Computational Complexity of the Hamiltonian Cycle Problem in Dense Hypergraphs

Marek Karpiński[1,⋆], Andrzej Ruciński[2,⋆⋆], and Edyta Szymańska[2,⋆⋆⋆]

[1] Department of Computer Science, University of Bonn
marek@cs.uni-bonn.de
[2] Faculty of Mathematics and Computer Science,
Adam Mickiewicz University, Poznań
{rucinski,edka}@amu.edu.pl

Abstract. We study the computational complexity of deciding the existence of a Hamiltonian Cycle in some dense classes of k-uniform hypergraphs. Those problems turned out to be, along with the hypergraph Perfect Matching problems, exceedingly hard, and there is a renewed algorithmic interest in them. In this paper we design a polynomial time algorithm for the Hamiltonian Cycle problem for k-uniform hypergraphs with density at least $\frac{1}{2}+\epsilon$, $\epsilon > 0$. In doing so, we depend on a new method of constructing Hamiltonian cycles from (purely) existential statements which could be of independent interest. On the other hand, we establish NP-completeness of that problem for density at least $\frac{1}{k} - \epsilon$. Our results seem to be the first complexity theoretic results for the Dirac-type dense hypergraph classes.

1 Introduction

We address the problem of deciding the existence and construction of a Hamiltonian Cycle in some dense classes of hypergraphs. The corresponding problem is being well understood for dense graphs (cf., e.g., [7] and [19]), as well as random graphs (cf., e.g., [2],[4], [5], and [10]). However, the computational status of the problem for hypergraphs was widely open and has become a challenging issue recently.

In this paper we shed some light on the computational complexity of that problem for k-uniform hypergraphs. For any $\epsilon > 0$, we design the first polynomial time algorithm for the Hamiltonian Cycle problem for k-uniform hypergraphs with Dirac-type density at least $1/2 + \epsilon$. We prove also a complementary intractability result for k-uniform hypergraphs with density at least $1/k - \epsilon$. The techniques used in this paper could be also of independent interest.

We consider *k-uniform hypergraphs*, that is, hypergraphs H whose edges are k-element subsets of $V := V(H)$. We refer to k-uniform hypergraphs as *k-graphs*.

⋆ Research supported by DFG grants and the Hausdorff Center grant EXC59-1.
⋆⋆ Research supported by grant N201 036 32/2546.
⋆⋆⋆ Research supported by grant N206 017 32/2452.

A. López-Ortiz (Ed.): LATIN 2010, LNCS 6034, pp. 663–674, 2010.

For k-graphs with $k \geq 3$, a cycle may be defined in many ways (see, e.g., [3], [13] and [14]). Here by *a cycle* of length $l \geq k+1$ we mean a k-graph whose vertices can be ordered cyclically $v_1, \ldots, v_l$ in such a way that for each $i = 1, \ldots, l$, the set $\{v_i, v_{i+1}, \ldots, v_{i+k-1}\}$ is an edge, where for $h > l$ we set $v_h = v_{h-l}$. Such cycles are sometimes called *tight. A Hamiltonian cycle* in a k-graph H is a spanning cycle in H. A k-graph containing a Hamiltonian cycle is called *Hamiltonian.*

For a k-graph H and a set of $k-1$ vertices S, let $N_H(S)$ be the set of vertices v of H such that $S \cup \{v\} \in H$. We define the degree of S as $deg_H(S) = |N_S(H)|$, and write $deg_H(S,T)$ for the degree restricted to the subset $T \subseteq V$, that is, $deg_H(S,T) = |N_S(H) \cap T|$. We define $\delta(H) = \min_S deg_H(S)$ and refer to it as the $(k-1)$*-wise, collective minimum degree of* H, or simply, *minimum co-degree.* The ratio $\delta(H)/|V(H)|$ is sometimes called a *Dirac-type density of* H.

We denote by $\mathbf{HAM}(k,c)$ the problem of deciding the existence of a Hamiltonian cycle in a k-graph with minimum co-degree $\delta(H)$ satisfying $\delta(H) \geq c|V(H)|$.

For graphs, that is, for $k = 2$, one of the classic theorems of Graph Theory by Dirac [8] states that if the minimum degree in an n-vertex graph is at least $n/2$, $n \geq 3$, then the graph is Hamiltonian. Hence, the problem $\mathbf{HAM}(2, 1/2)$ is trivial. Complementing this result, it was shown in [7] that $\mathbf{HAM}(2,c)$ is NP-complete for any $c < \frac{1}{2}$.

Turning to genuine hypergraphs ($k \geq 3$), it was recently shown in [16] that for all $k \geq 3$, $c > \frac{1}{2}$, and sufficiently large n, every k-graph H with $|V(H)| = n$ and $\delta(H) \geq cn$ contains a Hamiltonian cycle. Hence, again, $\mathbf{HAM}(k,c)$ is trivial for all $c > \frac{1}{2}$. In the case in which $c = \frac{1}{2}$ Rödl et al.[17] proved that the same holds for 3-graphs and the problem remains open for $k \geq 4$.

Our main contribution are two complementary results on $\mathbf{HAM}(k,c)$.

Theorem 1. *For all $k \geq 3$ and $c < \frac{1}{k}$ the problem $\mathbf{HAM}(k,c)$ is NP-complete.*

Interestingly, Theorem 1 leaves a similar hardness gap of $(\frac{1}{k}, \frac{1}{2})$ as for the problem of deciding the existence of a perfect matching in a k-graph with $\delta(H) \geq c|V(H)|$ (see [18] and [12]). Note that, in view of [7], this gap collapses for graphs. In Section 2, Theorem 1 is proved by a reduction from $\mathbf{HAM}(2,c)$, $c < \frac{1}{2}$.

In the second part of this paper, we strengthen the above mentioned result from [16], by designing a polynomial time algorithm for the search version of $\mathbf{HAM}(k,c)$.

Theorem 2. *For all $k \geq 3$ and $c > \frac{1}{2}$ there exists a polynomial time algorithm, called* HamCycle*, which finds a Hamiltonian cycle in every k-graph with $\delta(H) \geq c|V(H)|$.*

In view of [17], we believe that also the proof from there can be turned into a polynomial time algorithm extending Theorem 2 to $c = \frac{1}{2}$ for $k = 3$.

Our construction is based on the existential proofs from [16] and [17]. In short, the idea is as follows. First, procedure AbsorbingPath constructs a special, relatively short path A in H, called *absorbing.* Next, procedure AlmostHamCycle finds an almost Hamiltonian cycle C containing A. Finally, the remaining vertices are absorbed by A into C to form a Hamiltonian cycle. Along the way, two probabilistic lemmas from [16] are derandomized using the Erdős-Selfridge method of conditional expectations [1].

2 The Reduction

In this section we prove Theorem 1. We will show that for all $k \geq 3$ and all $\epsilon > 0$, the problem $\mathbf{HAM}(k-1, \frac{1}{k-1} - \epsilon')$, where $\epsilon' = \frac{k}{k-1}\epsilon$ reduces to $\mathbf{HAM}(k, \frac{1}{k} - \epsilon)$. This, together with the known fact proved in [7] that $\mathbf{HAM}(2, c)$ is NP-complete for all $c < \frac{1}{2}$, shows that also $\mathbf{HAM}(k, c)$ is NP-complete for all $c < \frac{1}{k}$.

Let H be a $(k-1)$-graph on $(k-1)n$ vertices with $\delta(H) \geq (\frac{1}{k-1} - \epsilon')(k-1)n$. We construct a (gadget) k-graph G as follows. Let $V(G) = A \cup B$ where $A = V(H)$ and B is disjoint from A with $|B| = n$. The edge set $E(G)$ is union of three sets:

$$E(G) = E_{\leq k-3} \cup E_H \cup E_k,$$

where for $i = 0, \ldots, k$, E_i consists of all k-element subsets of $V(G)$ which intersect A in precisely i vertices, $E_{\leq k-3} = \bigcup_{i \leq k-3} E_i$ and E_H consists of all k-element subsets of $V(G)$ whose intersection with A is an edge of H (see Figure 1). Let us check first that $\delta(G) \geq (\frac{1}{k} - \epsilon)kn$. We assume, as we can, that $\epsilon kn \geq 2$. Let $S \in \binom{V(G)}{k-1}$. If $|S \cap A| = k-3$ (and so $|S \cap B| = 2$) then $deg_G(S) = |B| - 2 = n-2$. If $|S \cap A| \leq k-4$ then $deg_G(S) = |V(G)| - (k-1) = kn - k + 1$. If $S \subset A$, then $deg_G(S) \geq |A| - (k-1) \geq n$, regardless whether $S \in E(H)$ or not. Finally, if $|S \cap A| = k-2$, we know by the assumption on $\delta(H)$ that there are at least

$$\left(\frac{1}{k-1} - \frac{k}{k-1}\epsilon\right)(k-1)n = \left(\frac{1}{k} - \epsilon\right)kn$$

vertices $v \in A \setminus S$ such that $(S \cap A) \cup \{v\} \in E(H)$, and hence, $S \cup \{v\} \in E_H \subset E(G)$.

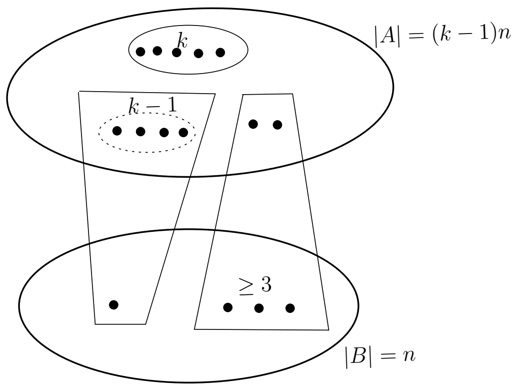

Fig. 1. The gagdet. The dotted oval represents an edge of H.

It remains to show that H has a Hamiltonian cycle if and only if G does. Let $v_1 v_2 \dots v_{(k-1)n}$ be a Hamiltonian cycle in H and let us order the vertices of B arbitrarily, say $B = \{w_1, \dots, w_n\}$. Then the sequence

$$v_1 \dots v_{k-1} w_1 v_k \dots v_{2k-2} w_2 \dots w_{n-1} v_{(k-1)(n-1)+1} \cdots v_{(k-1)n} w_n \tag{1}$$

forms a Hamiltonian cycle in G. Indeed, every k consecutive (cyclically) vertices of that string contain exactly one vertex of B and an edge of H, and thus, by the definition of E_H, form an edge of G.

Conversely, let G have a Hamiltonian cycle F. Note first that $E(F) \not\subseteq E_{\le k-3}$, because each edge of $E_{\le k-3}$ contains at most $k-3$ vertices of A and each vertex of F is contained in precisely k edges of F. Hence, F could cover only at most

$$kn \times (k-3) \times \frac{1}{k} = (k-3)n < |A|$$

vertices of A. But then $E(F) \cap E_{\le k-3} = \emptyset$, because, due to the lack of edges of E_{k-2} in G, the cycle cannot traverse from an edge of $E_{\le k-3}$ to any edge in $E_H \cup E_k$.

Secondly, no edge of E_k can be in F either. Indeed, since the edges of F covering B are all in $E_H \subseteq E_{k-1}$, each vertex of B has to be immediately preceded in F by exactly $k-1$ vertices of A, making no room for any edge of E_k in F. So, F looks exactly like in (1). Note that every k consecutive (cyclically) vertices of that string form an edge of G and contain exactly one vertex of B and a set S of $k-1$ vertices of A. Thus, by the definition of G, this set S must be an edge of H. Hence, the sequence $v_1 v_2 \dots v_{(k-1)n}$ forms a Hamiltonian cycle in H. $\square$

3 Subroutines

In this section we describe several subroutines which will be used by the main algorithm, HamCycle. We begin with procedures constructing tight paths in a dense hypergraph. Wherever convenient, we will identify a sequence of distinct vertices $(v_1, v_2, \dots)$ with the set of its elements $\{v_1, v_2, \dots\}$.

3.1 Paths

A path is a k-graph P, whose vertices can be ordered $v_1, \dots, v_l$, where $l = |V(P)|$, in such a way that for each $i = 1, \dots, l-k+1$, we have $\{v_i, v_{i+1}, \cdots, v_{i+k-1}\} \in P$. We say that P *connects* the sequences $(v_1, v_2, \dots v_{k-1})$ and $(v_l, \dots, v_{l-k+2})$, which will be called *the ends* of P. A path on l vertices (and thus with $l-k+1$ edges) will be said to have *length* l.

Our algorithm will frequently use the following subroutine. Let $\gamma > 0$. By Lemma 4 in [16], for sufficiently large n, every k-graph H with n vertices and $\delta(H) \ge (\frac{1}{2} + \gamma)n$ contains a path of length at most $2k/\gamma^2$ between any pair of $(k-1)$-element sequences of distinct vertices. Thus, an exhaustive search of all

$O(n^{2k/\gamma^2})$ sequences of distinct $2k/\gamma^2$ vertices would certainly find such a path. However, for better complexity, a BFS-type search can be applied.

Subroutine Connect

In: k-graph H with $\delta(H) \geq (\frac{1}{2}+\gamma)n$ and two disjoint $(k-1)$-element sequences of distinct vertices, $\mathbf{u}$ and $\mathbf{v}$
Out: Path P in H with ends $\mathbf{u}$ and $\mathbf{v}$ of length at most $2k/\gamma^2$.

Connect begins its BFS search at $\mathbf{u}$ and moves on by one vertex at a time until the reverse of $\mathbf{v}$ is found. Throughout it maintains a record of the path by which the current end has been reached, and uses this record to verify that the new vertex added is distinct from all previous on the current path. Each step corresponds to traversing one edge of H in a particular order and no edge is traversed twice in the same order. Hence, the time complexity of Connect is $O(n^k)$.

In fact, we will rather need a restricted version of Connect, where the connecting path is supposed to use, except for the ends $\mathbf{u}$ and $\mathbf{v}$, only the vertices from a specified ,,transfer set" T.

Subroutine ConnectVia

In: k-graph H, a subset $T \subset V$ such that for every $S \in \binom{V}{k-1}$ we have $deg_H(S,T) \geq (\frac{1}{2}+\gamma)|T|$, and two $(k-1)$-element sequences of distinct vertices from $V \setminus T$, $\mathbf{u} = (u_1, u_2, \ldots u_{k-1})$ and $\mathbf{v} = (v_1, v_2, \ldots v_{k-1})$
Out: Path P in H with ends $\mathbf{u}$ and $\mathbf{v}$ of length at most $2k/\gamma^2 + 2(k-1)$ and such that $V(P) \setminus (\mathbf{u} \cup \mathbf{v}) \subset T$

In its first $2k-2$ steps ConnectVia moves from $\mathbf{u}$ and $\mathbf{v}$ to, resp., $\mathbf{u}'$ and $\mathbf{v}'$, where all vertices of $\mathbf{u}'$ and $\mathbf{v}'$ are in the set T. Then it invokes Connect with $H[T]$, $\mathbf{u}'$, and $\mathbf{v}'$ as inputs.

Another subroutine finds a long path in any dense k-graph. It is an algorithmic generalization of Claim 6.1 from [17]. For a k-graph F denote by $\delta_{>0}(F)$ the minimum of $deg_F(S)$ taken over all $S \in \binom{V(F)}{k-1}$ with $deg_F(S) > 0$.

Subroutine LongPath

In: k-graph F with l vertices and $m > 0$ edges
Out: Path P in F of length at least $d := m/\binom{l}{k-1}$.

1. $V(F) := V,\ F' := F$
2. Find a set $S \in \binom{V}{k-1}$ for which $deg_{F'}(S) = \delta_{>0}(F')$;
3. If $\delta_{>0}(F') < d$, then $F' := F' \setminus \{e \in F' : e \supset S\}$ and go to Step 2;
4. Greedily find a maximal path P in F';
5. Return P.

Observe that at the outset of Step 3 we have $deg_{F'}(S) = 0$ and so, every set S is selected in Step 2 at most once. Note also that once we get to Step 4, we have

$F' \neq \emptyset$ and $\delta_{>0}(F') \geq d$. Hence, any maximal path in F' has length at least d. The time complexity of LongPath is $O(l^{k-1} + m)$.

3.2 Derandomization

At the heart of our algorithm lies the following procedure based on a simple probabilistic fact. Let $\tau > 0$, $\beta > \tau$, and m, N, and $r \leq N$, be positive integers. Set $\rho := 2mr/N^2$.

Algorithm SelectSubset

In: Graph $G = (U \cup W, E)$ such that

- $|U| = M$ and $e(G[U]) = 0$
- $|W| = N$ and $e(G[W]) = m$
- $\min_{u \in U} deg_G(u) \geq \beta N$,

and an integer r, $1 \leq r \leq N$
Out: Independent set $R \subset W$ with $(1-\rho)r \leq |R| \leq r$ and $\min_{u \in U} deg_G(u, R) \geq (\beta - \tau - \rho)\, r$.

1. Set $U = \{u_1, \ldots, u_M\}$, $W = \{w_1, \ldots, w_N\}$;
2. $R' := \emptyset$;
3. For $k = 1$ to r do:
   (a) For $i = 1$ to M and $j = 1$ to N do:
   $$d'_{i,j} := deg_G(u_i, R' \cup \{w_j\}) \quad \text{and} \quad d''_{i,j} := deg_G(u_i, W \setminus (R' \cup \{w_j\})).$$
   (b) For $j = 1$ to N do:
   $$e_j^{(0)} := e(G[R' \cup \{w_j\}]), e_j^{(1)} := e(G[R', W \setminus R']), e_j^{(2)} := e(G[W \setminus (R' \cup \{w_j\})]).$$
   (c) Find $w_{j_k} \in W \setminus R'$ such that, with $y := 2m(r/N)^2$,
   $$\sum_{i=1}^{M} \sum_{d \leq (\beta-\tau)r - d'_{i,j_k}} \frac{\binom{d''_{i,j_k}}{d}\binom{N-k-d''_{i,j_k}}{r-k-d}}{\binom{N-k}{r-k}} + \frac{1}{y}\left(e_{j_k}^{(0)} + e_{j_k}^{(1)} \frac{r-k}{N-k} + e_{j_k}^{(2)} \frac{(r-k)(r-k-1)}{(N-k)(N-k-1)}\right) < 1. \tag{2}$$
   (d) $R' := R' \cup \{w_{j_k}\}$.
4. Remove one vertex from each edge of $G[R']$ and call the resulting set R.
5. Return R.

Lemma 1. *If* $\log M = o(r)$, *then* SelectSubset *finds the desired set* R *in time* $O(M \times poly(N))$.

In the proof we use the following probabilistic fact, which together with Markov's inequality implies the existence of the required set R. Algorithm SELECTSUBSET derandomizes this fact.

Fact 3. *Let G be the graph given as an input of* SELECTSUBSET. *Further, let R' be a random subset of W chosen uniformly from $\binom{W}{r}$, let X be the number of vertices $u \in U$ with $deg_G(u, R') \leq (\beta - \tau)r$ and $Y = e(G[R'])$. Then*

$$EX = o(1) \qquad and \qquad EY \leq m\left(\frac{r}{N}\right)^2.$$

Proof. First observe that $X = \sum_{u \in U} I_u$, where I_u is the indicator of the event $\{deg_G(u, R') \leq (\beta - \tau)r\}$. Note also that $P(I_u = 1) = P(Z_u \leq (\beta - \tau)r)$, where Z_u is a hypergeometric random variable with parameters $N, deg_G(u), r$ and that, by the properties of G, the expectation of Z_u is $r deg_G(u)/N \geq \beta r$. Thus, by a Chernoff bound for hypergeometric distributions (see, e.g., [11], Theorem 2.10, formula (2.6)), $EX \leq Me^{-\Theta(r)} = o(1)$. Finally, by the linearity of expectation, $EY = m\binom{N-2}{r-2}/\binom{N}{r} \leq m(r/N)^2$. $\square$

Proof of Lemma 1: By Fact 3 and the definition of y, $EX + \frac{EY}{y} \leq o(1) + \frac{1}{2} < 1$. We can view the selection of R' as a result of a random process $w_{j_1}, \ldots, w_{j_r}$, where in step k, a vertex $w_{j_k} \in W$ is randomly selected without repetitions.

Let $\alpha_j = E(X|j_1 = j) + \frac{EY}{y}E(Y|j_1 = j)$. Then, by the law of total probability, $EX + \frac{EY}{y} = \frac{1}{N}\sum_{j=1}^{N} \alpha_j$, and so, there exists an index j such that $\alpha_j < 1$. Take that index as j_1. Repeat until the whole set R' is selected. Then, $E(X|R') + \frac{1}{y}E(Y|R') = X(R') + \frac{1}{y}Y(R') < 1$, which implies that $X(R') = 0$ and $Y(R') < y$. This proves that R', and consequently R, have the desired properties.

Note that the conditional expectations $E(X|j_1, \ldots, j_k)$ and $E(Y|j_1, \ldots, j_k)$ correspond to the quantities appearing in the expression (2) given in Step 3(c) of the algorithm. $\square$

4 The Algorithm

In this section we prove Theorem 2 by giving the main algorithm HAMCYCLE. It will be based on two major procedures, ABSORBINGPATH and ALMOSTHAMCYCLE which we will describe first.

In order to formulate our main procedures, we need a few definitions from [16]. We choose $0 < \epsilon < c - \frac{1}{2}$ small enough.

Given a vertex v we say that a $(2k-2)$-element sequence of vertices $\mathbf{x} = (x_1, \ldots, x_{2k-2})$ is *v-absorbing* in H if for every $i = 1, \ldots, k-1$ we have $\{x_i, x_{i+1}, \ldots, x_{i+k-1}\} \in H$ (that is, $\mathbf{x}$ spans a path in H) and for every $i = 1, \ldots, k$ we also have edges $\{x_i, x_{i+1}, \ldots, x_{i+k-2}, v\} \in H$. Note that, if $\mathbf{x}$ is actually a segment of a path P and v is not a vertex of P, then the segment $\mathbf{x}$ of P can be replaced by the new segment $(x_1, \ldots, x_{k-1}, v, x_k, \ldots, x_{2k-2})$, absorbing v onto P.

A path A in H is called *absorbing* if $|V(A)| \leq 8k\epsilon^{k-1}n$ and for every $v \in V$ there are at least

$$q := 2^{k-4}\epsilon^{2k}n \tag{3}$$

disjoint v-absorbing sequences, each of which is a segment of A. Note that if A is an absorbing path in H, then for every subset $U \subset V \setminus V(A)$ of size $|U| \leq q$ there is a path A_U in H with $V(A_U) = V(A) \cup U$ and such that A_U has the same ends as A.

The idea behind our algorithm is the same as the idea of the existential proofs in [16] and [17], and can be summarized as follows.

- Find an absorbing path A in H.
- Find a cycle C in H containing A as well as all but at most q vertices of $V(H) \setminus V(A)$.
- Extend C to a Hamiltonian cycle of H using the absorbing property of A with respect to $U = V(H) \setminus V(C)$.

To build an absorbing path we use Procedure AbsorbingPath and to build the long cycle – Procedure AlmostHamCycle, both described below.

We are now ready to give our main algorithm which finds a Hamiltonian cycle in every k-graph with $\delta(H) \geq cn$, for $c > \frac{1}{2}$.

Algorithm HamCycle

In: n-vertex k-graph H with $\delta(H) \geq cn$, $c > \frac{1}{2}$
Out: Hamiltonian cycle C in H

1. Fix a sufficiently small $0 < \epsilon < c - \frac{1}{2}$;
2. Apply AbsorbingPath to H obtaining an absorbing path A;
3. Apply AlmostHamCycle to H with $P_0 = A$, obtaining a cycle C in H of length at least $n - q$ which contains A;
4. For each vertex $v \in V \setminus V(C)$ do:
   (a) Find a v-absorbing sequence $\mathbf{x} = (x_1, \ldots, x_{2k-2})$ which is a segment of C;
   (b) Replace $(x_1, \ldots, x_{2k-2})$ by $(x_1, \ldots, x_{k-1}, v, x_k, \ldots, x_{2k-2})$ and call the new cycle C;
5. Return C.

It remains to describe how the two procedures used by HamCycle work. By Claim 3.2 in [16] we know that for every vertex $v \in V(H)$ there are at least $2^{k-2}\gamma^{k-1}$ v-absorbing sequences in H. In [16] a random selection of $(2k-2)$-sequences was chosen and proved to contain enough v-absorbing sequences for every v. Here we derandomize this step by invoking SelectSubset.

Procedure AbsorbingPath

In: n-vertex k-graph H with $\delta(H) \geq (\frac{1}{2} + \epsilon)n$
Out: Absorbing path A in H

1. Build an auxiliary graph $G = (U \cup W, E)$, where $U = V(H)$, W is the set of all $(2k-2)$-element sequence of vertices $\mathbf{x} = (x_1, \ldots, x_{2k-2})$ in H, and E consists of all pairs $v \in U, \mathbf{x} \in W$ such that $\mathbf{x}$ is v-absorbing in H, as well as of all pairs $\mathbf{x}, \mathbf{x}' \in W$ which share at least one element;

2. Apply SelectSubset to G with $r = \epsilon^{k+1} n$, $\rho = 8(k-1)^2\epsilon^{k+1}$, and $\tau = \beta/2$, to obtain a family $\mathcal{F}$ of $s \le r$ vertex-disjoint sequences and such that for each vertex v of H the number of v-absorbing sequences in $\mathcal{F}$ is at least $2^{k-4}\epsilon^{2k} n$;
3. Use repeatedly ConnectVia to connect all sequences of $\mathcal{F}$ into one path A.

Note that in the above application of SelectSubset, $M = n$, $N = (n)_{(2k-2)} \sim n^{2k-2}$, $m \le (2k-2)^2 n^{4k-5}$, and $\beta = 2^{k-2}\gamma^{k-1}$. Thus, SelectSubset does find a family $\mathcal{F}$ as described in Step 2. As the final path A contains all elements of $\mathcal{F}$ as disjoint segments, the absorbing property of A follows.

Our second major procedure constructs in H an almost Hamiltonian cycle containing any given, not too long path. In [16] this has been done by applying a weak regularity lemma to H and finding in the cluster k-graph an almost perfect matching. Then, applying repeatedly the existential analog of LongPath to the dense and regular clusters, a collection of finitely many paths covering almost all vertices of H was found. These paths were then connected into a cycle by applying ConnectVia with a preselected reservoir set R.

That proof can be turned into an algorithm by recalling the algorithmic version of the weak hypergraph regularity lemma from [6]. We, however, prefer to follow the more elementary approach from [17], generalizing it to k-graphs without any effort.

In fact, the single difficulty in both these approaches was the same: to derandomize the selection of a reservoir set R, a small subset of vertices which reflects the property of the entire hypergraph and can be used to connect paths during the whole procedure. This step is now derandomized by using algorithm SelectSubset (Steps 1 and 2 of procedure AlmostHamCycle).

Once we have R which is disjoint from P_0, we keep extending P_0 in $H - R$ by little increments until it reaches the desired length. Initially, we extend P_0 greedily (Step 3), using the fact that $\delta(H - R) > n/2$. After reaching the length of $n/2$, in every step we look at $L := V \setminus (V(P) \cup R)$, where P is the current path, and consider two cases.

If $H[L]$ is dense we apply LongPath to find a long path P' in $|H[L]|$ and connect it via R using ConnectVia with the transfer set R (Step 5(c)).

If $H[L]$ is sparse then many edges of H have $k-1$ vertices in L and one in P. By averaging, there must be a constant length segment I of P with many such edges incident to I, and, again by averaging, a subset $J \subset I$ with $|J| \ge \frac{4}{3k}|I|$ and whose *every* vertex is hit by *the same* set H_0 of $(k-1)$-tuples from L (Step 5(d)(i)). Next a $(k-1)$-partite $(k-1)$-clique K is found in H_0 and trivially extended, by adding J, to a k-partite k-clique K'. Clique K' contains a spanning Hamiltonian path Q whose length is $\frac{4}{3}|I|$. We then cut I out of P and reconnect the two remaining subpaths, P_1 and P_2, with Q, obtaining a path longer by $\frac{1}{3}|I|$ (Step 5(d)(ii)-(vi)). Finally, when P has grown long enough, we connect the two ends of P to form the desired cycle. All connections are via R using ConnectVia.

For details of the case $k = 3$ we refer to [17]. Since the general case has not appeared in the literature yet, we provide here a detailed pseudo-code, followed by a formal proof of the most crucial steps.

Let D be a large integer, say

$$D \gg n/q = 2^{4-k}\epsilon^{-2k},$$

where q is given by (3).

PROCEDURE ALMOSTHAMCYCLE

In: n-vertex k-graph H with $\delta(H) \geq (\frac{1}{2} + \epsilon)n$ and a path P_0 in H of length at most $\frac{1}{3}\epsilon n$
Out: Cycle C in H such that $P_0 \subset C$ and $|V(C)| \geq n - q$.

1. Build an auxiliary graph $G = (U \cup W, E)$, where $U = \binom{V}{k-1}$, $W = V \setminus V(P_0)$, and E consists of all pairs $S \in U, v \in W$ such that $S \cup \{v\} \in H$;
2. Apply SELECTSUBSET to G with $r = \frac{1}{2}q$, and $\tau = \epsilon/6$, to obtain a set $R \subset V \setminus V(P_0)$ of size $|R| = r$ with the property that $deg_H(S, R) \geq \frac{1}{2}(1+\epsilon)r$ for all $S \in \binom{V}{k-1}$.
3. Extend greedily P_0 (at one end only) to a path P in $H - R$ of length at least $n/2$;
4. Let $\mathbf{x}$ be the common end of P_0 and P;
5. While $|V(P)| < n - q$ do:
   (a) let $\mathbf{y}$ be the end of P other than $\mathbf{x}$;
   (b) $L := V \setminus (V(P) \cup R)$, $l := |L|$;
   (c) If $|H[L]| > D\binom{l}{k-1}$ then do:
      i. Apply LONGPATH to $H[L]$ obtaining a path P' of length at least D, disjoint from P.
      ii. Apply CONNECTVIA with $\gamma = \frac{1}{3}\epsilon$ and $T = R$, obtaining a path Q of length at most $20k/\epsilon^2$ from $\mathbf{y}$ to $\mathbf{x}'$, and thus connecting paths P and P' into a new path PQP';
      iii. $P := PQP'$, $R := R \setminus V(Q)$;
   (d) If $|H[L]| \leq D\binom{l}{k-1}$ then do:
      i. Find (by exhaustive search) a segment (that is, a set of consecutive vertices) $I \subset V(P) \setminus (V(P_0) \cup \mathbf{y})$, a subset $J \subset I$, and a $(k-1)$-graph $H_0 \in \binom{L}{k-1}$ such that $|I| = D$, $|J| = \frac{4}{3k}D$, $|H_0| \geq 2^{-D}(\frac{1}{2} - \frac{4}{3k})\binom{l}{k-1}$, and for every $e \in H_0$ and every $v \in J$ we have $e \cup \{v\} \in H$;
      ii. Find (by exhaustive search) a $(k-1)$-partite, complete $(k-1)$-graph K in H_0 with all partition classes of size $|J|$;
      iii. Let K' be the k-partite, complete k-graph spanned in H by the partition classes of K and J;
      iv. Take any Hamiltonian path Q in K' with ends $\mathbf{z}$ and $\mathbf{z}'$;
      v. Remove I from P obtaining two disjoint paths $P_1 \supset P_0$ and P_2;
      vi. Apply CONNECTVIA with $\gamma = \frac{1}{3}\epsilon$ and $T = R$, to connect P_1, Q, and P_2 together (see Figure 2); call the resulting path P;

6. Apply ConnectVia with $\gamma = \frac{1}{3}\epsilon$ and $T = R$, to the ends $\mathbf{x}$ and $\mathbf{y}$ of P, obtaining a path Q of length at most $20k/\epsilon^2$ from $\mathbf{x}$ to $\mathbf{y}$, and thus creating a cycle $C = PQ$ of length at least $n - q$;
7. Return C.

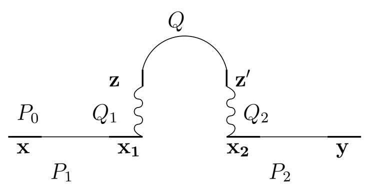

Fig. 2. Illustration to Step 4(d) of procedure AlmostHamCycle

Fact 4. AlmostHamCycle *constructs a cycle C in H such that $P_0 \subset C$ and $|V(C)| \geq n - q$.*

Proof. The graph G constructed in Step 1 has parameters $M = \binom{n}{k-1}$, $(1 - 8k\epsilon^{k-1})n \leq N \leq n$, $m = 0$, and $\beta \geq \frac{1}{2} + \frac{2}{3}\epsilon$, and so SelectSubset does find a set R as described in Step 2.

Now we prove that the sets I and J, and a $(k-1)$-graph H_0 searched for in Step 5(d)(i) do exist. By estimating the sum $\sum_{S \in \binom{L}{k-1}} deg_H(S)$ in two ways we derive the inequality

$$(\tfrac{1}{2} + \gamma)n\binom{l}{k-1} \leq kD\binom{l}{k-1} + |R \cup V(P_0) \cup \mathbf{y}|\binom{l}{k-1} + N,$$

where N counts the number of edges of H with $k-1$ vertices in L and one vertex in $V(P) \setminus (V(P_0) \cup \mathbf{y})$. Since $|R \cup V(P_0) \cup \mathbf{y}| \leq \frac{3}{4}\gamma n$, this yields that

$$N \geq [(\tfrac{1}{2} + \tfrac{1}{4}\gamma)n - O(1)]\binom{l}{k-1}.$$

Let N_i be the number of edges of H counted by N, with one vertex in the i-th D-element segment I_i of $V(P) \setminus (V(P_0) \cup \mathbf{y})$. Then, with $s := |V(P) \setminus (V(P_0) \cup \mathbf{y})|$ we have

$$\sum_{i=1}^{s-D+1} N_i \geq ND - O(1)\binom{l}{k-1} \geq (\tfrac{1}{2} + \tfrac{1}{5}\gamma)n\binom{l}{k-1}D,$$

so, by averaging, there exists i such that $N_i \geq \frac{1}{2}\binom{l}{k-1}D$. Let H_i be the $(k-1)$-graph of all $S \in \binom{L}{k-1}$ with at least $\frac{4}{3k}D$ neighbors in $I := I_i$. Then $|H_i| \geq (\frac{1}{2} - \frac{4}{3k})\binom{l}{k-1}$. For each $J \subset I$, $|J| \geq \frac{4}{3k}D$, let H_J be the set of those edges of H_i whose H-neighborhood in I is exactly J. By averaging there exists a set J such that $|H_J| \geq 2^{-D}|H_i|$.

The existence of a $(k-1)$-partite, complete $(k-1)$-graph K in H_0 with all partition classes of size $|J|$ searched for in Step 5(d)(ii) follows by an old result of Erdős [9], see also [15], Lemma 8. (Recall that $|L| > \frac{1}{2}q$.) Note that the initial path P_0 has stayed intact throughout the entire procedure and so, it is contained in C. $\square$

Finally, note that the time complexity of AlmostHamCycle is $O(poly(n))$. This completes the proof of Theorem 2.

References

1. Alon, N., Spencer, J.H.: The Probabilistic Method, 3rd edn. Wiley, Chichester (2008)
2. Angluin, D., Valiant, L.: Fast probabilistic algorithms for Hamiltonian circuits and matchings. J. Comput. System Sci. 18, 155–193 (1979)
3. Bermond, J.C., et al.: Hypergraphes hamiltoniens. Prob. Comb. Theorie Graph Orsay 260, 39–43 (1976)
4. Bollobas, B., Fenner, T.I., Frieze, A.: An algorithm for finding hamiltonian cycles in random graphs. In: Proceedings of the 17th Annual ACM Symposium on Theory of Computing, pp. 430–439 (1985)
5. Broder, A., Frieze, A., Shamir, E.: Finding hidden Hamilton cycles. Random Structures Algorithms 5, 395–410 (1994)
6. Czygrinow, A., Rödl, V.: An algorithmic regularity lemma for hypergraphs. SIAM Journal on Computing 30, 1041–1066 (2000)
7. Dalhaus, E., Hajnal, P., Karpiński, M.: On the parallel complexity of Hamiltonian cycle and matching problem on dense graphs. J. Algorithms 15, 367–384 (1993)
8. Dirac, G.A.: Some theorems of abstract graphs. Proc. London Math. Soc. 3, 69–81 (1952)
9. Erdős, P.: On extremal problems of graphs and generalized graphs, Israel. J. Math. 2, 183–190 (1964)
10. Frieze, A., Reed, B.: Probabilistic Analysis of Algorithms. In: Probabilistic Methods for Algorithmic Discrete Mathematics, pp. 36–92. Springer, Heidelberg (1998)
11. Janson, S., Łuczak, T., Ruciński, A.: Random Graphs. John Wiley and Sons, New York (2000)
12. Karpiński, M., Ruciński, A., Szymańska, E.: The Complexity of Perfect Matching Problems on Dense Hypergraphs. In: ISAAC 2009 (2009) (to appear)
13. Katona, G.Y., Kierstead, H.A.: Hamiltonian chains in hypergraphs. J. Graph Theory 30, 205–212 (1999)
14. Kühn, D., Osthus, D.: Loose Hamilton cycles in 3-uniform hypergraphs of high minimum degree. JCT B 96(6), 767–821 (2006)
15. Rödl, V., Ruciński, A., Schacht, M.: Ramsey Properties of Random k-Partite, k-Uniform Hypergraphs. SIAM J. of Discrete Math. 21(2), 442–460 (2007)
16. Rödl, V., Ruciński, A., Szemerédi, E.: An approximate Dirac-type theorem for k-uniform hypergraphs. Combinatorica 28(2), 229–260 (2008)
17. Rödl, V., Ruciński, A., Szemerédi, E.: Dirac-type conditions for hamiltonian paths and cycles in 3-uniform hypergraphs. The Mittag-Leffler preprint series (Spring 2009) (submitted)
18. Szymańska, E.: The Complexity of Almost Perfect Matchings in Uniform Hypergraphs with High Codegree. In: Fiala, J. (ed.) IWOCA 2009. LNCS, vol. 5874, pp. 438–449. Springer, Heidelberg (2009)
19. Särkózy, G.: A fast parallel algorithm for finding Hamiltonian cycles in dense graphs. Discrete Mathematics 309, 1611–1622 (2009)