# HYPERGRAPH CURVES

A Capstone Experience Manuscript

Presented by

Ilya Scheidwasser

Completion Date:
June 2009

Approved by:

Professor Jenia Tevelev, Mathematics

Professor Giancarlo Urzua, Mathematics

# 1. Abstract


Title: **Hypergraph Curves**
Author: **Ilya Scheidwasser, Computer Science**
CE Type: **Independent Capstone Thesis**
Approved By: **Jenia Tevelev, Mathematics Dept**
Approved By: **Giancarlo Urzua, Mathematics Dept**

I examine hypergraph curves as presented by Jenia Tevelev and Ana-Maria Castravet in the paper *Exceptional Loci on* $\overline{M}_{0,n}$ *and Hypergraph Curves*[1]. I present a combinatorial definition of hypergraphs and various conditions on them, several transformations on hypergraphs that preserve certain conditions, and a complete list of weak hypergraphs for $n \leq 9$ up to some transformations, along with the dual graphs of their stabilized versions. I also present a theorem about hypergraph stability, and an examination of two different Clifford Indices on various hypergraphs.

## 2. Introduction

I present the definition of a hypergraph curve and various basic properties on it which will be used throughout the paper.

**Definition 1.** *A hypergraph curve (or, simply, a hypergraph) is a set of abstract lines and points such that each point is contained in at least one line and each line contains at least three points. The number of points, $n$, must be equal to $\sum l_i - 2$ where $l_i$ is the number of points on some line $i$. I call this sum the "line sum" of the hypergraph, denoted by the function $lSum$. The genus of a hypergraph is given by $n - 3$. A point on a hypergraph is said to have a size $k$ when it has exactly $k$ lines containing it, and a line on a hypergraph is said to have a size $k$ when it has exactly $k$ points on it.*

**Definition 2.** *A hypergraph $X$ is weak when every subgraph (i.e., subset of lines) $Y$ of $X$ such that $1 < |Y| < |X|$ (where $|S|$ is the number of lines in $S$) has the property that $p(Y) \geq lSum(Y) + 2$.*

**Definition 3.** *A weak hypergraph $X$ is strong when it contains no weak subgraphs $Y$ such that $1 < |Y| < |X|$.*

**Definition 4.** *A hypergraph is planar when it can be mapped to the projective plane $\boldsymbol{P}^2$ in such a way that the graph lies on at least two separate lines, and no two points are mapped to the same point.*

**Definition 5.** *Stabilization is an operation performed on hypergraphs in the following way: given a point of size $k \geq 3$ with lines $b_1, b_2, ..., b_k$ containing it, we replace it with a line of size $k$ with exactly one $b_i$ intersecting it at each point. Doing this for every such point is called stabilization, and the resulting set of lines is called the stabilized version of the hypergraph.*

A hypergraph is stable when its stabilized version satisfies Gieseker's "basic inequality", described in the next section. Note that stabilization does not necessarily produce a stable hypergraph; in fact, it does not necessarily produce a hypergraph at all, since the number of points and the line count change. However, the stabilized version of a hypergraph has no points with size $> 2$.

**Definition 6.** *The dual graph of a hypergraph is defined so that every point in a dual graph corresponds to a line in the stabilized hypergraph, every edge in the dual graph corresponds to a point in the stabilized hypergraph, and two points in the dual graph are connected by an edge exactly when two lines in the stabilized hypergraph meet at a point.*

**Definition 7.** *The combinatorial Clifford index of a hypergraph, abbreviated by combCliff, is $f - 2$, where $f$ is the least number of edges one can remove from the hypergraph's dual graph to split it into two disconnected parts, each of which contains a cycle.*

The resolution Clifford index of a hypergraph, abbreviated by resCliff, is defined by Bayer and Eisenbud in [BE].

## 3. A Stability Theorem

I present a theorem about stability which is a generalization of a proposition in [CT]. The proposition is identical to (a) in this theorem, but it applies to hypergraphs with only point sizes 2 and 3 and only lines of size 3.

**Theorem 8.** *Given either (a) a stabilized weak hypergraph $X$ with $n$ points and with every point having at least two lines through it or (b) a stabilized strong hypergraph $X$ with $n$ points and with every point having at least one line through it, I show that the following equation holds $\forall Y \subset X$ with $1 < |Y| < |X|$:*

$$|n(Y) - n(X)\frac{m(X)}{m(Y)}| < \frac{1}{2}\#Y \tag{1}$$

*Here, $n(S)$ is the number of black lines in $S$. $m(S)$ is $\sum_{l_i \subset S}(|l_i| - 2) - p_1$, where $l$ are the lines (black or white) in $S$ and $p_1$ is the number of "loose points" (i.e. those with only one line through them) in $S$. $\#Y$ is $|Y \cap \overline{X \setminus Y}|$.*

*Proof.* First, I show that $m(X) = 2L - 4$, where $L$ is $n(X)$. Note that we can assume that $L > 2$, since $L = 1$ hypergraphs have no subsets and $L = 2$ hypergraphs cannot be weak when every point has at least two lines through it. We know:

$$m(X) = n - 2 + \sum(i-2)P_i \tag{2}$$

where $P_i$ is the number of points of size $i$ in the non-stabilized graph (correspondingly, for $i > 2$, the number of white lines of size $i$ in $X$). The quantity $n-2$ is the contribution from the black lines and $\sum(i-2)P_i$ is the contribution from the white lines plus the extra term $-p_1$. We also know:

$$\sum P_i = n \tag{3}$$

and:

$$\sum iP_i = \sum iL_i \tag{4}$$

where $L_i$ is the number of black lines of size $i$ in $X$, since the left hand side is the total number of times a point is hit by a line, and the right hand side is the total number of times that a line contains a point. We then combine (3) and (4) to get:

$$\sum(i-2)P_i = -2n + \sum iL_i \tag{5}$$

We know:

$$\sum(i-2)L_i = n - 2 \tag{6}$$

We then combine (2), (5), and (6) to get:

$$m(X) = n - 2 - 2n + n - 2 + 2L = 2L - 4 \tag{7}$$

We therefore need to show: $|n(Y) - L\frac{m(Y)}{2L-4}| < \frac{1}{2}\#Y$, i.e.

$$|(2L-4)b - Lm(Y)| < (L-2)\#Y \tag{8}$$

where $b$ is just $n(Y)$, the number of black lines in $Y$.

Next, we show that $Y^C$, the complement of $Y$, satisfies this inequality iff $Y$ does. First, consider the right hand side, $(L-2)\#Y$, i.e. $(L-2)|Y \cap \overline{X \setminus Y}|$. Clearly $L$ does not depend on $Y$, so we only need to consider $\#Y$. We know $Y = X \setminus Y^C$ and $X \setminus Y = Y^C$, so $\#Y = |Y \cap \overline{X \setminus Y}| = |\overline{X \setminus Y^C} \cap Y^C| = |Y^C \cap \overline{X \setminus Y^C}| = \#Y^C$. Now consider the left hand side: $|(2L-4)b - Lm(Y)|$. We know $b(Y^C) = L - b(Y)$ and $m(Y^C) = m(X) - m(Y) = 2L - 4 - m(Y)$. So

$$|(2L-4)b(Y^C) - Lm(Y^C)| =$$

$$|(2L-4)(L-b(Y))-L(2L-4-m(Y))| =$$
$$|2L^2-2Lb(Y)-4L+4b(Y)-2L^2+4L+Lm(Y)| =$$
$$|Lm(Y)-(2L-4)b(Y)| =$$
$$|(2L-4)b(Y)-Lm(Y)|.$$

Hence, by interchanging $Y$ with $Y^C$, we can assume that $Lm(Y)-(2L-4)b \geq 0$. We then need to show:

$$Lm(Y)-(2L-4)b-(L-2)\#Y < 0 \tag{9}$$

Consider every white line $w_1$ of arbitrary size $k$ not in $Y$, with at least one neighboring black line in $Y$. Enumerate the black lines in $Y$ touching $w_1$ as $b_1, b_2, ..., b_i$, and the rest as $b_{i+1}, ..., b_k$, with $1 \leq i \leq k$ (and $k \geq 3$). I show that adding every such $w_1$ to $Y$ does not decrease the value of the left hand side in (9), so showing that the inequality is satisfied in this case shows that it is satisfied in general. Clearly, this operation does not affect the values of $L$ or $b$, so we only need to show that the following does not decrease: $Lm(Y)-(L-2)\#Y$. Adding $w_1$ to $Y$ increases $m(Y)$ by $k-2$. If the original value of $\#Y$ is $x+i$, where $i$ is the contribution from $w_1$ intersecting $b_1$ through $b_i$, then the value after adding $w_1$ to $Y$ is $x+k-i$. Hence, $\#Y$ increases by $k-2i$.

Then $LHS_{new}-LHS_{old}$, the difference in values of the left hand side, is $L(k-2)-(L-2)(k-2i)$, and we need to show that this is $\geq 0$. So we need to show:

$$L(k-2)-(L-2)(k-2i) \geq 0, i.e.$$
$$Lk-2L-Lk+2Li+2k-4i \geq 0, i.e.$$
$$(L-2)i+k-L \geq 0.$$

Since $i \geq 1$ and $L > 2$, $(L-2)i-L \geq -2$, so we only need to show $k \geq 2$. But we already know that $k \geq 3$. Hence we can assume that all white lines hit by a black line in $Y$ are also in $Y$, and by showing (9) in this situation, we show (9) in general.

We know that $m(Y) = \sum(i-2)b_i + \sum(i-2)l_i - p_1$, where $b_i$ is the number of black lines in $Y$ with $i$ points and $l_i$ is the number of white lines in $Y$ with $i$ points. Let $f = \sum(i-2)b_i$ and $d = \sum(i-2)l_i - p_1$, so $m(Y) = f+d$. Define $l'_i$ to be the number of isolated white lines of size $i$ in $Y$ (i.e., those not hit by any black lines in $Y$) and $l'$ to be $\sum l'_i$. Also define $p_i$ to be the number of points of size $i$ hit by $Y$'s image in the non-stabilized hypergraph and $p$ to be $\sum p_i$. Then let $e = \sum(i-2)l'_i$ and $c = \sum ip_i$. Then we know:

$$d = e + c - 2p \tag{10}$$

since $e$ is the contribution to $d$ by isolated white lines in $Y$ and $c - 2p = \sum(i-2)p_i$ is the contribution to $d$ by white lines hit by $Y$'s black lines, which we can assume are all in $Y$, plus the extra term $-p_1$.

We also know:

$$\#Y = 2l' + e + c - 2b - f \tag{11}$$

$2l' + e$ is the contribution to $\#Y$ by isolated white lines in $Y$. The quantity $c$ is the total number of times a point in $Y$'s image in the non-stabilized hypergraph is hit by a line (not necessarily in $\#Y$) and $2b + f = \sum ib_i$ is the total number of times a black line in $Y$ hits one of these points, so their difference is the contribution to $\#Y$ by everything except isolated white lines.

We can therefore rewrite (9) as:

$$L(f+e+c-2p)-(2L-4)b-(L-2)(2l'+e+c-2b-f) < 0, i.e.$$
$$(2L-2)f+2e+2c-2Lp-(2L-4)l' < 0, i.e.$$
$$e-(L-2)l'+(L-1)f+c-Lp < 0 \tag{12}$$

Consider the first two terms of (12): $e - (L-2)l' = \sum -(L-i)l'_i$. I show that this is nonpositive by showing that every term is nonpositive. Suppose the $k^{th}$ term is positive: $-(L-k)l'_k > 0$. Then this means that $k - L > 0$. This is clearly impossible (as noted earlier), so $e - (L-2)l' \leq 0$, so we only need to show:

$$(L-1)f + c - Lp < 0, i.e.$$

$$Lp - c - (L-1)f > 0, i.e.$$

$$(L-3)p - (L-1)f + 3p - c > 0, i.e.$$

$$-\sum(i-3)p_i > (L-1)f - (L-3)p$$

The weak condition states that $p \geq f + 2$, so we need to show:

$$-\sum(i-3)p_i > (L-1)f - (L-3)(f+2), i.e.$$

$$-\sum(i-3)p_i > 2f - 2L + 6 \tag{13}$$

We know $\sum P_i = n$ and $\sum iP_i = 2L + F$, where $F$ is $\sum(i-2)L_i$. This is because $\sum iP_i$ is the total number of times a point is hit by a line on the non-stabilized hypergraph, and $2L + F = \sum iL_i$ is the sum of every line's number of points. Then we know $\sum(i-1)P_i = 2L + F - n$. Then $\sum(i-1)p_i \leq 2L + F - n$ since $p_i \leq P_i$. Then:

$$-\sum(i-1)p_i \geq -2L - F + n \tag{14}$$

We can rewrite (13) as:

$$-\sum(i-3)p_i > 2f + 6 + F - n + (-2L - F + n) \tag{15}$$

Suppose $Y$'s image on the non-stabilized hypergraph hits every point. Then (14) has equality. We therefore need to show:

$$2p > 2f + 6 + F - n$$

But in this case, we know that $p > f + 2$, since $p = n$, $n = F + 2$, and $f < F$. Then we need to show:

$$2f + 4 \geq 2f + 6 + F - n, i.e.$$

$$n \geq F + 2$$

Since we know $n = F + 2$, the inequality is satisfied.

Now suppose $Y$'s image on the non-stabilized hypergraph does not hit every point. In case (a), this means that (14) does not have equality. We therefore need to show:

$$2p \geq 2f + 6 + F - n$$

The weak condition tells us that $p \geq f + 2$, so we need to show:

$$2f + 4 \geq 2f + 6 + F - n, i.e.$$

$$n \geq F + 2$$

Again, since $n = F + 2$, the inequality is satisfied.

In case (b), if $Y$'s image on the non-stabilized hypergraph does not hit every non-loose point, then by the same argument, the inequality is satisfied. If, however, every non-loose point is hit but not every loose point is hit, then (14) has equality, so we need to show:

$$2p > 2f + 6 + F - n$$

Since the hypergraph is weak, we know that $p \geq f + 2$; however, since the hypergraph is strong, we also know that $p \neq f + 2$, since having $p = f + 2$ would imply that Y is a weak subgraph. Hence, $p > f + 2$, so the inequality is satisfied.

Remark: If the hypergraph is weak and contains loose points, then we can redefine $m(S)$ as $\sum_{l_i \subset S} (|l_i| - 2) - p_1 + \epsilon$, and the inequality holds for some small positive $\epsilon$ whose range of possible values is dependent on the hypergraph. □

## 4. Transformations and Enumeration of Weak Hypergraphs

I have written a program to generate all hypergraphs for some value of $n$, up to some transformations, and to test various conditions on them. The program's runtime increases considerably as $n$ does, as do the number of hypergraphs generated: The program generates hypergraphs for $n = 9$ in nine seconds and finds 41 hypergraphs, of which 27 are nonweak, 3 are strictly weak (i.e., weak but not strong), and 11 are strong. It generates hypergraphs for $n = 10$ in 25 seconds and finds 251 hypergraphs, of which 134 are nonweak, 24 are strictly weak, and 93 are strong. It generates hypergraphs for $n = 11$ in 9 minutes and 47 seconds and finds 2435 hypergraphs, of which 1164 are nonweak, 244 are strictly weak, and 1027 are strong.

The program generates hypergraphs by stepping through every possible collection of lines that satisfies the basic hypergraph conditions and checking whether the resulting hypergraph is identical to one already encountered, or can be ruled out because of certain transformations. I will first present the transformations used to narrow the list of hypergraphs and show that they preserve the weak condition and planarity; i.e., the original hypergraph has a given condition if and only if the transformed hypergraph does.

Transformation A starts with two points that are not connected by a line in a hypergraph $G1$ with $n$ points and $l$ lines, and produces a new triple (a line of size 3), $l_1$, which intersects both points and has a loose point. The resultant hypergraph is called $G2$.

**Theorem 9.** *Transformation A preserves the weak condition.*

*Proof.* First, I show that if $G1$ is weak, then $G2$ is weak.

Let $S2$ be a subgraph of $G2$ with $1 < |S2| < |G2| = l+1$, and let $S1$ be the corresponding subgraph in $G1$.

If $S2$ does not contain $l_1$ then it has the same number of points and the same line count as $S1$. If $S2$ has less than $l$ lines, it must satisfy the inequality $p(S2) \geq lsum(S2) + 2$ (the *weak inequality*), since $S1$ is not $G1$ and therefore satisfies the weak inequality. Otherwise, we know that $p(S2) = n = l + 2 = lsum(S2) + 2$, by the definition of a hypergraph, so it satisfies the weak inequality.

If $S2$ contains $l_1$, it has one more point than $S1$ and its line count is one higher. If $S2$ has more than two lines, it must satisfy the weak inequality because $S1$ has more than one line and therefore satisfies the weak inequality. Otherwise, $S1$ has one line, so we know that $p(S2) = p(S1) + 1 = lsum(S1) + 3 = lsum(S2) + 2$, by the definition of line sum, so it satisfies the weak inequality.

Hence, $S2$ always satisfies the weak inequality, so $G1$ being weak implies $G2$ being weak.

Next, I show that if $G2$ is weak, then $G1$ is weak. Let $S1$ be a subgraph of $G1$ with $1 < |S1| < |G1| = l$, and $S2$ be the corresponding subgraph of $G2$, so that $l_1$ is never in $S2$. Clearly, $S1$ always has the same number of points and the same line count as $S2$. Since $|S1| = |S2|$, we know that $1 < |S2| < |G1| < |G2|$, so $S2$ satisfies the weak inequality, so $S1$ does as well. Hence, $G2$ being weak implies $G1$ being weak and vice versa, so Transformation A preserves the weak condition. □

Note that applying A to a strong hypergraph always produces a strictly weak hypergraph, since the subgraph $S2$ of $G2$ corresponding to the entire graph $G1$ is a weak subgraph with the property $1 < |S2| = |G1| < |G2|$.

**Theorem 10.** *Transformation A preserves planarity.*

*Proof.* Suppose $G1$ is planar. Then some two lines $l_2$ and $l_3$ in $G1$ can be mapped to two different lines in $\mathbf{P}^2$. If $l_1$ hits two lines that were originally mapped to the same line, then $l_1$ maps to the same line. If $l_1$ hits two lines that mapped to different lines, these two lines can still be mapped to different lines. In either case, $l_2$ and $l_3$ can still be mapped to two different lines.

Suppose $G2$ is planar. Suppose that some two lines in $G2$ can be mapped to two different lines in $\mathbf{P}^2$, and that neither line is $l_1$. Then removing $l_1$ from $G2$ still allows this map.

Now suppose that one of the lines is $l_1$, and call the other $l_2$. Suppose that $l_1$ does not hit $l_2$; then call the lines that it hits $l_3$ and $l_4$. Then $l_3$ or $l_4$ must be mappable to a different line than $l_2$; if both had to map to the same line as $l_2$, then $l_1$ would also have to map to the same line as $l_2$. Then removing $l_1$ still lets one of these lines map to a different line than $l_2$.

If $l_1$ hits $l_2$, call the other line that it hits $l_3$. Then $l_3$ must map to a different line than $l_2$, since otherwise, $l_1$ and $l_2$ could not map to different lines. Then removing $l_1$ still allows these lines to map to different lines.

Hence, $G1$ is planar is $G2$ is planar and $G2$ is planar if $G1$ is planar, so Transformation A preserves planarity. □

Transformation B adds a loose point to any line $l_1$ of a hypergraph $G1$ to produce a new hypergraph $G2$.

**Theorem 11.** *Transformation B preserves the weak condition.*

*Proof.* Suppose $G1$ is weak. Let $S2$ be any subgraph of $G2$ with $1 < |S2| < |G2|$. Then the corresponding subgraph $S1$ has the property $1 < |S1| < |G1|$. $S2$ has one more point and one higher line count than $S1$, so $S2$ must satisfy the weak inequality since $S1$ does.

Suppose $G2$ is weak. Let $S1$ be any subgraph of $G1$ with $1 < |S1| < |G1|$. Then the corresponding subgraph $S2$ has the property $1 < |S2| < |G2|$. $S2$ has one more point and one higher line count than $S1$, so $S1$ must satisfy the weak inequality since $S2$ does. □

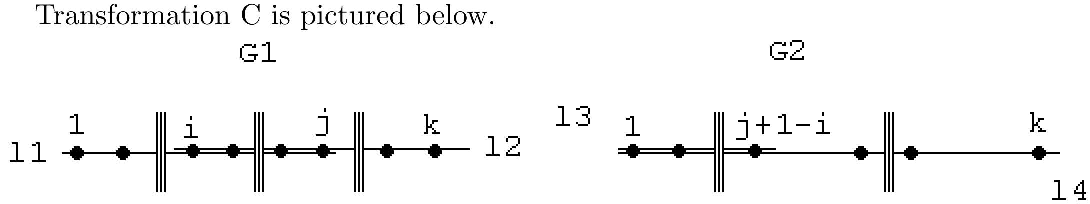
**Theorem 12.** *Transformation B preserves the strong condition.*

*Proof.* Suppose $G1$ is strong. Then $G2$ is weak, since B preserves the weak condition. Let $S2$ be any subgraph of $G2$ with $1 < |S2| < |G2|$. Then the corresponding subgraph $S1$ has the property $1 < |S1| < |G1|$. $S2$ has one more point and one higher line count than $S1$, so $S2$ cannot be a weak subgraph since $S1$ is not a weak subgraph.

Suppose $G2$ is strong. Then $G1$ is weak, since B preserves the weak condition. Let $S1$ be any subgraph of $G1$ with $1 < |S1| < |G1|$. Then the corresponding subgraph $S2$ has the property $1 < |S2| < |G2|$. $S2$ has one more point and one higher line count than $S1$, so $S1$ cannot be a weak subgraph since $S2$ is not a weak subgraph. □

We can easily see that Transformation B preserves planarity, since no lines are added or removed.

Transformation C is pictured below.




Transformation C starts with two lines $l_1$ and $l_2$ in a hypergraph $G1$ that intersect at at least two points, and which each have at least one point not shared with the other line. The points are numbered so that $l_1$ contains points 1 through $j$ and $l_2$ contains points $i$ through $k$, with $1 < i < j < k$, so that $l_1$ has $j$ points, $l_2$ has $k-i+1$ points, and their intersection has $j-i+1$ points.

Transformation C then produces a new hypergraph $G2$ in which the number of points is the same, but the lines are replaced with two new lines $l_3$ and $l_4$, such that $l_4$ contains all $k$ points and $l_3$ contains points 1 through $j-i+1$ points, i.e. the same number of points as the intersection of $l_1$ and $l_2$ in $G1$. If $l_1$ and $l_2$ intersect at only two points (the minimum for this transformation), then $l_3$ does not exist.

The line count of $l_1$ is $j-2$, the line count of $l_2$ is $k-i-1$, the line count of $l_3$ is $j-i-1$, and the line count of $l_4$ is $k-2$. Hence $G1$ and $G2$ have the same number of points and the same line count $(k+j-i-3)$, so Transformation C always produces a valid hypergraph.

**Theorem 13.** *Transformation C preserves the weak condition.*

*Proof.* First, I show that $G2$ is weak if $G1$ is weak.

Note that both graphs are nonweak if $l_1$ and $l_2$ intersect at more than two points: the number of points in the subset of $G1$ formed by $l_1$ and $l_2$ is $k$, while the line count is $k+j-i-3 = (k-2)+(j-1+i)$; since the size of the intersection, $j-1+i$, is at least 3, the subset violates the weak inequality. Since the subset of $G2$ formed by $l_3$ and $l_4$ has the same number of points and the same line count, it also violates the weak inequality. Hence, we only need to consider the case in which $j = i+1$ and $l_3$ does not exist.

Let $S2$ be a subgraph of $G2$ such that $1 < |S2| < |G2|$. Suppose $S2$ does not contain $l_4$. Then the corresponding subgraph $S1$ in $G1$ has the same number of points and the same number of lines as $S2$, and has the property that $1 < |S1| < |G1|$, so $S2$ satisfies the weak inequality.

Suppose $S2$ conatins $l_4$. Then the corresponding hypergraph $S1$ in $G1$ that contains $l_1$ and $l_2$ has the same number of points and the same line count $S2$, and has the property that $1 < |S1| < |G1|$, so $S2$ satisfies the weak inequality.

Next, I show that $G1$ is weak if $G2$ is weak.

Let $S1$ be a subgraph of $G1$ such that $1 < |S1| < |G1|$. Suppose $S1$ contains both $l_1$ and $l_2$. Then the corresponding subgraph $S2$ of $G2$ which contains $l_4$ has the same number of points and the same line count, and satisfies $1 \leq |S2| < |G2|$. Since a single line satisfies the weak inequality and $G2$ is weak, $S1$ satisfies the weak inequality.

Suppose $S1$ contains neither $l_1$ nor $l_2$. Then the corresponding subgraph $S2$ of $G2$ which does not contain $l_4$ has the same number of points and the same line count and satisfies $1 < |S2| < |G2|$, so $S1$ satisfies the weak inequality.

Suppose $S1$ contains $l_1$ but not $l_2$. Then the corresponding subgraph $S2$ of $G2$ that contains $l_4$. $S2$ has $a+k$ points, where $k$ is the number of points in $S2$ not on $l_4$. It has a line count of $k-2+b$, where $b$ is the line count due to lines in $S2$ other than $l_4$. Since $G2$ is a weak hypergraph and $1 < |S2| \leq |G2|$, we know that it satisfies the weak inequality: $a+k \geq k+b$, i.e. $a \geq b$. $S1$ contains $a+j+c$ points, where $c$ is the number of points numbered $j+1$ through $k$ on the subgraph. Its line count is $j-2+b$. Then the weak inequality for $S1$ is $a+j+c \geq j+b$, i.e. $a+c \geq b$. Since $a \geq b$, $a+c \geq b$, so $S1$ satisfies the weak inequality.

Suppose $S1$ contains $l_2$ but not $l_1$. Then this is essentially equivalent to the previous case, since there is no real distinction between $l_1$ and $l_2$. Hence, $S1$ satisfies the weak inequality.

$G2$ is weak if $G1$ is weak and $G1$ is weak if $G2$ is weak, so Transformation C preserves the weak condition. □

If $l_1$ and $l_2$ intersect at more than two points, then $G1$ is nonweak. If $l_1$ and $l_2$ intersect at exactly two points and $G1$ contains at least one other line, then the subgraph composed of $l_1$ and $l_2$ is a weak subgraph. Hence, the only case in which $G1$ is strong is when $l_1$ and

$l_2$ intersect at exactly two points and are the only lines in the hypergraph. We then know that $G2$ is strong if $G1$ is strong, since a single line is a strong hypergraph. $G1$ is only strong when $G2$ is a single line.

Transformation C clearly preserves planarity because $l_1$ and $l_2$ must map to the same line, and $l_3$ and $l_4$ must map to the same line.

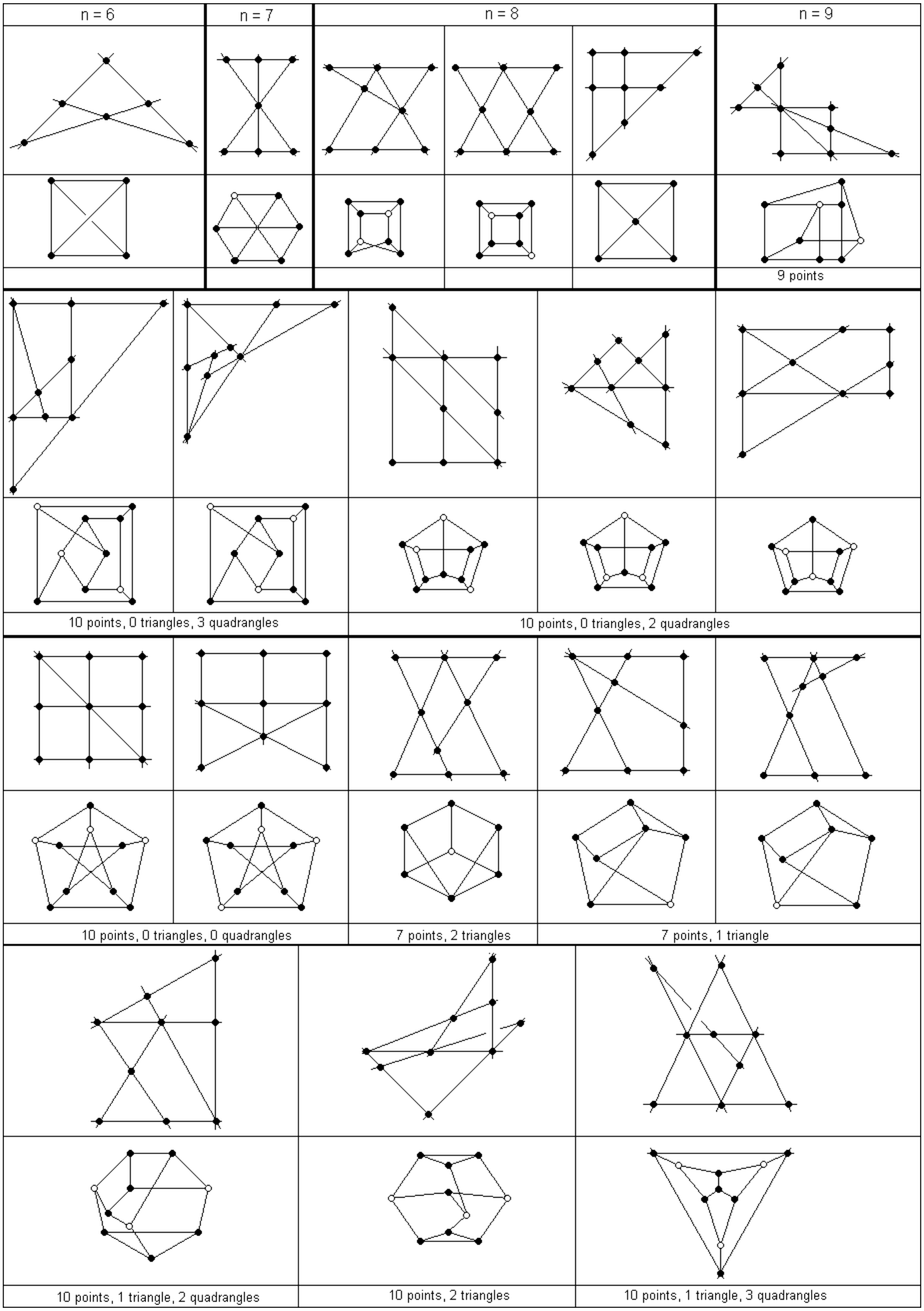
I use these transformations to narrow my list of weak hypergraphs. I do this by establishing one of the hypergraphs $G1$ or $G2$ in each transformation as the 'simpler' hypergraph, and then omitting every hypergraph that looks like the more 'complex' version. In particular:

For Transformation A, I consider $G1$ to be the simpler hypergraph. Hence, I omit all hypergraphs that contain a triple with exactly one loose point. Any hypergraph with such a line can be transformed through repeated applications of A to one without such a line, preserving the weak condition and planarity, and possibly producing a strong hypergraph from a weak one (as discussed earlier).

For Transformation B, I consider $G1$ to be the simpler hypergraph. Hence, I omit all hypergraphs that contain a line of size at least 4 with at least one loose point. Any such hypergraph can be transformed through repeated applications of B to one without such a line, preserving the weak condition, the strong condition, and planarity.

For Transformation C, I consider $G2$ to be the simpler hypergraph. Hence, I omit all hypergraphs that contain two lines intersecting at at least two points where one line is not contained in the other. Any such hypergraph can be transformed through repeated applications of C to one without such a pair of lines, preserving the weak condition and planarity, and sometimes producing a strong hypergraph (a single line) from a weak one.

A complete list of weak hypergraphs for $n \leq 9$ up to these transformations, as well as the dual graphs of their (not pictured) stabilized versions, is below:


n = 6
n = 7
n = 8
n = 9
9 points
10 points, 0 triangles, 3 quadrangles
10 points, 0 triangles, 2 quadrangles
10 points, 0 triangles, 0 quadrangles
7 points, 2 triangles
7 points, 1 triangle
10 points, 1 triangle, 2 quadrangles
10 points, 2 triangles
10 points, 1 triangle, 3 quadrangles

I include hypergraphs only up to $n = 9$ because drawing hypergraphs and dual graphs in an aesthetically appealing way given output from the program is very time-consuming as $n$ increases; also, higher values of $n$ yield many more hypergraphs, as has been shown.

Not pictured are the strong hypergraphs composed of single lines. The last three hypergraphs are the only strictly weak hypergraphs; note that all three contain at least one copy of the $n = 6$ strong hypergraph. Black points on dual graphs correspond to lines on the original hypergraphs, while white points correspond to inserted lines on the stabilized hypergraphs, i.e. points on the original hypergraphs with size at least 3. Each dual graph for $n = 9$ includes a justification for its uniqueness from the other dual graphs; the earlier dual graphs are simple enough that such justification is not necessary. In particular, these justifications list the number of points in the dual graph and the number of polygons of small size in the dual graph.

In order to run efficiently, my program relies on the idea of 'symmetries' within hypergraphs. A symmetry is a way of classifying a hypergraph to easily distinguish it from many (though not all) other hypergraphs. This is necessary because one of the central problems with a hypergraph generator is determining whether two hypergraphs are identical. Hypergraphs in my program are represented as arrays of lines, and each line is an array of points. Hence, the strong $n = 6$ hypergraph may be represented in the following way:

0 1 2
0 3 4
1 3 5
2 4 6

When generating new hypergraphs, the program needs to test whether each hypergraph has already been encountered and stored by comparing it with every hypergraph already encountered. Simply testing whether two hypergraphs are represented the same way does not determine whether they are identical since a single hypergraph can be represented in many different ways. One way to test whether two hypergraphs $G1$ and $G2$ are identical is to permute the points of $G1$ in every possible way, and then to check whether their lines are identical. However, this is extremely time consuming, as the number of permutations of $n$ points is $n!$, and such comparisons need to be made for every stored hypergraph every time a (potentially) new hypergraph is constructed.

To drastically cut down on the number of permutations tested, my program computes and stores every hypergraph's symmetries. A symmetry is computed using the following recursive algorithm:

First, each of the hypergraph's points is labeled with a number corresponding to its size. Hence, a hypergraph with points of size 1, 2, and 5 would have every size-1 point labeled '0', every size-2 point labeled '1', and every size-5 point labeled '2'.

Then, each point is given a list of lists of adjacent points' labels. I.e., if a point is on three lines, one of which contains (other than the point itself) a points labeled '0' and '1', another of which contains points labeled '1' and '1', and a third of which contains points labeled '0', '0', and '2', then it is given the following list of lists:

0 1
1 1
0 0 2

The top-level list stores three lists; each lower-level list represents one of the lines that the point lies on; and each entry in a lower-level list contains the labels of the points on that line. Each lower-level list is sorted, and the higher-level lists are sorted first by size and then by content. Hence, the list '0 0 2' is last because it has the highest size; the list '0 1' is before the list '1 1' because it comes before it in lexicographical order. This particular list would belong to a point of size 3 (since it lies on three different lines) which lies upon two triples and one line of size 4 (since it has two neighbors on two lines and three neighbors on one line).

After every point has such an entry, points are relabeled in the following way: for every original label, the points are given a new label, such that their new labels order them according to their lists, but the orders between different original labels stay the same. Suppose, for instance, that the original labels were '0', '1', and '2', and that the points in label '0' had two different lists among them, but the points in label '1' all had the same list, as did the points in label '2'. Then the points originally labeled '0' would be relabeled '0' and '1', so that '0' corresponded to the lower list (in lexicographical order) and '1' corresponded to the higher list. Then the points originally labeled '1' and '2' would be relabeled '2' and '3', respectively. In this way, every point originally labeled '0' still has a lower label than every point originally labeled '1' or '2', but a new distinction between points originally labeled '0' has been added.

At this point, the process repeats; every point is given a list of lists corresponding to its neighbors' labels, and points are then relabeled. This continues until the relabeling does not change any labels; this happens when all the points within every given label have the same list of lists. This end result, a description of each point's label along with each label's corresponding list of lists, is stored as the 'symmetry' of the hypergraph. This data is called a 'symmetry' because two points within the same list are in some sense identical; they have the same size, lie on lines of the same size, are adjacent to points of the same size that lie on lines of the same size, and so on. The process of finding a symmetry must always terminate eventually, since every step divides points in a given label into several separate labels; once every point has its own label, the process must end (though the process often ends before this happens).

Clearly, the same hypergraph represented in two different ways will have the same symmetry, since symmetries do not depend on point numberings. Moreover, if two hypergraphs have different symmetries, then they clearly cannot be the same hypergraph, since symmetries represent concrete data about the hypergraph's structure.

When comparing a newly-constructed hypergraph to a stored hypergraph, my program will first calculate the new hypergraph's symmetry, and then compare it to the stored symmetry of the stored hypergraph. If the two symmetries are different, then the hypergraphs are different. If the two symmetries are the same, then the program permutes points to see whether the hypergraphs are identical. However, every point does not need to be permuted with every other point; points only need to be permuted when they have the same label. This cuts down on permutation time significantly.

In fact, symmetries are used in intermediate steps of hypergraph generation to further speed up the program. Hypergraphs are not generated all at once; rather, they are generated line by line. I.e., the program will first generate every possible single line within $n$ points, up to symmetry; this just ends up being one line for each size from 3 to $n$. It then generates every possible set of two lines within $n$ points, up to symmetry, and throws out the previous results. It continues in this way, adding single lines, until it reaches the maximum amount of lines, $n-2$. At every step of the way, it checks whether each resulting set of lines qualifies as a hypergraph by checking whether every point is hit and whether the line count is $n-2$.

Transformation C is also used to throw out some sets of lines during intermediate steps; any time two lines are encountered that intersect at at least two points, and one line is not contained in the other, the line set is thrown out. Transformations A and B can only be checked at the end, when a complete hypergraph has been created, since they rely on loose points; a loose point in an intermediate set of lines may increase in size later on.

Another transformation that preserves some conditions (but which was not used for graph simplification due to time constraints) is called Transformation D. Transformation D begins with a hypergraph $G1$ with a line $l_1$ and a nonzero set of lines $l_2$ all intersecting $l_1$ at at the same point; each line in this set is given an index from 1 to $j$, so that $j \geq 1$. Some amount $i$ of the $l_2$ lines with $0 \leq i < j$ is kept at the point, while the rest of the lines are moved to a new point on $l_1$. Hence, no new lines are created, but $l_1$ is incremented in size. The resulting hypergraph is called $G2$ (the result is indeed a hypergraph because the number of points and the line count are both incremented). Note that when $i = 0$, this transformation is identical to Transformation B; however, this transformation does not preserve the same conditions as Transformation B and is not used to simplify the list of hypergraphs, so it is mentioned separately.

**Theorem 14.** *If $G1$ is weak in Transformation D, then $G2$ is weak.*

*Proof.* Let $S2$ be a subgraph of $G2$ with $1 < |S2| < |G2|$. Let $S1$ be the corresponding subgraph of $G1$. Then we know $1 < |S1| = |S2| < |G1| = |G2|$.

Suppose $S2$ does not contain $l_1$, and contains either some $l_2k$ lines with $k \leq i$ or some $l_2k$ lines with $k > i$, or neither (but not both). Then $S2$ has the same line count and the same number of points as $S1$, so $S2$ satisfies the weak inequality.

Suppose $S2$ contains $l_1$. Then $S2$ has one more point and one higher line count than $S1$, so $S2$ satisfies the weak inequality.

Suppose $S2$ does not contain $l_1$, and contains some $l_2k$ lines with $k \leq i$ and some $l_2k$ lines with $k > i$. Then $S2$ has the same line count as $S1$ and has one more point than $S1$, so it satisfies the weak inequality.

Since these are all the cases, $S2$ always satisfies the weak inequality, so $G2$ is weak. $\square$

**Theorem 15.** *If $G1$ is strong in Transformation D, then $G2$ is strong.*

*Proof.* Let $S2$ be a subgraph of $G2$ with $1 < |S2| < |G2|$. Since $G1$ is weak, $G2$ is weak, so $S2$ satisfies the weak inequality. Let $S1$ be the corresponding subgraph of $G1$.

Suppose $S2$ does not contain $l_1$, and contains either some $l_2k$ lines with $k \leq i$ or some $l_2k$ lines with $k > i$, or neither (but not both). Then $S2$ has the same line count and the same number of points as $S1$, so $S2$ is not a weak subgraph.

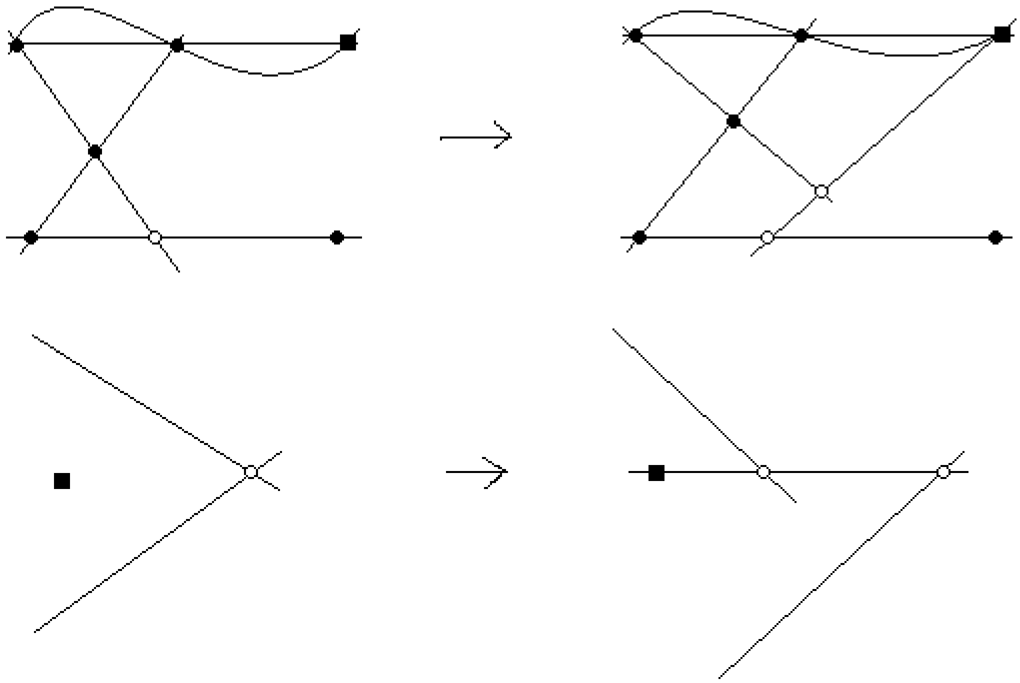
Suppose $S2$ contains $l_1$. Then $S2$ has one more point and one higher line count than $S1$, so $S2$ is not a weak subgraph.

Suppose $S2$ does not contain $l_1$, and contains some $l_2k$ lines with $k \leq i$ and some $l_2k$ lines with $k > i$. Then $S2$ has the same line count as $S1$ and has one more point than $S1$. If $S2$ is a weak subgraph, then it has $m$ points and a line count of $m - 2$ for some $m$; then $S1$ has $m - 1$ points and a line count of $m - 2$, and therefore does not satisfy the weak inequality. This is a contradiction, so $S2$ is not a weak subgraph.

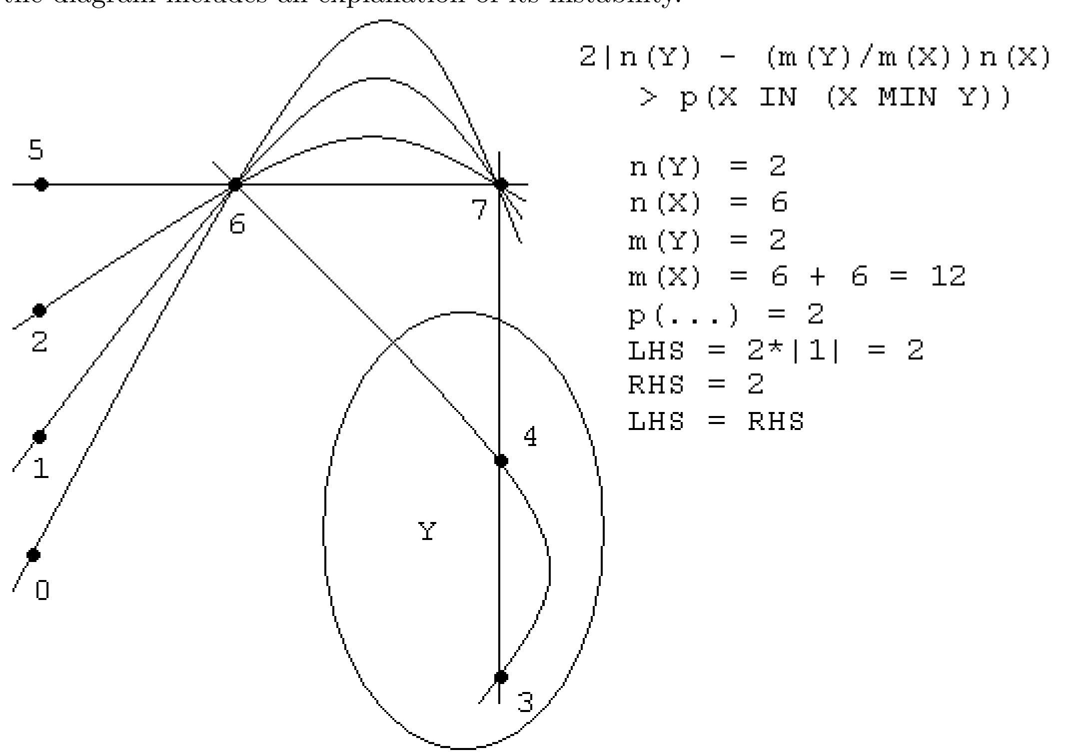
Since these are all the cases, $S2$ is never a weak subgraph, so $G2$ is strong. $\square$

Transformation D could be used to further narrow the list of hypergraphs, as well as to considerably speed up the program's runtime, by throwing out any hypergraphs or intermediate line sets containing points of size greater than 2, since any point with a higher size can be split into two points with D. Note that all hypergraphs would then be already stabilized.

The *Fibonacci-like construction* in [CT] can be shown to break the stability condition; i.e., a hypergraph that is initially stable may lose stability after the construction is applied to it. The Fibonacci-like construction begins with a hypergraph containing a point of size 2, $p_1$, which is contained in lines $l_1$ and $l_2$. Another point $p_2$ is then chosen which is not contained in $l_1$ or $l_2$. The lines $l_1$ and $l_2$ are then disconnected, and a new triple is created which intersects both lines at new points and also contains $p_2$. This construction was shown in [CT] to produce weak hypergraphs when given a weak hypergraph. The following diagram shows an example of the Fibonacci-like construction producing an unstable hypergraph when given a stable hypergraph. Below the two hypergraphs, the diagram shows the construction in general.



Not all weak hypergraphs are stable. The following hypergraph is unstable and weak; the diagram includes an explanation of its instability.




The subset Y examined consists of the lines 3-4-7 and 3-4-6. The stability condition is written to the right of the hypergraph, and the left hand side (LHS) of the inequality is shown to be equal to the right hand side (RHS), violating the condition. The expression p(X IN (X MIN Y)) is #Y.

## 5. Clifford Indices

I have found the resolution Clifford indices and the combinatorial Clifford indices of several hypergraphs from my list of weak hypergraphs. These indices are of interest to Green's conjecture.

Combinatorial Clifford indices were found by examining the dual graphs of the stabilized hypergraphs. Resolution Clifford indices were found using the well-known program Macaulay [Mac]: the canonical embedding was found for the stabilized hypergraph, then Macaulay was used to intersect the ideals corresponding to each line on the stabilized hypergraph, and then Macaulay was used to find the resolution Clifford index. We know that a stabilized hypergraph of size $n$ composed of triples (i.e., the original hypergraph is composed of triples and has no points of size greater than 3) with a 3-connected dual graph can be canonically embedded in $\mathbf{P}^{n-4}$, the projective space of dimension $n-4$, as a set of lines. The canonical embedding is discussed in pages 134 and 135 of [M]. The ideals of the individual lines were found by hand, and Macaulay was used to intersect them to find the ideals of the entire hypergraph.

The canonical embeddings were found using another program I wrote. This program finds canonical embeddings for hypergraphs whose stabilized versions have only triples. It is given the hypergraph and an order in which to process the lines; it then iterates through the lines in this given order and assigns weights to points according to the following rules: If a point on this line already has a weight on the other line it lies on, then assign it the negation of the weight. If two points on this line have weights on this line, then assign the third point a weight equal to the negation of their sum, so that the total weight is 0. Otherwise, add a new weight to a point on the line so that it is zero for every variable except a new, unused variable, which gets weight 1. Hence, for example, the first line visited will always get weights [1,0,0,...], [0,1,0,...], [-1,-1,0,...]; the first point gets the first variable, the second point gets the second variable, and the third point gets the negative sum of the first two. The order of the lines needs to be assigned in such a way that every line except the first line hits some previous line. Hence, in a properly-ordered hypergraph, the second line visited (if it hits the first line at the first point) will have weights [-1,0,0,...], [0,0,1,...], [1,0,-1,...].

The resolution Clifford index is relevant to Green's conjecture, mentioned in [BE]; the combinatorial and resolution Clifford indices are both relevant to Bayer and Eisenbud's "Combinatorial Clifford Index Conjecture", from [BE]. This latter conjecture states that if the resolution Clifford index $Cliff$ of a trivalent 3-connected graph $G$ of genus $g$ satisfies the inequality $CliffG < [(g-1)/2]$, then $CliffG$ is equal to the combinatorial Clifford index of G.

The following is a list of line ideals, hypergraph ideals, and Clifford indices for each hypergraph. The line ideals are first, followed by the single hypergraph ideal. Hypergraphs are numbered according to their order from left to right and top to bottom in the list of weak hypergraphs, from 1 to 19. Note that every hypergraph listed below has a dual graph which is trivalent and 3-connected; the dual graphs are trivalent since the hypergraphs only have triples and points of size less than three, and the dual graphs have been found to be 3-connected through examination. The coordinates range from $x_1$ to $x_{n-3}$.

3:

$x_3$, $x_4$, $x_5$

$x_2$, $x_4$, $x_5$

$x_1,\, x_2,\, x_5$
$x_1,\, x_2,\, x_3$
$x_3,\, x_4,\, x_1 - x_2$
$x_1 - x_2,\, x_3 + x_4,\, x_1 + x_5$
$x_1 + x_3,\, x_1 - x_4,\, x_1 + x_5$
$x_2,\, x_1 + x_3,\, x_4 + x_5$
$x_2x_4 - x_1x_5 + x_2x_5 - x_3x_5,\, x_1x_4 - x_3x_5, x_2x_3 + x_1x_5 - x_2x_5 + x_3x_5$
Resolution Clifford index 2, combinatorial Clifford index 2

4:
$x_3,\, x_4,\, x_5$
$x_2,\, x_4,\, x_5$
$x_1,\, x_3,\, x_5$
$x_1,\, x_2,\, x_5$
$x_2,\, x_4,\, x_1 - x_3$
$x_2,\, x_1 + x_5,\, x_1 + x_4 - x_3$
$x_3,\, x_4,\, x_1 - x_2$
$x_3,\, x_1 + x_5,\, x_1 + x_4 - x_2$
$x_1x_5 - x_2x_5 - x_3x_5 + x_4x_5,\, x_1x_4 + x_4x_5, x_2x_3$
Resolution Clifford index 2, combinatorial Clifford index 2

7:
$x_3,\, x_4,\, x_5,\, x_6$
$x_4,\, x_5,\, x_6,\, x_1 - x_2$
$x_5,\, x_6,\, x_1 - x_2,\, x_1 - x_3$
$x_2,\, x_3,\, x_5,\, x_6$
$x_2,\, x_3,\, x_6,\, x_1 - x_4$
$x_6,\, x_1 - x_2,\, x_1 - x_3,\, x_1 - x_4$
$x_1 - x_2,\, x_1 - x_3,\, x_1 - x_4,\, x_1 + x_5$
$x_1,\, x_3,\, x_4,\, x_5$
$x_3,\, x_1 - x_4,\, x_1 + x_5,\, x_2 - x_6$
$x_1 - x_2,\, x_1 - x_4,\, x_1 + x_5,\, x_1 - x_6$
$x_4x_6 + x_5x_6,\, x_1x_6 + x_5x_6,\, x_2x_5 - x_3x_5 - x_3x_6 - x_5x_6,\, x_1x_5 - x_4x_5,\, x_2x_4 - x_3x_4 + x_3x_6 + x_5x_6,$
$x_1x_3 - x_2x_3$
Resolution Clifford index 2, combinatorial Clifford index 2

10:
$x_3,\, x_4,\, x_5,\, x_6$
$x_2,\, x_4,\, x_5,\, x_6$
$x_1,\, x_2,\, x_5,\, x_6$
$x_1,\, x_2,\, x_3,\, x_6$
$x_1,\, x_3,\, x_4,\, x_6$
$x_1,\, x_3,\, x_4,\, x_2 - x_5$
$x_2,\, x_4,\, x_5,\, x_1 + x_3$
$x_2,\, x_1 + x_3,\, x_4 + x_5,\, x_3 + x_6$
$x_1 + x_3,\, x_3 + x_4,\, x_5 + x_6,\, x_1 - x_2 + x_5$
$x_5,\, x_6,\, x_1 - x_2,\, x_3 + x_4$
$x_2x_6 - x_4x_6 - x_5x_6,\, x_1x_6 + x_3x_6,\, x_3x_5 - x_4x_6,\, x_1x_5 + x_4x_6,\, x_1x_4 - x_2x_4 - x_4x_6,\, x_2x_3 + x_2x_4$
Resolution Clifford index 2, combinatorial Clifford index 2

12:
$x_3, x_4, x_5, x_6$
$x_2, x_4, x_5, x_6$
$x_1, x_2, x_5, x_6$
$x_1, x_2, x_3, x_6$
$x_1, x_2, x_3, x_4$
$x_1, x_3, x_4, x_5$
$x_1, x_5, x_3+x_4, x_2-x_6$
$x_3, x_6, x_1-x_2, x_4+x_5$
$x_1-x_5, x_2+x_4, x_3-x_6, x_1-x_2+x_3$
$x_2, x_4, x_1+x_3, x_5+x_6$
$x_1x_6+x_3x_6+x_4x_6,\ x_3x_5+x_3x_6+x_4x_6,\ x_1x_5-x_2x_5-x_3x_6-x_4x_6,\ x_2x_4+x_2x_5-x_4x_6,$
$x_1x_4+x_2x_5,\ x_2x_3+x_4x_6$
Resolution Clifford index 3, combinatorial Clifford index 2

17:
$x_3, x_4, x_5, x_6$
$x_2, x_4, x_5, x_6$
$x_2, x_5, x_6, x_1+x_3$
$x_2, x_6, x_1+x_3, x_1+x_4$
$x_1, x_2, x_3, x_4$
$x_2, x_6, x_1+x_4, x_1+x_5$
$x_1, x_3, x_5, x_6$
$x_3, x_4, x_5, x_1-x_2$
$x_3, x_6, x_1+x_5, x_1-x_2+x_4$
$x_3, x_4, x_1-x_2, x_1+x_5+x_6$
$x_4x_6,\ x_3x_6,\ x_1x_6-x_2x_6,\ x_1x_5-x_2x_5+x_4x_5,\ x_1x_4+x_3x_4-x_3x_5+x_4x_5,\ x_2x_3,\ x_2^2x_5+$
$x_3x_4x_5-x_4^2x_5+x_2x_5^2-x_3x_5^2+x_4x_5^2+x_2x_5x_6$
Resolution Clifford index 1, combinatorial Clifford index 1

18:
$x_3, x_4, x_5, x_6$
$x_2, x_4, x_5, x_6$
$x_1, x_2, x_5, x_6$
$x_1, x_2, x_3, x_6$
$x_1, x_2, x_3, x_4+x_5$
$x_1, x_2, x_3, x_4$
$x_3, x_4, x_1-x_2, x_5-x_6$
$x_1-x_2, x_3+x_4, x_5-x_6, x_1+x_5$
$x_1-x_2, x_1+x_3, x_4+x_5, x_4+x_6$
$x_4, x_5, x_6, x_1+x_3$
$x_1x_6-x_2x_6,\ x_3x_5-x_3x_6,\ x_2x_5-x_2x_6,\ x_1x_5-x_2x_6,\ x_2x_4-x_3x_6,\ x_1x_4-x_3x_6,\ x_3x_4x_6+$
$x_4^2x_6+x_4x_5x_6+x_3x_6^2,\ x_2x_3x_6+x_3^2x_6-x_4^2x_6-x_4x_5x_6,\ x_1x_2x_3+x_2x_3^2+x_4^2x_6+x_4x_5x_6$
Resolution Clifford index 1, combinatorial Clifford index 1

19:
$x_3, x_4, x_5, x_6$
$x_2, x_4, x_5, x_6$
$x_1, x_4, x_5, x_6$
$x_1, x_5, x_6, x_2-x_3$

$x_1$, $x_6$, $x_2 - x_3$, $x_2 + x_4$
$x_3$, $x_4$, $x_6$, $x_1 - x_2$
$x_3$, $x_4$, $x_1 - x_2$, $x_1 + x_5$
$x_4$, $x_2 + x_5$, $x_2 - x_6$, $x_1 - x_2 + x_3$
$x_1$, $x_2 - x_3$, $x_2 + x_5$, $x_2 - x_6$
$x_1$, $x_2 - x_3$, $x_2 + x_4$, $x_2 + x_5$
$x_2x_6 + x_5x_6$, $x_1x_6 + x_3x_6 + x_5x_6$, $x_3x_5 + x_4x_5 + x_3x_6 + x_4x_6$, $x_1x_5 - x_2x_5 - x_4x_5 - x_3x_6 - x_4x_6$, $x_2x_4 - x_3x_4$, $x_1x_4$, $x_1x_2x_3 + x_3^2x_6 - x_4x_5x_6 - x_3x_6^2 - x_4x_6^2$
Resolution Clifford index 1, combinatorial Clifford index 1

Note that when multiple hypergraphs have the same dual graph (for their stabilized versions), only one hypergraph is listed, since they will necessarily have the same ideals and Clifford indices.

The first two hypergraphs listed, 3 and 4, have 8 points and hence have genus 5; therefore, $[(g-1)/2] = 2 = CliffG$, so the Combinatorial Clifford Index Conjecture is supported. Hypergraphs 7, 10, and 12 have 9 points and genus 6, so $[(g-1)/2] = 2$; their resolution Clifford indices are all greater than or equal to 2, so the conjecture is supported. Hypergraphs 17, 18, and 19 also have genus 6, but their resolution Clifford indices are less than 2. However, their combinatorial Clifford indices are all equal to their resolution Clifford indices, so the conjecture is again supported.

## 6. Future Research

The most obvious possibility for future research is to examine hypergraphs for higher values of $n$. By implementing Transformation D to remove all hypergraphs with points of size greater than 2, the program could be sped up considerably. The resolution and combinatorial Clifford indices could then be examined for these hypergraphs.

In order to do this, however, the canonical embedding program would need to be modified to work with stabilized hypergraphs with non-triple lines. This would be a relatively trivial modification. It would also be helpful to automate the finding of combinatorial Clifford indices, as well as the determination of line ideals from the canonical embedding program's output weights. Both processes were relatively easy to do by hand, but would become more time-consuming for larger hypergraphs. If both processes were automated, then the entire process of generating a hypergraph, stabilizing it, finding the dual graph, and finding its Clifford indices could be automated by moving between C++ and Macaulay. This would then provide a fast way of testing Bayer and Eisenbud's conjecture for hypergraphs of sufficiently small sizes; depending on how much Transformation D speeds up runtime and narrows down the list of hypergraphs, this could be as high as $n = 20$ or so. The effects of the various transformations on Clifford indices of hypergraphs could also be examined.

## References


[BE] D. Bayer and D. Eisenbud, *Graph curves*, Advances in Mathematics. 86 (1991), no.1, 1-40.
[CT] A.-M. Castravet, J. Tevelev, *Exceptional Loci on $\bar{M}_{0,n}$ and Hypergraph Curves*, arXiv:0809.1699
[M] R. Miranda, *Graph Curves and Curves on K3 Surfaces*, Lectures on Riemann surfaces, Trieste, 1987, 119-176.
[Mac] D. Grayson, M. Stillman, *Macaulay 2*, http://www.math.uiuc.edu/Macaulay2/