A SHORT PROOF OF A VARIANT OF THE HYPERGRAPH CONTAINER LEMMA FOR 3-UNIFORM HYPERGRAPHS

JÓZSEF BALOGH

We present a variant hypergraph container theorem for 3-uniform hypergraphs, where the dependency of all constants are computed. Note that all the mathematical ideas are coming from [1] and [2].

Let $\mathcal{H}$ be a 3-uniform hypergraph with average degree d. For every $S \subseteq V(\mathcal{H})$, its co-degree, denoted by $d(S)$, is the number of edges in $\mathcal{H}$ containing S, i.e.,

$$d(S) = |\{e \in E(\mathcal{H}) : S \subseteq e\}|.$$

Denote by Δ_2 the maximum co-degree of $\mathcal{H}$, i.e.,

$$\Delta_2 = \max\{d(S) : S \subseteq V(\mathcal{H}), |S| = 2\}.$$

For parameters s, t and $A \subset V(\mathcal{H})$, a vertex $v \in A$ is (A, s, t)-*eligible*, if there is a subgraph G_v of its link graph spanned on A with maximum degree at most s and with at least t edges. We say that a set $A \subset V(\mathcal{H})$ is an (s, t)-*core* in $\mathcal{H}$ if it does not have an (A, s, t)-eligible vertex.

Theorem 0.1. [Container theorem for 3-uniform hypergraphs]
Fix positive parameters $\delta, \Delta_2, s, t, M$ such that $\delta < 1/20$, $2s \leqslant \delta t$ and $2\Delta_2 \leqslant \delta^3 t$ are satisfied. Let $q := \max\left\{s,\ \frac{\Delta_2}{\delta^2},\ \sqrt{\frac{\Delta_2 t}{2\delta}}\right\}$. Let $\mathcal{H}$ be a 3-uniform M-vertex hypergraph such that the maximum co-degree of $\mathcal{H}$ is at most Δ_2. Then there exists a family $\mathcal{C} \subseteq \mathcal{P}(V(\mathcal{H}))$ satisfying the following:

(1) For every independent set $I \subseteq V(\mathcal{H})$, there exists a $C_I \in \mathcal{C}$ such that $I \subseteq C_I$.
(2) For every $C \in \mathcal{C}$ there are $T_2, T_1 \subset V(\mathcal{H})$ with $|T_1| \leqslant (\Delta_2/\delta q)M$, $|T_2| \leqslant (2q/t)M$ such that T_1, T_2 determines C.
(3) Every $C \in \mathcal{C}$ either satisfies $|C| \leq (1-\delta)M$ or it contains an (s,t)-core of size at least $(1-20\delta)M$. □

Proof. We present an algorithm which deterministically finds small certificates T_1 and T_2 for an independent set I, and C_I will only depend on T_1 and T_2. The algorithm has two Phases. In Phase I, starting from $\mathcal{H}$, we produce a graph, and in Phase II we shall work on this graph.

Date: December 13, 2019.

Container algorithm

Input: An independent set $I \subseteq V(\mathcal{H})$.

Output: A set C such that $I \subseteq C \subseteq V(\mathcal{H})$, and two sets $T_1, T_2 \subseteq I$.

Phase I:

(1) Fix an arbitrary ordering of the vertices, and another arbitrary ordering of all graphs on the vertex set $V(\mathcal{H})$. These will be used to break ties.
(2) Set $A = V(\mathcal{H})$, which is the set of *available* vertices, and let $T_2 = \emptyset$ be the first *certificate* of I, and let L be the empty graph on the vertex set $V(\mathcal{H})$ (the *link graph* we build).
(3) If $|A| \leq (1-3\delta)|V(\mathcal{H})|$, then set $C := A \cup T_2$ and STOP. Otherwise, set $S := \{u \in A : |N_L(u) \cap A| \geq q\}$.
(4) If $\mathcal{H}$ spans on $A \setminus S$ an (s,t)-core if $|A \setminus S| > (1-20\delta)M$, then set $C_I := A \cup T_2$ and STOP. Otherwise, go to Phase II.
(5) Let $v \in A \setminus S$ be the largest degree vertex of $\mathcal{H}[A \setminus S]$ among $(A \setminus S, s, t)$-eligible vertices (break ties according to the ordering fixed in Step I 1). If $v \notin I$ then replace A by $A \setminus \{v\}$, replace L by $L \setminus \{v\}$ and return to Step I(3).
(6) We have $v \in I$ and v is $(A \setminus S, s, t)$-eligible. Let G_v be a subgraph of its link graph $\mathrm{L}_v\mathcal{H}([A \setminus S])$ with $\Delta(G_v) \leq s$ and $e(G_v) \geq t$ (to choose G_v break ties according to the ordering fixed in Step I(1)).
    (a) Set $T_2 := T_2 \cup \{v\}$.
    (b) Set $A := A \setminus \{v\}$.
    (c) Let $L := (L \cup G_v) \setminus \{v\}$. Note that since the input graph $\mathcal{H}$ has co-degree at most Δ_2, L is a multigraph with multiplicity of an edge is bounded by Δ_2.
    (d) Return to Step I(3).

Phase II:

(1) Initiate $T_1 = \emptyset$, the second *certificate*.
(2) Let v be the largest degree vertex in L (break ties according to the ordering fixed in Step I(1). If $d_L(v) < \delta q$ then set $C := A \cup T_2 \cup T_1$ and STOP.
(3) If $v \notin I$ then replace A by $A \setminus \{v\}$ and replace L by $L \setminus \{v\}$ and go to Step II(2).
(4) We have $v \in I$ and $d_L(v) \geq \delta q$. Set $T_1 := T_1 \cup \{v\}$, replace A by $A \setminus (N_L(v) \cup \{v\})$ and replace L by $L \setminus (N_L(v) \cup \{v\})$. Go to Step II(2).

End of algorithm.

Note that given the orderings in Step I(1), one can check that the containers only depend on the certificates T_1 and T_2, and not on (the unexplored part of) I.

Observation 0.2. *At every point in the algorithm, $\Delta(L) \leq 2q$.*

Proof. Every time we change L, we add to it a graph of maximum degree at most $s \leqslant q$. But as soon as some vertex gets degree at least q we put it in S and do not touch it until its degree goes below q again. Hence in L, the maximum possible degree is at most $q + s \leqslant 2q$. □

Observation 0.3. *[Small certificates.] After the algorithm stops we have $|T_2| \leq (4q/t) \cdot M$ and $|T_1| \leq \Delta_2 \cdot M/(\delta q)$.*

Proof. Suppose we have $|T_2| > (4q/t) \cdot M$ at some point in the algorithm. Stop the algorithm when this happens (in Step I $6(d)$) and count the edges in L. Every time we increase T we add at least t edges to L. The only times we delete edges from L is when we remove some vertices from $A \setminus S$. But these vertices had by definition at most q neighbours in L. Hence, in total we remove at most $2q \cdot M$ edges from L, and so $e(L) > (4q/t) \cdot t \cdot M - 2q \cdot M = 2q \cdot M$. By Observation 0.2, we have $\Delta(L) \leq 2q$, so $q|V(L)| \geq e(L) > 2q \cdot M$, a contradiction.

If the algorithm stopped in Phase I then $T_1 = \emptyset$ and the claim follows. Otherwise, using that the multiplicity of an edge in L is at most Δ_2, every time we put a vertex into T_1 we removed at least $\delta q/\Delta_2$ vertices from A. Hence $|T_1| \leq \Delta_2 \cdot M/\delta q$ and the claim follows. □

Observation 0.4. *After the algorithm stops we have $|A| \leq (1 - 3\delta)M$ or $\mathcal{H}$ contains an (s, t)-core of size at least $(1 - 20\delta)M$.*

Proof. If the algorithm terminated in Phase I then it stopped in Step I(3) or in Step I(4), and the claim follows. Now assume the algorithm entered Phase II. Denote A_1, S_1, L_1 the sets A, S and graph L right before entering Phase II of the algorithm, and A_2 the set A at the end of the algorithm (noting that $A_2 \subseteq A_1$). Since we did not terminate after Phase I, we know that $|A_1| \geq (1 - 3\delta)M$.

Since $A_1 \setminus S_1$ is an (s, t)-core in $\mathcal{H}$ and we did not stop in Step I(4), we have

$$|A_1 \setminus S_1| \leq (1 - 20\delta)M, \tag{1}$$

The relation (1) implies that

$$\frac{|S_1|}{|A_1|} > 1 - \frac{(1 - 20\delta)M}{|A_1|}. \tag{2}$$

Since $|A_1| \leq M$, if it is the case that $|A_2| \leq (1 - 3\delta)|A_1|$ then we are done, we assume that $|A_2| > (1 - 3\delta)|A_1|$, implying

$$|A_1 \setminus A_2| < 3\delta|A_1|. \tag{3}$$

As in $L_1[A_2]$ every vertex has degree at most δq, we get

$$e(L_1[A_2]) \leq \frac{|A_2|\delta q}{2} \leq \frac{|A_1|\delta q}{2}. \tag{4}$$

Recall that in L_1, every vertex has degree at most $2q$. Hence, counting those edges in L_1 which have at least one endpoint in $A_1 \setminus A_2$ we get

$$e(L_1[A_1]) - e(L_1[A_2]) \leq |A_1 \setminus A_2|2q < 3\delta|A_1|2q. \tag{5}$$

Note also that in L_1, every vertex in S has degree at least q. As $e(L_1) = e(L_1[A_1])$, we have

$$e(L_1) \geq \frac{|S|q}{2}. \tag{6}$$

Putting the relations (4), (5) and (6) together we get

$$\frac{|S|q}{2} \leqslant e(L_1) = e(L_1(A_1)) \leqslant e(L_1(A_2)) + 6\delta|A_1|q < \frac{|A_1|\delta q}{2} + 6\delta|A_1|q. \tag{7}$$

Hence by (2) and (7)

$$17\delta < 1 - \frac{1-20\delta}{1-3\delta} \leqslant 1 - \frac{(1-20\delta)M}{|A_1|} < \frac{|S|}{|A_1|} < \frac{\delta q}{q} + 12\delta,$$

which contradicts the restriction on the input parameters. This completes the proof. □

Observation 0.5. *[Small Containers] After the algorithm stops we have* $|C| \leq (1-\delta)\, M$ *or it contains an* (s,t)*-core of size at least* $(1-20\delta)M$*.*

Proof. If the algorithm stopped after (during) Phase I then $|C| \leq |A| + |T_2|$ or it contains an (s,t)-core of size at least $(1-20\delta)M$, and if the algorithm stopped after Phase II we get $|C| \leq |A| + |T_2| + |T_1|$. In both cases, we have $|C| \leq (1 - 3\delta + 2q/t + \Delta_2/\delta q)M \leq (1-\delta)\, M$ by Observations 0.3 and 0.4. □

Putting all these observations together, we get the proof of Theorem 0.1. □

We will obtain our main container theorem by iterating Theorem 0.1. The number of iterations is at most $(1/\delta) \cdot \log M$ as in each round the size of the vertex sets shrinks with a factor of $(1-\delta)$; or we stop if a container C contains an (s,t)-core of size at least $(1-20\delta)|C|$. Note that in many of the applications the aim is to decrease the size of containers to be smaller than εM, in that case the number of iterations is only $(1/\delta) \cdot \log(1/\varepsilon)$.

Theorem 0.6. [Container theorem for 3-uniform hypergraphs]
Fix positive parameters $\delta, \Delta_2, s, t, M$ *such that* $2s \leqslant \delta t$ *and* $2\Delta_2 \leqslant \delta^3 t$ *are satisfied. Let* $q := \max\left\{s,\ \frac{\Delta_2}{\delta^2},\ \sqrt{\frac{\Delta_2 t}{2\delta}}\right\}$*. Let* $\mathcal{H}$ *be a 3-uniform* M*-vertex hypergraph* $\mathcal{H}$ *such that the maximum co-degree of* $\mathcal{H}$ *is at most* Δ_2*. Then there exists a family* $\mathcal{C} \subseteq \mathcal{P}(V(\mathcal{H}))$ *satisfying the following:*

(1) For every independent set $I \subseteq V(\mathcal{H})$*, there exists a* $C_I \in \mathcal{C}$ *such that* $I \subseteq C_I$*.*
(2) For every $C \in \mathcal{C}$ *there are* $T_2, T_1 \subset V(\mathcal{H})$ *with* $|T_1| \leqslant (1/\delta) \cdot \log M \cdot (\Delta_2/\delta q)) \cdot M$*,* $|T_2| \leqslant (1/\delta) \cdot \log M \cdot (2q/t)) \cdot M$ *such that* T_1, T_2 *determines* C*.*
(3) Every $C \in \mathcal{C}$ *contains an* (s,t)*-core of size at least* $(1-20\delta) \cdot |C|$*.* □

References

[1] J. Balogh, R. Morris, and W. Samotij, Independent sets in hypergraphs, *J. Amer. Math. Soc.*, (28) 2015, 669–709.
[2] D. Saxton and A. Thomason, Hypergraph containers, *Invent. Math.*, (201) 2015, 925–992.

Department of Mathematics, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA. Email: jobal@math.uiuc.edu.
Email address: jobal@math.uiuc.edu