CHAPTER 2

Weak Positional Games

1. Tic-Tac-Toe

Let $H = (V, E)$ be a hypergraph, that is, $V = V(H)$ is a finite set and $E = E(H)$ is a set of subsets of V. The elements of V are called the *vertices* of H and the sets in E are the *edges* of H. 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 [**19**] where a variant of such games were first studied. The attribute "weak" has been coined later to distinguish them from the so-called "strong" games which we shall address soon. Briefly, "weak" accounts for the fact that Breaker does not win when he claims an edge $e \in E$ himself.

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. Formally, a *strategy* is a mapping σ from finite sequences $(x_1, x_2, \ldots, x_r)$ of distinct vertices of H to $V(H) \setminus \{x_1, x_2, \ldots, x_r\}$, where $r < |V(H)|$. The obvious semantic being that the x_i describe the course of play up to some point and then σ determines the next move. So in case of a Maker strategy σ is only defined for sequences of even length and only for sequences of odd length in case of Breaker strategies.

A *winning strategy* is a strategy that wins against all possible opponent plays. A fundamental theorem of combinatorial game theory tells us that either one of the two players must have such a winning strategy (games with this property are called *determined*) draw being impossible by the very definition of the game. This is easily shown by a simple game-tree backward-labeling argument, as described in many books on combinatorial games. The essential ingredient here is the finiteness of the game. See Section 2 of Chapter 1 for a brief discussion of some aspects of non-determined games.

Winning strategies for Maker will sometimes be called *making strategies* and such for Breaker *breaking strategies*. In our arguments, we usually like to consider a game out the perspective of Maker, which suggests the following convention.

1. Definition. 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*.

In this work, our main motivation to study positional games is the computational complexity of the question whether a given hypergraph is a winner. Note that an efficient decision procedure for this question would immediately yield winning strategies on any winner by a standard reduction. At each move, we could simply determine the value of the outcomes of all our options together with all possible opponent plays. From this we would then be able to tell which moves are the best.

However, a polynomial-time algorithm for arbitrary hypergraphs should not be hoped for. Schaefer [**39**] showed that this problem is PSPACE-complete, which is "the right" class for a two-person game. The paper does not use the term hypergraph, though, but works with games on DNF formulas, which behave equivalently. Thanks to Jesper Makholm Byskov for pointing me at that result.

We will focus on hypergraphs with edges of bounded size.

2. Definition. 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. Any edge of size 1 yields an immediate Maker win, so we may assume that the hypergraph is 2-uniform, i.e., an ordinary simple graph. If such a hypergraph has any vertex of degree greater than one, i.e., if any two edges share a vertex, Maker wins by playing at such a vertex because in his next move he will complete either of the two edges since Breaker can only play in one of them. On the other hand, Schaefer's proof requires no edges larger than 11, 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. Unfortunately, we do not succeed completely. There is a problem with too-much-overlapping edges. We shall solve the task only for hypergraphs with the following additional property.

3. Definition. A hypergraph is *almost-disjoint* if no two edges intersect in more than one vertex.

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

Theorem 4 is not about efficient algorithms. Our motivation is not the desire to actually play such games better, like with chess, but to understand the underlying principles which let you win or lose on a hypergraph. The above result rests on a classification of rank-3 hypergraphs into winners and losers, which is somehow the more important result. That classification (Theorem 38) depends on several notions that first need to be developed, so that we must defer its statement to a later place, where the actual work is done.

It might be suspected that by restricting ourselves to almost-disjoint hypergraphs, we have defined away the essential part of the problem. This is not the case. Our investigation of almost-disjoint hypergraphs exhibits a lot of structure and the techniques we employ during the analysis reveal some of the deeper mechanics behind such games. Moreover, we shall give some evidence that the almost-disjointness condition can be removed through a preprocessing step, so that our result could be immediately applied to all rank-3 hypergraphs without further modifications in the proof.

Strong games. Positional games can be seen as the natural generalization of the well-known game *Tic-Tac-Toe*, which is played by two players on a board of $3 \times 3 = 9$ squares. Alternately the opponents claim squares, the first player by drawing crosses the second by drawing noughts; reclaiming of previously taken squares being forbidden. Either player wins if he manages to get three squares in a row, horizontally, vertically, or on one of the two diagonals. Figure 1 shows a game in progress.

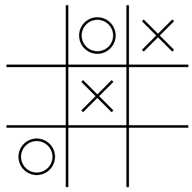


FIGURE 1. A game of Tic-Tac-Toe.

Note the obvious difference to weak positional games. In Tic-Tac-Toe the second player also tries to complete an edge of his own. In some sense, the game now seems fairer.

The natural generalization of this symmetric rule system to arbitrary hypergraphs $H = (V, E)$ leads to the definition of a *strong positional game*. Two players, not called Maker or Breaker now, alternately claim vertices in V until either one player has claimed all vertices of some edge $e \in E$, in which case he wins, or all vertices are claimed and neither player achieved this goal, in which case the game is a draw. The term "strong" will soon become clear when we relate these games to weak games.

The difference between weak and strong games already bears on the simple example of Tic-Tac-Toe. While every child knows that it is a draw in the strong version, Maker can win on the 3×3 board in the weak game because in certain situations Breaker lacks counter threats.

Due to the changed game definition we get a new type of strategy. A *drawing strategy* is a strategy that always leads to at least a draw, i.e., if you follow this strategy you can be sure not to lose and it may happen that you win. Similar to the case of weak games, a simple game-tree argument shows that either one of the two players has a winning strategy or both players have drawing strategies. A special feature of strong positional games is that this trichotomy (first player win, second player win, and draw) collapses to only two cases. The second player cannot win, as can be seen by the following well-known *strategy-stealing* argument. Assume for contradiction that the second player has a winning strategy. Then the first player can "steal" this strategy by playing his first move anywhere and then behaving as if he was the second player. The point is that the additional first move does not create any problems for the first player because of the *monotonicity* of the game. If the strategy prompts him to play a vertex he has already taken, he can just play this move anywhere else and still have all vertices taken that the strategy requires. Having more vertices claimed is never a disadvantage.

So, strong games, if played optimally, also have just two different possible outcomes: first player win or draw. The following trivial statement relates weak and strong games in terms of winning strategies, justifying the pair "weak"/"strong."

5. REMARK. If the first player can win the strong game on a hypergraph H, Maker can win the weak game on H. If Breaker can win the weak game on a hypergraph H, the second player can force a draw in the strong game.

So, taking Maker's respectively the first player's perspective again, being able to win a strong game is really a stronger statement than being able to win the corresponding weak game. Beck's survey paper [**7**] contains a detailed discussion of the relation between weak and strong positional games.

Previous Results. A main branch of research about positional games aims at the development of strong criteria for the existence of winning strategies, often in terms of the number of edges and vertices, like the following early result by Erdös and Selfridge [**15**].

6. THEOREM (Erdös-Selfridge). *Let $H = (V, E)$ be an n-uniform hypergraph. If $|E| < 2^{n-1}$ then Breaker wins the weak game on H and thus the second player can draw in the strong game.*

Beck [**4, 5**] has developed a variety of strong conditions of this kind. We refer to his extensive overview [**7**].

Sometimes hypergraphs are investigated that are implicitly defined by certain regular structures. For example, in [**20**] and [**6**] the two players pick edges from a complete graph and try to obtain a subgraph of a certain prescribed type. Another 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 [**19**]. Berlekamp, Conway, and Guy's classic [**8**] contains a whole chapter about some sorts of positional games, like five-in-a-row on a checker board and games with polyominoes. It also contains a detailed case analysis of the original 3 by 3 Tic-Tac-Toe.

Eventually, we should mention that also strong positional games are PSPACE-complete. Reisch [**37**] showed this for the special case of the board game Gomoku (five-in-a-row in the plane).

Our approach to positional games very much differs from most of the above in that it aims at *optimal* play for a limited class of hypergraphs. While density arguments like Theorem 6 usually give winning or losing criteria for much larger classes of games than the one we attempt to solve, 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 a lot of difficult instances unresolved.

The price we must pay for our desire for a complete analysis, are several lengthy case distinctions and sometimes a certain lack of beauty. Quite in contrast to the nice density theorems of [**19**] and [**15**]. Though we introduce tools to break hypergraphs into nice components, it cannot be avoided that eventually some dirty parts have to be sorted out by direct inspection. The ultimate result however, will be rather concise, a neat classification into winners and losers.

2. Winning Ways

Before we embark on the analysis of rank-3 games, let us briefly discuss a few very basic concepts and fix some related terminology. Consider a single move of Breaker at some vertex y. Clearly, all edges of H that contain this vertex will be of no use for Maker any more because he is not allowed to ever recolor y. So we may interpret Breaker's move as deleting the vertex y and all incident edges $f \ni y$ from H. On the other hand, a Maker move at some vertex x brings Maker one step closer to his goal in each edge that contains x. His move can be understood as shrinking all edges $e \ni x$ by the vertex x, i.e., deleting x from $V(H)$ and replacing each such e by $e' = e \setminus \{x\}$. In this interpretation, Maker wins iff he manages to produce an empty edge. Note how this point of view captures the inherent asymmetry of the game and it is very useful to analyze hypergraphs in which some vertices have already been played by any of the two players. We let

$$H^{[+x_1,\ldots,+x_r,-y_1,\ldots,-y_s]} \tag{12}$$

denote the hypergraph obtained from H by "shrinking away" the Maker vertices $x_1, \ldots, x_r$ and deleting all edges containing any of the Breaker vertices $y_1, \ldots, y_s$ in the above fashion. We shall also use obvious abbreviations of this expression like $H^{[+M]}$ with $M = \{x_1, \ldots, x_r\}$ a set of Maker moves.

Formally, there is no need for the numbers r and s in (12) to be related in any way. We can have, for example, a large number of Maker plays in H but no Breaker moves at all. This expression will still make sense. Hence, our notation can be used to describe the course of play on local fragments of a hypergraph, where the players not necessarily play in alternating fashion. In other words, we can treat *tenuki*—moves that do not directly answer the opponents preceding move locally but shift play to another part of the graph.[1] Second, the resulting hypergraph is clearly independent of the order of deletion and shrinking steps. This is convenient for analyzing snap shots of a game without bothering about the precise order of moves that lead to an actual position.

Playing along paths. We start our investigation of rank-3 hypergraphs by collecting some elementary, though important criteria that guarantee a Maker win. The crucial objects are paths.

[1] In the Asian board game *Go*, a move that stays away from a local fight is called *tenuki*.

7. DEFINITION. A *walk* (from a vertex v_0 to another vertex v_r) in a hypergraph is a sequence $W = (v_0, e_1, v_1, \dots, e_r, v_r)$, $r \geq 0$, of vertices v_i and edges e_i, such that $v_{i-1}, v_i \in e_i$ for $1 \leq i \leq r$. The index r is the *length* of the walk and we call v_0 and v_r the *start* and *end vertex* of the walk, respectively.

A walk is a *path* if all vertices v_i are distinct and $e_i \cap e_j = \emptyset$ for all pairs of indices i, j with $|i - j| > 1$. A *cycle* in a hypergraph is a walk of strictly positive length from a vertex to itself, satisfying all requirements of a path except that, of course, $e_1 \cap e_r$ must not be empty.

We shall often treat a path or cycle W itself as a hypergraph by letting $V(W) = \{v_0, \dots, v_r\} \cup e_1 \cup \dots \cup e_r$ and $E(W) = \{e_1, \dots, e_r\}$. As usual, we say that two vertices of a hypergraph are *connected* if there is a walk between them, and a hypergraph is *connected* if any two of its vertices are.

In contrast to simple graphs, one might come up with alternative definitions for the concept of paths in hypergraphs. We just chose the one that will best serve our purposes. Though the following two notions are absolutely standard and should not bear any ambiguities, we like to provide a rigorous definition.

8. DEFINITION. A *subhypergraph* of a hypergraph H is another hypergraph K with $V(K) \subseteq V(H)$ and $E(K) \subseteq E(H)$. The *induced subhypergraph* on a vertex set $W \subseteq V(H)$ of a hypergraph H is defined as the hypergraph $H[W] := \big(W, \{e \in E(H) \mid e \subseteq W\}\big)$.

If a and b are vertices of a path, we write aPb for the unique subpath of P from a to b. We often stack several such subpaths of different paths to obtain a single long path. For example, if a and b are vertices of paths P and Q, respectively, and P and Q intersect in some other vertex x, then we write $aPxQb$ for the path from a to b in $P \cup Q$ via x. Of course, we then have to check that the resulting walk is a path again but in most cases this will be obvious. When the path we want to use consists of only one edge, $f = \{a, b, x\}$, for example, we sometimes simply write afb. We also use constructs like $aPxQa$ to create a cycle from two paths that intersect in two vertices a and x.

The following lemma is rather trivial, but as we already emphasized, paths in hypergraphs require a slightly more careful treatment than paths in simple graphs. So we like to give a rigorous proof here to make sure not to overlook any details and to comply with our definitions.

9. LEMMA. *If two vertices a, b in a hypergraph are connected then there exists a path from a to b.*

PROOF. We claim that any shortest walk $(v_0, e_1, v_1, \dots, e_r, v_r)$ from a to b is actually a path. Otherwise there would be two edges e_i, e_j with $i < j - 1$, such that the intersection $e_i \cap e_j$ contains some vertex x. But then the sequence $(v_0, e_1, \dots, e_i, x, e_j, \dots, e_r, v_r)$ is a shorter walk from a to b; a contradiction. □

Figure 2 shows a path of length 7 with five 3-edges in the interior and a 2-edge at each end. Assume that Maker plays at x_1. Then Breaker must clearly answer at y_1. After that, Maker x_2 leaves only Breaker y_2 and then, Maker x_3 forces Breaker y_3. And so on. Maker can play all the way down to x_6, where he wins because Breaker will have to answer x_6 at y_6, leaving the singleton edge $\{x_7\}$ for Maker.

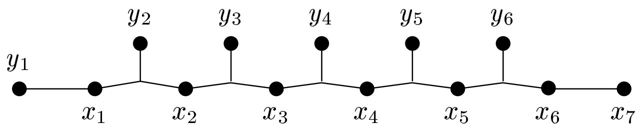


FIGURE 2. A winning path.

This scheme only works because any two adjacent edges of the path intersect in just one vertex. The hypergraph in Figure 3 is a loser. If Maker tries the same trick there, he gets stuck in the middle because after Maker x_4 there, Breaker y_4 will destroy his options for the right side. However, if the hypergraph at hand is almost disjoint then all paths are nice.

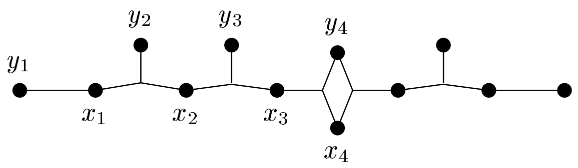


FIGURE 3. A non-almost-disjoint losing path.

10. LEMMA. *An almost-disjoint rank-3 hypergraph that has a path containing two 2-edges is a winner.*

PROOF. We may assume that the path contains exactly two 2-edges and that these are in the two terminal positions by simply removing further 2-edges and trailing 3-edges. So we have a path $(v_0, e_1, v_1, \ldots, e_r, v_r)$ where e_1 and e_r are 2-edges and the other e_i are 3-edges. Maker wins by playing along this paths as described above. $\square$

Combining Lemmas 9 and 10 we get the following useful win criterion.

11. COROLLARY. *Any connected almost-disjoint rank-3 hypergraph with at least two 2-edges is a winner.* $\square$

During the analysis of a game in progress, it will often be useful to have the following variant of Lemma 10 available, which tells us how Breaker has to reply to a Maker move in a component with a 2-edge.

12. LEMMA. *Let P be a path in an almost-disjoint rank-3 hypergraph and assume that P contains a 2-edge. If Maker plays somewhere in P then Breaker must answer somewhere in P, too; otherwise Maker wins.*

PROOF. If Maker plays inside the 2-edge the statement is trivial. Otherwise, Maker creates an additional new 2-edge that lies on a common path with the original 2-edge. By Lemma 10, Breaker must answer on this subpath of the original path. $\square$

Inner and outer vertices. Let us have a closer look at that carbon molecule in Figure 2 again. The vertices that Maker played there shall be of general importance for us.

13. Definition. Let P be a path or a cycle. A vertex of P that appears in more than one edge is called an *inner vertex* of P; the other vertices are the *outer vertices* of P.

The way Maker won in Figure 2 was not unique. It is not hard to see—though we won't prove this now—that he could have started at any of the inner vertices and still have won, while the outer vertices would have all lead to a loss. The reason for this is in a way to be found in the following absolutely trivial, yet important fact.

14. Remark. If x is an inner vertex on an almost-disjoint path P from a to b then the subpaths xPa and xPb only intersect in the vertex x. Similarly, if Maker plays at an inner vertex of an almost-disjoint cycle, this cycle is split into a path.

Note that outer vertices do *not* have this property. The following two lemmas, which will be useful in many situations, exploit the above observation for cycles.

15. Lemma. *Let C be a cycle in an almost-disjoint rank-3 hypergraph. If Maker plays at an inner vertex of C then Breaker must answer somewhere in C, too; otherwise Maker wins.*

Proof. Playing at an inner vertex, Maker turns the cycle into a path with a 2-edge at each end, which by Lemma 10 is a winner. See the left-hand side of Figure 4. □

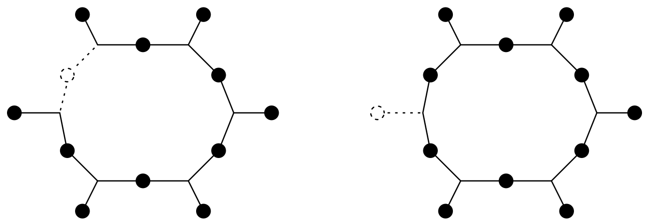


Figure 4. Playing an inner vertex of a 3-uniform cycle yields a path with two 2-edges (left), playing an outer vertex yields a cycle with a 2-edge (right).

Of course, it is crucial again to pick an inner vertex. Playing an outer vertex of a cycle yields just a cycle with a 2-edge, as shown on the right-hand side of Figure 4. If Maker then plays in such a cycle again, Breaker has only few options left.

16. Lemma. *Let C be a cycle in an almost-disjoint rank-3 hypergraph and assume that C contains exactly one 2-edge. If Maker plays at an inner vertex of C then Breaker must answer in the 2-edge; otherwise Maker wins.*

PROOF. If Maker plays in the 2-edge, the statement is trivial. Otherwise, his move, which breaks up the cycle into a path, creates two 2-edges. This leaves a path with three 2-edges altogether. If Breaker does not play in the original 2-edge now, which is clearly the middle one, he leaves behind two 2-edges connected by a path. A win, by Lemma 10. □

3. Decomposing Hypergraphs

The last two lemmas from the previous section demonstrated the potential of cycles for Maker. With a single move at an inner vertex of a cycle he could create an immediate threat. A key tool for our analysis of hypergraph games will be a decomposition lemma that allows us to reduce any hypergraph into parts that are doubly connected in a certain way. In those parts we will then have good chances to find cycles that yield several Maker threats, allowing us to construct winning strategies for Maker.

We start with a simple observation about disconnected hypergraphs. For two hypergraphs H_1 and H_2, their union $H = H_1 \cup H_2$ is given simply by $V(H) = V(H_1) \cup V(H_2)$ and $E(H) = E(H_1) \cup E(H_2)$. In case of disjoint vertex sets $V(H_1)$ and $V(H_2)$ this yields a disconnected union $H = H_1 \dot\cup H_2$. It appears plausible that in such a case, moves played in one component should not interfere with those played somewhere else. Let us formalize this intuition.

17. LEMMA. *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.*

PROOF. If A or B is a winner then clearly H is. So assume that neither A nor B can be won. So there are breaking strategies α and β for A and B, respectively. Against any Maker strategy, Breaker can use these to obtain a breaking strategy for H. Whenever Maker plays in A he answers according to α, when Maker plays in B Breaker follows β, at each move always ignoring anything that happened in the other component. This way Breaker can assure that in none of the two components Maker can get a monochromatic edge. Thus, H is a loser. □

Lemma 17 tells us that if Maker can win on some hypergraph H he only needs one component of H, never playing in the rest of H. And of course, this rule can be applied recursively to any stage of the game: Maker never needs to leave a component he once played in.

Splitting at articulations. Lemma 17 is not very deep. But it paves the way for a stronger result that will become a vital tool for our analysis of games on rank-3 hypergraphs. Suppose that the components A and B of H are not completely disjoint but almost, i.e., they share just one vertex. Then we can still relate the winning and losing behavior of A and B to that of H.

18. DEFINITION. We call a vertex p an *articulation vertex* of a connected hypergraph H 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 5 has exactly one articulation vertex, the square one. The central vertex in the hypergraph on the right is *not* an articulation.

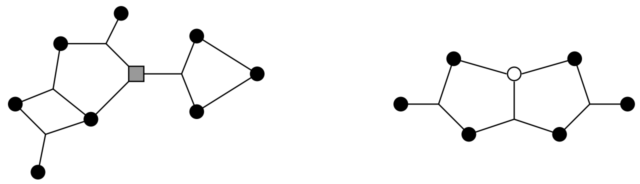


FIGURE 5. A hypergraph with an articulation vertex (left) and one without (right).

19. LEMMA (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.*

PROOF. First note that every single one of the three cases implies a win for H. For the first two this is clear. If the last case holds, Maker can win by playing his first move at p. This leaves two disjoint graphs both of which he can win. Breaker cannot answer in both, so at his second move, one of $A^{[+p]}$ and $B^{[+p]}$ will still be available to Maker and give him a win.

For the converse implication consider the case that none of the three options in the statement of the lemma is true. By symmetry we may assume that $B^{[+p]}$ is a loser. So we have breaking strategies α and β for A and $B^{[+p]}$, respectively. Breaker combines these strategies as follows. Against any Maker move in A he also answers in A, according to his strategy α. When Maker plays in $B \setminus \{p\}$ he answers there, following to strategy β. This way Maker can never complete one edge of H since the edges of A are taken care of by α and the strategy β guarantees that even if Maker should get the vertex p, it won't help him on B because not only B but even $B^{[+p]}$ was a loser. □

Note that we had to require the nontriviality and connectivity condition in the definition of an articulation vertex for technical reasons. (Otherwise every vertex would be an articulation.) Lemma 19 does obviously not depend on such restrictions.

Figure 5 indicates that in contrast to simple graphs, hypergraphs allow different notions of connectivity. If we removed the central vertex from the right hypergraph in that picture together with all incident edges, we would of course decompose the hypergraph into disjoint components. But that is not what we want because the Articulation Lemma does not apply to that hypergraph. The "right" notion of connectivity for us is the following.

20. DEFINITION. A hypergraph H with at least k vertices is *Maker-k-connected* if its reduction $H^{[+M]}$ is connected for every set $M \subseteq V(H)$ of Maker moves that has cardinality strictly less than k.

Practically, Maker-k-connectivity means that Maker would have to play at least k times until the hypergraph decomposes. Note also that Maker-1-connectivity is equivalent to ordinary connectivity because then $M = \emptyset$ is the only allowed set of Maker moves. We refrain from defining the analog concept of "Breaker-connectivity" since we shall not need it anyway.

21. LEMMA. *A hypergraph H with at least k vertices is Maker-k-connected iff it cannot be written as a union $H = A \cup B$ with $V(A), V(B) \neq V(H)$ and $|V(A) \cap V(B)| < k$.*

The crucial property here is, of course, that the hypergraphs A and B do not overlap on k vertices, the other restriction only makes sure that the decomposition is nontrivial in the sense that A and B are both really needed in the union.

PROOF OF LEMMA 21. Assume that we have such a representation $H = A \cup B$. Taking $M = V(A) \cap V(B)$ immediately gives us a Maker set such that $H^{[+M]}$ is disconnected. Conversely, assume that there exists a set $M \subset V(H)$ of cardinality $\ell < k$ such that the reduced hypergraph $H^{[+M]}$ is disconnected, i.e., $H^{[+M]} = A' \dot\cup B'$. This decomposition tells us that any H-edge lies completely in $V(A') \cup M$ or $V(B') \cup M$. Therefore, we can write H as the union $H = H^{[-V(B)]} \cup H^{[-V(A)]}$, where the two vertex sets intersect in the set M which has cardinality $\ell < k$. $\square$

22. COROLLARY. *A hypergraph with at least two vertices is Maker-2-connected iff it is connected and contains no articulation vertex.* $\square$

Path decompositions. Through repeated application of Lemmas 17 and 19 we will reduce statements about general hypergraphs to such about Maker-2-connected hypergraphs. Those are then amenable to the following path-adjoining lemma, which is very much redolent of classical ear-decomposition theorems. Here, however, it appears in a slightly technical guise, due to the special requirements of our analysis in the subsequent sections.

23. LEMMA. *Let H be a rank-3 Maker-2-connected hypergraph and let (B, M, T), with $\emptyset \neq B, M, T \subsetneq V(H)$, be a nontrivial partition of the vertices of H such that no vertex in B is adjacent to a vertex in T. In other words, the "middle layer" separates "bottom" from "top." Then there exists a path in $H[M \cup T]$ connecting two distinct vertices a and b in M and using no further vertices in M and no edges of $H[M]$.*

In one sentence, Lemma 23 tells us that if we step from the middle layer into the top layer then we find a path through T that brings us back to M.

Have a look at Figure 6. In the typical application of Lemma 23, the middle layer M will be a part of a hypergraph H that we are currently reconstructing and about which we already know a lot of structure, while the top layer T contains the unexplored parts of H that are somehow connected to M. The lemma then tells us that we can extend M into T path by path in a regular fashion. The lower layer B contains all the remaining vertices that are of no interest for the local situation.

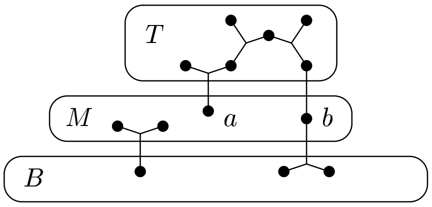


FIGURE 6. Finding paths with Lemma 23.

PROOF OF LEMMA 23. Pick a connected component C of the hypergraph $H[M \cup T] - E(H[M])$ (i.e., the subhypergraph induced by the vertex set $M \cup T$ without those edges that lie entirely in M) that contains at least one vertex in T. For example, in Figure 6, the path from a to b through T would be such a component C. The intersection $X = V(C) \cap M$ has cardinality at least two because H is Maker-2-connected. ($C^{[+X]}$ contains no vertex in M, so it is disconnected from B by assumption; therefore, the hypergraph $H^{[+X]}$ itself is disconnected and thus, $X \geq 2$.)

For each pair u, v of distinct vertices from X, pick a shortest path $P_{u,v}$ from u to v in C. Such paths exist by Lemma 9. Amongst all these paths (for all possible pairs u, v) pick one, $P_{a,b}$, say, of minimal length. We claim that this is a path as required by the statement of the lemma. Assume for contradiction that $P_{a,b}$ contains more vertices in M than only a and b, some additional vertex c, say. The three vertices a, b, c cannot lie in the same edge of C because then they would form an induced edge of $H[M]$ which we had excluded. Consequently, one of the paths $P_{a,c}$ and $P_{c,b}$ must be shorter than $P_{a,b}$—a contradiction to minimality. □

Creating a 2-edge. In Section 2 we emphasized that in a rank-3 hypergraph, 2-edges are good for Maker. Already two of them lead to a win if the hypergraph is almost disjoint, by Corollary 11. In this section we show how to reduce the problem whether a 3-uniform hypergraph is a winner, to the question whether a rank-3 hypergraph with a 2-edge is a winner. Those will then be easier to analyze.

24. LEMMA. *Let H be a 3-uniform hypergraph that is a winner. Then there exists a Maker move x such that for any Breaker answer y, the hypergraph $H^{[+x,-y]}$ has a connected component that contains a 2-edge and is a winner.*

PROOF. By induction on the size of H. Take the first move x from any making strategy for H. Assume for contradiction that for some Breaker answer y the hypergraph $H' = H^{[+x,-y]}$ has no connected component that is a winner with a 2-edge. By Lemma 17, H' must have a component W that is a winner and by assumption, W contains only 3-edges. But such a W is actually a proper subhypergraph of H, so by induction there is a Maker move $\hat{x} \in V(W)$ such that for every Breaker answer $\hat{y} \in V(W) \setminus \{\hat{x}\}$ the remainder $W^{[+\hat{x},-\hat{y}]}$ has a winning component that contains a 2-edge. Since W was a subgraph of H, we can use $\hat{x}$ as the first Maker move in H as well and this will then guarantee a winning component with a 2-edge after any Breaker answer. □

One could easily generalize the proof of Lemma 24 to show that Maker can actually win by always playing inside a component that contains at least one 2-edge, except for his first and last move, of course. But all we need here is a 2-edge after the first move as guaranteed by Lemma 24 because it gives us the following reduction from a 3-uniform hypergraph H to hypergraphs with at least one 2-edge.

For each pair x, y of first Maker and Breaker moves, check whether among those components of $H^{[+x,-y]}$ that contain a 2-edge there is at least one winner. If for some x this is the case for all possible answers y then H is a winner, otherwise it's a loser.

Once we have a 2-edge, we use the Articulation Lemma to cut our hypergraph recursively at articulation vertices, so that eventually we will be left with Maker-2-connected hypergraphs only. Having created a 2-edge is really important for this step. In the proof of the subsequent lemma, the presence of the 2-edge eliminates one alternative in the Articulation Lemma, giving us sufficient information to avoid a possible combinatorial explosion during the decomposition process.

25. Lemma. *Let H be an almost-disjoint connected rank-3 hypergraph with exactly one 2-edge f. Let $H = A \cup B$ be a decomposition of H with $V(A) \cap V(B) = \{p\}$ for some articulation vertex p, such that f lies in A. Let $B_1, \dots, B_r$ be the connected components of the hypergraph $B^{[+p]}$. Then each of the connected hypergraphs $A, B_1, \dots, B_r$ contains at least one 2-edge, and H is a winner iff at least one of them is a winner.*

Proof. Since H was connected, each of the B_i has at least one edge that contained the deleted vertex p in H. Hence, those edges are 2-edges. Clearly A is connected simply because H is and it contains a 2-edge by assumption.

For the stated equivalence, first observe that the preconditions alone imply that $A^{[+p]}$ is a winner: if $p \in f$ then because $A^{[+p]}$ contains a 1-edge and otherwise because $A^{[+p]}$ has at least two 2-edges. Now the Articulation Lemma tells us that H is a winner iff one of A and $B^{[+p]}$ is. (Since $A^{[+p]}$ is a winner, the third case of the Articulation Lemma reduces to "$B^{[+p]}$ is a winner," which makes the second case obsolete.) And by Lemma 17, $B^{[+p]}$ is a winner iff one of the B_i is. $\square$

We use Lemma 25 as an algorithmic recipe for reducing the problem of deciding whether a given connected almost-disjoint rank-3 hypergraph H with exactly one 2-edge is a winner, to such hypergraphs that are even Maker-2-connected instead of just connected.

If an application of Lemma 25 yields any B_i with more than one 2-edge, this B_i is a winner by Corollary 11, and then H is one, too. Otherwise, we apply Lemma 25 recursively to each of $A, B_1, \dots, B_r$ until we either find a component with two 2-edges or no articulation vertices are left and hence, all pieces are Maker-2-connected. (Remember that a single 2-edge is by definition Maker-2-connected.) Eventually we know that the original hypergraph H is a winner iff one of those Maker-2-connected fragments is.

4. Between the Docks

We are left with the task of finding out whether a given almost-disjoint rank-3 Maker-2-connected hypergraph H with exactly one 2-edge is a winner. Figuratively, we shall view the unique 2-edge, which we will henceforth denote by $\phi = \{\alpha, \beta\}$, as sitting at the center of H and everything else arranged around it. We then try to understand how this environment can look like, under what conditions it yields a win and why it perhaps does not.

Call all edges adjacent to ϕ *dock edges*, motivated by the fact that the rest of H is connected to ϕ through them. Anything else between the docks, that is, the subhypergraph of H with α, β, and all dock edges removed, will be called the *core*, denoted by K.

By almost-disjointness, each dock edge contains only one of α and β, so anticipating the way we shall draw pictures, we may speak of *lower docks*, those incident with α, and *upper docks*, incident with β. The vertices in the docks, except α and β, are called *dock vertices*. The two sets of upper and lower docks will sometimes be referred to as the upper and lower *shore*, respectively. We distinguish two types of docks, which have to be treated very differently. Call a dock *closed* if its dock vertices are connected in K, otherwise call it *open*.

Figure 7 gives an overview. It displays a hypergraph with four upper and four lower docks. Connections between docks being indicated as mere paths, though they can, in principle, be arbitrarily complicated, of course. As in most figures in this section, we omit the 2-edge ϕ between α and β from the drawing for graphical reasons.

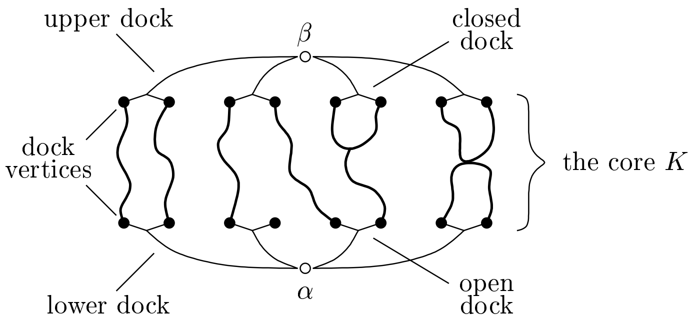


FIGURE 7. A schematic picture of docks and core.

To decide whether a hypergraph arranged as above is a winner or a loser, we take the following approach. Throughout this section we make the general assumption that the hypergraph H at hand is a loser and try to rule out configurations that would conflict with this assumption because they yield a Maker win. Eventually, we shall find that only a few connection types between the docks are possible. After that, in the next section, we shall prove that our classification is valid, i.e., none of the left-over configurations can be won by Maker.

We begin our analysis on a global scale. Our first observation accounts how many docks of what type can be connected to a single open or closed dock.

26. OBSERVATION. *In the core K of a loser H, no two different docks from the same shore are connected. A closed dock is connected to exactly one dock on the other shore. Each dock vertex of an open dock is either connected to one dock on the other shore or to no docks at all, but at least one of them is connected to another dock.*

PROOF. The first statement is the basic observation, which then implies the others. Assume for contradiction that two different lower docks $e = \{\alpha, a, a'\}$ and $f = \{\alpha, c, c'\}$ are connected in K, i.e., there is a path from a or a' to c or c'. Pick a shortest such path P and change vertex labels if necessary, to have P going from a to c; this guarantees that none of a' and c' are touched by P. (Note that $a = c$ is possible.) See Figure 8. Maker can win by playing at α because by Lemma 15 this move requires an immediate answer in the cycle $\alpha eaPcf\alpha$ but Breaker must also destroy the now singleton edge $\{\beta\}$.

The rest is an easy implication of the above. Every dock must be connected to at least one other dock to make H Maker-2-connected and in each case a connection to some further dock would induce a connection between docks from the same shore. $\square$

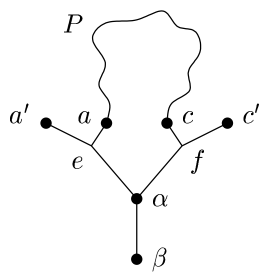


FIGURE 8. Two connected lower docks yield a win.

Figure 7 already contained schematic representations of all dock connections allowed by Observation 26.

In the following, we investigate the local structure of the different connection types between docks: open-open, closed-closed, and closed-open. In each case, we face a lower dock $g = \{\alpha, a, c\}$ and an upper dock $h = \{\beta, b, d\}$ that are somehow connected in the core K of H. As in Observation 26 above, we always make the general assumption that the whole hypergraph H is a loser.

Between two open docks. The situation between vertices from two open docks is very simple.

27. OBSERVATION. *Let the docks $g = \{\alpha, a, c\}$ and $h = \{\beta, b, d\}$ both be open, with a and b connected in the core K. Then the connected component of K that contains a and b is simply a path between these two vertices that contains no further dock vertices.*

Figure 9 visualizes Observation 27. We postpone the proof for a second to discuss a general issue. Almost all arguments throughout this section require us to pick inner vertices on paths that lie "between" certain given

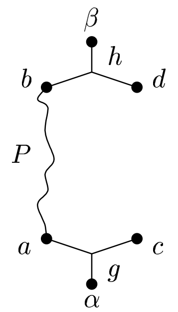


FIGURE 9. Two open docks connected by a path.

vertices. While such a notion would be clear for ordinary graphs, we should make it precise for our hypergraphs.

28. DEFINITION. For two distinct vertices u, v of a path or cycle P, we say that some other vertex x lies *between u and v on P* if x is an inner vertex of the subpath uPv or $x = u$ or $x = v$.

As an example, we have marked the vertices between u and v on the path on the left-hand side of Figure 10 with circles. We will use this concept in situations where there exists some other path Q from u to v, with Q disjoint from P except for the terminal vertices u and v. Then a vertex between u and v on P will be an inner vertex of the cycle $uPvQu$.

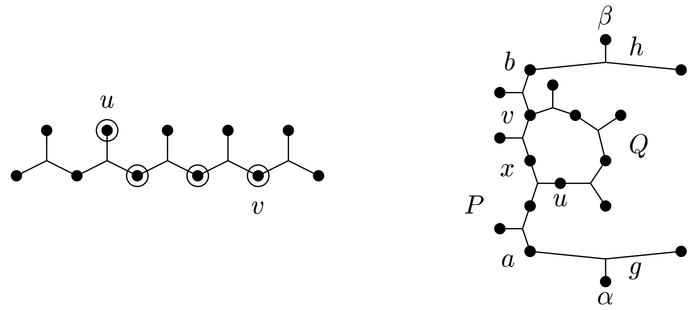


FIGURE 10. An example path with all vertices *between* two vertices u and v marked (left) and an extension of Figure 9 by another path (right).

The main ingredient for the proof of Observation 27 is Lemma 23, which we use here for the first time. It is the technical tool to provide us with the intuitively obvious fact that if we add any further edge to the hitherto constructed part of H between two docks, there will be a whole new path between two distinct vertices of this subgraph because H is Maker-2-connected. We will see this argument repeatedly in the following and we give it here in great detail as a general example.

PROOF OF OBSERVATION 27. Pick any path P from a to b as shown in Figure 9. Maker will use the cycle $C = \alpha g a P b h \beta \phi \alpha$ (i.e., the path P closed to a cycle by the two docks and the 2-edge ϕ) to set up threats against

Breaker. To show that no further edges are incident to vertices of P, we assume for contradiction that some edge $e \in E(K) \setminus E(P)$ is connected to P.

In the general case, when e contains a vertex $z \notin V(P)$, we apply Lemma 23 as follows. The middle layer M in that lemma is $V(P)$. The bottom layer B consists of all vertices of H that are connected to α and β in $H - P$, i.e., it contains α, β, and the vertices between all the other docks of H. The top layer T is the rest $V(H) \setminus (M \cup B)$, which is not empty because we have $z \in T$. Now Lemma 23 tells us that there is a path Q in K that connects two distinct vertices u, v of P and contains no further vertices of P.

Between the vertices u and v on P we find an inner vertex x of our cycle C. (We refer to the inner vertices of C rather than those of P because we need to include the end vertices a and b as well.) The right-hand side of Figure 10 shows a concrete example where one of u and v is an outer vertex of P and the other an inner. A suitable x is found between them. This x is clearly also an inner vertex of the cycle $D = xPuQvPx$.

In the special case $e \subseteq V(P)$ we do not need the Lemma 23 for path finding, of course. Simply pick two vertices u, v of e that are closest to each other on P. Again there is an inner vertex x of C between u and v, which is also an inner vertex of the cycle $D = xPuevPx$.

In any case, Maker wins by playing at x because Lemma 15 restricts Breaker's reply to the cycle D, while Lemma 16 requires a move in the 2-edge ϕ of C, which does not touch D. □

Since Observation 26 leaves the possibility that one of the two dock vertices of an open dock is connected to no other dock vertex as long as the other one is, we must note this simple case, too.

29. Observation. *If a dock vertex of an open dock is connected to no other dock vertices then it is not incident to any edge of K.*

Proof. If some K-edge was connected to such a vertex, this vertex would be an articulation point of H, in contradiction to Maker-2-connectedness. □

Between two closed docks. The situation of two closed docks connected in K is considerably more complicated to analyze than the previous case of two open docks. A waterproof discussion requires the investigation of many potential configurations. In the end, however, we shall see that all but one simple arrangement can be excluded because they would lead to immediate Maker wins.

Let us begin with the construction of the objects that we know must be there. Pick two paths A and B in K, the former from a to c and the latter from b to d. See Figure 11 for two concrete example configurations. We do not require, nor can we prove, disjointness of A and B but we know these paths cannot intersect too deeply. As it turns out, the vertex x in the left example from the figure already leads to a Maker win.

30. Observation. *The paths A and B cannot share a vertex that is at the same time an inner vertex of A or one of the two dock vertices a and c, and an inner vertex of B or one of the two dock vertices b and d. In particular, the docks g and h do not intersect.*

PROOF. Assume there exists such a common vertex and pick such an x, if possible one of the dock vertices. We show that Maker wins by playing at x. If x is an inner vertex of A, we have the two paths $P_a := xAag\alpha\phi\beta$ and $P_c := xAcg\alpha\phi\beta$ (see left-hand side of Figure 11) in each of which Breaker must answer. So Breaker must play a vertex in $g \cup \phi$. If $x = a$ or $x = c$, Lemma 12 forces Breaker to answer in the same set. A symmetric argument for the upper shore shows that Breaker must also play in $h \cup \phi$.

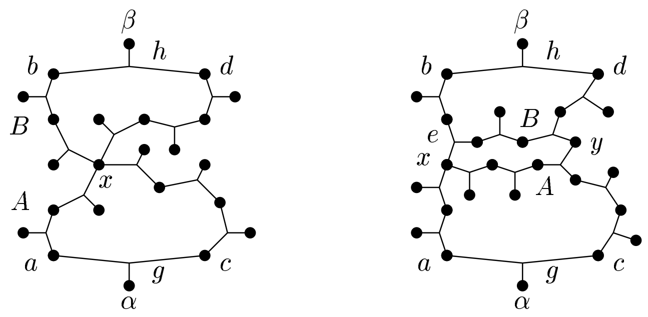


FIGURE 11. The two connecting paths A and B touching in a common inner vertex (left) and touching in two different vertices (right).

If g and h do not intersect then this already tells us that Breaker can only play at α or β. If g and h do intersect then their intersection is by almost-disjointness only one vertex, which by our choice must be x and is thus already taken by Maker. Therefore, Breaker is restricted to play at α or β in this case, too. In addition to this, Lemma 15 requires an answer in each of the two cycles $xAagcAx$ and $xBbhdBx$, whose intersection does not contain α and β. So Maker wins. $\square$

Note that Observation 30 also excludes the possibility that the paths A and B share an edge because at least one vertex of such an edge would be an inner vertex in both paths (or dock vertex). This tells us that A and B cannot overlap too much. We now show that they cannot even intersect in two vertices.

31. OBSERVATION. *The upper and lower path, A and B, share at most one vertex.*

PROOF. Assume for contradiction that the two paths intersect in more than one vertex; we show that this gives a Maker win. Pick a shortest path P in $A \cup B$ from g to h. By symmetry we may assume that P goes from $a \in g$ to $b \in h$ and then minimality implies that $c, d \notin V(P)$. Hence, $C = \alpha gaPbh\beta\phi\alpha$ is a cycle.

Starting at a, we walk along C into the core until we enter the first edge e that does not lie in A. (In an extremal case, e might actually be the dock edge h.) Denote the inner vertex of C that came just before e by x; clearly $x \in V(A) \cap V(B)$. Note that we do not claim that x be an inner vertex of A or B. Compare the right drawing of Figure 11.

Let $y \neq x$ be a further contact point of A and B such that the total length of the paths xAy and xBy is minimal. Observation 30 implies that these two subpaths share no edges and therefore, by minimality and almost-disjointness, the composition $D = xAyBx$ is a cycle (not self-touching) which clearly has x as an inner vertex. We have constructed two cycles, C and D, which share x as an inner vertex. Maker plays at x. By Lemma 16 Breaker must answer at α or β and by Lemma 15, he must play somewhere in D, but these vertex sets are disjoint. $\square$

In case that A and B do not touch at all, we now extend our construction by a connecting path. Let F be a shortest such path from any vertex u of A to any vertex v of B. (See the right-hand side of Figure 12 for an example.) Note that in contrast to ordinary graphs, the minimality of F does not guarantee that F contains no further vertices of A or B. So we have to prove this property.

32. Observation. *The connecting path F touches A and B each in only one vertex.*

Proof. Assume for contradiction that F touches B in two vertices, u, v, say. If one of these vertices is a dock vertex, let x be this dock vertex. (By almost-disjointness there can be only one.) Otherwise there lies at least one inner vertex of B between u and v; let x be such an inner vertex then. See the left-hand side of Figure 12.

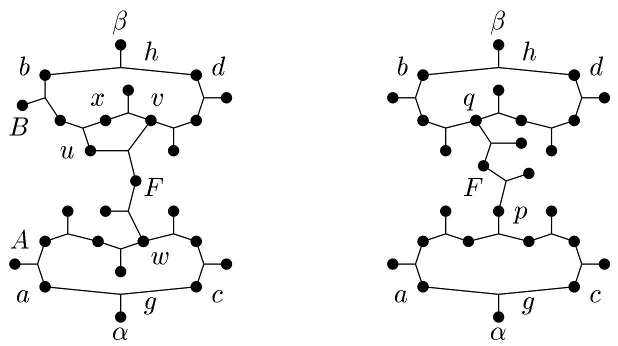


Figure 12. The connecting path F touching B in two places (left) and touching an inner vertex of B (right).

Maker plays at x. If x is a dock vertex then Lemma 12 requires an answer in the path $xh\beta\phi\alpha$. Otherwise we have the two paths $xBbh\beta\phi\alpha$ and $xBdh\beta\phi\alpha$ in both of which Breaker must play, which leaves the same replies b, d, β, α. By Lemma 15, Breaker must also answer in the cycle $xBuFvBx$ since it contains x as an inner vertex. Together, even in the best case for Breaker, when b and d both lie in that cycle, he is left with no more replies than b and d.

There are further threats on the lower side. We have the two paths $xBuFwAsg\alpha\phi\beta$ and $xBvFwAsg\alpha\phi\beta$, where w is a contact vertex of F and A, and s is either a or c. Lemma 12 forces Breaker to play in both paths but their intersection clearly contains none of b and d; hence Maker wins. $\square$

We now know that F connects exactly one vertex p of A to one vertex q of B. (Where the case that A and B touch is included as the degenerate case where F has length 0 and consists of just one vertex $p = q$.) See Figure 13. We can say even a little bit more. The contact points p and q cannot be arbitrary vertices of A and B. Only outer vertices, as drawn in the figure, are allowed.

33. OBSERVATION. *The vertices p and q are outer vertices of A and B, respectively.*

PROOF. Assume for contradiction that one of them, q, say, is an inner vertex, of B. See the right-hand side of Figure 12. We assume by symmetry w.l.o.g. that c is no closer to p than a so that $qFpAag\alpha\phi\beta$ is a path (i.e., does not use a vertex twice).

Maker plays at a. Then Lemma 12 requires an answer in the path $ag\alpha\phi\beta$ and Lemma 15 one in the cycle $aAcga$. Therefore Breaker must play at α or c. Maker's next move is at q. It lies on the path $(qFpAa)^{[+a]}$ and is an inner vertex of the cycle $qBbhdBq$. Lemma 12 and Lemma 15 require an answer in the path and the cycle, respectively, and since these substructures intersect only in Maker's vertex q, Maker wins. $\square$

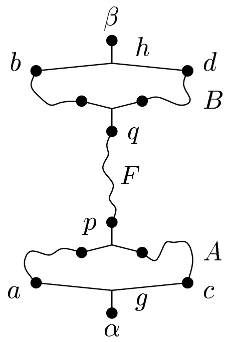


FIGURE 13. The final configuration between two closed docks consisting of the three paths A, B, and F.

We are almost done. It remains to show that Figure 13 is complete.

34. OBSERVATION. *There are no further edges in the core K touching any vertex of the three paths A, B, and F.*

PROOF. Let $M := A \cup B \cup F$. We assume for contradiction that there exists some further edge $e \in E(K) \setminus E(M)$ that contains a vertex of M. If e contains also some vertex outside of M, we can apply Lemma 23 to obtain a path P in K connecting two vertices $u, v \in V(M)$ and containing no other vertex of M. In the degenerate case, when $e \subseteq V(M)$, we pick two vertices $u, v \in e$ such that the unique path from u to v in M does not contain the third vertex w of e.

Denote by Q the unique path from u to v in M, precisely, Q is of the form uAv, uBv, uFv, $uApFv$, $uBqFv$, or $uApFqBv$, depending on the locations of u and v. Together with the path P, respectively the edge e, this path forms a cycle $C = uQvPu$ respectively $C = uQveu$ in K. Next we pick a shortest path R in M from the lower dock to the upper dock, w.l.o.g. $R = aApFqBb$. Minimality guarantees that this path does not contain the other two dock vertices c and d, so that the composition $D = \alpha g a R b h \beta \phi \alpha$ is a cycle which contains the 2-edge ϕ. The left-hand side of Figure 14 shows what we have constructed so far.

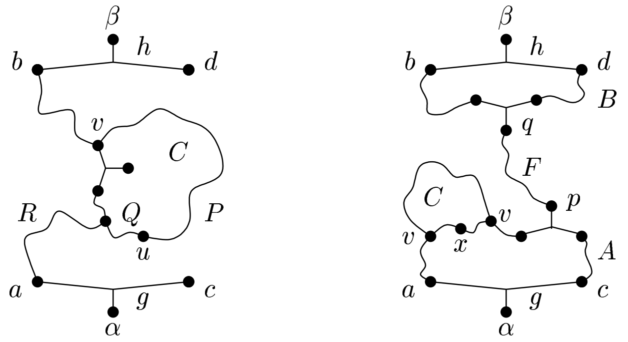


FIGURE 14. Cycle constructions for the proof of Observation 34.

We now have to distinguish the different types of Q. If the cycle C contains an edge of F then this edge contains a vertex x that is inner to both cycles. As in many situations before, Lemmas 15 and 16 then show that if Maker plays at x, Breaker has no reply to the threats of the two cycles C and D, so he loses.

The situation is similarly easy for Maker if $u, v \in V(A)$ and the subpath uAv contains the contact point p (the case $u, v \in V(B)$ being completely symmetric to this). Then the cycle C again shares an edge with R, namely the one edge of A that contains the vertex p. So Maker wins at a vertex in this edge.

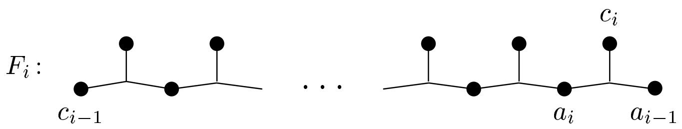
The only remaining configuration is one that has u and v on the same side of p on the path A, as depicted in the right drawing of Figure 14. Between u and v lies an inner vertex x of A ($x \in \{u, v\}$ being allowed) and this x is clearly also an inner vertex of C. We claim that Maker wins at x. Consider the two paths $P_1 = xAag\alpha\phi\beta$ and $P_2 = xApFqBbh\beta\phi\alpha$. Lemma 12 requires a reply in their intersection, the 2-edge ϕ plus possibly the dock vertex c. The cycle C, in which Breaker must also play, contains none of these vertices, so Maker wins. □

This concludes the analysis of the core between two closed docks. It must look exactly as shown in Figure 13.

Between a closed and an open dock. To analyze the core between a closed and an open dock, we cannot proceed as in the previous cases. If we started with a few basic connections and then added new paths provided by Lemma 6, trying to sort out winning configurations, we would never reach an end. As we shall see, there exists an infinite family of topologically different core types. So we have to take a different approach, which unfortunately comes not as naturally as the incremental one. We first present a uniform class of hypergraphs—without further motivation—and afterwards prove that the core between a closed and an open dock must come from this class.

35. Definition. A 3-uniform hypergraph L is called *a ladder of height* $h \geq 0$ *on* a_0 *and* c_0 if it can be constructed by the following procedure:

- begin with the empty hypergraph $L_0 = (\{a_0, c_0\}, \emptyset)$ on two vertices a_0, c_0;
- `for` $i = 1, \ldots, h$ `do`
  (if $h = 0$ then simply skip the loop)
  - take a new path F_i of length ≥ 2 with start vertex c_{i-1}, end vertex a_{i-1} (which are both vertices of L_{i-1}) and no further vertices common with L_{i-1};
  - denote the last inner vertex of F_i by a_i and the last outer vertex different from a_{i-1} by c_i; as shown in this figure:



    the vertices a_i and c_i will be the contact points for the next path F_{i+1};
  - let $L_i := L_{i-1} \cup F_i$;
- `either` end the construction of L by letting $L := L_h$
  `or` take an optional additional path R from c_h to some vertex r of the path $c_{h-1}F_ha_h$ except a_h (but $r = c_{h-1}$ allowed) that contains no further points of L_h and let $L := L_h \cup R$.

Figure 15 shows a ladder of height 4. The dotted bubbles indicate level sets, defined as follows. The ith level, $1 \leq i \leq h$, consists of the set $V(F_i) \setminus \{a_{i-1}, c_i\}$, i.e., the vertices of the path $c_{i-1}F_ia_i$. On level 0 lies only the vertex a_0; and the remaining vertices at the top of L, which are exactly those in $V(R) \setminus \{r\}$ or only the single vertex c_h, in case the optional path R is not present, form the highest level $h + 1$.

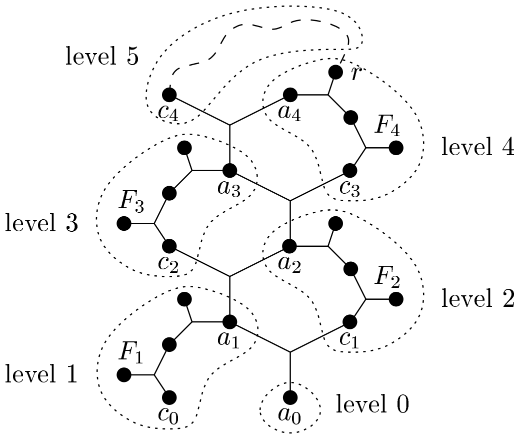


Figure 15. A ladder of height 4 with the optional top path R drawn dashed and the level sets indicated as dotted bubbles.

We let the highest level of a ladder be one above its formal height because we like to think of the vertex c_h and the optional path R as parts that do not belong to the regular structure. This convention shall turn out convenient.

The introduction of ladders is motivated by the next observation, which describes the closed-open case completely. We still face two docks $g = \{\alpha, a, c\}$ and $h = \{\beta, b, d\}$ in a hypergraph H, which we assume to be a loser. This time, g shall be closed and h shall be open, with a and c connected to b in the core K. Let J denote the connected component of K that contains the dock vertices a, b, c, extended by the vertices α and β and the dock edges g and h. We can now put all we have to say about J in one brief statement.

36. OBSERVATION. *The hypergraph J is a ladder on $a_0 = \alpha$ and $c_0 = \beta$. Its height is at least 1, and at least 2 if it does not contain the additional path R.*

Figure 16 shows such a ladder on α and β with the two contained docks arranged in the way we usually draw them.

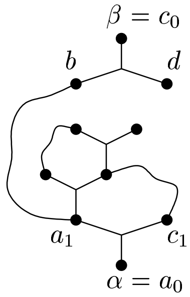


FIGURE 16. A ladder of height 3 connecting a closed lower and an open upper dock.

In order to allow the rather long and technical proof of Observation 36 to focus on the basic ideas, we prepare the main technical tools separately in advance. Like in the open-open and closed-closed case, we will argue that if J contains any further edges not in L then the whole hypergraph H must be a winner. Therefore we again need a suitable set of paths that end in a 2-edge and can thus be used as threats against Breaker. For the present open-closed case, we shall make repeated use of certain paths that connect some vertex x somewhere up in the ladder to one of the base vertices a_0 and c_0, which we define recursively as follows.

For a level-1 vertex x let

$$P_a(x) = xF_1a_0 \qquad \text{and} \qquad P_c(x) = xF_1c_0$$

denote the shortest path from x to the respective base vertex. For x on a level j with $2 \le j \le h$, let

$$P_a(x) = \begin{cases} xF_jc_{j-1}P_a(a_{j-1}) & \text{if } j \text{ even,} \\ xF_ja_{j-1}P_a(a_{j-1}) & \text{if } j \text{ odd} \end{cases}$$

and

$$P_c(x) = \begin{cases} xF_ja_{j-1}P_c(a_{j-1}) & \text{if } j \text{ even,} \\ xF_jc_{j-1}P_c(a_{j-1}) & \text{if } j \text{ odd.} \end{cases}$$

These somewhat cumbersome definitions describe rather simple geometrical objects: two kinds of paths that climb down the ladder on its left and its right rail. The paths P_a all head for a_0 while the P_c aim for c_0. The parity conditions simply take care of the alternating orientations of the paths F_i: $P_a(x)$ goes through the a_i with even i and through the c_i with odd i; for $P_c(x)$ vice-versa. Figure 17 depicts the two complementary paths $P_a(x)$ and $P_c(x)$ for a level-4 vertex x (compare to Figure 15). The path $P_c(x)$ starts from x along F_4 to its left end, from where it descends down the ladder along the left rim. Likewise, the path $P_a(x)$ climbs down the right-most edges of the ladder.

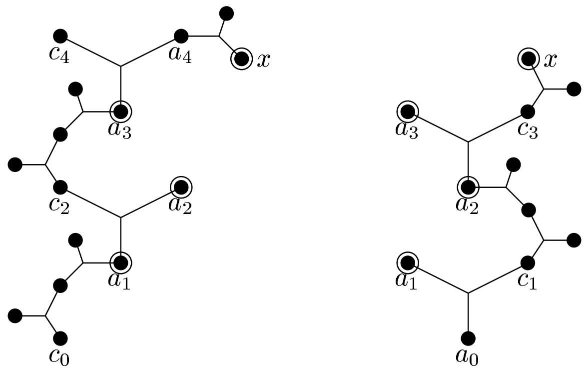


FIGURE 17. The paths $P_c(x)$ (left) and $P_a(x)$ (right) for a vertex x on level 4 of the ladder from Figure 15. Common vertices marked.

The following property makes the paths P_a and P_c useful for Maker.

37. LEMMA. *Let x be a level-j vertex of some ladder of height h. If x is an inner vertex of F_j or its starting vertex c_{j-1} then the two paths $P_a(x)$ and $P_c(x)$ intersect in no vertices other than x and all a_i with $1 \le i < j$.*

PROOF. From their starting point x on F_j the two paths in consideration head in opposite directions. (Note that in the case $x = c_{j-1}$ this is guaranteed because the level of this vertex was defined to be j, not $j-1$.) Once the two paths enter F_{j-1}, they stay on opposite sides of the ladder as far as possible. Hence, they can only intersect in the middle vertices a_i that lie below. $\square$

The paths P_a and P_c shall now be used to derive Maker wins for any configuration that deviates from a ladder shape.

PROOF OF OBSERVATION 36. Pick any inclusion-maximal ladder L on $a_0 = \alpha$ and $c_0 = \beta$ in J, which will have height at least 1 because any path from β to α can serve as the path F_1. We do not demand that L have greatest possible height but only that we cannot extend it with J-edges to a larger ladder. It might be helpful to convince oneself that this means exactly that either L contains the optional path R—which in a way seals off the top part of L—or that there is no additional path from c_h to any other vertex of F_h; although formally, this fact shall not be needed in this proof.

So assume for contradiction that $J \supsetneq L$. As before we either employ Lemma 23 to get a J-path P between two distinct vertices of L or, in the degenerate case, we find a single edge $e \in E(J) \setminus E(L)$ with $e \subseteq V(L)$. Let j be the lowest level touched by P respectively e. We distinguish different possible contact configurations.

If the second contact point of the path P lies also on level j or, in the degenerate situation, if at least one further vertex of the additional edge e does, then Maker wins as follows. Denote the two contact points of P and L by u and v. In the degenerate case, pick $u, v \in e$ such that the third vertex of e does not lie between u and v on the path F_j (respectively R, if $j = h+1$). Then there exists an inner vertex x of F_j respectively R between u and v. See Figure 18. This x is an inner vertex of the cycle $C = xF_juPvF_jx$ respectively $C = xRuPvRx$, which in either case contains no vertices on levels strictly less than j. (Observe that calling c_j a level-$(j+1)$ vertex was again necessary to guarantee that the cycle C cannot use the edge $\{a_{j-1}, a_j, c_j\}$.) For the case $j = 1$ we note that C does surely not contain c_0 because otherwise it would include the upper dock, making it a closed dock.

Now Maker plays at x. By Lemma 15, Breaker's reply must be in C but Lemma 12 prompts for an answer in each of the paths $P_a(x)a_0\phi c_0$ and $P_c(x)c_0\phi a_0$. By Lemma 37 the intersection of these two paths and the cycle C contains no vertex other than x, so Maker wins.

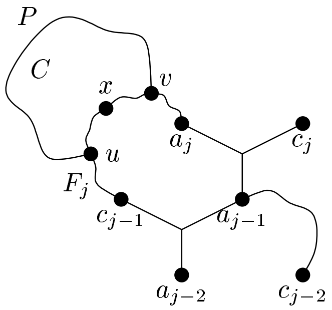


FIGURE 18. Both contact points u and v on level j.

Our analysis of the situation where there is only one contact point, u, say, on the lowest contact level $j \geq 1$ splits into two cases. First the general case: $j < h$. The union of all paths F_i with $i > j$ together with the path R (provided it is present), i.e., the induced subhypergraph of L on all vertices on levels above j and the vertex a_j, forms a connected subhypergraph M of L, shown in Figure 19. Pick a shortest path Q in M from a_j to v, the second contact point of the new path P (respectively e) and L, which must lie in M because u is the only contact on level j. If there is a third contact point w, relabel v and w if necessary, such that v lies no farther from u than w, so that by almost disjointness w does not lie on Q. We obtain a cycle $C = a_jQvPuF_ja_j$ with a_j as an inner vertex. By construction, C contains no vertices strictly below level j. Maker plays at a_j. Just like above, Breaker is forced to answer in C but also in each of the two paths $P_a(a_j)a_0\phi c_0$ and

$P_c(a_j)c_0\phi a_0$, whose intersection contains no vertex of C, except a_j, of course. Hence Maker wins.

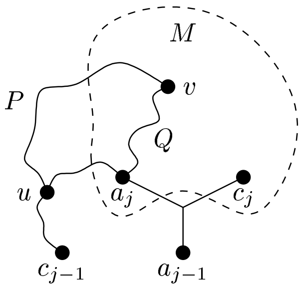


FIGURE 19. Second contact point v on a higher level.

It remains to consider the case $j = h$. ($j = h + 1$ is impossible because that would leave no higher levels for the second contact point.) First observe that L surely contains the optional path R since otherwise the new path P (or the edge e) would have to connect to the only level-$(h + 1)$ vertex c_h, forming such a path R itself, thereby contradicting the maximality of L. We know that P (resp. e) connects $u \in V(F_h) \setminus \{a_{h-1}, c_h\}$ to some $v \in V(R) \setminus V(F_h)$. See Figure 20. A possible further contact point w would also have to lie in this set, in which case we assume w.l.o.g. that v come before w on rRc_h, so that w does not lie on the path rRv.

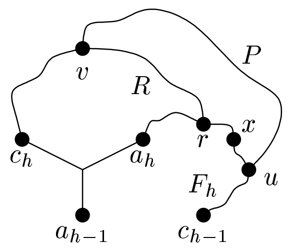


FIGURE 20. Contact points on levels h and $h + 1$.

If $u = r$ then Maker wins easily at u because this is then an inner vertex of the cycle $uRvPu$, which intersects at least one of the paths $P_a(u)$ and $P_c(u)$ only at x (depending on the parity of h). Note that u need not be an inner vertex of F_h for this to work. So we are left with the case $u \neq r$. Between u and r on F_h we find an inner vertex x of F_h ($x = u$ and $x = r$ being allowed). This x is also an inner vertex of the cycle $xF_huPvRrF_hx$, which contains no vertices strictly below level h. Like we argued before, Maker wins by playing at x because Breaker cannot play in this cycle and the two paths $P_a(x)a_0\phi c_0$ and $P_c(x)c_0\phi a_0$ at the same time.

This eventually shows that our assumption $J \supsetneq L$ must be false. The additional statements about the height of L follow immediately from the fact that the lower dock is closed. $\square$

5. Playing for Breaker

The classification into different connection types in the core started from the assumption that the whole hypergraph at hand was a loser. We do not know yet, whether any hypergraph with one 2-edge whose core uses only those connections singled out in the previous section, could perhaps be a winner. We settle this issue by proving the open implication of the following theorem.

38. THEOREM. *An almost-disjoint Maker-2-connected hypergraph with only 3-edges except exactly one 2-edge is a loser if and only if its core connections are of the following three types:*

- *between two open docks there is only a path as described in Observation 27 and shown in Figure 9 on page 44,*
- *between two closed docks the connection satisfies all properties stated in Observations 30 through 34 as shown in Figure 13 on page 48,*
- *between a closed and an open dock the connection is a ladder as stated in Observation 36 and indicated in Figure 16 on page 51.*

Elementary losers. Essentially, the task in this section will be to prove that certain hypergraphs, usually subhypergraphs of the given hypergraph at hand, are losers. Besides some side remarks along our discussion of winning paths and cycles in Section 2, we have by now not really proven any hypergraphs losers. So let us start by collecting some necessary basic facts, again about paths and cycles.

39. LEMMA. *Any almost-disjoint 3-uniform path P is a loser. Moreover, even $P^{[+u]}$ is a loser for any vertex $u \in V(P)$.*

PROOF. It suffices to prove the second, stronger statement; by induction. Let v be any Maker move in $P^{[+u]}$. Breaker can always separate u and v in the following way. If u and v do not lie in a common edge of P, Breaker plays a vertex y between them. (For example, in Figure 10 on page 44, y would be one of the two marked vertices between u and v.) Otherwise he plays the third vertex y in the edge that contains u and v.

The hypergraph $P^{[-y]}$ is then the disjoint union of two paths, where u lies in one component and v in the other. Each of those components are losers by induction and consequently, the whole graph $P^{[+u,+v,-y]}$ is a loser by Lemma 17. A length-zero path with just one vertex is trivially a loser because it contains no edges that Maker could fill. □

40. LEMMA. *An almost-disjoint cycle of 3-edges is a loser. Even more, it remains a loser if we replace one 3-edge by a 2-edge.*

PROOF. It obviously suffices to prove the second statement. (The right-hand side of Figure 4 on page 36 showed how a cycle with one 2-edge can be interpreted as a 3-uniform cycle with a Maker play at an outer vertex.) Irrespective of where in the cycle Maker plays his first move x, Breaker always takes one vertex of the 2-edge, destroying that edge. The resulting hypergraph can be interpreted as a path of 3-edges in which Maker has played one vertex, x. Hence it is a loser by Lemma 39. □

Typical applications of Lemma 39 will be configurations in which some path is only connected through a single vertex to the rest of the hypergraph. In such a situation, the Articulation Lemma tells us that we can either remove that path completely or, if it already contains a Maker move, replace it by another Maker move at the contact point. The precise conditions are captured by the following corollaries to Lemma 39.

41. Corollary. *Let $H = P \cup B$ be the union of an almost-disjoint 3-uniform path P and an arbitrary hypergraph B that have exactly one point in common. Then H is a winner if and only if B is.* $\square$

42. Corollary. *Let P be an almost-disjoint 3-uniform path and B be an arbitrary hypergraph, such that $V(P) \cap V(B) = \{p\}$ for some articulation vertex p. Let furhter x be any vertex of P. Then the union $H = P^{[+x]} \cup B$ is a winner if and only if $B^{[+p]}$ is.* $\square$

Since ladders play an important role in our classification, we shall need losing conditions for them, too.

43. Lemma. *A ladder on a 2-edge is a loser.*

44. Lemma. *Let x be a vertex on the 1st level of a ladder L on a_0 and c_0. Then the hypergraph $L^{[+x,-a_0]}$ is a loser.*

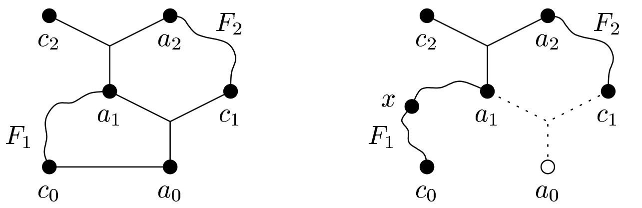


Figure 21. The ladder configurations of Lemma 43 (left) and Lemma 44 (right).

Figure 21 shows the respective configurations of these lemmas. The two statements are closely related. We prove them together by an interleaved induction.

Proof of Lemmas 43 and 44. We parameterize the lemmas by the height: $\mathcal{A}(h)$ denote the statement of Lemma 43 restricted to ladders of height h and $\mathcal{B}(h)$ denote the statement of Lemma 44 restricted to ladders of height h. We perform a mixed induction on h by reducing $\mathcal{B}(h)$ to $\mathcal{A}(h-2)$, and $\mathcal{A}(h)$ to $\mathcal{A}(k)$ and $\mathcal{B}(\ell)$ with $k < h$ and $\ell \leq h$. Note that this avoids circular arguments although $\mathcal{A}(h)$ may use $\mathcal{B}(h)$, because $\mathcal{B}(h)$ does not rely on $\mathcal{A}(h)$.

Induction bases. Since a height-0 ladder on a 2-edge is just that 2-edge, $\mathcal{A}(0)$ is obviously true, and $\mathcal{B}(0)$ is true simply because the respective hypergraph does not contain any edges on which Maker could win. Let us also treat $\mathcal{B}(1)$ at this point to take care of some irregularities which result from the path R. If the optional path R is not present, the hypergraph is just the path F_1 with one vertex played, a loser by Lemma 39. If R is present, we can simply remove it because Breaker's move a_0 has destroyed the second contact point c_1 of R and F_1. So we get the same loser as before. (Though Figure 21 shows the regular path F_2 instead of the path R, one can still see there that the rightmost path can be deleted because of Breaker's move at a_0.)

The induction step for $\mathcal{B}(h)$, $h \geq 2$, works similarly. We use Corollary 42 to replace the path F_1 by a single Maker move at a_1. Then we delete the dangling path F_2 (see right of Figure 22) by Corollary 41. What's left is a ladder of height $h-2$ on the new 2-edge $\{a_2, c_2\}$.

Induction step for $\mathcal{A}(h)$, $h \geq 1$. If Maker plays his first move x on level 0, i.e., $x = a_0$, then Breaker answers at c_0. We can then delete the path $c_0 F_1 a_1$ (or a slightly shorter path up to r if $h = 1$ and R is present). This leaves a ladder on the 2-edge $\{a_1, c_1\}$, a loser by induction.

If Maker's first move x is on level h or $h+1$, Breaker answers at a_{h-1}. This disconnects levels h and $h+1$ from all lower levels. See the left-hand side of Figure 22. If x lies on level h, the top part is a loser by $\mathcal{B}(1)$ and if x lies on R then we can remove most of F_h so that the rest of the top part is a loser by Lemma 39. The lower part is (after removal of the dangling path F_{h-1}) a ladder of smaller height, hence also a loser by induction.

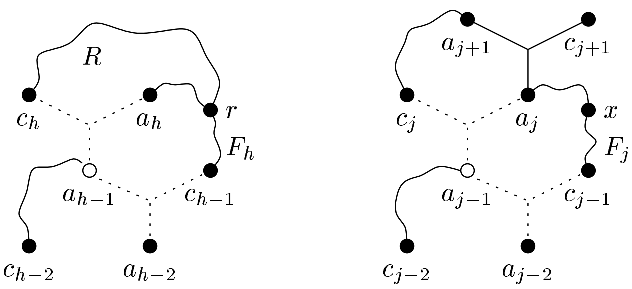


FIGURE 22. Maker plays x on level h or $h+1$ (left) and Maker plays on an intermediate level $j < h$ (right).

We turn to the general case: Maker x on a level j with $1 \leq j < h$. In this situation Breaker plays a_{j-1}. See the right-hand side of Figure 22. As in the previous situation, the ladder breaks up into a lower and an upper part, the former again (after removal of the dangling path F_{j-1}) being a shorter ladder on the 2-edge $\{a_0, c_0\}$, a loser by induction. The upper part can be interpreted as a ladder on a_{j-1} and c_{j-1} with a_{j-1} already played by Breaker and the vertex x (now on level 1) already played by Maker. A loser by induction. $\square$

Almost all arguments during our classification in Section 4 were in a sense written out of Maker's perspective. Usually, we proved that some configuration cannot occur in a loser by presenting a winning strategy for Maker. The case distinctions were set up in such a way that in each step we could derive a Maker win with very few explicit moves—often just one—by listing several threats in the form of paths and cycles, that could not all be countered by Breaker at the same time.

The present situation is very different. We want to show that Maker cannot win on certain hypergraphs. So we pick good Breaker moves and must, in principle, provide counters against *all* possible Maker attacks. The obvious problem here is: Breaker has no threats; by the very definition of the game.

In the proof of the two preceding ladder lemmas, we could exploit the strong symmetry of ladders, which allowed an induction. The question now is: How to get control over all possible Maker strategies on the whole hypergraph H? The key is again the central role of the 2-edge ϕ. If we manage to get a Maker or Breaker move into that edge, the hypergraph will lose its Maker-2-connectivity. Precisely, if β is taken then α becomes an articulation vertex, which makes the hypergraph amenable to an application of the Articulation Lemma to break it into smaller parts. The resulting components will then be simple enough to be analyzed by the above lemmas about paths, cycles, and ladders.

The basic components. Let us collect all such components that arise when Breaker plays one vertex of the 2-edge ϕ, at β, say. Precisely, we list all types of hypergraphs M such that $H^{[-\beta]}$ can be written as a union $M \cup D$ with $V(M) \cap V(D) = \alpha$ and such that α is not an articulation vertex of M, i.e., we only consider minimal components.

First observe that such a component M contains no more than 3 docks because any lower dock g is connected to at most two upper docks and in $H^{[-\beta]}$ any upper dock vertex is connected to at most one lower dock. Closed upper docks have unique lower partners anyway and all open docks are destroyed at β so that they no longer link their partners on the lower shore.

Out of the three connection types from the previous section, we assemble again three essentially different types of such components M.

(i) Two connected closed docks, where the upper dock has been destroyed. See the upper left of Figure 23.
(ii) A closed lower dock connected to an open upper dock. This is simply a ladder, shown on the upper right of Figure 23.
(iii) An open lower dock between two closed upper docks, which both have their base point $a_0 = \beta$ deleted. This is the union of two ladders with the base point a_0 deleted in each, glued together on the first edge of their F_1-paths. See the lower part of Figure 23.

The remaining possibilities of an open lower dock between two open upper docks or one open and one closed upper dock, or an open lower dock linked to just one upper dock, can be interpreted as subhypergraphs of configurations covered by case 3 since a path to an open dock can be seen as the first level of a ladder. So we omitted them from the above list since it will suffice just to observe that all relevant properties of components of type (iii) will carry over to them.

The base case. Our analysis of possible Maker moves begins with the easiest situation, where Maker takes α and Breaker gratefully answers at β so that afterwards everything is nicely decomposed. Although this is a very special case, it forms the basic result to which we shall later reduce all the other possible Maker plays.

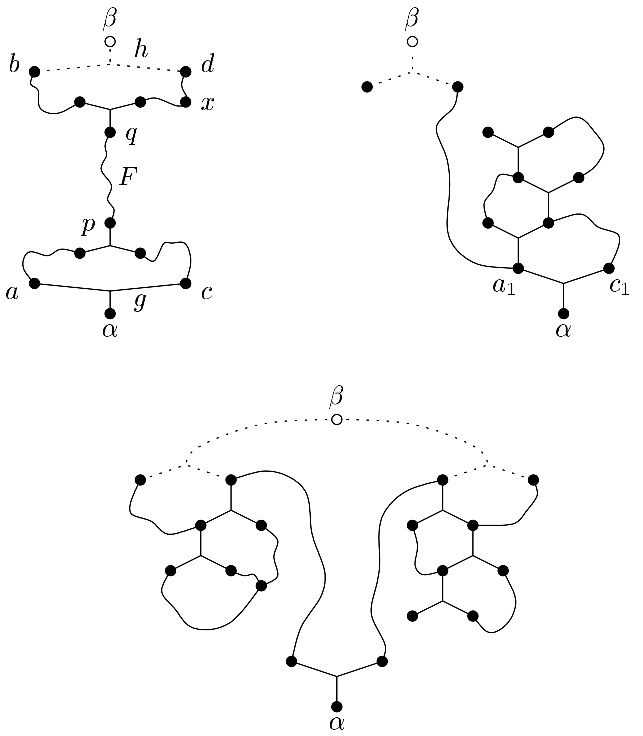


FIGURE 23. Components of $H^{[-\beta]}$.

45. OBSERVATION. *If in the first move each player takes one vertex from the 2-edge ϕ, the game is lost for Maker.*

PROOF. To go conform with the above classification, we assume by symmetry that Maker has played at α and Breaker has answered at β. We now simply go through our list and verify for each type whether $M^{[+\alpha]}$ is a loser.

Case (i). Two closed docks. Maker's move has produced a 2-edge in the lower cycle. Two applications of Corollary 41 remove the upper cycle entirely, together with the path in the middle, leaving only the lower cycle which is lost by Lemma 40.

Case (ii). A closed lower dock connected to an open upper dock. We interpret the ladder as sitting on the two dock vertices of the lower dock, which are now linked by a 2-edge. This is a loser by Lemma 43.

Case (iii). An open lower dock between two closed upper docks. The two ladders overlap on the lower dock. We shorten one ladder by this edge so that afterwards they only touch on one vertex. Then one ladder contains the additional Maker vertex α while the other does not. Applying the Articulation Lemma to this common point, we see that the whole component must be a loser by Lemma 44.

The remaining cases are covered by case 3, as remarked above. $\square$

Although Observation 45 deals with only two very special first Maker moves, it is the essential step towards the proof of Theorem 38. In the following we check all possible first Maker moves outside of ϕ. The analysis is again split into the old three classes: whether Maker plays between two open docks, between two closed docks, or between an open and a closed dock; the classification above, into components M of $H^{[-\beta]}$, will be used as a tool only.

The general scheme is the same for all cases. Breaker answers Maker's move x by a move in the 2-edge ϕ, at β, say. Then α has become an articulation vertex, so we can write

$$H^{[-\beta]} = M \cup D \quad \text{with} \quad M \cap D = \{\alpha\}$$

and such that M contains Maker's vertex x, which we technically consider as not deleted for a second to get a sound definition of M. The component M is then of one of the three types in Figure 23.

Now comes the decisive trick. We show two things: $M^{[+x]}$ is a loser but $M^{[+x,+\alpha]}$ is a winner. Then by the Articulation Lemma, this implies that the whole hypergraph $H^{[+x,-\beta]}$ is a winner if and only if $D^{[+\alpha]}$ is a winner! But $D^{[+\alpha]}$ is by construction a subhypergraph of $H^{[-\beta,+\alpha]}$. Note that we don't have to put an additional $+x$ in the exponent because the vertex x lies not in D. Now Observation 45 tells us that this hypergraph is lost, so we are done.

What we did in the previous paragraph could be termed less formally in the following way. When we know that $M^{[+x]}$ is a loser but $M^{[+x,+\alpha]}$ is a winner, the Articulation Lemma tells us that α is a reasonable move for Maker. Since he cannot win on $M^{[+x]}$ he makes the best of this part by playing the threat α which turns it into a winner. Now, since we may legitimately assume that Maker will play at α, the problem is reduced to the question whether the rest $D^{[+\alpha]}$ is a winner. Which, as we know, is not.

46. Observation. *If Maker plays his first move between two open docks (including the respective dock vertices) he loses.*

Proof. Breaker answers Maker's move x by playing at β, destroying the upper docks. We write $H^{[-\beta]} = M \cup D$ as described above, where M contains two open docks, so it's type is one of those subtypes of case (iii) in our classification.

Clearly $M^{[+x,+\alpha]}$ is a winner, and since M is a subhypergraph of a type-(iii) component, Lemma 44 tells us that it is a loser. As described above, we conclude that the whole hypergraph $H^{[+x,-\beta]}$ must be a loser. □

47. Observation. *If Maker plays his first move between two closed docks (including the respective dock vertices) he loses.*

Proof. Breaker again takes a vertex from the 2-edge. He has to be a little careful with his choice, however. Have a look at Figure 13 from page 48 again. If Maker's first move x is a vertex of the lower path A then Breaker replies at α, breaking the lower cycle. Likewise, Breaker answers a move in the upper path B at β. In the remaining case $x \in V(F)$ he picks one of α and β arbitrarily. (In the special case when F has length 0 and Maker plays the unique contact vertex in $V(A) \cap V(B)$, we also let Breaker pick one of α and β at will.)

Assume by symmetry that Breaker plays β, i.e., x was played on the upper cycle or the connecting path F. (Have a look again at Figure 23, where the vertex x was already marked.) As the upper cycle has been broken, we can apply Corollaries 41 and 42 to replace the complete upper part $B \cup F$ by a single Maker move at the contact point $p \in V(B)$. Then Lemma 40 tells us that the resulting cycle $A^{[+p]}$ is lost. In terms of our general recipe, we have thus shown that $M^{[+x]}$ is a loser. On the other hand, $M^{[+x,+\alpha]}$ is clearly a winner. Again the general argument described above now settles the issue. □

The remaining closed-open case again bears a difficulty. The general argument we used in the previous cases will only work for the special situation that Maker's move is on the first level of the ladder. (Recall that the core between an open and a closed dock is a ladder.) Plays at higher levels require an inductive argument and are deferred to the moment when we compile all our observations into a proof of Theorem 38.

48. OBSERVATION. *If Maker plays his first move on the first level of the ladder between a closed and an open dock, he loses.*

PROOF. Assume by symmetry that the lower dock is the open one. Breaker plays at β, destroying all upper docks. Then we know that the resulting component M that contains x is of type (iii) (or a subhypergraph with just one ladder) with Maker's move x on the first level of one ladder. Lemma 44 tells us that $M^{[+x]}$ is a loser and $M^{[+x,+\alpha]}$ is as always trivially a winner. □

The alert reader might have noticed that case (ii) of our classification did not show up in the last three observations. This does not mean that it has been overlooked. It simply was not needed for the proofs. Remember that Observations 46 to 48 are statements about the three connection types from Theorem 38, they only *used* the three M-types from this section as a tool.

Eventually, almost all details of Theorem 38 have been studied. It is time to put our observations together.

PROOF OF THEOREM 38. That the core of a loser can only have the listed connection is obviously true, simply because they are just those types that survived our lengthy discussion from Section 4.

The converse almost follows from Observations 45 to 48. They provide successful Breaker counters against all first Maker moves except for a play on a higher level of a ladder between a closed and an open dock.

This remaining case is the only situation where Breaker must not play in the 2-edge ϕ. Instead, he chops a few steps off the ladder. We prove that Breaker wins if Maker plays on a level $j \geq 2$ of some ladder between a closed and an open dock by induction on the sum S of the heights of all ladders in the core.

At the induction base $S = 0$ there are no ladders, so the statement is trivially true. For the induction step, we let Breaker answer Maker's move x at a_{j-1}, just like in the proof of Lemmas 43 and 44. (See the right-hand side of Figure 22 from page 57 again.) This decomposes the ladder into an upper and a lower part such that the upper is lost by Lemma 44 and the lower remains, after removal of the dangling path F_{j-1}, a ladder of smaller height. Since Maker's move x does not lie in the lower part, we have reduced the original hypergraph to one that still satisfies all requirement of our Theorem but has smaller ladder-height sum S. This finishes the proof. $\square$

The algorithm. It is time to return to our initial complexity question. In the following proof of Theorem 4 we compile the results of the preceding sections into a polynomial-time algorithm for the decision problem of winning and losing. This is a straightforward procedure, simply revisiting all reduction steps and showing that the core types from Theorem 38 are checkable efficiently. We emphasize again that a detailed runtime analysis of the below method is not our goal. Neither do we strive for an actual implementation of the described procedures nor for an improvement of asymptotic runtime bounds. Theorem 38 is a purely qualitative result, identifying the games at hand as a *tractable* subclass of general hypergraph games.

PROOF OF THEOREM 4. Let H be the given almost-disjoint hypergraph of rank-3. By Lemma 17 we can assume that H is connected. If H contains more than one 2-edge, it is a winner by Corollary 11. If it contains no 2-edges, we create all first-move hypergraphs $H^{[+x,-y]}$ with $x, y \in V(H)$ as described in Section 3 in connection with Lemma 24. This produces a quadratic number of hypergraphs, amongst which we have to check those that contain a 2-edge for winning or losing.

All hypergraphs with one 2-edge can be severed at articulation vertices, as described in Lemma 25, until we are left with Maker-2-connected hypergraphs only, each of which contains exactly one 2-edge. (Remember that whenever this process produces two 2-edges, we are done by Corollary 11.)

The core of each of those Maker-2-connected hypergraphs is then decomposed into links between the docks, as we did in Section 4. For each such link we check whether it complies with the specifications of Theorem 38 to see if Maker can win. This is not a difficult task. Each admissible connection type is expressed in terms of paths that are built upon each other. We can use a simple greedy path-finding method to successively reconstruct any required or allowed connection. Whenever we spot a violation of the admissible topology we know that we face a winner. $\square$

6. Almost-Disjointness

We promised some comments on the influence of the almost-disjointness restriction on our games on rank-3 hypergraphs. Have a look at the two overlapping 3-edges in Figure 24, who violate this condition. Assume this configuration occurs within a hypergraph H in such a way that no further edges touch upon the vertices a and b, so that our edge pair is linked to the rest of H only through p and q. We claim that in such a configuration the two 3-edges are of no use for Maker.

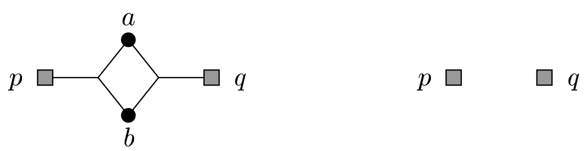


FIGURE 24. Two worthless 3-edges.

49. Lemma. *A hypergraph H containing the left configuration of Figure 24 with no further edges connected to a and b is a winner iff $H^{[-a,-b]}$, the same hypergraph with this configuration replaced by the one to the right, is a winner.*

Proof. The hypergraph $H^{[-a,-b]}$ is a subhypergraph of H, so if Maker wins on the former he clearly also wins on the latter. We show that a making strategy σ for H on the left yields also a Maker win on the reduced hypergraph on the right. Therefore we follow this strategy on both hypergraphs, copying our Maker moves given by σ from the left to the right and the Breaker answers, which are played on the right, back to the original hypergraph H. This works fine as long as our strategy σ does not prompt us to play at a or b. In that case, if we must play a, say, we actually do so on the left and then—this is the trick—answer it immediately by a fake Breaker move at b. In the reduced hypergraph on the right side, these two half-moves are simply left out. After a and b are taken on the left, we can continue with σ until the whole board is full.

Who has won? Since we followed the winning strategy σ on the left, we must have won there, i.e., some edge $e \in V(H)$ is completely ours. But since we have given Breaker a vertex in each of the two 3-edges on a and b, this winning edge is neither of them. Consequently, we have also occupied all vertices of e on the right. □

A similar situation—or rather the opposite—is shown in Figure 25. Again the two edges are part of some bigger hypergraph H in such a way that no further edge contains a or b and everything else is linked through p and q, who now are the inner vertices of this little cycle.

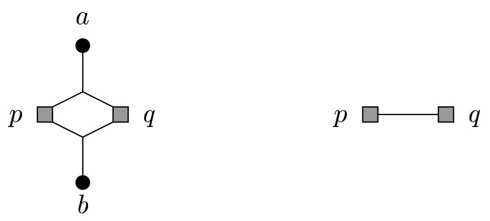


Figure 25. Two 3-edges behave like a single 2-edge.

50. Lemma. *A hypergraph H containing the left configuration of Figure 25 with no further edges connected to a and b is a winner iff $H^{[+a,+b]}$, the same hypergraph with this configuration replaced by the one to the right, is a winner.*

Proof. Assume a making strategy σ for the left hypergraph H. As above we follow σ on the right until a move in $\{a, b\}$ is required. In this case, play this vertex, a, say, and as above, reply by a fake Breaker move at b. This deletes one of the two 3-edges and turns the other one into a 2-edge on p and q. From then on we just pursue σ again on both sides to the end of the game. As in the proof of Lemma 49, we conclude from the fact that σ has lead to a win on the left that we have also won on the right because all edges on the left are also present on the right. The newly created 2-edge is just the one that was present on the right in the first place.

The other implication works very similar, with exchanged sides. Assume we have a making σ on the right. Against Breaker on the left we also follow σ—until Breaker takes one of the vertices a and b. (We won't play there first because our strategy does not know those vertices.) In this case, we take the other vertex and then resume our strategy σ again. Just as above the two hypergraphs are now completely identical, so we win on the left because we are sure to win on the right. □

Let us call a pair of two 3-edges that overlap on two vertices a *diamond*. The previous discussions have shown again that the inner vertices are, as so often, the valuable ones, while the outer vertices are of minor interest.

Assume we try to find out whether some given rank-3 hypergraph that is not almost-disjoint is a winner. If we find a configuration like the one on the left of Figure 26, Maker can win if the path P connecting the two diamonds is almost-disjoint because the terminal diamonds behave like 2-edges. If P is not so nicely behaved and there sits a diamond somewhere on P, as shown on the right-hand side of the figure, we may assume that this diamond has some further edge f attached to one of its inner vertices because otherwise, we could just remove that diamond without changing the value of the game. From where f is connected, the new diamond looks like a 2-edge again; so if we trace a path from f back to one of the two terminal diamonds (using Maker-2-connectivity) we win as soon as we meet another diamond at an inner vertex.

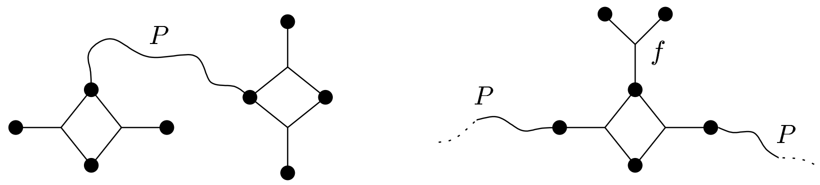


FIGURE 26. Two diamonds connected at their "good" vertices.

Though we have only just started the discussed of a simple example, it appears as if the presence of only two or three diamonds in a Maker-2-connected rank-3 hypergraph create an influence of "pseudo 2-edges" that should, in general, lead to a win like in the left of Figure 26. What this "general case" should precisely be, is of course unclear and a proper analysis appears to bring a lot of case distinctions about. Yet, this brief discussion might indicate that the problem might be solvable in a way that rids a given hypergraph from its diamonds so that we may afterwards apply Theorem 4 directly, as a black box, without unrolling the tedious proof of Theorem 38 again.

7. Comparing Games

We close this chapter by introducing a new view on positional hypergraph games that incorporates several concepts we have met so far.

Let us have a closer look at our favorite tool, the Articulation Lemma. Intuitively, it tells us that the two halves of a hypergraph that are only connected through a single articulation vertex, can interact in only three different ways. So in a sense, seen through an articulation, there exist only three different types of hypergraphs: those halves A that win on their own, those that do not help the B on the other side at all, and those "semi-winners" who are not winners themselves but for which $A^{[+p]}$ is a winner. Cutting such a hypergraph in two at the articulation, we get an isolated "half" with a marked contact point.

51. Definition. A *pointed hypergraph* is a pair (H, p) of a hypergraph $H = (V, E)$ and a *point* $p \in V$. The *one-point union* $(A, p) \sqcup (B, q)$ of two pointed hypergraphs (A, p) and (B, q) is the pointed hypergraph

$$\big((A \mathbin{\dot\cup} B)/\{p = q\}, \{p, q\}\big),$$

meaning that we take the disjoint union of A and B and then identify the two points p and q, choosing this merged vertex as the point of the union.

The term "one-point union" is borrowed from topology, confer [**10**, Chp. 1, Sec. 13]. Sometimes, when the precise choice of the point is not relevant, we shall treat a pointed hypergraph just as a hypergraph, simply ignoring the point, speaking of winners and losers, for example.

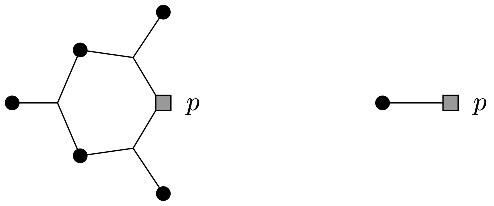


Figure 27. Two equivalent pointed hypergraphs.

Of course, we want to *play* on such one-point unions. Compare the two pointed hypergraphs in Figure 27. We claim that with respect to composition at the point p, these pointed hypergraphs have the same value in any game. Whatever partner (X, q) you plug in at p from the right, either you win on both one-point unions or on neither of them. We defer the proof of this statement until we have prepared suitable notions, which shall allow for a much more general result.

The partial order $\mathcal{H}_1$. Define a partial quasi-order on the class of all pointed hypergraphs by letting

$$A \le B$$

for two pointed hypergraphs A, B iff

$$A \sqcup X \text{ is a winner} \Rightarrow B \sqcup X \text{ is a winner} \tag{13}$$

for all pointed hypergraphs X.

This relation is obviously reflexive and transitive but clearly not anti-symmetric. Call A and B equivalent if $A \le B$ and $B \le A$, denoted by $A \equiv B$. We define $\mathcal{H}_1$ to be the partially ordered set that results from identifying equivalent pointed hypergraphs.

This notion of equivalence captures all information about a pointed hypergraph with respect to 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 $B \equiv A$ without changing Makers prospects of winning—independent of the partner X. Note that by the very definition of $\leq$, two pointed hypergraphs A and B are not equivalent iff there exists some "separating" pointed hypergraph Z such that $A \sqcup Z$ is a winner but $B \sqcup Z$ is a loser or vice versa. So with respect to this Z, the pointed hypergraphs A and B show a different behavior.

What can we say about $\mathcal{H}_1$? First note that it contains a maximal and a minimal element. Any winner A with any vertex $p \in V(A)$ as its point, is greater or equal than any other pointed hypergraph. Hence, there is a maximal element 1 in $\mathcal{H}_1$ that contains all pointed winners. To see that it contains only winners, consider some winner A together with an arbitrary loser B and let U be a pointed hypergraph without any edges. Then $A \sqcup U$ is a winner while $B \sqcup U$ is still a loser. Hence, $A \not\leq B$. This means that no loser lies above any winner and consequently the class 1 contains only winners (each with an arbitrary vertex as point). This observation allows us to abbreviate the expression "A is a winner" as $A \in 1$.

A similar argument shows that $\mathcal{H}_1$ has a minimal element, 0, which contains all *absolute losers*—pointed hypergraphs that do not contribute anything. All empty graphs, like U from above, fall into this class. Trivially, because whenever $U \sqcup X$ becomes a winner for such a U and some X then X alone must already be a winner. Hence, for any pointed C the one-point union $C \sqcup X$ is also a winner and thus $U \leq C$. Note that unlike the case of the maximal element, 0 is *not* the class of all losers but much smaller. So $U \in 0$ is really a stronger statement than saying that U is a loser!

What lies between 0 and 1 in $\mathcal{H}_1$? The answer is simple, we already know. The following theorem is the Articulation Lemma in disguise.

52. THEOREM. *The poset $\mathcal{H}_1$ is a linear order of exactly three elements.*

PROOF. We show that $A \equiv B$ for any two arbitrary pointed hypergraphs, neither of which is a winner nor an absolute loser, i.e., $A, B \notin \{0, 1\}$. Then we know that there can be at most one further class besides 0 and 1.

Since $B \notin 0$, there exists a $Y \notin 1$ with $B \sqcup Y \in 1$. Then the Articulation Lemma tells us that $B^{[+q]}$ must be a winner, where q be the point of B. On the other hand, we know that for any X with $A \sqcup X \in 1$ the reduction $X^{[+p]}$ must be a winner (p being the point of X), also by the Articulation Lemma, because $A \notin 1$. Together this means that for any such X the composition $B \sqcup X$ is also a winner. Hence, $A \leq B$. Exchanging the roles of A and B we also obtain the converse relation and therefore, $A \equiv B$.

To see that a third class in $\mathcal{H}_1$ exists at all, simply note that the 2-edge in Figure 27 is neither a winner nor an absolute loser. $\square$

Our original claim about the two pointed hypergraphs from Figure 27 is now almost proven. We just argued that the single edge lies in the unique intermediate class of $\mathcal{H}_1$. By Lemma 40 the cycle on the left is no winner either and it also no absolute loser because it gives a win if composed with itself. Hence, by Theorem 52 the two pointed hypergraphs must lie in the same equivalence class. The whole order $\mathcal{H}_1$ is shown in Figure 28, with a typical representative for each class.

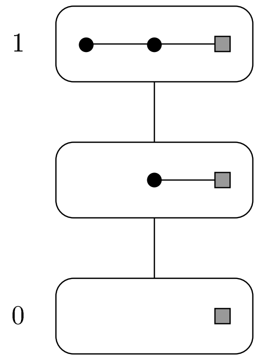


FIGURE 28. The poset $\mathcal{H}_1$.

Merging along many points. One can generalize the union at just one point to amalgamations along larger sets. Actually, the index of $\mathcal{H}_1$ already calls for the following definitions.

53. DEFINITION. A *k-pointed hypergraph* is a tuple $(H, p_1, \ldots, p_k)$ consisting of a hypergraph $H = (V, E)$ and a list of 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

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

meaning that we take the disjoint union of A and B and then merge each individual point pair $\{p_1, q_1\}$ through $\{p_k, q_k\}$ into a single new point.

Our partial quasi-order generalizes naturally by letting $A \leq B$ for two k-pointed hypergraphs iff (13) holds for all k-pointed hypergraphs X. Then $\mathcal{H}_k$ is defined as the partially ordered set of equivalence classes of k-pointed hypergraphs with the order induced by $\leq$.

As an example for 2-pointed hypergraphs we remark that we have already worked with the partial order $\mathcal{H}_2$: in the previous section on almost-disjointness. The reader will have already noticed the similarity of Figure 27 with Figures 24 and 25 from pages 62 and 63. This is, of course, no coincidence. Phrased in our new terminology, the respective Lemmas 49 and 50 are actually equivalence proofs for 2-pointed hypergraphs.

As with $\mathcal{H}_1$ we observe that each $\mathcal{H}_k$ has a maximal element 1, which contains exactly all winners, and a minimal element 0, the class of absolute losers. The respective arguments are exactly the same as for the case $k = 1$ above. We note that the degenerate case $k = 0$ has already appeared, in form of Lemma 17. With no points, $A \sqcup B$ is just $A \dot\cup B$ and therefore the dichotomy of Lemma 17 applies: $\mathcal{H}_0$ consists of only two classes, 0 and 1. (Here losers are always absolute losers.)

Can we say anything more about $\mathcal{H}_k$ for $k \geq 2$? Unfortunately, our knowledge amounts to pretty little. We have the following basic lower bounds.

54. PROPOSITION. *For each $k \geq 0$, the partial order $\mathcal{H}_k$ contains a chain of length $k+2$.*

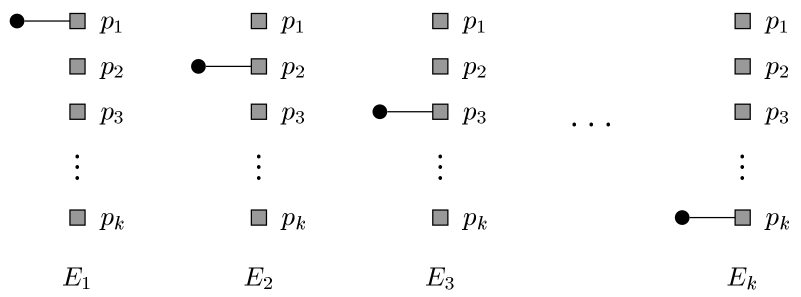


FIGURE 29. Some basic k-pointed hypergraphs.

PROOF. From the basic k-pointed hypergraphs E_i in Figure 29 we construct a chain of length $k+2$ in $\mathcal{H}_k$ as follows. Let U_r denote the k-point union $E_1 \sqcup \cdots \sqcup E_r$ of the first r such hypergraphs, $0 \leq r \leq k$. So the k-pointed hypergraph U_r contains exactly r independent 2-edges on the points p_1 through p_r, and $k-r$ isolated points. Further let U_{k+1} be an arbitrary k-pointed winner. We have

$$U_0 < U_1 < \cdots < U_k < U_{k+1} \quad \text{in } \mathcal{H}_k$$

because for each $r \leq k$ the k-point union $U_r \sqcup E_r$ is a winner while $U_{r-1} \sqcup E_r$ is obviously lost; and U_{k+1} is larger than all the other U_r. $\square$

55. PROPOSITION. *For each $k \geq 1$, the partial order on $\mathcal{H}_k$ contains an antichain of length $\binom{k}{\lfloor k/2 \rfloor}$.*

PROOF. For each index set $I \subseteq \{1, \ldots, k\}$ of cardinality $\lfloor k/2 \rfloor$ we let U_I denote the composition of all E_i with $i \in I$. For any pair $J \neq J'$, the k-pointed hypergraphs U_J and $U_{J'}$ are incomparable because for $r \in J \setminus J'$ the composition $U_J \sqcup E_r$ is a winner but $U_{J'} \sqcup E_r$ is not, i.e., $U_J \not\leq U_{J'}$; and likewise, any $E_{r'}$ with $r' \in J' \setminus J$ shows that $U_J \not\geq U_{J'}$. $\square$

These basic calculations might give us some first feeling for the complexity of the $\mathcal{H}_k$. However, they do not address the important point. The crucial question is:

Are all $\mathcal{H}_k$ finite?

If some $\mathcal{H}_k$ is finite then so are all $\mathcal{H}_j$ with $j \leq k$, obviously, because any $\mathcal{H}_j$ is embeddable in $\mathcal{H}_k$ by adding $k-j$ isolated dummy points to any j-pointed hypergraph. We know that $\mathcal{H}_1$ is finite. Is there a level in the hierarchy $(\mathcal{H}_k)$ where the complexity explodes from finite to infinite? If so, this should probably happen quite early, maybe on level two or three. However, any such statement appears to be difficult to prove.

The finiteness of $\mathcal{H}_k$ would have strong implications on the complexity of weak positional games on hypergraphs that are only Maker-k-connected. Such a hypergraph H can be written as a nontrivial union of two subhypergraphs A and B who overlap on no more than k vertices. If we interpret A

and B as k-pointed hypergraphs with these vertices as points, we can write $H = A \sqcup B$. If $\mathcal{H}_k$ should be finite we could, in principle, identify the equivalence classes of A and B *independently*—by solving the k-point unions $A \sqcup X$ and $B \sqcup X$ for a complete set of representatives X of $\mathcal{H}_k$. The outcomes of those subproblems would then tell us the value of H. This way we decompose the big problem whether H is a winner into a constant number of such questions for smaller hypergraphs. (Note that the size of the representatives is bounded.) For a decision problem that is PSPACE-complete in general, this would be quite a remarkable result: we could divide and conquer with very little overhead.

Actually, we have used this principle already extensively throughout this chapter—for the case $k = 1$. In Section 3 we repeatedly cut at articulations until we obtained Maker-2-connectivity. Each decomposition step used implicitly, through the Articulation Lemma, the fact that $\mathcal{H}_1$ contains only three classes, one of which could always be excluded because of the existence of a 2-edge in one half.

I have constructed an approximation of $\mathcal{H}_2$ that carries a lot of symmetries and which might already be the complete picture but I see by now no way of proving such a statement. Intuitively, finiteness of $\mathcal{H}_k$ means that through only k points, the two halves cannot exchange an arbitrary amount of information. It should be that during a play across a small interface, the points soon get congested—until the graph eventually decomposes into completely disjoint parts. I am strongly convinced of the following.

56. CONJECTURE. *The poset $\mathcal{H}_k$ is finite for every $k \geq 0$.*