EuroComb 2005 DMTCS proc. **AE**, 2005, 31–36

Weak Positional Games on Hypergraphs of Rank Three [†]

Martin Kutz [‡]

Max-Planck-Institut für Informatik, Saarbrücken, Germany
mkutz@mpi-inf.mpg.de

In a weak positional game, two players, Maker and Breaker, alternately claim vertices of a hypergraph until either Maker wins by getting a complete edge or all vertices are taken without this happening, a Breaker win. For the class of almost-disjoint hypergraphs of rank three (edges with up to three vertices only and edge-intersections on at most one vertex) we show how to find optimal strategies in polynomial time. Our result is based on a new type of decomposition theorem which might lead to a better understanding of weak positional games in general.

1 Introduction

Let $H = (V, E)$ be a hypergraph, that is, $V = V(H)$ is a finite set (the *vertices*) and $E = E(H)$ is a set of subsets of V (the *edges*). Two players, called *Maker* and *Breaker*, play the following game on H. Maker begins by picking some vertex of H, then Breaker chooses some different vertex. They alternate in this fashion until all vertices of H are taken, retaking of vertices being forbidden. Maker wins if he manages to claim all vertices of some edge $e \in E$, otherwise Breaker wins. Note the obvious unfairness, or rather asymmetry in the game. Breaker does not win by getting a complete edge as Maker does. His moves are only meant to block vertices and make the incident edges useless for Maker. Also observe that by definition, there cannot be a draw.

Such a game is called a *weak positional game* on the hypergraph H. The term "positional game" goes back to Hales and Jewett [7] where a first variant of such games was studied. The attribute "weak" has been coined later to distinguish them from the so-called "strong" games in which *both* players try to complete an edge and which end in a draw if neither succeeds.

The relevant question about a game on a fixed hypergraph is, of course, who can win. That is, does Maker or Breaker have a strategy that always wins. A hypergraph H is a *winner* if Maker, playing first, has a winning strategy on H; otherwise, when Breaker has a winning strategy, we call it a *loser*.

We investigate the computational problem to decide whether a given hypergraph is a winner or a loser. As with any combinatorial game, an efficient decision procedure for this question would entail an efficiently computable winning strategy for any winner by a standard reduction. However, a polynomial-time algorithm for arbitrary hypergraphs should not be hoped for. Schaefer [10] showed that this problem is

[†] A full version of this abstract appeared as part of the author's PhD thesis [9].

[‡] Supported by the Deutsche Forschungsgemeinschaft within the European graduate program "Combinatorics, Geometry, and Computation" (GRK 588/2).

PSPACE-complete, which is "the right" class for a two-person game, and recently Byskov [5] gave a new, very short proof for this result. We focus on hypergraphs with edges of bounded size. The *rank* of a hypergraph is the size of a largest edge. A hypergraph is called k-*uniform* if all its edges are of size k.

Hypergraphs of rank 2 are not very interesting from the point of positional games. Maker wins iff there are two adjacent 2-edges or the graph contains a singleton edge. On the other hand, Schaefer's proof requires no edges with more than 11 vertices, so that the decision problem is already PSPACE-complete for hypergraphs of rank 11. In this interval, between 2 and 11, the smallest interesting rank is 3. We set out to distinguish rank-3 winners from rank-3 losers efficiently, i.e., in polynomial time. We almost succeed. There is a problem with too-much-overlapping edges. We solve the task for rank-3 hypergraphs that are also *almost-disjoint*: no two edges intersect in more than one vertex.

Theorem 1 *The question whether a given almost-disjoint hypergraph of rank 3 is a winner or a loser can be decided in polynomial time.*

Theorem 1 is not so much about efficient algorithms. Our aim is rather to understand the underlying principles which let you win or lose on a hypergraph. Solving the rank-3 case is the first non-trivial step in this direction. We claim that the restriction to almost-disjoint hypergraphs does not define away the problem. The proof of Theorem 1 employs a new decomposition theorem for hypergraph games and eventually rests on a complete classification into winners and losers, which entails several non-trivial steps that give new insight into the mechanics of the game.

Eventually, in Section 3, we introduce a new notion of hypergraph decompositions that allows to compare the values that individual parts of a hypergraph contribute to a game that is played on it.

Previous results. The main branch of research about positional games aims at the development of criteria for the existence of winning strategies, often in terms of the number of edges and vertices, like an early result by Erdös and Selfridge [6], stating that if the number of edges in an n-uniform hypergraph $H = (V, E)$ is bounded by $|E| < 2^{n-1}$ then Breaker can win, i.e., H is a loser. Beck [1, 2] has developed a variety of strong conditions of this kind. We refer to his extensive overview [4].

Other works investigate hypergraphs that are implicitly defined by certain regular structures. For example, in [8] and [3] the two players pick edges from a complete graph and try to obtain a subgraph of a certain prescribed type. A famous class of hypergraphs are generalized Tic-Tac-Toe boards, where the vertex set is the n^d grid cube $\{1, \ldots, n\}^d$ embedded in d-space with exactly all collinear n-sets as edges. These games have already been studied in Hales and Jewett's original paper [7].

Our approach to positional games very much differs from most of the above in that it aims at optimal play for *all* members form a limited class of hypergraphs. While density arguments provide winning or losing criteria for much larger classes of hypergraphs than the one we attack, they cannot give definite answers how to play on any arbitrary given instance. Usually the gap between the best winning criterion and the best losing criterion is rather large, leaving the bulk of difficult instances unresolved.

2 Decomposition and Classification

The basic tools for Maker strategies on rank-3 hypergraphs are paths and cycles. A path is just a sequence of edges such that two consecutive edges share a vertex and other pairs of edges are disjoint. The left path in Figure 1 is a winner because a Maker play at x_1 forces Breaker to take y_1, after which Maker x_2 calls for Breaker y_2, and so on, until Maker gets to play both x_5 and x_6. The path propagates the Maker threat in the left 2-edge to the right. In contrast, the path on the right does not have this property. The double connection x_4, y_4 there effectively separates the two 2-edges. This is the reason for restricting ourselves to almost-disjointness.

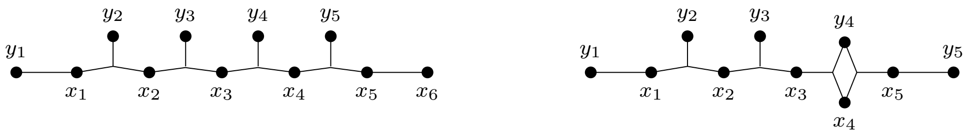

Fig. 1: A winning path (left) and a losing path (right).

It is not difficult to show that a connected almost-disjoint rank-3 hypergraph with at least two 2-edges is a winner. However, even for almost-disjoint hypergraphs winner detection is not trivial. In the absence of 2-edges, or even with just one 2-edge present, our classification requires the discussion of several cases. There is some evidence that it should be possible to overcome almost-disjointness in a preprocessing step without modification of the present result but until now our attempts have not been successful.

Decomposing hypergraphs. Before analyzing a hypergraph in detail, we try to decompose it into as small parts as possible. As one might expect, the components of a disconnected hypergraph do not interact in a game.

Lemma 1 *The disjoint union $H = A \dot\cup B$ of two hypergraphs A and B is a winner iff at least one of A and B is a winner.*

Lemma 1 is not very deep but it is the first step towards a stronger result that serves as the vital tool for our analysis. A vertex p of a connected hypergraph H is an *articulation vertex* if H can be written as a union $H = A \cup B$ of two nontrivial hypergraphs A and B with $V(A) \cap V(B) = \{p\}$. (The left hypergraph in Figure 2 has exactly one articulation vertex, the square one. The central vertex in the hypergraph on the right is *not* an articulation.) We shall see that also in the case of articulations, we can describe how the value of the union H is related to that of A and B.

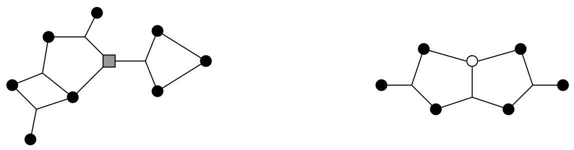

Fig. 2: A hypergraph with an articulation vertex (left) and one without (right).

To this end, we introduce the term $H^{[+x]}$ for a hypergraph in which Maker has already played at x and $H^{[-y]}$ for one in which Breaker has taken y. Formally, we can define these expressions as purely graph-theoretic concepts, without any reference to games: Since a Breaker play at y ruins Maker's prospects in all edges that contain y, we let $H^{[-y]}$ denote the hypergraph H with all edges $e \ni y$ deleted; and because a Maker play at x brings Maker closer to winning in each edge that contains x, we let $H^{[+x]}$ denote the hypergraph H with x deleted from the vertex set and from all edges that contained it. In this interpretation, Maker wins as soon as he produces an empty edge.

Lemma 2 (Articulation Lemma) *Let $H = A \cup B$ be the union of two hypergraphs A and B which have exactly one point p in common, i.e., $V(A) \cap V(B) = \{p\}$. Then H is a winner if and only if one of the following holds:*

- *A is a winner,*
- *B is a winner,*
- *$A^{[+p]}$ and $B^{[+p]}$ are both winners.*

Getting connected components with a 2-edge. For the analysis of a given almost-disjoint rank-3 hypergraph H, we first try to obtain at least one 2-edge. Using Lemma 1, we can show that if H is a winner, there exists a Maker move after which every Breaker answer leaves at least one winning component with a 2-edge. This reduces the general problem to (a quadratic number of) connected hypergraphs with a 2-edge (each). We then cut each such component C repeatedly at articulation points until they are "2-connected" in the sense that they don't have any more articulations.

The Articulation Lemma is the essential tool for these successive decompositions. It guarantees that cutting C at an articulation creates a 2-edge in each resulting subcomponent; and in turn, this can be used to control a potential combinatorial explosion of possible game values on the subcomponents due to the trichotomy in the Articulation Lemma. Altogether, we will be left with a polynomial number of articulation-free hypergraphs with a 2-edge each and it remains to check for each one of them independently whether it is a winner or not.

Classification by case distinction. The classification of articulation-free almost-disjoint rank-3 hypergraphs with a 2-edge makes up the bulk of the proof of Theorem 1. Having removed all articulations turns out crucial there because it grants Maker enough threats along paths such that we can get control over sufficient winning conditions. In the full version [9] of this abstract, the analysis covers over 20 pages of case distinctions. However, the majority of potential configurations lead to contradictions so that eventually, the classification boils down to three topologically different types of winning blocks. Yet, some technical details make the final classification result too complex to be stated here.

3 Comparing Hypergraphs

Eventually, we want to take the Articulation Lemma a bit further, aiming at decomposition theorems for weak positional games on general hypergraphs.

Definition 1 *A k-*pointed hypergraph *is a tuple $(H, p_1, \ldots, p_k)$ consisting of a hypergraph $H = (V, E)$ and a list of pairwise distinct vertices $p_1, \ldots, p_k \in V$ called* points*. The k-*point union *$(A, p_1, \ldots, p_k) \sqcup (B, q_1, \ldots, q_k)$ of two k-pointed hypergraphs is the k-pointed hypergraph*

$$\big((A \mathbin{\dot\cup} B)/\{p_i = q_i : 1 \le i \le k\}, \{p_1, q_1\}, \ldots, \{p_k, q_k\}\big),$$

meaning that we take the disjoint union of A and B and then merge each individual point pair $\{p_i, q_i\}$ into a single new point.

In this terminology, the Articulation Lemma is actually a statement about 1-pointed hypergraphs. Its trichotomy expresses that with respect to 1-point unions, there are essentially three different kinds of hypergraphs: winners, absolute losers, and semi-winners, which yield a win if two of them are plugged together.

To make this precise, we introduce a partial quasi-order on the class of all k-pointed hypergraphs by letting $A \leq B$ for two such hypergraphs A, B iff "$A \sqcup X$ is a winner" implies "$B \sqcup X$ is a winner" for all k-pointed hypergraphs X. Define $\mathcal{H}_k$ to be the partially ordered set that results from identifying equivalent (related by $\leq$ and $\geq$) pointed hypergraphs. This notion of equivalence captures exactly all information about a pointed hypergraph relevant for its impact on winning and losing when plugged into some other pointed hypergraph. In the union $A \sqcup X$ we may replace A by any equivalent B without changing Maker's prospects of winning—independent of the partner X!

Some basic observations about these game orders are not too difficult to prove. Each $\mathcal{H}_k$ contains a unique maximal and a unique minimal element. The former consists of exactly all winners (hypergraphs that are winners on their own) and the latter, the class of "absolute losers," consists of pointed hypergraphs that do not contribute anything, like the empty graph, for example.

We also have some lower bounds on the size of $\mathcal{H}_k$. For each $k \geq 0$, the partial order $\mathcal{H}_k$ contains a chain of length $k+2$ and an antichain of length $\binom{k}{\lfloor k/2 \rfloor}$. Beyond this we do not know much about the structure of $\mathcal{H}_k$. Except for $\mathcal{H}_1$, which by the Articulation Lemma must be a chain of exactly three classes. The main open question is: *Are all $\mathcal{H}_k$ finite?*

A positive answer would have a remarkable implication on the complexity of weak positional games on certain hypergraphs. If a hypergraph allows a successive decomposition into components of bounded size, we could find an optimal strategy by a simple divide-and-conquer approach. For a problem that is PSPACE-complete in general, this would be a quite remarkable result.

References

[1] József Beck. Van der Waerden and Ramsey type games. *Combinatorica*, 2:103–116, 1981.

[2] József Beck. On positional games. *Journal of Combinatorial Theory, Series A*, 30:117–131, 1991.

[3] József Beck. Deterministic graph games and a probabilistic intuition. *Combinatorics, Probability and Computing*, 3:13–26, 1994.

[4] József Beck. Tic-Tac-Toe. In Béla Bollobás, editor, *Contemporary Combinatorics*, number 10 in Bolyai Society Mathematical Studies, pages 93–137. Springer, 2002.

[5] Jesper Makholm Byskov. Maker-maker and maker-breaker games are PSPACE-complete. Technical report, BRICS, University Aarhus, 2004.

[6] Paus Erdös and J. L. Selfridge. On a combinatorial game. *Journal of Combinatorial Theory, Ser. A*, 14:298–301, 1973.

[7] A.W. Hales and R.I. Jewett. Regularity and positional games. *Transactions of the American Mathematical Society*, 106:222–229, 1963.

[8] Frank Harary. Achievement and avoidance games for graphs. *Annals of Discrete Mathematics*, 13:111–120, 1982.

[9] Martin Kutz. *The Angel Problem, Positional Games, and Digraph Roots*. PhD thesis, Freie Universität Berlin, 2004. `http://www.diss.fu-berlin.de/2004/250/indexe.html`.

[10] Thomas J. Schaefer. On the complexity of some two-person perfect-information games. *Journal of Computer and System Science*, 16:185–225, 1978.