}}".format(self.num)
```

```
    def __unicode__(self):
        return u"\u221e_{{{}}}".format(self.num)

    def __repr__(self):
        return "INF({})".format(self.num)

    def __eq__(self, other):
        return isinstance(other, FixedPoint) and self.num == other.num

    def __lt__(self, other):
        return (not isinstance(other, FixedPoint)
                or self.num < other.num)

    def __le__(self, other):
        return (not isinstance(other, FixedPoint)
                or self.num <= other.num)

    def __gt__(self, other):
        return not (self <= other)

    def __ge__(self, other):
        return not (self < other)

    def __hash__(self):
        return hash(self.num)
```

A.2 Generating Decompositions

The following file, `decompositions.py`, implements methods defined in Chapter 3 for generating decompositions of certain families of hypergraphs.

```
#!/usr/bin/env python3

import itertools
import functools
import sys

from hypergraphs import *


#################
# Algorithm 3.1 #
#################

def get_complete_types(v, n):
    """Return the set of all orbit types of the complete 3-uniform
    hypergraph."""
```

```
    if not v - 2 <= n <= v:
        raise ValueError("Can't have more than two fixed points")

    orbits = set()

    for a in range(1, n // 2 + 1):
        for b in range(1, n // 2 + 1):
            if a + b < (n + 1) // 2:
                # (a, b, c) s.t. a + b = c
                c = a + b
                orbits.add((a, b, c))
            elif a + b < n:
                # (a, b, c) s.t. a + b + c = n
                # (b, c, a) and (c, a, b) are also triples;
                # take the lexicographically smallest.
                c = n - (a + b)
                if (a, b, c) <= (b, c, a) and (a, b, c) <= (c, a, b):
                    orbits.add((a, b, c))

    # Do some sanity checking on the number of triples
    if n % 3 == 0:
        assert len(orbits) == n * (n - 3) / 6 + 1, n
    else:
        assert len(orbits) == (n - 1) * (n - 2) / 6, n

    if n <= v - 1:
        for j in range(1, (n + 1)//2):
            for i in range(v-n):
                orbits.add((FixedPoint(i + 1), j))
    if n == v - 2:
        # Add the pair (infty1, infty2)
        orbits.add((FixedPoint(1), FixedPoint(2)))

    # Check we have the correct number of orbit types
    assert len(orbits) == v * (v - 1) * (v - 2) // 6 / n

    return sorted(orbits)


#################
# Algorithm 3.2 #
#################

def get_complete_orbits(block, v, n):
    """For a given H-block, compute the orbit
    types which its edges include."""
    covered = []

    for e in block.edges:
```

```
        u, v, w = sorted(e)
        if isinstance(u, int):      # No fixed points
            # Get the distances and normalise
            a, b, c = v-u, w-v, n-(w-u)
            if a > n // 2: a = n - a
            if b > n // 2: b = n - b
            if c > n // 2: c = n - c

            if a + b + c == n:
                edge_type = min((a,b,c), (b,c,a), (c,a,b))
            elif a + b == c:
                edge_type = (a, b, c)
            elif b + c == a:
                edge_type = (b, c, a)
            elif c + a == b:
                edge_type = (c, a, b)
            else:
                raise RuntimeError("Unreachable")

        elif isinstance(v, int):    # One fixed point
            diff = abs(v-w)
            diff = min(diff, n-diff)
            edge_type = (u, diff)

        else:   # Two fixed points
            edge_type = (u, v)

        covered.append(edge_type)

    return covered


#################
# Algorithm 3.3 #
#################

def get_admissible(H, K, get_orbits, *args):
    """For each admissible set B, find a corresponding H-block.

    Return a mapping {admissible sets} -> {corresponding H-blocks}
    """
    coverings = {}

    for V in itertools.permutations(K.vertices, len(H.vertices)):
        # A small optimisation
        if get_orbits != get_complete_orbits and ('u', 0) not in V:
            continue

        block = H.copy(V)
```

```
        if K.contains(block):
            B = get_orbits(block, *args)
            B = tuple(sorted(set(B)))
            if B not in coverings and len(B) == len(H.edges):
                coverings[B] = block

    return coverings


#################
# Algorithm 3.4 #
#################

def find_decomposition(H, K, get_types, get_orbits, *args):
    """Given hypergraphs H and K, find an H-decomposition of K,
    if one exists."""
    orbits = tuple(get_types(*args))
    possible_blocks = get_admissible(H, K, get_orbits, *args)

    @functools.lru_cache(maxsize=None)
    def find_partition(orbits, depth=0):
        """Solve the subproblem of finding only the given edge orbits"""
        if not orbits:
            return []
        for block in itertools.combinations(orbits, len(H.edges)):
            if tuple(sorted(block)) in possible_blocks:
                print("| "*depth+"Try", block)
                remain = tuple(x for x in orbits if x not in block)
                rest = find_partition(remain, depth+1)
                if rest is not None:
                    return [(block, possible_blocks[block])] + rest
        return None

    return find_partition(orbits)


#################
# Algorithm 3.5 #
#################

def get_candelabra_types(m, epsilon):
    """Get the set of orbit types of L_{m,m,[epsilon]}"""
    orbits = set()
    for a, b in itertools.combinations(range(m), 2):
        orbits.add(('u', a, b))
        orbits.add(('v', a, b))

    for d in range(m):
```

```
        for k in range(epsilon):
            orbits.add((FixedPoint(k+1), d))

    return sorted(orbits)


#################
# Algorithm 3.6 #
#################

def get_candelabra_orbits(block, m, epsilon):
    """For a given H-block, find the orbit types it covers."""
    covered = []

    for e in block.edges:
        x, y, z = sorted(e)
        if isinstance(x, FixedPoint):
            assert y[0] == 'u' and z[0] == 'v'
            dist = (z[1] - y[1]) % m
            edge_type = (x, dist)
        else:
            assert x[0] == 'u' and z[0] == 'v'
            if y[0] == 'u':
                # z is by itself
                d1, d2 = sorted([(_[1] - z[1]) % m for _ in [x, y]])
            elif y[0] == 'v':
                # x is by itself
                d1, d2 = sorted([(_[1] - x[1]) % m for _ in [y, z]])
            edge_type = (y[0], d1, d2)

        covered.append(edge_type)

    return covered


#################
# Algorithm 3.7 #
#################

def get_gdd_types(m):
    """Get the set of orbit types of K_{m,m,m}"""
    orbits = set()
    for a in range(m):
        for b in range(m):
            orbits.add((a, b))

    return sorted(orbits)
```

```
#################
# Algorithm 3.7 #
#################

def get_gdd_orbits(block, m):
    """For a given H-block, find the orbit types it covers."""
    covered = []

    for e in block.edges:
        x, y, z = e
        assert x[0] == 'u' and y[0] == 'v' and z[0] == 'w'
        edge_type = ((y[1] - x[1]) % m, (z[1] - x[1]) % m)

        covered.append(edge_type)

    return covered


######################################################################
# Helper functions to run the above methods

def find_complete_decomposition(H, v):
    """Find a complete decomposition of a hypergraph H of order v"""
    # First, find the parameter `n`, the length of the orbit
    num_edges = v * (v - 1) * (v - 2) // 6
    if num_edges % len(H.edges) != 0:
        raise ValueError("v={}, |E(H)|={} does not satisfy "
                         "obvious necessary conditions"
                         .format(v, len(H.edges)))
    num_blocks = num_edges // len(H.edges)

    n = max(i for i in range(v-2, v+1) if num_blocks % i == 0)

    K = complete_hypergraph(list(range(n)) +
                            [FixedPoint(i+1) for i in range(v-n)])

    return find_decomposition(H, K, get_complete_types,
                              get_complete_orbits, v, n)


def find_candelabra_decomposition(H, m, epsilon):
    parts = [[(c, i) for i in range(m)] for c in 'uv']
    stem = [FixedPoint(i+1) for i in range(epsilon)]

    K = candelabra(parts, stem)
    return find_decomposition(H, K, get_candelabra_types,
                              get_candelabra_orbits, m, epsilon)
```

```
def find_gdd_decomposition(H, m):
    parts = [[(c, i) for i in range(m)] for c in 'uvw']
    K = multipartite(parts)
    return find_decomposition(H, K, get_gdd_types, get_gdd_orbits, m)


######################################################################


def output(result):
    print()
    if result is None:
        print("No Solution\n")
        return

    for orbits, block in result:
        print("{!r:<50}  covers orbit types {}"
              .format(block.vertices, orbits))
    print()
    print()


H32 = Hypergraph([1,2,3,4,5,6,7,8], [[1,2,3],[1,4,5],[6,7,8]])
H33 = Hypergraph([1,2,3,4,5,6,7], [[1,2,3],[1,2,4],[5,6,7]])
H34 = Hypergraph([1,2,3,4,5,6,7], [[1,2,3],[1,4,5],[1,6,7]])
H35 = Hypergraph([1,2,3,4,5,6,7], [[1,2,3],[1,4,5],[4,6,7]])

H39 = Hypergraph([1,2,3,4,5], [[1,2,3],[1,2,4],[1,2,5]])


def main():
    # Find complete decompositions:
    for v in [9, 10, 11]:
        print("H39-decomposition of K_{{{}}}^3".format(v))
        output(find_complete_decomposition(H39, v))

    # Find candelabra systems
    for epsilon in [0, 1, 2]:
        print("H39-decomposition of L_{{9,9,[{}]}}^3".format(epsilon))
        output(find_candelabra_decomposition(H39, 9, epsilon))

    # Find group divisible designs
    print("H39-decomposition of K_{9,9,9}^3")
    output(find_gdd_decomposition(H39, 9))


if __name__ == '__main__':
    main()
```

A.3 Checking Decompositions

The following file, `check.py`, implements Algorithm 3.9. The attached file `check-cases.py`[1] uses this implementation to verify that the decompositions generated are indeed correct (both computer-generated and hand-generated decompositions are checked).

```
from hypergraphs import *
from itertools import product, repeat
INF = FixedPoint

def permutation(*cycles):
    """Generate a permutation with given cycles"""
    perm = {}
    for points in cycles:
        for i in range(len(points)):
            perm[points[i-1]] = points[i]
    return perm


def check_decomposition(K, *args):
    """Given a set of base blocks and permutations on the vertex sets,
    check that the orbits give a decomposition of the hypergraph K."""
    all_blocks = []
    for base_blocks, permutations in zip(args[::2], args[1::2]):
        this_orbit = list(base_blocks)
        for block in this_orbit:
            for p in permutations:
                new = block.copy([p.get(v,v) for v in block.vertices])
                if new not in this_orbit:
                    this_orbit.append(new)
        all_blocks.extend(this_orbit)

    edges = []
    for block in all_blocks:
        edges.extend(block.edges)

    # Sort all the edges and all the vertices within each edge.
    edges = sorted(map(sorted, edges))
    required = sorted(map(sorted, K.edges))

    if edges == required:
        print("OK")
        return True
```

[1] A hard-copy of `check-cases.py` is not included due to size constraints.

```
    # There was something wrong with the decomposition.
    # Tell the user what was wrong.

    # Check if the right number of edges were covered.
    if len(edges) != len(required):
        print("ERROR - expected {} edges, got {}"
               .format(len(required), len(edges)))
        print()
    else:
        print("ERROR:")
        print()

    # Check if there were edges missed
    print("Edges not covered:")
    missed = {tuple(e) for e in required
              if edges.count(e) < required.count(e)}
    total = 0
    for e in sorted(map(list, missed)):
        diff = required.count(e) - edges.count(e)
        if diff == 1:
            print(e)
        else:
            print(e, '*', diff)
        total += diff
    print("Total:", total, end="\n\n")

    # Check if there were any edges gained or duplicated
    print("Unwanted edges:")
    missed = {tuple(e) for e in edges
              if required.count(e) < edges.count(e)}
    total = 0
    for e in sorted(map(list, missed)):
        diff = edges.count(e) - required.count(e)
        if diff == 1:
            print(e)
        else:
            print(e, '*', diff)
        total += diff
    print("Total:", total, end="\n\n")

    return False
```

Bibliography

[1] R. F. Bailey and B. Stevens, Hamiltonian decompositions of complete k-uniform hypergraphs. *Discrete Math.* 310 (2010), 3088-3095.

[2] Zs. Baranyai, On the factorization of the complete uniform hypergraph, Infinite and finite sets, Vol. I, pp.91-108. *Colloq. Math. Soc. Janos Bolyai*, 10 North-Holland, Amsterdam, 1975.

[3] C. Berge, *Graphs and Hypergraphs*. 7 North-Holland, Amsterdam, 1973.

[4] J.C. Bermond, Hamiltonian decompositions of graphs, directed graphs and hypergraphs. *Ann. Discrete Math.* 3 (1978), 21-28.

[5] J.C. Bermond et al., Hypergraphes hamiltoniens. *Problèmes combinatoires et théorie des graphes, Colloq. Internat. CNRS, Paris* 260 (1973) 39-43.

[6] D. Bryant et al., Decompositions of complete 3-uniform hypergraphs into small 3-uniform hypergraphs. *Australas. J. Combin.* 60 (2014), 227-254.

[7] D. Bryant and S. El-Zanati, Graph Decompositions, in *The CRC Handbook of Combinatorial Designs*, 2nd edition (Eds. C.J. Colbourn and J.H. Dinitz), CRC Press, Boca Raton (2007), 477-486.

[8] T. Feng and Y. Chang, Decompositions of the 3-uniform hypergraphs $K_v^{(3)}$ into hypergraphs of a certain type. *Science in China(A)*. 50 (2007), 1035-1044.

[9] T. Feng and Y. Chang, Decompositions of 3-uniform hypergraph $K_v^{(3)}$ into hypergraph $K_4^{(3)} + e$. *Util. Math.* 87 (2012), 49-64.

[10] H. Hanani, On quadruple systems, *Canad. J. Math.* 12 (1960), 145-157.

[11] G.Y. Katona and H.A. Kierstead, Hamiltonian chains in hypergraphs, *J. Graph Theory* 30 (1999), 205-212.

[12] P. Keevash, The existence of designs, `arXiv:1401.3665` (2014), 56 pages.

[13] G.B. Khosrovshahi and R. Laue, t-Designs with $t \geq 3$, in *The CRC Handbook of Combinatorial Designs*, 2nd edition (Eds. C.J. Colbourn and J.H. Dinitz), CRC Press, Boca Raton (2007), 79-101.

[14] J. Kuhl and M. W. Schroeder, Hamilton cycle decompositions of k-uniform k-partite hypergraphs. *Australas. J. Combin.* 56 (2013), 23-37.

[15] D. Kühn and D. Osthus, Decompositions of complete uniform hypergraphs into Hamilton Berge cycles. *J. Combin. Theory, Series A* 126 (2014), 128-135.

[16] M. Meszka and A. Rosa, Decomposing complete 3-uniform hypergraphs into Hamiltonian cycles. *Australas. J. Combin.* 45 (2009), 291-302.

[17] H. Mohácsy and D. K. Ray-Chaudhuri, A Construction for Infinite Families of Steiner 3-Designs, *J. Combin. Theory, Series A* 94 (2001), 127-141.

[18] R. C. Mullin and H-D. O. K. Gronau, PBDs and GDDs: The Basics, in *The CRC Handbook of Combinatorial Designs*, 2nd edition (Eds. C.J. Colbourn and J.H. Dinitz), CRC Press, Boca Raton (2007), 231-236.

[19] H. Verall, Hamilton decompositions of complete 3-uniform hypergraphs, *Discrete Math.* 132 (1994), 333-348.

Acknowledgements

To my supervisor, Barbara Maenhaut, thanks for all of the time and effort you've put into having me as a student. I appreciated the time set aside for meetings, reading drafts, tolerating my inability to meet my own goals, and giving support in all other aspects of my honours year.

To the 2014 mathematics honours cohort, thanks for the ride. It would have been much less interesting year without you.

To my family and friends, thanks for putting up with my attempts to tell you how interesting my studies are, especially after you'd already given up trying to understand.

To the LORD God, who created all things, I owe everything that I am. I praise God for the ability and opportunity to study and understand His creation, without which this work would not be possible. As I have increasingly learned to consider all things as loss because of the surpassing worth of knowing Christ Jesus my Lord, I present this work with the hope that it might proclaim His glory.