



HAL
open science

Hypergraphs and the Maker-Breaker game: a structural approach

Florian Galliot

► **To cite this version:**

Florian Galliot. Hypergraphs and the Maker-Breaker game: a structural approach. Computer Science and Game Theory [cs.GT]. Université Grenoble Alpes [2020-..], 2023. English. NNT: 2023GRALM028 . tel-04249805

HAL Id: tel-04249805

<https://theses.hal.science/tel-04249805v1>

Submitted on 19 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : MSTII - Mathématiques, Sciences et technologies de l'information, Informatique

Spécialité : Mathématiques et Informatique

Unité de recherche : Institut Fourier

Hypergraphes et jeu Maker-Breaker : une approche structurale

Hypergraphs and the Maker-Breaker game: a structural approach

Présentée par :

Florian GALLIOT

Direction de thèse :

Sylvain GRAVIER

DIRECTEUR DE RECHERCHE, CNRS, Université Grenoble Alpes

Directeur de thèse

Isabelle SIVIGNON

CHARGÉE DE RECHERCHE, CNRS, Université Grenoble Alpes

Co-encadrante de
thèse

Rapporteurs :

Milos STOJAKOVIC

FULL PROFESSOR, Université de Novi Sad

Eric DUCHENE

PROFESSEUR DES UNIVERSITES, Université Lyon 1

Thèse soutenue publiquement le **3 juillet 2023**, devant le jury composé de :

Milos STOJAKOVIC

FULL PROFESSOR, Université de Novi Sad

Rapporteur

Eric DUCHENE

PROFESSEUR DES UNIVERSITES, Université Lyon 1

Rapporteur

Louis ESPERET

DIRECTEUR DE RECHERCHE, CNRS, Université Grenoble Alpes

Examineur

Valia MITSOU

MAITRE DE CONFERENCES, Université Paris-Cité

Examinatrice

Nadia BRAUNER

PROFESSEUR DES UNIVERSITES, Université Grenoble Alpes

Examinatrice

Présidente du jury

Invités :

Sylvain GRAVIER

DIRECTEUR DE RECHERCHE, CNRS, Université Grenoble Alpes

Isabelle SIVIGNON

CHARGÉE DE RECHERCHE, CNRS, Université Grenoble Alpes



Abstract

This PhD thesis discusses games and hypergraphs. In particular, our focus is on games played on hypergraphs, whose analysis ends up leading to the study of hypergraph problems that are interesting on their own. Positional games involve two players alternately claiming vertices of a hypergraph, with objectives that can vary: for instance, tic-tac-toe is an example where each player wants to be the first to pick all the vertices of some edge. In the Maker-Breaker convention, one player ("Maker") wants to pick all the vertices of some edge, while the other ("Breaker") wants to prevent this from happening. The main theme is the study of hypergraph structures and their implications for the Maker-Breaker positional game. We are especially interested in hypergraphs of rank 3: this means all edges are of size at most 3, which allows for some structural results and helps understand the Maker-Breaker game in this case. Another theme is the design of polynomial-time algorithms to solve games and hypergraph problems. These two themes are very much linked together, since any structural characterization yields an algorithm which is as efficient as the structure in question is simple.

Chapter I presents all notions featured in this dissertation, as well as preliminary results. We provide some state of the art around positional games, and we then introduce our structural approach to the Maker-Breaker game, centered on a notion of dangers created by Maker which Breaker must destroy immediately. Some elementary structures in hypergraphs, particularly in hypergraphs of rank 3, are also studied. Chapter II carries out the in-depth structural studies at the core of this dissertation. We obtain a structural characterization for the outcome of the Maker-Breaker game on hypergraphs of rank 3, as well as optimal strategies for both players, all based on danger intersections. A direct link appears with a particular connectivity problem in hypergraphs, for which we give a fine structural description of the associated connected components. Chapter III reaps the algorithmic rewards of the structural studies from the previous chapter, along with some further investigations. We explain how the structural results yield polynomial-time algorithms, firstly to compute the aforementioned connected components, and secondly to solve the Maker-Breaker game on hypergraphs of rank 3 as a corollary. Other complexness facets of the game, besides the algorithmic aspect, are also considered. Chapter IV addresses a reconfiguration problem on the square grid, of a type that can somewhat be likened to a one-player version of Maker-Breaker games. We first recall results from Demaine et al., who introduced this problem. A case which the authors thought was solved turns out to be complicated, and we bring some contributions towards its study.

Résumé

Cette thèse traite de jeux et d'hypergraphes. Nous nous concentrons tout particulièrement sur des jeux joués dans les hypergraphes, dont l'analyse s'avère conduire à l'étude de problèmes d'hypergraphes qui sont intéressants en soi. Les jeux positionnels mettent en scène deux joueurs qui sélectionnent tour à tour des sommets d'un hypergraphe, avec des objectifs variables : par exemple, au jeu du morpion, chaque joueur veut être le premier à posséder entièrement l'une des arêtes de l'hypergraphe. En convention Maker-Breaker, un joueur ("Maker") veut posséder entièrement l'une des arêtes, tandis que l'autre ("Breaker") veut l'en empêcher. Le thème principal est l'étude de structures dans les hypergraphes et leurs implications pour le jeu positionnel Maker-Breaker. En particulier, nous sommes intéressés par les hypergraphes de rang 3 : cela signifie que toutes les arêtes sont de taille au plus 3, ce qui permet des résultats structurels et aide à comprendre le jeu Maker-Breaker dans ce cas. Un autre thème est la conception d'algorithmes en temps polynomial pour résoudre des jeux et des problèmes d'hypergraphes. Ces deux thèmes sont intimement liés, puisque toute caractérisation structurelle fournit un algorithme aussi efficace que la structure en question est simple.

Le Chapitre I présente toutes les notions utilisées dans ce mémoire, ainsi que des résultats préliminaires. Après un état de l'art autour des jeux positionnels, nous introduisons notre approche structurelle du jeu Maker-Breaker, centrée sur une notion de dangers créés par Maker et que Breaker doit détruire immédiatement. Certaines structures élémentaires dans les hypergraphes, de rang 3 en particulier, sont aussi étudiées. Le Chapitre II réalise les études structurelles approfondies qui sont au cœur de ce mémoire. Nous obtenons une caractérisation structurelle de l'issue du jeu Maker-Breaker sur les hypergraphes de rang 3, ainsi que des stratégies optimales pour les deux joueurs, tout cela basé sur des propriétés d'intersections de dangers. Un lien direct apparaît avec un problème de connectivité dans les hypergraphes, pour lequel nous donnons une description structurelle fine des composantes connexes associées. Le Chapitre III récolte les fruits algorithmiques des études structurelles du chapitre précédent, et poursuit un peu plus loin. Nous expliquons comment les résultats structurels impliquent des algorithmes en temps polynomial, d'abord pour le calcul des composantes connexes susmentionnées, puis pour la résolution du jeu Maker-Breaker sur les hypergraphes de rang 3 comme corollaire. D'autres aspects de complexité du jeu, pas seulement algorithmique, sont également considérés. Le Chapitre IV concerne un problème de reconfiguration dans la grille carrée, d'un type qui peut être rapproché d'une version à un seul joueur du jeu Maker-Breaker. Nous rappelons quelques résultats de Demaine et al., qui ont introduit ce problème. Un cas que les auteurs pensaient résolu s'avère compliqué, et nous apportons notre contribution à son étude.

*« Do not the most moving
moments of our lives find us
all without words ? »*

— Marcel Marceau

Contents

Summary in French	9
Introduction	23
I Preparatory notions and preliminary results	29
I.1 The Maker-Breaker game: an introduction	29
I.1.1 State of the art on positional games	29
I.1.1.1 Some vocabulary around hypergraphs	29
I.1.1.2 Various conventions and problems	30
I.1.1.3 Achievement vs Avoidance, Strong vs Weak	32
I.1.1.4 Elementary strategies and principles	33
I.1.1.5 Difficulty comparison between conventions	40
I.1.2 State of the art on the Maker-Breaker game	42
I.1.2.1 General results	42
I.1.2.2 Games played on the edge set or vertex set of a graph	44
I.1.2.3 Our case of interest: hypergraphs of small rank	45
I.1.3 Our approach to the Maker-Breaker game and a few basic results	48
I.1.3.1 Playing the game on marked hypergraphs	48
I.1.3.2 Optimizing time or tokens	51
I.1.3.3 A reminder on two key principles	53
I.1.3.4 Some operations on marked hypergraphs	53
I.2 Subhypergraph collections and their intersection properties	55
I.2.1 Some notions on marked hypergraphs	55
I.2.1.1 Intersections and unions of subhypergraph collections	55
I.2.1.2 Pointed marked hypergraphs	57
I.2.2 Dangers in the Maker-Breaker game	57
I.2.2.1 Definitions and first results	58
I.2.2.2 Considering a fixed family of dangers	59
I.2.2.3 Danger prevention	61
I.2.2.4 Restricted obstructions	64
I.2.3 First results: the example of trivial dangers	64
I.2.3.1 A matter of survival... and more	65
I.2.3.2 General counting results on the biased Maker-Breaker game	66
I.2.3.3 An application to a specific game around sums	69
I.3 Elementary structures in (marked) hypergraphs	70
I.3.1 In general hypergraphs	71
I.3.1.1 Walks and paths	71
I.3.1.2 Forcing paths	73
I.3.2 In 3-uniform hypergraphs	75
I.3.2.1 Chains, cycles and tadpoles	75
I.3.2.2 Substructure lemmas	77

I.3.2.3	Projections	80
I.3.2.4	Union lemmas	81
II	In-depth structural studies in hypergraphs	89
II.1	The Maker-Breaker game: structural results in hypergraphs of rank 3	89
II.1.1	Presentation of the problem and state of the art	89
II.1.2	The game on 3-uniform marked hyperforests	90
II.1.2.1	Solution in terms of nunchakus	90
II.1.2.2	Interpretation in terms of the family of dangers \mathcal{S}	91
II.1.3	The game on general 3-uniform marked hypergraphs	92
II.1.3.1	The family of dangers \mathcal{C}	92
II.1.3.2	The family of dangers \mathcal{D}_0	93
II.1.3.3	Statement of the main results	95
II.1.4	Approximating \mathcal{D}_0^* and \mathcal{D}_0^{*2}	95
II.1.4.1	The families of dangers \mathcal{T} and \mathcal{D}_1	96
II.1.4.2	The families of dangers \mathcal{D}_1^\emptyset and \mathcal{D}_2	97
II.1.5	Structure of the $\mathcal{D}_1^{\emptyset, \text{rest}}$ -dangers	99
II.1.5.1	First properties	100
II.1.5.2	Union lemmas	102
II.1.5.3	Inside structure	107
II.1.6	Proof of the main results... first assuming a key lemma	110
II.1.6.1	Proof of Theorem II.1.12	110
II.1.6.2	Proof of Theorem II.1.13	112
II.1.7	Proof of the key lemma	112
II.1.7.1	Preliminary statements	113
II.1.7.2	Roadmap of the proof	114
II.1.7.3	Finishing the proof when D is of type (2)	119
II.1.7.4	Finishing the proof when D is of type (1)	120
II.1.8	Conclusion and prospects	123
II.2	Structure of a $(k - 2)$ -linear connected component	124
II.2.1	Presentation of the problem and state of the art	124
II.2.2	The archipelago structure	125
II.2.2.1	An overview	125
II.2.2.2	Compatible walks	127
II.2.2.3	Islands and archipelagos	130
II.2.2.4	Archipelago properties	131
II.2.3	Characterization as unique maximal archipelago	134
II.2.3.1	A -types and statement of the main result	135
II.2.3.2	Augmenting archipelagos	136
II.2.3.3	Proof of the main result	140
II.2.4	Conclusion and prospects	141
III	How structure contributes to algorithms and game complexness	143
III.1	Algorithms for connectivity problems in graphs and hypergraphs	143
III.1.1	Presentation of the problem and state of the art	143
III.1.2	The λ -linear connectivity problem $\text{HYPCON}_{k,\lambda}$	144
III.1.2.1	A polynomial-time algorithm for the case $\lambda = k - 2$	144
III.1.2.2	Reduction to uniform hypergraphs	147
III.1.3	The "Paths Avoiding Forbidden Pairs" problem PAFP	147
III.1.3.1	Reducing $\text{HYPCON}_{k,\lambda}$ to PAFP	147

III.1.3.2	Reducing some instances of PAFP to $\text{HYPCON}_{k,\lambda}$	148
III.1.4	Conclusion and prospects	149
III.2	The Maker-Breaker game: complexity and bounds on optimization parameters	150
III.2.1	Presentation of the problem and state of the art	150
III.2.2	A polynomial-time algorithm for the game on hypergraphs of rank 3	151
III.2.3	When Maker wants to win in a minimum number of rounds	152
III.2.3.1	In hypergraphs of rank 3	152
III.2.3.2	In hypergraphs of rank $k \geq 4$	154
III.2.4	When Maker wants to win using a minimum number of tokens	156
III.2.4.1	In hypergraphs of rank 3	156
III.2.4.2	In hypergraphs of rank $k \geq 4$	156
III.2.5	Conclusion and prospects	162
IV	From positional games to reconfiguration problems	165
IV.1	Coin-moving puzzles with 2-adjacency restriction on the square grid	166
IV.1.1	Notations and first observations	167
IV.1.2	Picking up and dropping coins	168
IV.1.3	Span of a configuration	168
IV.1.3.1	Definition and a key necessary condition	168
IV.1.3.2	Structural properties	169
IV.1.4	Minimal/minimum configurations	170
IV.1.5	‘L’s and canonical configurations	171
IV.1.6	Extra coins and redundant coins	172
IV.1.6.1	Definitions and first observations	172
IV.1.6.2	A central factor for the game	173
IV.2	The case of two extra coins	174
IV.2.1	Presentation of the problem and state of the art	174
IV.2.2	Canonicalization process	174
IV.2.3	A slight improvement on Theorem IV.2.1	176
IV.2.4	The general case with two extra coins	178
IV.2.4.1	Worst-case puzzles	178
IV.2.4.2	A new sufficient condition for a puzzle to be solvable	182
IV.2.5	Conclusion and prospects	186
	Conclusion	189

Résumé en français

Introduction

Alors que j'obtenais fièrement mon tout premier résultat de recherche, réfutant une conjecture du mathémagicien Persi Diaconis sur le jeu de prédiction du mélange dit "américain" d'un jeu de cartes, la réalisation m'est venue : au fond, ce qui me plaisait n'était pas tant la théorie des probabilités, mais surtout la combinatoire et les jeux mathématiques. Une brève recherche des thèses soutenues sur le sujet en France a suffi à faire ressortir un nom, qui semblait apparaître partout, et j'ai vite compris que je ferais bien de contacter un certain Sylvain Gravier. Une réponse enthousiaste plus tard, j'ai déménagé à Grenoble, où Sylvain m'a présenté Isabelle Sivignon. Pour mon stage de master, ils m'avaient préparé ensemble un sujet portant sur des puzzles de déplacement de jetons, un jeu solitaire sur lequel j'ai fini par revenir pendant ma thèse (mais je m'avance un peu). Comme récompense en fin de stage, j'ai débloqué un deuxième joueur, Sylvain et Isabelle m'offrant de poursuivre en thèse avec eux sur le sujet de jeux à deux joueurs appelés *jeux positionnels*. C'est le moment où je ressors mon sempiternel exemple :

- Alors, c'est quoi que tu fais exactement ?
- Bon : tu vois le jeu du morpion, j'imagine ?

Le jeu du *morpion*, ou *tic-tac-toe*, est le jeu positionnel le plus célèbre. Deux joueurs, Alice et Bob, choisissent tour à tour des cases d'une grille 3×3 , dessinant des ronds et des croix respectivement. Le premier joueur qui aligne trois de ses symboles l'emporte. Si aucun joueur n'y parvient, ce qui est bien connu pour être le cas lorsque les deux joueurs jouent correctement, alors la partie est nulle. Imaginons qu'Alice, lassée de toutes ces parties nulles (et avec une petite idée derrière la tête), offre à Bob de changer légèrement les règles. Elle jouera en premier, essayant d'aligner trois ronds, et le but de Bob sera "simplement" de l'en empêcher. Comme ça, il y aura toujours un vainqueur. Le jeu devient ce qu'on appelle un jeu positionnel *Maker-Breaker*, puisqu'Alice ("Maker") essaie de *construire* un alignement tandis que Bob ("Breaker") essaie de *détruire* ses rêves. Il se trouve que c'est Alice qui a une stratégie gagnante dans ce cas particulier, mais ce jeu peut en fait être vastement généralisé, formant la famille des jeux Maker-Breaker qui sont le sujet principal de ma thèse.

- Attends... t'es quand même pas payé pour jouer, si ?
- Ben, si ça peut te rassurer, on n'est pas payé beaucoup...

Pour comprendre comment marche cette généralisation, il faut voir que le jeu décrit ci-dessus est fondamentalement défini comme suit : on a neuf positions jouables, que sont les neuf cases, et huit manières différentes de gagner pour Alice, correspondant aux huit alignements (trois lignes, trois colonnes, deux diagonales). Comme cette description contient toute l'information du jeu, considérons une représentation graphique alternative. On dessine neuf points, un pour

chaque case, et huit patates, correspondant aux huit alignements gagnants. Au lieu de dessiner des ronds (resp. des croix), Alice (resp. Bob) colore les points correspondants en rouge (resp. en bleu). Voir Figure 1. Alice gagne s'il existe une patate dont tous les points sont rouges (resp. en bleu). Maintenant, on peut dessiner un plateau de jeu avec n'importe quel nombre de points et n'importe quelles patates (de n'importe quelles tailles, pas nécessairement égales d'ailleurs), jouer avec les mêmes règles, et cela définit un nouveau jeu Maker-Breaker. Un ensemble de points et de patates s'appelle un *hypergraphe* : les points sont appelés *sommets* et les patates sont appelées *arêtes*. Il y a autant de jeux Maker-Breaker qu'il y a d'hypergraphes, c'est-à-dire... beaucoup.

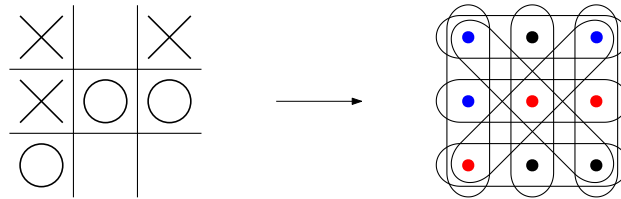


FIGURE 1 : Deux manières de représenter le même jeu (en cours de partie).

En effet, les hypergraphes sont des objets très généraux : un ensemble V , un ensemble E de parties de V , et c'est tout. Ainsi, leur structure peut être très complexe. Ceci explique que, dans la plupart des résultats généraux de la littérature, les hypothèses ne portent pas sur la structure de l'hypergraphe mais plutôt sur des quantités globales telles que le nombre d'arêtes. L'objectif principal fixé au début de ma thèse était de contribuer à combler ce vide en obtenant des résultats structurels sur le jeu Maker-Breaker. Cela signifie identifier des critères pour l'*issue* du jeu (quel joueur l'emporte dans une partie optimale?) qui soient purement liés à la structure de l'hypergraphe. Un autre centre d'intérêt était l'aspect algorithmique. Il était déjà connu depuis longtemps qu'il n'existait aucun algorithme efficace pour résoudre le jeu Maker-Breaker en toute généralité, c'est-à-dire pour déterminer l'issue du jeu sur un hypergraphe donné quelconque. Plus précisément, et en des termes plus techniques, ce problème de décision est PSPACE-complet [Sch78].

Par conséquent, l'idée était de se restreindre à des familles d'hypergraphes avec une structure plus simple : je pourrais alors espérer obtenir des caractérisations structurelles pour l'issue du jeu, dont il serait peut-être possible de tirer des algorithmes en temps polynomial. Nous avons identifié deux familles qui pourraient être un bon point de départ pour moi :

- Les hypergraphes de nature géométrique étaient une première idée. On peut notamment penser aux hypergraphes comme celui du tic-tac-toe, où les sommets peuvent être identifiés aux cases d'une grille rectangulaire et les arêtes correspondent à des alignements (sans forcément inclure tous les alignements possibles). Nous avons également pensé à des hypergraphes représentant des intersections d'objets géométriques dans le plan, par exemple des rectangles.
- Les hypergraphes de faible *rang* (taille d'une plus grosse arête) étaient une autre idée. Plus les arêtes sont petites, plus la structure est simple. On a une bonne compréhension de la structure des graphes, qui ne sont rien d'autre que les hypergraphes dont toutes les arêtes sont de taille 2, et le jeu s'avère trivial dans ce cas. Cependant, on savait beaucoup moins de choses concernant la structure des hypergraphes de rang 3, et une unique étude de Kutz [Kut04] existait sur le jeu Maker-Breaker dans ce cas (résolvant un sous-cas). Malgré les sérieuses complications causées par le saut de 2 à 3 en taille des arêtes, j'avais le sentiment qu'il y avait la place pour des résultats structurels dans les hypergraphes de rang 3 généraux. D'un autre côté, le point de vue algorithmique tendait à tempérer cet optimisme, puisqu'il est fréquent d'observer un saut de complexité entre les valeurs 2 et 3 dans les problèmes d'optimisation combinatoire ou de logique propositionnelle. Le

problème de satisfaisabilité booléenne en est un exemple notoire : 2-SAT est résolu en temps polynomial, mais le problème devient NP-complet lorsqu'on autorise des clauses de taille 3.

J'ai d'abord considéré des hypergraphes combinant les deux idées ci-dessus. Toutefois, il nous est apparu que le rang 3 était le facteur déterminant, dans le sens où l'ajout de propriétés géométriques ne semblait pas faciliter le problème.

Par conséquent, je me suis rapidement concentré sur le jeu Maker-Breaker sur les hypergraphes de rang 3 généraux. Les travaux de Kutz sont ceux qui ont eu la plus grande influence sur ma thèse. Son résultat donnait une caractérisation structurelle pour l'issue dans le cas *linéaire*, c'est-à-dire lorsque deux arêtes distinctes quelconques s'intersectent en au plus un sommet (comme c'est le cas au tic-tac-toe par exemple). La structure en question est identifiable efficacement, ce qui permet de déterminer l'issue en temps polynomial. J'ai souhaité étendre les deux résultats, structurel et algorithmique, aux hypergraphes de rang 3 généraux. Un des outils utilisés par Kutz pour démontrer les cas où Maker gagne consiste à exhiber un sommet x tel que, si Maker joue x comme premier coup, alors cela crée plusieurs menaces urgentes pour Breaker, qui n'a aucun coup traitant toutes ces menaces à la fois. En d'autres termes, s'il existe un x créant des menaces, que j'appelle des *dangers en x* , qui ne s'intersectent pas, alors Maker a une stratégie gagnante en jouant x comme premier coup. J'ai opté pour un point de vue qui diffère de celui de Kutz, dans le sens où j'ai fait de cette notion d'intersections de dangers le centre de mon approche plutôt qu'un outil occasionnel. Cette idée vaut pour n'importe quel hypergraphe, quel que soit le rang. Cependant, dans les hypergraphes de rang 3 spécifiquement, je pressentais que l'issue pouvait être lue directement sur l'hypergraphe en termes d'intersections de dangers, en n'ayant à prendre en compte que certains dangers spécifiques relativement élémentaires. En particulier, cela donnerait un algorithme en temps polynomial déterminant l'issue du jeu Maker-Breaker sur les hypergraphes de rang 3, statuant ainsi sur un problème ouvert significatif. La résolution de cette conjecture et la rédaction des parties liées aux dangers ont accaparé la majorité de mon temps et de mon énergie au cours de cette thèse. Premièrement, les preuves sont longues et très techniques. De plus, beaucoup d'arguments de différentes natures sont utilisés : les identifier, démêler lesquels relèvent du jeu et lesquels relèvent des hypergraphes, comprendre ce qui les fait fondamentalement fonctionner, tout cela a été un long processus.

Bien que cette thèse parle principalement de jeux, elle porte aussi sur les hypergraphes. En effet, je me suis retrouvé à étudier des problèmes d'hypergraphes qui, quoique motivés initialement par le jeu Maker-Breaker dans mon cas, sont davantage que de simples outils pour le jeu et ont du potentiel pour d'autres applications. En particulier, un type de chemin linéaire appelé *chaîne* joue un rôle clé pour le jeu Maker-Breaker sur les hypergraphes de rang 3, l'existence d'une chaîne entre deux sommets donnés s'avérant être une question cruciale. Ainsi, nous avons été amenés à étudier ce problème de connectivité dans les hypergraphes, qui est intéressant en soi : les chaînes sont des chemins naturels à considérer dans les hypergraphes de rang 3, et elles apparaissaient déjà dans de nombreux articles non liés aux jeux. L'étude structurelle de la connectivité par chaînes et le problème algorithmique associé représentent une part importante de cette thèse. Ces questions se sont avérées liées à un problème autour des *line graphs* d'hypergraphes, un sujet également très présent dans la littérature. Enfin, certaines propriétés structurelles d'hypergraphes élémentaires de rang 3 tels que les chaînes et les cycles, qui sont des lemmes préliminaires essentiels pour nos résultats sur le jeu Maker-Breaker, pourraient être utiles à n'importe quelle étude structurelle dans les hypergraphes de rang 3.

J'ai également étudié deux paramètres d'hypergraphes en rapport avec le jeu Maker-Breaker, correspondant chacun à des stratégies de Maker visant à optimiser une certaine quantité. La première de ces quantités est le nombre de tours : si Maker a une stratégie gagnante, alors de combien de temps a-t-elle besoin, i.e. quel est le nombre minimum de tours en lequel elle peut s'assurer d'obtenir une arête monochrome rouge ? Cette question est très étudiée dans la

littérature des jeux Maker-Breaker, cependant rien n'était connu sur le cas des hypergraphes à petites arêtes spécifiquement. La deuxième quantité est le nombre de *jetons*, dans la variante suivante du jeu Maker-Breaker que nous inaugurons dans cette thèse. Supposons que Maker, au lieu de colorer des sommets en rouge de façon indélébile, pose des jetons rouges sur les sommets qu'elle joue. A chaque tour, elle a le choix entre déplacer un jeton qu'elle avait posé précédemment (abandonnant alors le contrôle d'un sommet, qui pourra par la suite être subtilisé par Breaker), ou bien utiliser un nouveau jeton (supposons qu'elle en a une infinité à disposition). Breaker, quant à lui, colore permanemment les sommets en bleu comme d'habitude. Si Maker a une stratégie gagnante, quel est le nombre minimum de jetons dont elle a besoin ? Dans le contexte de cette thèse, je préfère utiliser le point de vue de l'optimisation, mais on pourrait aussi incorporer ceci aux règles du jeu, pour les deux joueurs en réalité : ils auraient chacun un nombre fixé (fini ou infini) de jetons rouges et bleus respectivement, pas nécessairement égal, et on s'intéresserait à l'issue correspondante. Ces deux paramètres sont des indicateurs du degré de complexité des stratégies gagnantes de Maker sur un hypergraphe donné. Mon but était d'évaluer les valeurs maximum atteintes par ces paramètres, en fonction de la taille des arêtes. En particulier, dans les hypergraphes de rang 3, le résultat structural autour des dangers mentionné précédemment a permis de répondre facilement à cette question.

Dans la variante du jeu Maker-Breaker avec un nombre limité de jetons, Maker essaie de réorganiser ses jetons pour que leur configuration couvre une arête, et Breaker tente de l'entraver en bloquant des sommets (en effet, les sommets occupés par des jetons bleus sont inutilisables pour Maker). Si ces sommets bloqués, au lieu d'être choisis intelligemment par un deuxième joueur qui s'adapte aux coups de Maker, étaient déterminés automatiquement par une certaine règle préétablie, alors le jeu deviendrait un problème de reconfiguration. Ce type de problèmes de reconfiguration peut ainsi être assimilé à une version à un joueur du jeu Maker-Breaker. Demaine et al. [DDV02] ont introduit des puzzles de déplacements de pièces dans la grille carrée qui rentrent dans cette catégorie. La restriction est qu'une pièce ne peut être déplacée que vers une case vide dont au moins deux cases voisines orthogonales sont déjà occupées par des pièces. J'avais étudié ce jeu pendant mon stage de master, à la fin duquel j'avais découvert une imprécision dans le résultat principal de [DDV02], avec trop peu de temps pour regarder cela plus en détail. Je suis revenu à ce problème pendant la thèse, pour examiner l'étendue de cette erreur et chercher des résultats alternatifs.

Nous allons maintenant résumer les quatre chapitres de ce mémoire. Le Chapitre I présente toutes les notions figurant dans ce mémoire, ainsi que des résultats préliminaires préparant les études des futurs chapitres. Le Chapitre II réalise les études structurelles approfondies qui sont au cœur de ce mémoire, au sujet du jeu Maker-Breaker sur les hypergraphes de rang 3 puis du problème de connectivité mentionné précédemment. Le Chapitre III récolte les fruits algorithmiques des études structurelles du chapitre précédent, et poursuit avec l'examen de nos deux paramètres d'hypergraphes (optimisation en nombre de tours ou de jetons). Le Chapitre IV porte sur le problème de reconfiguration dans la grille carrée.

Chapitre I : Notions préparatoires et résultats préliminaires

Dans un jeu positionnel, Alice et Bob colorent tour à tour des sommets d'un hypergraphe en rouge et bleu respectivement, et l'objectif des joueurs dépend de la *convention* utilisée. Il existe également une version avec *biais* : à chaque tour, au lieu que les joueurs choisissent un sommet

chacun, Alice choisit p sommets d'un coup et Bob choisit q sommets d'un coup. Les quatre principales conventions sont les suivantes :

- *Maker-Maker* (Chvátal et Erdős, 1978 [CE78]) : le premier joueur qui obtient une arête monochrome de sa couleur a gagné, et la partie est nulle si aucun joueur n'y parvient (exemple : le tic-tac-toe).
- *Maker-Breaker* (Hales et Jewett, 1963 [HJ63]) : Alice ("Maker") gagne si elle obtient une arête monochrome rouge au cours de la partie, sinon elle perd.
- *Avoider-Avoider* (Harary, 1981 [Har81]) : le premier joueur qui obtient une arête monochrome de sa couleur a perdu, et la partie est nulle si aucun joueur n'y est contraint.
- *Avoider-Enforcer* (Lu, 1992 [Lu92]) : Alice ("Avoider") perd si elle obtient une arête monochrome rouge au cours de la partie, sinon elle gagne.

Pendant cette thèse, j'aurais aimé avoir une référence compilant tous les résultats de base sur les jeux positionnels en général. Certains sont éparpillés dans la littérature, d'autres sont connus de la plupart des chercheurs du domaine sans être réellement formulés où que ce soit. C'est pourquoi j'ai tenu à lister au début de ce mémoire un certain nombre de principes stratégiques élémentaires, énoncés dans leur contexte de validité le plus général (conventions, biais). Un exemple est le "vol de stratégie", qui permet notamment de montrer que le deuxième joueur n'a jamais une stratégie gagnante en convention Maker-Maker sans biais.

On dit que les conventions Maker-Maker et Maker-Breaker définissent des *jeux d'accomplissement* (*achievement games*), tandis que les conventions Avoider-Avoider et Avoider-Enforcer sont leurs versions *misère* respectives et définissent des *jeux d'évitement* (*avoidance games*). Les jeux d'accomplissement ont des propriétés agréables qui rendent leur étude plus simple, comme l'absence de *zugzwang* : chaque joueur préfère toujours que ce soit à lui/elle de jouer, alors que cela dépend des situations dans les jeux d'évitement. Selon une deuxième manière de classer, les conventions symétriques Maker-Maker et Avoider-Avoider définissent des *jeux forts* (*strong games*) tandis que les conventions antagonistes Maker-Breaker et Avoider-Enforcer définissent des *jeux faibles* (*weak games*). Les jeux faibles sont les plus abordables, ce qui peut se comprendre intuitivement par le fait qu'il y a un attaquant (Maker/Enforcer) et un défenseur (Breaker/Avoider) au lieu que les deux joueurs aient à trouver l'équilibre entre attaque et défense. C'est donc logiquement que le jeu d'accomplissement faible Maker-Breaker est le plus étudié dans la littérature, et c'est aussi le thème principal de cette thèse. A noter que, quitte à considérer tous les premiers coups possibles de Breaker lorsqu'il commence, on suppose toujours que Maker commence : ainsi, lorsqu'on dit par exemple que "Maker gagne sur H ", cela signifie que Maker a une stratégie gagnante pour le jeu Maker-Breaker sur l'hypergraphe H lorsqu'elle joue en premier. Nous faisons également le choix de jouer dans des *hypergraphes marqués*, une généralisation des hypergraphes dans laquelle certains sommets peuvent être marqués (graphiquement, on les représente entourés). L'idée est que les sommets marqués représentent les sommets possédés par Maker. Ainsi, plutôt que de colorer des sommets en rouge et bleu, Maker *marque* des sommets et Breaker *supprime* des sommets (supprimer un sommet signifie également supprimer toutes les arêtes contenant ce sommet). Maker gagne lorsqu'une arête a tous ses sommets marqués. Quitte à rajouter à certaines arêtes des sommets fictifs marqués d'office, le jeu sur les hypergraphes de rang k est équivalent au jeu sur les hypergraphes marqués k -uniformes, une classe qui a notamment l'avantage d'être stable par les coups des deux joueurs.

Dans la plupart des instances du jeu Maker-Breaker étudiées dans la littérature, le jeu est naturellement formulé comme joué sur les arêtes d'un graphe complet, si bien que l'hypergraphe est en réalité sous-jacent. Maker veut obtenir un ensemble d'arêtes du graphe qui contienne un certain objet (un triangle, un arbre couvrant, un couplage parfait...). On a alors des résultats de nature quantitative, notamment sur la valeur critique du biais à partir de laquelle le vainqueur du jeu change. Au contraire, nous sommes intéressés par les aspects structurels du jeu Maker-

Breaker, le plus souvent sans biais.

La base de notre approche structurelle est une propriété très commode du jeu Maker-Breaker qu'est la monotonie (sous-entendu : monotonie par sous-hypergraphe). Cela signifie que, pour tout sous-hypergraphe H' de H , si Maker gagne sur H' alors Maker gagne également sur H . En effet, si Maker gagne sur H' , alors il lui suffit de se concentrer uniquement sur H' et d'y appliquer sa stratégie gagnante pour y obtenir une arête entièrement marquée. En particulier, soit x un sommet non marqué de H , et soit D un sous-hypergraphe de H contenant x sur lequel Maker gagnerait si x était marqué. Alors, jouer x permet à Maker de créer une menace urgente : Breaker devrait *détruire* D immédiatement (i.e. répondre par un sommet de D), sans quoi Maker gagnerait par monotonie. On dira que D est un *danger en x* , un vocabulaire dû au fait que nous adoptons le point de vue de Breaker.

Considérons désormais plusieurs dangers en un même sommet x . Plus précisément, fixons une famille \mathcal{F} de "modèles" de dangers, et considérons tous les \mathcal{F} -dangers en x i.e. les dangers en x appartenant à cette famille. Alors, si Maker joue x , cela forcera Breaker à répondre immédiatement dans l'*intersection* de tous ces dangers (où l'intersection exclut bien sûr x et les sommets qui sont déjà marqués, puisqu'elle est censée représenter les réponses jouables de Breaker si Maker joue x). Ce principe se poursuit tout au long de la partie, pendant laquelle les dangers évoluent : les coups de Maker créent de nouveaux \mathcal{F} -dangers en d'autres sommets, et les coups de Breaker en font disparaître. Pour tout r , en notant $J_r(\mathcal{F}, H)$ la propriété affirmant que "Breaker peut s'assurer que, à chacun des r premiers tours, il pourra détruire les \mathcal{F} -dangers en le sommet joué par Maker", on a donc la condition nécessaire suivante qu'on notera (CN) : si Breaker gagne sur H , alors $J_r(\mathcal{F}, H)$ est vérifiée.

Il se trouve qu'il est possible de voir directement sur l'hypergraphe H si $J_r(\mathcal{F}, H)$ est vérifiée ou non. En effet, pour $r = 2$ par exemple, il existe une famille $\mathcal{F}^* \supseteq \mathcal{F}$ qui fait en quelque sorte la prévention des problèmes d'intersection des \mathcal{F} -dangers que Breaker pourrait avoir au deuxième tour, si bien que Breaker peut détruire les \mathcal{F} -dangers aux deux premiers tours si et seulement si il peut détruire les \mathcal{F}^* -dangers au premier tour. A chaque itération de cet opérateur "étoile", on étend cette prévention un tour plus loin, d'où l'équivalence $J_r(\mathcal{F}, H) \iff J_1(\mathcal{F}^{*(r-1)}, H)$. La propriété $J_r(\mathcal{F}, H)$, dont la définition parcourt un arbre de coups lorsque $r \geq 2$, se retrouve ainsi exprimée au lieu de cela comme une propriété statique de l'hypergraphe H en termes d'intersections de collections de sous-hypergraphes, au prix d'une famille de dangers plus grande et plus complexe. Notons que toutes ces considérations s'étendent à la version biaisée du jeu Maker-Breaker, bien que nous ne l'évoquions pas dans ce résumé.

Nous nous intéressons à la réciproque de (CN) : étant donné une classe \mathcal{H} d'hypergraphes marqués, nous nous demandons pour quels \mathcal{F} et r est-ce que la propriété $J_r(\mathcal{F}, H)$ caractérise les hypergraphes marqués $H \in \mathcal{H}$ sur lesquels Breaker gagne. En particulier, si r est une constante, alors décider l'issue du jeu Maker-Breaker sur la classe \mathcal{H} se réduit au problème d'existence d'un élément de \mathcal{F} dans un hypergraphe marqué donné : dans le cas d'une famille \mathcal{F} "raisonnable", on obtiendrait alors un algorithme en temps polynomial.

Chapitre II : Etudes structurelles approfondies dans les hypergraphes

Pour ce qui est des hypergraphes de rang 3, ce qui revient à considérer les hypergraphes marqués 3-uniformes, un article de Rahman et Watson [RW20] laissait à penser qu'une telle caractérisation pourrait être établie. Les auteurs étudient un jeu plus général que le jeu Maker-Breaker, où l'hypergraphe est remplacé par une formule logique en forme normale conjonctive. A chaque

tour, chaque joueur choisit une variable et lui assigne une valeur booléenne. Le premier joueur ("False") veut rendre la formule fausse tandis que le deuxième joueur ("True") veut rendre la formule vraie. Le cas d'une formule positive, i.e. dont tous les littéraux sont positifs, revient au jeu Maker-Breaker, où les variables sont les sommets et les clauses sont les arêtes. Rahman et Watson étudient le cas où les clauses sont de taille au plus 3, avec la contrainte que chaque clause contient une variable qui n'apparaît dans aucune autre clause. Il n'est pas difficile de montrer que, pour des formules positives, cette contrainte implique en particulier que l'hypergraphe est linéaire : par conséquent, on retombe dans le cas déjà résolu par Kutz [Kut04]. Cependant, ce qui nous intéresse est la nature de la caractérisation structurelle obtenue par Rahman et Watson. Celle-ci, dans le cas des formules positives et traduite en nos termes, affirme que : pour tout hypergraphe marqué 3-uniforme H satisfaisant la contrainte des auteurs, Breaker gagne sur H si et seulement si $J_3(\mathcal{D}_0, H)$ est vérifiée, où \mathcal{D}_0 est une famille de dangers très élémentaires. Les auteurs conjecturent que cette équivalence reste vraie pour toutes les formules dont les clauses sont de taille au plus 3, et donc en particulier pour tous les hypergraphes marqués 3-uniformes. La famille \mathcal{D}_0 est définie comme suit. Il existe deux types de \mathcal{D}_0 -dangers en un sommet x : le x -snake et le x -cycle (voir Figure 2, haut). Un x -snake est une *chaîne* (chemin simple linéaire) reliant x à un sommet marqué. Un x -cycle est une chaîne reliant x à lui-même (sauf dans le cas d'un x -cycle de longueur 2, qui n'est pas linéaire puisque formé de deux arêtes $\{x, y, a\}$ et $\{x, y, b\}$). Un x -snake et un x -cycle sont bien des dangers en x puisque, si Maker joue x , alors ils deviennent ce qu'on appelle respectivement un *nunchaku* et un *necklace* (voir Figure 2, bas). En effet, Maker gagne sur un nunchaku ou un necklace, avec une technique de *forçage*. Maker joue un sommet de degré 2 adjacent à un sommet marqué, ce qui force Breaker à jouer le troisième sommet de l'arête en question, puis Maker joue un sommet de degré 2 adjacent à son dernier coup, etc., et Breaker n'a que des coups forcés jusqu'à ce qu'il soit piégé à l'autre bout du chemin où Maker menacera dans deux arêtes à la fois.

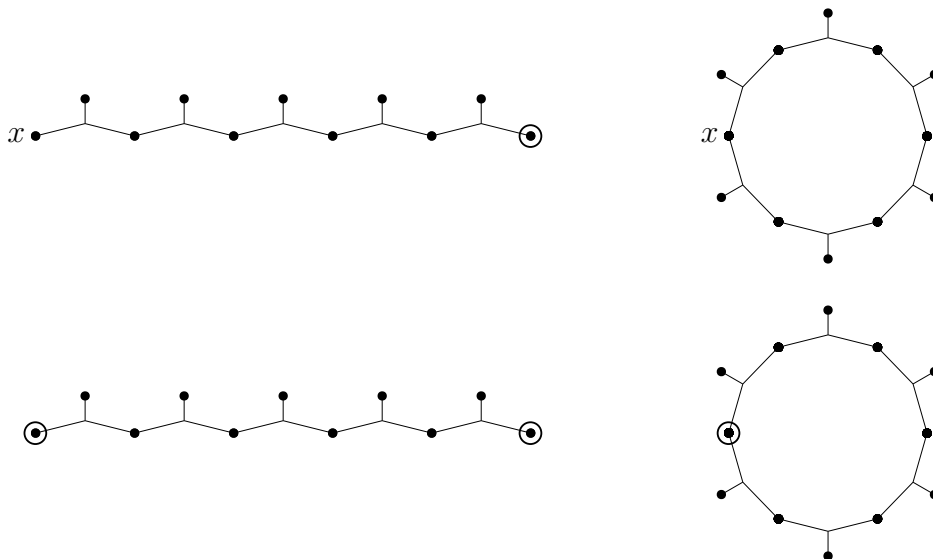


FIGURE 2 : Un x -snake et un x -cycle (haut), qui deviennent respectivement un nunchaku et un necklace (bas) si Maker joue x . Chaque "griffe" représente une arête de taille 3.

Nous démontrons la conjecture de Rahman et Watson pour les formules positives, c'est-à-dire : pour tout hypergraphe marqué 3-uniforme H , Breaker gagne sur H si et seulement si $J_3(\mathcal{D}_0, H)$ i.e. $J_1(\mathcal{D}_0^{*2}, H)$ est vérifiée. Il s'agit du résultat le plus central de cette thèse. Nous exhibons également une sous-classe conséquente d'hypergraphes marqués 3-uniformes sur laquelle la même caractérisation est vraie avec la propriété $J_2(\mathcal{D}_0, H)$ au lieu de $J_3(\mathcal{D}_0, H)$. Enfin, nous décrivons les stratégies optimales des deux joueurs en termes d'intersections de dangers. Si Maker gagne,

alors tout x tel que l'intersection des \mathcal{D}_0^{*2} -dangers est vide est un premier coup optimal pour Maker. Si Breaker gagne, alors tout y dans l'intersection des \mathcal{D}_0^{*2} -dangers en x , où x est le premier coup de Maker, est un premier coup optimal pour Breaker. La preuve est longue et technique, et se fait par récurrence : on montre que si $J_1(\mathcal{D}_0^{*2}, \cdot)$ est vérifiée par l'hypergraphe marqué de départ, alors elle l'est encore par l'hypergraphe marqué obtenu après le premier tour si Breaker a joué un coup optimal. Ainsi, la propriété $J_1(\mathcal{D}_0^{*2}, \cdot)$ se propage jusqu'à la fin de la partie, ce qui signifie que Breaker gagne puisqu'en particulier la famille \mathcal{D}_0^{*2} comprend les snakes de longueur 1 (qui correspondent aux menaces de victoire au prochain tour de Maker). Cette preuve est majoritairement constituée d'applications répétées de lemmes structurels élémentaires énoncés dans le chapitre précédent, portant sur l'existence de chaînes et de cycles à l'intérieur de certaines unions de chaînes et/ou de cycles dans les hypergraphes 3-uniformes.

Il est également très important de comprendre ce critère du point de vue de Maker. Le fait que la propriété $J_3(\mathcal{D}_0, H)$ ne soit pas vérifiée signifie que Maker peut s'assurer que, à la fin de l'un des trois premiers tours, l'hypergraphe marqué mis à jour contiendra un nunchaku ou un necklace. D'un point de vue algorithmique, le jeu Maker-Breaker sur les hypergraphes de rang 3 se réduit donc (avec un facteur n^6 où n est le nombre de sommets) au problème d'existence d'un nunchaku ou d'un necklace dans un hypergraphe marqué 3-uniforme, qui lui-même se réduit trivialement au problème d'existence d'une chaîne entre deux sommets donnés dans un hypergraphe 3-uniforme.

Nous en sommes donc venus à étudier la connectivité par chaînes dans les hypergraphes 3-uniformes, un problème relevant de la théorie des hypergraphes et non lié au jeu intrinsèquement. La question algorithmique n'avait jamais été étudiée auparavant. Il existait bien des études sur les chaînes, et autres types de chemins dans les hypergraphes définis par la taille des intersections d'arêtes (inférieure ou égale à 1 dans le cas des chaînes), mais il s'agissait toujours de résultats extrémaux. Dans ce chapitre, nous nous intéressons à l'aspect structurel, qui nous fournira l'algorithme. Il se trouve que tous les résultats se généralisent aux hypergraphes k -uniformes où $k \geq 4$, en remplaçant la linéarité qui définit les chaînes par une notion de $(k-2)$ -linéarité : cela signifie qu'on s'intéresse aux chemins où les intersections d'arêtes sont de taille inférieure ou égale à $k-2$, ou autrement dit qu'on interdit les intersections de taille $k-1$.

Nous étudions la structure des *composantes connexes* $(k-2)$ -linéaires associées à ce problème de connectivité. Il est important de noter (et c'est toute la difficulté) que les chemins $(k-2)$ -linéaires ne définissent pas une relation transitive : il peut exister des chemins $(k-2)$ -linéaires de x à y et de y à z sans qu'il y en ait un de x à z dans l'union des deux. En particulier, les composantes connexes $(k-2)$ -linéaires ne forment pas une partition de l'ensemble des sommets. Nous montrons que le sous-hypergraphe A induit par la composante connexe $(k-2)$ -linéaire d'un sommet x^* a une structure très particulière que nous appelons un x^* -archipel. Plus précisément, A est l'unique x^* -archipel maximal de l'hypergraphe : on en obtient ainsi une caractérisation structurelle. La structure de x^* -archipel décrit finement l'organisation interne de A (comment se décomposent les chaînes issues de x^*) ainsi que son interface avec le reste de l'hypergraphe (quelle forme ont les arêtes de la coupe).

Chapitre III : Contributions de la structure pour les algorithmes et la complexité du jeu

Ce chapitre effectue en premier lieu la transition des aspects structurels aux aspects algorithmiques. Tout d'abord, la preuve du résultat structurel sur les archipels suggère une procédure

incrémentale pour calculer la composante connexe $(k - 2)$ -linéaire d'un sommet x^* donné. Cet algorithme construit un x^* -archipel qui est grossi arête par arête jusqu'à maximalité, et termine en temps polynomial. On obtient donc un algorithme en temps polynomial capable de décider, étant donné un hypergraphe k -uniforme et deux sommets x et y , s'il existe un chemin $(k - 2)$ -linéaire entre x et y . Nous mentionnons deux conséquences de ce résultat algorithmique. La première conséquence est liée à un problème de théorie des graphes appelé PAFP ou parfois PFPP ("paths avoiding forbidden pairs"). La question est de savoir, étant donné un graphe arête-bicoloré bleu/rouge G et deux sommets u et v , s'il existe un chemin induit bleu entre u et v dans G . Si G est le *line graph bicoloré* d'un hypergraphe k -uniforme H , défini comme le line graph standard dont on colore les arêtes en bleu pour une intersection de taille entre 1 et $k - 2$ ou en rouge pour une intersection de taille $k - 1$, alors l'existence d'un chemin induit bleu entre u et v dans G équivaut trivialement à l'existence d'un chemin $(k - 2)$ -linéaire dans H reliant les deux (hyper)arêtes correspondant à u et v . Ainsi, le problème PAFP, connu comme étant NP-complet en général [GMO76], est résolu en temps polynomial sur les line graphs bicolorés pour lesquels on peut efficacement : d'une part les identifier en tant que tels, et d'autre part reconstruire un hypergraphe antécédent. Malheureusement, ce problème d'identification et de reconstruction, qui est facile pour les line graphs de graphes, est NP-complet pour les line graphs d'hypergraphes même 3-uniformes [PRT81]. Cependant, il n'est pas impossible que l'ajout des couleurs simplifie le problème, et permette parfois d'obtenir une caractérisation par sous-graphes induits interdits comme il en existe pour les line graphs standards d'hypergraphes dans certains cas.

La deuxième conséquence est que le jeu Maker-Breaker sur les hypergraphes de rang 3 est résolu en temps polynomial, puisque, comme expliqué précédemment, il se réduit au cas $k = 3$ de notre problème de connectivité (chaînes dans les hypergraphes 3-uniformes). Il s'agit du résultat algorithmique principal de cette thèse. On sait que décider l'issue du jeu Maker-Breaker sur les hypergraphes 6-uniformes est PSPACE-complet [RW21], mais la question algorithmique reste ouverte pour les hypergraphes de rang 4 ou 5.

La complexité du jeu Maker-Breaker ne s'arrête pas nécessairement à l'acceptation algorithmique du mot "complexité". En particulier, nous nous intéressons aux deux paramètres d'hypergraphes présentés dans l'introduction, notés τ_M et θ_M (correspondant respectivement à une optimisation en tours ou en jetons de la part de Maker), qui aident à mesurer à quel point les stratégies gagnantes de Maker sont complexes. En effet, étant donné un hypergraphe H sur lequel Maker gagne : une valeur élevée de $\tau_M(H)$ signifie que ces stratégies sont longues, tandis qu'une valeur élevée de $\theta_M(H)$ signifie en substance que ces stratégies reposent sur un grand nombre de menaces simultanées. Notre but est d'évaluer quel est le maximum atteint par chacun de ces paramètres (hors valeur infinie : on suppose que Maker gagne) sur la classe des hypergraphes de rang k , pour chaque k fixé. Le paramètre τ_M est très analysé dans la littérature, mais aucune étude n'avait été faite en fonction du rang. Quant au paramètre θ_M , il n'avait jamais été introduit avant cette thèse. Tout d'abord, il est à noter qu'on a trivialement l'inégalité $\theta_M(H) \leq \tau_M(H) \leq \lceil \frac{|V(H)|}{2} \rceil$.

Dans les hypergraphes de rang 3, la réponse vient comme corollaire du résultat structurel du chapitre précédent. Si Maker gagne sur un hypergraphe marqué 3-uniforme H , alors elle peut faire apparaître un nunchaku/necklace N après au plus trois tours, et on a vu que Maker peut gagner sur N par forçage. En réalité, le forçage est coûteux en temps, car chaque sommet de N finira par être joué. Au lieu de cela, Maker peut gagner par dichotomie : Maker joue au centre de N , ce qui crée deux nunchakus de la moitié de la taille de N , Breaker ne peut pas détruire les deux, ainsi Maker peut jouer au centre du nunchaku resté intact, etc. Cette méthode permet à Maker de gagner en $O(\log(|V(H)|))$ tours et en utilisant seulement trois jetons, de plus il est impossible de faire mieux sur un nunchaku donc ces bornes sont optimales.

En revanche, les choses sont très différentes dans les hypergraphes de rang k où $k \geq 4$. Il n'est pas difficile de généraliser le principe de forçage, en remplaçant le nunchaku par un *chemin de forçage* $(k - 2)$ -linéaire bien choisi. On constate alors que la méthode par dichotomie n'a pas son équivalent pour $k \geq 4$, si bien que le forçage est la seule manière de gagner sur un tel chemin : nous obtenons ainsi, pour tout n , un hypergraphe k -uniforme H à n sommets tel que $\tau_M(H) = \lceil \frac{n}{2} \rceil$, ce qui réalise la borne triviale. En ce qui concerne θ_M , l'exemple est plus difficile à construire et à démontrer, mais nous obtenons, pour tout n , un hypergraphe k -uniforme H à n sommets tel que $\theta_M(H) \geq \frac{n}{6}$, soit le même ordre de grandeur que la borne triviale. En conclusion, à partir du rang 4, il existe des exemples où Maker a besoin de beaucoup de temps et de jetons pour mener à bien sa stratégie gagnante, alors que ce n'est jamais le cas pour le rang 3. On observe ainsi un saut de complexité, et on peut penser qu'il en est de même au sens algorithmique : ceci conforte l'intuition prédominante dans la communauté selon laquelle décider l'issue du jeu Maker-Breaker sur les hypergraphes de rang 4 serait PSPACE-complet.

Chapitre IV : Des jeux positionnels aux problèmes de reconfiguration

Considérons le jeu Maker-Breaker, dans une variante où les deux joueurs ont un nombre de jetons fixe à leur disposition : α jetons rouges pour Maker, β jetons bleus pour Breaker. On suppose que α est fini. A noter que le cas $\beta = \infty$ correspond à l'étude de θ_M dans le chapitre précédent, à la différence qu'ici on fixe α plutôt que de chercher la valeur minimum de α telle que Maker gagne.

Après α tours de jeu, Maker a placé tous les jetons rouges sur un certain ensemble de sommets A . L'objectif de Maker à partir de là est que, à terme, ses jetons recouvrent entièrement l'une des arêtes de l'hypergraphe. En notant \mathcal{B} l'ensemble des parties de $V(H)$ de taille α contenant au moins une arête de H , on peut donc formuler la situation comme suit. Alice va déplacer les jetons rouges, un seul à la fois, pour tenter de les faire passer de la configuration A à une configuration $B \in \mathcal{B}$. A chaque déplacement de jeton, Alice n'a pas de contrainte sur l'*origine* du jeton (elle déplace le jeton rouge de son choix), en revanche elle est restreinte sur sa *destination* (certaines positions sont interdites).

Bien sûr, dans le jeu que l'on considère ici, la restriction sur la destination est due aux coups de Bob : les positions interdites sont les sommets sur lesquels il y a un jeton bleu. Cependant, on peut imaginer d'autres restrictions, définissant autant de jeux différents. En particulier, si les positions interdites étaient déterminées non pas par un deuxième joueur mais par une règle "automatique" fixée avant le début de la partie, alors on obtiendrait un type particulier de *problèmes de reconfiguration*. Ce terme regroupe les problèmes où, étant donné une certaine structure discrète munie d'une notion de *configuration* et d'une règle définissant les *déplacements* autorisés entre les configurations, on étudie la possibilité de passer d'une configuration donnée à une autre par une suite finie de déplacements. Les problèmes de reconfiguration de la forme ci-dessus, c'est-à-dire les problèmes de déplacements de jetons avec restriction sur la destination, peuvent être assimilés à une version à un joueur du jeu Maker-Breaker.

Demaine et al. [DDV02] ont introduit des puzzles de déplacements de pièces dans la grille carrée qui rentrent dans cette catégorie. La restriction est qu'une pièce ne peut être déplacée que vers une case vide dont au moins deux cases voisines orthogonales sont déjà occupées par des pièces. Une conséquence de cette restriction est qu'il est impossible, quel que soit le nombre de déplacements effectués, de placer une pièce en-dehors d'un certain ensemble fini ayant la forme

d'une union de rectangles. Cet ensemble, qui ne dépend que de la configuration de départ A , est appelé le *span* de A . En particulier, une condition nécessaire pour pouvoir passer de A à B est que le span de A contienne celui de B . Demaine et al. ont identifié une notion cruciale de *pièces bonus*, qui sont des pièces qu'on pourrait retirer à la configuration A tout en préservant cette inclusion des spans. Le nombre de pièces bonus d'un puzzle quantifie en quelque sorte la marge de manœuvre qu'a le joueur pour ses déplacements. Les auteurs séparent les puzzles non triviaux en deux catégories, selon qu'ils possèdent deux pièces bonus ou bien une seule. Ce dernier cas est partiellement étudié dans mon mémoire de master [Gal19].

Il est affirmé dans [DDV02] que deux pièces bonus, en plus d'une condition similaire sur la configuration d'arrivée, sont toujours suffisantes pour résoudre un puzzle. Ce qui a échappé aux auteurs est que l'inclusion des spans ne suffit pas : des problèmes peuvent survenir lorsque le puzzle requiert de scinder le span en plusieurs morceaux très éloignés. Nous construisons ainsi une famille de contre-exemples à cette affirmation, qui montrent même qu'aucun nombre constant de pièces bonus n'est suffisant pour garantir la résolubilité d'un puzzle. En revanche, nous montrons que deux pièces bonus suffisent toujours pour les puzzles dont le nombre total de pièces est suffisamment grand. Notre borne est même serrée dans le sens où, pour tout N inférieur à cette borne, il existe des puzzles comptant N pièces au total et qui ne sont pas résolubles. Notre algorithme est en deux parties. La première réutilise la technique de Demaine et al., qui consiste à utiliser les deux pièces bonus pour transformer une configuration en sa *configuration canonique* associée. La deuxième est une méthode nouvelle, à la fois efficace et naturelle, consistant en un unique balayage du plateau au cours duquel on dépose les pièces aux positions requises.

Conclusion

Au cours de cette thèse, j'ai étudié divers problèmes autour des jeux et des hypergraphes, qui n'ont pas tous conduit au même sentiment d'accomplissement.

Tout d'abord, les puzzles de déplacements de pièces sur la grille carrée sont encore loin d'être entièrement compris, non seulement dans le cas d'une seule pièce bonus [Gal19] mais aussi dans le cas de deux pièces bonus qui est plus compliqué que ce qui était pensé dans [DDV02]. Nous avons identifié que la difficulté repose dans le fait de scinder le span, et nous avons montré comment surpasser cet obstacle dans certains cas. Il s'avère que la quantité même de pièces joue un grand rôle dans la résolubilité des puzzles, une surprise qui a contrasté avec mes habitudes considérations structurelles dans les jeux. Au final, de nombreux cas demeurent ouverts, et nous n'avons pas de réelles pistes pour les attaquer. Nous quittons ainsi ce problème avec davantage de questions que nous en avions en arrivant.

Au contraire, l'étude du jeu Maker-Breaker sur les hypergraphes de rang 3 a été bien plus satisfaisante. Nous avons obtenu la caractérisation structurelle de l'issue et la description des stratégies optimales que nous désirions, en termes d'intersections de dangers. Il est dommage que la preuve ne soit pas aussi esthétique que le résultat. Je ne peux pas exclure la possibilité qu'il existe une preuve plus courte et moins technique, concentrant l'essence de ce qui fait fondamentalement fonctionner le raisonnement. A partir de ce résultat, nous avons obtenu l'algorithme en temps polynomial que nous espérions. Il n'était pas évident, notamment avec 3-SAT à l'esprit, qu'il n'y aurait pas un saut de complexité entre les rangs 2 et 3. Quant aux études futures, on peut se demander si l'approche par les dangers pourrait être utile pour d'autres classes d'hypergraphes. En particulier, pour $k \geq 4$, existe-t-il un nombre constant de tours $r(k)$ tel que Maker gagne sur un hypergraphe de rang k si et seulement si elle peut garantir l'apparition d'un chemin de forçage pendant les $r(k)$ premiers tours ? Le problème d'existence

d'un chemin de forçage étant clairement dans NP, on sait que la réponse est négative pour $k \geq 6$ à moins que NP=PSPACE, puisque décider l'issue du jeu Maker-Breaker dans ce cas est PSPACE-complet [RW21]. La question reste cependant ouverte pour $k \in \{4, 5\}$, bien qu'il soit généralement suspecté que ces cas soient également PSPACE-complets. On pourrait également considérer d'autres classes d'hypergraphes, qui ne soient pas définies par le rang.

En ce qui concerne les bornes sur les paramètres d'hypergraphes τ_M et θ_M (optimisation en tours et en jetons respectivement) en fonction du rang k , nous avons pratiquement obtenu des réponses exactes. Pour $k = 3$, nous avons montré qu'un nombre logarithmique de tours et seulement trois jetons suffisent toujours à Maker pour mener à bien sa stratégie gagnante (à savoir, faire apparaître un nunchaku/necklace dans les trois premiers tours puis gagner par dichotomie), et que ces bornes sont serrées. Pour $k \geq 4$, les choses sont radicalement différentes, puisque la borne triviale (nombre de sommets divisé par deux) est atteinte par τ_M et quasiment atteinte par θ_M . En un certain sens, les stratégies gagnantes de Maker sont donc bien plus complexes dans le cas $k \geq 4$ qu'elles ne le sont lorsque $k = 3$, ce qui conforte moralement l'intuition selon laquelle décider l'issue du jeu Maker-Breaker serait PSPACE-complet pour $k \in \{4, 5\}$. Pour aller plus loin, une idée pourrait être d'étudier ces bornes dans la version biaisée du jeu.

Enfin, nous avons exploré certains problèmes d'hypergraphes. Nous avons introduit la notion de λ -linéarité pour étudier des chemins dans lesquels la taille des intersections d'arêtes est majorée par un λ fixé. En particulier, nous avons examiné le problème d'existence de chemins $(k - 2)$ -linéaires dans les hypergraphes de rang k . Nous avons montré que les composantes connexes associées sont caractérisées par leur structure d'archipel maximal, et qu'elles peuvent être calculées en temps polynomial. A l'inverse de notre étude du jeu Maker-Breaker sur les hypergraphes de rang 3, nous avons ici des structures très visuelles et des preuves relativement simples. Un lien a également été établi avec le problème PAFP, à travers un concept de line graph bicoloré d'un hypergraphe portant une information supplémentaire sur la taille des intersections d'(hyper)arêtes.

Dans le futur, j'aimerais également étudier d'autres jeux en lien avec le jeu Maker-Breaker sur les hypergraphes de rang 3.

Le jeu Maker-Maker sur les hypergraphes de rang 3 est une continuation naturelle. Sa complexité algorithmique est inconnue. La technique de forçage, qui n'était pas vitale en convention Maker-Breaker puisqu'on pouvait la supplanter par la technique de dichotomie, pourrait s'avérer essentielle ici. En effet, elle permet à un joueur de contrôler exactement quels sommets il/elle laisse à son adversaire, ce qui peut aider à l'empêcher de créer des menaces de son côté. Il semble important pour le premier joueur de garder l'initiative dans son attaque. Le deuxième joueur peut parfois s'emparer de l'initiative, et ainsi obtenir une partie nulle, sans nécessairement détruire tous les hypergraphes qui constituent des dangers en convention Maker-Breaker. C'est ce qui arrive au tic-tac-toe par exemple. Par conséquent, il existe peut-être une notion différente d'intersection de dangers en convention Maker-Maker, telle que jouer un sommet adjacent à un danger soit parfois suffisant pour le détruire.

Le jeu Avoïder-Enforcer sur les hypergraphes de rang 3 est un autre sujet d'intérêt. J'étudie actuellement le cas linéaire conjointement avec Valentin Gledel¹ et Aline Parreau². Tout comme en convention Maker-Breaker, le cas non linéaire semble plus compliqué. Il est possible que les lemmes structurels sur les chaînes et les cycles établis dans cette thèse trouvent une nouvelle utilité ici.

Enfin, je pense que notre idée d'introduire des jetons dans le jeu a du potentiel. On pourrait étudier des jeux positionnels à jetons, où chaque joueur a un nombre fixé de jetons à sa disposition

¹ Université d'Umeå, Suède.

² CNRS. LIRIS, Université Lyon 1, France.

(fini ou infini, et pas nécessairement le même nombre pour les deux). Cette version est naturelle, de plus elle est conforme à la manière dont certains jeux positionnels étaient joués historiquement [Kra42][Gar59]. Nacim Oijid¹ et moi-même sommes intéressés par la complexité algorithmique de ces jeux. Il est probable que le problème reste PSPACE-complet en général pour toutes les conventions.

Je termine avec une remarque concernant une potentielle unification et généralisation de certains jeux positionnels.

Un *transversal* d'un ensemble d'arêtes E est un ensemble de sommets T tel que $T \cap e \neq \emptyset$ pour tout $e \in E$. Etant donné un ensemble d'arêtes E , soit $t(E)$ l'ensemble des transversaux minimaux de E (notons que t est une involution). Dans le jeu Maker-Breaker sur un hypergraphe H , le but de Maker est de colorer une arête de H en rouge, tandis que le but de Breaker revient à colorer un transversal de $E(H)$ en bleu. Ainsi, l'appellation du jeu est faussement asymétrique quant aux rôles des deux joueurs : en réalité, ils sont tous les deux "Maker", mais pas sur les mêmes arêtes. On peut donc définir un jeu plus général où : on a deux ensembles d'arêtes E_A et E_B sur le même ensemble de sommets V , Alice colore des sommets en rouge pour obtenir une arête monochrome rouge dans E_A , Bob colore des sommets en bleu pour obtenir une arête monochrome bleue dans E_B , et le premier joueur qui remplit son objectif a gagné. Une partie peut être nulle. Ceci définirait une famille unifiée des jeux d'accomplissement, contenant les jeux Maker-Maker et Maker-Breaker mais également tout ce qui existe entre les deux. Le jeu Maker-Maker correspond au cas $E_A = E_B$. Le jeu Maker-Breaker correspond au cas $E_A = t(E_B)$ i.e. $E_B = t(E_A)$.

Il serait instructif de voir, parmi les principes valables dans les conventions Maker-Maker et Maker-Breaker, lesquels restent vrais pour le jeu d'accomplissement général décrit ci-dessus. En effet, certains sont dus au lien particulier entre E_A et E_B , alors que d'autres sont inhérents à la nature de jeu d'accomplissement. Par exemple, le vol de stratégie continue de s'appliquer, montrant que les deux joueurs préfèrent jouer en premier et sont toujours contents de jouer plus de coups. Une autre perspective serait de résoudre le jeu d'accomplissement général pour des classes d'hypergraphes simples. Je pense que le cas 2-uniforme, c'est-à-dire où tous les éléments de $E_A \cup E_B$ sont de taille 2, serait un bon début. A première vue, les couplages et les chemins augmentants semblent jouer un rôle significatif.

On pourrait pousser la généralisation encore plus loin, en considérant une version *sommet-partisane* où $V = V_A \cup V_B$ (pas nécessairement une partition). Alice ne peut colorer que les sommets de V_A , et *compléter* une arête $e \in E_A$ signifie pour elle que tous les sommets de $e \cap V_A$ sont colorés en rouge avant qu'un seul sommet de $e \cap V_B$ ne soit coloré en bleu. Les règles sont analogues pour Bob, et le premier joueur qui complète une arête a gagné, sans quoi la partie est nulle. En particulier, cette version inclut des jeux naturels qui ne sont pas classifiés comme des jeux positionnels autrement, comme par exemple la variante du jeu Maker-Breaker où Breaker sélectionne des arêtes et non pas des sommets. Ceci signifie que, à chaque tour, Maker choisit un sommet puis Breaker retire une arête (sans toucher aux sommets de cette arête). Ce jeu peut être modélisé comme un jeu sommet-partisan, en ajoutant dans chaque arête un sommet qui n'est jouable que par Bob, le reste n'étant jouable que par Alice.

¹ LIRIS, Université Lyon 1, France.

Introduction

As I proudly obtained my first ever research result, solving a conjecture by mathematician Persi Diaconis on the riffle-shuffle card-guessing game, I realized something: at heart, what I enjoyed so much was not really probability theory, but rather combinatorics and mathematical games. After a quick internet search for PhD theses on the subject in France, the same name seemed to come up again and again, and it soon became apparent that it would be a good idea to email a certain Sylvain Gravier. One enthusiastic response later, I was on my way to Grenoble, where Sylvain introduced me to Isabelle Sivignon. For my Master's internship, the two of them had prepared a subject about some coin-moving puzzles, a one-player game which I ended up coming back to during my PhD (but more on that later). As a reward at the end of my internship, I unlocked a second player, as Sylvain and Isabelle offered me to pursue my studies with them on the topic of some two-player games called *positional games*. Cue the example I have used countless times in the last few years:

- So, what is it exactly that you do?
- OK: you know what tic-tac-toe is, right?

Tic-tac-toe, also known as *Naughts and crosses*, is the most famous positional game. Two players, Alice and Bob, take turns claiming cells of a 3×3 grid, by drawing naughts and crosses respectively. The player who first claims three aligned cells wins. If neither player succeeds, which is the well-known outcome when both players play perfectly, then the game is a draw. Imagine that, growing tired of all these draws (and with trickery in mind), Alice offers Bob to change the rules slightly. She will play first, trying to align three naughts, and Bob's goal will "simply" be to prevent her from doing so. This way, there will always be a winner. The game becomes what is called a *Maker-Breaker* positional game, as Alice ("Maker") attempts to *make* an alignment and Bob ("Breaker") tries to *break* her dreams. It turns out Alice is the one who has a winning strategy in this particular case, but this game can be greatly generalized, forming the family of Maker-Breaker games which are the main topic of my PhD thesis.

- Wait... you're not getting paid to play games, are you?
- Well, if it helps, we're not being paid much...

To understand how this generalization works, think that the above game is fundamentally defined as follows: there are nine possible positions that the players can claim, which are the nine cells, and eight possible ways for Alice to win, which correspond to the eight possible alignments (three rows, three columns, two diagonals). Since this is the only relevant information that defines the game, let us consider an alternative graphical representation. We draw nine dots, one for each cell, and eight potatoes, one for each winning alignment. Instead of drawing naughts (resp. crosses), Alice (resp. Bob) colors the corresponding dots in red (resp. blue). See Figure 3. Alice wins if at any point there exists a potato in which all dots are red, otherwise Bob wins. Now, drawing a board made of any number of dots and any set of potatoes (of any size, not even the same size for all necessarily), and playing with the same rules, defines a new

Maker-Breaker game. A collection of dots and potatoes is called a *hypergraph*: the dots are the *vertices*, the potatoes are the *edges*. There are as many Maker-Breaker games as there are hypergraphs, which is to say, a lot.

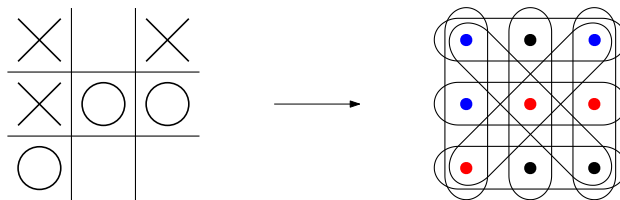


Figure 3: Two representations of the same game (in progress).

Hypergraphs are very general objects indeed: a set V , a set E of subsets of V , and that is it. Therefore, their structure can be very complex. This explains why, in most general results from the literature, assumptions are not related to structure but instead have to do with global quantities such as the total number of edges. The main goal set at the start of my PhD was to contribute to filling this void by getting structural results on the Maker-Breaker game. This means obtaining criteria for the *outcome* of the game (who wins with optimal play?) based purely on the hypergraph structure. The algorithmic aspects were another area of interest. It was long-known already that no efficient algorithm exists to solve the Maker-Breaker game in all generality i.e. to decide, given any hypergraph, what is the outcome for the Maker-Breaker game played on that hypergraph. More specifically, and in more technical terms, this problem is PSPACE-complete [Sch78].

Therefore, the idea was to restrict the problem to some families of hypergraphs with a simpler structure: I could then hope to get structural characterizations for the outcome, from which polynomial-time algorithms could potentially be derived. We identified two such families which could be a good place for me to start:

- Hypergraphs of a geometrical nature were a first idea. For instance, consider hypergraphs like that of tic-tac-toe, where the vertices can be identified with the cells of a rectangular grid so that the edges correspond to alignments (not necessarily all possible alignments, but some of them). We also thought of hypergraphs that represent intersections of geometrical objects in the plane, such as rectangles.
- Hypergraphs of small *rank* (size of a biggest edge) were another idea. The smaller the edges, the less complex the structure. A lot is known about the structure of graphs, which are nothing but hypergraphs where all edges are of size 2, and the game turns out to be trivial in this case. However, a lot less was known about the structure of hypergraphs of rank 3, and just a single paper of Kutz [Kut04] existed about the Maker-Breaker game in this case (solving a subcase). Despite the jump in edge size from 2 to 3 complicating things a lot, it still felt like there was room for structural results on general hypergraphs of rank 3. On the other hand, the algorithmic point of view made us less optimistic, as it is frequent for problems in combinatorial optimization and propositional logic to display a complexity gap between the values 2 and 3. The boolean satisfiability problem is a notorious example of this: 2-SAT is tractable, but allowing clauses of size 3 makes the problem NP-complete.

I first looked at hypergraphs combining both ideas above. However, it appeared to us that the rank 3 was the deciding factor, in the sense that adding geometrical traits on top did not seem to make the problem easier.

Therefore, I quickly turned my focus to the Maker-Breaker game on general hypergraphs of rank 3. Kutz's paper was by far my most influential read during this PhD. His result gave a structural characterization for the outcome in the *linear* subcase, meaning any two distinct edges intersect on at most one vertex (as is the case in tic-tac-toe for instance). The structure

in question is identifiable efficiently, so that the outcome can be determined in polynomial time. I set out to extend both the structural result and the algorithmic result to general hypergraphs of rank 3. One of the tools that Kutz uses to establish a Maker win consists in finding some vertex x such that, if Maker picks x as her first move, then several urgent threats are created from the point of view of Breaker, who has no move that addresses all these threats at once. In other words, if there exists some x creating threats, which I call *dangers at x* , that do not intersect, then Maker has a winning strategy with x as her first move. I opted for an angle that differs from Kutz's in that I made this notion of danger intersections the center of my approach rather than an occasional tool. This idea is valid for any hypergraph, regardless of the rank. However, in hypergraphs of rank 3 specifically, I was under the impression that the outcome could be read directly on the hypergraph in terms of danger intersections, where we would only need to take into account specific dangers of a somewhat elementary type. In particular, this would yield a polynomial-time algorithm determining the outcome of the Maker-Breaker game on all hypergraphs of rank 3, thus solving a valuable open problem. Settling this conjecture and writing out everything to do with dangers ended up taking up most of my time and effort throughout my PhD. For one, the proofs are very long and technical. Moreover, many arguments of a different nature are used: identifying them, sorting out which are game-related and which are hypergraph-related, understanding what fundamentally makes them work and what is the most general context in which they apply, was a long process.

Even though this PhD thesis is mainly about games, it is also about hypergraphs. Indeed, I ended up working on hypergraph problems which, although initially motivated by the Maker-Breaker game in my case, are more than just a tool to study games and have potential for applications beyond. Most notably, a type of linear path called a *chain* plays a key role for the Maker-Breaker game on hypergraphs of rank 3, with the existence of a chain between two given vertices proving to be a crucial question. Therefore, we were led to investigate a particular connectivity problem in hypergraphs, which is interesting on its own: chains are natural paths to consider in hypergraphs of rank 3, and many papers that are not game-related already featured chains in other contexts. The structural study around chain connectivity, along with the associated algorithmic problem, constitutes an important part of this PhD thesis. These questions turned out to be linked with a problem around line graphs of hypergraphs, a topic which is also very present in the literature. Finally, some structural properties of elementary hypergraphs of rank 3 such as chains and cycles, which are key preliminary lemmas for our results on the Maker-Breaker game, could be useful for any structural study in hypergraphs of rank 3.

I also studied two hypergraph parameters related to the Maker-Breaker game, both corresponding to Maker strategies which optimize a certain quantity. The first quantity is the number of rounds: if Maker has a winning strategy, then what is the minimum amount of time that she needs, i.e. what is the minimum number of rounds in which she can ensure that she will get some edge with all red vertices? This question is vastly studied in the Maker-Breaker literature, however nothing was known specifically about hypergraphs with small edges. The second quantity is the number of *tokens*, in the following variation of the Maker-Breaker game which we introduce in this PhD thesis. Suppose that Maker, instead of permanently coloring vertices in red, places red tokens on the vertices that she picks. In each round, she has the choice between moving a token that she had placed previously (thus losing possession of a vertex, which may later be stolen by Breaker) or using a new token (say she has infinite tokens at her disposal). Breaker, on the other hand, permanently colors vertices in blue as usual. If Maker has a winning strategy, then what is the minimum number of tokens that she needs? In the context of this PhD thesis, I prefer to use an optimization viewpoint, but this idea could also be incorporated in the rules of the game, for both players actually: give them a fixed number of tokens (red and blue respectively) to play with, finite or infinite and not necessarily the same for

both, and see who wins. These two parameters are indicators of how complex Maker's winning strategies are on a given hypergraph. My goal was to evaluate the maximum value of both parameters depending on the size of the edges. In particular, for hypergraphs of rank 3, the structural results around dangers which I alluded to previously proved to be key in answering this question.

In the version of the Maker-Breaker game played with limited tokens, Maker tries to rearrange her tokens into any configuration which contains an edge, while Breaker impedes her by blocking vertices (indeed, vertices occupied by Breaker's tokens are unavailable for Maker). Now, if these blocked vertices were not chosen intelligently by a second player adapting to Maker's moves, but were instead chosen automatically by some preestablished rule, then the game would become a reconfiguration problem. This particular type of reconfiguration problem can thus be likened to a one-player version of the Maker-Breaker game. Demaine et al. [DDV02] introduced some coin-moving puzzles in the square grid which fall into this category. The restriction rule is that a coin (i.e. token) can only be moved to an empty square that has at least two orthogonally adjacent squares occupied by other coins. I had studied this game during my Master's internship, and towards the end I discovered an inaccuracy in the main result of [DDV02] with too little time to delve deeper into it. I came back to this problem during my PhD, to investigate the extent of the authors' mistake and look for alternative results.

The organization of this dissertation is as follows. There are four chapters, themselves divided into sections. Apart from the introductory Chapter I, each chapter corresponds to a research theme, and each section corresponds to a specific research problem inside that theme. Moreover, each section begins with a presentation of its associated problem along with some state of the art, and ends with some concluding remarks and prospects around that problem.

- Chapter I presents all the notions that feature in this dissertation, as well as preliminary results that prepare the deeper studies from future chapters.

One thing that I felt was missing in the literature was a place compiling all the most basic results around positional games, not only the Maker-Breaker game but also other conventions. This is why Section I.1 contains a list of easy strategic principles, stated in the most general context possible. After that, I provide some state of the art around the Maker-Breaker game specifically, and I then present my personal approach towards this game. In particular, marked hypergraphs are introduced, which we use to update the board throughout the game in a way that keeps track of Maker's moves.

Section I.2 is dedicated to the aforementioned notion of danger. We adopt Breaker's point of view: in short, Breaker must be able to destroy the dangers created by Maker, so those must intersect. This brings up the topic of intersecting collections of subhypergraphs. If a collection is not intersecting but it is Breaker's turn, can he make it intersecting and thus be safe in the next round? When can Breaker ensure that dangers of a given type will intersect, not just in the first round, but during several rounds, or during the whole game as needed?

Finally, Section I.3 explores some elementary structures in hypergraphs, such as paths and cycles. The emphasis is on hypergraphs of rank 3, where the size of the edges allows us to establish several structural lemmas that are the building blocks for the next chapter.

- Chapter II carries out the in-depth structural studies in hypergraphs that are at the core of this dissertation.

First of all, Section II.1 is dedicated to the Maker-Breaker game on hypergraphs of rank 3. We obtain a structural characterization for the outcome of the Maker-Breaker game on hypergraphs of rank 3, as well as optimal strategies for both players, all based on danger intersections. This is probably the most central result of this PhD thesis. More specifically, we exhibit a family of dangers such that Breaker has a winning strategy if and only if

those dangers at x intersect for any possible first move x of Maker, in which case any Breaker answer inside their intersection is optimal. What is remarkable is that we only ask for the intersection to be nonempty in the first round, but that ends up staying the case in all subsequent rounds. The dangers in said family are unions of the elementary structures which we mentioned previously, in such a way that our result has the following consequence which validates a conjecture by Rahman and Watson [RW20]: Maker wins if and only if she can ensure that, after at most three rounds of play, she has forced the appearance of a *nunchaku* or a *necklace*. Those two are elementary hypergraphs on which Maker has a straightforward winning strategy that forces all of Breaker's moves.

Since a nunchaku is a chain, and a necklace is a cycle i.e. a chain that loops onto itself basically, this last result shows that, on hypergraphs of rank 3, the Maker-Breaker game is directly related to questions of existence of chains. We thus explore this connectivity problem in Section II.2. We show that the connected components associated to this problem have a very specific structure, which actually characterizes them. All ideas generalize to hypergraphs of any rank k , when replacing the notion of linearity that defines chains with a notion of $(k - 2)$ -*linearity*: consecutive edges of the path must intersect on at most $k - 2$ vertices. Therefore, this section is written in the general case.

- Chapter III reaps the algorithmic rewards of the structural studies from the previous chapter, along with some further investigations.

Section III.1 details how the aforementioned connected components can be computed efficiently thanks to the structural result. A link is made with the "paths avoiding forbidden pairs" problem (PAFP), an NP-complete graph connectivity problem from the literature, for which our algorithm may imply some new tractable cases.

We go back to the Maker-Breaker game in Section III.2. The main algorithmic result of this PhD thesis is that the outcome of the Maker-Breaker game can be determined in polynomial time on hypergraphs of rank 3. Indeed, the structural result reduces this problem to the chain connectivity problem, which is solved in polynomial time by the previous algorithm. We also address other complexity facets of the game besides the algorithmic aspect, by studying the two hypergraph parameters that we introduced earlier (optimization by Maker of the number of rounds and tokens respectively). A corollary of the structural results from the previous chapter is that, in hypergraphs of rank 3, Maker never needs more than a logarithmic number of rounds and just three tokens. Things are very different in hypergraphs of rank 4 or more, where Maker sometimes needs almost the whole board to be covered before she completes the win.

- Chapter IV contains results obtained when, during my PhD, I came back to the coin-moving puzzles which I had studied during my Master's internship.

Section IV.1 is an exception to the organization of this dissertation, in that it does not contain new significant results on a particular problem, but rather lists some preliminary properties of this game which are essential for the next section. Most of these results were already given in [DDV02]. In particular, a crucial information is the number of *extra coins* that a puzzle contains, which quantifies a form of margin for maneuvering coins.

Section IV.2 addresses puzzles with (at least) two extra coins. It was roughly claimed in [DDV02] that all such puzzles were solvable. However, we show that this is untrue. Actually, we exhibit worst-case puzzles which show that no constant number of extra coins (let alone two) is sufficient to guarantee solvability. On the other hand, we show that all puzzles with two extra coins and a "large" number of total coins are solvable. The proof uses a new algorithm to solve puzzles, which is both natural and efficient.

Chapter I

Preparatory notions and preliminary results

This chapter introduces all the objects that will be studied in the next two chapters, and all the notions that will help in that direction. Preexisting concepts are presented together with some state of the art. A few new concepts are also established, along with some first results that illustrate their interest or serve as preliminary for the upcoming chapters.

I.1 The Maker-Breaker game: an introduction

The primary object of study in the present dissertation is the Maker-Breaker game, which is part of the family of positional games. This section provides some background around positional games in general, and then focuses on the Maker-Breaker game specifically.

I.1.1 State of the art on positional games

I.1.1.1 Some vocabulary around hypergraphs

In positional games, the game board is a hypergraph. Therefore, let us start with some general (and, mostly, standard) definitions and notations regarding hypergraphs.

Definition I.1.1. A *hypergraph* H is defined by a finite *vertex set* $V(H)$ and an *edge set* $E(H)$ consisting of subsets of $V(H)$.

Notation I.1.2. A hypergraph H consisting of a single edge e , i.e. such that $V(H) = e$ and $E(H) = \{e\}$, may be simply denoted by e .

Definition I.1.3. Let H be a hypergraph.

- A *subhypergraph* of H is a hypergraph K such that $V(K) \subseteq V(H)$ and $E(K) \subseteq E(H)$. A *strict subhypergraph* of H is a subhypergraph K of H such that $K \neq H$. We may say H *contains* K , or write $K \subseteq H$, to signify that K is a subhypergraph of H .
- Let $U \subseteq V(H)$. The *subhypergraph of H induced by U* is the subhypergraph of H , denoted by $H[U]$, defined by $V(H[U]) = U$ and $E(H[U]) = \{e \in E(H), e \subseteq U\}$.

Definition I.1.4. Let H be a hypergraph.

- We say H is k -uniform if all its edges are of size exactly k .
- The *rank* of H , denoted by $\text{rk}(H)$, is defined as the size of its biggest edge.
- The *anti-rank* of H , denoted by $\text{ark}(H)$, is defined as the size of its smallest edge.

Remark. A 2-uniform hypergraph is a *graph*.

Definition I.1.5. Let H be a hypergraph.

- Let $v_1, v_2 \in V(H)$ be distinct. We say v_1 and v_2 are *adjacent in H* if there exists $e \in E(H)$ such that $\{v_1, v_2\} \subseteq e$.
- Let $v \in V(H)$. We say an edge $e \in E(H)$ is *incident to v* if $v \in e$. The *degree of v in H* is defined as the number of edges of H that are incident to v .

Definition I.1.6. Let H be a hypergraph, and let $v \in V(H)$. We say v is an *inner vertex* of H if v is of degree at least 2 in H , otherwise we say v is an *outer vertex* of H . We denote by $\text{inn}(H)$ (resp. $\text{out}(H)$) the set of all inner (resp. outer) vertices of H .

Let us also introduce a useful notation which is not hypergraph-related in itself:

Notation I.1.7. Let S be a set, and let m be an integer. We denote by $\mathbb{P}_m(S)$ the set of all subsets of S that are exactly of size m .

I.1.1.2 Various conventions and problems

Definition I.1.8. A *positional game* is a two-player game played on a hypergraph H , in which Alice and Bob alternate turns coloring previously uncolored vertices of H in red and blue respectively. In the standard *unbiased* version, vertices are colored one by one, but it is possible to introduce a *bias* ($p : q$) meaning Alice and Bob respectively color p vertices and q vertices at once in each round (except possibly for the final round: if there are not enough vertices left then the player simply colors all the remaining vertices). The result of the game is decided by one of several *conventions*, all of which are about monochromatic edges i.e. edges whose vertices are all of the same color, the main four being the following:

- *Maker-Maker* convention: the first player to complete a monochromatic edge of their color wins; if that never happens then the game ends in a draw.
- *Maker-Breaker* convention: Alice ("Maker") wins if she completes a monochromatic red edge at any point, otherwise she loses; in particular no draw is possible.
- *Avoider-Avoider* convention: the first player to complete a monochromatic edge of their color loses; if that never happens then the game ends in a draw.
- *Avoider-Enforcer* convention: Alice ("Avoider") loses if she completes a monochromatic red edge at any point, otherwise she wins; in particular no draw is possible.

Note that we have not specified which of Alice or Bob plays the first move (which is equivalent to specifying who plays the last move in the case where all vertices end up colored). This information is important and both cases shall be considered, even in the Maker-Maker and Avoider-Avoider conventions since the bias may be unbalanced (i.e. $p \neq q$) despite the players' roles being symmetrical.

Remark. Technically, an edge of size 0 is monochromatic of both colors. If $\emptyset \in E(H)$, then a positional game on H is over before it begins, and we naturally declare that: Maker-Maker is a first player win, Maker-Breaker is an Alice (Maker) win, Avoider-Avoider is a second player win, and Avoider-Enforcer is a Bob (Enforcer) win.

Let us start with a brief history of positional games:

- The Maker-Maker convention was the first to be introduced in the literature in all generality. The game of *tic-tac-toe*, which can be traced back all the way to ancient Egypt, is a well-known example of it: the nine cells are the vertices, and the eight lines (three horizontal, three vertical, two diagonal) are the edges. In 1963, Hales and Jewett introduced the "positional game" terminology as they gave the first general formulation of what we now call Maker-Maker games [HJ63], even though they only studied a generalized n^d *tic-tac-toe* with lines of size n and d dimensions (the original tic-tac-toe corresponds to $n = 3$ and $d = 2$). Erdős and Selfridge [ES73] later established the first general results on the Maker-Maker convention, and several results are compiled in [Bec08] and [HKS⁺14]. Another generalization of tic-tac-toe is the game of *k-in-a-row* which is played on the infinite square grid and where the winning sets are the lines of size k . Note that *k-in-a-row* technically fails our definition since the underlying hypergraph is infinite, even though the edges are of bounded size. This game is known to end in a first player win for $k \leq 4$ and a draw for $k \geq 8$ [GSZ80] if both players play optimally, whereas solving the case $5 \leq k \leq 7$ is a famous open problem. Note that 5-in-a-row restricted to a 15×15 board coincides with *gomoku*, another popular Maker-Maker game for which a winning strategy for the first player has been found with the help of computer assistance [AvH96], however it is unclear whether this strategy can be adapted to an infinite board.
- The Maker-Breaker convention owes its first general formulation to Chvátal and Erdős in 1978, who studied games played on the edge set of the complete graph K_n where Maker tries to get some specific subgraph of her color [CE78]. Independently, Schaefer also introduced Maker-Breaker games (among others) to study their algorithmic complexity [Sch78], though the description of the game and the terminology used were different. In 1982, Beck obtained the first general results on the Maker-Breaker convention [Bec82]. A lot of results on Maker-Breaker games are compiled in [Bec08] and [HKS⁺14]. The board game *Hex*, created in 1942, where the players try to connect opposite sides of a board with hexagonal cells, is an interesting example. It fails to classify under the Maker-Maker convention since the winning sets are not the same for both players. However, a graph theory argument shows that every final position of the game contains either a winning red path or a winning blue path and not both, therefore winning equates to blocking the opponent, which makes Hex a Maker-Breaker game. Another example is the Shannon switching game [Gar61][Leh64][CE78], which is played on the edges of a graph with two special vertices u and v , where one player tries to connect u and v while the other attempts to prevent them.
- The Avoider-Avoider convention mostly originates from the work of Harary in 1981, who formulated this convention in a Ramsey context (no possible draw) [Har81] and studied games played on the edge set of the complete graph K_n where the players must avoid getting some specific subgraph of their color [Har82]. Some cases of this problem have been studied very recently [SS23]. The oldest and most famous example of an Avoider-Avoider instance is *Sim*, a game introduced by Simmons in 1968 [Sim68], where two players color the edges of K_6 while trying to avoid creating a triangle of their color. It is remarkable that, despite the apparent innocence of Sim, it is very difficult to describe a winning strategy that a human can understand and apply. It has been known for a long time that the second player has a winning strategy [MRH74], but only recently has a significantly simpler one been found [Wrz20].
- The Avoider-Enforcer convention owes its first general formulation to Lu in 1992 [Lu92], who designated the players as "anti-Maker" and "anti-Breaker" respectively, and studied the game played on the edge set of K_n where the losing sets are the hamiltonian cycles. Other Avoider-Enforcer games played on the edge set of K_n have been studied since, such as the case where the losing sets are the copies of K_k for example [Bec02], which

generalizes Sim in an Avoider-Enforcer variant. Many results on Avoider-Enforcer games are compiled in [Bec08] and [HKS⁺14]. Likely because of its somewhat counterintuitive rules, the Avoider-Enforcer convention, unlike the three previous ones, does not have a simple or notorious historical instance that we know of.

Generally speaking, research on positional games mainly consists in considering some convention as well as some hypergraph class \mathcal{H} and studying the following questions:

- Criteria for the outcome: What is the outcome of the game on $H \in \mathcal{H}$ i.e. the result when both players play optimally? Can we at least state some necessary or sufficient conditions on $H \in \mathcal{H}$ for such player to win (or not lose) with optimal play?
- Description of optimal strategies: Is there an optimal strategy for such player on $H \in \mathcal{H}$ that is intuitive or at least easy to describe? If such player has a winning strategy on $H \in \mathcal{H}$, what is the fastest way to win i.e. what is the minimum number of rounds in which that player can ensure the win?
- Algorithmic complexity: What is the complexity of deciding whether such player has a winning strategy on some input hypergraph $H \in \mathcal{H}$?

I.1.1.3 Achievement vs Avoidance, Strong vs Weak

The four aforementioned conventions fall into two brackets, as defined by Harary [Har82]: Maker-Maker and Maker-Breaker are *achievement games*, whereas Avoider-Avoider and Avoider-Enforcer are *avoidance games*. The last two are the respective *misère* versions of the first two, since the winning condition becomes the losing condition for both players. Edges can be seen as winning sets in achievement games and as losing sets in avoidance games.

Another way to categorize positional games is via the vocabulary introduced by Beck and Csirmaz [BC82]: Maker-Maker and Avoider-Avoider are *strong games*, whereas Maker-Breaker and Avoider-Enforcer are *weak games*. This denomination stems from achievement games seen from Maker's point of view, and the following observation: for Maker, it is not more difficult to face Breaker than to face another Maker, therefore a win against Breaker only represents a "weak win" for Maker. Let us state this fact formally along with its avoidance counterpart. Note that these results hold even in the biased version.

Proposition I.1.9. *Let H be a hypergraph.*

- *If the first (resp. second) player has a winning strategy for the Maker-Maker game on H , then Maker has a winning strategy as first (resp. second) player for the Maker-Breaker game on H . More specifically: any winning strategy of the first (resp. second) player for the Maker-Maker game on H is a winning Maker strategy as first (resp. second) player for the Maker-Breaker game on H .*
- *If the first (resp. second) player has a winning strategy for the Avoider-Avoider game on H , then Enforcer has a winning strategy as first (resp. second) player for the Avoider-Enforcer game on H . More specifically: any winning strategy of the first (resp. second) player for the Avoider-Avoider game on H is a winning Enforcer strategy as first (resp. second) player for the Avoider-Enforcer game on H . ■*

The converses are not true. Let H be the hypergraph of standard 3×3 tic-tac-toe for instance. It is well known that the first player does not have a winning strategy for the Maker-Maker game on H (which is the usual convention here), however it is very easy to see that Maker has a winning strategy as first player for the Maker-Breaker game on H : we will come back to this example in Section II.1. Indeed, the second player's drawing strategy for tic-tac-toe heavily relies on the creation of threats that the first player has to defend. On the contrary, in the Maker-Breaker variant, Breaker aligning three cells becomes irrelevant: the game simply

continues and Maker (as first player) will end up aligning three cells of her own with optimal play. Morally, in weak games, one player (Maker or Enforcer) is relinquished of their defensive duties, which simplifies their task compared to the strong game where a subtle balance between attack and defense must be found.

We have seen that, in all historical instances of positional games, edges represent some substructures inside a bigger structure (e.g. lines in a grid, subgraphs in a graph). Therefore, it is not surprising that the field's pioneers come from extremal graph and hypergraph theory, which studies the existence of such substructures depending on the size of the ambient structure. In particular, a typical question in Ramsey theory [Ram30] is whether every partition of said structure has an element satisfying some property. The possibility of a draw in strong games is one such problem: can the vertex set be partitioned into red and blue vertices so that there is no monochromatic edge? This is the hypergraph (vertex-)2-coloring problem.

Definition I.1.10. Let H be a hypergraph. A *2-coloring* of H is a function that assigns to each vertex of H the color red or blue. A *partial 2-coloring* of H is defined the same except that some vertices might be uncolored. A (partial) 2-coloring c of H is said to be *proper* if no edge of H is monochromatic respective to c , or *improper* otherwise. We say H is *2-colorable* if there exists a proper 2-coloring of H .

In all positional games, the two players build a partial 2-coloring that evolves as they play moves. In strong games specifically, there are two possibilities: that partial 2-coloring either becomes improper at some point, in which case the player responsible for it is then declared the winner in the Maker-Maker game or the loser in the Avoider-Avoider game, or remains proper throughout, in which case a draw is declared. In particular, we have the following result, which holds for any bias:

Proposition I.1.11. *If a hypergraph H is not 2-colorable, then a strong game played on H cannot end in a draw, regardless of the strategies used by both players.* ■

The previous proposition can help build examples of hypergraphs on which strong games cannot end in draws. For example, in a paper on extremal hypergraph theory [EL74], Erdős and Lovász have studied hypergraphs in which any two distinct edges intersect on at most one vertex, building ones with as few edges as possible that have no big stable set (i.e. where any set including at least half the vertices contains an edge). Since such hypergraphs are obviously not 2-colorable, strong games played on them cannot end in draws, so we see how extremal hypergraph theory can yield results in positional game theory. It should still be noted that, in contrast to the graph 2-colorability problem which is easy, hypergraph 2-colorability is an NP-complete problem even when restricted to hypergraphs of rank 3 [Lov73], which limits the potential of Proposition I.1.11.

I.1.1.4 Elementary strategies and principles

Several basic results on positional games can be derived from the ability to adapt a winning strategy from one situation to another. We present a few such strategy constructions and their consequences. For each one, we proceed as follows: we first describe how the strategy Σ' is built from the strategy Σ in all generality and regardless of optimality, then we explain why Σ' is a winning strategy if Σ itself is one. We consider games with a bias $(p : q)$ unless otherwise specified: this includes the unbiased case which corresponds to $p = q = 1$.

a) Strategy stealing

The word "stealing" should be generally understood, not as Alice or Bob stealing the other's strategy (which, by the way, could only make sense in the case $p = q$), but rather as the first player stealing the second player's strategy: any strategy of Alice (resp. Bob) as second player may be replicated by Alice (resp. Bob) as first player, as we now explain. On a given hypergraph, consider some strategy Σ of, say, Alice as second player, and suppose that Alice plays first. Then Alice can play an arbitrary first move X_0 , act as second player from there, and play according to the strategy Σ adapted as follows. Visualize a fictitious game $\mathcal{G}_{\text{fict}}$ next to the real game $\mathcal{G}_{\text{real}}$, where Alice plays second and has not played X_0 , so that Alice has an extra set X_{extra} (initialized to X_0) of red vertices in $\mathcal{G}_{\text{real}}$ compared to $\mathcal{G}_{\text{fict}}$. Whenever Bob plays in $\mathcal{G}_{\text{real}}$, Alice transcribes this move in $\mathcal{G}_{\text{fict}}$, answers according to Σ in $\mathcal{G}_{\text{fict}}$, and copies her move in $\mathcal{G}_{\text{real}}$. Anytime Alice's fictitious move X intersects X_{extra} , which prevents her from copying that exact move in $\mathcal{G}_{\text{real}}$, she plays $(X \setminus X_{\text{extra}}) \cup X'$ instead where X' is arbitrary. Continuing so, Alice ensures that, in $\mathcal{G}_{\text{real}}$ and at all times before it ends, the blue vertices (Bob's) are the same as in $\mathcal{G}_{\text{fict}}$ whereas the red vertices (Alice's) are the same as in $\mathcal{G}_{\text{fict}}$ plus the p extra red vertices of X_{extra} . In the last round of play on $\mathcal{G}_{\text{fict}}$ (where Alice may have to simulate Bob's move in part or in full), if it reaches that point, these extra vertices may be shared between both players, but the final fictitious red vertices will be a subset of the real red vertices. A fundamental consequence of strategy stealing is that playing first is an advantage in achievement games:

Proposition I.1.12 (Initiative Principle). *In achievement games, both players prefer to play first. In other words: if Alice (resp. Bob) has a winning strategy on H as second player, then Alice (resp. Bob) has a winning strategy on H as first player also.*

Proof. Suppose Alice has a winning strategy Σ as second player. Then, as first player, Alice can simply apply the "stolen" strategy described above. Since the red (resp. blue) vertices in $\mathcal{G}_{\text{real}}$ form a superset (resp. a subset) at all times of what they are in $\mathcal{G}_{\text{fict}}$, Alice's win in $\mathcal{G}_{\text{fict}}$ implies a (possibly quicker) win in $\mathcal{G}_{\text{real}}$. ■

Similar arguments also show that, in achievement games, the players benefit from picking more vertices in each round:

Proposition I.1.13 (Bias Monotonicity Principle). *In achievement games, if Alice has a winning strategy on H with bias $(p : q)$ as first (resp. second) player then Alice has a winning strategy on H with bias $(p' : q)$ as first (resp. second) player for all $p' \geq p$. The same holds for Bob when exchanging the roles of p and q .* ■

As a consequence of Proposition I.1.12, a famous proof by contradiction [HJ63] shows that the best result the second player can hope for in a Maker-Maker game is a draw (unless the bias is in the second player's favor), as is the case in optimal tic-tac-toe for example:

Proposition I.1.14. *In Maker-Maker games with a bias $(p : q)$ where $p \geq q$, Bob as second player cannot have a winning strategy, therefore optimal play leads either to a first player win or a draw.*

Proof. We first consider the case $p = q$. Suppose for a contradiction that Bob has a winning strategy Σ as second player. By Proposition I.1.12 (Initiative Principle), Bob would then also have a winning strategy Σ' were he the first player. Since $p = q$, Alice can apply Σ' herself as first player, and thus win against Bob which is a contradiction. Now, the case $p > q$ ensues by

Proposition I.1.13 (Bias Monotonicity Principle). ■

The moral of this is, in achievement games, both players are always happy to make moves: the more vertices of their own color, the better for them. One could think that the reverse holds in avoidance games: the less vertices of your own color, the better? Unfortunately, this is not true. Consider the graph H which is the union of a path P_3 and an isolated vertex u for instance. It is easy to see that Alice (Avoider), as first player, wins both the Avoider-Avoider and the Avoider-Enforcer game on H , by playing u as her first move. However, precolor u in blue (Bob's color), and all of a sudden Alice loses both games as first player. Moreover, it is not difficult to show that avoidance games have no bias monotonicity in general. Nevertheless, some natural results do hold in Avoider-Enforcer games. Notably, in unbiased Avoider-Enforcer games, strategy stealing does work (if the number of vertices is even, otherwise it must be complemented with different arguments) and yields the following:

Proposition I.1.15. *In unbiased Avoider-Enforcer games, both players prefer not to play last, i.e. not to play first if $|V(H)|$ is odd or not to play second if $|V(H)|$ is even. In other words: if Avoider (resp. Enforcer) has a winning strategy on H as last player, then Avoider (resp. Enforcer) has a winning strategy on H as second-to-last player also.* ■

Nothing of the sort can be said about Avoider-Avoider games: no general rule holds stating that it is better to play first, or second, or last, or second-to-last, as simple counter-examples exist in all cases. Therefore, contrary to what strategy stealing implies for Maker-Maker games, all three outcomes are possible for Avoider-Avoider games: a first player win, a second player win, or a draw. Very few proofs using strategy stealing exist in the Avoider-Avoider literature, although some have emerged recently to solve certain positions in a generalization of Sim [Mal20]. Beck declares that a "general open problem is to find the avoidance version of the strategy stealing argument" [Bec02].

Finally, even though our description has the first player choosing their extra vertices arbitrarily throughout, it is possible that strategy stealing with some non-arbitrary way of selecting the extra vertices could be interesting in some cases. However, we are not aware of any such example.

b) Strategy self-stealing

We now present another simple construction, where the idea of the first player FP is to improve on an existing strategy of their own. On a given hypergraph, consider a strategy Σ of FP: if Σ dictates to play X_0 as the first move, then FP can play a different first move X'_0 instead but then replicate Σ in the subsequent moves, as follows. Again, visualize a fictitious game $\mathcal{G}_{\text{fict}}$ next to the real game $\mathcal{G}_{\text{real}}$: as their first move, FP plays X'_0 in $\mathcal{G}_{\text{real}}$ and X_0 in $\mathcal{G}_{\text{fict}}$. Since $|X_0 \setminus X'_0| = |X'_0 \setminus X_0| =: r$, choose numberings $X_0 \setminus X'_0 = \{x_1, \dots, x_r\}$ and $X'_0 \setminus X_0 = \{x'_1, \dots, x'_r\}$: FP is going to update both games in parallel, identifying x_i in $\mathcal{G}_{\text{real}}$ with x'_i in $\mathcal{G}_{\text{fict}}$ for all i . Whenever the second player SP plays in $\mathcal{G}_{\text{real}}$, FP transcribes this move in $\mathcal{G}_{\text{fict}}$, answers according to Σ in $\mathcal{G}_{\text{fict}}$, and copies this last move in $\mathcal{G}_{\text{real}}$. Since the x_i are taken in $\mathcal{G}_{\text{fict}}$ and the x'_i are taken in $\mathcal{G}_{\text{real}}$, FP makes the following adjustments when transcribing the moves: any x_i inside SP's move in $\mathcal{G}_{\text{real}}$ is replaced by x'_i in $\mathcal{G}_{\text{fict}}$, and any x'_i inside FP's move in $\mathcal{G}_{\text{fict}}$ is replaced by x_i in $\mathcal{G}_{\text{real}}$. By construction, at all times and for all i , FP (resp. SP) owns x_i in $\mathcal{G}_{\text{real}}$ if and only if FP (resp. SP) owns x'_i in $\mathcal{G}_{\text{fict}}$. As for the vertices other than $x_1, \dots, x_r, x'_1, \dots, x'_r$, their situation is the same in both games at all times. This construction can be used to show that some moves are better than others, as we now illustrate.

Proposition I.1.16 (Domination Principle). *Let X_0 and X'_0 be two possible first moves for the first player on H . Suppose that there exist numberings $X_0 \setminus X'_0 = \{x_1, \dots, x_r\}$ and $X'_0 \setminus X_0 =$*

$\{x'_1, \dots, x'_r\}$ such that, for all $1 \leq i \leq r$, every edge of H containing x_i also contains x'_i (call this the "domination hypothesis"). Then:

- In achievement games on H , if the first player has a strategy to win (resp. draw) with first move X_0 then the first player has a strategy to win at least as fast (resp. draw) with first move X'_0 .
- In avoidance games on H , if the first player has a strategy to win (resp. draw) with first move X'_0 then the first player has a strategy to win at least as fast (resp. draw) with first move X_0 .

Proof. We consider achievement games (the proof is analogous for avoidance games). Let Σ be a strategy for FP in which X_0 is the first move. Let FP play according to the strategy Σ' constructed from Σ as described above, in which X'_0 is the first move, with numberings satisfying the domination hypothesis.

- Firstly, suppose that, at some point, SP gets an edge e of their color in $\mathcal{G}_{\text{real}}$. Since FP possesses all the x'_i in $\mathcal{G}_{\text{real}}$, we know e contains none of the x'_i . By the domination hypothesis, this implies e contains none of the x_i either. Since all vertices other than $x_1, \dots, x_r, x'_1, \dots, x'_r$ have the same owner in both games by construction of Σ' , all vertices of e also belong to SP in $\mathcal{G}_{\text{fict}}$ at that moment.
- Secondly, suppose that, at some point, FP gets an edge e of their color in $\mathcal{G}_{\text{fict}}$. We know that all vertices of e are in possession of FP in $\mathcal{G}_{\text{real}}$ at the same moment apart maybe from the x_i , which we now address. If $x_i \in e$ for some $1 \leq i \leq r$, then $x'_i \in e$ by the domination hypothesis, therefore FP possesses x'_i in $\mathcal{G}_{\text{fict}}$ which implies that FP possesses x_i in $\mathcal{G}_{\text{real}}$ by construction of Σ' . In conclusion, all vertices of e also belong to FP in $\mathcal{G}_{\text{real}}$ at that moment.

All in all, we have shown that FP cannot win slower or lose faster with Σ' than with Σ . ■

c) Strategy importing

Let H be a subhypergraph of some hypergraph H' . Given a strategy Σ for Alice or Bob, say Alice, on H , there are several ways to construct a strategy Σ' for Alice on H' . A natural one is to have Alice play exclusively inside of H during a first phase of play, as follows. Visualize a fictitious game $\mathcal{G}_{\text{fict}}$ played on H next to the real game $\mathcal{G}_{\text{real}}$ played on H' . Whenever Bob plays a move Y in $\mathcal{G}_{\text{real}}$, Alice transcribes this move in $\mathcal{G}_{\text{fict}}$, replacing it with $(Y \cap V(H)) \cup Y'$ for some arbitrary $Y' \subseteq V(H)$ if $Y \not\subseteq V(H)$ to make it a valid move for Bob in $\mathcal{G}_{\text{fict}}$, then Alice answers according to Σ in $\mathcal{G}_{\text{fict}}$ and copies her move in $\mathcal{G}_{\text{real}}$. Alice continues doing so until all vertices of H are colored in $\mathcal{G}_{\text{fict}}$ (if we reach that point), after which a second phase of play starts where Alice plays arbitrarily until the end. Note that the last move played in $\mathcal{G}_{\text{fict}}$ may not have enough vertices to be valid in $\mathcal{G}_{\text{real}}$, in which case Alice completes it into a full move using arbitrary vertices. This way of importing the strategy Σ ensures that, at all times, the red vertices inside H are the same in both games, whereas the set of blue vertices inside H in $\mathcal{G}_{\text{real}}$ is a subset of what it is in $\mathcal{G}_{\text{fict}}$ since Bob may have picked some vertices outside of the subhypergraph H . In achievement games, if Σ is a winning strategy on H , then Σ' is a winning strategy on H' as long as Alice's local play happens to block Bob globally, which is always the case in Maker-Breaker games (recall that Alice is Maker and Bob is Breaker in this case):

Proposition I.1.17 (Local Win Principle). *Let H be a subhypergraph of some hypergraph H' , and suppose that there exists a strategy Σ on H which is winning for Alice as first (resp. second) player in the Maker-Breaker game (note that, by Proposition I.1.9, this is a weaker assumption than asking the same in the Maker-Maker game). Let Σ' be Alice's strategy on H' as first (resp. second) player imported from Σ as detailed above. Then:*

- For the Maker-Maker game on H' , either Σ' is a winning strategy as fast as Σ is on H , or Σ' is a losing strategy.
- For the Maker-Breaker game on H' , Σ' is a winning strategy as fast as Σ is on H .

Proof. We have mentioned that, by construction, the red vertices inside H are the same in $\mathcal{G}_{\text{real}}$ as in $\mathcal{G}_{\text{fict}}$ at all times. Since Σ is a winning Maker strategy for the Maker-Breaker game on H , Alice is guaranteed to complete a red edge in $\mathcal{G}_{\text{real}}$ during the first phase of play (i.e. the phase where Alice plays exclusively in H), at the same time she does in $\mathcal{G}_{\text{fict}}$. No draw is thus possible and it becomes a question of whether Bob can win faster or not. For the Maker-Breaker game on H' , the fact that Bob cannot prevent the appearance of a red edge in H trivially implies the same in H' since H is a subhypergraph of H' , so he necessarily loses. Finally, to show that Σ' is exactly as fast as Σ in the winning case, rather than being "at least as fast", it suffices to consider the case where Bob's moves in $\mathcal{G}_{\text{real}}$ are all inside H and coincide with the moves that slow down Alice the most in $\mathcal{G}_{\text{fict}}$. ■

If H is a subhypergraph of H' , it can happen that the first player has a winning strategy for the Maker-Maker game on H but has no winning strategy at all for the Maker-Maker game on H' (let alone one obtained via strategy importing). The simplest example may be the following: $V(H) = \{x, y_1, y_2, y_3, z_1, z_2, z_3\}$, $E(H) = \{\{x, y_1, y_2\}, \{x, y_1, y_3\}, \{x, z_1, z_2\}, \{x, z_1, z_3\}\}$, $V(H') = V(H)$, $E(H') = E(H) \cup \{\{y_3, z_3\}\}$. For the unbiased Maker-Maker game, the first player FP has a winning strategy on H , however the second player SP has a drawing strategy on H' . The game on H' could go as follows: FP picks x (optimal move on H), SP picks y_3 , now FP would like to pick z_1 (optimal move on H) however the presence of the extra winning set $\{y_3, z_3\}$ ruins those plans and forces FP to pick z_3 instead, so SP can pick z_1 and clinch a draw. Beck refers to this phenomenon as the "extra set paradox" [Bec08]. This also explains why the winning strategy for gomoku does not easily extend to an infinite grid: though the first player can fix some 15×15 subgrid and emulate this strategy inside the subgrid, it is unclear how to manage the second player's outside threats.

Strategy importing does not work as well in avoidance games, where it is important that the moves made inside the subhypergraph H actually alternate. However, there is a different way to successfully import a strategy as Enforcer against Avoider under certain conditions:

Proposition I.1.18. *Let H be a subhypergraph of some hypergraph H' such that $|V(H') \setminus V(H)|$ is even, and suppose that Enforcer has a winning strategy Σ as first (resp. second) player for the unbiased Avoider-Enforcer game on H . Then Enforcer also has a winning strategy as first (resp. second) player for the unbiased Avoider-Enforcer game on H' .*

Proof. If Enforcer is the first player, then he plays the first move dictated by Σ (which is inside H). After that, Enforcer follows Avoider around, as we now explain. Whenever Avoider plays inside H , Enforcer answers inside H as well according to the winning strategy Σ . Whenever Avoider plays outside H , Enforcer answers outside H as well with an arbitrary move, which is always possible because of the parity assumption. The alternation of moves inside H is thus preserved, so Avoider will complete an edge e of her color inside H . Since e is also an edge of H' , this means Enforcer wins on H' . ■

d) Pairing strategy

Let us start by considering the unbiased case, which is the most natural for pairing strategies.

Definition I.1.19. A *pairing* is a set Π of pairwise disjoint pairs.

Definition I.1.20. Let H be a hypergraph and let Π be a pairing. We say Π is *complete in H* if every edge of H contains some pair from Π . Otherwise, we say Π is *incomplete in H* .

Given some pairing Π , a *pairing strategy associated with Π* is any strategy that satisfies the following rule: if the opponent has just picked some $x \in \{x, y\} \in \Pi$ where y is free, then answer by picking y . The first move (if that player plays first), or any move following a move where the opponent has played outside of Π or played some $x \in \{x, y\} \in \Pi$ where y was already colored, may be selected in any manner. Usually, these moves are selected arbitrarily: each selected move then depends solely on the opponent's last move, whereas general strategies select moves depending on all the moves made since the beginning of the game. Pairing strategies are thus among the simplest ones, and they are easy to apply as a human or to implement computationally using minimum storage space. Using this strategy, a player can ensure to hit every pair from Π : this is Breaker's idea against Maker. In contrast, a player can also ensure that their opponent will hit every pair from Π : this is Avoider's idea against Enforcer. This is sufficient to win on H if Π is complete in H :

Proposition I.1.21 (Pairing Principle). *Let H be a hypergraph, and suppose that there exists a pairing Π which is complete in H . Then Breaker (resp. Avoider), as first or second player, has a winning strategy for the unbiased Maker-Breaker (resp. unbiased Avoider-Enforcer) game on H . More specifically:*

- *Any pairing strategy associated with Π is a winning strategy for Breaker, as first or second player, for the unbiased Maker-Breaker game on H .*
- *There exists a pairing strategy associated with Π that is a winning strategy for Avoider, as last player (hence the existence of a winning strategy as second-to-last player also), for the unbiased Avoider-Enforcer game on H .*

Proof. The result for Breaker is clear: any pairing strategy associated with Π ensures that Breaker hits every pair from Π , and thus hits every edge of H since Π is complete in H , meaning Breaker wins. On the other hand, Avoider must be careful not to get both vertices of some pair. Suppose Avoider is the last player. If Avoider is also the first player (i.e. $|V(H)|$ is odd), then she plays an arbitrary first move outside Π . After that, the number of free vertices outside Π is even. Therefore, whenever Enforcer plays outside Π , Avoider can answer arbitrarily outside Π as well. Of course, whenever Enforcer plays inside Π , he is the first to hit the pair in question, so Avoider can pick the other vertex of that pair. The strategy that we have just described is a pairing strategy associated with Π , moreover this strategy ensures that Enforcer hits every pair from Π , and thus hits every edge of H since Π is complete in H , meaning Avoider wins. By Proposition I.1.15 (strategy stealing), Avoider also has a winning strategy as second-to-last player. ■

The pairing argument is commonly used with complete pairings, however it can also yield results with smaller pairings, even ones with a single pair. The following proposition, which presents some game-neutral operations on hypergraphs, is an example of that:

Proposition I.1.22. *Consider the following operations on hypergraphs:*

- (i) *Adding two vertices x, y and adding edges containing both of them.*
- (ii) *Adding two vertices x, y and, for some edge e or several, replacing e with two edges $e \cup \{x\}$ and $e \cup \{y\}$.*

Operation (i) does not affect the outcome of unbiased strong or weak games. Operation (ii) does not affect the outcome of unbiased weak games.

Proof. Let H be a hypergraph and let H' be the hypergraph obtained from H after performing

operation (i) or (ii). Consider some strategy Σ on H for some player, say Alice (exchange the players' names if it is Bob). Then Alice may use the following strategy Σ' on H' . If Alice is the first player, then she plays the first move dictated by Σ , which is a vertex of H . After that, whenever Bob plays inside H , Alice answers inside H according to Σ . If Bob ever plays x (resp. y), then Alice answers by playing y (resp. x). It can happen that x and y are the last two vertices remaining and it is Alice's turn, in which case she picks one arbitrarily and Bob picks the other. In all cases, Alice ends up getting exactly one vertex in $\{x, y\}$ and Bob gets the other, whereas the other vertices are colored in accordance with Σ . We can now conclude for each operation:

- (i) We consider all games, strong or weak. The edges in $E(H') \setminus E(H)$ contain both x and y by assumption, so none of them end up monochromatic when the player uses the strategy Σ' on H' , meaning the result of the game only depends on the edges of H . Therefore, if Σ is a winning (resp. drawing) strategy on H , then Σ' is a winning (resp. drawing) strategy on H' .
- (ii) We consider weak games. The result of the game is decided by the existence of a monochromatic red edge. It thus suffices to show that, when the player (Alice or Bob) uses the strategy Σ' on H' , the final set R of red vertices in H' contains an edge of H' if and only if it contains an edge of H . If $e \in E(H) \setminus E(H')$ is such that $e \subseteq R$, then $e \cup \{x\} \subseteq R$ or $e \cup \{y\} \subseteq R$ by construction, which concludes since $e \cup \{x\}$ and $e \cup \{y\}$ are edges of H' . If $e \in E(H') \setminus E(H)$ is such that $e \subseteq R$, then $e \setminus \{x\} \subseteq R$ and $e \setminus \{y\} \subseteq R$, which concludes since $e \setminus \{x\}$ or $e \setminus \{y\}$ is an edge of H . ■

It is interesting to note that operation (ii) in the previous proposition is not neutral for strong games. As much as this operation does not alter any player's capacity to make/avoid monochromatic edges, which is all that is needed in weak games, it does change how fast they can manage it since the new edges are bigger, which can make a crucial difference in strong games. For example, take the hypergraph H' from our "extra set paradox" illustration (after Proposition I.1.17): we know the second player has a drawing strategy for the Maker-Maker game on H' , but it is not difficult to show that replacing the edge $\{y_3, z_3\}$ with two edges $\{y_3, z_3, a\}$ and $\{y_3, z_3, b\}$ shifts the game back to being a first player win with optimal play, because the second player's threat is not immediate anymore when they play y_3 or z_3 .

In Avoider-Enforcer games, it seems difficult to define a biased version of pairing strategies. However, in Maker-Breaker games, the pairing principle naturally generalizes to any bias of the form $(1 : q)$ as follows.

Definition I.1.23. A q -pairing is a set Π of pairwise disjoint sets such that $2 \leq |\pi| \leq 1 + q$ for all $\pi \in \Pi$.

Remark. A 1-pairing is simply a pairing. Moreover, a q -pairing is a q' -pairing for all $q' \geq q$.

Definition I.1.24. Let H be a hypergraph and let Π be a q -pairing. We say Π is *complete in H* if, for all $e \in E(H)$, there exists $\pi \in \Pi$ such that $|e \cap \pi| \geq 2$. Otherwise, we say Π is *incomplete in H* .

Given some q -pairing Π , a q -pairing strategy associated with Π is any Breaker strategy that satisfies the following rule: if Maker has just picked some $x \in \pi \in \Pi$, then Breaker's answer must contain all the remaining free vertices of π (which is always possible since $|\pi| \leq 1 + q$). This way, Breaker ensures that Maker gets at most one vertex from each element of Π , which is sufficient to guarantee that Breaker wins on H if Π is complete in H . Therefore:

Proposition I.1.25. Let H be a hypergraph, and suppose that there exists a q -pairing Π which is

complete in H . Then any q -pairing strategy associated with Π is a winning strategy for Breaker, as first or second player, for the Maker-Breaker game on H with bias $(1 : q)$. ■

An alternative point of view on q -pairing strategies, which may also be useful for a general bias $(p : q)$ with $p \geq 2$, will be given in Subsection I.1.2 (page 47).

I.1.1.5 Difficulty comparison between conventions

The previous discussion shows that some natural and convenient properties hold in some conventions but not in others. First of all, achievement games are easier to handle, notably because they have the following properties that avoidance games do not have:

- strategy stealing arguments: "more moves is always better";
- importance of "who plays first" rather than "who plays last" (for example, playing first reduces to playing second up to considering all possibilities of the first move, whereas this argument has no equivalent for playing last or second-to-last);
- bias monotonicity;
- no parity considerations.

Moreover, weak games prove to be more accessible than strong games:

- question of "existence" rather than "first existence";
- no attack and defense going on simultaneously: it is attacker versus defender;
- subhypergraph monotonicity (this is key).

It also makes sense to mention algorithmic complexity here. For such questions, we consider the unbiased case. First of all, we should explain how deciding the outcome comes down to a decision problem (binary output) for each convention. For the Maker-Maker convention, Proposition I.1.14 ensures that there are only two possible outcomes. For the Maker-Breaker and Avoider-Enforcer conventions, there are only two possible outcomes once we establish who plays first (both cases reduce to each other anyway). However, we have seen that the Avoider-Avoider convention has three possible outcomes, so say we consider the question "Does the first player have a winning strategy?", of which it can be shown that it is equivalent to the same question for the second player and hereby equivalent to deciding the outcome.

Now, our observation about the difficulty of weak games as opposed to strong games is comforted by the following result and its proof, which show that Maker-Breaker games can be seen as a subcase of Maker-Maker games (this actually holds for any bias).

Proposition I.1.26. [Bys04] *Deciding the outcome of Maker-Breaker games reduces to deciding the outcome of Maker-Maker games.*

Proof. Let H be a hypergraph. Create two new vertices x and y , and let H' be the hypergraph defined by $V(H') = V(H) \cup \{x, y\}$ and $E(H') = \{e \cup \{x\}, e \in E(H)\} \cup \{\{x, y\}\}$. Consider the Maker-Maker game played on H' . By Proposition I.1.16 (Domination Principle), it is optimal for the first player to start the game by picking x since all edges of H' contain x . Obviously, it is then optimal for the second player to pick y , otherwise the first player would pick y in the next round and win on the spot because $\{x, y\} \in E(H')$. After this optimal first round of play, the situation is as follows. The second player cannot threaten anything anymore, since all edges of H' contain x which is already owned by the first player. As for the first player, completing an edge of H' is now equivalent to completing an edge of H , since the set of edges of H' that have not already been hit by the second player is $\{e \cup \{x\}, e \in E(H)\}$. All in all, the Maker-Maker game on H' becomes the Maker-Breaker game on H (where Maker plays first) after one round of optimal play. Therefore, Maker has a winning strategy as first player for the Maker-Breaker game on H if and only if the first player has a winning strategy for the Maker-Maker game on

H' . ■

Complexity of positional games is studied through discriminating by edge size: the bigger the edges, the more complicated the game becomes. A simple observation can be made on this subject when it comes to weak games:

Proposition I.1.27. *Let k be a fixed integer. For unbiased weak games, deciding the outcome for hypergraphs of rank k reduces to deciding the outcome for k -uniform hypergraphs.*

Proof. Operation (ii) from Proposition I.1.22 allows to replace any edge e with two edges of size $|e| + 1$ without altering the outcome. Therefore, given a hypergraph of rank k , it suffices to perform this operation several times until all edges are of size exactly k . ■

Known results about algorithmic complexity of unbiased positional games prior to our work are summed up in Table I.1. Solving any of the four conventions is PSPACE-complete, even when restricting the input hypergraph to be 7-uniform.

k	Maker-Breaker	Maker-Maker	Avoider-Enforcer	Avoider-Avoider
1	P	P	P	P
2	[folklore]	[folklore]	[folklore][GGP ⁺ 22]	[folklore]
3	unknown	unknown	unknown	PSPACE-complete [FGM ⁺ 15]
4				
5				
6	PSPACE-complete	PSPACE-complete [RW21][Bys04]	PSPACE-complete [GO23]	
7	[RW21]			

Table I.1: Complexity classes of unbiased positional games on k -uniform hypergraphs, for different values of k .

Let us provide a brief history of these results and some specifications regarding Table I.1:

- Some complexity results have long been known for achievement games. In 1978, Schaefer showed PSPACE-completeness for the Maker-Breaker game even when restricted to hypergraphs of rank 11 [Sch78]. Byskov later published a much simpler proof in the general case i.e. with no restriction on the size of the edges [Bys04]. Recently, Rahman and Watson have improved Schaefer's result from 11 to 6 [RW21]. This implies PSPACE-completeness for the Maker-Maker game even when restricted to 7-uniform hypergraphs, using the reduction from the proof of Proposition I.1.26 and then applying Proposition I.1.27. On the other end of the spectrum, tractability is trivial for achievement games when the input H is a graph (i.e. $k = 2$): the first player wins if and only if H contains a path P_3 on three vertices (for the Maker-Breaker convention, we choose Maker as first player here). As for the Maker-Breaker game restricted to 3-uniform hypergraphs, which is flagged as "unknown" here, one of the main results of the present dissertation will be to show that it is also tractable.

- Complexity of avoidance games had been elusive until very recently. In 2023, Gledel and Oijid have shown PSPACE-completeness for the Avoider-Enforcer and Avoider-Avoider games, even when restricted to 6-uniform and 7-uniform hypergraphs respectively [GO23]. An interesting fact about the Avoider-Avoider convention, however, is that it had actually been proved for years to be PSPACE-complete even restricted to graphs, without the positional games community realizing. Indeed, in their founding book on combinatorial games, Berlekamp et al. have introduced a game named Col [BCG82, p.37], where two players alternate turns coloring vertices of a graph G in red and blue respectively, with the constraint that the partial 2-coloring thus built by the players must remain proper throughout. The first player unable to move, either because all free vertices have a neighbor of that player's color or because all vertices of G have been colored already, loses the game. The only fundamental difference with the Avoider-Avoider game is that no draw is possible in Col. However, the outcome of the Avoider-Avoider game does imply the outcome of Col: if the Avoider-Avoider game on G is a draw, then the first player wins Col on G if and only if the first player is also the last player i.e. $|V(G)|$ is odd. Since Col is known to be PSPACE-complete [FGM⁺15][BH19], so is the Avoider-Avoider game on graphs, hence the rightmost column of Table I.1. This is a striking difference with achievement games where the case of graphs is trivial. The Avoider-Enforcer game on graphs is nontrivial either, however there exists a simple proof of its tractability [GGP⁺22].

In conclusion, Maker-Breaker is the most accessible of the four conventions, since it possesses a lot of convenient properties from being both an achievement game and a weak game, whereas Avoider-Avoider is the most difficult without doubt.

I.1.2 State of the art on the Maker-Breaker game

From now on, we will exclusively consider the Maker-Breaker convention, and we will always assume that Maker plays first (the case where Breaker plays first reduces to that case up to considering all possibilities of the first move). In the game with bias $(p : q)$, p corresponds to Maker (Alice, red) and q corresponds to Breaker (Bob, blue).

I.1.2.1 General results

We have just seen several elementary principles on positional games in general. As memorandum, let us first compile some of them that apply to the Maker-Breaker convention:

Proposition I.1.28. *The following properties hold for the Maker-Breaker game on any hypergraph H :*

- *Bias monotonicity: If Maker has a winning strategy on H with bias $(p : q)$, then she also has a winning strategy on H with bias $(p' : q')$ for any $p' \geq p$ and $q' \leq q$. If Breaker has a winning strategy on H with bias $(p : q)$, then he also has a winning strategy on H with bias $(p' : q')$ for any $p' \leq p$ and $q' \geq q$.*
- *Subhypergraph monotonicity: If Maker has a winning strategy on some subhypergraph of H with bias $(p : q)$, then Maker also has a winning strategy on H with bias $(p : q)$.*
- *If H admits a complete q -pairing, then Breaker has a winning strategy on H with bias $(1 : q)$.*
- *If H is not 2-colorable, then Maker has a winning strategy on H with bias $(p : q)$ for any $p \geq q$.*

Proof. We should briefly discuss the last item, which was not exactly stated as such previously.

On the one hand, since $p \geq q$, Bob cannot win the Maker-Maker game on H as second player by Proposition I.1.14. On the other hand, since H is not 2-colorable, no draw is possible in the Maker-Maker game on H by Proposition I.1.11. Putting the two together, we see that Alice wins the Maker-Maker game on H as first player, therefore she (Maker) also wins the Maker-Breaker game on H by Proposition I.1.9. ■

Proposition I.1.29. [Sch78][RW21] *Deciding the outcome of the unbiased Maker-Breaker game is a PSPACE-complete problem, even when restricted to 6-uniform hypergraphs.* ■

Aside from these few properties, and despite Maker-Breaker being the most favorable convention to study as we have mentioned, very few results are known that hold in all generality. There are basically two, which give a sufficient condition for Breaker and Maker respectively to have a winning strategy. Historically, the former is due to Erdős and Selfridge [ES73]. They stated it as a sufficient condition for the second player to have a drawing strategy in the Maker-Maker game, as the Maker-Breaker convention had not yet been introduced. Since their strategy consists in blocking the first player, it actually constitutes a winning strategy for Breaker in the Maker-Breaker game. Beck realized this and published the Maker-Breaker version of the Erdős-Selfridge theorem, which he generalized to the biased game as follows:

Theorem I.1.30 (Erdős-Selfridge Theorem). [Bec82] *A sufficient condition for Breaker to have a winning strategy for the Maker-Breaker game with bias $(p : q)$ on a hypergraph H is:*

$$\sum_{e \in E(H)} (1 + q)^{-\frac{|e|}{p}} < \frac{1}{1 + q}.$$

For a k -uniform hypergraph H , this condition can be rewritten as:

$$|E(H)| < (1 + q)^{\frac{k}{p} - 1}.$$

In the same paper, Beck also gave a sufficient condition for Maker to have a winning strategy:

Theorem I.1.31. [Bec82] *A sufficient condition for Maker to have a winning strategy for the Maker-Breaker game with bias $(p : q)$ on a hypergraph H is:*

$$\sum_{e \in E(H)} \left(1 + \frac{q}{p}\right)^{-|e|} > \frac{p^2 q^2}{(p + q)^3} \Delta_2(H) |V(H)|,$$

where $\Delta_2(H)$ denotes the maximum number of edges of H containing x and y over all pairs $\{x, y\} \subseteq V(H)$. For a k -uniform hypergraph H , this condition can be rewritten as:

$$|E(H)| > \frac{(p + q)^{k-3} q^2}{p^{k-2}} \Delta_2(H) |V(H)|.$$

The proofs of these results are rather short and both use the same technique. At any point during the game, each edge e is assigned a "potential": the closer Maker is from completing the edge e , the higher the potential. If e contains a blue vertex, then its potential is obviously 0, otherwise its potential is a well-chosen decreasing function of its number of uncolored vertices. Each vertex x is also assigned a potential, equal to the sum of the potentials of the edges incident to x . The strategy, be it for Breaker (Theorem I.1.30) or for Maker (Theorem I.1.31), then simply consists in always picking vertices of maximum potential. In the assumptions of both theorems, the inequality is conveniently chosen so that this strategy is indeed winning, as shown after some calculations.

Since general criteria are difficult to obtain, most studies focus on some particular type of Maker-Breaker game i.e. on a specific hypergraph class. Let us mention some of the main ones.

I.1.2.2 Games played on the edge set or vertex set of a graph

The hypergraph representation is not always the most intuitive one for a positional game. Indeed, a lot of games have a natural description that does not use hypergraphs explicitly, although they do categorize as positional games through some underlying hypergraph. For example, tic-tac-toe and its generalizations have a geometrical representation by essence: the game board is a grid, and the winning sets are lines.

a) Games played on the edge set of a complete graph

An important part of the Maker-Breaker literature revolves around instances where the game board is the complete graph K_n in which the players color the edges alternately, and Maker wants the subgraph formed by her edges to satisfy some increasing graph property \mathcal{P} (which may depend on n). These are game adaptations of problems studied notably by the Hungarian school of extremal graph theory, which ask the question of how many edges are needed to ensure some property \mathcal{P} . The underlying hypergraph $H_{\mathcal{P}}^{(n)}$ may then be defined as follows: $V(H_{\mathcal{P}}^{(n)}) = E(K_n)$, and $E(H_{\mathcal{P}}^{(n)})$ is the set of all minimal subgraphs of K_n satisfying property \mathcal{P} . This corresponds to the study of the hypergraph class $\mathcal{H}_{\mathcal{P}} = \{H_{\mathcal{P}}^{(n)}, n \geq 1\}$. The founding paper on Maker-Breaker games by Chvátal and Erdős introduces three such examples:

- Connectivity game [CE78]: \mathcal{P} is "contain a spanning subtree";
- Hamiltonian cycle game [CE78]: \mathcal{P} is "contain a hamiltonian cycle";
- Clique game [CE78]: \mathcal{P} is "contain a clique of size k " for some fixed k .

Other properties have been studied since then, including the following:

- H -game [BL00]: \mathcal{P} is "contain a copy of H " for some fixed graph H that does not depend on n (the case $H = K_k$ corresponds to the clique game);
- Minimum-degree game [KS08]: \mathcal{P} is "have minimum degree at least d " for some fixed d ;
- k -Connectivity game [KS08]: \mathcal{P} is "contain a k -vertex-connected spanning subgraph";
- Non-planarity game [HKS⁺08]: \mathcal{P} is "be non-planar";
- Non- k -colorability game [HKS⁺08]: \mathcal{P} is "be non- k -colorable" for some fixed k ;
- Perfect matching game [HKS⁺09]: \mathcal{P} is "contain a perfect matching";

For almost all of the games above, it is easy to show that Maker has a winning strategy in the unbiased case. For this reason, these games are studied with a $(1 : q)$ bias to make them more interesting. The bias monotonicity ensures that, provided all edges are of size at least 2 (failing which Maker trivially wins whatever the bias since we assume she plays first), there exists a value of q under which Maker has a winning strategy and over which Breaker has a winning strategy:

Definition I.1.32. Let H be a hypergraph such that $\text{ark}(H) \geq 2$. The *threshold bias* of H , denoted by $q_{\text{thr}}(H)$, is defined as the smallest integer $q \geq 1$ such that Breaker has a winning strategy for the Maker-Breaker game on H with bias $(1 : q)$.

Determining the value (or, at least, the behavior asymptotically in n) of $q_{\text{thr}}(H_{\mathcal{P}}^{(n)})$ is the central question. Several bounds on the threshold bias for the games listed above are featured in [HKS⁺14]. The *probabilistic intuition* is a heuristic argument introduced by Chvátal and Erdős [CE78] to predict the asymptotic behavior of the threshold bias: it expresses the idea that the threshold value of q where the winner of the game switches might be the same with optimal play as it would be with random play. For games played on the edge set of K_n with bias $(1 : q)$, Maker gets a total of $N(n, q) := \left\lceil \frac{\binom{n}{2}}{1+q} \right\rceil$ edges of K_n (assuming all edges end up colored). Therefore, the probabilistic intuition suggests that the threshold bias is approximately the value of q where

the probability that the Erdős-Rényi random graph $G(n, N(n, q))$ has property \mathcal{P} switches from $1 - o(1)$ to $o(1)$. This phenomenon is not very well understood, as it holds for numerous natural instances but also fails for some. Indeed, the probabilistic intuition is correct for most of the aforementioned games, but it is incorrect for the triangle game (clique game with $k = 3$).

Finally, for values of the bias where Maker has a winning strategy, the number of rounds in which Maker can ensure to win has also been studied extensively in the literature. Again, many such results are compiled in [HKS⁺14].

b) Games played on the edge set or vertex set of a general graph

Instead of introducing a bias for the games played on the edge set of K_n , another way to give Breaker a fair chance is to replace K_n with a sparse graph G . For instance, a lot of results are known about games played on an Erdős-Rényi random graph $G(n, p)$ [NSS16][CFK⁺12][HKS⁺14]. The case of a general (non-random) graph G has not been studied as much, but it has gathered interest very recently: Duchêne et al. [DGM⁺23] have obtained a structural characterization of the outcome for the P_4 -game on any graph G and for the star-game when G is a tree (for any fixed star $K_{1,t}$), yielding linear-time algorithms to solve both. They also show that deciding the outcome of the connectivity game on general graphs can be done in polynomial time, whereas the same problem is NP-complete for the perfect matching game and the H -game for some choices of H .

Contrary to complete graphs, in which all vertex subsets of a given size are the same, it makes sense in general graphs to also consider Maker-Breaker games where the players pick vertices of G rather than edges. Let us mention a few of them:

- Domination game [DGP⁺20]: the winning sets are the dominating sets of G i.e. the subsets $U \subseteq V(G)$ such that every vertex of G either is in U or has a neighbor in U . It is shown that the outcome can be decided in polynomial time on trees and cographs, whereas this problem is NP-complete on bipartite graphs and split graphs. Fast-winning Maker strategies have also been studied [GIK19].
- Total domination game [GHI⁺20]: the winning sets are the total dominating sets of G i.e. the subsets $U \subseteq V(G)$ such that every vertex of G has a neighbor in U . The game is solved on grids and cacti, whereas deciding the outcome is an NP-complete problem on bipartite graphs and split graphs.
- Resolving game [KKY20]: the winning sets are the resolving sets of G i.e. the subsets $U \subseteq V(G)$ such that, for any distinct $u, v \in V(G)$, there exists $w \in U$ satisfying $\text{dist}_G(u, w) \neq \text{dist}_G(v, w)$. On several graph classes, including trees and grids, the game is solved and fastest-winning Maker strategies are obtained.
- Largest connected subgraph game [BFM⁺23]: the winning sets are the subsets $U \subseteq V(G)$ such that $G[U]$ has a connected component of size at least k , for some fixed k . It is shown that the outcome can be decided in polynomial time on cographs, whereas this problem is NP-complete on bipartite graphs of diameter 4, split graphs and planar graphs. Surprisingly, the case of trees remains open.

I.1.2.3 Our case of interest: hypergraphs of small rank

We have just presented a variety of Maker-Breaker games that, truly, are played on a game board which is a graph even though they can also be represented by a hypergraph. Their study is much helped by the graph structures in action. What about Maker-Breaker games on a general hypergraph H , with no geometrical or graph-related structure behind it? Hypergraphs of small rank are a natural place to start with: the smaller the edges, the more structure there is, so the easier it should be to get results.

First of all, the case of hypergraphs containing the empty edge is trivial: Maker wins before the game even begins. The case where there exists a singleton edge $e = \{x\}$ is also straightforward: since Maker plays first, she can pick x and win immediately, whatever the bias. Next, we should consider 2-uniform hypergraphs i.e. graphs.

a) Solution of the game on graphs

Note that this is very different from the games played on graphs that we have presented previously: this time, the winning sets are the actual edges of the graph, which is much easier than the case where the winning sets are, say, the dominating sets of the graph. The solution is simple:

Theorem I.1.33. *Consider the Maker-Breaker game with bias $(p : q)$ on a graph G . If at least one of the following conditions is satisfied, then Breaker has a winning strategy:*

(i) $E(G) = \emptyset$.

(ii) $p = 1$ and $q \geq \Delta(G)$, where $\Delta(G)$ denotes the maximum degree of G .

Otherwise, Maker has a winning strategy which wins in just two rounds if $p = 1$ or one round if $p \geq 2$.

Proof. Obviously, Breaker wins if $E(G) = \emptyset$, so assume $E(G) \neq \emptyset$. If $p \geq 2$ then Maker wins on her first move by picking both vertices of some edge, so also assume $p = 1$. If $q < \Delta(G)$, then Maker wins in two rounds: she can pick a vertex x of maximum degree, Breaker cannot pick all the neighbors of x at once since $q < \Delta(G)$, therefore Maker can pick one of them herself as her second move and win. If $q \geq \Delta(G)$ then Breaker can always pick all the neighbors of the vertex that Maker has just picked (completing his move with arbitrary vertices if needed), which obviously is a winning strategy since it prevents Maker from ever getting both vertices of any edge. ■

Let us mention that a study of *scoring positional games* has been initiated recently by Bagan et al. [BDD⁺22]. For the Maker-Breaker convention, the score is defined as the number of winning sets that Maker completes. For the Maker-Maker convention, the score is defined as the number of winning sets that the first player completes minus the number of winning sets that the second player completes. In all cases, the game continues until all vertices are colored, and the first (resp. second) player aims at maximizing (resp. minimizing) the score. The authors study the scoring game on graphs with no bias, and they obtain a surprising result: in the scoring version, in contrast with standard positional games, the strong game is easier than the weak game. Indeed, the problem of determining whether the score with optimal play is at least k is PSPACE-complete for Maker-Breaker but tractable for Maker-Maker. However, the Maker-Breaker score with optimal play can be computed in polynomial time for paths and cycles.

b) The game on hypergraphs of rank 3

Back to the standard Maker-Breaker game, the next case to consider should be hypergraphs of rank 3, starting with the unbiased version. Things become much more complicated, and very few publications have tackled the subject prior to our work. There exist mainly two, which both address restricted subcases simplifying the hypergraph structure: Kutz added the constraint that any two distinct edges must intersect on at most one vertex [Kut04], whereas Rahman and Watson (actually studying a more general game) added the constraint that every edge must contain a vertex of degree 1 [RW20]. Both papers give a structural characterization of the outcome, from which they derive a polynomial-time algorithm to solve the game under these

respective constraints. We will go back to these two results in detail at the beginning of Section II.1, in which we will prove one of the main results of this dissertation, namely a structural characterization of the outcome for general hypergraphs of rank 3.

c) An alternative view on q -pairing strategies

We conclude with a remark on q -pairing strategies. It starts from a straightforward observation, which holds for any bias $(p : q)$:

Notation I.1.34. Let H and H' be hypergraphs. We write $H' \leq H$ if $V(H) = V(H')$ and, for all $e \in E(H)$, there exists $e' \in E(H')$ such that $e' \subseteq e$.

Proposition I.1.35. *Let H and H' be hypergraphs such that $H' \leq H$. Then any winning strategy for Maker on H is also winning on H' , and any winning strategy for Breaker on H' is also winning on H .* ■

The fact that $H' \leq H$ means H' can be obtained from H by shrinking edges. Such an operation is not neutral in general (it can only benefit Maker, as per Proposition I.1.35) but it simplifies the hypergraph, since it reduces the size of some edges and edge intersections. Given a hypergraph H , an idea for Breaker could then be to: virtually shrink some edges, evaluate the outcome on the resulting simpler hypergraph H' , and if he has a winning strategy on H' then follow the same strategy on H to ensure a win.

In particular, since the game is solved on graphs, an idea is to turn H into a graph G by shrinking each edge into an edge of size 2. Note that this is only useful in the case $p = 1$, otherwise all graphs are losing for Breaker. We get:

Proposition I.1.36. *Let H be a hypergraph. If there exists a graph $G \leq H$ such that $\Delta(G) \leq q$, then Breaker has a winning strategy for the Maker-Breaker game on H with bias $(1 : q)$.*

Proof. Theorem I.1.33 ensures that Breaker wins on G , so Breaker wins on H by Proposition I.1.35. ■

It turns out q -pairing strategies are a particular case of this principle:

Proposition I.1.37. *A hypergraph H admits a complete q -pairing if and only if there exists a graph $G \leq H$ in which all connected components are of size at most $1 + q$.*

Proof. Let \mathcal{F} be the set of all functions $f : E(H) \rightarrow \mathbb{P}_2(V(H))$ such that $f(e) \subseteq e$ for all $e \in E(H)$. Then, the existence of a complete q -pairing in H can be formulated as follows: there exists a q -pairing Π , and there exists $f \in \mathcal{F}$, such that: for all $e \in E(H)$, there exists $\pi \in \Pi$ satisfying $f(e) \subseteq \pi$. Since \mathcal{F} describes all the ways to shrink H into a graph, swapping the first two existential quantifiers yields a formulation of the existence of a graph $G \leq H$ whose edges can be put in bags of size at most $q + 1$ (the bags being the elements of the q -pairing). ■

A q -pairing strategy for Breaker can thus be understood as follows: virtually shrink H into a graph G in which all connected components are of size at most $1 + q$, and, each time Maker picks a vertex x , pick all free vertices in the connected component of x . Proposition I.1.36 is stronger, as it recognizes that picking all free neighbors of x suffices.

Moreover, whereas q -pairing strategies have no natural extension to the case $p \geq 2$, Proposition I.1.35 is valid for any bias. If $p = 2$ for instance, then graphs are useless since Maker always wins, but an idea could be to shrink H into a 3-uniform hypergraph H' . However, unlike the

outcome on graphs, the outcome on general 3-uniform hypergraphs with a bias $(2 : q)$ is not known.

I.1.3 Our approach to the Maker-Breaker game and a few basic results

Up to this point, we have maintained somewhat of an external point of view and kept any formalism to a minimum. For our own upcoming study of Maker-Breaker games, we now need to make choices in terms of definitions, notations, and overall approach. Some notions are new, while some are reformulations of standard concepts adapted to our approach.

I.1.3.1 Playing the game on marked hypergraphs

To justify our assumption that Maker plays first, we have said that we can always reduce to that case anyway up to considering all possibilities of Breaker's first move if he plays first. This actually calls for an explanation, as the game is supposed to start on a hypergraph that is not precolored. The important point is that the precolored case reduces to the non-precolored case. Indeed, instead of coloring the vertices in red and blue, another way of updating the hypergraph throughout the moves is the following (e.g. [Kut04]). When Maker picks a vertex x , that vertex is removed from the hypergraph and any edge e containing x is replaced by $e \setminus \{x\}$. When Breaker picks a vertex y , that vertex is removed from the hypergraph and any edge e that contained y is deleted. That way, the updated edges represent what remains to be done for Maker to complete the original edges, and Maker wins if and only if some edge gets shrunk all the way to size 0. Note that the same principle works for the Avoider-Enforcer convention when replacing Maker with Avoider and Breaker with Enforcer, but not for strong games, in which precolored positions really are a case in their own right (another reason why weak games are easier to handle).

Our point of view is in between, as we adopt the above way of updating the hypergraph for Breaker but not for Maker. Rather than the "shrink and delete" update, we use "mark and delete":

- Maker *marks* vertices. This means that vertices picked by Maker are not removed from the hypergraph, instead they are marked so as to signal that Maker owns them and that they are not playable anymore. This is fundamentally the same as coloring them in red, however there will only be one color involved, so it is really just a marking of the vertices.
- Breaker *deletes* vertices. This means that vertices picked by Breaker are removed from the hypergraph as well as all edges containing any of them.

Therefore, we consider the game as played on marked hypergraphs, which are a superclass of hypergraphs:

Definition I.1.38. A *marked hypergraph* H is defined by:

- a finite *vertex set* $V(H)$;
- an *edge set* $E(H)$ consisting of subsets of $V(H)$;
- a set of *marked vertices* $M(H) \subseteq V(H)$.

A marked hypergraph H with $M(H) = \emptyset$ may be referred to as a *non-marked hypergraph*.

Definition I.1.39. Let H be a marked hypergraph. The *non-marked reduction* of H is the hypergraph H' defined by $V(H') = V(H) \setminus M(H)$ and $E(H') = \{e \setminus M(H), e \in E(H)\}$.

Because of the obvious correspondence between standard hypergraphs and non-marked hyper-

graphs, all notions that we are about to define for marked hypergraphs apply to hypergraphs as well. Conversely, we extend all definitions and notations given for hypergraphs at the very beginning of this chapter to marked hypergraphs, by applying them to the underlying hypergraph (ignoring the marked vertices), apart from the notion of subhypergraph which is defined as follows:

Definition I.1.40. Let H be a marked hypergraph. A *subhypergraph* of H is a marked hypergraph K such that: $V(K) \subseteq V(H)$, $E(K) \subseteq E(H)$ and $M(K) = V(K) \cap M(H)$. We may say H *contains* K , or write $K \subseteq H$, to signify that K is a subhypergraph of H .

We can now give a formal definition for the Maker-Breaker game played on a marked hypergraph, which clearly coincides with the usual game in the non-marked case.

Definition I.1.41. Let H be a marked hypergraph. Note that some vertices may be marked already before the game starts. The *Maker-Breaker* $(p : q)$ -*game on* H , or simply $(p : q)$ -*game on* H , is defined as follows. In each round, Maker selects p non-marked vertices and marks them, then Breaker selects q non-marked vertices and deletes them. During the last round, if some player cannot play a full move because there are not enough non-marked vertices left, then that player simply marks (resp. deletes) all the remaining non-marked vertices. Maker wins if and only if some edge has all its vertices marked at some point, otherwise Breaker wins.

Remark. It is technically true that an empty edge has all its vertices marked, so if $\emptyset \in E(H)$ then Maker automatically wins before the game even begins.

Example. An example of the $(1 : 1)$ -game on the "tic-tac-toe hypergraph" is given in Figure I.1 (with optimal play, actually): here, we see that Maker wins by completing the middle row.

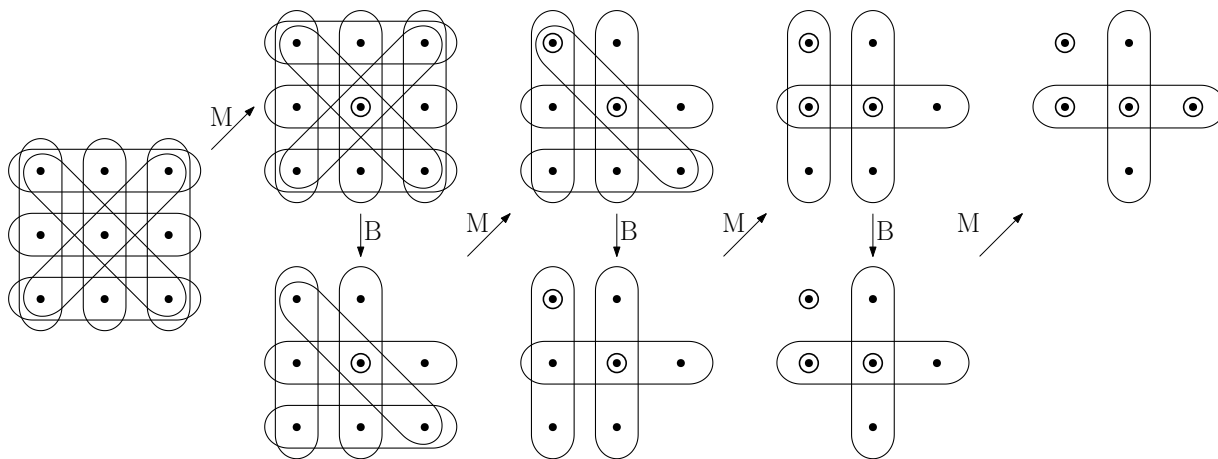


Figure I.1: Evolution of the marked hypergraph during a game. The marked vertices are circled, as they will be in all figures, and should not be mistaken for edges of size 1 (this ambiguity will not exist in subsequent figures, where edges will be represented differently).

The "shrink and delete" update would correspond to playing on the non-marked reductions. Instead, playing on marked hypergraphs has some advantages. For one, it prevents loss of information as to past moves of Maker: marking a vertex is a reversible operation, whereas the original hypergraph cannot be reconstructed after shrinking edges. In particular, edge intersections are preserved throughout the game (for edges that Breaker has not hit, which are the only relevant edges to look at anyway), which will be useful later. Moreover, it preserves

uniformity, meaning a k -uniform marked hypergraph remains k -uniform when updated during the game: this will be very useful to limit cases in our structural studies. Conversely, non-uniform hypergraphs can be made uniform by adding marked vertices to all edges of size smaller than the rank. Now, let us introduce some useful notations:

Notation I.1.42. Let H be a marked hypergraph, and let $X, Y \subseteq V(H) \setminus M(H)$.

- We denote by H^{+X} the marked hypergraph obtained from H by marking the vertices in X , i.e.: $V(H^{+X}) = V(H)$, $E(H^{+X}) = E(H)$, $M(H^{+X}) = M(H) \cup X$.
By convention, if $K \subseteq H$ does not entirely contain X then we define $K^{+X} = K^{+(X \cap V(K))}$.
- We denote by H^{-Y} the marked hypergraph obtained from H by deleting the vertices in Y , i.e.: $V(H^{-Y}) = V(H) \setminus Y$, $E(H^{-Y}) = \{e \in E(H), Y \cap e = \emptyset\}$, $M(H^{-Y}) = M(H)$.
By convention, if $K \subseteq H$ does not entirely contain Y then we define $K^{-Y} = K^{-(Y \cap V(K))}$.
- We may combine these notations, as in $H^{+X-Y} = (H^{+X})^{-Y} = (H^{-Y})^{+X}$ if X and Y are disjoint for instance.

If $X = \{x\}$ is a singleton then we may write H^{+x} instead of H^{+X} , and the same goes for all the above notations.

Remark. It should be noted that H^{-Y} is a subhypergraph of H , while H^{+X} is not because of the additional marked vertices.

When considering the Maker-Breaker game, Notation I.1.42 can be used to update the marked hypergraph during the game. Indeed, the operators $^{+X}$ and $^{-Y}$ signify the effect of Maker picking X and Breaker picking Y respectively, so that after a round of play on a marked hypergraph H where Maker marks X and Breaker deletes Y , it is as though a new game starts on the marked hypergraph H^{+X-Y} . Using these notations, we can give a recursive definition for the outcome of the Maker-Breaker game.

First of all, we identify a trivial case, which corresponds to situations where Maker either has already won (i.e. $e \subseteq M(H)$ for some edge e) or can win with her first move by picking all remaining non-marked vertices of some edge (i.e. $1 \leq |e \setminus M(H)| \leq p$ for some edge e):

Definition I.1.43. Let H be a marked hypergraph. We say H is a *trivial* $(p : q)$ -Maker win if some edge $e \in E(H)$ satisfies $|e \setminus M(H)| \leq p$. In the unbiased case $p = q = 1$, we simply say *trivial Maker win*.

Remark. Note that the definition of a trivial $(p : q)$ -Maker win actually does not depend on q , as Breaker does not get to play in this case.

We define the outcome as follows, depending on the number of non-marked vertices. If there is less than a full round to play (base case), then Maker wins if and only if we are in the trivial case of Definition I.1.43. If there is at least one full round to play (recursive case), then Maker wins if and only if she can ensure that she wins after the first round i.e.: there exists a first move of Maker such that, for any first move of Breaker, Maker wins on the resulting marked hypergraph. Let us write this formally:

Definition I.1.44. Let H be a marked hypergraph. The fact that H is a $(p : q)$ -Maker win is defined recursively as follows:

- (1) If $|V(H) \setminus M(H)| < p + q$, then H is a $(p : q)$ -Maker win if and only if H is a trivial $(p : q)$ -Maker win.
- (2) If $|V(H) \setminus M(H)| \geq p + q$, then H is a $(p : q)$ -Maker win if and only if there exists $X \in \mathbb{P}_p(V(H) \setminus M(H))$ such that, for all $Y \in \mathbb{P}_q(V(H^{+X}) \setminus M(H^{+X}))$, H^{+X-Y} is a $(p : q)$ -Maker win.

Otherwise, we say H is a $(p : q)$ -Breaker win. In the unbiased case $p = q = 1$, we simply say *Maker win* or *Breaker win*.

Remark. In a way, this definition assumes that the game continues even if Maker has already won, until all vertices are taken. Indeed, H might be a trivial $(p : q)$ -Maker win even in the recursive case.

I.1.3.2 Optimizing time or tokens

When Maker has a winning strategy for the $(p : q)$ -game on H , she may challenge herself by trying to "optimize" her win in some way. We now present two (marked) hypergraph parameters that correspond to two different optimization goals. We have already mentioned the first one before: it is about time optimization i.e. winning in as few rounds as possible (this time, of course, we consider that the game stops when Maker gets a fully marked edge). The following notation is introduced in [HKS⁺14], and we adapt it to marked hypergraphs.

Notation I.1.45. Let H be a marked hypergraph. We define $\tau_M^{(p:q)}(H)$ as the minimum number of rounds in which Maker can guarantee to get a fully marked edge when playing the $(p : q)$ -game on H , with $\tau_M^{(p:q)}(H) = \infty$ by convention if H is a $(p : q)$ -Breaker win.

Equivalently, $\tau_M^{(p:q)}(H)$ may be defined recursively as follows:

- (0) If H is a trivial $(p : q)$ -Maker win, then define $\tau_M^{(p:q)}(H) = 0$ if H has a fully marked edge or $\tau_M^{(p:q)}(H) = 1$ otherwise.
- (1) If H is not a trivial Maker win and $|V(H) \setminus M(H)| < p + q$, then define $\tau_M^{(p:q)}(H) = \infty$.
- (2) If H is not a trivial Maker win and $|V(H) \setminus M(H)| \geq p + q$, then define

$$\tau_M^{(p:q)}(H) = 1 + \min_{X \in \mathbb{P}_p(V(H) \setminus M(H))} \max_{Y \in \mathbb{P}_q(V(H+X) \setminus M(H+X))} \tau_M^{(p:q)}(H+X-Y).$$

In the unbiased case $p = q = 1$, we may simply write $\tau_M(H)$.

Proposition I.1.46. *A marked hypergraph H is a trivial $(p : q)$ -Maker win if and only if $\tau_M^{(p:q)}(H) \leq 1$. ■*

Studying fast-winning strategies makes sense in a lot of two-player games. On the contrary, we now introduce our second parameter, which is very specific to positional games and has not appeared in the literature before to our knowledge. Suppose that Maker, additionally to her normal moves and at will during the game, is allowed to unmark vertices that she had previously marked. Since this action does not benefit Maker, the outcome is obviously unchanged, however Maker can now try and win with as few simultaneous marked vertices as possible:

Notation I.1.47. Let H be a marked hypergraph. Consider the $(p : q)$ -game played on H with the added rule that Maker, on her turn and before making her normal move, can unmark any number of vertices that she had previously marked. We define $\theta_M^{(p:q)}(H)$ as the minimum N such that Maker has a winning strategy ensuring that the updated marked hypergraph has at most $|M(H)| + N$ marked vertices throughout, with $\theta_M^{(p:q)}(H) = \infty$ by convention if H is a $(p : q)$ -Breaker win.

In the unbiased case $p = q = 1$, we may simply write $\theta_M(H)$.

This can be seen as a version of the Maker-Breaker game played with tokens. Suppose that Breaker deletes vertices as usual but that Maker, instead of marking vertices, places tokens on them. To place a token on a vertex, Maker may either move a token that was already placed and put it somewhere else, or use a new token. Maker wins if and only if, at any point during

the game, all the vertices of some edge either are marked (meaning they were marked before the game began) or have tokens on them. Then, $\theta_M^{(p:q)}(H)$ is understood as the minimum number of tokens that Maker needs in order to win the game with bias $(p : q)$. This interpretation in terms of tokens is the main one we will use when considering this parameter. Although we do not study it here, it also seems interesting to consider a version of the game where Maker and Breaker both have a fixed number of tokens at their disposal, say α and β respectively: $\theta_M^{(p:q)}(H)$ would then equal the minimum α so that Maker has a winning strategy on H when $\beta = \infty$. Such "token positional games" do have some history. In the Roman Empire, people reportedly played a version of 3×3 tic-tac-toe called *Terni Lapilli* [Zas82], where each player had three tokens that they would place in the first three rounds and move around afterwards (until one player would align their three tokens, which would never happen with optimal play). However, a token could only be moved to a square adjacent to it either orthogonally or along one of the two main diagonals, like a wheel graph. Kraitichik and Gardner mention that the same game without the adjacency restriction exists in France, under the name *Les Pendus* [Kra42][Gar59]. The game of *Nine Men's Morris*, which likely dates back to the Roman Empire as well [Wal14], has a similar nature but is a more distant cousin since aligning three tokens simply allows to remove one of the opponent's tokens instead of winning the game on the spot. Back to our two optimization parameters, it is important to note that they are linked together. Indeed, we have the following inequalities:

Proposition I.1.48. *Let H be a $(p : q)$ -Maker win. Then:*

$$\frac{\theta_M^{(p:q)}(H)}{p} \leq \tau_M^{(p:q)}(H) \leq \left\lceil \frac{|V(H) \setminus M(H)|}{p+q} \right\rceil.$$

Moreover, if H is non-marked then we also have $\text{ark}(H) \leq \theta_M^{(p:q)}(H)$.

Proof. Since $p + q$ non-marked vertices are picked in every round (except possibly for the very last round), we obviously have $\tau_M^{(p:q)}(H) \leq \left\lceil \frac{|V(H) \setminus M(H)|}{p+q} \right\rceil$. Moreover, after $\tau_M^{(p:q)}(H)$ rounds of play, Maker has picked at most $p\tau_M^{(p:q)}(H)$ vertices in total, hence why $p\tau_M^{(p:q)}(H)$ tokens are sufficient for Maker i.e. $\theta_M^{(p:q)}(H) \leq p\tau_M^{(p:q)}(H)$. Finally, if H is non-marked then Maker needs at least as many tokens as the size of a smallest edge in order to fully cover an edge with tokens, hence the final assertion. ■

In the unbiased non-marked case, we get a simplified statement:

Corollary I.1.49. *Let H be a non-marked Maker win. Then:*

$$\text{ark}(H) \leq \theta_M(H) \leq \tau_M(H) \leq \left\lceil \frac{|V(H)|}{2} \right\rceil. \quad \blacksquare$$

Remark. For any $k \geq 1$ and any $n \geq 2k - 1$, there exists a k -uniform non-marked hypergraph H on n vertices such that $k = \theta_M(H) = \tau_M(H)$, thus achieving the lower bound from Corollary I.1.49 for both θ_M and τ_M . Indeed, simply consider the complete k -uniform hypergraph on n vertices i.e. $E(H) = \mathbb{P}_k(V(H))$. Maker can play arbitrary moves, and her first k picks will necessarily form an edge.

We will further investigate these bounds in Section III.2.

I.1.3.3 A reminder on two key principles

Let us insist on two very important principles mentioned at the beginning of this section, which we now re-state in terms of the definition and notations that we have just introduced. These results give a sufficient condition for Maker and Breaker to win respectively. The former will be absolutely crucial in our study. It states that winning on a marked hypergraph cannot necessitate more rounds or tokens than winning on any of its subhypergraphs:

Lemma I.1.50 (Monotonicity Lemma). *Let H be a marked hypergraph, and let K be a subhypergraph of H . If K is a $(p : q)$ -Maker win, then H is a $(p : q)$ -Maker win also. More precisely, we have $\tau_M^{(p:q)}(H) \leq \tau_M^{(p:q)}(K)$ and $\theta_M^{(p:q)}(H) \leq \theta_M^{(p:q)}(K)$.*

Proof. Assume K is a $(p : q)$ -Maker win, otherwise there is nothing to show. Let Σ_τ be a fastest-winning Maker strategy for the $(p : q)$ -game on K . We use strategy importing and Proposition I.1.17 (Local Win Principle): the strategy Σ'_τ imported from Σ_τ on H wins as fast as Σ_τ does on K , so $\tau_M^{(p:q)}(H) \leq \tau_M^{(p:q)}(K)$. The idea is the same when it comes to tokens: by taking a winning Maker strategy Σ_θ on K that uses a minimum number of tokens and importing it on H , we get a winning Maker strategy on H which plays the same moves as Σ_θ until the win and therefore uses the same number of tokens. ■

The latter is about pairing strategies for Breaker in the unbiased case, which we adapt to marked hypergraphs:

Definition I.1.51. Let H be a marked hypergraph and let Π be a pairing.

- Let $e \in E(H)$. We say Π covers e in H if there exists $\pi \in \Pi$ such that $\pi \subseteq e \setminus M(H)$.
- We say Π is complete in H if Π covers e in H for all $e \in E(H)$, otherwise we say Π is incomplete in H .

Proposition I.1.52. *Let H be a marked hypergraph. If H admits a complete pairing, then H is a Breaker win.*

Proof. This is simply the marked counterpart of the assertion on Maker-Breaker games from Proposition I.1.21 (Pairing Principle). ■

I.1.3.4 Some operations on marked hypergraphs

We should mention two elementary game-neutral operations on marked hypergraphs:

Proposition I.1.53. *The following operations on marked hypergraphs do not affect the outcome of the Maker-Breaker game for any bias, and they even preserve both $\tau_M^{(p:q)}$ and $\theta_M^{(p:q)}$:*

- (i) Adding a new vertex m which is marked and, for some edge e or several, replacing e with $e \cup \{m\}$.
- (ii) Deleting a marked vertex m and replacing each edge e that contained it with an edge $e \setminus \{m\}$.

Proof. It suffices to address operation (ii), since operation (i) is its reverse. Let H be a marked hypergraph, and let H' be obtained from H by deleting some $m \in M(H)$ and replacing each edge e that contained m with an edge $e \setminus \{m\}$. Since marked vertices are not playable, the array of available moves is the exact same in H as in H' . Moreover, given the same sequence of moves played on H and H' , Maker has completed an edge on H if and only if she has completed an edge on H' . Therefore, the result is straightforward. ■

Playing on a marked hypergraph is thus equivalent to playing on its non-marked reduction, even when it comes to optimizing time or tokens:

Corollary I.1.54. *Replacing a marked hypergraph by its non-marked reduction does not affect the outcome of the Maker-Breaker game for any bias, and it even preserves both $\tau_M^{(p:q)}$ and $\theta_M^{(p:q)}$.*

Proof. This is just iterating operation (ii) from Proposition I.1.53 until there are no more marked vertices. ■

A consequence is that it is algorithmically equivalent to consider hypergraphs that are marked or non-marked, uniform or non-uniform:

Proposition I.1.55. *Let $k \geq 1$ be an integer. The problems of deciding the outcome of the unbiased Maker-Breaker game for the following four classes all reduce to each other:*

- (1) *hypergraphs of rank k ;*
- (2) *k -uniform hypergraphs;*
- (3) *marked hypergraphs of rank k ;*
- (4) *k -uniform marked hypergraphs.*

Proof. Proposition I.1.27 gives the reduction from (1) to (2). Obviously, (2) reduces to (3), being a subclass. Iterating operation (i) from Proposition I.1.53 reduces (3) to (4). Finally, Corollary I.1.54 reduces (4) to (1). ■

Note that the "uniformization" reduction is not as innocent for non-marked hypergraphs ((1) to (2)) as it is for marked hypergraphs ((3) to (4)), since it alters the hypergraph structure and edge intersections in particular. Moreover, uniform hypergraphs are not as handy as they look, because they cannot be made stable under Maker's moves: they become either non-uniform (with the "shrinking" update) or marked (with our "marking" update). For both these reasons, we usually prefer considering the class of k -uniform marked hypergraphs.

We conclude this section with a construction that will prove useful in Chapter III:

Proposition I.1.56. *Let $k \geq 1$ be an integer. For every k -uniform non-marked Maker win H , there exists a $(k + 1)$ -uniform non-marked Maker win H' such that: $|V(H')| = |V(H)| + 2$, $\tau_M(H') = \tau_M(H) + 1$, $\theta_M(H') = \theta_M(H) + 1$.*

Proof. First note that $|V(H)| \geq 2k - 1$: indeed, Maker winning means she picks at least k vertices in total since $M(H) = \emptyset$, so Breaker picks at least $k - 1$ vertices in the meantime. Let H' be defined by: $V(H') = V(H) \cup \{v, \bar{v}\}$ where v and \bar{v} are new vertices, and $E(H') = E_1 \cup E_2$ where $E_1 := \{e \cup \{v\}, e \in E(H)\}$ and $E_2 := \{\{v, \bar{v}\} \cup U, U \subseteq V(H), |U| = k - 1\}$. We have $|V(H')| = |V(H)| + 2 \geq 2k + 1$, so Maker will get to pick at least $k + 1$ vertices in total.

Claim 1. *Be it in terms of number of rounds or number of tokens, an optimal first round of play on H' is Maker picking v and Breaker picking \bar{v} .*

Proof of Claim 1. Maker has to pick v , otherwise Breaker can pick v himself and win since all edges of H' contain v . Now, if Breaker does not answer by picking \bar{v} , then Maker can pick \bar{v} herself in the second round and play arbitrarily in the $k - 1$ following rounds: the definition of E_2 guarantees that Maker wins, having only used $k + 1$ rounds and $k + 1$ tokens which is best for her since H' is a $(k + 1)$ -uniform hypergraph. □

Claim 1 yields $\tau_M(H') = 1 + \tau_M((H')^{+v-\bar{v}})$ and $\theta_M(H') = 1 + \theta_M((H')^{+v-\bar{v}})$, where the latter equality comes from the fact that Maker's token on v is immobilized eternally since all edges

of H' contain v . To conclude, it thus suffices to show that $\tau_M((H')^{+v-\bar{v}}) = \tau_M(H)$ and $\theta_M((H')^{+v-\bar{v}}) = \theta_M(H)$. This is given by Corollary I.1.54: indeed, we have $V((H')^{+v-\bar{v}}) = V(H) \cup \{v\}$, $E((H')^{+v-\bar{v}}) = E_1$ and $M((H')^{+v-\bar{v}}) = \{v\}$, so the definition of E_1 ensures that the non-marked reduction of $(H')^{+v-\bar{v}}$ is none other than H . ■

I.2 Subhypergraph collections and their intersection properties

Any potential move X of Maker possibly comes with urgent threats to Breaker, in the form of subhypergraphs where Maker would win if Breaker failed to answer the menace immediately. Our approach to the Maker-Breaker game is centered around the following question: given some family \mathcal{F} of identified *danger* types, is there, for any possible first move X of Maker, a Breaker answer Y that hits all associated dangers of type \mathcal{F} ? Therefore, we are interested in collections of subhypergraphs and whether they intersect in some sense. In this section, we introduce hypergraph notions that address such questions, and then we apply them to the Maker-Breaker game to express the aforementioned idea of Maker's danger creation and Breaker's ability to answer it.

I.2.1 Some notions on marked hypergraphs

I.2.1.1 Intersections and unions of subhypergraph collections

Let us start with the union, whose definition is standard.

Definition I.2.1. Let $\mathcal{K} = \{K_1, \dots, K_s\}$ be a finite collection of marked hypergraphs. The *union* of \mathcal{K} , denoted by $\langle \mathcal{K} \rangle$, is the marked hypergraph defined by: $V(\langle \mathcal{K} \rangle) = \bigcup_{K \in \mathcal{K}} V(K)$, $E(\langle \mathcal{K} \rangle) = \bigcup_{K \in \mathcal{K}} E(K)$ and $M(\langle \mathcal{K} \rangle) = \bigcup_{K \in \mathcal{K}} M(K)$. We may also use the notation $\langle \mathcal{K} \rangle = K_1 \cup \dots \cup K_s$.

Remark. It is possible for two elements of \mathcal{K} to share a vertex that is marked in one and non-marked in the other, in which case that vertex is marked in the union. However, this will not happen in practice, since we will always consider collections whose elements are all subhypergraphs of some common marked hypergraph.

When it comes to intersections, we only want to consider non-marked vertices since they are the only playable ones in the Maker-Breaker game, hence the following definition.

Notation I.2.2. Let \mathcal{K} be a collection of marked hypergraphs, and let Y be a set. We define $\mathcal{K} - Y := \{K \in \mathcal{K}, V(K) \cap Y = \emptyset\}$.

Definition I.2.3. Let $q \geq 1$ be an integer. Let \mathcal{K} be a collection of marked hypergraphs and let H be a marked hypergraph.

- A q -*transversal* of \mathcal{K} in H is a set $Y \in \mathbb{P}_q(V(H) \setminus M(H))$ such that $\mathcal{K} - Y = \emptyset$ i.e. $Y \cap V(K) \neq \emptyset$ for all $K \in \mathcal{K}$.
- The q -*intersection* of \mathcal{K} in H is the set $I_H^{(q)}(\mathcal{K})$ of all q -transversals of \mathcal{K} in H . In the case $q = 1$, we simply call it the *intersection* of \mathcal{K} in H and we write $I_H^{(1)}(\mathcal{K}) = I_H(\mathcal{K})$, which

we see as a set of vertices rather than a set of singletons to alleviate notations.

- We say \mathcal{K} is q -intersecting in H , or simply *intersecting in H* in the case $q = 1$, if $I_H^{(q)}(\mathcal{K}) \neq \emptyset$.

Remark. For $\mathcal{K} = \emptyset$, we have $I_H^{(q)}(\emptyset) = \mathbb{P}_q(V(H) \setminus M(H))$. For $\mathcal{K} \neq \emptyset$ and $q = 1$, note that we simply have $I_H(\mathcal{K}) = (\bigcap_{K \in \mathcal{K}} V(K)) \setminus M(H)$.

Proposition I.2.4. *Let $q \geq 1$ be an integer. Let \mathcal{K} and \mathcal{K}' be collections of marked hypergraphs, and let H be a marked hypergraph. If $\mathcal{K} \subseteq \mathcal{K}'$, then $I_H^{(q)}(\mathcal{K}') \subseteq I_H^{(q)}(\mathcal{K})$. ■*

We can state a straightforward characterization of q -intersecting collections in terms of what we call q -obstructions:

Definition I.2.5. Let $q \geq 1$ be an integer. Let \mathcal{K} be a collection of marked hypergraphs and let H be a marked hypergraph. A q -obstruction of \mathcal{K} in H is a subcollection $\mathcal{O} \subseteq \mathcal{K}$ such that $I_H^{(q)}(\mathcal{O}) = \emptyset$. The set of all q -obstructions of \mathcal{K} in H is denoted by $\mathfrak{O}_H^{(q)}(\mathcal{K})$. In the case $q = 1$, we simply call this an *obstruction of \mathcal{K} in H* and we write $\mathfrak{O}_H^{(1)}(\mathcal{K}) = \mathfrak{O}_H(\mathcal{K})$.

Proposition I.2.6. *Let $q \geq 1$ be an integer. Let \mathcal{K} be a collection of marked hypergraphs and let H be a marked hypergraph. Then $I_H^{(q)}(\mathcal{K}) = \emptyset$ if and only if $\mathfrak{O}_H^{(q)}(\mathcal{K}) \neq \emptyset$.*

Proof. If $I_H^{(q)}(\mathcal{K}) = \emptyset$, then $\mathcal{K} \in \mathfrak{O}_H^{(q)}(\mathcal{K})$ hence $\mathfrak{O}_H^{(q)}(\mathcal{K}) \neq \emptyset$. Conversely, if \mathcal{K} is q -intersecting in H , then so are all of its subcollections by Proposition I.2.4 hence $\mathfrak{O}_H^{(q)}(\mathcal{K}) = \emptyset$. ■

If a collection is not q -intersecting, when is it possible to make it q -intersecting by removing q' non-marked vertices, given some q' (not necessarily related to q)? This question can also be answered in terms of q -obstructions. By Proposition I.2.6, the fact that a collection \mathcal{K} is not q -intersecting in H equates to \mathcal{K} admitting q -obstructions in H . We now show that \mathcal{K} can be made q -intersecting in H , by removing q' non-marked vertices, if and only if the unions of its q -obstructions in H form a q' -intersecting collection in H . More precisely:

Proposition I.2.7. *Let $q, q' \geq 1$ be integers. Let \mathcal{K} be a finite collection of marked hypergraphs and let H be a marked hypergraph. Let $Y \in \mathbb{P}_{q'}(V(H) \setminus M(H))$. Then $I_H^{(q)}(\mathcal{K} - Y) \neq \emptyset$ if and only if $Y \in I_H^{(q')}(\{\langle \mathcal{O} \rangle, \mathcal{O} \in \mathfrak{O}_H^{(q)}(\mathcal{K})\})$.*

Proof. If $I_H^{(q)}(\mathcal{K} - Y) = \emptyset$, then define $\mathcal{O} := \mathcal{K} - Y \subseteq \mathcal{K}$: we have $\mathcal{O} \in \mathfrak{O}_H^{(q)}(\mathcal{K})$ and $Y \cap V(\langle \mathcal{O} \rangle) = \emptyset$, so $Y \notin I_H^{(q')}(\{\langle \mathcal{O} \rangle, \mathcal{O} \in \mathfrak{O}_H^{(q)}(\mathcal{K})\})$. Conversely, if $Y \notin I_H^{(q')}(\{\langle \mathcal{O} \rangle, \mathcal{O} \in \mathfrak{O}_H^{(q)}(\mathcal{K})\})$, then let $\mathcal{O} \in \mathfrak{O}_H^{(q)}(\mathcal{K})$ such that $Y \cap V(\langle \mathcal{O} \rangle) = \emptyset$: we have $\mathcal{O} \subseteq \mathcal{K} - Y$, so $I_H^{(q)}(\mathcal{K} - Y) \subseteq I_H^{(q)}(\mathcal{O}) = \emptyset$. ■

Remark. The fact that the collection \mathcal{K} is finite ensures that its q -obstructions also are, so that their unions are well defined. In practice, as mentioned before, we will only consider collections whose elements are all subhypergraphs of some common marked hypergraph, and such collections are obviously finite.

Example. Figure I.2 illustrates Proposition I.2.7 on an example, in the case $q = q' = 1$. The edges inside hypergraphs are not represented as they are irrelevant to intersections. We consider the collection $\mathcal{K} = \{H_1, H_2, H_3, H_4\}$ (left of Figure I.2) where H_1, H_2, H_3 and H_4 are subhypergraphs of some common marked hypergraph H (not drawn).

- On the one hand, we can see that \mathcal{K} is not intersecting in H , but there are four non-marked vertices y such that $\mathcal{K} - y$ is intersecting in H : those are y_1, y_2, y_3 and y_4 . For instance,

$I_H(\mathcal{K} - y_1) = I_H(\{H_3, H_4\}) = \{y_3\} \neq \emptyset$. Note that, if y_5 denotes the bottom-right vertex, we have $I_H(\mathcal{K} - y_5) = I_H(\{H_1, H_4\}) = \emptyset$ because the only common vertex of H_1 and H_4 is marked, hence why y_5 is not on the list.

- On the other hand, there are six obstructions of \mathcal{K} in H : $\mathcal{O}_1 = \{H_1, H_2, H_3, H_4\} = \mathcal{K}$, $\mathcal{O}_2 = \{H_1, H_4\}$, $\mathcal{O}_3 = \{H_1, H_2, H_3\}$, $\mathcal{O}_4 = \{H_1, H_2, H_4\}$, $\mathcal{O}_5 = \{H_1, H_3, H_4\}$, $\mathcal{O}_6 = \{H_2, H_3, H_4\}$. Their respective unions, which are represented on the right of Figure I.2, have exactly four common non-marked vertices: y_1, y_2, y_3 and y_4 . In other words, we have $I_H(\{\langle \mathcal{O} \rangle, \mathcal{O} \in \mathfrak{O}_H(\mathcal{K})\}) = \{y_1, y_2, y_3, y_4\}$, in accordance with Proposition I.2.7.

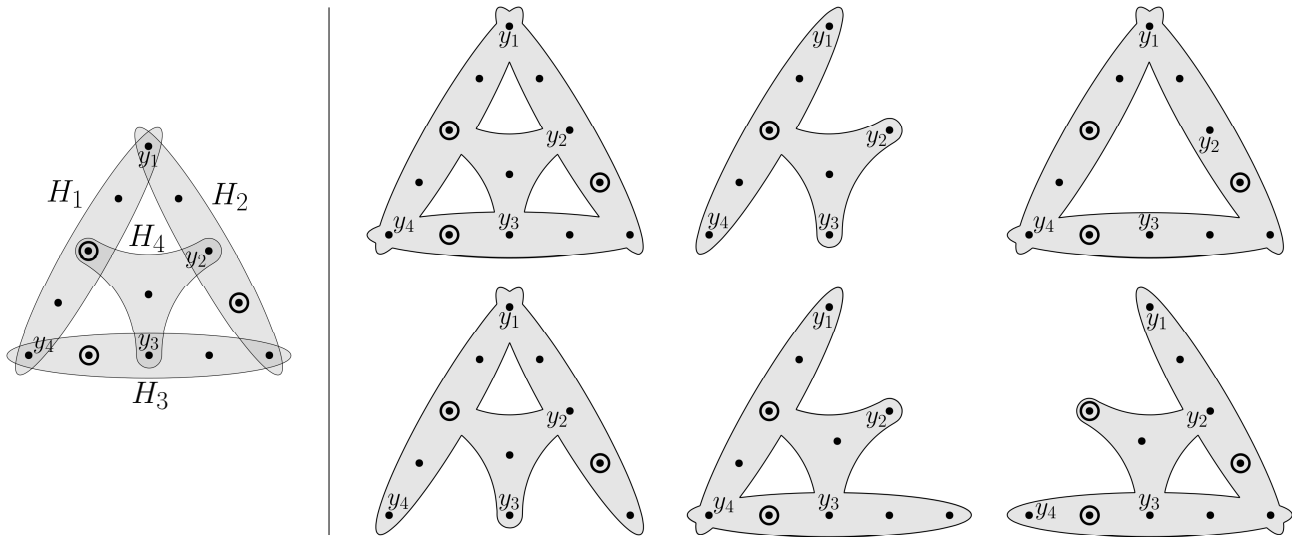


Figure I.2: Left: definition of H_1, H_2, H_3, H_4 . Right (top row then bottom row, from left to right): $\langle \mathcal{O}_1 \rangle, \langle \mathcal{O}_2 \rangle, \langle \mathcal{O}_3 \rangle, \langle \mathcal{O}_4 \rangle, \langle \mathcal{O}_5 \rangle, \langle \mathcal{O}_6 \rangle$.

I.2.1.2 Pointed marked hypergraphs

Definition I.2.8. A *pointed marked hypergraph* is a pair (H, X) where H is a marked hypergraph and $X \subseteq V(H) \setminus M(H)$ is nonempty. When $X = \{x\}$ is a singleton, the pair is usually written as (H, x) .

Definition I.2.9. We say two pointed marked hypergraphs (H, X) and (H', X') are *isomorphic*, and we write $(H, X) \sim (H', X')$, if there exists a bijection $\varphi : V(H) \rightarrow V(H')$ such that:

- For all $e \subseteq V(H)$: $e \in E(H) \iff \varphi(e) \in E(H')$.
- For all $v \in V(H)$: $v \in M(H) \iff \varphi(v) \in M(H')$.
- $\varphi(X) = X'$.

Notation I.2.10. Let \mathcal{F} be a family of pointed marked hypergraphs. Let H be a marked hypergraph, and let $X \subseteq V(H) \setminus M(H)$ be nonempty. We denote by $X\mathcal{F}(H)$ the collection of all subhypergraphs K of H such that $X \subseteq V(K)$ and (K, X) is isomorphic to an element of \mathcal{F} .

I.2.2 Dangers in the Maker-Breaker game

Back to the Maker-Breaker game, we adopt Breaker's point of view. The idea is to consider strategies for Breaker that consist, on each turn, in focusing solely on some identified immediate threats and playing a move that eliminates all these specific threats (if possible). Let $(p : q)$ be

any fixed bias.

I.2.2.1 Definitions and first results

Definition I.2.11. A *danger* is a pointed marked hypergraph (D, X) where $|X| = p$ such that D^{+X} is a $(p : q)$ -Maker win.

Remark. Even though this definition depends on the bias $(p : q)$, we prefer to avoid a more rigorous but heavier denomination such as " $(p : q)$ -danger".

Definition I.2.12. Let H be a marked hypergraph and $X \in \mathbb{P}_p(V(H) \setminus M(H))$. A *danger at X in H* is a subhypergraph D of H containing X such that (D, X) is a danger.

Dangers at X constitute urgent threats for Breaker in the case Maker plays X . Indeed, if Maker plays X then any danger D at X must be immediately destroyed i.e. Breaker must play some Y such that $Y \cap V(D) \neq \emptyset$ next, otherwise the resulting marked hypergraph would contain D^{+X} and thus be a $(p : q)$ -Maker win according to Monotonicity Lemma I.1.50. Therefore, if \mathcal{K}_X is any collection of dangers at X in H and Maker plays X , then Breaker is forced to "destroy" all elements of \mathcal{K}_X i.e. answer with some Y in the q -intersection of \mathcal{K}_X in H^{+X} (the reason why the q -intersection is taken in H^{+X} is because the vertices in X are no longer playable for Breaker after Maker has played X). We thus introduce the following key property, which is necessary for Breaker to win:

Notation I.2.13. Let H be a marked hypergraph such that $|V(H) \setminus M(H)| \geq p + q$. For all $X \in \mathbb{P}_p(V(H) \setminus M(H))$, let \mathcal{K}_X be a collection of dangers at X in H . We denote by $J^{(p:q)}((\mathcal{K}_X)_X, H)$ the following property:

$$\forall X \in \mathbb{P}_p(V(H) \setminus M(H)), I_{H^{+X}}^{(q)}(\mathcal{K}_X) \neq \emptyset.$$

Remark. Dangers are not relevant when there is less than one full round of play left, hence the assumption that $|V(H) \setminus M(H)| \geq p + q$. This also avoids some dull cases where the property would fail on a technicality: indeed, if $\mathcal{K}_X = \emptyset$ then $I_{H^{+X}}^{(q)}(\mathcal{K}_X) = \mathbb{P}_q(V(H^{+X}) \setminus M(H^{+X}))$, which is nonempty if and only if $|V(H) \setminus M(H)| \geq p + q$.

Proposition I.2.14. Let H be a marked hypergraph such that $|V(H) \setminus M(H)| \geq p + q$. For all $X \in \mathbb{P}_p(V(H) \setminus M(H))$, let \mathcal{K}_X be a collection of dangers at X in H . Then, for all $X \in \mathbb{P}_p(V(H) \setminus M(H))$ and for all $Y \in \mathbb{P}_q(V(H^{+X}) \setminus M(H^{+X}))$ such that $Y \notin I_{H^{+X}}^{(q)}(\mathcal{K}_X)$, H^{+X-Y} is a $(p : q)$ -Maker win.

Proof. Since $Y \notin I_{H^{+X}}^{(q)}(\mathcal{K}_X)$, there exists $D \in \mathcal{K}_X$ such that $Y \cap V(D) = \emptyset$. By definition of a danger at X , D^{+X} is a $(p : q)$ -Maker win, and it is a subhypergraph of H^{+X-Y} because $Y \cap V(D) = \emptyset$. Therefore, H^{+X-Y} is a $(p : q)$ -Maker win by Monotonicity Lemma I.1.50. ■

Corollary I.2.15. Let H be a marked hypergraph such that $|V(H) \setminus M(H)| \geq p + q$. For all $X \in \mathbb{P}_p(V(H) \setminus M(H))$, let \mathcal{K}_X be a collection of dangers at X in H . If H is a $(p : q)$ -Breaker win, then $J^{(p:q)}((\mathcal{K}_X)_X, H)$ holds.

Proof. suppose $J^{(p:q)}((\mathcal{K}_X)_X, H)$ does not hold. Maker can then play some X such that $I_{H^{+X}}^{(q)}(\mathcal{K}_X) = \emptyset$, so that Breaker's answer Y cannot be in $I_{H^{+X}}^{(q)}(\mathcal{K}_X)$, thus ensuring that H^{+X-Y} is a $(p : q)$ -Maker win by Proposition I.2.14. Therefore, H is a $(p : q)$ -Maker win. ■

When considering the collection of all dangers at each X , this condition is also sufficient:

Theorem I.2.16. *Let H be a marked hypergraph such that $|V(H) \setminus M(H)| \geq p + q$. For all $X \in \mathbb{P}_p(V(H) \setminus M(H))$, let \mathcal{K}_X be the collection of all dangers at X in H . Then H is a $(p : q)$ -Breaker win if and only if $J^{(p:q)}((\mathcal{K}_X)_X, H)$ holds.*

Proof. The "only if" direction is given by Corollary I.2.15, so we show the "if" direction. Suppose $J^{(p:q)}((\mathcal{K}_X)_X, H)$ holds. Maker plays some $X \in \mathbb{P}_p(V(H) \setminus M(H))$, and Breaker answers with some $Y \in I_{H+X}^{(q)}(\mathcal{K}_X)$. Since $Y \cap V(H^{-Y}) = \emptyset$, we have $H^{-Y} \notin \mathcal{K}_X$ i.e. H^{-Y} is not a danger at X in H . By definition, this means $(H^{-Y})^{+X} = H^{+X-Y}$ is a $(p : q)$ -Breaker win, so H is a $(p : q)$ -Breaker win. ■

I.2.2.2 Considering a fixed family of dangers

Theorem I.2.16 is unlikely to be useful from an algorithmic point of view, since identifying general dangers at a given X is as difficult as identifying $(p : q)$ -Maker wins. We would like the same equivalence to hold for smaller collections \mathcal{K}_X so that property $J^{(p:q)}(\cdot, \cdot)$ is easier to check. A natural idea is to consider dangers at X of the same type for all X , belonging to some fixed family of dangers \mathcal{F} that would be independent of X and easy to recognize:

Definition I.2.17. Let \mathcal{F} be a family of dangers. An element of \mathcal{F} may be referred to as an \mathcal{F} -danger. If H is a marked hypergraph and $X \in \mathbb{P}_p(V(H) \setminus M(H))$, then an element of the collection $X\mathcal{F}(H)$ (recall Notation I.2.10) is called an \mathcal{F} -danger at X in H .

For any family of dangers \mathcal{F} , Breaker needs the ability to destroy all \mathcal{F} -dangers at whatever X that Maker plays on her first move, according to Corollary I.2.15. Actually, this remains true for all subsequent rounds, hence the following notation and necessary condition for a Breaker win:

Notation I.2.18. Let \mathcal{F} be a family of dangers. Let $r \geq 1$ be an integer, and let H be a marked hypergraph such that $|V(H) \setminus M(H)| \geq r(p + q)$. We recursively define the following properties:

- Property $J_1^{(p:q)}(\mathcal{F}, H)$ refers to property $J^{(p:q)}((X\mathcal{F}(H))_X, H)$ i.e.:

$$\forall X \in \mathbb{P}_p(V(H) \setminus M(H)), I_{H+X}^{(q)}(X\mathcal{F}(H)) \neq \emptyset.$$

- Property $J_r^{(p:q)}(\mathcal{F}, H)$, for $r \geq 2$, means that:

$$\forall X \in \mathbb{P}_p(V(H) \setminus M(H)), \exists Y \in I_{H+X}^{(q)}(X\mathcal{F}(H)) \text{ such that } J_{r-1}^{(p:q)}(\mathcal{F}, H^{+X-Y}) \text{ holds.}$$

In the unbiased case $p = q = 1$, we will simply write $J_r(\mathcal{F}, H)$ instead of $J_r^{(p:q)}(\mathcal{F}, H)$. For any $r \geq 1$, property $J_r^{(p:q)}(\mathcal{F}, H)$ should be understood as: "in each of the first r rounds of the Maker-Breaker game played on H , Breaker will be able to destroy all \mathcal{F} -dangers at the subset that Maker has just played".

Proposition I.2.19. *Let \mathcal{F} be a family of dangers. Let $r \geq 1$ be an integer, and let H be a marked hypergraph such that $|V(H) \setminus M(H)| \geq r(p + q)$. If H is a $(p : q)$ -Breaker win, then $J_r^{(p:q)}(\mathcal{F}, H)$ holds.*

Proof. We proceed by induction on r . For $r = 1$, this is simply Corollary I.2.15 with $\mathcal{K}_X = X\mathcal{F}(H)$. Now let $r \geq 2$ such that property $J_{r-1}^{(p:q)}(\mathcal{F}, \cdot)$ is necessary for Breaker to win. Let $X \in \mathbb{P}_p(V(H) \setminus M(H))$: the condition $Y \in I_{H+X}^{(q)}(X\mathcal{F}(H))$ is necessary by Proposition I.2.14, and the condition that $J_{r-1}^{(p:q)}(\mathcal{F}, H^{+X-Y})$ holds is necessary by the induction hypothesis, which concludes. ■

We can make some observations:

Proposition I.2.20. *Let \mathcal{F} be a family of dangers. Let $r \geq 1$ be an integer, and let H be a marked hypergraph such that $|V(H) \setminus M(H)| \geq r(p+q)$.*

- (i) *For any integer $1 \leq s \leq r$: $J_r^{(p:q)}(\mathcal{F}, H) \implies J_s^{(p:q)}(\mathcal{F}, H)$.*
- (ii) *For any family of dangers $\mathcal{G} \subseteq \mathcal{F}$: $J_r^{(p:q)}(\mathcal{F}, H) \implies J_r^{(p:q)}(\mathcal{G}, H)$.*
- (iii) *For any subhypergraph $K \subseteq H$ such that $|V(K) \setminus M(K)| \geq r(p+q)$: $J_r^{(p:q)}(\mathcal{F}, H) \implies J_r^{(p:q)}(\mathcal{F}, K)$.*

Proof. Item (i) is straightforward. Item (ii) comes from the fact that, for all $X \in \mathbb{P}_p(V(H) \setminus M(H))$, we have $X\mathcal{G}(H) \subseteq X\mathcal{F}(H)$ hence $I_{H+X}^{(q)}(X\mathcal{F}(H)) \subseteq I_{H+X}^{(q)}(X\mathcal{G}(H))$. Let us now prove item (iii) by induction on r .

- Let us first show the implication for $r = 1$. Suppose $J_1^{(p:q)}(\mathcal{F}, H)$ holds. Let $X \in \mathbb{P}_p(V(K) \setminus M(K))$: we want to show that there exists $Y \in I_{K+X}^{(q)}(X\mathcal{F}(K))$. By $J_1^{(p:q)}(\mathcal{F}, H)$, there exists $Y' \in I_{H+X}^{(q)}(X\mathcal{F}(H))$. We would like to define $Y := Y'$, but we might not have $Y' \subseteq V(K)$. Instead, let $Y \in \mathbb{P}_q(V(K+X) \setminus M(K+X))$ such that $Y' \cap V(K) \subseteq Y$. For all $D \in X\mathcal{F}(K) \subseteq X\mathcal{F}(H)$, we have $Y' \cap V(D) \neq \emptyset$ since $Y' \in I_{H+X}^{(q)}(X\mathcal{F}(H))$ hence $Y \cap V(D) \neq \emptyset$, therefore $Y \in I_{K+X}^{(q)}(X\mathcal{F}(K))$.
- Now, let $r \geq 2$ such that the implication is true for $J_{r-1}^{(p:q)}(\mathcal{F}, \cdot)$. Suppose $J_r^{(p:q)}(\mathcal{F}, H)$ holds. Let $X \in \mathbb{P}_p(V(K) \setminus M(K))$: we want to show that there exists $Y \in I_{K+X}^{(q)}(X\mathcal{F}(K))$ such that $J_{r-1}^{(p:q)}(\mathcal{F}, K+X-Y)$ holds. By $J_r^{(p:q)}(\mathcal{F}, H)$, there exists $Y' \in I_{H+X}^{(q)}(X\mathcal{F}(H))$ such that $J_{r-1}^{(p:q)}(\mathcal{F}, H+X-Y')$ holds. Again, let $Y \in \mathbb{P}_q(V(K+X) \setminus M(K+X))$ such that $Y' \cap V(K) \subseteq Y$. For all $D \in X\mathcal{F}(K) \subseteq X\mathcal{F}(H)$, we have $Y' \cap V(D) \neq \emptyset$ since $Y' \in I_{H+X}^{(q)}(X\mathcal{F}(H))$ hence $Y \cap V(D) \neq \emptyset$, therefore $Y \in I_{K+X}^{(q)}(X\mathcal{F}(K))$. Moreover, the fact that $Y' \cap V(K) \subseteq Y$ ensures that $K+X-Y$ is a subhypergraph of $H+X-Y'$, so the fact that $J_{r-1}^{(p:q)}(\mathcal{F}, H+X-Y')$ holds implies that $J_{r-1}^{(p:q)}(\mathcal{F}, K+X-Y)$ also holds by the induction hypothesis. ■

In general, $J_r^{(p:q)}(\mathcal{F}, H)$ is stronger than $J_{r-1}^{(p:q)}(\mathcal{F}, H)$, because dangers can appear during the game: every time Maker plays some X , that might create new \mathcal{F} -dangers elsewhere since the vertices in X are now marked. Of course, dangers can also disappear during the game: every time Breaker plays some Y , that removes all \mathcal{F} -dangers intersecting Y .

Example. The (non-marked) hypergraph H from Figure I.3 illustrates the difference between properties $J_1^{(p:q)}(\mathcal{F}, H)$ and $J_2^{(p:q)}(\mathcal{F}, H)$. Assume $p = q = 1$, and suppose the \mathcal{F} -dangers in H are as follows: $x\mathcal{F}(H) = \{D_1, D_2\}$, $z\mathcal{F}(H) = \{D_3, D_4\}$, and $a\mathcal{F}(H) = \emptyset$ for all $a \in V(H) \setminus \{x, z\}$. Since $I_{H+x}(x\mathcal{F}(H)) = \{y\} \neq \emptyset$ and $I_{H+z}(z\mathcal{F}(H)) = \{u, v\} \neq \emptyset$, property $J_1(\mathcal{F}, H)$ holds. Suppose that Maker picks x : Breaker has to pick y to destroy the \mathcal{F} -dangers at x . Now suppose that, though C was not an \mathcal{F} -danger at z , $C+x$ is one: this means that, by picking x in the first round, Maker has created a third \mathcal{F} -danger at z in addition to the already existing ones D_3 and D_4 . Since $y \notin V(D_3) \cup V(D_4) \cup V(C+x)$, we have $z\mathcal{F}(H+x-y) = \{D_3, D_4, C+x\}$ hence $I_{H+x-y+z}(z\mathcal{F}(H+x-y)) = \emptyset$, so $J_1(\mathcal{F}, H+x-y)$ does not hold and neither does $J_2(\mathcal{F}, H)$. After the first round, Maker can simply pick z and go on to win.

Given a class \mathcal{H} of marked hypergraphs, which we will usually assume contains no trivial $(p : q)$ -Maker wins, we would like to find a family of dangers \mathcal{F} as simple as possible and a constant r as small as possible such that the necessary condition from Proposition I.2.19 is actually sufficient on \mathcal{H} , that is:

$$\text{For all } H \in \mathcal{H}: H \text{ is a } (p : q)\text{-Breaker win if and only if } J_r^{(p:q)}(\mathcal{F}, H) \text{ holds.} \quad (*)$$

Since property $J_r^{(p:q)}(\mathcal{F}, H)$ does not seem to guarantee anything after the first r rounds, a statement such as $(*)$ is very strong. In particular, if \mathcal{F} is efficiently identifiable i.e. deciding

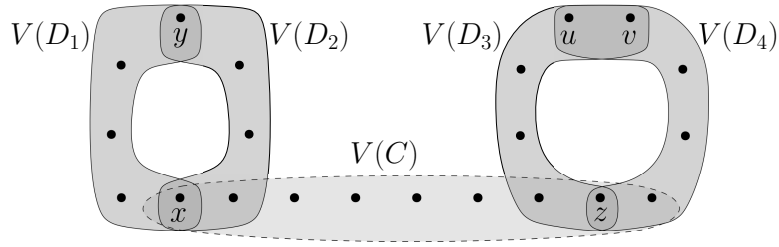


Figure I.3: Some vertex subsets in a hypergraph. The edges are not represented.

whether there exists an \mathcal{F} -danger at a given X in a given $H \in \mathcal{H}$ on n vertices can be done in polynomial time $P(n)$, then (*) would yield a $O(n^{2r}P(n))$ polynomial-time algorithm determining the outcome of the Maker-Breake game on the class \mathcal{H} .

One of the main results of this dissertation will be to exhibit a simple and efficiently identifiable family of dangers \mathcal{F} such that (*) holds for $p = q = 1$ and $r = 3$ on the class \mathcal{H} of all 3-uniform marked hypergraphs.

I.2.2.3 Danger prevention

The goal of this segment is to show that $J_r^{(p;q)}(\mathcal{F}, H)$ is equivalent to $J_1^{(p;q)}(\mathcal{F}^{*(r-1)}, H)$ for some family of dangers $\mathcal{F}^{*(r-1)}$ that we are going to introduce. In other words, preventing issues with the \mathcal{F} -dangers that could arise in the first r rounds comes down to dealing with a larger family of dangers as soon as the first round. Even though all arguments generalize to any bias, we set $p = q = 1$ to not aggravate the notations, which is enough for us anyway since the results below will only be applied in the unbiased case.

The idea is the following. Say Breaker wants to be able to manage the \mathcal{F} -dangers in the second round. Maker now picks x . As Breaker ponders his answer y , he already needs to think about the (yet unknown) vertex z that Maker is going to pick next. Now that x is marked, the collection of \mathcal{F} -dangers at z is $z\mathcal{F}(H^{+x})$: Breaker must choose a vertex y such that $z\mathcal{F}(H^{+x}) - y$ is intersecting, so as to be able to destroy all the remaining \mathcal{F} -dangers at z in the next round. By Proposition I.2.7, this means y must hit all unions of obstructions of the collection $z\mathcal{F}(H^{+x})$, or equivalently (from the viewpoint of H rather than H^{+x}) of the collection $\{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\}$. This must hold for all possibilities of Maker's next pick z . Here is the rigorous result:

Proposition I.2.21. *Let \mathcal{F} be a family of dangers. Let H be a marked hypergraph such that $|V(H) \setminus M(H)| \geq 4$, and let $x \in V(H) \setminus M(H)$ and $y \in V(H^{+x}) \setminus M(H^{+x})$. Then the following two assertions are equivalent:*

(a) $J_1(\mathcal{F}, H^{+x-y})$ holds.

(b) $y \in I_{H^{+x}} \left(\bigcup_{z \in V(H^{+x}) \setminus M(H^{+x})} \{ \langle \mathcal{O} \rangle, \mathcal{O} \in \mathfrak{O}_{H^{+x}+z}(\{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\}) \} \right)$.

Proof. We make a series of innocuous rewritings before applying Proposition I.2.7. First of all, recall that by definition:

(a) $\iff \forall z \in V(H^{+x-y}) \setminus M(H^{+x-y}), I_{H^{+x-y}+z}(z\mathcal{F}(H^{+x-y})) \neq \emptyset$.

The subhypergraphs of H^{+x-y} are exactly the subhypergraphs of H^{+x} that do not contain y , so:

(a) $\iff \forall z \in V(H^{+x-y}) \setminus M(H^{+x-y}), I_{H^{+x-y}+z}(z\mathcal{F}(H^{+x}) - y) \neq \emptyset$.

Consider the collection $z\mathcal{F}(H^{+x}) - y$: since its elements do not contain y , if it is nonempty then

its intersection in H^{+x-y+z} is the same as in H^{+x+z} . Therefore:

$$(a) \iff \forall z \in V(H^{+x-y}) \setminus M(H^{+x-y}), I_{H^{+x+z}}(z\mathcal{F}(H^{+x}) - y) \neq \emptyset.$$

Since the intersection of a collection does not depend on the marked vertices of its elements, this can be reformulated in terms of subhypergraphs of H rather than H^{+x} :

$$(a) \iff \forall z \in V(H^{+x-y}) \setminus M(H^{+x-y}), I_{H^{+x+z}}(\{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\} - y) \neq \emptyset.$$

We now use Proposition I.2.7 applied in H^{+x+z} with $q = q' = 1$ and $\mathcal{K} = \{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\}$, which yields:

$$(a) \iff \forall z \in V(H^{+x-y}) \setminus M(H^{+x-y}), \\ y \in I_{H^{+x+z}}(\{\langle \mathcal{O} \rangle, \mathcal{O} \in \mathfrak{O}_{H^{+x+z}}(\{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\})\}).$$

Since $y \neq z$, this can be rewritten as:

$$(a) \iff \forall z \in V(H^{+x-y}) \setminus M(H^{+x-y}), \\ y \in I_{H^{+x}}(\{\langle \mathcal{O} \rangle, \mathcal{O} \in \mathfrak{O}_{H^{+x+z}}(\{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\})\}).$$

Finally, the assertion " $y \in I_{H^{+x}}(\{\langle \mathcal{O} \rangle, \mathcal{O} \in \mathfrak{O}_{H^{+x+z}}(\{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\})\})$ " would trivially be true for $z = y$ since all elements of $\{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\}$ contain z . Therefore:

$$(a) \iff \forall z \in V(H^{+x}) \setminus M(H^{+x}), \\ y \in I_{H^{+x}}(\{\langle \mathcal{O} \rangle, \mathcal{O} \in \mathfrak{O}_{H^{+x+z}}(\{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\})\}) \\ \iff y \in I_{H^{+x}}\left(\bigcup_{z \in V(H^{+x}) \setminus M(H^{+x})} \{\langle \mathcal{O} \rangle, \mathcal{O} \in \mathfrak{O}_{H^{+x+z}}(\{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\})\}\right). \blacksquare$$

The subhypergraphs $\langle \mathcal{O} \rangle$ from Proposition I.2.21 that contain x may thus be interpreted as dangers at x , since Breaker has to destroy them. We will call them $\mathcal{F}^\mathfrak{O}$ -dangers at x :

Notation I.2.22. Let \mathcal{F} be a family of dangers. We denote by $\mathcal{F}^\mathfrak{O}$ the family of all pointed marked hypergraphs (D, x) such that, for some non-marked $z \neq x$ which we call an \mathcal{F} -dangerous vertex in (D, x) , we can write $D = \langle \mathcal{O} \rangle$ where the collection \mathcal{O} satisfies the following properties:

- each $K \in \mathcal{O}$ containing x is such that K^{+x} is an \mathcal{F} -danger at z ;
- each $K \in \mathcal{O}$ not containing x is already an \mathcal{F} -danger at z ;
- $I_{D^{+x+z}}(\mathcal{O}) = \emptyset$.

In other words, given a marked hypergraph H and a vertex $x \in V(H) \setminus M(H)$, we have:

$$x\mathcal{F}^\mathfrak{O}(H) = \bigcup_{z \in V(H^{+x}) \setminus M(H^{+x})} \{\langle \mathcal{O} \rangle, \mathcal{O} \in \mathfrak{O}_{H^{+x+z}}(\{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\}), x \in V(\langle \mathcal{O} \rangle)\}.$$

Example. Going back to the example in Figure I.3, we have $\{D_3, D_4, C^{+x}\} \subseteq z\mathcal{F}(H^{+x})$ i.e. $\mathcal{O} := \{D_3, D_4, C\} \subseteq \{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\}$, moreover $I_{H^{+x+z}}(\mathcal{O}) = \emptyset$ so $D := \langle \mathcal{O} \rangle = D_3 \cup D_4 \cup C \in x\mathcal{F}^\mathfrak{O}(H)$.

Proposition I.2.23. Let \mathcal{F} and \mathcal{G} be families of dangers. If $\mathcal{G} \subseteq \mathcal{F}$, then $\mathcal{G}^\mathfrak{O} \subseteq \mathcal{F}^\mathfrak{O}$.

Proof. This is clear since a \mathcal{G} -danger at z is also an \mathcal{F} -danger at z . ■

Proposition I.2.24. Let \mathcal{F} be a family of dangers. Then $\mathcal{F}^\mathfrak{O}$ is a family of dangers.

More precisely: for all $(D, x) \in \mathcal{F}^\mathfrak{O}$, if $|V(D^{+x}) \setminus M(D^{+x})| \geq 2$ then $J_1(\mathcal{F}, D^{+x})$ does not hold so D^{+x} is a Maker win, otherwise D^{+x} is a trivial Maker win.

Proof. Let $(D, x) \in \mathcal{F}^\mathfrak{O}$, and write $D = \langle \mathcal{O} \rangle$ as in the definition with z an \mathcal{F} -dangerous vertex in

(D, x) . If $|V(D^{+x}) \setminus M(D^{+x})| \geq 2$, then we can apply Proposition I.2.19: since $\mathcal{O}^{+x} \subseteq z\mathcal{F}(D^{+x})$, we have $I_{D^{+x+z}}(z\mathcal{F}(D^{+x})) \subseteq I_{D^{+x+z}}(\mathcal{O}^{+x}) = I_{D^{+x+z}}(\mathcal{O}) = \emptyset$, therefore $J_1(\mathcal{F}, D^{+x})$ does not hold so D^{+x} is a Maker win. If $|V(D^{+x}) \setminus M(D^{+x})| \leq 1$ i.e. $V(D^{+x}) \setminus M(D^{+x}) = \{z\}$, then let $K \in \mathcal{O}$: $(K^{+x})^{+z}$ is a Maker win whose vertices are all marked, so $(K^{+x})^{+z}$ has a fully marked edge, therefore K^{+x} is a trivial Maker win and so is $D^{+x} \supseteq K^{+x}$. ■

Proposition I.2.21 instantly yields the following result:

Proposition I.2.25. *Let \mathcal{F} be a family of dangers. Let H be a marked hypergraph such that $|V(H) \setminus M(H)| \geq 4$, and let $x \in V(H) \setminus M(H)$ and $y \in V(H^{+x}) \setminus M(H^{+x})$. Moreover, suppose that $J_1(\mathcal{F}, H)$ holds. Then the following two assertions are equivalent:*

- (a) $J_1(\mathcal{F}, H^{+x-y})$ holds.
- (b) $y \in I_{H^{+x}}(x\mathcal{F}^\heartsuit(H))$.

Proof. Given the characterization of $x\mathcal{F}^\heartsuit(H)$ from Notation I.2.22, the only difference with Proposition I.2.21 is that the subhypergraphs $\langle \mathcal{O} \rangle$ from Proposition I.2.21 do not necessarily contain x , whereas \mathcal{F}^\heartsuit -dangers at x do. This is where we use the additional assumption that $J_1(\mathcal{F}, H)$ holds. It is impossible that $x \notin V(\langle \mathcal{O} \rangle)$ for some $\mathcal{O} \in \mathcal{O}_{H^{+x+z}}(\{K \subseteq H, K^{+x} \in z\mathcal{F}(H^{+x})\})$: indeed, we would then have $\mathcal{O} \subseteq z\mathcal{F}(H)$ hence $I_{H^{+z}}(z\mathcal{F}(H)) \subseteq I_{H^{+z}}(\mathcal{O}) = \emptyset$, contradicting $J_1(\mathcal{F}, H)$. Therefore, under property $J_1(\mathcal{F}, H)$, the collection from item (b) in Proposition I.2.21 coincides exactly with $x\mathcal{F}^\heartsuit(H)$. ■

Let us now introduce the families of dangers that correspond to the multiple-round prevention of intersection issues with \mathcal{F} -dangers.

Notation I.2.26. Let \mathcal{F} be a family of dangers. For all $r \in \mathbb{N}$, we define a family of dangers \mathcal{F}^{*r} , recursively as follows:

- $\mathcal{F}^{*0} := \mathcal{F}$.
- For $r \geq 1$: $\mathcal{F}^{*r} := \mathcal{F} \cup (\mathcal{F}^{*(r-1)})^\heartsuit$. The family $\mathcal{F}^{*1} = \mathcal{F} \cup \mathcal{F}^\heartsuit$ may be denoted as \mathcal{F}^* .

Proposition I.2.27. *Let \mathcal{F} be a family of dangers, and let $r \in \mathbb{N}$.*

- (i) *For any family of dangers $\mathcal{G} \subseteq \mathcal{F}$: $\mathcal{G}^{*r} \subseteq \mathcal{F}^{*r}$.*
- (ii) *For any integer $0 \leq s \leq r$: $\mathcal{F}^{*s} \subseteq \mathcal{F}^{*r}$.*
- (iii) *$\mathcal{F}^{*r} = (\mathcal{F}^{*(r-1)})^*$.*

Proof. (i) We proceed by induction on r . For $r = 0$, there is nothing to show. Now suppose that $r \geq 1$ and that the result holds for $r - 1$. Let $\mathcal{G} \subseteq \mathcal{F}$. By definition: $\mathcal{G}^{*r} = \mathcal{G} \cup (\mathcal{G}^{*(r-1)})^\heartsuit$. We have $\mathcal{G} \subseteq \mathcal{F}$, moreover $\mathcal{G}^{*(r-1)} \subseteq \mathcal{F}^{*(r-1)}$ by the induction hypothesis hence $(\mathcal{G}^{*(r-1)})^\heartsuit \subseteq (\mathcal{F}^{*(r-1)})^\heartsuit$, so in conclusion $\mathcal{G}^{*r} \subseteq \mathcal{F} \cup (\mathcal{F}^{*(r-1)})^\heartsuit = \mathcal{F}^{*r}$.

(ii) Again, we proceed by induction on r . For $r = 0$, there is nothing to show. Now suppose that $r \geq 1$ and that the result holds for $r - 1$. Let $0 \leq s \leq r$ be an integer: we have $\mathcal{F}^{*s} = \mathcal{F} \cup (\mathcal{F}^{*(s-1)})^\heartsuit$ by definition. Moreover, the induction hypothesis ensures that $\mathcal{F}^{*(s-1)} \subseteq \mathcal{F}^{*(r-1)}$ hence $(\mathcal{F}^{*(s-1)})^\heartsuit \subseteq (\mathcal{F}^{*(r-1)})^\heartsuit$, therefore $\mathcal{F}^{*s} \subseteq \mathcal{F} \cup (\mathcal{F}^{*(r-1)})^\heartsuit = \mathcal{F}^{*r}$.

(iii) Since $\mathcal{F} \subseteq \mathcal{F}^{*(r-1)}$, we have $\mathcal{F}^{*r} = \mathcal{F} \cup (\mathcal{F}^{*(r-1)})^\heartsuit \subseteq \mathcal{F}^{*(r-1)} \cup (\mathcal{F}^{*(r-1)})^\heartsuit = (\mathcal{F}^{*(r-1)})^*$. On the other hand, we have $\mathcal{F}^{*(r-1)} \subseteq \mathcal{F}^{*r}$ by item (ii) and $(\mathcal{F}^{*(r-1)})^\heartsuit \subseteq \mathcal{F}^{*r}$ by definition of \mathcal{F}^{*r} , therefore $(\mathcal{F}^{*(r-1)})^* = \mathcal{F}^{*(r-1)} \cup (\mathcal{F}^{*(r-1)})^\heartsuit \subseteq \mathcal{F}^{*r}$. ■

We can now rephrase our intersection property in terms of dangers in the first round only:

Proposition I.2.28. *Let \mathcal{F} be a family of dangers and let $r \geq 1$ be an integer. Then, for all marked hypergraph H such that $|V(H) \setminus M(H)| \geq 2r$, the properties $J_r(\mathcal{F}, H)$ and $J_1(\mathcal{F}^{*(r-1)}, H)$ are equivalent.*

Proof. We proceed by induction on r . For $r = 1$, this statement is a tautology. Let $r \geq 2$ such that the equivalence holds for $r - 1$. By definition, $J_r(\mathcal{F}, H)$ means that:

$$\begin{aligned}
& \forall x \in V(H) \setminus M(H), \exists y \in I_{H+x}(x\mathcal{F}(H)) \text{ such that } J_{r-1}(\mathcal{F}, H^{+x-y}) \text{ holds} \\
\stackrel{\text{ind. hyp.}}{\iff} & \forall x \in V(H) \setminus M(H), \exists y \in I_{H+x}(x\mathcal{F}(H)) \text{ such that } J_1(\mathcal{F}^{*(r-2)}, H^{+x-y}) \text{ holds} \\
\stackrel{\text{Pro. I.2.25}}{\iff} & \forall x \in V(H) \setminus M(H), \exists y \in I_{H+x}(x\mathcal{F}(H)) \text{ such that } y \in I_{H+x}(x(\mathcal{F}^{*(r-2)})^\ominus(H)) \\
& \iff \forall x \in V(H) \setminus M(H), \exists y \in I_{H+x}(x(\mathcal{F} \cup (\mathcal{F}^{*(r-2)})^\ominus)(H)) \\
& \iff J_1(\mathcal{F} \cup (\mathcal{F}^{*(r-2)})^\ominus, H) \\
& \iff J_1(\mathcal{F}^{*(r-1)}, H).
\end{aligned}$$

The use of Proposition I.2.25 is justified by the fact that both $J_r(\mathcal{F}, H)$ and $J_1(\mathcal{F}^{*(r-1)}, H)$ imply $J_1(\mathcal{F}^{*(r-2)}, H)$: indeed, $J_r(\mathcal{F}, H)$ implies $J_{r-1}(\mathcal{F}, H)$ which is equivalent to $J_1(\mathcal{F}^{*(r-2)}, H)$ by the induction hypothesis, while $J_1(\mathcal{F}^{*(r-1)}, H)$ implies $J_1(\mathcal{F}^{*(r-2)}, H)$ because $\mathcal{F}^{*(r-1)} \supseteq \mathcal{F}^{*(r-2)}$. ■

The advantage of $J_1(\mathcal{F}^{*(r-1)}, H)$ over the equivalent property $J_r(\mathcal{F}, H)$ is that we study a single fixed hypergraph H , instead of having to consider all hypothetical evolutions of H during r rounds. However, this is done at the cost of a bigger and possibly much more complex family of dangers. If $\mathcal{F}^{*(r-1)}$ is somewhat manageable, then we will prefer to work with property $J_1(\mathcal{F}^{*(r-1)}, H)$.

I.2.2.4 Restricted obstructions

There can be redundancies in the family $\mathcal{F}^* = \mathcal{F} \cup \mathcal{F}^\ominus$, in the sense that an \mathcal{F}^\ominus -danger at x might contain an \mathcal{F} -danger at x . Such \mathcal{F}^\ominus -dangers may be ignored:

Notation I.2.29. Let \mathcal{F} be a family of dangers. We denote by $\mathcal{F}^{\ominus, \text{rest}} \subseteq \mathcal{F}^\ominus$ the family of all $(D, x) \in \mathcal{F}^\ominus$ such that D contains no \mathcal{F} -danger at x .

Proposition I.2.30. *Let \mathcal{F} is a family of dangers. Let H be a marked hypergraph and let $x \in V(H) \setminus M(H)$. Then $I_{H+x}(x\mathcal{F}^*(H)) = I_{H+x}(x(\mathcal{F} \cup \mathcal{F}^{\ominus, \text{rest}})(H))$.*

Proof. Obviously, $I_{H+x}(x\mathcal{F}^*(H)) \subseteq I_{H+x}(x(\mathcal{F} \cup \mathcal{F}^{\ominus, \text{rest}})(H))$ since $\mathcal{F} \cup \mathcal{F}^{\ominus, \text{rest}} \subseteq \mathcal{F}^*$. Moreover, let $y \in I_{H+x}(x(\mathcal{F} \cup \mathcal{F}^{\ominus, \text{rest}})(H))$: for all $D \in x\mathcal{F}^*(H)$, either D contains an \mathcal{F} -danger D' at x hence $y \in V(D') \subseteq V(D)$, or by definition $D \in x\mathcal{F}^{\ominus, \text{rest}}(H)$ hence $y \in V(D)$. Therefore $I_{H+x}(x(\mathcal{F} \cup \mathcal{F}^{\ominus, \text{rest}})(H)) \subseteq I_{H+x}(x\mathcal{F}^*(H))$, which concludes. ■

I.2.3 First results: the example of trivial dangers

As a concrete example, let us consider one of the simplest families of dangers imaginable: dangers that have exactly one edge. Let $(p : q)$ be any fixed bias.

Definition I.2.31. A p -trivial danger of size k is a pointed marked hypergraph (D, X) with $|X| = p$ such that: D has exactly one edge e , $V(D) = X \cup e$, $|e| = k$, $|e \setminus M(e)| > p$ and $|e \setminus (M(e) \cup X)| \leq p$. In the case $p = 1$, we may simply call this a *trivial danger of size k* .

Notation I.2.32. We denote by $\mathcal{F}_{\text{triv}}^{(p)}$ the family of all p -trivial dangers (of all sizes).

Remark. A p -trivial danger (D, X) is indeed a danger for any q , since D^{+X} contains an edge with between 1 and p non-marked vertices and thus is a trivial $(p : q)$ -Maker win. For $p = 1$, there exists a unique trivial danger (D, x) of given size up to isomorphism: D consists of a single edge e that has exactly two non-marked vertices including x (see Figure I.4). For $p \geq 2$, there is no unicity (see Figure I.5).

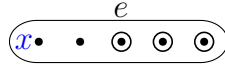


Figure I.4: The trivial danger (D, x) of size 5.

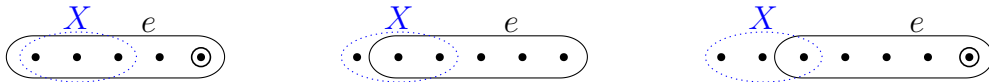


Figure I.5: Examples of 3-trivial dangers (D, X) of size 5.

I.2.3.1 A matter of survival... and more

Proposition I.1.46 has already stated the straightforward observation that Breaker survives the first round (i.e. $\tau_M^{(p:q)}(H) \geq 2$) if and only if H is not a trivial $(p : q)$ -Maker win (i.e. all edges of H have more than p non-marked vertices). So, when does Breaker survive the first two rounds? For this, he needs all edges to have more than p non-marked vertices, not only in H , but also in the updated marked hypergraph obtained after the first round: therefore, his first move Y must destroy all p -trivial dangers at Maker's first move X . More generally, the exact value of $\tau_M^{(p:q)}(H)$ is determined by how long Breaker can cope with the p -trivial dangers for. For as long as Breaker destroys the p -trivial dangers at each X that Maker plays, he cannot lose, since all edges have more than p non-marked vertices after his move. Conversely, as soon as Breaker fails to destroy all the p -trivial dangers at some X played by Maker, he loses, since this means some edge will have at most p non-marked vertices after his move: Maker can win by simply taking all of them in the next round. Formally, this reasoning yields:

Proposition I.2.33. *Let H be a marked hypergraph that is not a trivial $(p : q)$ -Maker win, and let $r \geq 1$ be an integer. Then $\tau_M^{(p:q)}(H) \geq r + 2$ if and only if $J_r^{(p:q)}(\mathcal{F}_{\text{triv}}^{(p)}, H)$ holds. ■*

Corollary I.2.34. *Let H be a marked hypergraph that is not a trivial $(p : q)$ -Maker win. Then H is a $(p : q)$ -Breaker win if and only if $J_R^{(p:q)}(\mathcal{F}_{\text{triv}}^{(p)}, H)$ holds, where $R := \lfloor \frac{|V(H) \setminus M(H)|}{p+q} \rfloor - 1$.*

Proof. Since $\lfloor \frac{|V(H) \setminus M(H)|}{p+q} \rfloor$ is the total number of rounds played when all vertices are taken, H is a $(p : q)$ -Breaker win if and only if $\tau_M^{(p:q)}(H) > \lfloor \frac{|V(H) \setminus M(H)|}{p+q} \rfloor$, hence the result by Proposition I.2.33. ■

Putting together Corollary I.2.34 and Theorem I.2.16, we now have two equivalences satisfying the type $(*)$ that we are interested in. Indeed, denoting by \mathcal{F}_{all} the family of all dangers, we get for all H that is not a trivial $(p : q)$ -Maker win:

$$\begin{aligned} H \text{ is a } (p : q)\text{-Breaker win} &\iff J_1^{(p:q)}(\mathcal{F}_{\text{all}}, H) \text{ holds,} \\ &\iff J_{\lfloor \frac{|V(H) \setminus M(H)|}{p+q} \rfloor - 1}^{(p:q)}(\mathcal{F}_{\text{triv}}^{(p)}, H) \text{ holds.} \end{aligned}$$

Unsurprisingly given how extreme these two families of dangers are, neither of the two equivalences above is satisfactory: the first one because \mathcal{F}_{all} is huge and not understood, the second one because the number of rounds is unbounded. Their interest, be it structurally or algorithmically, is therefore very limited. As we have mentioned before, given some class \mathcal{H} of marked hypergraphs, we would like a perfect middle ground: a family \mathcal{F} whose elements are well understood and efficiently identifiable (like $\mathcal{F}_{\text{triv}}^{(p)}$) but also satisfies the equivalence for some constant r (like \mathcal{F}_{all}), so that in particular we get a polynomial-time algorithm solving the game on the class \mathcal{H} . An easy example is the class of 2-uniform hypergraphs i.e. graphs. The solution of the Maker-Breaker game on graphs is straightforward and has already been stated in Theorem I.1.33. In the case $p \geq 2$, the $(p : q)$ -Maker wins are exactly the trivial $(p : q)$ -Maker wins i.e. the graphs containing at least one edge. For $p = 1$, the result can be formulated in terms of trivial dangers, so that we get (*) with $\mathcal{F} = \mathcal{F}_{\text{triv}}^{(1)}$ and $r = 1$:

Theorem I.2.35. *Let G be a graph on at least $1 + q$ vertices. Then G is a $(1 : q)$ -Breaker win if and only if $J_1^{(1:q)}(\mathcal{F}_{\text{triv}}^{(1)}, G)$ holds.*

Proof. Let $x \in V(G)$. The collection $x\mathcal{F}_{\text{triv}}^{(1)}(G)$ is precisely the set of edges of G incident to x , so its q -transversals in G^{+x} are the $Y \in \mathbb{P}_q(V(G) \setminus \{x\})$ that contain all neighbors of x in G . Therefore, we have $I_{G^{+x}}(x\mathcal{F}_{\text{triv}}^{(1)}(G)) \neq \emptyset$ if and only if the degree of x in G is at most q . In conclusion, $J_1^{(1:q)}(\mathcal{F}_{\text{triv}}^{(1)}, G)$ holds if and only if $q \geq \Delta(G)$, which is equivalent to G being a $(1 : q)$ -Breaker win according to Theorem I.1.33. ■

Corollary I.2.36. *Any graph G that is a $(1 : q)$ -Maker win satisfies $\tau_M^{(1:q)}(G) = \theta_M^{(1:q)}(G) = 2$.*

Proof. Proposition I.1.48 ensures that $2 \leq \theta_M^{(1:q)}(G) \leq \tau_M^{(1:q)}(G)$ for any graph G . If G is a $(1 : q)$ -Maker win, then $J_1^{(1:q)}(\mathcal{F}_{\text{triv}}^{(1)}, G)$ does not hold by Theorem I.2.35, so $\tau_M^{(1:q)}(G) < 3$ by Proposition I.2.33 which concludes. ■

For hypergraphs of rank $k \geq 3$ however, the p -trivial dangers are not sufficient on their own to get (*) for any constant r . Indeed, by Proposition I.2.33, such a statement would imply that τ_M is bounded over all $(p : q)$ -Maker wins from the considered class, and it is not difficult to show that this is false if $k = 3$ and $p = q = 1$ for instance (see Subsection III.2.3).

Nevertheless, given some class \mathcal{H} , suppose that we can find a family \mathcal{F} containing $\mathcal{F}_{\text{triv}}^{(p)}$ and such that the property is eventually hereditary, meaning that there exists r such that $J_r^{(p:q)}(\mathcal{F}, H)$ implies $J_{r+1}^{(p:q)}(\mathcal{F}, H)$. By induction, this means $J_r^{(p:q)}(\mathcal{F}, H)$ implies $J_s^{(p:q)}(\mathcal{F}, H)$ for all s and thus implies that H is a $(p : q)$ -Breaker win by Corollary I.2.34. Therefore, \mathcal{F} and r then satisfy (*) for \mathcal{H} . This is how we are going to proceed in Section II.1 when dealing with the class \mathcal{H} of 3-uniform marked hypergraphs.

I.2.3.2 General counting results on the biased Maker-Breaker game

We consider the following Breaker strategy: in each round, if it is possible, Breaker plays any move Y that destroys the p -trivial dangers at whatever X Maker has just played, otherwise he plays arbitrarily. How does that very naive short-term strategy perform? We know it is optimal in graphs, and even though it is obviously not the best in general, we will see that it can do surprisingly well. Our results use counting arguments involving the following generalization of the maximum degree:

Notation I.2.37. Let H be a (marked) hypergraph.

- Let $U \subseteq V(H)$. The *degree of U in H* is defined as $d_H(U) := |\{e \in E(H), U \subseteq e\}|$.

- Let $1 \leq j \leq |V(H)|$ be an integer. The *maximum j -degree* of H is defined as $\Delta_j(H) := \max_{U \subseteq V(H), |U|=j} d_H(U)$.

Interestingly, it is intuitively unclear how a parameter such as $\Delta_j(H)$ should influence the game. On the one hand, a low value of $\Delta_j(H)$ could be seen as beneficial to Breaker, as it means that Maker cannot create too many simultaneous threats. For instance, Theorem I.1.33 about graphs illustrates this with $j = 1$. On the other hand, a low value of $\Delta_j(H)$ could just as well be seen as beneficial to Maker, as it means that Breaker cannot hit too many of the edges that she has played in herself. For instance, Theorem I.1.31 illustrates this with $j = 2$, as it features $\Delta_2(H)$ on the lesser side of the inequality.

We are now going to further illustrate the former case: a low value of $\Delta_j(H)$ helps Breaker survive longer (sometimes, the entire game) with the naive strategy consisting in addressing nothing but the p -trivial dangers. We are going to give results in the form of lower bounds on our hypergraph parameters $\tau_M^{(p:q)}$ and $\theta_M^{(p:q)}$ (corresponding to time optimization and token optimization respectively), which imply results on the winner of the game by Proposition I.1.48. The key is the following lemma, which gives an upper bound on the number of p -trivial dangers that can exist simultaneously. Recall that $\text{ark}(H)$ denotes the anti-rank of H i.e. the size of a smallest edge in H , where we count the marked vertices as well.

Lemma I.2.38. *Let H be a marked hypergraph, and let $1 \leq j \leq \text{ark}(H) - p$ be an integer. Then, for all $X \in \mathbb{P}_p(V(H) \setminus M(H))$, we have $|X\mathcal{F}_{\text{triv}}^{(p)}(H)| \leq p \binom{|M(H)|+p-1}{j-1} \Delta_j(H)$.*

Proof. The lone edge e of a p -trivial danger at X in H has at least $|e| - p \geq j$ vertices in $M(H) \cup X$, including at least 1 vertex in X . There are p choices of $x \in X$ and $\binom{|M(H)|+p-1}{j-1}$ choices of $x_1, \dots, x_{j-1} \in M(H) \cup (X \setminus \{x\})$, and for each possibility there exist at most $\Delta_j(H)$ edges containing x, x_1, \dots, x_{j-1} , hence the result. ■

Suppose that H is not a trivial $(p : q)$ -Maker win. Since Breaker survives in the $(p : q)$ -game for as long as he can destroy the p -trivial dangers, and since the maximum j -degree obviously does not increase during play, Lemma I.2.38 ensures that Breaker wins if the number of marked vertices never becomes too big throughout the game. When is that number capped? For instance, it is if we are only considering a given finite number of rounds T , since the number of marked vertices then cannot exceed $|M(H)| + pT$: this gives us a lower bound for $\tau_M^{(p:q)}(H)$. It also is if we are only considering Maker strategies that use a given finite number of tokens N , since the number of marked vertices then cannot exceed $|M(H)| + N$: this gives us a lower bound for $\theta_M^{(p:q)}(H)$. Note that $\theta_M^{(p:q)}$ is the more relevant of the two parameters here: indeed, we are counting a number of threats that Maker creates, which is a function of how many vertices she possesses simultaneously, and this is precisely what the game with tokens accounts for. The result on $\theta_M^{(p:q)}$ then implies the one on $\tau_M^{(p:q)}$.

Proposition I.2.39. *Let H be a marked hypergraph that is not a trivial $(p : q)$ -Maker win, and let $1 \leq j \leq \text{ark}(H) - p$ be an integer. Let $N \geq p$ be an integer, and suppose that $q \geq p \binom{|M(H)|+N-1}{j-1} \Delta_j(H)$. Then $\theta_M^{(p:q)}(H) \geq N + 1$ and $\tau_M^{(p:q)}(H) \geq \frac{N+1}{p}$. In particular, if $\frac{N+1}{p} > \left\lceil \frac{|V(H)| - |M(H)|}{p+q} \right\rceil$ then H is a $(p : q)$ -Breaker win.*

Proof. Suppose that Maker's strategy uses at most N tokens. Breaker's strategy in each round is, if possible, to destroy all p -trivial dangers at the subset X that Maker has just played, and for the rest he picks arbitrary vertices. Let the $(p : q)$ -game on H play out. When Maker moves tokens that were already placed, we decompose the move into two consecutive actions: Maker removes tokens then places tokens, where we consider the "remove" action as part of the

previous round, so that at the end of each round she has at least p tokens in hand ready for the next round. Defining H_t as the marked hypergraph obtained at the end of round $t \geq 1$ (with $H_0 := H$), this implies $|M(H_t)| \leq |M(H)| + N - p$ for all $t \geq 0$. Also note that $t \mapsto \Delta_j(H_t)$ is nonincreasing since $E(H_{t+1}) \subseteq E(H_t)$ for all $t \geq 0$.

Claim 2. *For all $t \geq 0$ and for all $e \in E(H_t)$, we have $|e \setminus M(H_t)| > p$.*

Proof of Claim 2. We proceed by induction on t . For $t = 0$, this is simply our assumption that H is not a trivial $(p : q)$ -Maker win. Now, let $t \geq 1$ and suppose the result holds for $t - 1$: this means that, in the marked hypergraph H_{t-1} obtained at the end of round $t - 1$ i.e. at the beginning of round t , all edges have more than p non-marked vertices. The edges that will have at most p non-marked vertices after Maker's move are by definition the edges of p -trivial dangers at X in H_{t-1} , where X is the subset of vertices on which Maker places tokens in round t . By Lemma I.2.38:

$$|X\mathcal{F}_{\text{triv}}^{(p)}(H_{t-1})| \leq p \binom{|M(H_{t-1})| + p - 1}{j - 1} \Delta_j(H_{t-1}) \leq p \binom{|M(H)| + N - 1}{j - 1} \Delta_j(H) \leq q.$$

Following his strategy, Breaker thus picks q vertices that hit all p -trivial dangers at X . This ensures that, at the end of round t , all remaining edges are back to having more than p non-marked vertices. \square

For all $t \geq 1$, there is no edge with at most p non-marked vertices at the beginning of round t according to Claim 2, so Maker does not complete an edge during round t . This ensures that Breaker wins. In conclusion, we have $\theta_M^{(p;q)}(H) \geq N + 1$, and the rest ensues by Proposition I.1.48. \blacksquare

For $j = 1$, where the maximum j -degree coincides with the standard notion of maximum degree which we have already used in the case of graphs, the result from Proposition I.2.39 is obvious. Indeed, since the binomial coefficient equals 1 for all N , it simply states that if $q \geq p\Delta_1(H)$ then H is a $(p : q)$ -Breaker win. This is clear since, in each round, Breaker can then hit all edges incident to the vertices Maker has just picked. For $j \geq 2$ however, we get the following nontrivial lower bounds, which hold in all generality:

Corollary I.2.40. *Let H be a marked hypergraph that is not a trivial $(p : q)$ -Maker win, and let $2 \leq j \leq \text{ark}(H) - p$ be an integer. Then:*

- $\theta_M^{(p;q)}(H) \geq \left\lfloor \left(\frac{q(j-1)!}{p\Delta_j(H)} \right)^{\frac{1}{j-1}} \right\rfloor - |M(H)| + 2.$
- $\tau_M^{(p;q)}(H) \geq \frac{1}{p} \left(\left\lfloor \left(\frac{q(j-1)!}{p\Delta_j(H)} \right)^{\frac{1}{j-1}} \right\rfloor - |M(H)| + 2 \right).$

Proof. Assume $N := \left\lfloor \left(\frac{q(j-1)!}{p\Delta_j(H)} \right)^{\frac{1}{j-1}} \right\rfloor - |M(H)| + 1 \geq p$. We have $|M(H)| + N - 1 \leq \left(\frac{q(j-1)!}{p\Delta_j(H)} \right)^{\frac{1}{j-1}}$ hence $q \geq p \frac{(|M(H)| + N - 1)^{j-1}}{(j-1)!} \Delta_j(H) \geq p \binom{|M(H)| + N - 1}{j-1} \Delta_j(H)$. Therefore, Proposition I.2.39 concludes that $\theta_M^{(p;q)}(H) \geq N + 1$ and $\tau_M^{(p;q)}(H) \geq \frac{N+1}{p}$. \blacksquare

A final consequence of Proposition I.2.39 is a lower bound on the threshold bias, defined as the smallest q such that Breaker wins the $(1 : q)$ -game. To simplify computations, we state it for non-marked hypergraphs.

Corollary I.2.41. *Let H be a non-marked hypergraph that is not a trivial Maker win, and let $1 \leq j \leq \text{ark}(H) - 1$ be an integer. Then:*

$$q_{\text{thr}}(H) \leq \left\lceil |V(H)|^{1-\frac{1}{j}} \left(\frac{\Delta_j(H)}{(j-1)!} \right)^{\frac{1}{j}} \right\rceil.$$

Proof. We must show that H is a $(1 : q)$ -Breaker win where $q := \left\lceil |V(H)|^{1-\frac{1}{j}} \left(\frac{\Delta_j(H)}{(j-1)!} \right)^{\frac{1}{j}} \right\rceil$. Let $N := \left\lceil \frac{|V(H)|}{1+q} \right\rceil$. The definition of q yields $q^j \geq \frac{|V(H)|^{j-1}}{(j-1)!} \Delta_j(H)$, from which we have:

$$q \geq \frac{\left(\frac{|V(H)|}{q} \right)^{j-1}}{(j-1)!} \Delta_j(H) \geq \frac{\left(\frac{|V(H)|}{1+q} \right)^{j-1}}{(j-1)!} \Delta_j(H) \geq \frac{(N-1)^{j-1}}{(j-1)!} \Delta_j(H) \geq \binom{N-1}{j-1} \Delta_j(H).$$

Since $N+1 > \left\lceil \frac{|V(H)|}{1+q} \right\rceil$ by our choice of N , Proposition I.2.39 concludes that H is a $(1 : q)$ -Breaker win. \blacksquare

I.2.3.3 An application to a specific game around sums

We introduce a game played on integers, where Maker aims at claiming k of them that form a sum:

Notation I.2.42. For any $k \geq 3$ and any $n \geq 1$, define $H_{k,n}^{\text{sum}}$ by $V(H_{k,n}^{\text{sum}}) = \{1, \dots, n\}$ and $E(H_{k,n}^{\text{sum}}) = \{\{x_1, \dots, x_k\} \subseteq \{1, \dots, n\}, x_1 + \dots + x_{k-1} = x_k\}$. The x_i are assumed to be pairwise distinct in each edge, so that $H_{k,n}^{\text{sum}}$ is a k -uniform hypergraph on n vertices.

Proposition I.2.43. *For all $k \geq 3$, the following holds, where the multiplicative constants depend on k :*

- $|E(H_{k,n}^{\text{sum}})| = \Theta(n^{k-1})$.
- $\Delta_j(H_{k,n}^{\text{sum}}) = \Theta(n^{k-1-j})$ for all $1 \leq j \leq k-1$.

For $k = 3$, we have more specifically: $|E(H_{3,n}^{\text{sum}})| = \frac{(n-1)^2}{4}$ for all odd $n \geq 1$, $|E(H_{3,n}^{\text{sum}})| = \frac{n(n-2)}{4}$ for all even $n \geq 2$, and $\Delta_2(H_{3,n}^{\text{sum}}) = 2$ for all $n \geq 4$.

Proof. Since a sum $x_1 + \dots + x_{k-1} = x_k$ is entirely determined by the choice of x_1, \dots, x_{k-1} , we have $|E(H_{k,n}^{\text{sum}})| = O(n^{k-1})$. Moreover, every choice of pairwise distinct x_1, \dots, x_{k-1} between 1 and $\frac{n}{k-1}$ (for example) defines a different sum, so $|E(H_{k,n}^{\text{sum}})| = \Omega(n^{k-1})$ hence $|E(H_{k,n}^{\text{sum}})| = \Theta(n^{k-1})$. Similarly, there are $O(n^{k-1-j})$ sums containing any fixed x_1, \dots, x_j and there are $\Omega(n^{k-1-j})$ sums containing $1, \dots, j$ for instance, hence $\Delta_j(H_{k,n}^{\text{sum}}) = \Theta(n^{k-1-j})$.

Now suppose $k = 3$. For each integer $1 \leq x < \frac{n}{2}$, there are exactly $n - 2x$ sums that have x as their smallest addend. This yields $|E(H_{3,n}^{\text{sum}})| = (n-2) + (n-4) + \dots + 1 = \frac{(n-1)^2}{4}$ for odd n and $|E(H_{3,n}^{\text{sum}})| = (n-2) + (n-4) + \dots + 2 = \frac{n(n-2)}{4}$ for even n . Finally, for any fixed $1 \leq x < y \leq n$, there are at most two sums containing both x and y which are $x + y = (x + y)$ and $x + (y - x) = y$, moreover these two sums are valid for $x = 1$ and $y = 3$ provided that $n \geq 4$, hence $\Delta_2(H_{3,n}^{\text{sum}}) = 2$ in that case. \blacksquare

Theorems I.1.30 and I.1.31 are the go-to for early bounds, upper and lower respectively, on the threshold bias of any hypergraph. However, applying Theorem I.1.30 on $H_{k,n}^{\text{sum}}$, we get an upper bound $O(n)$, which is trivial on n vertices. On the contrary, as we are about to show, Corollary I.2.41 applied on $H_{k,n}^{\text{sum}}$ yields a nontrivial upper bound $O(n^{1-\frac{1}{k-1}})$. Therefore, the sum game constitutes an example where our result gives a better bound on the threshold bias than the Erdős-Selfridge criterion. Applying Theorem I.1.31 on $H_{k,n}^{\text{sum}}$, we get a lower bound $\Omega(n^{\frac{1}{k-1}})$. In

conclusion, we have lower and upper bounds in the order of \sqrt{n} for $k = 3$, whereas a significant gap subsists for $k \geq 4$:

Proposition I.2.44. *The threshold bias of $H_{k,n}^{\text{sum}}$ satisfies the following:*

- For $k = 3$: $q_{\text{thr}}(H_{3,n}^{\text{sum}}) = \Theta(\sqrt{n})$, and more precisely $\left\lceil \sqrt{\frac{n-2}{8}} \right\rceil \leq q_{\text{thr}}(H_{3,n}^{\text{sum}}) \leq \left\lceil \sqrt{2n} \right\rceil$.
- For $k \geq 4$: $q_{\text{thr}}(H_{k,n}^{\text{sum}}) = \Omega(n^{\frac{1}{k-1}})$ and $q_{\text{thr}}(H_{k,n}^{\text{sum}}) = O(n^{1-\frac{1}{k-1}})$.

Proof. Applying Corollary I.2.41 with $j = k - 1$, we get $q_{\text{thr}}(H_{k,n}^{\text{sum}}) \leq \left\lceil n^{1-\frac{1}{k-1}} \left(\frac{\Delta_{k-1}(H)}{(k-2)!} \right)^{\frac{1}{k-1}} \right\rceil$. Using Proposition I.2.43, this yields $q_{\text{thr}}(H_{k,n}^{\text{sum}}) = O(n^{1-\frac{1}{k-1}})$, and for $k = 3$ we get $q_{\text{thr}}(H_{3,n}^{\text{sum}}) \leq \left\lceil n^{1-\frac{1}{2}} \left(\frac{2}{1!} \right)^{\frac{1}{2}} \right\rceil = \left\lceil \sqrt{2n} \right\rceil$. Applying Theorem I.1.31, a necessary condition for a $(1 : q)$ -Breaker win is that $(1 + q)^{k-3} q^2 \geq \frac{|E(H_{k,n}^{\text{sum}})|}{\Delta_2(H_{k,n}^{\text{sum}})n}$. Using Proposition I.2.43, this yields $q_{\text{thr}}(H_{k,n}^{\text{sum}}) = \Omega(n^{\frac{1}{k-1}})$, and for $k = 3$ we get $q_{\text{thr}}(H_{3,n}^{\text{sum}}) \geq \sqrt{\frac{n(n-2)}{4 \times 2n}} = \sqrt{\frac{n-2}{8}}$. ■

Remark. Let us make a couple of observations in the case $k = 3$ (3-sum game):

- The naive Breaker strategy consisting in addressing nothing but the trivial dangers is actually optimal for the $(1 : q)$ -game on $H_{3,n}^{\text{sum}}$, except perhaps for the small window $q \in \left[\left\lceil \sqrt{\frac{n-2}{8}} \right\rceil, \left\lceil \sqrt{2n} \right\rceil \right]$. Indeed, if $q \geq \left\lceil \sqrt{2n} \right\rceil$ then Breaker wins with that strategy, and if $q < \left\lceil \sqrt{\frac{n-2}{8}} \right\rceil$ then Maker wins against any Breaker strategy.
- The 3-sum game is an example where the probabilistic intuition is wrong. Since Maker picks a total of $N := \left\lceil \frac{n}{1+q} \right\rceil$ vertices during the $(1 : q)$ -game, it comes down to the proportion of subsets of $\{1, \dots, n\}$ of size N that contain a 3-sum (also called a *Schur triple* in the literature). It is known [AMS⁺19] that this proportion is $o(1)$ if $N = o(n^{\frac{1}{3}})$ and $1 - o(1)$ if $N = \omega(n^{\frac{1}{3}})$, so the switch happens towards $q = n^{\frac{2}{3}}$, whereas we have seen that $q_{\text{thr}}(H_{3,n}^{\text{sum}}) = \Theta(\sqrt{n})$.

I.3 Elementary structures in (marked) hypergraphs

The short study that we have just performed around p -trivial dangers gave results of a purely quantitative nature, which say basically nothing about the unbiased case and are not relevant to sparse hypergraphs. To address all types of hypergraphs, and to hopefully get an efficient algorithm for the 3-uniform case which we are particularly interested in, quantities such as the number of edges or the maximum j -degree are not enough: we need to delve into the hypergraph structure. This section provides definitions, notations and preliminary results which the in-depth structural studies of Chapter II will be based upon. Since this section is about hypergraph structure, marked vertices will often be irrelevant, so most of it will be stated for hypergraphs despite extending to marked hypergraphs (via the structure of the underlying hypergraph obtained when ignoring which vertices are marked). For example, a central definition will be that of a path, which is a type of hypergraph, but a path in which some vertices are marked will also be called a path.

I.3.1 In general hypergraphs

I.3.1.1 Walks and paths

Definition I.3.1. A *walk* is a finite sequence $\vec{W} = (U_0, \dots, U_l)$, where U_0, \dots, U_l are subsets of some common set, such that $U_i \cap U_{i+1} \neq \emptyset$ for all $0 \leq i \leq l-1$. We define the *vertex set* of \vec{W} as $V(\vec{W}) := \bigcup_{0 \leq i \leq l} U_i$, and the *edge set* of \vec{W} as $E(\vec{W}) := \{U_i, 0 \leq i \leq l \text{ and } |U_i| \geq 2\}$.

Notation I.3.2. In a walk, a singleton $U_i = \{x\}$ might be simply denoted as x .

We are going to use walks as a way to navigate inside hypergraphs. They also help defining some elementary structures whose edge sets have a natural ordering. Usually, the elements of the walk will correspond to edges, plus some singletons which are useful to give information about intersections: for example, if a subsequence (\dots, e, x, e', \dots) appears inside of a walk where e and e' are edges and x is a vertex, then we know that $x \in e \cap e'$. Note that elements of a walk are not necessarily pairwise distinct.

Definition I.3.3. Two walks are said to be *equivalent* if they have the same vertex set and they coincide when removing all their singleton elements.

Notation I.3.4. Let $\vec{W} = (U_0, \dots, U_l)$ be a walk.

- Provided U_0, \dots, U_l are not all singletons, we denote by $\text{start}(\vec{W})$ (resp. $\text{end}(\vec{W})$) the non-singleton element U_i of smallest (resp. largest) index i .
- We define the reverse walk $\overleftarrow{W} = (U_l, \dots, U_0)$.
- If $\vec{W}' = (U'_0, \dots, U'_l)$ is another walk such that $U_l \cap U'_0 \neq \emptyset$, then we define the concatenated walk $\vec{W} \oplus \vec{W}' = (U_0, \dots, U_l, U'_0, \dots, U'_l)$.
- Given a set Z such that $Z \cap V(\vec{W}) \neq \emptyset$, we define $\vec{W}|_Z = (U_0, \dots, U_j)$ where $j = \min\{0 \leq i \leq l, Z \cap U_i \neq \emptyset\}$.

As alluded to before, some elementary hypergraphs can be defined as induced by a walk:

Definition I.3.5. The hypergraph *induced* by a walk \vec{W} is the hypergraph, denoted by $[\vec{W}]$, defined by $V([\vec{W}]) = V(\vec{W})$ and $E([\vec{W}]) = E(\vec{W})$.

Remark. It is important to note that, by definition, $E(\vec{W})$ does not include singletons. In particular, $[\vec{W}]$ always has anti-rank at least 2.

Paths are the main hypergraph structure that we are going to consider. Berge's definition of a path in a hypergraph [Ber73] corresponds to walks of the form $(v_1, e_1, v_2, e_2, \dots, v_L, e_L, v_{L+1})$ alternating between vertices and edges that are pairwise distinct. Instead, we define a path as a type of hypergraph, and we always keep the distinction between the two kinds of mathematical objects: the sequences of edges/vertices (walks) on one side, and the hypergraphs that they induce (e.g. paths) on the other.

Definition I.3.6. Let P be a hypergraph, and let $a, b \in V(P)$. We say P is an *ab-path* if there exists a walk inducing P of the form (a, e_1, \dots, e_L, b) where the e_i are pairwise distinct and of size at least 2. We then say $L = |E(P)|$ is the *length* of P . An *ab-path* may also be referred to as an *a-path* if we desire to highlight just one end point, or a *path* if we desire to highlight none.

Remark. Any ab -path is also a ba -path (take the reverse walk in the definition), an a -path, a b -path and a path. Note that the case $a = b$ is not excluded, and neither is the case $L = 0$ which implies $a = b$ by definition of a walk since we then get (a, b) . In particular, a hypergraph P with $V(P) = \{a\}$ and $E(P) = \emptyset$ is an aa -path of length 0.

This definition of a path is very loose (even looser than Berge's, in the absence of the intermediate vertices). In practice however, we will only consider paths having more specific properties, which are inspired by the Maker-Breaker game. In particular, the size of edge intersections will prove to be a crucial factor. Several paths defined by such restrictions have been studied in extremal hypergraph theory. For instance, a t -tight path (resp. a t -quasi-linear path) is one where any two consecutive edges intersect on at least (resp. exactly) t vertices [GKL10][Tom12]. On the contrary, we are interested in upper bounded edge intersection sizes. A linear path (sometimes: loose path), is one where any two consecutive edges intersect on exactly one vertex [FJS14]. The following global definition also exists: a linear (sometimes: almost-disjoint) hypergraph is one where any two distinct edges intersect on at most one vertex [NRS⁺82]. For reasons that we are going to explain very shortly, our main focus is on linear paths in 3-uniform hypergraphs, and we introduce the following notion in general hypergraphs which is the upper bounded counterpart of t -tightness (note that the case $\lambda = 1$ corresponds to the usual notion of linearity in hypergraphs):

Definition I.3.7. Let λ be an integer.

- We say a hypergraph H is λ -linear if $|e \cap e'| \leq \lambda$ for all distinct $e, e' \in E(H)$.
- We say a walk \vec{W} is λ -linear if its induced hypergraph $[\vec{W}]$ is λ -linear.

For $\lambda = 1$, we may simply say *linear* instead of "1-linear".

Another important notion is that of simplicity, which is a property of walks first and foremost:

Definition I.3.8. Let $\vec{W} = (U_0, \dots, U_l)$ be a walk.

- Let $x \in V(\vec{W})$. We say x is a *repeated vertex* in \vec{W} if there exist indices i, j such that $|i - j| \geq 2$ and $x \in U_i \cap U_j$.
- We say \vec{W} is *simple* if there is no repeated vertex in \vec{W} , i.e. if $U_i \cap U_j = \emptyset$ for all i, j such that $|i - j| \geq 2$.

Simplicity for paths is defined by asking the walk to be simple in Definition I.3.6. Note that this is dependent on the choice of a and b . We get:

Definition I.3.9. Let P be a hypergraph, and let $a, b \in V(P)$. We say P is an *ab-simple-path* if there exists a simple walk inducing P of the form (a, e_1, \dots, e_L, b) where each e_i is of size at least 2. An *ab-simple-path* may also be referred to as an *a-simple-path* if we desire to highlight just one end point, or a *simple path* if we desire to highlight none. See Figure I.6.

Remark. Any ab -simple-path is also a ba -simple-path (take the reverse walk in the definition), an a -simple-path, a b -simple-path and a simple path. Note that we have $L = 0$ if and only if $a = b$ this time: indeed, if $a = b$ then the two singletons must be consecutive for the walk to be simple hence $L = 0$.

Notation I.3.10. We introduce notations to retrieve the walk from the path, which will often be useful.

- Let P be an ab -simple-path. For fixed a and b , there is a unique walk (a, e_1, \dots, e_L, b) satisfying Definition I.3.9, since simplicity forces the ordering of the edges. That walk will

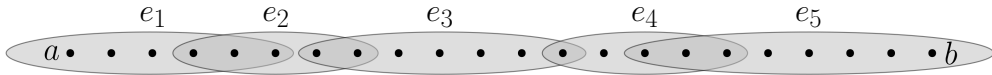


Figure I.6: An ab -simple-path of length $L = 5$. This particular one is 3-linear, as the biggest intersection between two edges is of size 3.

be denoted by \overrightarrow{aPb} .

- Let P be an a -simple-path. For fixed a , the walks satisfying Definition I.3.9 can only differ in the choice of the final singleton b , since simplicity forces the ordering of the edges. Their common part (a, e_1, \dots, e_L) , without the final singleton, will be denoted by $\overrightarrow{a\dot{P}}$.

Proposition I.3.11. *Any ab -path contains an ab -simple-path, and any λ -linear ab -path contains a λ -linear ab -simple-path.*

Proof. The second assertion directly ensues from the first since a subhypergraph of a λ -linear hypergraph is obviously λ -linear. Therefore, let us prove the first assertion. Let P be an ab -path. Assume $a \neq b$, otherwise the edgeless subhypergraph of P consisting of the single vertex $a = b$ is an ab -simple-path which concludes. Consider walks of the form (a, e_1, \dots, e_L, b) satisfying the following condition (C): e_1, \dots, e_L are pairwise distinct edges of P .

We know there exists a walk satisfying (C): indeed, by definition of an ab -path, there even exists one that induces P . Now, let $\overrightarrow{W} = (a, e_1, \dots, e_L, b)$ be a walk satisfying (C) with minimum L . Note that \overrightarrow{W} does not necessarily induce P , but it induces an ab -path which is a subhypergraph of P . To finish the proof, it thus suffices to show that \overrightarrow{W} is simple. We know $a \neq b$, moreover it is impossible that $a \in e_i$ for some $2 \leq i \leq L$ because the walk $W' := (a, e_i, \dots, e_L, b)$ would then satisfy (C) and contradict the minimality of \overrightarrow{W} , therefore a is not a repeated vertex in \overrightarrow{W} . An analogous reasoning shows that b is not a repeated vertex in \overrightarrow{W} . Finally, it is impossible that $e_i \cap e_j \neq \emptyset$ for some i, j such that $j - i \geq 2$, because the walk $W' := (a, e_1, \dots, e_i, e_j, \dots, e_L, b)$ would then satisfy (C) and contradict the minimality of \overrightarrow{W} . ■

I.3.1.2 Forcing paths

Our interest in 3-uniform linear paths comes from the role of the following elementary marked hypergraph in the unbiased Maker-Breaker game. In Figure I.7, the marked vertices are circled, and each edge of size 3 is represented by a "claw" shape joining its three vertices: this graphical representation will be used throughout this dissertation.

Definition I.3.12. An ab -nunchaku is a 3-uniform linear ab -simple-path N of positive length such that $M(N) = \{a, b\}$. An ab -nunchaku may also be referred to as an a -nunchaku or a nunchaku. See Figure I.7.

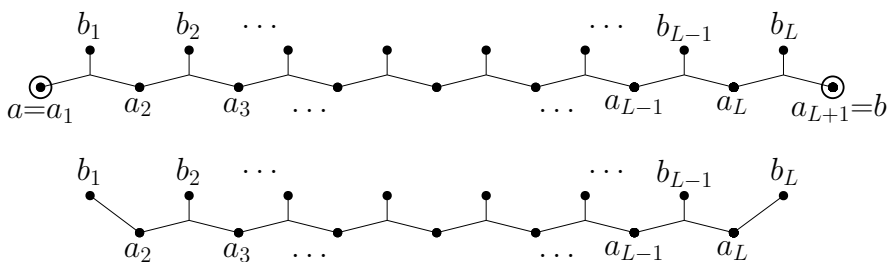


Figure I.7: An ab -nunchaku of length L (here $L = 8$), with its non-marked reduction below.

Nunchakus are named after their non-marked reduction in the case where the length is at least 2 (failing which the non-marked reduction is a single edge of size 1): the two edges of size 2 are like two handles with a chain between them. Their importance for the Maker-Breaker game on 3-uniform marked hypergraphs, or rather (equivalently) the importance of their non-marked reduction for the Maker-Breaker game on hypergraphs of rank 3, has been first identified in [Kut04]. An analogous object in a more general context is defined in [RW20] as a *manriki*, after another Japanese weapon. We will go back to both these papers in detail in Subsection II.1.1: for now, we are mainly giving definitions. The key idea is that a nunchaku is a basic example of a Maker win of rank 3, and there exists a winning "forcing strategy" where Maker forces all of Breaker's moves along the nunchaku starting from one end until Breaker is trapped at the other end.

Proposition I.3.13. *Any nunchaku is a Maker win.*

Proof. Let N be a nunchaku of length L . If $L = 1$ then N is obviously a trivial Maker win. Therefore, assume $L \geq 2$ and define $a = a_1, b_1, a_2, b_2, \dots, a_L, b_L, a_{L+1} = b$ as in Figure I.7. Maker picks a_2 , threatening to complete the edge $\{a, b_1, a_2\}$ on her next go: Breaker is forced to pick b_1 . Maker continues to force all of Breaker's picks along the path, by picking a_3, a_4, \dots, a_{L-1} successively which forces Breaker to pick b_2, b_3, \dots, b_{L-2} successively. Maker now picks a_L , threatening to pick either b_{L-1} or b_L on her next go, which would see her complete the edge $\{a_{L-1}, b_{L-1}, a_L\}$ or $\{a_L, b_L, b\}$ respectively. Breaker will lose in the next round as he cannot address both threats at once. ■

Remark. Two remarks regarding this forcing principle:

- In a 3-uniform marked hypergraph that has a linear ab -simple-path as a strict subhypergraph, this forcing technique might also be useful if a is marked but not b : it will not be enough to win the game, but it is a way for Maker to get all of a_2, a_3, \dots, a_{L+1} while making sure that Breaker gets exactly b_1, b_2, \dots, b_L in the meantime.
- Maker actually has a faster way to win on a nunchaku, by essentially cutting it in half each round: we will come back to this in Subsection III.2.3.

For $k \geq 4$, it is also possible to define a type of path that generalizes this idea. As we need more edges and bigger edge intersections for Breaker's moves to be forced compared to the 3-uniform case, this path is neither linear nor simple. However, it does satisfy a relaxed version of both notions: it is $(k-2)$ -linear, and each vertex is in at most $k-1$ consecutive edges of the path (exactly $k-1$ for those that Maker will pick bar the last one). The general definition is as follows:

Definition I.3.14. Let $k \geq 3$ be an integer. A k -uniform *forcing path* is a marked hypergraph P so that we can write: $V(P) = \{a_1, \dots, a_{L+k-2}, b_1, \dots, b_L\}$, $M(P) = \{a_1, \dots, a_{k-2}, a_{L+k-2}\}$ and $E(P) = \{e_1, \dots, e_L\}$ where $e_i = \{a_i, \dots, a_{i+k-2}, b_i\}$ for all $1 \leq i \leq L$. It is indeed a path, of length L , represented by the walk $\vec{W} = (e_1, \dots, e_L)$ for example. A 3-uniform forcing path is simply a nunchaku. See Figure I.8 for an illustration of the case $k = 4$.

Proposition I.3.15. *Any forcing path is a Maker win.*

Proof. With the same notations as in Definition I.3.14, Maker can pick $a_{k-1}, a_k, \dots, a_{L+k-4}$ successively, forcing Breaker to pick b_1, b_2, \dots, b_{L-2} respectively. Maker now picks a_{L+k-3} , threatening to complete an edge on her next go by picking either b_{L-1} or b_L , and Breaker cannot address both threats at once. ■

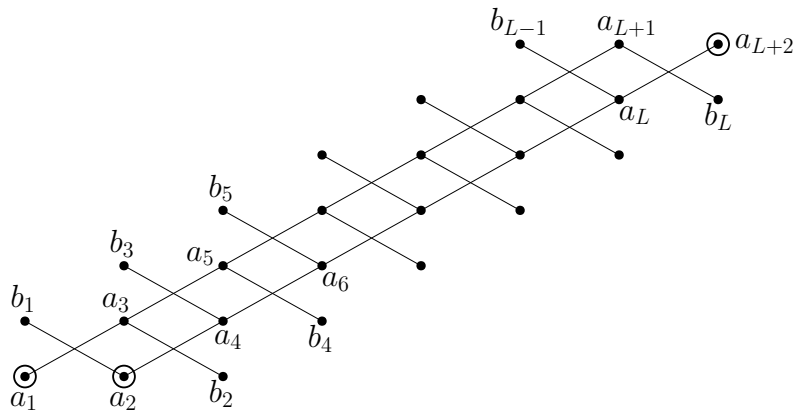


Figure I.8: A 4-uniform forcing path of length L (here $L = 12$). Each cross-brace joining 4 vertices represents an edge.

We will see in Subsection III.2.3 that, in the case $k \geq 4$, a k -uniform forcing path actually constitutes a slowest Maker win. However, our main focus by far will be on the case $k = 3$, with nunchakus being at the core of the structural study of the unbiased Maker-Breaker game on 3-uniform marked hypergraphs which will be the subject of Section II.1.

I.3.2 In 3-uniform hypergraphs

Before we can perform this structural study, we need to introduce some notions that are specific to the 3-uniform case.

I.3.2.1 Chains, cycles and tadpoles

Let us start by introducing the basic structures that we are going to consider in 3-uniform (marked) hypergraphs. The most central ones are chains, for which our definition coincides with that of [RW20]. An ab -chain is a 3-uniform linear ab -simple-path (note that, if a and b are the only marked vertices, then we exactly get an ab -nunchaku). To underline which properties this asks of the underlying walk, we also give the following definition which is equivalent:

Definition I.3.16. Let P be a hypergraph, and let $a, b \in V(P)$. We say P is an ab -chain if there exists a walk inducing P of the form $\vec{W} = (a, e_1, \dots, e_L, b)$ where:

- e_1, \dots, e_L are of size exactly 3;
- \vec{W} is linear;
- \vec{W} is simple.

Any walk \vec{W} that satisfies this definition or is equivalent to one that does is then said to *represent* P . An ab -chain may also be referred to as an a -chain if we desire to highlight just one end point, or a *chain* if we desire to highlight none. See Figure I.9.

Remark. Any ab -chain is also a ba -chain (take the reverse walk in the definition), an a -chain, a b -chain and a chain. Recall that Notation I.3.10 allows us to refer to the following useful walks that all represent an ab -chain P : \overrightarrow{aPb} , $\overrightarrow{bPa} = \overleftarrow{aPb}$, \overrightarrow{aP} , \overrightarrow{bP} .

Definition I.3.17. An a -snake is an ab -chain S of positive length for some marked vertex b . We may also refer to S as an ab -snake or simply a *snake*.

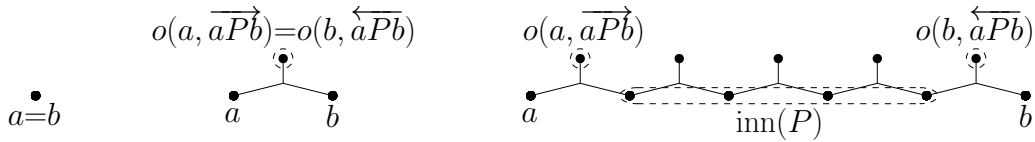


Figure I.9: An ab -chain P of length 0 (left), length 1 (middle), length 5 (right).

Remark. A snake might have more than one marked vertex. For example, a nunchaku is technically a snake.

Recall Definition I.1.6 as to inner and outer vertices in a general hypergraph. For chains, we introduce the following notation to designate specific outer vertices which we will often refer to in practice:

Notation I.3.18. Let P be an ab -chain of positive length. We denote by $o(a, \overrightarrow{aPb})$ the only vertex in $\text{start}(\overrightarrow{aPb}) \setminus (\text{inn}(P) \cup \{a, b\})$. See Figure I.9.

Next up, we introduce cycles. Our definition coincides with that of [Kut04] (for cycles of length at least 3) and [RW20].

Definition I.3.19. Let C be a hypergraph, and let $a \in V(C)$. We say C is an a -cycle if there exists a walk inducing C of the form $\overrightarrow{W} = (a, e_1, \dots, e_L, a)$ where:

- e_1, \dots, e_L are of size exactly 3;
- $L \geq 2$;
- if $L \geq 3$ then \overrightarrow{W} is linear, and if $L = 2$ then $|e_1 \cap e_2| = 2$;
- a is the only repeated vertex in \overrightarrow{W} , and $\{1 \leq i \leq L, a \in e_i\} = \{1, L\}$.

Any walk \overrightarrow{W} that satisfies this definition or is equivalent to one that does is then said to *represent* C . We say $L = |E(C)|$ is the *length* of C . An a -cycle may simply be referred to as a *cycle*. See Figure I.10.

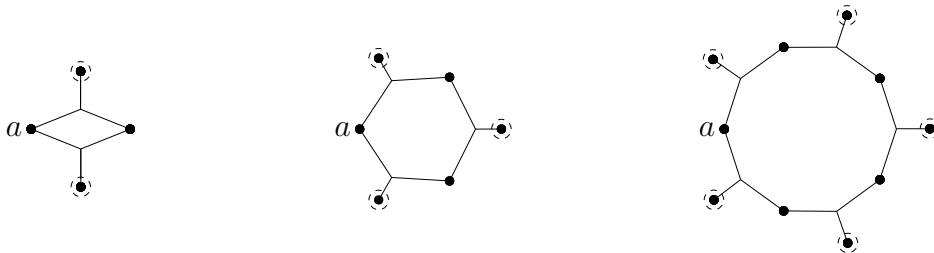


Figure I.10: An a -cycle C of length 2 (left), length 3 (middle), length 5 (right). The outer vertices are highlighted, the others are inner vertices.

Remark. Note that a cycle is a linear hypergraph except if it is of length 2.

Notation I.3.20. Let C be an a -cycle. For fixed a , there are exactly two walks satisfying Definition I.3.19: if the first one is written as (a, e_1, \dots, e_L, a) , then the second one is (a, e_L, \dots, e_1, a) . We denote the former by $\overrightarrow{(a - e_1)C}$ and the latter by $\overrightarrow{(a - e_L)C}$. When wishing to consider one of the two arbitrarily, we may use the notation \overrightarrow{aC} .

Remark. An a -cycle C is also a b -cycle for any $b \in \text{inn}(C)$ (note that $a \in \text{inn}(C)$ for instance), however it is not a b -cycle if $b \in \text{out}(C)$.

Definition I.3.21. An a -necklace is an a -cycle C such that $M(C) = \{a\}$. An a -necklace may simply be referred to as a *necklace*.

Definition I.3.22. A 3 -uniform hyperforest is a 3 -uniform hypergraph that contains no cycle.

Remark. Obviously, a 3 -uniform hypergraph is linear if and only if it contains no cycle of length 2 . In particular, 3 -uniform hyperforests are linear.

Finally, we introduce tadpoles, a less standard hypergraph structure which we have not seen defined anywhere else but that will also play a prominent part in our structural studies. This terminology is inspired from graph theory, in which a *tadpole graph* is defined as the union of a path and a cycle whose only shared vertex is one of the extremities of the path.

Definition I.3.23. Let T be a hypergraph, and let $a \in V(T)$. We say T is an a -tadpole if there exists a walk inducing T of the form $\overrightarrow{W} = (a, e_1, \dots, e_s, b, e_{s+1}, \dots, e_t, b)$ where:

- a and b are the only singletons;
- e_1, \dots, e_t are of size exactly 3 ;
- (a, e_1, \dots, e_s, b) represents an ab -chain P_T ;
- $(b, e_{s+1}, \dots, e_t, b)$ represents a b -cycle C_T ;
- $V(P_T) \cap V(C_T) = \{b\}$.

Any walk \overrightarrow{W} that satisfies this definition or is equivalent to one that does is then said to *represent* T . We may simply say T is a *tadpole*. The ab -chain P_T and the b -cycle C_T are clearly unique, so we may keep these notations. It is important to note that an a -cycle is a particular case of an a -tadpole, where $s = 0$ i.e. $a = b$. See Figure I.11.

Remark. In other words, an a -tadpole is the union, for some vertex b , of an ab -chain and a b -cycle whose only common vertex is b . Also note that a tadpole T is a linear hypergraph except if C_T is of length 2 .

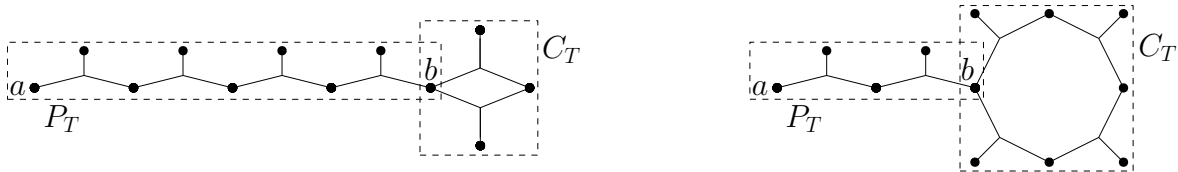


Figure I.11: An a -tadpole T (that is not an a -cycle), two examples.

Notation I.3.24. Let T be an a -tadpole. For fixed a , there are exactly two walks satisfying Definition I.3.23: if the first one is written as $(a, e_1, \dots, e_s, b, e_{s+1}, \dots, e_t, b)$, then the second one is $(a, e_1, \dots, e_s, b, e_t, e_{t-1}, \dots, e_{s+1}, b)$. The notation \overrightarrow{aT} refers to any of the two arbitrarily.

I.3.2.2 Substructure lemmas

We now address the existence, and sometimes unicity, of chains and tadpoles inside other chains and tadpoles. These results are easy and intuitive, but we give rigorous proofs using walks.

Lemma I.3.25. Let P be a chain and let $u, v \in V(P)$. Then there exists a unique uv -chain in P .

Proof. Let a, b such that P is an ab -chain, and write $\overrightarrow{aPb} = (a, e_1, \dots, e_L, b)$.

- Firstly, suppose $u = v$. Then that single vertex forms the only uv -chain in P .
- Secondly, suppose $u \neq v$ and there exists some $1 \leq i \leq L$ such that $\{u, v\} \subseteq e_i$ (note that i is unique since two distinct edges of a chain cannot intersect on two vertices). Then (u, e_i, v) represents a uv -chain. Moreover, if some walk $\overrightarrow{W'}$ represents a uv -chain in P , then we have $u \in \text{start}(\overrightarrow{W'})$ and $v \in \text{end}(\overrightarrow{W'})$, so $\text{start}(\overrightarrow{W'}) = \text{end}(\overrightarrow{W'}) = e_i$ hence the unicity.
- Finally, suppose $u \neq v$ and no edge of P contains both u and v . For $x \in \{u, v\}$, define $j(x) = \min\{1 \leq i \leq L, x \in e_i\}$ and $j'(x) = \max\{1 \leq i \leq L, x \in e_i\}$: note that $j'(x) = j(x) + 1$ if $x \in \text{inn}(P)$ and $j'(x) = j(x)$ otherwise. Up to swapping the roles of u and v , assume $j(u) \leq j(v)$: we actually have $j(u) < j(v)$, otherwise $e_{j(u)} = e_{j(v)}$ would contain both u and v . Since $j'(u) \in \{j(u), j(u) + 1\}$, this yields $j'(u) \leq j(v)$ hence $j'(u) < j(v)$ for the same reason. We claim that $\overrightarrow{W} := (u, e_{j'(u)}, e_{j'(u)+1}, \dots, e_{j(v)}, v)$ is a walk that represents a uv -chain. Indeed:

- The fact that \overrightarrow{aPb} is a linear walk by definition of a chain, coupled with the fact that $u \in e_{j'(u)}$ and $v \in e_{j(v)}$, implies that \overrightarrow{W} is a linear walk.
- The fact that the walk \overrightarrow{aPb} is simple by definition of a chain, coupled with the maximality of $j'(u)$ and the minimality of $j(v)$, implies that \overrightarrow{W} is also simple.

Let us now address the unicity. Let $\overrightarrow{W'} := (u, e_{i_1}, e_{i_2}, \dots, e_{i_t}, v)$ be a walk representing a uv -chain in P , where i_1, \dots, i_t are pairwise distinct indices in $\{1, \dots, L\}$. Since $u \in e_{i_1}$ and $v \in e_{i_t}$, we have $i_1 \in \{j(u), j'(u)\}$ and $i_t \in \{j(v), j'(v)\}$. We have seen that $j'(u) < j(v)$, so $i_1 < i_t$. For all $1 \leq s \leq t - 1$, we have $|e_{i_s} \cap e_{i_{s+1}}| = 1$ by definition of a chain hence $|i_s - i_{s+1}| = 1$. Since $i_1 < i_t$ and the indices i_1, \dots, i_t are pairwise distinct, this implies $i_{s+1} = i_s + 1$ for all $1 \leq s \leq t - 1$. To conclude that $\overrightarrow{W'} = \overrightarrow{W}$, it only remains to show that $i_1 = j'(u)$ and $i_t = j(v)$. We have mentioned that $i_1 \in \{j(u), j'(u)\}$: if $i_1 = j(u) = j'(u) - 1$, then $e_{i_2} = e_{j'(u)} \ni u$, hence a repetition in $\overrightarrow{W'}$ which contradicts the definition of a chain. Therefore $i_1 = j'(u)$, and an analogous reasoning yields $i_t = j(v)$. ■

We are also interested in the existence of chains inside cycles. First of all, we need to describe what happens when we delete a vertex from a cycle:

Lemma I.3.26. *Let C be a cycle and let $w \in V(C)$. Let w_1, w_2 be the two inner vertices of C that are adjacent to w in C (if C is of length 2 and $w \in \text{inn}(C)$ then $w_1 = w_2$).*

- *If $w \in \text{out}(C)$ then C^{-w} is a w_1w_2 -chain.*
- *If $w \in \text{inn}(C)$ then C^{-w} is the union of a w_1w_2 -chain and two isolated vertices which are the two outer vertices of C that are adjacent to w in C .*

Proof. Let us first address the case where C is of length 2. If $w \in \text{out}(C)$, then write $E(C) = \{\{w_1, w, w_2\}, \{w_1, u, w_2\}\}$: C^{-w} consists of the edge $\{w_1, u, w_2\}$, which forms a w_1w_2 -chain. If $w \in \text{inn}(C)$, then write $E(C) = \{\{w, u_1, w_1\}, \{w, u_2, w_1\}\}$: C^{-w} consist of the three isolated vertices $w_1 = w_2, u_1$ and u_2 .

Now assume that C is of length at least 3. Let e be the edge of C containing both w and w_1 , and write $\overrightarrow{(w_1 - e)C} = (w_1, e = e_1, e_2, \dots, e_L, w_1)$. We have $e_1 \cap e_L = \{w_1\}$. If $w \in \text{out}(C)$, then $e_1 = \{w_1, w, w_2\}$ so $e_1 \cap e_2 = \{w_2\}$. If $w \in \text{inn}(C)$, then $e_1 \cap e_2 = \{w\}$ hence $e_2 \cap e_3 = \{w_2\}$ since w_2 is adjacent to w . Therefore, defining $i = 2$ if $w \in \text{out}(C)$ and $i = 3$ if $w \in \text{inn}(C)$, the only edges of C containing w_2 are e_{i-1} and e_i . We claim that $\overrightarrow{W} := (w_2, e_i, \dots, e_L, w_1)$ is a linear simple walk. Indeed:

- By definition of a cycle, the walk $\overrightarrow{(w_1 - e)C}$ is linear since C is of length at least 3, and w_1 is its only repeated vertex with $\{1 \leq i \leq L, w_1 \in e_i\} = \{1, L\}$. Therefore, its subsequence (e_i, \dots, e_L, w_1) is also a linear walk, and has no repeated vertex since it does not contain

the edge e_1 .

- The addition of w_2 at the start of (e_i, \dots, e_L, w_1) preserves the fact that it is a linear walk since $w_2 \in e_i$, and also preserves the absence of any repeated vertex since $w_2 \notin e_j$ for all $j > i$.

Therefore, by definition, \overrightarrow{W} represents a w_2w_1 -chain. We can now conclude:

- If $w \in \text{out}(C)$, then $V(C^{-w}) = V(C) \setminus \{w\} = e_2 \cup \dots \cup e_L = V(\overrightarrow{W})$ and $E(C^{-w}) = E(C) \setminus \{e_1\} = \{e_2, \dots, e_L\} = E(\overrightarrow{W})$, so C^{-w} is the w_1w_2 -chain represented by \overrightarrow{W} .
- If $w \in \text{inn}(C)$, then let u_1 and u_2 be the outer vertices of C in e_1 and e_2 respectively: we have $V(C^{-w}) = V(C) \setminus \{w\} = (e_3 \cup \dots \cup e_L) \cup \{u_1, u_2\} = V(\overrightarrow{W}) \cup \{u_1, u_2\}$ and $E(C^{-w}) = E(C) \setminus \{e_1, e_2\} = \{e_3, \dots, e_L\} = E(\overrightarrow{W})$, so C^{-w} is the union of the w_1w_2 -chain represented by \overrightarrow{W} and the two isolated vertices u_1 and u_2 . ■

We can now conclude about the existence of chains between two given vertices of a cycle, first when trying to avoid a third vertex, then in general.

Lemma I.3.27. *Let C be a cycle and let $u, v, w \in V(C)$ with $w \neq u, v$. Then there exists a unique uv -chain in C that does not contain w , unless all the following hold: $w \in \text{inn}(C)$, $u \neq v$, and u or v is an outer vertex of C that is adjacent to w (in which case there exists none).*

Proof. First of all, note that a uv -chain in C that does not contain w is exactly a uv -chain in C^{-w} . Assume $u \neq v$, otherwise the result is trivial. If $w \in \text{out}(C)$, then C^{-w} is a chain according to Substructure Lemma I.3.26, which contains a unique uv -chain by Substructure Lemma I.3.25. Now assume $w \in \text{inn}(C)$: then C^{-w} is the union of a chain P and two isolated vertices u_1, u_2 that are the two outer vertices of C adjacent to w according to Substructure Lemma I.3.26. If $u \in \{u_1, u_2\}$ or $v \in \{u_1, u_2\}$, then there obviously cannot exist a uv -chain in C^{-w} . Otherwise $u, v \in V(P)$, so there exists a unique uv -chain in P (and in C^{-w} as a result) by Substructure Lemma I.3.25. ■

Lemma I.3.28. *Let C be a cycle and let $u, v \in V(C)$. Then there exists a uv -chain in C , unless C is of length 2 and $\text{out}(C) = \{u, v\}$.*

Proof. If C is of length 2 and $\text{out}(C) = \{u, v\}$, then there is no uv -chain in C , because $|e_u \cap e_v| = 2$ where e_u (resp. e_v) denotes the only edge of C containing u (resp. v). Otherwise, there exists $w \in \text{out}(C) \setminus \{u, v\}$: by Substructure Lemma I.3.27, there exists a unique uv -chain in C that does not contain w , so in particular C contains a uv -chain. ■

We now give analogous results for tadpoles.

Lemma I.3.29. *Let T be a tadpole and let $u, v, w \in V(T)$. If $w \in \text{out}(C_T) \setminus \{u, v\}$, then there exists a uv -chain in T that does not contain w .*

Proof. Note that $w \notin V(P_T)$, so that Substructure Lemma I.3.25 concludes if $u, v \in V(P_T)$. If $u, v \in V(C_T)$, then Substructure Lemma I.3.27 concludes. Therefore, assume $u \in V(P_T)$ and $v \in V(C_T)$. Let b be the only vertex in $V(P_T) \cap V(C_T)$. By Substructure Lemma I.3.25, there exists a ub -chain P_{ub} in P_T , that does not contain w since $w \notin V(P_T)$. By Substructure Lemma I.3.27, there exists a bv -chain P_{bv} in C_T that does not contain w . Since $V(P_{ub}) \cap V(P_{bv}) = \{b\}$, it is clear that $\overrightarrow{uP_{ub}b} \oplus \overrightarrow{bP_{bv}v}$ represents a uv -chain in T that does not contain w . ■

Lemma I.3.30. *Let T be a tadpole and let $u, v \in V(T)$. Then there exists a uv -chain in T , unless C_T is of length 2 and $\text{out}(C_T) = \{u, v\}$.*

Proof. If C_T is of length 2 and $\text{out}(C_T) = \{u, v\}$, then there is no uv -chain in T , because $|e_u \cap e_v| = 2$ where e_u (resp. e_v) denotes the only edge of T containing u (resp. v). Otherwise, there exists $w \in \text{out}(C) \setminus \{u, v\}$: by Substructure Lemma I.3.29, there exists a uv -chain in T that does not contain w , so in particular T contains a uv -chain. ■

On the subject of tadpoles, let us make one final remark:

Lemma I.3.31. *Let T be a tadpole and let $u \in V(T) \setminus \text{out}(C_T)$. Then T contains a u -tadpole.*

Proof. Let b be the only vertex in $V(P_T) \cap V(C_T)$. Since $u \notin \text{out}(C_T)$, we have $u \in \text{inn}(C_T)$ or $u \in V(P_T)$. If $u \in \text{inn}(C_T)$, then C_T is a u -cycle. If $u \in V(P_T)$, then there exists a ub -chain P_{ub} in P_T by Substructure Lemma I.3.25, so $\overrightarrow{uP_{ub}b} \oplus \overrightarrow{bC_T}$ represents a u -tadpole. ■

I.3.2.3 Projections

One of the most common tools that we will use is, inside a chain or a tadpole, to follow a subchain starting from some vertex u until reaching some vertex set Z , as made possible by the previous results:

Proposition I.3.32. *Let H be a hypergraph. Let K be a chain or a tadpole in H , let $u \in V(K)$, and let $Z \subseteq V(H)$ such that $Z \cap V(K) \neq \emptyset$. In the case where K is a tadpole with C_K of length 2 and $u \in \text{out}(C_K)$, also suppose that $Z \cap V(K) \neq \text{out}(C_K) \setminus \{u\}$. Then there exists a u -chain $\mathbf{P}_Z(u, K)$ in K such that:*

- If $u \in Z$, then $\mathbf{P}_Z(u, K)$ is of length 0.
- If $u \notin Z$, then $\mathbf{P}_Z(u, K)$ is of positive length and its only edge intersecting Z is $\text{end}(\overrightarrow{u\mathbf{P}_Z(u, K)})$, with $|\text{end}(\overrightarrow{u\mathbf{P}_Z(u, K)}) \cap Z| \in \{1, 2\}$.

See Figure I.12.

Proof. Let us start by showing the existence of $w \in Z \cap V(K)$ such that there exists a uw -chain in K . If K is a chain, then any $w \in Z \cap V(K)$ is suitable by Substructure Lemma I.3.25. If K is a tadpole, then any $w \in Z \cap V(K)$ is suitable by Substructure Lemma I.3.30, unless C_K is of length 2 and $u \in \text{out}(C_K)$ in which case we choose $w \in Z \cap V(K) \setminus (\text{out}(C_K) \setminus \{u\})$ as allowed by the assumption.

Let $w \in Z \cap V(K)$ minimizing the length of a shortest uw -chain in K , and let P be a shortest uw -chain in K . We claim that $\mathbf{P}_Z(u, K) := P$ has the desired properties. Clearly, P is of positive length if and only if $u \notin Z$. Assume $u \notin Z$. By definition, the walk $\overrightarrow{uP}|_Z$ only has one edge intersecting Z , which is $\text{end}(\overrightarrow{uP}|_Z)$, so in particular $|\text{end}(\overrightarrow{uP}|_Z) \cap Z| \in \{1, 2\}$. Therefore, it suffices to show that $\overrightarrow{uP}|_Z = \overrightarrow{u\mathbf{P}_Z(u, K)}$ to finish the proof. Let $w' \in \text{end}(\overrightarrow{u\mathbf{P}_Z(u, K)})$. The walk $\overrightarrow{u\mathbf{P}_Z(u, K)}$ induces a uw' -chain, which cannot be shorter than P by minimality of w , hence why $\overrightarrow{uP}|_Z = \overrightarrow{u\mathbf{P}_Z(u, K)}$. ■

Remark. There is not necessarily unicity, even if K is a chain: indeed, it is possible that there are vertices of Z on both sides of u in the chain.

Definition I.3.33. For K, u, Z satisfying the required conditions, a u -chain $\mathbf{P}_Z(u, K)$ from Proposition I.3.32 is called a *projection of u onto Z in K* . As there is no unicity in general, we will consider that the notation $\mathbf{P}_Z(u, K)$ always refers to the same chain for given K, u, Z . We normally use the notation once anyway, to give ourselves one arbitrary such projection and then work with that one.

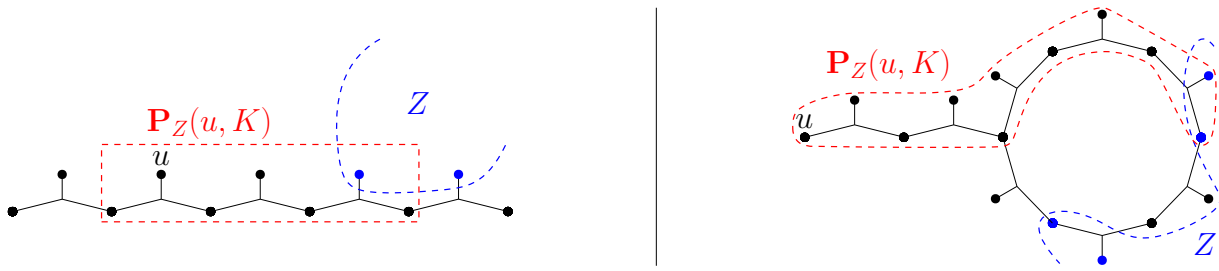


Figure I.12: Examples of projections. Left: K is a chain and $|\text{end}(\overrightarrow{uP_Z(u, K)}) \cap Z| = 1$. Right: K is a tadpole and $|\text{end}(\overrightarrow{uP_Z(u, K)}) \cap Z| = 2$.

I.3.2.4 Union lemmas

We now look at some structures that appear in unions of chains and tadpoles. The following three lemmas are immediately deduced from the concatenation of the walks representing the chains, cycles and tadpoles involved in their statements. We will use them often without necessarily referencing them.

Lemma I.3.34. *If P is an ab -chain and P' is a bc -chain such that $V(P) \cap V(P') = \{b\}$, then $P \cup P'$ is an ac -chain.*

Lemma I.3.35. *If P and P' are ab -chains such that $V(P) \cap V(P') = \{a, b\}$, then $P \cup P'$ is an a -cycle and a b -cycle.*

Lemma I.3.36. *If P is an ab -chain and T is a b -tadpole such that $V(P) \cap V(T) = \{b\}$, then $P \cup T$ is an a -tadpole.*

However, when the intersection of the two objects is more complex, it is less clear what their union contains. For example, Figure I.13 illustrates the fact that the union of an ab -chain and a bc -chain does not necessarily contain an ac -chain.

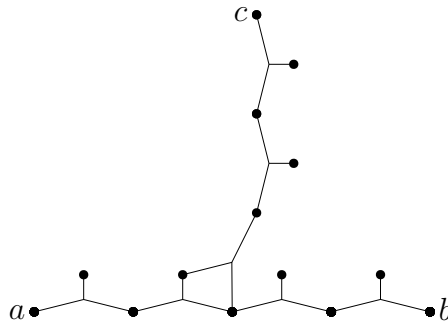
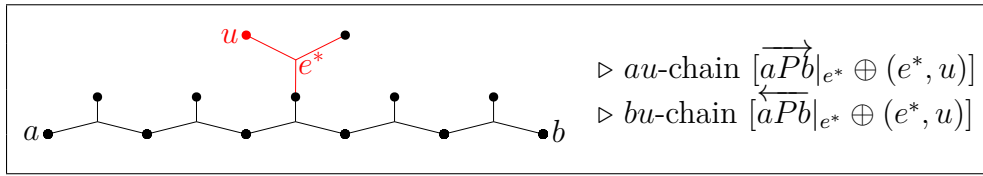


Figure I.13: The ac -path is not linear: there is an edge intersection of size 2.

Let us first consider the union of an ab -chain P of positive length and an edge e^* such that $e^* \cap V(P) \neq \emptyset$ and there exists $u \in e^* \setminus V(P)$. When is it possible to prolong a subchain of P with the edge e^* to get an au -chain and/or a bu -chain?

If $|e^* \cap V(P)| = 1$, then we get both an au -chain and a bu -chain, represented by the walks $\overrightarrow{aPb}|_{e^*} \oplus (e^*, u)$ and $\overleftarrow{aPb}|_{e^*} \oplus (e^*, u)$ respectively, as illustrated in Table I.2.

If $|e^* \cap V(P)| = 2$ though, then the walk $\overrightarrow{aPb}|_{e^*} \oplus (e^*, u)$ does not necessarily represent an au -chain (same for b). If $a \in e^*$ i.e. $\overrightarrow{aPb}|_{e^*} = (a)$, then it obviously does. But if $a \notin e^*$ i.e. $\overrightarrow{aPb}|_{e^*}$ represents a chain of positive length, then it does if and only if $|e^* \cap \text{end}(\overrightarrow{aPb}|_{e^*})| = 1$. We see a key notion appearing here:

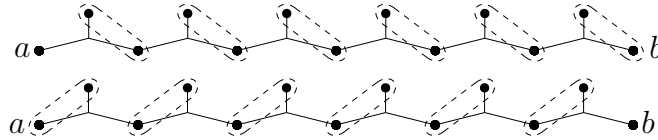
Table I.2: An edge e^* intersecting an ab -chain P on one vertex.

	$e^* \not\perp \overleftarrow{aPb}$	$e^* \perp \overleftarrow{aPb}$
$e^* \not\perp \overrightarrow{aPb}$	 $\triangleright au\text{-chain } [\overrightarrow{aPb}]_{e^*} \oplus (e^*, u)$ $\triangleright bu\text{-chain } [\overleftarrow{aPb}]_{e^*} \oplus (e^*, u)$	 $\triangleright au\text{-chain } [\overrightarrow{aPb}]_{e^*} \oplus (e^*, u)$ $\triangleright a\text{-tadpole } [\overrightarrow{aPb}]_{e^*} \oplus (e^*, \text{end}(\overleftarrow{aPb} _{e^*}))$
$e^* \perp \overrightarrow{aPb}$	 $\triangleright bu\text{-chain } [\overleftarrow{aPb}]_{e^*} \oplus (e^*, u)$ $\triangleright b\text{-tadpole } [\overleftarrow{aPb}]_{e^*} \oplus (e^*, \text{end}(\overrightarrow{aPb} _{e^*}))$	impossible

Table I.3: An edge e^* intersecting an ab -chain P on two vertices: all cases. The a -tadpole or b -tadpole, when one appears, is highlighted.

Notation I.3.37. Let P be an ab -chain of positive length and let e^* be an edge. Write $\overrightarrow{aPb} = (a, e_1, \dots, e_L, b)$. The notation $e^* \perp \overrightarrow{aPb}$ (or $e^* \perp \overrightarrow{aP}$ equivalently) means that either $e_1 \setminus \{a\} \subseteq e^*$ or $e_i \setminus e_{i-1} \subseteq e^*$ for some $2 \leq i \leq L$. See Figure I.14.

Remark. Note that it is technically possible to have both $e^* \perp \overrightarrow{aPb}$ and $e^* \perp \overleftarrow{aPb}$. This is the case if, for some j , we have $e^* = e_j$ or $e^* = \{o_j, o_{j+1}\} \cup (e_j \cap e_{j+1})$ where o_i denotes the only vertex in $e_i \setminus (\{a, b\} \cup \text{inn}(P))$. However, this will never happen for us, as in practice we will always have either $e^* \not\subseteq V(P)$ or $a \in e^*$.

Figure I.14: We have $e^* \perp \overrightarrow{aPb}$ (resp. $e^* \perp \overleftarrow{aPb}$) if and only if e^* contains one of the pairs of vertices highlighted at the top (resp. at the bottom).

In the case at hand $|e^* \cap V(P)| = 2$, we can see that $e^* \perp \overrightarrow{aPb}$ if and only if $a \notin e^*$ and $|e^* \cap \text{end}(\overrightarrow{aPb}|_{e^*})| = 2$. Therefore, the walk $\overrightarrow{aPb}|_{e^*} \oplus (e^*, u)$ represents an au -chain if and only if $e^* \not\perp \overrightarrow{aPb}$, and similarly the walk $\overleftarrow{aPb}|_{e^*} \oplus (e^*, u)$ represents a bu -chain if and only if $e^* \not\perp \overleftarrow{aPb}$. All of this is illustrated in Table I.3: note that, if no au -chain (resp. no bu -chain) appears, then we get a b -tadpole (resp. an a -tadpole).

Let us now consider the union of an ab -chain P of positive length and some edge $e^* \neq \text{start}(\overrightarrow{aPb})$ that intersects P on at least two vertices including a : do we get an a -cycle? If $|e^* \cap V(P)| = 2$ then the answer is yes, as illustrated in Table I.4. If $|e^* \cap V(P)| = 3$ then Table I.5 shows that it is possible that no a -cycle appears, in which case we get a b -tadpole.

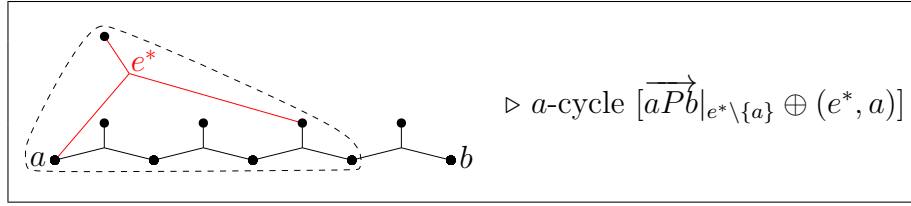


Table I.4: An edge e^* intersecting an ab -chain P on two vertices including a . The a -cycle is highlighted.

	$e^* \not\perp \overleftarrow{aPb}$	$e^* \perp \overleftarrow{aPb}$
$e^* \not\perp \overrightarrow{aPb}$	<p>$\triangleright a\text{-cycle } [\overrightarrow{aPb} _{e^* \setminus \{a\}} \oplus (e^*, a)]$</p>	<p>$\triangleright a\text{-cycle } [\overrightarrow{aPb} _{e^* \setminus \{a\}} \oplus (e^*, a)]$</p>
$e^* \perp \overrightarrow{aPb}$	<p>$\triangleright b\text{-tadpole } [\overleftarrow{aPb} _{e^*} \oplus (e^*, \text{end}(\overrightarrow{aPb} _{e^* \setminus \{a\}}))]$</p>	impossible

Table I.5: An edge e^* intersecting an ab -chain P on three vertices including a (and $e^* \neq \text{start}(\overrightarrow{aPb})$): all cases. The a -cycle or b -tadpole is highlighted.

Using these tables, we get the following four union lemmas, which are fundamental in our structural study of 3-uniform hypergraphs. They give us some basic information about the union of two chains or the union of a chain and a tadpole.

Lemma I.3.38. *Let a, b, c be distinct vertices. Let P_{ab} be an ab -chain, and let P_c be a c -chain such that $c \notin V(P_{ab})$ and $V(P_c) \cap V(P_{ab}) \neq \emptyset$. In particular, $e^* := \text{end}(\overrightarrow{cP_{V(P_{ab})}(c, P_c)})$ is well defined. Suppose there is no ca -chain in $P_{ab} \cup P_c$. Then $|e^* \cap V(P_{ab})| = 2$ and $e^* \perp \overrightarrow{aPb}$, moreover there is a cb -chain in $P_{ab} \cup P_c$ and a b -tadpole in $P_{ab} \cup e^* \subseteq P_{ab} \cup P_c$. See Figure I.15.*

Proof. By definition of a projection, we have $|e^* \cap V(P_{ab})| \in \{1, 2\}$. Let $u \in e^* \setminus V(P_{ab})$. All ways that e^* might intersect P_{ab} are summarized in Tables I.2 and I.3. There is no au -chain P_{au} in $P_{ab} \cup e^*$, otherwise the walk $\overrightarrow{cP_c|_{\{u\}}} \oplus \overrightarrow{uP_{au}a}$ would represent a ca -chain in $P_{ab} \cup P_c$, contradicting the assumption of the lemma. Therefore, we are necessarily in the bottom-left case of Table I.3, which means that: $|e^* \cap V(P_{ab})| = 2$, $e^* \perp \overrightarrow{aPb}$, there is a b -tadpole in $P_{ab} \cup e^*$, and there is a bu -chain P_{bu} in $P_{ab} \cup e^*$. The walk $\overrightarrow{cP_c|_{\{u\}}} \oplus \overrightarrow{uP_{bu}b}$ represents a cb -chain in $P_{ab} \cup P_c$. ■

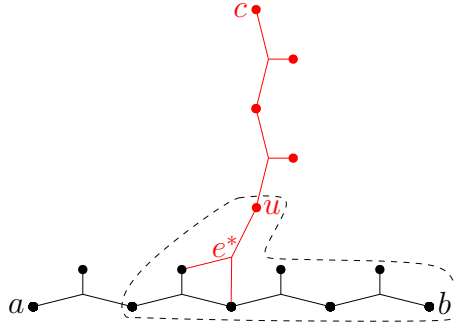


Figure I.15: Illustration of Lemma I.3.38. The represented chains are P_{ab} and $\mathbf{P}_{V(P_{ab})}(c, P_c)$. The b -tadpole is highlighted.

Lemma I.3.39. *Let a, b, c be distinct vertices, where b is marked. Let S_{ab} be an ab -snake, and let P_c be a c -chain such that $c \notin V(S_{ab})$ and $V(P_c) \cap V(S_{ab}) \neq \emptyset$.*

- *Suppose there is no c -snake in $S_{ab} \cup P_c$. Then there is both a ca -chain and an a -tadpole in $S_{ab} \cup P_c$.*
- *Suppose there is no ca -chain in $S_{ab} \cup P_c$. Then there is both a cb -snake and a b -tadpole in $S_{ab} \cup P_c$.*

Proof. The second item is exactly Lemma I.3.38. The first item is Lemma I.3.38 where the roles of a and b are reversed. ■

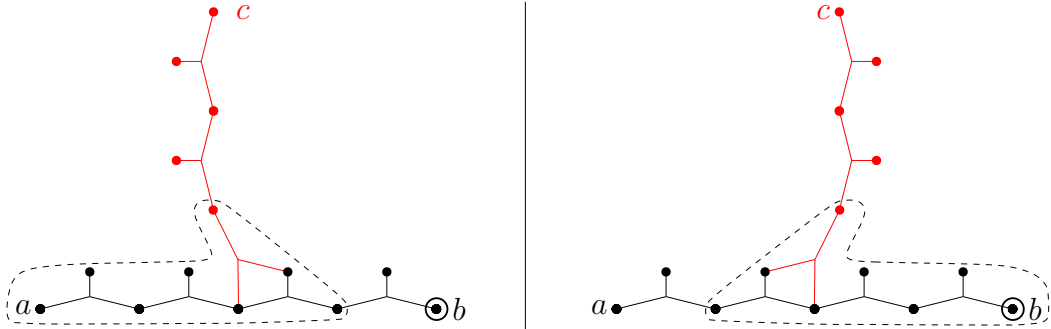


Figure I.16: Illustration of Lemma I.3.39 (first item on the left, second item on the right). The represented chains are S_{ab} and $\mathbf{P}_{V(S_{ab})}(c, P_c)$.

Lemma I.3.40. *Let a, b be distinct vertices. Let P_{ab} be an ab -chain, and let P_a be an a -chain such that $\text{start}(\overrightarrow{aP_a}) \neq \text{start}(\overrightarrow{aP_{ab}b})$ and $V(P_a) \cap (V(P_{ab}) \setminus \{a\}) \neq \emptyset$. In particular, $e^* := \text{end}(\overrightarrow{a\mathbf{P}_{V(P_{ab}) \setminus \{a\}}(a, P_a)})$ is well defined. Suppose there is no a -cycle in $P_{ab} \cup P_a$. Then $e^* \perp \overrightarrow{aP_b}$ and there is a b -tadpole in $P_{ab} \cup e^* \subseteq P_{ab} \cup P_a$. See Figure I.17.*

Proof. We distinguish between two cases:

- First suppose $a \in e^*$. Since $e^* \neq \text{start}(\overrightarrow{aP_{ab}b})$, all ways that e^* might intersect P_{ab} are summarized in Tables I.4 and I.5. Since there is no a -cycle in $P_{ab} \cup P_a \supseteq \overrightarrow{P_{ab} \cup e^*}$ by assumption, we are necessarily in the bottom-left case of Table I.5, so $e^* \perp \overrightarrow{aP_b}$ and there is a b -tadpole in $P_{ab} \cup e^*$.
- Now suppose $a \notin e^*$, meaning the projection $\mathbf{P}_{V(P_{ab}) \setminus \{a\}}(a, P_a)$ is of length at least 2. Write $\text{start}(\overrightarrow{aP_a}) = \{a, c, c'\}$ where $c \in \text{inn}(P_a)$, as in Figure I.17. Define the c -chain $P_c := P_a^{-a-c'}$: we have $c \notin V(P_{ab})$ and $e^* = \text{end}(\overrightarrow{c\mathbf{P}_{V(P_{ab})}(c, P_c)})$, so the idea is to apply Lemma I.3.38 to P_{ab} and P_c . If there was a ca -chain P_{ca} in $P_{ab} \cup P_c$, then $(a, \text{start}(\overrightarrow{aP_a}), c) \oplus \overrightarrow{cP_{ca}a}$ would

represent an a -cycle in $P_{ab} \cup P_a$, contradicting the assumption of the lemma. Therefore, there is no ca -chain in $P_{ab} \cup P_c$, so Lemma I.3.38 ensures that $e^* \perp \overrightarrow{aPb}$ and that there is a b -tadpole in $P_{ab} \cup e^*$. ■

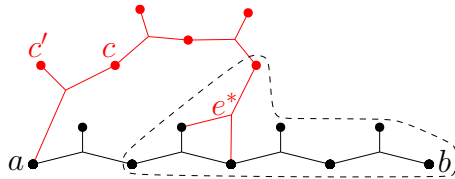


Figure I.17: Illustration of Lemma I.3.40. The represented chains are P_{ab} and $\mathbf{P}_{V(P_{ab}) \setminus \{a\}}(a, P_a)$. The b -tadpole is highlighted.

Lemma I.3.41. *Let a, c be distinct vertices. Let T be an a -tadpole, and let P_c be a c -chain such that $c \notin V(T)$ and $V(P_c) \cap V(T) \neq \emptyset$. In particular, $e^* := \text{end}(\overrightarrow{c\mathbf{P}_{V(T)}(c, P_c)})$ is well defined. Suppose there is no ca -chain in $T \cup P_c$. Then T is not a cycle, $|e^* \cap V(T)| = 2$ and $e^* \perp \overrightarrow{aP_T}$, moreover there is a c -tadpole in $T \cup P_c$. See Figure I.18.*

Proof. Up to replacing P_c with the projection $\mathbf{P}_{V(T)}(c, P_c)$, assume that e^* is the only edge of P_c intersecting T . Let b be the only vertex in $V(P_T) \cap V(C_T)$.

Claim 3. $e^* \cap (V(P_T) \setminus \{a\}) \neq \emptyset$.

Proof of Claim 3. We already know $a \notin e^*$, otherwise P_c would be a ca -chain, contradicting the assumption of the lemma. Therefore we must show that $e^* \cap V(P_T) \neq \emptyset$. Suppose for a contradiction that $e^* \cap V(P_T) = \emptyset$. There are two possibilities:

- Suppose $|e^* \cap V(C_T)| = 1$, and write $e^* \cap V(C_T) = \{v\}$. Note that P_c is a cv -chain. By Substructure Lemma I.3.28 (with $u = b$), there exists a bv -chain P_{bv} in C_T . The walk $\overrightarrow{cP_c v} \oplus \overrightarrow{vP_{bv} b} \oplus \overrightarrow{bP_T a}$ represents a ca -chain in $T \cup P_c$, contradicting the assumption of the lemma.
- Suppose $|e^* \cap V(C_T)| = 2$, and write $e^* \cap V(C_T) = \{v, w\}$. Note that P_c is both a cv -chain and a cw -chain. Up to swapping the roles of v and w , we can assume that $w \in \text{out}(C_T)$ or $v \in \text{inn}(C_T)$. Since $b \in \text{inn}(C_T)$ and $b \neq w$ (indeed $b \in V(P_T)$ whereas $e^* \cap V(P_T) = \emptyset$), Substructure Lemma I.3.27 (with $u = b$) thus ensures that there exists a bv -chain P_{bv} in C_T that does not contain w . The fact that $w \notin V(P_{bv})$ implies that $V(P_c) \cap V(P_{bv}) = \{v\}$. The walk $\overrightarrow{cP_c v} \oplus \overrightarrow{vP_{bv} b} \oplus \overrightarrow{bP_T a}$ thus represents a ca -chain in $T \cup P_c$, contradicting the assumption of the lemma. □

Claim 3 implies that $V(P_T) \setminus \{a\} \neq \emptyset$ i.e. P_T is of positive length i.e. T is not a cycle. It also implies that $V(P_c) \cap V(P_T) \neq \emptyset$, so we can apply Lemma I.3.38 with P_c and the ab -chain $P_{ab} = P_T$. Since there is no ca -chain in $T \cup P_c \supseteq P_T \cup P_c$ by assumption, Lemma I.3.38 tells us that: $|e^* \cap V(P_T)| = 2$, $e^* \perp \overrightarrow{aP_T b}$, and there is a cb -chain P_{cb} in $P_T \cup P_c$. Since $|e^* \cap V(P_T)| = 2$, we have $e^* \cap (V(C_T) \setminus \{b\}) = \emptyset$, hence $V(P_{cb}) \cap V(C_T) = \{b\}$. Therefore $P_{cb} \cup C_T$ is a c -tadpole in $T \cup P_c$, which concludes. ■

In linear hypergraphs, the above considerations are trivialized since the " \perp " cases do not exist. Therefore, we get simplified versions of the union lemmas in the linear case, the most important ones being the following two.

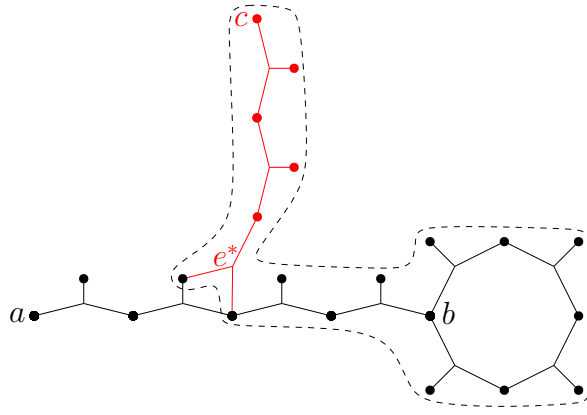


Figure I.18: Illustration of Lemma I.3.41. The represented objects are T and $\mathbf{P}_{V(T)}(c, P_c)$. The c -tadpole is highlighted.

Corollary I.3.42. *Let H be a linear hypergraph, and let $a, b, c \in V(H)$ be distinct. Let P_{ab} be an ab -chain in H , and let P_c be a c -chain in H such that $c \notin V(P_{ab})$ and $V(P_c) \cap V(P_{ab}) \neq \emptyset$. Then there is both a ca -chain and a cb -chain in $P_{ab} \cup P_c$.*

Proof. This is an immediate consequence of Lemma I.3.38. ■

Corollary I.3.43. *Let H be a linear hypergraph, and let $a, b \in V(H)$ be distinct. Let P_{ab} be an ab -chain in H , and let P_a be an a -chain in H such that $\text{start}(\overrightarrow{aP_a}) \neq \text{start}(\overrightarrow{aP_{ab}b})$ and $V(P_a) \cap (V(P_{ab}) \setminus \{a\}) \neq \emptyset$. Then there is an a -cycle in $P_{ab} \cup P_a$.*

Proof. This is an immediate consequence of Lemma I.3.40. ■

In 3-uniform hyperforests, which are a subcase of 3-uniform linear hypergraphs, another corollary of the union lemmas is the following natural property:

Corollary I.3.44. *In a 3-uniform hyperforest, there exists at most one ab -chain for given vertices a and b .*

Proof. Suppose for a contradiction that there exist distinct chains P and P' that are ab -chains for the same vertices a and b (in particular $a \neq b$, otherwise P and P' would both consist of the single vertex $a = b$). Choose P and P' such that the sum of the two lengths is minimum. Write $\overrightarrow{aPb} = (a, e_1, \dots, e_L, b)$ and $\overrightarrow{aP'b} = (a, e'_1, \dots, e'_M, b)$. If $e_1 \neq e'_1$ then $P \cup P'$ contains an a -cycle by Corollary I.3.43, contradicting the definition of a hyperforest. Therefore $e_1 = e'_1$, hence $L, M \geq 2$. Let c be the only vertex in $e_1 \cap e_2$, and let P_{cb} be the cb -chain represented by the walk (c, e_2, \dots, e_L, b) . There are two cases:

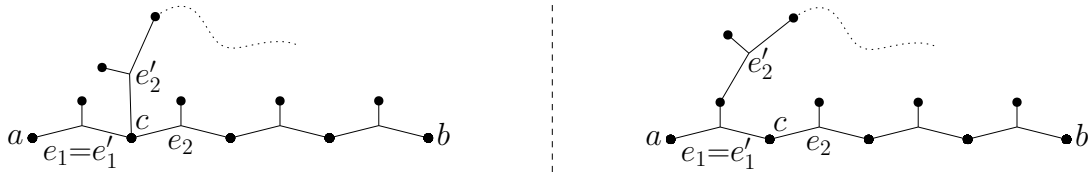


Figure I.19: Left: $e'_1 \cap e'_2 = \{c\}$. Right: $e'_1 \cap e'_2 \neq \{c\}$.

- If $e'_1 \cap e'_2 = \{c\}$ (Figure I.19, left), then $(c, e'_2, \dots, e'_M, b)$ represents a cb -chain P'_{cb} . Since $P \neq P'$ and $e_1 = e'_1$, we have $P_{cb} \neq P'_{cb}$. In conclusion, P_{cb} and P'_{cb} are distinct cb -chains, contradicting our choice of P and P' since the sum of lengths is $(L - 1) + (M - 1) < L + M$.

- If $e'_1 \cap e'_2 \neq \{c\}$ i.e. $c = o(a, \overrightarrow{aP'b})$ (Figure I.19, right), then P' is a cb -chain. Since $\text{start}(\overrightarrow{cP'b}) = e'_1 = e_1 \neq e_2 = \text{start}(\overrightarrow{cP_{cb}b})$, we have $P_{cb} \neq P'$. In conclusion, P_{cb} and P' are distinct cb -chains, contradicting our choice of P and P' since the sum of lengths is $(L - 1) + M < L + M$. ■

Chapter II

In-depth structural studies in hypergraphs

We are about to explore hypergraph structures in much detail to obtain the main results of this dissertation. This chapter is the most important, and also the most technical. It is important at this point to be familiar with all preliminary notions from Chapter I, which will be used extensively, as well as its (sometimes heavy) notations which will be necessary to carry out the proofs with the required care and precision. Section II.1 is dedicated to the unbiased Maker-Breaker game on hypergraphs of rank 3. The crucial role of chains in this game calls for a study of linear connectivity in 3-uniform hypergraphs: this is the subject of Section II.2, in a more general version which addresses $(k - 2)$ -linear connectivity in k -uniform hypergraphs.

II.1 The Maker-Breaker game: structural results in hypergraphs of rank 3[†]

II.1.1 Presentation of the problem and state of the art

The main goal of this section is to obtain a structural characterization of the outcome for the unbiased Maker-Breaker game on hypergraphs of rank 3, based on danger intersections. We would like to get an equivalence of type $(*)$ for this class i.e. to find an elementary family of 3-uniform dangers \mathcal{F} and a constant number of rounds r such that Breaker wins if and only if $J_r(\mathcal{F}, \cdot)$ holds. Let us talk about previous work on this specific class and why it gives us hope that such an equivalence does exist.

The first results on hypergraphs of rank 3 are due to Kutz [Kut04], who studied the linear case exclusively. He first reduces to the subclass \mathcal{H} of linear hypergraphs of rank 3 that: are connected, have no articulation vertex, and contain exactly one edge of size 2. He then gives a structural characterization of Breaker wins on the class \mathcal{H} , providing an exact description of their structure which we shall denote by Σ . The definition of the structure Σ is centered on the types of connections that exist in the hypergraph between the two vertices of the edge of size 2. Unfortunately, Kutz's approach seems very difficult to adapt to the non-linear case, where the possibilities multiply as to how linear objects may intersect, as we have seen with our union lemmas from Subsection I.3.2. Nevertheless, reading Kutz's proof that any hypergraph $H \in \mathcal{H}$ which does not have the structure Σ is a Maker win, some interesting observations can be made.

[†] These results are part of the article [GGS22b], which has been submitted to *Journal of Combinatorial Theory, Series A*.

The idea of danger intersections is present between the lines, although not truly identified by the author. Indeed, in all cases, the conclusion that H is a Maker win comes down to the existence of some vertex x and of some elementary subhypergraphs containing x with no common vertex apart from x itself and in which Breaker would be forced to answer if Maker picked x . It can thus be derived from Kutz's proof, and we will actually give an independent proof of it in this section, that a hypergraph $H \in \mathcal{H}$ is a Breaker win if and only if $J_1(\mathcal{D}_1, H)$ holds, where \mathcal{D}_1 is a very simple family of dangers. Even though it is not difficult to find counterexamples to this equivalence outside of the subclass \mathcal{H} (even linear ones), this new characterization has the potential to be adapted to general hypergraphs of rank 3, unlike the characterization via the structure Σ which is too specific to the linear case. This interpretation of Kutz's proof is the inspiration behind our danger-based approach of Maker-Breaker games.

More recently, the paper by Rahman and Watson [RW20] is just as relevant. They study a more general game, which is played on a CNF formula instead of a hypergraph. Two players take turns picking variables and assigning them a truth value of their choice: the first player (False) wants the formula to be false while the second player (True) wants the formula to be true. If the formula is positive i.e. all its literals are positive, then False always assigns the value 0 to the variable she picks, and True always assigns the value 1 to the variable he picks, so that the game is equivalent to the Maker-Breaker game: False is Maker, True is Breaker, and clauses correspond to edges. Rahman and Watson study the game played on a 3-CNF formula, meaning all clauses are of size at most 3, with the added constraint that each clause must possess a "spare variable" which appears in no other clause. This is a very strong constraint, and it may seem that this study is of no use to us, since for positive formulas it reduces to the linear case already settled by Kutz. Indeed, if there exist edges $e_1 = \{a, b, c\}$ and $e_2 = \{b, c, d\}$, then a and d appear in no other edge by the spare variable constraint, so item (ii) from Proposition I.1.22 ensures that these two edges may be replaced by a single edge $\{b, c\}$ without affecting the outcome. Nevertheless, Rahman and Watson's paper is still very interesting to us because of the methods that are used. Indeed, they define some "obstacles" for True, which are elementary subformulas on which False wins: again, this relates to our notion of danger. The main one is called a manriki, which in the positive case is none other than the non-marked reduction of a nunchaku, as mentioned before (recall Figure I.7). The authors show that, in all nontrivial cases, True wins if and only if he can break any manriki that appears during the first three rounds of play. They conjecture that this remains true for general 3-CNF formulas, without the spare variable constraint. For positive 3-CNF formulas, using our terminology on 3-uniform marked hypergraphs, this conjecture means that Breaker wins if and only if $J_3(\mathcal{D}_0, H)$ holds, where \mathcal{D}_0 is the family of dangers corresponding to nunchakus (i.e. the dangers (D, x) such that D^{+x} is a nunchaku or at least has the same non-marked reduction as a nunchaku). The main result of this section validates this conjecture and provides optimal strategies for both players based on danger intersections.

II.1.2 The game on 3-uniform marked hyperforests

Let us start with a simple subclass of 3-uniform marked hypergraphs, namely hyperforests.

II.1.2.1 Solution in terms of nunchakus

Being linear, 3-uniform marked hyperforests fall under Kutz's study. However, they admit the following criterion characterizing the winner of the Maker-Breaker game, which is much easier than the general result of Kutz:

Theorem II.1.1. *Let H be a 3-uniform marked hyperforest with no fully marked edge. Then H is a Maker win if and only if H contains a nunchaku.*

Proof. The "if" direction is already known: a nunchaku is a Maker win by Proposition I.3.13, so any H containing a nunchaku is a Maker win by Monotonicity Lemma I.1.50. Let us show the "only if" direction, by induction on $|V(H) \setminus M(H)|$. If H is a Maker win with $|V(H) \setminus M(H)| \leq 1$, then H is a trivial Maker win: since H contains no fully marked edge, this means H contains an edge with exactly two marked vertices i.e. a nunchaku of length 1. Now, assume $|V(H) \setminus M(H)| \geq 2$ and the implication to be true for all 3-uniform marked hyperforests with less non-marked vertices than H . We show the contrapositive: suppose that H contains no nunchaku. We want to show that Breaker can ensure that there is still no nunchaku after one round. Let $x \in V(H) \setminus V(H)$. Since H contains no nunchaku, all nunchakus in H^{+x} are x -nunchakus. Moreover, since H is linear, there cannot exist an xm_1 -chain and an xm_2 -chain in H for some distinct $m_1, m_2 \in M(H)$: indeed, by Corollary I.3.42, the union of the two would contain an m_1m_2 -nunchaku in H , a contradiction. Finally, since H is a hyperforest, there cannot exist two distinct xm -chains in H for some $m \in V(H)$ by Corollary I.3.44. In conclusion, H^{+x} contains at most one nunchaku. If that nunchaku exists, then it has at least one non-marked vertex y . Otherwise, let $y \in V(H^{+x}) \setminus M(H^{+x})$ be arbitrary. In both cases, H^{+x-y} contains no nunchaku and no fully marked edge, so it is a Maker win by the induction hypothesis. Therefore, H is a Maker win. ■

II.1.2.2 Interpretation in terms of the family of dangers \mathcal{S}

Notation II.1.2. We define the family \mathcal{S} of all pointed marked hypergraphs (S, x) such that S is an x -snake and $|M(S)| = 1$. See Figure II.1.

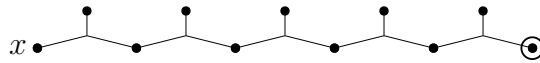


Figure II.1: An example of an \mathcal{S} -danger at x .

From Breaker's point of view, the fact that a nunchaku is a Maker win may be reformulated as follows:

Proposition II.1.3. *\mathcal{S} is a family of dangers.*

Proof. Let $(S, x) \in \mathcal{S}$: S^{+x} is a nunchaku, therefore it is a Maker win by Proposition I.3.13 i.e. (S, x) is a danger. ■

Moreover, we get the equivalence (*) for the class of 3-uniform marked hyperforests, with $\mathcal{F} = \mathcal{S}$ and $r = 1$:

Proposition II.1.4. *Let N be a nunchaku of length at least 2. Then $J_1(\mathcal{S}, N)$ does not hold.*

Proof. Since N is of length at least 2, we have $\text{inn}(N) \neq \emptyset$. Let $x \in \text{inn}(N)$: N is the union of two x -snakes S_1 and S_2 such that $V(S_1) \cap V(S_2) = \{x\}$. We have $S_1, S_2 \in x\mathcal{S}(N)$ hence $I_{N^{+x}}(x\mathcal{S}(N)) = \emptyset$, so $J_1(\mathcal{S}, N)$ does not hold. ■

Theorem II.1.5. *Let H be a 3-uniform marked hyperforest that is not a trivial Maker win, with $|V(H) \setminus M(H)| \geq 2$. Then H is a Breaker win if and only if $J_1(\mathcal{S}, H)$ holds.*

Proof. Recall that the "only if" direction is automatic by Proposition I.2.19. Suppose that H is a Maker win: since H is not a trivial Maker win, H has no fully marked edge, therefore Theorem II.1.1 ensures that H contains a nunchaku N . Again, since H is not a trivial Maker win, N is of length at least 2. By Proposition II.1.4, $J_1(\mathcal{S}, N)$ does not hold, so neither does $J_1(\mathcal{S}, H)$. ■

II.1.3 The game on general 3-uniform marked hypergraphs

In general 3-uniform marked hypergraphs, with cycles allowed, the equivalence from Theorem II.1.5 does not hold. For instance, there exist 3-uniform Maker wins that have no marked vertex (so they trivially contain no nunchakus). A famous example, which we have already mentioned, is the 3×3 tic-tac-toe hypergraph.

II.1.3.1 The family of dangers \mathcal{C}

An important observation is that nunchakus have a cycle counterpart: necklaces. Recall that a necklace is a cycle where one inner vertex is marked and all other vertices are non-marked. On an x -necklace, the same forcing technique used for nunchakus works (the only difference is that we come back to x at the end). Fundamentally, when Maker plays x , the x -snakes become x -nunchakus and the x -cycles become x -necklaces. Therefore, cycles are also dangers:

Notation II.1.6. We define the family \mathcal{C} of all pointed marked hypergraphs (C, x) such that C is an x -cycle and $M(C) = \emptyset$.

Proposition II.1.7. *A nunchaku and a necklace of same length have the same non-marked reduction.*

Proof. This is straightforward. See Figure II.2. ■

Proposition II.1.8. *\mathcal{C} is a family of dangers.*

Proof. Let $(C, x) \in \mathcal{C}$. Since C^{+x} is a necklace, it has the same non-marked reduction as a nunchaku, so the outcome for the game is the same for both according to Corollary I.1.54. Since a nunchaku is a Maker win, C^{+x} is also a Maker win i.e. (C, x) is a danger. ■

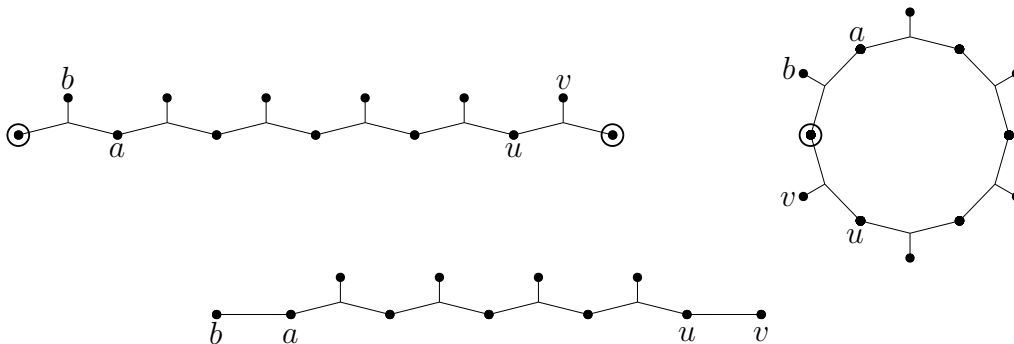


Figure II.2: Top: a nunchaku and a necklace, both of length 6. Bottom: their common non-marked reduction. Some vertices are named to help identification.

A necklace can also be seen as a union of two snakes:

Proposition II.1.9. *Let C be a necklace. Then $J_1(\mathcal{S}, C)$ does not hold.*

Proof. Let x be the only marked vertex of C and let $z \in \text{inn}(C) \setminus \{x\}$. Since x and z are distinct inner vertices of C , we can write $C = S_1 \cup S_2$ where $V(S_1) \cap V(S_2) = \{z, x\}$, as in Figure II.3. Since $M(C) = \{x\}$, we have $S_1, S_2 \in z\mathcal{S}(C)$ hence $I_{C+z}(z\mathcal{S}(C)) = \emptyset$, so $J_1(\mathcal{S}, C)$ does not hold. ■

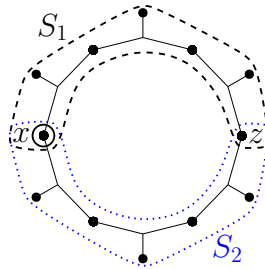


Figure II.3: An x -necklace C as a union of two z -snakes whose intersection in C^{+z} is empty.

Example. Consider the 3×3 tic-tac-toe hypergraph H . The \mathcal{C} -dangers help understand why and how Maker wins on H . Indeed, let u be the center vertex: it is not difficult to exhibit four u -cycles in H that have no common vertex apart from u (see Figure II.4, top). Therefore, $I_{H+u}(u\mathcal{C}(H)) = \emptyset$, so $J_1(\mathcal{C}, H)$ does not hold. Actually, the same is true for any corner vertex v (see Figure II.4, bottom). This means Maker can win by picking the center vertex or any corner vertex as her first move, and then force her way along whichever cycle has been left intact by Breaker's first move.

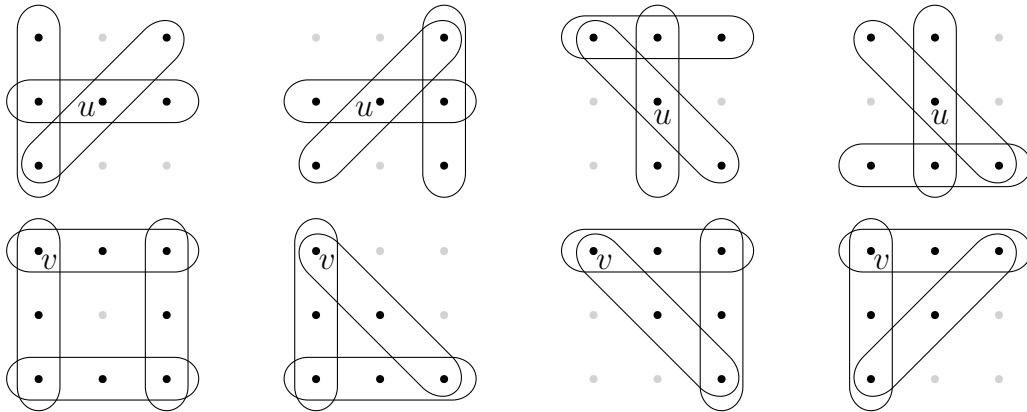


Figure II.4: Some cycles in the tic-tac-toe hypergraph. For clarity, we make an exception to our usual representation of edges in 3-uniform hypergraphs.

II.1.3.2 The family of dangers \mathcal{D}_0

All in all, the elementary family of dangers that we are going to consider comprises snakes and cycles:

Notation II.1.10. We define the family of dangers $\mathcal{D}_0 := \mathcal{S} \cup \mathcal{C}$. See Figure II.5.

Remark. Note that \mathcal{D}_0 contains the trivial danger of size 3, which is nothing but an \mathcal{S} -danger of length 1.

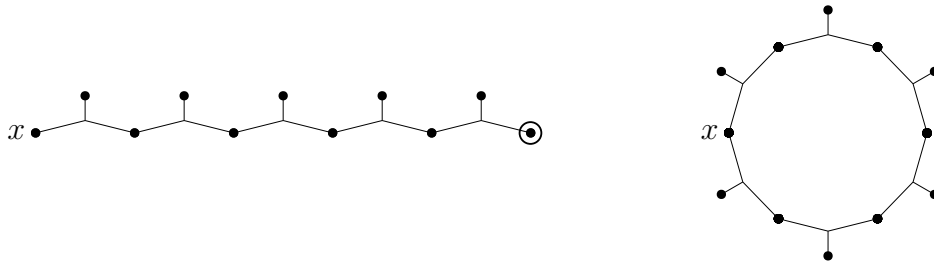


Figure II.5: Examples of \mathcal{D}_0 -dangers at x .

All the basic 3-uniform examples that we have looked at so far, namely hyperforests and the tic-tac-toe hypergraph, have the property that H is a Breaker win if and only if $J_1(\mathcal{D}_0, H)$ holds. Could this be true in general? Unfortunately, the answer is no. In fact, not only is $J_1(\mathcal{D}_0, H)$ not sufficient for H to be a Breaker win in general, but even $J_2(\mathcal{D}_0, H)$ is not. Figure II.6 (left) features an instance of a 3-uniform Maker win H such that it can be checked that $J_2(\mathcal{D}_0, H)$ holds but not $J_3(\mathcal{D}_0, H)$. Note that this counterexample is even linear.

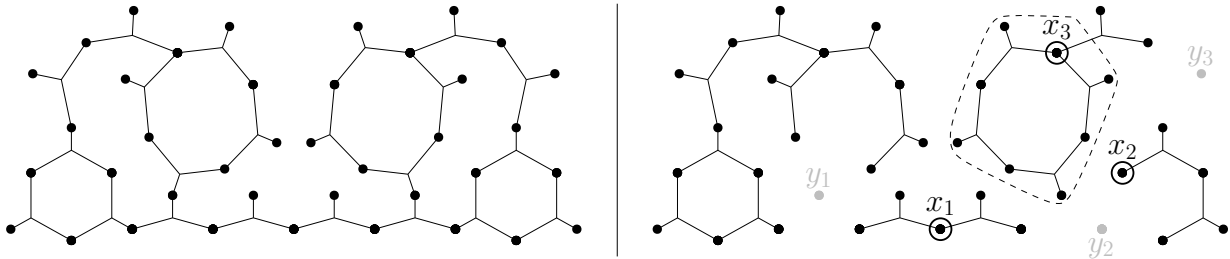


Figure II.6: H is on the left, $H^{+x_1-y_1+x_2-y_2+x_3-y_3}$ is on the right (the necklace is highlighted).

From Maker’s point of view, property $J_r(\mathcal{D}_0, H)$ not holding means that Maker can force the appearance of a nunchaku or a necklace after at most r rounds of play (we are talking about full rounds of play, i.e. the marked hypergraph updated after Breaker has played contains a nunchaku or a necklace):

Proposition II.1.11. *Let $r \geq 1$ be an integer. Let H be a 3-uniform marked hypergraph that is not a trivial Maker win, with $|V(H) \setminus M(H)| \geq 2r$. Then $J_r(\mathcal{D}_0, H)$ does not hold if and only if Maker has a strategy ensuring that, after r rounds of play on H with successive picks $x_1, y_1, \dots, x_r, y_r$, the updated marked hypergraph $H^{+x_1-y_1+\dots+x_r-y_r}$ contains a fully marked edge, a nunchaku or a necklace.*

(We make the harmless assumption that the players complete r rounds of play, even in the case where Maker has actually won during the first $r - 1$ rounds.)

Proof. Suppose $J_r(\mathcal{D}_0, H)$ holds. In particular $J_1(\mathcal{S}, H)$ holds, so H contains no necklace by Proposition II.1.9 and no nunchaku of length at least 2 by Proposition II.1.4. Since H is not a trivial Maker win, this means H contains no nunchaku at all (and no fully marked edge). When Maker picks x_i , the nunchakus and necklaces that he creates are exactly all the D^{+x_i} where D is a \mathcal{D}_0 -danger at x_i . By definition of $J_r(\mathcal{D}_0, H)$, Breaker is thus able, in each of the first r rounds, to destroy all the nunchakus and necklaces that Maker has just created. For the nunchakus of length 1, this means Maker never gets a fully marked edge.

Conversely, suppose $J_r(\mathcal{D}_0, H)$ does not hold. Then Maker can ensure that the updated hypergraph at the end of one of the first r rounds will contain a nunchaku or a necklace. If it happens before the r -th round, then Maker may for instance use the dichotomy strategy to get a nunchaku (or, eventually, a fully marked edge) at the end of each subsequent round as well,

until r rounds are played. ■

In the hypergraph from Figure II.6 (left), Maker needs exactly three rounds to guarantee the appearance of a nunchaku or a necklace: an example of the first three picks by both players is shown on the right. We will soon explain how we have come up with such a hypergraph for this example.

II.1.3.3 Statement of the main results

We have now introduced all concepts and notations needed to state our two main results about the Maker-Breaker game on 3-uniform marked hypergraphs, which we will prove in this section. As we have just seen, property $J_2(\mathcal{D}_0, H)$ is not equivalent to H being a Breaker win in general. However, a central result of this dissertation certifies that property $J_3(\mathcal{D}_0, H)$ is, proving the conjecture by Rahman and Watson [RW20] in the case of positive 3-CNF formulas. We can even give optimal strategies for both players based on the intersection of the \mathcal{D}_0^{*2} -dangers:

Theorem II.1.12. *Let H be a 3-uniform marked hypergraph that is not a trivial Maker win, with $|V(H) \setminus M(H)| \geq 6$. Then H is a Breaker win if and only if $J_3(\mathcal{D}_0, H)$ holds i.e. $J_1(\mathcal{D}_0^{*2}, H)$ holds. More precisely:*

- (i) *If $J_1(\mathcal{D}_0^{*2}, H)$ does not hold, then H is a Maker win and: any $x_1 \in V(H) \setminus M(H)$ such that $I_{H+x_1}(x_1\mathcal{D}_0^{*2}(H)) = \emptyset$ is a winning first pick for Maker.*
- (ii) *If $J_1(\mathcal{D}_0^{*2}, H)$ holds then H is a Breaker win and: for any first pick $x_1 \in V(H) \setminus M(H)$ of Maker, any $y_1 \in I_{H+x_1}(x_1\mathcal{D}_0^{*2}(H))$ is a winning answer for Breaker.*

Therefore, H is a Maker win if and only if Maker has a strategy ensuring that, after three rounds of play on H with successive picks $x_1, y_1, x_2, y_2, x_3, y_3$, the updated marked hypergraph $H^{+x_1-y_1+x_2-y_2+x_3-y_3}$ contains a fully marked edge, a nunchaku or a necklace.

We also exhibit a substantial subclass in which two rounds are sufficient instead of three, as our second main result:

Theorem II.1.13. *Let H be a 3-uniform marked hypergraph that is not a trivial Maker win, with $|V(H) \setminus M(H)| \geq 4$. Suppose that, for any $x \in V(H) \setminus M(H)$, there exists an x -snake in H . Then H is a Breaker win if and only if $J_2(\mathcal{D}_0, H)$ holds i.e. $J_1(\mathcal{D}_0^*, H)$ holds. Therefore, H is a Maker win if and only if Maker has a strategy ensuring that, after two rounds of play on H with successive picks x_1, y_1, x_2, y_2 , the updated marked hypergraph $H^{+x_1-y_1+x_2-y_2}$ contains a fully marked edge, a nunchaku or a necklace.*

II.1.4 Approximating \mathcal{D}_0^* and \mathcal{D}_0^{*2}

In order to tackle Theorem II.1.12, we can choose which property to consider between $J_3(\mathcal{D}_0, H)$ and $J_1(\mathcal{D}_0^{*2}, H)$, which are equivalent according to Proposition I.2.28. As explained in Subsection I.2.2, $J_1(\mathcal{D}_0^{*2}, H)$ is preferable as long as we get a reasonable understanding of the family \mathcal{D}_0^{*2} . In this subsection, we exhibit subfamilies $\mathcal{D}_1 \subseteq \mathcal{D}_0^*$ and $\mathcal{D}_2 \subseteq \mathcal{D}_0^{*2}$ which will be sufficient approximations, in the sense that Theorems II.1.12 and II.1.13 will actually hold in a stronger version where \mathcal{D}_0^{*2} and \mathcal{D}_0^* are replaced by their respective approximations \mathcal{D}_2 and \mathcal{D}_1 :

Theorem II.1.14. *Let H be a 3-uniform marked hypergraph that is not a trivial Maker win, with $|V(H) \setminus M(H)| \geq 2$. Then H is a Breaker win if and only if $J_1(\mathcal{D}_2, H)$ holds. More precisely:*

- (i) If $J_1(\mathcal{D}_2, H)$ does not hold, then H is a Maker win and: any $x_1 \in V(H) \setminus M(H)$ such that $I_{H+x_1}(x_1 \mathcal{D}_2(H)) = \emptyset$ is a winning first pick for Maker.
- (ii) If $J_1(\mathcal{D}_2, H)$ holds, then H is a Breaker win and: for any first pick $x_1 \in V(H) \setminus M(H)$ of Maker, any $y_1 \in I_{H+x_1}(x_1 \mathcal{D}_2(H))$ is a winning answer for Breaker.

Theorem II.1.15. *Let H be a 3-uniform marked hypergraph that is not a trivial Maker win, with $|V(H) \setminus M(H)| \geq 2$. Suppose that, for any $x \in V(H) \setminus M(H)$, there exists an x -snake in H . Then H is a Breaker win if and only if $J_1(\mathcal{D}_1, H)$ holds.*

Note that Theorem II.1.15 applies in particular to any 3-uniform marked hypergraph that is linear connected and has at least one marked vertex (i.e. its non-marked reduction has at least one edge of size 2). Therefore, Theorem II.1.15 contains the result on linear 3-uniform hypergraphs of which we mentioned in Subsection II.1.1 that it could be deduced from Kutz's proof [Kut04].

Let us now define \mathcal{D}_1 and \mathcal{D}_2 . Recall that $\mathcal{D}_0 \subseteq \mathcal{D}_0^* \subseteq \mathcal{D}_0^{*2}$. To build \mathcal{D}_1 from \mathcal{D}_0 , we are only going to add the most elementary new dangers that appear in the jump from \mathcal{D}_0 to \mathcal{D}_0^* , which are tadpoles.

II.1.4.1 The families of dangers \mathcal{T} and \mathcal{D}_1

Recall that $\mathcal{D}_0^* = \mathcal{D}_0 \cup \mathcal{D}_0^\heartsuit$ by definition. When looking for basic examples of \mathcal{D}_0^\heartsuit -dangers, we can see tadpoles appear:

Notation II.1.16. We define the family $\mathcal{T} \supseteq \mathcal{C}$ of all pointed marked hypergraphs (T, x) such that T is an x -tadpole and $M(T) = \emptyset$. We also define $\mathcal{D}_1 := \mathcal{D}_0 \cup \mathcal{T} = \mathcal{S} \cup \mathcal{T}$. See Figure II.7.

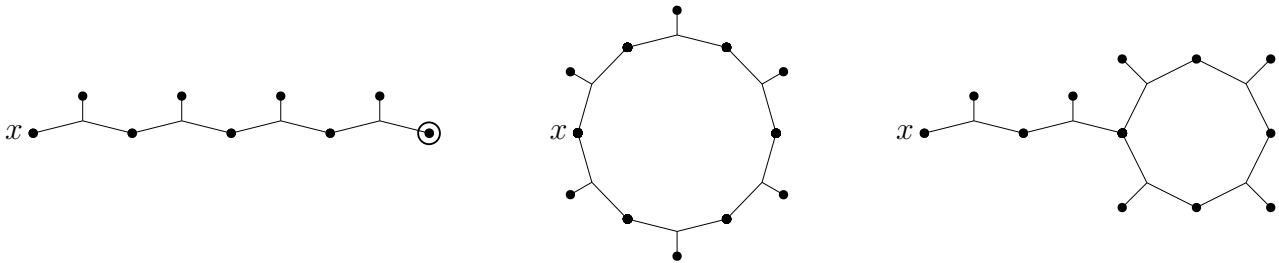


Figure II.7: Examples of \mathcal{D}_1 -dangers at x .

Proposition II.1.17. *We have $\mathcal{T} \setminus \mathcal{C} \subseteq \mathcal{D}_0^\heartsuit$. In particular: $\mathcal{D}_1 \subseteq \mathcal{D}_0^*$, so \mathcal{D}_1 is a family of dangers.*

Proof. Let $(T, x) \in \mathcal{T} \setminus \mathcal{C}$, and let z be the only vertex in $V(P_T) \cap V(C_T)$. Note that $z \neq x$ since T is not a cycle. We can write $T = \langle \mathcal{O} \rangle$ with $\mathcal{O} := \{P_T, C_T\}$. Since $V(P_T) \cap V(C_T) = \{z\}$, we have $I_{T+x+z}(\mathcal{O}) = \emptyset$. Moreover, since $M(P_T) = M(C_T) = \emptyset$, we have $P_T^{+x} \in z\mathcal{S}(T^{+x}) \subseteq z\mathcal{D}_0(T^{+x})$ and $C_T^{+x} = C_T \in z\mathcal{C}(T^{+x}) \subseteq z\mathcal{D}_0(T^{+x})$ i.e. $\mathcal{O} \subseteq \{K \subseteq T, K^{+x} \in z\mathcal{D}_0(T^{+x})\}$. See Figure II.8. In conclusion, we get $\mathcal{O} \in \mathcal{D}_0^\heartsuit(T^{+x+z})$.

This proves that $\mathcal{T} \setminus \mathcal{C} \subseteq \mathcal{D}_0^\heartsuit \subseteq \mathcal{D}_0^*$. Since $\mathcal{D}_0 \subseteq \mathcal{D}_0^*$, this yields $\mathcal{D}_1 \subseteq \mathcal{D}_0^*$. ■

Although the tadpoles from \mathcal{T} (resp. the snakes from \mathcal{S}) are required to have exactly 0 (resp. 1) marked vertex by definition, the next proposition ensures that in practice we will never have to worry about the number of marked vertices in a tadpole or a snake.

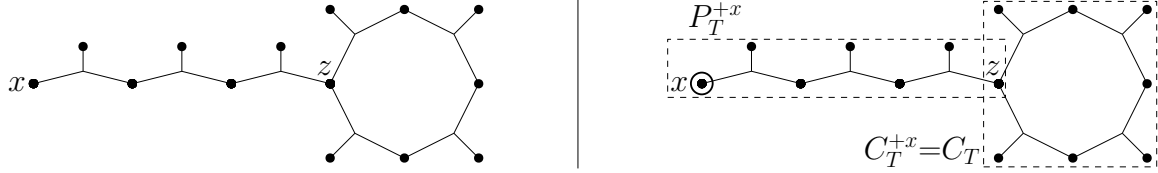


Figure II.8: Left: $(T, x) \in \mathcal{T} \setminus \mathcal{C}$. Right: T^{+x} is the union of two \mathcal{D}_0 -dangers at z whose intersection in T^{+x+z} is empty.

Proposition II.1.18. *Let H be a marked hypergraph that is not a trivial Maker win, and let $u \in V(H) \setminus M(H)$. Then any u -tadpole or u -snake in H contains a \mathcal{D}_1 -danger at u . In particular, if $u' \in I_{H+u}(u\mathcal{D}_1(H))$, then any u -tadpole or u -snake in H contains u' .*

Proof. Let K be a u -tadpole or u -snake in H . If $M(K) = \emptyset$, then K is necessarily a u -tadpole, and $K \in u\mathcal{T}(H) \subseteq u\mathcal{D}_1(H)$. Therefore, assume that $M(K) \neq \emptyset$, so that the chain $S := \mathbf{P}_{M(K)}(u, K) \subseteq K$ is well defined. By definition of a projection, the only edge of S that intersects $M(K)$ is $\text{end}(\overrightarrow{uS})$. Moreover, since H is not a trivial Maker win, that edge contains exactly one marked vertex hence $S \in u\mathcal{S}(H) \subseteq u\mathcal{D}_1(H)$.

The final assertion of this proposition ensues immediately: all u -snakes and u -tadpoles contain some $D \in u\mathcal{D}_1(H)$, and $u' \in V(D)$ since $u' \in I_{H+u}(u\mathcal{D}_1(H))$. ■

Let us also mention another useful property:

Proposition II.1.19. *Let H be a marked hypergraph that is not a trivial Maker win, with $|V(H) \setminus M(H)| \geq 2$, and suppose $J_1(\mathcal{D}_0, H)$ holds. Then, for any $m \in M(H)$, there is no m -tadpole and no m -snake in H .*

Proof. Suppose for a contradiction that there exists a subhypergraph K of H that is an m -tadpole or an m -snake for some marked vertex m . Let H_0 (resp. K_0) be the same as H (resp. K) except that m is non-marked. By Proposition II.1.18 applied to K_0 with $u = m$, there exists some $D \subseteq K_0$ such that $(D, m) \in \mathcal{D}_1$. We have $D^{+m} \subseteq K \subseteq H$.

- First suppose $(D, m) \in \mathcal{S}$ i.e. D^{+m} is a nunchaku. Since H is not a trivial Maker win, D^{+m} is of length at least 2, so $J_1(\mathcal{S}, D^{+m})$ does not hold according to Proposition II.1.4. Therefore, $J_1(\mathcal{S}, H)$ and $J_1(\mathcal{D}_0, H)$ do not hold either, a contradiction.
- Now suppose $(D, m) \in \mathcal{C}$. By Proposition II.1.9, $J_1(\mathcal{S}, D^{+m})$ does not hold. Therefore, $J_1(\mathcal{S}, H)$ and $J_1(\mathcal{D}_0, H)$ do not hold either, a contradiction.
- Finally, suppose $(D, m) \in \mathcal{T} \setminus \mathcal{C}$. By Proposition II.1.17, we have $(D, m) \in \mathcal{D}_0^\ominus$. Moreover, since D is a tadpole that is not a cycle, we have $|V(D)| \geq 6$ hence $|V(D^{+m}) \setminus M(D^{+m})| = |V(D)| - 1 \geq 5 \geq 2$, so Proposition I.2.24 ensures that $J_1(\mathcal{D}_0, D^{+m})$ does not hold. Therefore, $J_1(\mathcal{D}_0, H)$ does not hold either, a contradiction. ■

II.1.4.2 The families of dangers \mathcal{D}_1^\ominus and \mathcal{D}_2

We want to define $\mathcal{D}_2 \subseteq \mathcal{D}_0^{*2}$ such that property $J_1(\mathcal{D}_2, H)$ is sufficient for a 3-uniform marked hypergraph H to be a Breaker win. The idea is to prove this sufficiency result by induction, as follows:

1. Assume $J_1(\mathcal{D}_2, H)$ holds. Maker picks some x , Breaker picks some $y \in I_{H+x}(x\mathcal{D}_2(H))$.
2. If we have chosen $\mathcal{D}_2 \supseteq \mathcal{D}_1$ in such a way that $I_{H+x}(x\mathcal{D}_2(H)) \subseteq I_{H+x}(x\mathcal{D}_1^\ominus(H))$, then $y \in I_{H+x}(x\mathcal{D}_1^\ominus(H))$, therefore $J_1(\mathcal{D}_1, H^{+x-y})$ holds by Proposition I.2.25.
3. To complete the induction step, it remains to show that $J_1(\mathcal{D}_1, H^{+x-y})$ implies $J_1(\mathcal{D}_2, H^{+x-y})$, which will be the difficult part of the proof.

Therefore, for step 2 above, we must define $\mathcal{D}_2 \supseteq \mathcal{D}_1$ so that destroying the \mathcal{D}_2 -dangers destroys the \mathcal{D}_1^\ominus -dangers as well. This will force us to include (almost) all of \mathcal{D}_1^\ominus inside of \mathcal{D}_2 . As a consequence, we need to understand the structure of the \mathcal{D}_1^\ominus -dangers in all generality.

A \mathcal{D}_1^\ominus -danger at x is by definition a union of subhypergraphs having the common property that they will be \mathcal{D}_1 -dangers at some common z after x is marked. Which subhypergraphs have this property? The following result is elementary and answers this question: those not containing x are \mathcal{D}_1 -dangers at z already, while those containing x are zx -chains (which will become zx -snakes after x is marked).

Notation II.1.20. Let H be a marked hypergraph and $u, v \in V(H)$. We denote by $\mathcal{P}_{uv}(H)$ the set of all uv -chains P in H such that $M(P) = \emptyset$.

Proposition II.1.21. Let H be a marked hypergraph and let $x, z \in V(H) \setminus M(H)$ be distinct. We have $\{K \subseteq H, K^{+x} \in z\mathcal{D}_1(H^{+x})\} = z\mathcal{D}_1(H^{-x}) \cup \mathcal{P}_{zx}(H)$.

Proof. Let $K \subseteq H$ such that $K^{+x} \in z\mathcal{D}_1(H^{+x})$. There are two possibilities:

- Suppose $x \notin V(K)$: then $K = K^{+x}$ is a \mathcal{D}_1 -danger at z in H^{+x} , moreover $K \subseteq H^{-x}$ so $K \in z\mathcal{D}_1(H^{-x})$.
- Suppose $x \in V(K)$. By definition of \mathcal{D}_1 , the only \mathcal{D}_1 -dangers containing a marked vertex are the \mathcal{S} -dangers, and they contain exactly one marked vertex. Therefore K^{+x} is a zx -snake whose only marked vertex is x , so K is a zx -chain with no marked vertex i.e. $K \in \mathcal{P}_{zx}(H)$. ■

Remark. The notation $z\mathcal{D}_1(H^{-x})$ is just a compact way to refer to the collection of all \mathcal{D}_1 -dangers at z in H that do not contain x .

From this, we deduce the structural characterization of the \mathcal{D}_1^\ominus -dangers.

Notation II.1.22. Let D be a marked hypergraph and let $x, z \in V(D) \setminus M(D)$ be distinct. We define the collection $\mathcal{O}_{x,z}(D) := z\mathcal{D}_1(D^{-x}) \cup \mathcal{P}_{zx}(D) = z\mathcal{T}(D^{-x}) \cup z\mathcal{S}(D^{-x}) \cup \mathcal{P}_{zx}(D)$.

Proposition II.1.23. A pointed marked hypergraph (D, x) is in \mathcal{D}_1^\ominus , with \mathcal{D}_1 -dangerous vertex z , if and only if $D = \langle \mathcal{O}_{x,z}(D) \rangle$ and $I_{D^{+x+z}}(\mathcal{O}_{x,z}(D)) = \emptyset$.

Proof. By definition of the family \mathcal{D}_1^\ominus : a pointed marked hypergraph (D, x) is in \mathcal{D}_1^\ominus , with \mathcal{D}_1 -dangerous vertex z , if and only if $D = \langle \mathcal{O} \rangle$ for some $\mathcal{O} \subseteq \{K \subseteq D, K^{+x} \in z\mathcal{D}_1(D^{+x})\}$ such that $I_{D^{+x+z}}(\mathcal{O}) = \emptyset$. Moreover, we have $\{K \subseteq D, K^{+x} \in z\mathcal{D}_1(D^{+x})\} = \mathcal{O}_{x,z}(D)$ by Proposition II.1.21. Therefore: a pointed marked hypergraph (D, x) is in \mathcal{D}_1^\ominus , with \mathcal{D}_1 -dangerous vertex z , if and only if $D = \langle \mathcal{O} \rangle$ for some $\mathcal{O} \subseteq \mathcal{O}_{x,z}(D)$ such that $I_{D^{+x+z}}(\mathcal{O}) = \emptyset$. Finally, since $\mathcal{O}_{x,z}(D)$ is a collection of subhypergraphs of D , we always have $D \supseteq \langle \mathcal{O}_{x,z}(D) \rangle$, so saying that $D = \langle \mathcal{O} \rangle$ for some $\mathcal{O} \subseteq \mathcal{O}_{x,z}(D)$ is obviously equivalent to saying that $D = \langle \mathcal{O}_{x,z}(D) \rangle$. ■

Example. Figure II.9 features some examples of \mathcal{D}_1^\ominus -dangers. The middle one was actually used to build the example H from Figure II.6: it has been "duplicated" at $x = x_1$ so that $I_{H^{+x_1}}(x_1\mathcal{D}_1^\ominus(H)) = \emptyset$, ensuring that $J_1(\mathcal{D}_1^\ominus, H)$ does not hold from which $J_1(\mathcal{D}_0^{*2}, H)$ i.e. $J_3(\mathcal{D}_0, H)$ does not hold either.

Instead of defining \mathcal{D}_2 as $\mathcal{D}_1 \cup \mathcal{D}_1^\ominus = \mathcal{D}_1^*$, we have seen in Subsection I.2.2 (page 64) that we can actually avoid some redundancies by defining it as follows:

Notation II.1.24. We define $\mathcal{D}_2 := \mathcal{D}_1 \cup \mathcal{D}_1^{\ominus, \text{rest}}$.

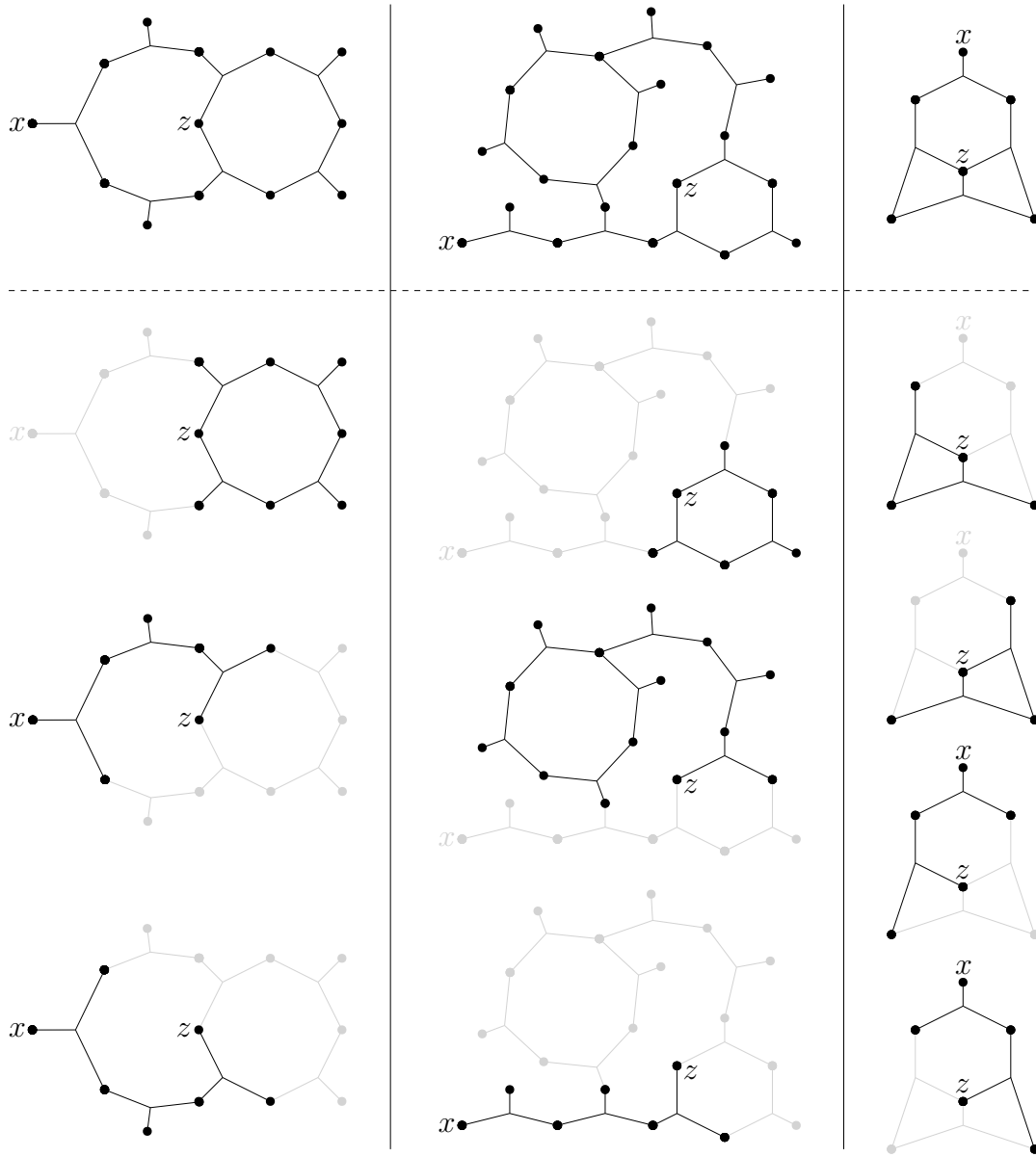


Figure II.9: Three examples of \mathcal{D}_1^ϕ -dangers (D, x) . Each one is the union of the subhypergraphs highlighted below it, which only intersect at z .

Indeed, destroying the \mathcal{D}_1 -dangers automatically destroys the $(\mathcal{D}_1^\phi \setminus \mathcal{D}_1^{\phi, \text{rest}})$ -dangers as well. As a concrete example, take $\mathcal{T} \setminus \mathcal{C}$ for instance: we have $\mathcal{T} \setminus \mathcal{C} \subseteq \mathcal{D}_0^\phi \subseteq \mathcal{D}_1^\phi$, however we already have $\mathcal{T} \setminus \mathcal{C} \subseteq \mathcal{D}_1$, so defining $\mathcal{D}_2 = \mathcal{D}_1 \cup \mathcal{D}_1^{\phi, \text{rest}}$ instead of $\mathcal{D}_2 = \mathcal{D}_1 \cup \mathcal{D}_1^\phi$ means we do not consider the $(\mathcal{T} \setminus \mathcal{C})$ -dangers twice in a way.

Proposition II.1.25. *We have $\mathcal{D}_2 \subseteq \mathcal{D}_0^{*2}$.*

Proof. We have $\mathcal{D}_2 = \mathcal{D}_1 \cup \mathcal{D}_1^{\phi, \text{rest}} \subseteq \mathcal{D}_1 \cup \mathcal{D}_1^\phi = \mathcal{D}_1^* \subseteq (\mathcal{D}_0^*)^* = \mathcal{D}_0^{*2}$. ■

II.1.5 Structure of the $\mathcal{D}_1^{\phi, \text{rest}}$ -dangers

The family $\mathcal{D}_2 = \mathcal{D}_1 \cup \mathcal{D}_1^{\phi, \text{rest}} \subseteq \mathcal{D}_0^{*2}$ will be our approximation for the pivotal family \mathcal{D}_0^{*2} . While the \mathcal{D}_1 -dangers are very basic objects, the $\mathcal{D}_1^{\phi, \text{rest}}$ -dangers must be studied as to better understand their shape and structural behavior. We have just given the general structure of

the elements of $\mathcal{D}_1^\mathfrak{O}$, but the restriction that defines the family $\mathcal{D}_1^{\mathfrak{O},\text{rest}}$ compared to the family $\mathcal{D}_1^\mathfrak{O}$ comes with added structural properties, which are the subject of this segment.

II.1.5.1 First properties

a) General properties

Proposition II.1.26. *Let $(D, x) \in \mathcal{D}_1^{\mathfrak{O},\text{rest}}$, with \mathcal{D}_1 -dangerous vertex z . We have the following properties:*

- (a) $D = \langle \mathcal{O}_{x,z}(D) \rangle = \langle z\mathcal{D}_1(D^{-x}) \cup \mathcal{P}_{zx}(D) \rangle$.
- (b) $I_{D+z}(\mathcal{O}_{x,z}(D)) = \emptyset$.
- (c) $z\mathcal{D}_1(D^{-x}) \neq \emptyset$.
- (d) $\mathcal{P}_{zx}(D) \neq \emptyset$.
- (e) D is not a trivial Maker win.
- (f) There is no x -tadpole and no x -snake in D .
- (g) There exists a z -cycle in D .

Proof. Let us start with items (a), (d) and (e), which actually hold for general $\mathcal{D}_1^\mathfrak{O}$ -dangers. Proposition II.1.23 gives us item (a). As for item (d), it is impossible that $\mathcal{P}_{zx}(D) = \emptyset$, because we would get $D = \langle z\mathcal{D}_1(D^{-x}) \rangle$, contradicting the fact that D contains x while the subhypergraphs in the collection $z\mathcal{D}_1(D^{-x})$ do not. Finally, the elements of the collection $z\mathcal{D}_1(D^{-x})$ have no edge with more than one marked vertex by definition of \mathcal{D}_1 , and the elements of the collection $\mathcal{P}_{zx}(D)$ have no marked vertex by definition, hence item (e).

We now check the remaining properties. Before this, using item (d), let $P_{zx} \in \mathcal{P}_{zx}(D)$ be shortest, and define $v := o(x, \overrightarrow{xP_{zx}z})$ and $w := o(z, \overleftarrow{xP_{zx}z})$: this chain will be useful. Note that $M(P_{zx}) = \emptyset$ by definition of the collection $\mathcal{P}_{zx}(D)$.

- Item (f) is straightforward. Since D is not a trivial Maker win by item (e), Proposition II.1.18 ensures that if there was an x -tadpole or an x -snake in D then D would contain a \mathcal{D}_1 -danger at x , contradicting the definition of the restricted family $\mathcal{D}_1^{\mathfrak{O},\text{rest}}$.
- Let us prove item (c). Suppose for a contradiction that $z\mathcal{D}_1(D^{-x}) = \emptyset$ hence $D = \langle \mathcal{P}_{zx}(D) \rangle$. We are going to use the chain P_{zx} . We know $I_{D+x+z}(\mathcal{O}_{x,z}(D)) = \emptyset$ by Proposition II.1.23, so $v \notin I_{D+x+z}(\mathcal{O}_{x,z}(D))$. Since $v \notin M(D^{+x+z})$, this means some element of the collection $\mathcal{O}_{x,z}(D) = \mathcal{P}_{zx}(D)$ does not contain v : let $P^{\bar{v}} \in \mathcal{P}_{zx}(D)$ such that $v \notin V(P^{\bar{v}})$. We have $\text{start}(\overrightarrow{xP^{\bar{v}}z}) \neq \text{start}(\overrightarrow{xP_{zx}z}) \ni v$ and $V(P^{\bar{v}}) \cap (V(P_{zx}) \setminus \{x\}) \supseteq \{z\} \neq \emptyset$, so we can apply Union Lemma I.3.40: since $P_{zx} \cup P^{\bar{v}} \subseteq D$ contains no x -cycle by item (f), it contains a z -tadpole T . If $x \notin V(T)$ as on the left of Figure II.10, then $T \in z\mathcal{T}(D^{-x})$, contradicting the fact that $z\mathcal{D}_1(D^{-x}) = \emptyset$. If $x \in V(T)$, then the only possibility is that the projection $\mathbf{P}_{V(P_{zx}) \setminus \{x\}}(x, P^{\bar{v}})$ consists of a single edge e as illustrated on the right of Figure II.10: since $v \notin e$, we get a zx -chain P'_{zx} that is strictly shorter than P_{zx} , a contradiction.

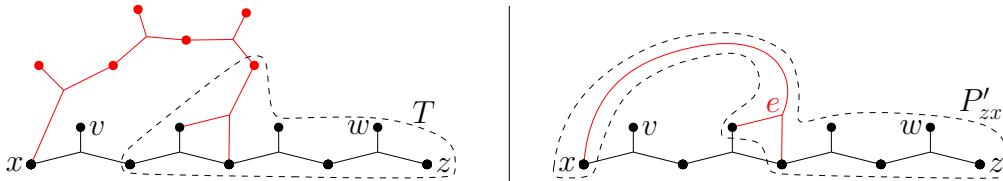


Figure II.10: The contradiction that yields item (c), if $x \notin V(T)$ (left) or if $x \in V(T)$ (right). The represented chains are P_{zx} (bottom) and $\mathbf{P}_{V(P_{zx}) \setminus \{x\}}(x, P^{\bar{v}})$.

- Item (b) directly ensues from item (c). Indeed, we already know that $I_{D+x+z}(\mathcal{O}_{x,z}(D)) = \emptyset$ by Proposition II.1.23, hence $I_{D+z}(\mathcal{O}_{x,z}(D)) \subseteq \{x\}$. Since the collection $z\mathcal{D}_1(D^{-x})$ is nonempty and none of its elements contain x , we have $x \notin I_{D+z}(z\mathcal{D}_1(D^{-x})) \supseteq I_{D+z}(\mathcal{O}_{x,z}(D))$, so in conclusion $I_{D+z}(\mathcal{O}_{x,z}(D)) = \emptyset$.
- Finally, let us prove item (g). We are going to use the chain P_{zx} again. We know $I_{D+z}(\mathcal{O}_{x,z}(D)) = \emptyset$ by item (b), so $w \notin I_{D+z}(\mathcal{O}_{x,z}(D))$. Since $w \notin M(D^{+z})$, this means some element of the collection $\mathcal{O}_{x,z}(D)$ does not contain w : let $K^{\bar{w}} \in \mathcal{O}_{x,z}(D)$ such that $w \notin V(K^{\bar{w}})$. By definition $\mathcal{O}_{x,z}(D) = z\mathcal{T}(D^{-x}) \cup z\mathcal{S}(D^{-x}) \cup \mathcal{P}_{zx}(D)$, so there are three possibilities for $K^{\bar{w}}$, and we claim that $V(K^{\bar{w}}) \cap (V(P_{zx}) \setminus \{z\}) \neq \emptyset$ for all of them:
 - If $K^{\bar{w}} \in \mathcal{P}_{zx}(D)$, then this is obvious because $x \in V(K^{\bar{w}})$.
 - If $K^{\bar{w}} \in z\mathcal{T}(D^{-x})$, then this is true because otherwise $P_{zx} \cup K^{\bar{w}}$ would be an x -tadpole in D , contradicting item (f).
 - If $K^{\bar{w}} \in z\mathcal{S}(D^{-x})$, then this is true because otherwise $P_{zx} \cup K^{\bar{w}}$ would be an x -snake in D , contradicting item (f).

Therefore, the projection $P := \mathbf{P}_{V(P_{zx}) \setminus \{z\}}(z, K^{\bar{w}})$ is well defined. Since $w \notin V(P)$ and $w \in \text{start}(\overleftarrow{xP_{zx}z})$, we have $\text{start}(z\overleftarrow{P}) \neq \text{start}(\overleftarrow{xP_{zx}z})$ so we can apply Union Lemma I.3.40: since $P_{zx} \cup P \subseteq D$ contains no x -tadpole by item (f), it contains a z -cycle. ■

The proofs of items (c) and (g) are typical of the methods that we will use extensively. The key is that, thanks to item (b), for any non-marked vertex $u \neq z$ there exists some element of $\mathcal{O}_{x,z}(D)$ that does not contain u . Therefore, item (b) is a powerful existence tool, providing us with subhypergraphs of D which we can use to partially reconstruct D and establish structural properties.

Beyond these basic characteristics, it is difficult to say much about the structure of $\mathcal{D}_1^{\mathfrak{O}, \text{rest}}$ -dangers in general. However, we now give additional properties that hold in all interesting cases.

b) Additional properties when $I_{H+z}(z\mathcal{D}_1(H)) \neq \emptyset$

In practice, we will always consider $\mathcal{D}_1^{\mathfrak{O}, \text{rest}}$ -dangers in some hypergraph H such that $J_1(\mathcal{D}_1, H)$ is satisfied. Given some $\mathcal{D}_1^{\mathfrak{O}, \text{rest}}$ -danger D at x in H , with z a \mathcal{D}_1 -dangerous vertex in (D, x) , this implies that $I_{H+z}(z\mathcal{D}_1(H)) \neq \emptyset$. In other words, even though the intersection in H^{+z} of $z\mathcal{D}_1(H) \cup \mathcal{P}_{zx}(H)$ is empty by Proposition II.1.26, the intersection in H^{+z} of $z\mathcal{D}_1(H)$ alone is not: it contains some s . This vertex s will often be useful.

Proposition II.1.27. *Let H be a marked hypergraph that is not a trivial Maker win. Let D be a $\mathcal{D}_1^{\mathfrak{O}, \text{rest}}$ -danger at some x in H , and let z be a \mathcal{D}_1 -dangerous vertex in (D, x) . Suppose $I_{H+z}(z\mathcal{D}_1(H)) \neq \emptyset$, and let $s \in I_{H+z}(z\mathcal{D}_1(H))$. Then:*

- Any z -tadpole or z -snake in H contains s .
- $s \in V(D) \setminus (M(D) \cup \{x, z\})$.
- There exists $P^{\bar{s}} \in \mathcal{P}_{zx}(D)$ such that $s \notin V(P^{\bar{s}})$. Moreover, the edges $\text{start}(\overrightarrow{xP^{\bar{s}}z})$ and $\text{end}(\overrightarrow{xP^{\bar{s}}z})$ are the same for any choice of $P^{\bar{s}}$.

Proof. We prove all three assertions separately:

- Since $s \in I_{H+z}(z\mathcal{D}_1(H))$ and H is not a trivial Maker win by assumption, Proposition II.1.18 applies with $u = z$ and $u' = s$, hence the first assertion.
- By definition of $I_{H+z}(\cdot)$, we have $s \notin M(H^{+z}) = M(H) \cup \{z\}$. Let $K \in z\mathcal{D}_1(D^{-x})$, which exists by Proposition II.1.26(c): since $z\mathcal{D}_1(D^{-x}) \subseteq z\mathcal{D}_1(H)$ and $s \in I_{H+z}(z\mathcal{D}_1(H))$, we have $s \in V(K) \subseteq V(D^{-x}) = V(D) \setminus \{x\}$. All in all, we get $s \in V(D) \setminus (M(D) \cup \{x, z\})$.

- Since $s \notin M(H^{+z})$, Proposition II.1.26(b) ensures the existence of some $K^{\bar{s}} \in \mathcal{O}_{x,z}(D)$ such that $s \notin V(K^{\bar{s}})$. Since $s \in I_{H^{+z}}(z\mathcal{D}_1(H)) \subseteq I_{H^{+z}}(z\mathcal{D}_1(D^{-x}))$, it is impossible that $K^{\bar{s}} \in z\mathcal{D}_1(D^{-x})$, so necessarily $K^{\bar{s}} \in \mathcal{P}_{zx}(D)$. Finally, let $P_1^{\bar{s}}, P_2^{\bar{s}} \in \mathcal{P}_{zx}(D)$ such that $s \notin V(P_1^{\bar{s}})$ and $s \notin V(P_2^{\bar{s}})$. Suppose for a contradiction that $\overrightarrow{\text{start}(xP_1^{\bar{s}}z)} \neq \overrightarrow{\text{start}(xP_2^{\bar{s}}z)}$: by Union Lemma I.3.40, $P_1^{\bar{s}} \cup P_2^{\bar{s}} \subseteq D$ contains an x -cycle (contradicting Proposition II.1.26(f)) or a z -tadpole (which does not contain s , also a contradiction). Similarly, suppose for a contradiction that $\overleftarrow{\text{end}(xP_1^{\bar{s}}z)} \neq \overleftarrow{\text{end}(xP_2^{\bar{s}}z)}$ i.e. $\overleftarrow{\text{start}(xP_1^{\bar{s}}z)} \neq \overleftarrow{\text{start}(xP_2^{\bar{s}}z)}$: by Union Lemma I.3.40, $P_1^{\bar{s}} \cup P_2^{\bar{s}} \subseteq D$ contains an x -tadpole (contradicting Proposition II.1.26(f)) or a z -cycle (which does not contain s , also a contradiction). ■

We now establish some important properties of the $\mathcal{D}_1^{\mathfrak{O},\text{rest}}$ -dangers in an ambient hypergraph H where $I_{H^{+z}}(z\mathcal{D}_1(H)) \neq \emptyset$, or sometimes under the stronger assumption that $J_1(\mathcal{D}_1, H)$ holds. We will also make the costless assumption that H is not a trivial Maker win, as we have already done in Proposition II.1.27.

II.1.5.2 Union lemmas

The next two lemmas are the analog for $\mathcal{D}_1^{\mathfrak{O},\text{rest}}$ -dangers of the union lemmas from Subsection I.3.2. We look at what happens in the union of a $\mathcal{D}_1^{\mathfrak{O},\text{rest}}$ -danger and a chain.

Lemma II.1.28. *Let H be a marked hypergraph that is not a trivial Maker win, and let $x \in V(H) \setminus M(H)$. Let D be a $\mathcal{D}_1^{\mathfrak{O},\text{rest}}$ -danger at x in H , with z a \mathcal{D}_1 -dangerous vertex in (D, x) . Let $c \in V(H) \setminus V(D)$, and let P_c be a c -chain such that $V(P_c) \cap V(D) \neq \emptyset$.*

- If $I_{H^{+z}}(z\mathcal{D}_1(H)) \neq \emptyset$, then there is a c -tadpole, a c -snake or a cx -chain in $D \cup P_c$.*
- If $J_1(\mathcal{D}_1, H)$ holds, then there is a c -tadpole or a cx -chain in $D \cup P_c$.*

Proof. First of all, we can assume that P_c consists of a single edge e . Indeed, let $e := \overrightarrow{\text{end}(c\mathbf{P}_{V(D)}(c, P_c))}$ and $c' \in e \setminus V(D)$:

- If there is a $c'x$ -chain P in $D \cup e$, then $\overrightarrow{cP_c|_{\{c'\}} \oplus c'Px}$ represents a cx -chain in $D \cup P_c$.
- If there is a c' -snake S in $D \cup e$, then $\overrightarrow{cP_c|_{\{c'\}} \oplus c'S}$ represents a c -snake in $D \cup P_c$.
- If there is a c' -tadpole T in $D \cup e$, then $\overrightarrow{cP_c|_{\{c'\}} \oplus c'T}$ represents a c -tadpole in $D \cup P_c$.

Therefore, we are working in $D \cup P_c = D \cup e$. Now suppose for a contradiction that:

$D \cup e$ contains no c -tadpole and no cx -chain, and also no c -snake in the case of item (i). (C)

Since $I_{H^{+z}}(z\mathcal{D}_1(H)) \neq \emptyset$ by assumption, let $s \in I_{H^{+z}}(z\mathcal{D}_1(H))$, and let $P^{\bar{s}} \in \mathcal{P}_{zx}(D)$ such that $s \notin V(P^{\bar{s}})$ as per Proposition II.1.27. Define $w := \overleftarrow{o(z, xP^{\bar{s}}z)}$. These notations are summed up in Figure II.11.

The key to the proof is the fact that every z -tadpole contains s , whereas $P^{\bar{s}}$ does not. For example, we can start by making a simple observation:

Claim 4. *Let P_{cz} be a cz -chain in $D \cup e$, and write $\overrightarrow{c\mathbf{P}_{V(P^{\bar{s}})}(c, P_{cz})} = (c, e_1, \dots, e_j)$. Then: $j > 1$, $e_j \perp xP^{\bar{s}}z$, and $e_{j-1} \cap e_j = \{s\}$. In particular, the cs -chain in P_{cz} is disjoint from $P^{\bar{s}}$.*

Proof of Claim 4. Since there is no cx -chain in $D \cup e$ by (C), we apply Union Lemma I.3.38 with $a = x$, $b = z$, $P_{ab} = P^{\bar{s}}$. Note that e_j is precisely the edge $e^* := \overrightarrow{\text{end}(c\mathbf{P}_{V(P^{\bar{s}})}(c, P_{cz}))}$ from Union Lemma I.3.38. We get that: $|e_j \cap V(P^{\bar{s}})| = 2$, $e_j \perp xP^{\bar{s}}z$, and there is a z -tadpole T in $P^{\bar{s}} \cup e_j$.

Since $|e_j \cap V(P^{\bar{s}})| = 2$, there is exactly one vertex of T that is not in $P^{\bar{s}}$. That vertex is

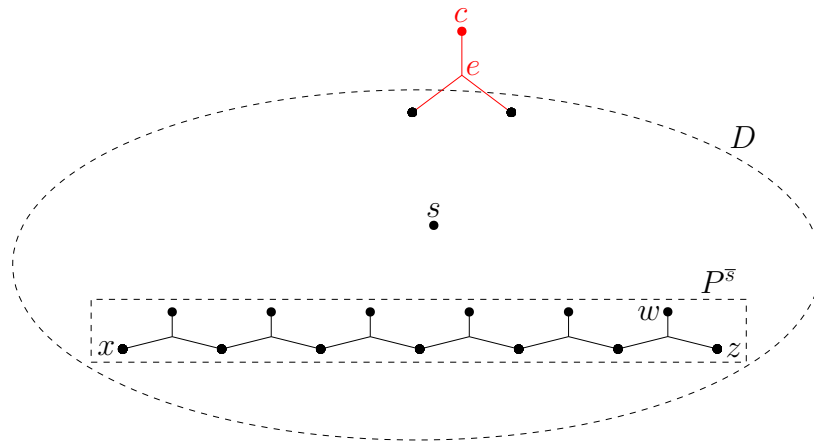


Figure II.11: D is only partially represented. In this picture we have $|e \cap V(D)| = 2$, but it is also possible that $|e \cap V(D)| = 1$.

necessarily s , as pictured on the left of Figure II.12: indeed, we know $s \in V(T)$ by definition of s , and $s \notin V(P^{\bar{s}})$ by definition of $P^{\bar{s}}$. In particular, since $c \neq s$ ($s \in V(D)$ whereas $c \notin V(D)$), we get $j > 1$. We know $(e_1 \cup \dots \cup e_{j-1}) \cap V(P^{\bar{s}}) = \emptyset$ by definition of a projection, therefore $e_{j-1} \cap e_j = \{s\}$ and (e_1, \dots, e_{j-1}) represents the unique cs -chain in P_{cz} , which is disjoint from $P^{\bar{s}}$. \square

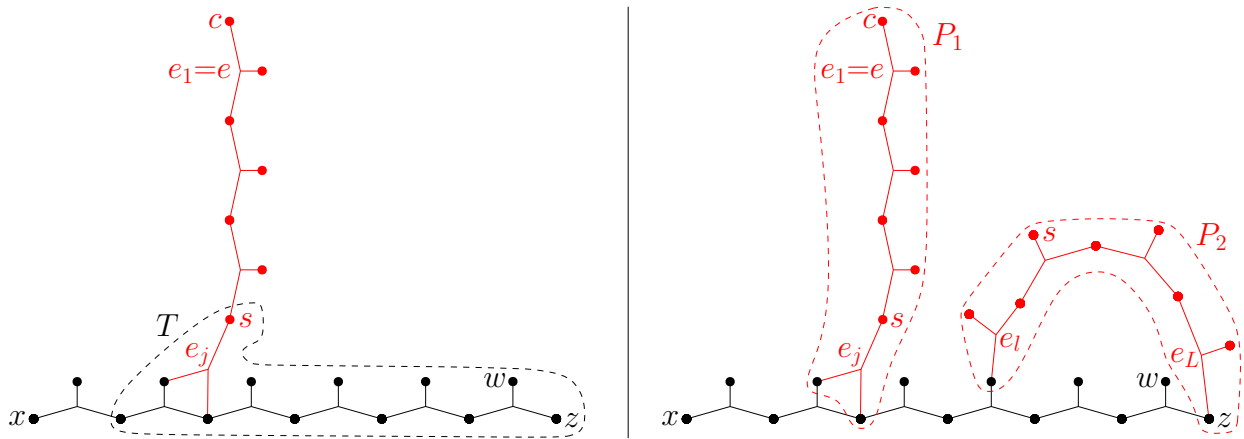


Figure II.12: Left: illustration of Claim 4. Right: the desired contradiction, with s having two different locations at once.

Therefore, the idea of the proof is the following, which is illustrated on the right of Figure II.12. We want to show that there exists a cz -chain $P_{cz}^{\bar{w}}$ in $D \cup e$ that does not contain w . Indeed, suppose we manage to exhibit one. On the one hand, following $P_{cz}^{\bar{w}}$ starting from c until touching $P^{\bar{s}}$, we get a chain P_1 which contains s as in Claim 4. On the other hand, following $P_{cz}^{\bar{w}}$ starting from z until touching $P^{\bar{s}}$ again, we get a chain P_2 which creates a z -cycle and thus must also contain s . This is a contradiction about the location of s . We now proceed with the proof, in three steps. We prove items (i) and (ii) jointly: there are only two times during the proof where we will have to differentiate the two very briefly to make separate arguments.

- 1) Firstly: we show there exists a cz -chain P_{cz} in $D \cup e$.
 Since $e \cap V(D) \neq \emptyset$, there exists $K \in \mathcal{O}_{x,z}(D)$ such that $e \cap V(K) \neq \emptyset$. By definition of $\mathcal{O}_{x,z}(D)$, there are three possibilities for K , and for each of them we can use an adequate union lemma from Subsection I.3.2:

- Suppose $K =: T \in z\mathcal{T}(D^{-x})$. Since there is no c -tadpole in $D \cup e \supseteq T \cup e$ by (C), Union Lemma I.3.41 ensures that there is a cz -chain in $T \cup e$.
- Suppose $K =: S \in z\mathcal{S}(D^{-x})$. We address items (i) and (ii) separately. For (i), there is no c -snake in $D \cup e \supseteq S \cup e$ by (C). For (ii), let m be the marked vertex such that S is a zm -snake: since $J_1(\mathcal{D}_1, H)$ holds, Proposition II.1.19 tells us there is no m -tadpole in $D \cup e \supseteq S \cup e$. In both cases, Union Lemma I.3.39 ensures that there is a cz -chain in $S \cup e$.
- Suppose $K =: P \in \mathcal{P}_{zx}(D)$. Since there is no cx -chain in $D \cup e \supseteq P \cup e$ by (C), Union Lemma I.3.38 ensures that there is a cz -chain in $P \cup e$.

In all cases, we get a cz -chain P_{cz} in $D \cup e$.

2) Secondly: we show there exists a cz -chain $P_{cz}^{\overline{w}}$ in $D \cup e$ that does not contain w .

Recall that, by Claim 4, P_{cz} contains a cs -chain P_{cs} such that $V(P_{cs}) \cap V(P^{\overline{s}}) = \emptyset$: in particular $w \notin V(P_{cs})$. Moreover, since w is non-marked (otherwise $P^{\overline{s}}$ would contain an x -snake, contradicting Proposition II.1.26(f)), Proposition II.1.26(b) ensures that there exists $K^{\overline{w}} \in \mathcal{O}_{x,z}(D)$ such that $w \notin V(K^{\overline{w}})$. We thus find $P_{cz}^{\overline{w}}$ inside $P_{cs} \cup K^{\overline{w}}$:

- Suppose $K^{\overline{w}} =: T \in z\mathcal{T}(D^{-x})$. In particular $s \in V(T)$, so $V(P_{cs}) \cap V(T) \neq \emptyset$. Since $D \cup e \supseteq P_{cs} \cup T$ does not contain a c -tadpole by (C), Union Lemma I.3.41 ensures that $P_{cs} \cup T$ contains a cz -chain.
- Suppose $K^{\overline{w}} =: S \in z\mathcal{S}(D^{-x})$. In particular $s \in V(S)$, so $V(P_{cs}) \cap V(S) \neq \emptyset$. For the second and last time in this proof, we address items (i) and (ii) separately. For (i), there is no c -snake in $D \cup e \supseteq P_{cs} \cup S$ by (C). For (ii), let m be the marked vertex such that S is a zm -snake: Proposition II.1.19 tells us there is no m -tadpole in $D \cup e \supseteq P_{cs} \cup S$. In both cases, Union Lemma I.3.39 ensures that there is a cz -chain in $P_{cs} \cup S$.
- Suppose $K^{\overline{w}} =: P \in \mathcal{P}_{zx}(D)$. Since $w \notin V(P)$, we have $w \notin \text{start}(\overleftarrow{xPz})$ hence $\text{start}(\overleftarrow{xPz}) \neq \text{start}(\overleftarrow{xP^{\overline{s}}z})$. By the final assertion of Proposition II.1.27, this implies $s \in V(P)$, so $V(P_{cs}) \cap V(P) \neq \emptyset$. Since $D \cup e \supseteq P_{cs} \cup P$ does not contain a cx -chain by (C), Union Lemma I.3.38 ensures that $P_{cs} \cup P$ contains a cz -chain.

In all cases, we get a cz -chain $P_{cz}^{\overline{w}}$ in $P_{cs} \cup K^{\overline{w}} \subseteq D \cup e$, that does not contain w since neither P_{cs} nor $K^{\overline{w}}$ does.

3) Finally: we conclude by getting the desired contradiction illustrated on the right of Figure II.12. We now work exclusively inside $P_{cz}^{\overline{w}} \cup P^{\overline{s}}$.

We start by defining the chains P_1 and P_2 pictured on the right of Figure II.12. Define the projection $P_1 := \mathbf{P}_{V(P^{\overline{s}})}(c, P_{cz}^{\overline{w}})$. By (C), it is impossible that $V(P_{cz}^{\overline{w}}) \cap V(P^{\overline{s}}) = \{z\}$, because $P_{cz}^{\overline{w}} \cup P^{\overline{s}}$ would then be a cx -chain. Therefore, the projection $P_2 := \mathbf{P}_{V(P^{\overline{s}}) \setminus \{z\}}(z, P_{cz}^{\overline{w}})$ is also well defined. Write $\overrightarrow{cP_{cz}^{\overline{w}}z} = (c, e_1, \dots, e_L, z)$, $\overrightarrow{cP_1} = (c, e_1, \dots, e_j)$, and $\overrightarrow{zP_2} = (z, e_L, e_{L-1}, \dots, e_l)$, i.e. $j = \min\{1 \leq i \leq L, e_i \cap V(P^{\overline{s}}) \neq \emptyset\}$ and $l = \max\{1 \leq i \leq L, e_i \cap (V(P^{\overline{s}}) \setminus \{z\}) \neq \emptyset\}$. Note that necessarily $e_1 = e$, since e is the only edge incident to c .

- First of all, we show that $1 < j < l$ and that $s \in e_{j-1}$. By Claim 4, we have: $j > 1$, $e_j \perp \overrightarrow{xP^{\overline{s}}z}$, and $e_{j-1} \cap e_j = \{s\}$. Moreover, since $w \notin V(P_{cz}^{\overline{w}})$, we have $w \notin e_j$: since $e_j \perp \overrightarrow{xP^{\overline{s}}z}$, this implies $z \notin e_j$. Therefore $j < L$, so we can consider the edge e_{j+1} . Since $s \in e_{j-1} \cap e_j$, we have $s \notin e_{j+1}$, so $e_j \cap e_{j+1} \subseteq e_j \setminus \{s\} \subseteq V(P^{\overline{s}}) \setminus \{z\}$: in particular $j < l$ by maximality of l .
- Finally, we show that $s \in e_i$ for some $l \leq i \leq L$ i.e. $s \in V(P_2)$. Note that $P_2 \subseteq D$: indeed, we have $P_2 \subseteq P_{cz}^{\overline{w}} \subseteq D \cup e$, and $e = e_1$ is not an edge of P_2 because $l \geq 2$.

Since $P_2 \subseteq P_{cz}^{\overline{w}}$ does not contain w , we have $\text{start}(z\overrightarrow{P_2}) \neq \text{start}(z\overrightarrow{P^s})$, so we can apply Union Lemma I.3.40. There is no x -tadpole in $P_2 \cup P^s \subseteq D$ by Proposition II.1.26(f), so we get a z -cycle C in $P_2 \cup P^s$. Since C must contain s , we have $s \in V(P_2) \cup V(P^s)$ hence $s \in V(P_2)$.

Since $j < l$, e_{j-1} is disjoint from e_l, \dots, e_L by definition of a chain. However, we have just shown that $s \in e_{j-1}$ and $s \in e_i$ for some $l \leq i \leq L$. This is a contradiction. \blacksquare

Lemma II.1.29. *Let H be a marked hypergraph that is not a trivial Maker win, and let $x \in V(H) \setminus M(H)$. Let D be a $\mathcal{D}_1^{\mathfrak{O}, \text{rest}}$ -danger at x in H , with z a \mathcal{D}_1 -dangerous vertex in (D, x) such that $I_{H+z}(z\mathcal{D}_1(H)) \neq \emptyset$. Then there is a unique edge e_x in D that is incident to x . Moreover, let P_x be an x -chain in H such that $V(P_x) \cap (V(D) \setminus \{x\}) \neq \emptyset$ and $\text{start}(x\overrightarrow{P_x}) \neq e_x$: then $D \cup P_x$ contains an x -snake or an x -tadpole.*

Proof. Let $s \in I_{H+z}(z\mathcal{D}_1(H))$, and let $P^s \in \mathcal{P}_{zx}(D)$ such that $s \notin V(P^s)$ as per Proposition II.1.27. We define $e_x := \text{start}(x\overrightarrow{P^s})$. We will show at the end of the proof that e_x is the unique edge of D containing x .

For now, let P_x be an x -chain in H such that $V(P_x) \cap (V(D) \setminus \{x\}) \neq \emptyset$ and $\text{start}(x\overrightarrow{P_x}) \neq e_x$. Up to replacing P_x with the projection $\mathbf{P}_{V(D) \setminus \{x\}}(x, P_x)$, assume that $\text{end}(x\overrightarrow{P_x})$ is the only edge of P_x that intersects $V(D) \setminus \{x\}$. Suppose for a contradiction that:

There is no x -snake and no x -tadpole in $D \cup P_x$. (C)

Let $e := \text{start}(x\overrightarrow{P_x})$. We distinguish between two cases.

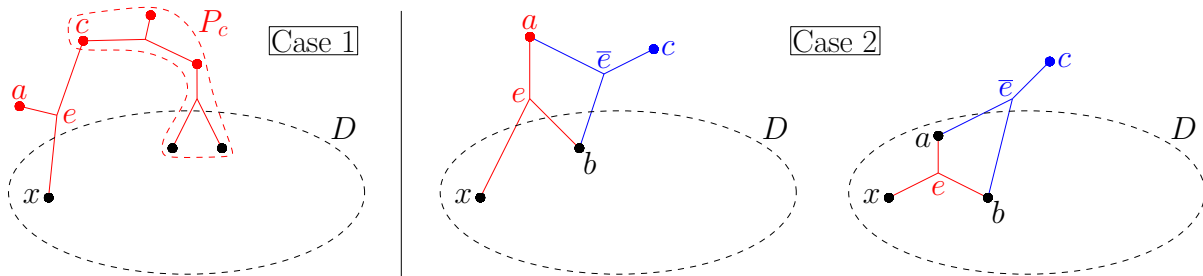


Figure II.13: Case 1: $e \cap V(D) = \{x\}$. Case 2: $|e \cap V(D)| = 2$ or $|e \cap V(D)| = 3$.

(1) Case 1: P_x is of length at least 2, i.e. $e \cap V(D) = \{x\}$.

Write $e = \{x, a, c\}$ where c is the only vertex in $\text{inn}(P_x) \cap e$, and let P_c be the c -chain defined as $P_c = P_x^{-x-a}$ (see Figure II.13). Since H is not a trivial Maker win and $I_{H+z}(z\mathcal{D}_1(H)) \neq \emptyset$, we can apply item (i) of Union Lemma II.1.28 in H to D and P_c , which tells us that $D \cup P_c$ contains one of the following:

- a cx -chain P . Then the walk $(x, e, c) \oplus \overrightarrow{cPx}$ represents an x -cycle in $D \cup P_x$.
- a c -tadpole T . If $x \in V(T)$, then T contains a cx -chain so we simply go back to that case. If $x \notin V(T)$, then the walk $(x, e, c) \oplus \overrightarrow{cT}$ represents an x -tadpole in $D \cup P_x$.
- a c -snake S . If $x \in V(S)$, then S contains a cx -chain so we simply go back to that case. If $x \notin V(S)$, then the walk $(x, e, c) \oplus \overrightarrow{cS}$ represents an x -snake in $D \cup P_x$.

All three possibilities contradict (C).

(2) Case 2: $P_x = e$ is of length 1, i.e. $|e \cap V(D)| \geq 2$.

Write $e = \{x, a, b\}$. As a gadget, we create a new non-marked vertex c and an edge $\bar{e} = \{a, b, c\}$ (see Figure II.13).

Claim 5. Let K be a subhypergraph of $D \cup \bar{e}$ such that $\bar{e} \in E(K)$ and $x \notin V(K)$, and define the subhypergraph $\varphi(K)$ of $D \cup e$ obtained from K by replacing c with x and \bar{e} with e . Then we have the isomorphisms of pointed marked hypergraphs: $(K, c) \sim (\varphi(K), x)$ and $(K, v) \sim (\varphi(K), v)$ for all $v \in V(K) \setminus (M(K) \cup \{c\})$.

Proof of Claim 5. This is straightforward. □

The idea is to apply Union Lemma II.1.28 in $D \cup \bar{e}$ to D and $P_c := \bar{e}$, and then contradict (C) through replacing \bar{e} with e in the obtained subhypergraph as per Claim 5. To do so, we need to check that $D \cup \bar{e}$ is not a trivial Maker win and that $I_{(D \cup \bar{e})+z}(z\mathcal{D}_1(D \cup \bar{e})) \neq \emptyset$. The former is clear: we know $D \subseteq H$ is not a trivial Maker win, moreover there is no x -snake in P_x by (C) so $M(e) = \emptyset$ hence $M(\bar{e}) = \emptyset$, so $D \cup \bar{e}$ is not a trivial Maker win either.

The latter is more difficult, because the addition of \bar{e} may create new \mathcal{D}_1 -dangers at z . However, we now show that they all contain s i.e. $s \in I_{(D \cup \bar{e})+z}(z\mathcal{D}_1(D \cup \bar{e}))$. Indeed, let K be a \mathcal{D}_1 -danger at z in $D \cup \bar{e}$: we want to show that $s \in V(K)$.

- Suppose $\bar{e} \notin E(K)$. Then $K \in z\mathcal{D}_1(H)$, hence $s \in V(K)$ by definition of s .
- Suppose $\bar{e} \in E(K)$ and $x \notin V(K)$. By Claim 5, we have $(K, z) \sim (\varphi(K), z)$, therefore $\varphi(K)$ is a \mathcal{D}_1 -danger at z in $D \cup e$ hence $s \in V(\varphi(K))$. Since $s \neq x$ by Proposition II.1.27, this yields $s \in V(K)$.
- Finally, suppose $\bar{e} \in E(K)$ and $x \in V(K)$. In particular, we have $c, x \in V(K)$.
 - If there exists a cx -chain P in K , then necessarily $\text{start}(\overrightarrow{cPx}) = \bar{e}$ since \bar{e} is the only edge incident to c in $D \cup \bar{e}$. Either a or b , say b , is an inner vertex of P , so that P^{-c-a} is a bx -chain in D that does not contain a . This means that $P^{-c-a} \cup e$ is an x -cycle in $D \cup e$, contradicting (C).
 - If there is no cx -chain in K , then the only possibility according to Substructure Lemmas I.3.25 and I.3.30 is that $K =: T$ is a z -tadpole such that C_T is of length 2 and $\text{out}(C_T) = \{c, x\}$ as in Figure II.14. Therefore, the edges incident respectively to c and x in T intersect on two vertices. Since the edge incident to c in T is necessarily $\bar{e} = \{a, b, c\}$, the edge incident to x in T is precisely $\{a, b, x\} = e$. Define the zx -chain $P := T^{-c}$, as in Figure II.14. Since $T \subseteq D \cup \bar{e}$, we have $P \subseteq D$ i.e. $P \in \mathcal{P}_{zx}(D)$. Moreover $\text{start}(\overrightarrow{xPz}) = e \neq e_x = \text{start}(\overrightarrow{xP^s z})$ by assumption, so the last assertion of Proposition II.1.27 ensures that $s \in V(P) \subseteq V(K)$.

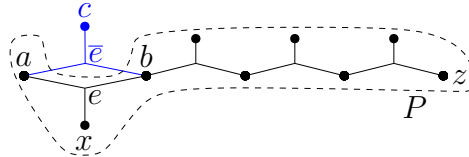


Figure II.14: Illustration of $K = T$ if there is no cx -chain in K .

Now that we have shown that $I_{(D \cup \bar{e})+z}(z\mathcal{D}_1(D \cup \bar{e})) \neq \emptyset$, we can apply Union Lemma II.1.28 in $D \cup \bar{e}$ to D and $P_c = \bar{e}$, which tells us that $D \cup \bar{e}$ contains one of the following:

- a cx -chain P . In this case, taking P and replacing \bar{e} with e yields an x -cycle. Indeed, write $\overrightarrow{cPx} = (c, e_1, \dots, e_L, x)$: since \bar{e} is the only edge incident to c , we have $e_1 = \bar{e}$, so $(x, e, e_2, \dots, e_L, x)$ represents an x -cycle in $D \cup e$.
- a c -tadpole T . Since \bar{e} is the only edge incident to c , we have $\bar{e} \in E(T)$. If $x \in V(T)$, then T contains a cx -chain so we simply go back to that case. If $x \notin V(T)$, then by Claim 5 we have $(T, c) \sim (\varphi(T), x)$, so $\varphi(T)$ is an x -tadpole in $D \cup e$.

- a c -snake S . Since \bar{e} is the only edge incident to c , we have $\bar{e} \in E(S)$. If $x \in V(S)$, then S contains a cx -chain so we simply go back to that case. If $x \notin V(S)$, then by Claim 5 we have $(S, c) \sim (\varphi(S), x)$, so $\varphi(S)$ is an x -snake in $D \cup e$.

All three possibilities thus contradict (C), which concludes the proof of the final assertion of this lemma.

Finally, we prove that e_x is the only edge of D that is incident to x : suppose for a contradiction that there exists $e'_x \in E(D)$ such that $x \in e'_x$ and $e'_x \neq e_x$. Define $P_x := e'_x$: we have $V(P_x) \cap (V(D) \setminus \{x\}) = e'_x \setminus \{x\} \neq \emptyset$ and $\text{start}(\overrightarrow{xP_x}) = e'_x \neq e_x$. Therefore, we can apply what we have shown above to the chain P_x : we get an x -snake or an x -tadpole in $D \cup P_x = D$, contradicting Proposition II.1.26(f). ■

II.1.5.3 Inside structure

The two previous lemmas are about the union of a $\mathcal{D}_1^{\Phi, \text{rest}}$ -danger and a chain. We now look at a $\mathcal{D}_1^{\Phi, \text{rest}}$ -danger alone. In Figure II.9, all featured examples were unions of z -tadpoles and zx -chains only, no z -snakes. Also, x was of degree 1 in all of them. We can now show these properties hold in all interesting cases:

Proposition II.1.30. *Let $(D, x) \in \mathcal{D}_1^{\Phi, \text{rest}}$, with \mathcal{D}_1 -dangerous vertex z . If $J_1(\mathcal{D}_1, D)$ holds, then $M(D) = \emptyset$. In particular, we have $\mathcal{O}_{x,z}(D) = z\mathcal{T}(D^{-x}) \cup \mathcal{P}_{zx}(D)$.*

Proof. Suppose for a contradiction that there exists some $m \in M(D)$. As a gadget, we add two new non-marked vertices a and c as well as a new edge $\bar{e} = \{a, c, m\}$. This does not create any new \mathcal{D}_1 -danger at z : indeed, it is obvious that a z -snake or a z -tadpole cannot contain an edge with two non-marked vertices of degree 1 other than z . For that reason, the fact that $J_1(\mathcal{D}_1, D)$ holds implies that $J_1(\mathcal{D}_1, D \cup \bar{e})$ holds as well. Moreover, since D is not a trivial Maker win by Proposition II.1.26(e), $D \cup \bar{e}$ is not either. Therefore, item (ii) of Union Lemma II.1.28 applied to D and $P_c := \bar{e}$ ensures that $D \cup \bar{e}$ contains a cx -chain or a c -tadpole. Since \bar{e} is the only edge containing c , it is easy to see by removing \bar{e} that D contains an mx -chain or an m -tadpole respectively (see Figure II.15). The former is impossible because an mx -chain in D is an x -snake in D , which cannot exist by Proposition II.1.26(f). The latter is impossible by Proposition II.1.19. We can conclude that $M(D) = \emptyset$, which implies $z\mathcal{S}(D^{-x}) = \emptyset$ hence the last assertion. ■

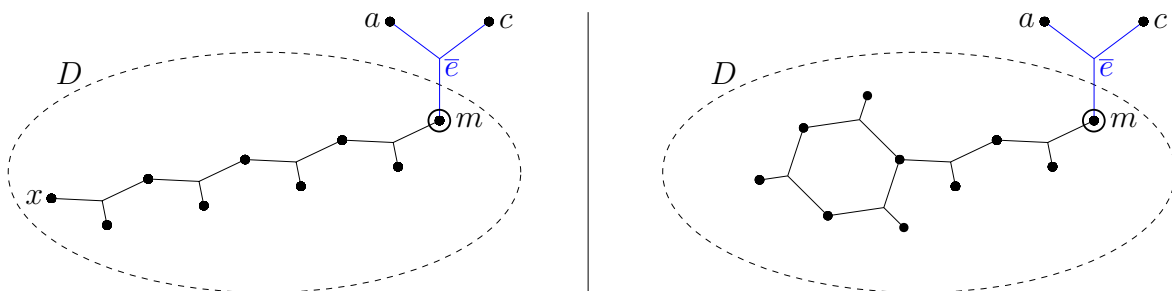


Figure II.15: Left: a cx -chain yields an xm -snake. Right: a c -tadpole yields an m -tadpole.

Proposition II.1.31. *Let $(D, x) \in \mathcal{D}_1^{\Phi, \text{rest}}$, with \mathcal{D}_1 -dangerous vertex z . If $I_{D+z}(z\mathcal{D}_1(D)) \neq \emptyset$, then x is of degree 1 in D .*

Proof. This is the first assertion of Union Lemma II.1.29 applied in $H = D$. ■

The next result delves into the inside structure of the $\mathcal{D}_1^{\mathfrak{O}, \text{rest}}$ -dangers with much more precision.

Proposition II.1.32. *Let $(D, x) \in \mathcal{D}_1^{\mathfrak{O}, \text{rest}}$, with \mathcal{D}_1 -dangerous vertex z . Suppose that $J_1(\mathcal{D}_1, D)$ holds. Then D is of at least one of the two following types (see Figure II.16):*

(1) D contains:

- a z -cycle C such that $x \notin V(C)$;
- an xw -chain P_{xw} for some $w \in \text{out}(C)$ such that $V(P_{xw}) \cap V(C) = \{w\}$;
- some $K \in \mathcal{O}_{x,z}(D)$ such that $V(K) \cap V(P_{xw}) \neq \emptyset$ and $e \setminus \{w\} \not\subseteq V(K)$ where e denotes the unique edge of C containing w .

(2) D contains:

- a z -cycle C such that $x \notin V(C)$;
- an xw -chain P_{xw} for some $w \in V(C)$ such that $V(P_{xw}) \cap V(C) = \{w, w'\}$ where $w' := o(w, \overleftarrow{xP_{xw}w})$.

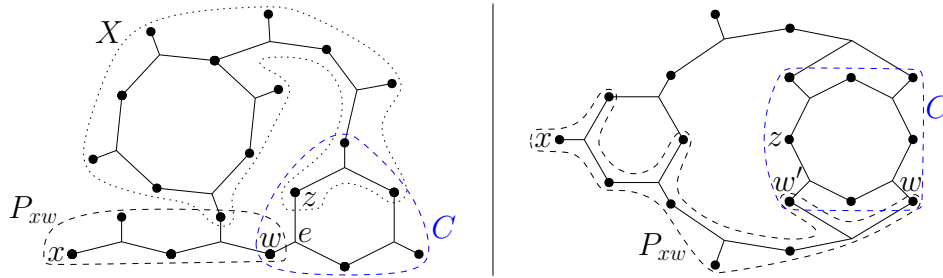


Figure II.16: Two $\mathcal{D}_1^{\mathfrak{O}, \text{rest}}$ -dangers. The left one is of type (1) only (same for the other two from Figure II.9). The right one is of type (2) only.

Proof. Assume that D is not of type (2): we show that D is of type (1).

Claim 6. *There exists a pair (C, P_{xw}) where C is a z -cycle and P_{xw} is an xw -chain for some $w \in \text{out}(C)$ such that $V(P_{xw}) \cap V(C) = \{w\}$.*

Proof of Claim 6. The existence of C is given by Proposition II.1.26(g). The existence of P_{xw} is also straightforward:

- Suppose $x \in V(C)$. Necessarily $x \in \text{out}(C)$, otherwise C would be an x -cycle, contradicting Proposition II.1.26(f). Therefore, simply take $w := x$ and P_{xw} of length 0.
- Suppose $x \notin V(C)$. Let $P \in \mathcal{P}_{zx}(D)$ and define $P_x := \mathbf{P}_{V(C)}(x, P)$. By definition of a projection: $|\text{end}(\overrightarrow{xP_x}) \cap V(C)| \in \{1, 2\}$. We cannot have $|\text{end}(\overrightarrow{xP_x}) \cap V(C)| = 2$ because D would be of type (2), therefore $|\text{end}(\overrightarrow{xP_x}) \cap V(C)| = 1$. Let w be the only vertex in $\text{end}(\overrightarrow{xP_x}) \cap V(C)$. Necessarily $w \in \text{out}(C)$, otherwise $P_x \cup C$ would be an x -tadpole, contradicting Proposition II.1.26(f). Take $P_{xw} := P_x$. \square

Of all pairs (C, P_{xw}) as in Claim 6, we choose one where P_{xw} is longest. This choice ensures that:

Claim 7. *For any z -cycle C' in D , we have $V(C') \cap V(P_{xw}) \neq \emptyset$.*

Proof of Claim 7. Suppose for a contradiction that there exists a z -cycle C' such that $V(C') \cap V(P_{xw}) = \emptyset$. Since $z \in V(C') \cap (V(C) \setminus \text{out}(C))$, the projection $P := \mathbf{P}_{V(C')}(w, C')$ is well defined, and it is of positive length because $w \notin V(C')$. Therefore, the chain $P'_x := [\overrightarrow{xP_{xw}w} \oplus \overrightarrow{wP}]$ is strictly longer than P_{xw} . For the same reason as P_x in the proof of Claim 6 above, P'_x satisfies

$\text{end}(\overrightarrow{xP'_x}) \cap V(C') = \{w'\}$ for some $w' \in \text{out}(C')$, and P'_x is an xw' -chain. The pair (C', P'_x) thus contradicts the maximality of the length of P_{xw} . \square

We will show that $x \notin V(C)$ at the end of the proof. For now, let e be the only edge of C containing w , and let us show the existence of $K \in \mathcal{O}_{x,z}(D)$ such that $V(K) \cap V(P_{xw}) \neq \emptyset$ and $e \setminus \{w\} \not\subseteq V(K)$.

Let us first address the case $z \in e$. Since $z \in \text{inn}(C)$ and $w \in \text{out}(C)$, we have $z \neq w$. Let v be the third vertex of e , so that $e = \{w, z, v\}$. By Proposition II.1.26(b), there exists $K^{\bar{v}} \in \mathcal{O}_{x,z}(D)$ such that $v \notin V(K^{\bar{v}})$, which implies $e \setminus \{w\} \not\subseteq V(K^{\bar{v}})$. Suppose for a contradiction that $V(K^{\bar{v}}) \cap V(P_{xw}) = \emptyset$. In particular $K^{\bar{v}}$ is not a zx -chain. We also know $K^{\bar{v}}$ is not a z -snake by Proposition II.1.30, so $K^{\bar{v}} =: T$ is a z -tadpole. Since $V(T) \cap (V(P_{xw}) \cup e) = \{z\}$, the walk $\overrightarrow{xP_{xw}w} \oplus (w, e, z) \oplus \overrightarrow{zT}$ represents an x -tadpole in D , contradicting Proposition II.1.26(f). In conclusion, we have $V(K^{\bar{v}}) \cap V(P_{xw}) \neq \emptyset$.

We can now assume that $z \notin e$. Write $\overrightarrow{zC} = (z, e_1, \dots, e_L, z)$: we have $e = e_i$ for some $1 \leq i \leq L$. Actually, since $z \notin e$, we have $L \geq 3$ and $2 \leq i \leq L - 1$. We can thus define w_1 (resp. w_2) as the only vertex in $e_{i-1} \cap e_i$ (resp. in $e_i \cap e_{i+1}$), and we have $e = \{w, w_1, w_2\}$. Therefore, $P_1 := [(z, e_1, \dots, e_{i-1}, w_1)]$ is a zw_1 -chain and $P_2 := [(z, e_L, e_{L-1}, \dots, e_{i+1}, w_2)]$ is a zw_2 -chain. These notations are summed up in Figure II.17.

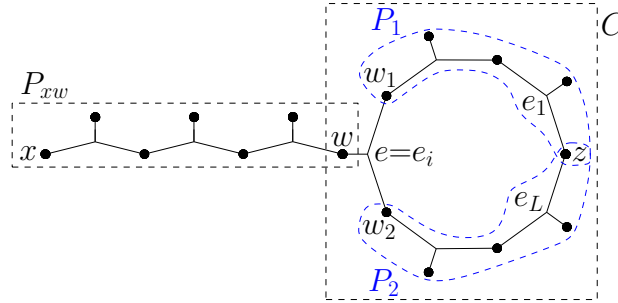


Figure II.17: Summary of the notations in place.

Since $z \notin e$, we have $w_1, w_2 \neq z$. By Proposition II.1.26(b), for all $j \in \{1, 2\}$, there exists $K^{\bar{w}_j} \in \mathcal{O}_{x,z}(D)$ such that $w_j \notin V(K^{\bar{w}_j})$, which implies $e \setminus \{w\} \not\subseteq V(K^{\bar{w}_j})$. We choose $K^{\bar{w}_1} = K^{\bar{w}_2}$ if possible i.e. if there exists an element of $\mathcal{O}_{x,z}(D)$ containing neither w_1 nor w_2 . Suppose for a contradiction that $V(K^{\bar{w}_1}) \cap V(P_{xw}) = \emptyset$ and $V(K^{\bar{w}_2}) \cap V(P_{xw}) = \emptyset$.

In particular, $K^{\bar{w}_1}$ and $K^{\bar{w}_2}$ are not zx -chains, moreover they are not z -snakes by Proposition II.1.30 and they are not z -cycles by Claim 7. Therefore, $K^{\bar{w}_1} =: T^{\bar{w}_1}$ and $K^{\bar{w}_2} =: T^{\bar{w}_2}$ are z -tadpoles that are not cycles. We distinguish between two cases, obtaining a contradiction for both.

- First case: $e_L \notin E(T^{\bar{w}_1})$ or $e_1 \notin E(T^{\bar{w}_2})$. By symmetry, assume that $e_L \notin E(T^{\bar{w}_1})$. It is impossible that $V(T^{\bar{w}_1}) \cap V(P_2) = \{z\}$, otherwise we would have $V(T^{\bar{w}_1}) \cap (V(P_{xw}) \cup e \cup V(P_2)) = \{z\}$ so the walk $\overrightarrow{xP_{xw}w} \oplus (w, e, w_2) \oplus \overrightarrow{w_2P_2z} \oplus \overrightarrow{zT^{\bar{w}_1}}$ would represent an x -tadpole in D , contradicting Proposition II.1.26(f). Therefore, the projection $P^{\bar{w}_1} := \mathbf{P}_{V(P_2) \setminus \{z\}}(z, T^{\bar{w}_1})$ is well defined. Since $P^{\bar{w}_1} \subseteq T^{\bar{w}_1}$, we have $V(P^{\bar{w}_1}) \cap (V(P_{xw}) \cup \{w_1\}) = \emptyset$ and $e_L \notin E(P^{\bar{w}_1})$. In particular $\text{start}(zP^{\bar{w}_1}) \neq e_L = \text{start}(zP_2w_2)$, so we can apply Union Lemma I.3.40. Since $P_2 \cup P^{\bar{w}_1}$ cannot contain a z -cycle by Claim 7, it contains a w_2 -tadpole T . We have $V(T) \subseteq V(P_2) \cup V(P^{\bar{w}_1})$ hence $V(T) \cap (V(P_{xw}) \cup e) = \{w_2\}$, so the walk $\overrightarrow{xP_{xw}w} \oplus (w, e, w_2) \oplus \overrightarrow{w_2T}$ represents an x -tadpole in D , contradicting Proposition II.1.26(f).
- Second case: $e_L \in E(T^{\bar{w}_1})$ and $e_1 \in E(T^{\bar{w}_2})$. Since $T^{\bar{w}_1}$ and $T^{\bar{w}_2}$ are not cycles, z is of degree 1 in both of them, hence $e_1 \notin E(T^{\bar{w}_1})$ and

$e_L \notin E(T^{\overline{w_2}})$. Since $e_1 \in E(T^{\overline{w_2}})$ and $e_1 \notin E(T^{\overline{w_1}})$, we have $T^{\overline{w_1}} \neq T^{\overline{w_2}}$, so our initial choice of $T^{\overline{w_1}}$ and $T^{\overline{w_2}}$ ensures that $w_2 \in V(T^{\overline{w_1}})$ and $w_1 \in V(T^{\overline{w_2}})$.

- Firstly, suppose that $w_2 \notin \text{out}(C_{T^{\overline{w_1}}})$ or $w_1 \notin \text{out}(C_{T^{\overline{w_2}}})$. By symmetry, assume that $w_2 \notin \text{out}(C_{T^{\overline{w_1}}})$. By Substructure Lemma I.3.31, $T^{\overline{w_1}}$ contains a w_2 -tadpole T . Since $V(T) \cap (V(P_{xw}) \cup e) = \{w_2\}$, the walk $\overrightarrow{xP_{xw}w} \oplus (w, e, w_2) \oplus \overrightarrow{w_2T}$ represents an x -tadpole in D , which contradicts Proposition II.1.26(f).
- Finally, suppose that $w_2 \in \text{out}(C_{T^{\overline{w_1}}})$ and $w_1 \in \text{out}(C_{T^{\overline{w_2}}})$. Since $J_1(\mathcal{D}_1, D)$ holds, there exists $s \in I_{D+z}(z\mathcal{D}_1(D))$. In particular, $s \in V(T^{\overline{w_1}}) \cap V(C)$. Since $w \notin V(T^{\overline{w_1}})$, we have $s \neq w$, hence $s \in V(P_1)$ or $s \in V(P_2)$. By symmetry, assume $s \in V(P_1)$. In particular, $s \neq w_2$: by Substructure Lemma I.3.29, the fact that $w_2 \in \text{out}(C_{T^{\overline{w_1}}})$ thus ensures the existence of a zs -chain $P_{zs}^{\overline{w_2}}$ in $T^{\overline{w_1}}$ that does not contain w_2 . Since $e_1 \notin E(T^{\overline{w_1}})$, we have $e_1 \notin E(P_{zs}^{\overline{w_2}})$. Therefore $\text{start}(zP_{zs}^{\overline{w_2}}) \neq e_1 = \text{start}(zP_1w_1)$, moreover $V(P_{zs}^{\overline{w_2}}) \cap (V(P_1) \setminus \{z\}) \supseteq \{s\} \neq \emptyset$ so we can apply Union Lemma I.3.40: since $P_1 \cup P_{zs}^{\overline{w_2}}$ cannot contain a z -cycle by Claim 7, it contains a w_1 -tadpole T . We have $V(T) \subseteq V(P_1) \cup V(P_{zs}^{\overline{w_2}})$ hence $V(T) \cap (V(P_{xw}) \cup e) = \{w_1\}$, so the walk $\overrightarrow{xP_{xw}w} \oplus (w, e, w_1) \oplus \overrightarrow{w_1T}$ represents an x -tadpole in D , contradicting Proposition II.1.26(f).

In conclusion, we have shown the existence of $K \in \mathcal{O}_{x,z}(D)$ such that $V(K) \cap V(P_{xw}) \neq \emptyset$ and $e \setminus \{w\} \not\subseteq V(K)$. To prove that D is of type (1), it only remains to show that $x \notin V(C)$. Suppose for a contradiction that $x \in V(C)$ i.e. $x = w$ i.e. $V(P_{xw}) = \{x\}$. Since $V(K) \cap V(P_{xw}) \neq \emptyset$ by definition of K , we get $x \in V(K)$, so there exists an edge e' of K that is incident to x . Moreover, e is also incident to $w = x$. Since $e \notin E(K)$ by definition of K , we have $e' \neq e$. Therefore, e and e' are two distinct edges of D that are incident to x , contradicting Proposition II.1.31. This ends the proof. \blacksquare

II.1.6 Proof of the main results... first assuming a key lemma

Theorems II.1.12 and II.1.13 can be deduced relatively easily from the following intermediate result, which we will prove to round off this section:

Lemma II.1.33. *Let H be a 3-uniform marked hypergraph that is not a trivial Maker win, with $|V(H) \setminus M(H)| \geq 2$. Suppose that $J_1(\mathcal{D}_1, H)$ holds. Then, for any $x \in V(H) \setminus M(H)$ such that there exists an x -snake in H , we have $I_{H+x}(x\mathcal{D}_2(H)) \neq \emptyset$.*

In other words, under the assumptions of Lemma II.1.33, if x sees a marked vertex then not only do its \mathcal{D}_1 -dangers intersect (which is a given, since $J_1(\mathcal{D}_1, H)$ holds) but actually even its \mathcal{D}_2 -dangers do.

II.1.6.1 Proof of Theorem II.1.12

As announced, we actually prove Theorem II.1.14 first, which uses the approximation \mathcal{D}_2 . Theorem II.1.12 then follows as a corollary.

Proof of Theorem II.1.14 assuming Lemma II.1.33. Item (i) is a direct consequence of Proposition I.2.14. We now show item (ii) by induction on $|V(H) \setminus M(H)|$.

Let us start with the base case $|V(H) \setminus M(H)| \in \{2, 3\}$. Let $x_1 \in V(H) \setminus M(H)$ and $y_1 \in I_{H+x_1}(x_1\mathcal{D}_2(H))$, which exists since $J_1(\mathcal{D}_2, H)$ holds. The trivial danger of size 3 is in $S \subseteq \mathcal{D}_2$, therefore all trivial dangers at x_1 in H contain y_1 , so the fact that H is not a trivial Maker

win implies that $H^{+x_1-y_1}$ is not a trivial Maker win either. Since $|V(H^{+x_1-y_1}) \setminus M(H^{+x_1-y_1})| \leq 1$, this means $H^{+x_1-y_1}$ is a Breaker win, so H is a Breaker win.

For the induction step, assume $|V(H) \setminus M(H)| \geq 4$ and the implication to be true for marked hypergraphs with less non-marked vertices than H . Let $x_1 \in V(H) \setminus M(H)$ and $y_1 \in I_{H+x_1}(x_1\mathcal{D}_2(H))$, which exists since $J_1(\mathcal{D}_2, H)$ holds: we must show that $H^{+x_1-y_1}$ is a Breaker win. Let us first list a few important properties of $H^{+x_1-y_1}$:

- (a) $|V(H^{+x_1-y_1}) \setminus M(H^{+x_1-y_1})| = |V(H) \setminus M(H)| - 2 \geq 2$.
- (b) $H^{+x_1-y_1}$ is not a trivial Maker win. Indeed, the trivial danger of size 3 is in $\mathcal{S} \subseteq \mathcal{D}_2$, therefore all trivial dangers at x_1 in H contain y_1 , so the fact that H is not a trivial Maker win implies that $H^{+x_1-y_1}$ is not a trivial Maker win either.
- (c) $J_1(\mathcal{D}_1, H^{+x_1-y_1})$ holds. Indeed, $J_1(\mathcal{D}_1, H)$ holds because $\mathcal{D}_1 \subseteq \mathcal{D}_2$ and $J_1(\mathcal{D}_2, H)$ holds. Besides, since $\mathcal{D}_2 = \mathcal{D}_1 \cup \mathcal{D}_1^{\mathfrak{O}, \text{rest}}$ by definition, we have $I_{H+x_1}(x_1\mathcal{D}_2(H)) = I_{H+x_1}(x_1\mathcal{D}_1^*(H))$ by Proposition I.2.30 hence $y_1 \in I_{H+x_1}(x_1\mathcal{D}_1^*(H)) \subseteq I_{H+x_1}(x_1\mathcal{D}_1^{\mathfrak{O}}(H))$. Therefore, Proposition I.2.25 with $\mathcal{F} = \mathcal{D}_1$ ensures that $J_1(\mathcal{D}_1, H^{+x_1-y_1})$ holds.

Thanks to (a) and (b), checking that property $J_1(\mathcal{D}_2, H^{+x_1-y_1})$ holds is sufficient to prove that $H^{+x_1-y_1}$ is a Breaker win, according to the induction hypothesis. Let $x \in V(H^{+x_1-y_1}) \setminus M(H^{+x_1-y_1})$: we want to show that $I_{(H^{+x_1-y_1})+x}(x\mathcal{D}_2(H^{+x_1-y_1})) \neq \emptyset$. Assume that there exists some $D_0 \in x\mathcal{D}_2(H^{+x_1-y_1})$, otherwise $I_{(H^{+x_1-y_1})+x}(x\mathcal{D}_2(H^{+x_1-y_1})) = I_{(H^{+x_1-y_1})+x}(\emptyset) = V((H^{+x_1-y_1})^{+x}) \setminus M((H^{+x_1-y_1})^{+x}) \neq \emptyset$ trivially since $|V(H) \setminus M(H)| \geq 4$.

- 1) First case: there is no xx_1 -snake in $H^{+x_1-y_1}$.

What happens here is that any vertex that hits all the \mathcal{D}_2 -dangers at x in H still works in $H^{+x_1-y_1}$, because the marking of x_1 has not created any new \mathcal{D}_2 -danger at x . Indeed, for all $D \in x\mathcal{D}_2(H^{+x_1-y_1})$ (recall that $\mathcal{D}_2 = \mathcal{S} \cup \mathcal{T} \cup \mathcal{D}_1^{\mathfrak{O}, \text{rest}}$ by definition):

- If $(D, x) \in \mathcal{S}$, then $x_1 \notin V(D)$ since we are assuming that there is no xx_1 -snake in $H^{+x_1-y_1}$.
- If $(D, x) \in \mathcal{T}$, then $x_1 \notin V(D)$ since $M(D) = \emptyset$ by definition of the family \mathcal{T} .
- If $(D, x) \in \mathcal{D}_1^{\mathfrak{O}, \text{rest}}$, then $x_1 \notin V(D)$ since $M(D) = \emptyset$ by Proposition II.1.30 (which (c) allows us to use).

Therefore, we have $x\mathcal{D}_2(H^{+x_1-y_1}) \subseteq x\mathcal{D}_2(H^{-x_1-y_1}) \subseteq x\mathcal{D}_2(H)$. Now, let $y \in I_{H+x}(x\mathcal{D}_2(H))$. To show that $y \in I_{(H^{+x_1-y_1})+x}(x\mathcal{D}_2(H^{+x_1-y_1}))$, since $x\mathcal{D}_2(H^{+x_1-y_1}) \subseteq x\mathcal{D}_2(H)$, it suffices to check that $y \notin \{x_1, y_1\}$. For this, we use D_0 . On the one hand, we have $D_0 \in x\mathcal{D}_2(H^{+x_1-y_1}) \subseteq x\mathcal{D}_2(H)$ hence $y \in V(D_0)$. On the other hand, we have $D_0 \in x\mathcal{D}_2(H^{+x_1-y_1}) \subseteq x\mathcal{D}_2(H^{-x_1-y_1})$ hence $x_1, y_1 \notin V(D_0)$. In conclusion, we do have $y \notin \{x_1, y_1\}$, so $y \in I_{(H^{+x_1-y_1})+x}(x\mathcal{D}_2(H^{+x_1-y_1}))$ hence $I_{(H^{+x_1-y_1})+x}(x\mathcal{D}_2(H^{+x_1-y_1})) \neq \emptyset$.

- 2) Second case: there is an xx_1 -snake in $H^{+x_1-y_1}$.

Here, we have the x -snake that is necessary to apply Lemma II.1.33 to $H^{+x_1-y_1}$. The other assumptions of this lemma are also verified thanks to (a), (b) and (c). In conclusion, Lemma II.1.33 applies and yields $I_{(H^{+x_1-y_1})+x}(x\mathcal{D}_2(H^{+x_1-y_1})) \neq \emptyset$ as desired. ■

Proof of Theorem II.1.12 assuming Lemma II.1.33. Item (i) is a direct consequence of Proposition I.2.14. Item (ii) follows from Theorem II.1.14: indeed, since $\mathcal{D}_2 \subseteq \mathcal{D}_0^{*2}$, $J_1(\mathcal{D}_0^{*2}, H)$ implies $J_1(\mathcal{D}_2, H)$ and $I_{H+x_1}(x_1\mathcal{D}_2(H)) \supseteq I_{H+x_1}(x_1\mathcal{D}_0^{*2}(H))$. As for the equivalence between $J_1(\mathcal{D}_0^{*2}, H)$ and $J_3(\mathcal{D}_0, H)$, it is given by Proposition I.2.28. Finally, the ultimate assertion of Theorem II.1.12 is simply Proposition II.1.11 with $r = 3$. ■

II.1.6.2 Proof of Theorem II.1.13

Similarly, we actually prove Theorem II.1.15 first, which uses the approximation \mathcal{D}_1 . Theorem II.1.13 then follows as a corollary.

Proof of Theorem II.1.15 assuming Lemma II.1.33. If H is a Breaker win then $J_1(\mathcal{D}_1, H)$ holds by Proposition I.2.19. Now assume that $J_1(\mathcal{D}_1, H)$ holds. We claim that, actually, $J_1(\mathcal{D}_2, H)$ holds: indeed, for all $x \in V(H) \setminus M(H)$, there exists an x -snake in H by assumption hence $I_{H+x}(x\mathcal{D}_2(H)) \neq \emptyset$ by Lemma II.1.33. Therefore, H is a Breaker win according to Theorem II.1.14. ■

Proof of Theorem II.1.13 assuming Lemma II.1.33. The "only if" direction is given by Proposition I.2.19. The "if" direction follows from Theorem II.1.15 since $J_1(\mathcal{D}_0^*, H)$ implies $J_1(\mathcal{D}_1, H)$. Finally, the ultimate assertion of Theorem II.1.13 is simply Proposition II.1.11 with $r = 2$. ■

II.1.7 Proof of the key lemma

As we have just seen, all structural results will be proved once Lemma II.1.33 is. Let H be a 3-uniform marked hypergraph that is not a trivial Maker win, and suppose that $J_1(\mathcal{D}_1, H)$ holds. Let $x \in V(H) \setminus M(H)$ and $m \in M(H)$ such that there exists an xm -snake in H : we want to find some $y \in I_{H+x}(x\mathcal{D}_2(H)) = I_{H+x}(x(\mathcal{D}_1 \cup \mathcal{D}_1^{\text{rest}})(H))$. Since $J_1(\mathcal{D}_1, H)$ holds, we already know that $I_{H+x}(x\mathcal{D}_1(H)) \neq \emptyset$, however picking an arbitrary $y \in I_{H+x}(x\mathcal{D}_1(H))$ does not work in general, as shown in Figure II.18. In this example, we can see that H satisfies the conditions of Lemma II.1.33, and that the only \mathcal{D}_1 -dangers at x in H are two xm -snakes whose intersection $I_{H+x}(x\mathcal{D}_1(H))$ is represented by the vertices in square boxes. We can see that several of them are not in $I_{H+x}(x\mathcal{D}_2(H))$, because they miss D which is a $\mathcal{D}_1^{\text{rest}}$ -danger at x : this is the case for the vertex y' for instance.

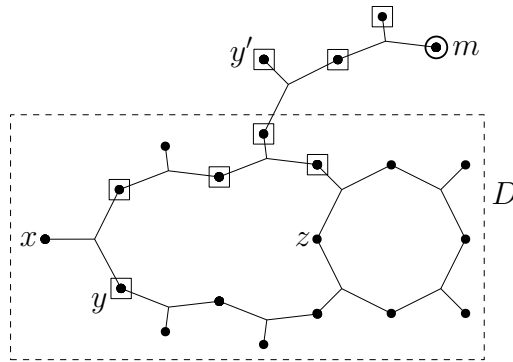


Figure II.18: In this example, we have $y \in I_{H+x}(x\mathcal{D}_2(H))$, but $y' \in I_{H+x}(x\mathcal{D}_1(H)) \setminus I_{H+x}(x\mathcal{D}_2(H))$ since $y' \notin V(D)$.

This inspires us to choose $y \in I_{H+x}(x\mathcal{D}_1(H))$ furthest away from m , as in Figure II.18 for example:

Notation II.1.34. For all $a, b \in V(H)$, we denote by $\text{sep}_H(a, b)$ the length of a shortest ab -chain in H , where $\text{sep}_H(a, b) = \infty$ by convention if there exists none.

We now fix $y \in I_{H+x}(x\mathcal{D}_1(H))$ maximizing $\text{sep}_H(y, m)$, and we suppose for a contradiction that $y \notin I_{H+x}(x\mathcal{D}_2(H))$: there exists $D \in x\mathcal{D}_1^{\text{rest}}(H)$ such that $y \notin V(D)$. The idea of the proof is to eventually exhibit a vertex $w \in V(D)$ such that $I_{H+w}(w\mathcal{D}_1(H)) = \emptyset$, contradicting the fact

that $J_1(\mathcal{D}_1, H)$ holds.

Let z be a \mathcal{D}_1 -dangerous vertex in (D, x) .

II.1.7.1 Preliminary statements

Since H is not a trivial Maker win and $J_1(\mathcal{D}_1, H)$ holds, all results from Subsection II.1.5 apply to D . In particular:

Proposition II.1.35. *D has the following properties:*

- $M(D) = \emptyset$. In particular, $m \notin V(D)$.
- $\mathcal{O}_{x,z}(D) = z\mathcal{T}(D^{-x}) \cup \mathcal{P}_{zx}(D)$.
- There is exactly one edge of D that is incident to x : we call it e_x .

Proof. This is given by Propositions II.1.30 and II.1.31. ■

The next two properties can be summed up as follows:

- When following a chain starting from m , we cannot enter D strictly before encountering y .
- When following a chain starting from x by an edge other than e_x , we cannot re-enter D strictly before encountering y .

Proposition II.1.36. *Any m -chain P_m in H such that $V(P_m) \cap V(D) \neq \emptyset$ contains y .*

Proof. Since $J_1(\mathcal{D}_1, H)$ holds and $m \notin V(D)$, Union Lemma II.1.28 with $c = m$ ensures that $D \cup P_m$ contains an m -tadpole or an mx -chain (i.e. an xm -snake). There cannot be an m -tadpole in H according to Proposition II.1.19, therefore $D \cup P_m$ contains an xm -snake. Since $y \in I_{H+x}(x\mathcal{D}_1(H))$, that xm -snake must contain y , moreover $y \notin V(D)$ by assumption so $y \in V(P_m)$. ■

Proposition II.1.37. *Any x -chain P_x in H such that $\text{start}(\overrightarrow{xP_x}) \neq e_x$ and $V(P_x) \cap (V(D) \setminus \{x\}) \neq \emptyset$ contains y .*

Proof. Since $J_1(\mathcal{D}_1, H)$ holds, we have $I_{H+z}(z\mathcal{D}_1(H)) \neq \emptyset$, so Union Lemma II.1.29 ensures that $D \cup P_x$ contains an x -tadpole or an x -snake. In both cases, it contains y , moreover $y \notin V(D)$ by assumption, so $y \in V(P_x)$. ■

We now state a useful preliminary lemma:

Lemma II.1.38. *Any $v \in V(D) \setminus \{x\}$ satisfies $\text{sep}_H(v, m) \geq \text{sep}_H(y, m)$, moreover:*

- If $\text{sep}_H(v, m) > \text{sep}_H(y, m)$, then there exists an xm -snake $S_{xm}^{\overrightarrow{v}}$ in H that does not contain v .
- If $\text{sep}_H(v, m) = \text{sep}_H(y, m)$, then any shortest vm -snake S_{vm} in H satisfies $V(S_{vm}) \cap V(D) = \{v\}$ and $o(v, \overrightarrow{vS_{vm}m}) = y$, moreover there is no v -tadpole in D .

Proof. The fact that $\text{sep}_H(v, m) \geq \text{sep}_H(y, m)$ is a direct consequence of Proposition II.1.36: since $v \in V(D)$, any vm -snake in H contains y .

- Suppose $\text{sep}_H(v, m) > \text{sep}_H(y, m)$.

Let S_{ym} be a shortest ym -snake in H : note that there does exist one, since there exists an xm -snake by assumption, which must contain y and therefore contains a ym -snake by Substructure Lemma I.3.25. Since S_{ym} is shortest and $\text{sep}_H(v, m) > \text{sep}_H(y, m)$, we have $v \notin V(S_{ym})$.

We necessarily have $v \notin I_{H+x}(x\mathcal{D}_1(H))$, otherwise the fact that $\text{sep}_H(v, m) > \text{sep}_H(y, m)$ would contradict our choice of y . Since v is non-marked (recall that $M(D) = \emptyset$) and

distinct from x , this means there exists some $K \in x\mathcal{D}_1(H)$ such that $v \notin V(K)$. On the other hand, since $y \in I_{H+x}(x\mathcal{D}_1(H))$, we have $y \in V(K)$ hence $y \in V(K) \cap V(S_{ym})$. This allows us to use the adequate union lemma in $K \cup S_{ym}$ to find the desired xm -snake $S_{xm}^{\bar{v}}$ (ensuring that $v \notin V(S_{xm}^{\bar{v}})$ since $v \notin V(K) \cup V(S_{ym})$):

- Suppose $K =: S \in x\mathcal{S}(H)$. If the marked vertex of S is m , then $S_{xm}^{\bar{v}} := S$ is the desired xm -snake. Otherwise $m \notin V(S)$, so apply Union Lemma I.3.39 with $a = x$, $S_{ab} = S$, $c = m$ and $P_c = S_{ym}$. Since $S \cup S_{ym} \subseteq H$ cannot contain an m -snake by Proposition II.1.19, it contains an mx -chain i.e. an xm -snake $S_{xm}^{\bar{v}}$.
- Suppose $K =: T \in x\mathcal{T}(H)$. We have $M(T) = \emptyset$ hence $m \notin V(T)$, so apply Union Lemma I.3.41 with $a = x$, $c = m$ and $P_c = S_{ym}$. Since $T \cup S_{ym} \subseteq H$ cannot contain an m -tadpole by Proposition II.1.19, it contains an mx -chain i.e. an xm -snake $S_{xm}^{\bar{v}}$.
- Suppose $\text{sep}_H(v, m) = \text{sep}_H(y, m)$.

Let S_{vm} be a shortest vm -snake in H . Since $v \in V(D)$, we have $y \in V(S_{vm})$ by Proposition II.1.36. If $y \neq o(v, \overrightarrow{vS_{vm}m})$ (see Figure II.19, top), then S_{vm} contains a ym -snake that is shorter than S_{vm} : this is impossible since $\text{sep}_H(v, m) = \text{sep}_H(y, m)$ and S_{vm} has been chosen shortest. Therefore $y = o(v, \overrightarrow{vS_{vm}m})$ (see Figure II.19, bottom). The m -chain S_{vm}^{-y-v} does not contain y and thus contains no vertex in D by Proposition II.1.36, hence why $V(S_{vm}) \cap V(D) = \{v\}$. Finally, there cannot be a v -tadpole T in D , because $S_{vm} \cup T$ would then be an m -tadpole in H since $V(S_{vm}) \cap V(T) = \{v\}$, contradicting Proposition II.1.19. ■

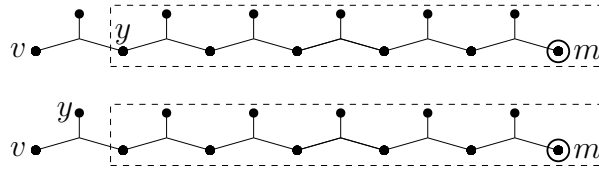


Figure II.19: The snake S_{vm} if $y \neq o(v, \overrightarrow{vS_{vm}m})$ (top, the contradictory ym -snake is highlighted) or if $y = o(v, \overrightarrow{vS_{vm}m})$ (bottom, the snake S_{vm}^{-y-v} is highlighted).

As we have often done in Subsection II.1.5, we fix a vertex $s \in I_{H+z}(z\mathcal{D}_1(H))$, given by Proposition II.1.27 (which also tells us that $s \in V(D)$). Before we engage in the core of the proof, let us summarize the objects involved and some of their basic properties that will be used thereafter, with Table II.1:

II.1.7.2 Roadmap of the proof

As previously stated, the idea of the proof is to eventually exhibit a vertex $w \in V(D)$ such that $I_{H+w}(w\mathcal{D}_1(H)) = \emptyset$, contradicting the fact that $J_1(\mathcal{D}_1, H)$ holds. The roadmap to achieve this is given by the following result, which we will prove in this segment:

Proposition II.1.39. *Let $w \in V(D) \setminus \{x\}$. Suppose that D contains the following three subhypergraphs:*

- (i) a z -cycle C containing w ;
- (ii) an xw -chain P_{xw} such that $V(P_{xw}) \cap \text{inn}(C) \subseteq \{w\}$;
- (iii) a w -tadpole that does not contain s .

Then $I_{H+w}(w\mathcal{D}_1(H)) = \emptyset$.

Showing that $I_{H+w}(w\mathcal{D}_1(H)) = \emptyset$ in the proof of Proposition II.1.39 will require the ability, for every non-marked vertex $v \neq w$, to exhibit a \mathcal{D}_1 -danger at w that does not contain v . The

H	<ul style="list-style-type: none"> · not a trivial Maker win · $J_1(\mathcal{D}_1, H)$ holds 	
x	<ul style="list-style-type: none"> · $x \in V(H) \setminus M(H)$ 	
D	<ul style="list-style-type: none"> · $\mathcal{D}_1^{\mathcal{O}, \text{rest}}$-danger at x in H · $\mathcal{O}_{x,z}(D) = z\mathcal{T}(D^{-x}) \cup \mathcal{P}_{zx}(D)$ 	
z	<ul style="list-style-type: none"> · \mathcal{D}_1-dangerous vertex in (D, x) 	
m	<ul style="list-style-type: none"> · $m \in M(H)$ · $m \notin V(D)$ 	
y	<ul style="list-style-type: none"> · $y \in I_{H+x}(x\mathcal{D}_1(H))$ · $y \notin V(D)$ 	
s	<ul style="list-style-type: none"> · $s \in I_{H+z}(z\mathcal{D}_1(H))$ · $s \in V(D)$ 	
e_x	<ul style="list-style-type: none"> · unique edge incident to x in D 	

Table II.1: The objects involved and some of their properties.

following lemma applied to $d = w$ gives us that object under certain conditions.

Lemma II.1.40. *Let $d, v \in V(D) \setminus \{x\}$. Suppose that $\text{sep}_H(v, m) > \text{sep}_H(y, m)$ and that there exists a dx -chain $P_{dx}^{\bar{v}}$ in D that does not contain v . Then there exists a dm -snake in H that does not contain v .*

Proof. Suppose for a contradiction that:

All dm -snakes in H contain v . (C)

We are going to exhibit an m -tadpole in H , contradicting Proposition II.1.19. This m -tadpole will be obtained inside the union of an xy -chain and an xm -snake having specific properties, whose existence is given by the following two claims which we prove independently from each other. Define $t := o(x, \overrightarrow{xP_{dx}^{\bar{v}}d})$, and note that $\text{start}(\overrightarrow{xP_{dx}^{\bar{v}}d}) = e_x$ since e_x is the only edge incident to x in D .

Claim 8. *There exists an xy -chain $P_{xy}^{\bar{v}}$ in H such that:*

- $V(P_{xy}^{\bar{v}}) \subseteq V(P_{dx}^{\bar{v}}) \cup \{y\}$.
- $o(x, \overrightarrow{xP_{xy}^{\bar{v}}y}) = t$.

Proof of Claim 8. We have $\text{sep}_H(v, m) > \text{sep}_H(y, m)$ by assumption, so by Lemma II.1.38 there exists an xm -snake $S_{xm}^{\bar{v}}$ in H that does not contain v . Since $P_{dx}^{\bar{v}} \subseteq D$ and $m \notin V(D)$, the edge $e^* := \text{end}(\overrightarrow{m\mathbf{P}_{V(P_{dx}^{\bar{v}})}(m, S_{xm}^{\bar{v}})})$ is well defined. According to (C), there is no dm -snake in $P_{dx}^{\bar{v}} \cup S_{xm}^{\bar{v}}$. Therefore, by Union Lemma I.3.38 applied to $a = d, b = x, c = m, P_{ab} = P_{dx}^{\bar{v}}$ and $P_c = S_{xm}^{\bar{v}}$: $|e^* \cap V(P_{dx}^{\bar{v}})| = 2$, $e^* \perp \overleftarrow{xP_{dx}^{\bar{v}}d}$, moreover there is an x -tadpole T in $P_{dx}^{\bar{v}} \cup e^*$. Since $|e^* \cap V(P_{dx}^{\bar{v}})| = 2$, there is exactly one vertex of T that is not in $P_{dx}^{\bar{v}}$. That vertex is necessarily y , as illustrated in Figure II.20: indeed, we know $y \in V(T)$ because $y \in I_{H+x}(x\mathcal{D}_1(H))$, and $y \notin V(D) \supseteq V(P_{dx}^{\bar{v}})$. By Substructure Lemma I.3.30, T contains an xy -chain $P_{xy}^{\bar{v}}$, and we have $V(P_{xy}^{\bar{v}}) \subseteq V(T) \subseteq V(P_{dx}^{\bar{v}}) \cup \{y\}$. See Figure II.20.

Finally, let us check that $o(x, \overrightarrow{xP_{xy}^{\bar{v}}y}) = t$. Since $|e^* \cap V(P_{dx}^{\bar{v}})| = 2$ and $e^* \perp \overleftarrow{xP_{dx}^{\bar{v}}d}$, there are two possibilities:

- First possibility: $\{x, t\} \subseteq e^*$. Then $P_{xy}^{\bar{v}}$ consists of the single edge $e^* = \{x, t, y\}$, so obviously $o(x, \overrightarrow{xP_{xy}^{\bar{v}}y}) = t$.

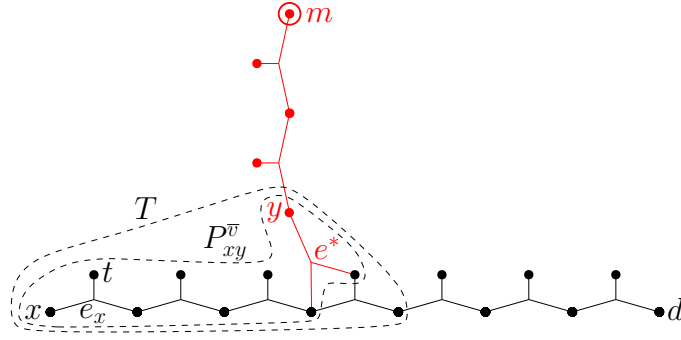


Figure II.20: Definition of $P_{xy}^{\bar{v}}$. The represented chains are $P_{dx}^{\bar{v}}$ (horizontal) and $\mathbf{P}_{V(P_{dx}^{\bar{v}})}(m, S_{xm}^{\bar{v}})$ (vertical).

- Second possibility: $\{x, t\} \cap e^* = \emptyset$. Then $\text{start}(\overrightarrow{xP_{xy}^{\bar{v}}}) \neq e^*$, so $\text{start}(\overrightarrow{xP_{xy}^{\bar{v}}}) = \text{start}(\overrightarrow{xP_{dx}^{\bar{v}}d}) = e_x \ni t$. Moreover t is of degree 1 in $P_{dx}^{\bar{v}} \cup e^* \supseteq P_{xy}^{\bar{v}}$, so necessarily $t \notin \text{inn}(P_{xy}^{\bar{v}})$ hence $t = o(x, \overrightarrow{xP_{xy}^{\bar{v}}})$. \square

Claim 9. *There exists an xm -snake $S_{xm}^{\bar{t}}$ in H such that:*

- $t \notin V(S_{xm}^{\bar{t}})$.
- $y \in \text{inn}(S_{xm}^{\bar{t}})$. (We define $P_{xy}^{\bar{t}}$, resp. $S_{ym}^{\bar{t}}$, as the unique xy -chain, resp. ym -snake, in $S_{xm}^{\bar{t}}$.)
- $(V(S_{xm}^{\bar{t}}) \cap V(P_{dx}^{\bar{v}})) \setminus \{x\} \subseteq \{u\}$ where $u := o(y, \overleftarrow{xP_{xy}^{\bar{t}}})$.

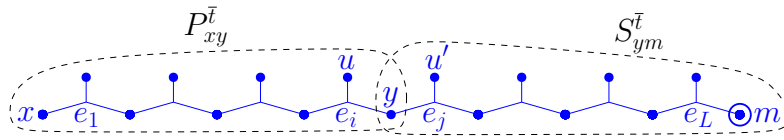
Proof of Claim 9. It is impossible that $\text{sep}_H(t, m) = \text{sep}_H(y, m)$: indeed, a shortest tm -snake S_{tm} in H would then satisfy $V(S_{tm}) \cap V(D) = \{t\}$ by Lemma II.1.38, hence $v \notin V(S_{tm})$, so the walk $\overrightarrow{dP_{dx}^{\bar{v}}t} \oplus \overrightarrow{tS_{tm}m}$ would represent a dm -snake in H that does not contain v , contradicting (C). Therefore, Lemma II.1.38 ensures that $\text{sep}_H(t, m) > \text{sep}_H(y, m)$, and that there exists an xm -snake $S_{xm}^{\bar{t}}$ in H such that $t \notin V(S_{xm}^{\bar{t}})$. Since $y \in I_{H+x}(x\mathcal{D}_1(H))$, we obviously have $y \in V(S_{xm}^{\bar{t}})$.

Write $\overleftarrow{xS_{xm}^{\bar{t}}m} = (x, e_1, \dots, e_L, m)$. Recalling Notation I.3.4, write $\overrightarrow{xS_{xm}^{\bar{t}}m|_{\{y\}}} = (x, e_1, \dots, e_i)$ and $\overleftarrow{xS_{xm}^{\bar{t}}m|_{\{y\}}} = (m, e_L, e_{L-1}, \dots, e_j)$. Note that $j = i + 1$ if $y \in \text{inn}(S_{xm}^{\bar{t}})$ and $j = i$ otherwise.

- By definition of the walk $\overrightarrow{xS_{xm}^{\bar{t}}m|_{\{y\}}}$, the x -chain $[(x, e_1, \dots, e_{i-1})]$ does not contain y . Moreover $e_1 \neq e_x$ because $t \notin V(S_{xm}^{\bar{t}}) \supseteq e_1$. Therefore, $(e_1 \cup \dots \cup e_{i-1}) \cap (V(D) \setminus \{x\}) = \emptyset$ by Proposition II.1.36.
- By definition of the walk $\overleftarrow{xS_{xm}^{\bar{t}}m|_{\{y\}}}$, the m -chain $[(m, e_L, e_{L-1}, \dots, e_{j+1})]$ does not contain y . Therefore, $(e_{j+1} \cup \dots \cup e_L) \cap V(D) = \emptyset$ by Proposition II.1.37.

Suppose that $y \notin \text{inn}(S_{xm}^{\bar{t}})$: then $i = j$ hence $V(S_{xm}^{\bar{t}}) = e_1 \cup \dots \cup e_{i-1} \cup \{y\} \cup e_{j+1} \cup \dots \cup e_L$. By the above, this yields $V(S_{xm}^{\bar{t}}) \cap V(D) = \{x\}$ and in particular $v \notin V(S_{xm}^{\bar{t}})$, therefore the walk $\overrightarrow{dP_{dx}^{\bar{v}}x} \oplus \overrightarrow{xS_{xm}^{\bar{t}}m}$ represents a dm -snake that does contain v . This contradicts (C).

Therefore, we have $y \in \text{inn}(S_{xm}^{\bar{t}})$. Let $P_{xy}^{\bar{t}}$ (resp. $S_{ym}^{\bar{t}}$) be the unique xy -chain (resp. ym -snake) in $S_{xm}^{\bar{t}}$, and define $u := o(y, \overleftarrow{xP_{xy}^{\bar{t}}})$ and $u' := o(y, \overrightarrow{yS_{ym}^{\bar{t}}m})$, as in Figure II.21. Since $y \in \text{inn}(S_{xm}^{\bar{t}})$, we have $j = i + 1$ hence $V(S_{xm}^{\bar{t}}) = e_1 \cup \dots \cup e_{i-1} \cup \{u, y, u'\} \cup e_{j+1} \cup \dots \cup e_L$. By the above, this yields $(V(S_{xm}^{\bar{t}}) \cap V(D)) \setminus \{x\} \subseteq \{u, u'\}$, hence $(V(S_{xm}^{\bar{t}}) \cap V(P_{dx}^{\bar{v}})) \setminus \{x\} \subseteq \{u, u'\}$. Finally, it is impossible that $u' \in V(P_{dx}^{\bar{v}})$: indeed, this would imply that $V(S_{ym}^{\bar{t}}) \cap V(P_{dx}^{\bar{v}}) = \{u'\}$ and that $u' \neq v$ hence $v \notin V(S_{ym}^{\bar{t}})$, so the walk $\overrightarrow{dP_{dx}^{\bar{v}}x|_{\{u'\}}} \oplus \overrightarrow{u'S_{ym}^{\bar{t}}m}$ would represent a dm -snake

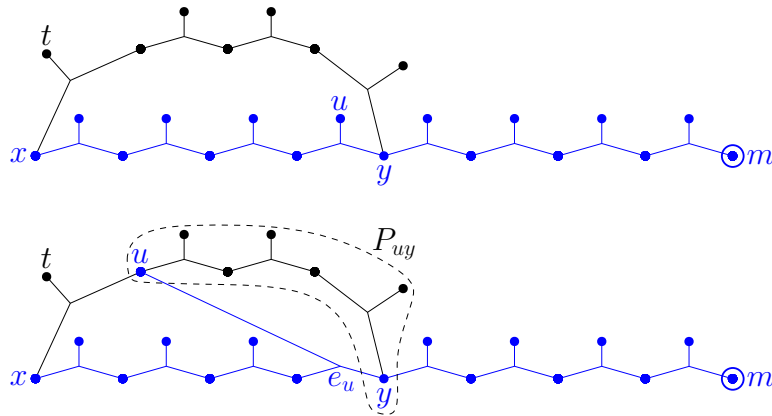
Figure II.21: The xm -snake S_{xm}^t .

not containing v , contradicting (C). Therefore $(V(S_{xm}^t) \cap V(P_{dx}^{\bar{v}})) \setminus \{x\} \subseteq \{u\}$, which concludes the proof of the claim. \square

We can now conclude by exhibiting an m -tadpole in H , which contradicts Proposition II.1.19 since H is not a trivial Maker win and $J_1(\mathcal{D}_1, H)$ holds.

Let $P_{xy}^{\bar{v}}$ be as in Claim 8, and let $S_{xm}^t, P_{xy}^t, S_{ym}^t, u$ be as in Claim 9. We have $V(P_{xy}^{\bar{v}}) \subseteq V(P_{dx}^{\bar{v}}) \cup \{y\}$ by Claim 8, and $(V(S_{xm}^t) \cap V(P_{dx}^{\bar{v}})) \setminus \{x\} \subseteq \{u\}$ by Claim 9: therefore, $V(S_{xm}^t) \cap V(P_{xy}^{\bar{v}}) = \{x, y\}$ or $V(S_{xm}^t) \cap V(P_{xy}^{\bar{v}}) = \{x, y, u\}$.

- Case 1: $V(S_{xm}^t) \cap V(P_{xy}^{\bar{v}}) = \{x, y\}$. The walk $y \overleftarrow{S_{ym}^t} m \oplus y \overleftarrow{P_{xy}^t} x \oplus x \overrightarrow{P_{xy}^{\bar{v}}} y$ clearly represents an m -tadpole (see Figure II.22, top).

Figure II.22: Conclusion of Lemma II.1.40. The represented chains are S_{xm}^t (horizontal) and $P_{xy}^{\bar{v}}$. Top: Case 1. Bottom: Case 2.

- Case 2: $V(S_{xm}^t) \cap V(P_{xy}^{\bar{v}}) = \{x, y, u\}$. Let e_u be the edge of S_{xm}^t containing u , and let P_{uy} be the unique uy -chain in $P_{xy}^{\bar{v}}$. Since $u \neq t$, we have $x \notin V(P_{uy})$, therefore the walk $y \overleftarrow{S_{ym}^t} m \oplus (y, e_u, u) \oplus u \overrightarrow{P_{uy}} y$ represents an m -tadpole (see Figure II.22, bottom). \blacksquare

Corollary II.1.41. *We have $\text{sep}_H(s, m) = \text{sep}_H(y, m)$.*

Proof. Since $s \in I_{H+z}(z\mathcal{D}_1(H))$, there can be no zm -snake in H that does not contain s . Therefore, we can apply the contrapositive of Lemma II.1.40 to $d = z$ and $v = s$, which tells us that $\text{sep}_H(s, m) \leq \text{sep}_H(y, m)$ or there is no zx -chain in D that does not contain s . We know the latter is false: such a chain $P^{\bar{s}}$ is given by Proposition II.1.27. Therefore, the conclusion is that $\text{sep}_H(s, m) \leq \text{sep}_H(y, m)$ hence $\text{sep}_H(s, m) = \text{sep}_H(y, m)$ by Lemma II.1.38. \blacksquare

The previous corollary has a simple consequence which we will use extensively:

Proposition II.1.42. *There is no s -tadpole in D . In particular, any z -tadpole T in D satisfies $s \in \text{out}(C_T)$.*

Proof. We have $\text{sep}_H(s, m) = \text{sep}_H(y, m)$ by Corollary II.1.41, so there is no s -tadpole in D according to Lemma II.1.38. Let T be a z -tadpole in D : we know $s \in V(T)$ by definition of s .

If we had $s \notin \text{out}(C_T)$, then there would be an s -tadpole in $T \subseteq D$ by Substructure Lemma I.3.31, therefore $s \in \text{out}(C_T)$. ■

For example, one application of the previous proposition is the following:

Proposition II.1.43. *There is no z -cycle of length 2 in D .*

Proof. Suppose for a contradiction that there exists a z -cycle C of length 2 in D . We have $s \in \text{out}(C)$ by Proposition II.1.42: write $V(C) = \{z, s, u, v\}$ and $E(C) = \{e_1, e_2\}$ where $e_1 = \{z, u, s\}$ and $e_2 = \{z, u, v\}$. By Proposition II.1.26(b), we know there exists some $K^{\bar{u}} \in \mathcal{O}_{x,z}(D)$ such that $u \notin V(K^{\bar{u}})$.

- First suppose $s \in V(K^{\bar{u}})$. By Substructure Lemma I.3.25 (if $K^{\bar{u}}$ is a zx -chain) or Substructure Lemma I.3.30 (if $K^{\bar{u}}$ is a z -tadpole), there exists a zs -chain P_{zs} in $K^{\bar{u}}$. Since $u \notin V(P_{zs})$, we get an s -cycle $P_{zs} \cup e_1$ in D , contradicting Proposition II.1.42.
- Now suppose $s \notin V(K^{\bar{u}})$. Since $s \in I_{H+z}(z\mathcal{D}_1(H))$, this implies that $K^{\bar{u}} =: P$ is a zx -chain. If $v \notin V(P)$, then $V(C) \cap V(P) = \{z\}$, therefore $P \cup C$ is an x -tadpole in D , contradicting Proposition II.1.26(f). If $v \in V(P)$, then there exists a zv -chain P_{zv} in P by Substructure Lemma I.3.25, and we get a z -cycle $P_{zv} \cup e_2$ in D that does not contain s , also a contradiction. ■

We can now prove Proposition II.1.39.

Proof of Proposition II.1.39. Let $w \in V(D) \setminus \{x\}$ such that D contains the following three subhypergraphs:

- (i) a z -cycle C containing w ;
- (ii) an xw -chain P_{xw} such that $V(P_{xw}) \cap \text{inn}(C) \subseteq \{w\}$;
- (iii) a w -tadpole T that does not contain s .

We are going to consider C , P_{xw} and T successively. Each of these three objects will imply the existence of some \mathcal{D}_1 -dangers at w , which will improve our upper bound on $I_{H+w}(w\mathcal{D}_1(H))$ until we get the desired conclusion that $I_{H+w}(w\mathcal{D}_1(H)) = \emptyset$.

- 1) Step 1: we show that $I_{H+w}(w\mathcal{D}_1(H)) \subseteq \text{inn}(C) \cup \{s\} \cup (V(H) \setminus V(D))$.

In this step, we use C . Recall that $s \in \text{out}(C)$ by Proposition II.1.42 and that C is of length at least 3 by Proposition II.1.43.

Claim 10. *We have $I_{H+w}(w\mathcal{D}_1(H)) \subseteq I_{C+w}(\mathcal{P}_{ws}(C)) \cup (V(H) \setminus V(D))$.*

Proof of Claim 10. We know $\text{sep}_H(s, m) = \text{sep}_H(y, m)$ by Corollary II.1.41. Let S_{sm} be a shortest sm -snake in H : Lemma II.1.38 thus ensures that $V(S_{sm}) \cap V(D) = \{s\}$ hence $V(S_{sm}) \cap V(C) = \{s\}$. Therefore, any ws -chain P_{ws} in C yields a wm -snake $S_{wm} := P_{ws} \cup S_{sm}$ in H and:

$$\begin{aligned} I_{H+w}(w\mathcal{D}_1(H)) &\subseteq V(S_{wm}) \setminus \{w\} \\ &\subseteq (V(P_{ws}) \setminus \{w\}) \cup (V(S_{sm}) \setminus \{s\}) \\ &\subseteq I_{C+w}(\mathcal{P}_{ws}(C)) \cup (V(H) \setminus V(D)). \end{aligned} \quad \square$$

Using Claim 10, it suffices to show that $I_{C+w}(\mathcal{P}_{ws}(C)) \subseteq \text{inn}(C) \cup \{s\}$. This is a straightforward fact that actually holds in general for any two vertices w, s in a cycle C , but we do give a rigorous proof of it using walks:

- Suppose $w \in \text{inn}(C)$, and write $\overrightarrow{wC} = (w, e_1, \dots, e_L, w)$. Let $1 \leq i \leq L$ be the unique index such that $s \in e_i$. See Figure II.23 (left).

- If $i \in \{1, L\}$, then $w \in e_i$, so e_i is a ws -chain of length 1 in C hence $I_{C+w}(\mathcal{P}_{ws}(C)) \subseteq e_i \setminus \{w\} \subseteq \text{inn}(C) \cup \{s\}$.
- If $i \notin \{1, L\}$, then $P_1 := [(w, e_1, \dots, e_i)]$ and $P_2 := [(w, e_L, e_{L-1}, \dots, e_i)]$ are two ws -chains in C , so $I_{C+w}(\mathcal{P}_{ws}(C)) \subseteq (V(P_1) \cap V(P_2)) \setminus \{w\} = e_i \subseteq \text{inn}(C) \cup \{s\}$.
- Suppose $w \in \text{out}(C)$, and let e be the only edge of C containing w . Write $e = \{w, w_1, w_2\}$ (we have $w_1, w_2 \in \text{inn}(C)$) and $\overrightarrow{w_1 C} = (w_1, e_1, \dots, e_L, w_1)$. We have $e \in \{e_1, e_L\}$: without loss of generality, assume $e = e_1$. Since $L \geq 3$, we have $e_1 \cap e_2 = \{w_2\}$ and $e_1 \cap e_L = \{w_1\}$. Let $1 \leq i \leq L$ be the unique index such that $s \in e_i$. See Figure II.23 (right).
 - If $i = 1$ then $w = s$, so $[(w)]$ is a ws -chain of length 0 in C , hence $I_{C+w}(\mathcal{P}_{ws}(C)) = \emptyset \subseteq \text{inn}(C) \cup \{s\}$.
 - If $i \in \{2, L\}$, then $[(w, e_1, e_i, s)]$ is a ws -chain of length 2 in C (because $L \geq 3$), so $I_{C+w}(\mathcal{P}_{ws}(C)) \subseteq (e_1 \cup e_i) \setminus \{w\} \subseteq \text{inn}(C) \cup \{s\}$.
 - If $3 \leq i \leq L - 1$, then C contains two ws -chains $P_1 := [(w, e_1, e_2, \dots, e_i)]$ and $P_2 := [(w, e_1, e_L, e_{L-1}, \dots, e_i)]$, so $I_{C+w}(\mathcal{P}_{ws}(C)) \subseteq (V(P_1) \cap V(P_2)) \setminus \{w\} = (e_1 \cup e_i) \setminus \{w\} \subseteq \text{inn}(C) \cup \{s\}$.

We have $I_{C+w}(\mathcal{P}_{ws}(C)) \subseteq \text{inn}(C) \cup \{s\}$ in all cases, so this concludes Step 1.

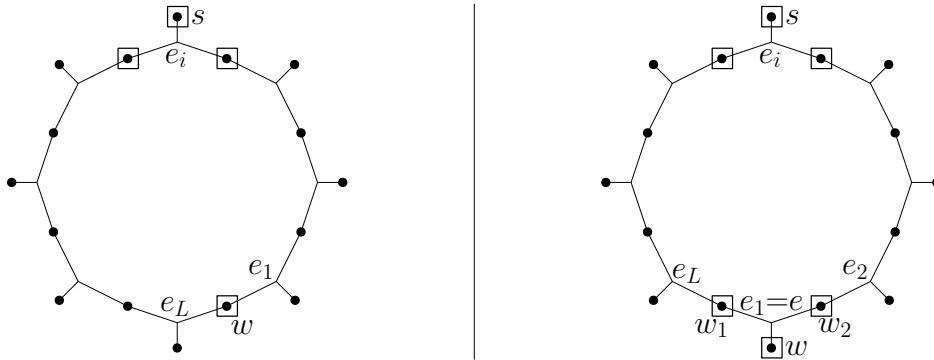


Figure II.23: The cycle C if $w \in \text{inn}(C)$ (left) or $w \in \text{out}(C)$ (right). The vertices in square boxes represent the intersection of the ws -chains in C .

- 2) Step 2: we show that $I_{H+w}(w\mathcal{D}_1(H)) \subseteq \{s\} \cup (V(H) \setminus V(D))$.
In this step, we use P_{xw} . Comparing with Step 1, we need to show that $I_{H+w}(w\mathcal{D}_1(H))$ is disjoint from $\text{inn}(C)$. Let $v \in \text{inn}(C)$. If $v = w$, then obviously $v \notin I_{H+w}(w\mathcal{D}_1(H))$, so assume $v \neq w$. By definition of P_{xw} , we then have $v \notin V(P_{xw})$, therefore Lemma II.1.40 applies with: $d = w$, our vertex v , and $P_{dx} = P_{xw}$. We get a wm -snake in H that does not contain v , hence $v \notin I_{H+w}(w\mathcal{D}_1(H))$, which concludes Step 2.
- 3) Step 3: we show that $I_{H+w}(w\mathcal{D}_1(H)) = \emptyset$.
In this step, we use T . We already know that $I_{H+w}(w\mathcal{D}_1(H)) \subseteq \{s\} \cup (V(H) \setminus V(D))$. Moreover, $I_{H+w}(w\mathcal{D}_1(H)) \subseteq V(T)$ because T is a w -tadpole, where $V(T)$ is disjoint from $\{s\} \cup (V(H) \setminus V(D))$ by definition. In conclusion, $I_{H+w}(w\mathcal{D}_1(H)) = \emptyset$. ■

Our goal is now to show that, for a suitable vertex w , D contains all three subhypergraphs listed in Proposition II.1.39. A lot of the work has already been done through Proposition II.1.32: we now separate the case where D is of type (1) from the case where D is of type (2).

II.1.7.3 Finishing the proof when D is of type (2)

We first suppose that D is of type (2). By definition (recall Proposition II.1.32), this means D contains a z -cycle C such that $x \notin V(C)$ as well as an xw -chain P_{xw} , for some $w \in V(C)$, such

that $V(P_{xw}) \cap V(C) = \{w, w'\}$ where $w' := o(w, \overleftarrow{xP_{xw}w})$.

Recall that C is of length at least 3 by Proposition II.1.43, and that $s \in \text{out}(C)$ by Proposition II.1.42. Define $e^* := \overrightarrow{\text{end}(xP_{xw}w)}$.

Note that $s \notin \{w, w'\}$: indeed, let P be a ww' -chain in C (which exists by Substructure Lemma I.3.28), if we had $s \in \{w, w'\}$ then $P \cup e^*$ would be an s -cycle in D , contradicting Proposition II.1.42. Therefore, since $s \in \text{out}(C)$, Substructure Lemma I.3.27 ensures that there exists a unique ww' -chain $P_{ww'}$ in C that does not contain s .

Define $C' := P_{ww'} \cup e^*$: C' is both a w -cycle and a w' -cycle in D , and it does not contain s . Moreover, we have $z \notin V(P_{ww'})$: indeed, we would otherwise have $z \in \text{inn}(C) \cap V(P_{ww'}) = \{w, w'\} \cup \text{inn}(P_{ww'}) = \text{inn}(C')$, so C' would be a z -cycle not containing s , contradicting Proposition II.1.42. See Figure II.24.

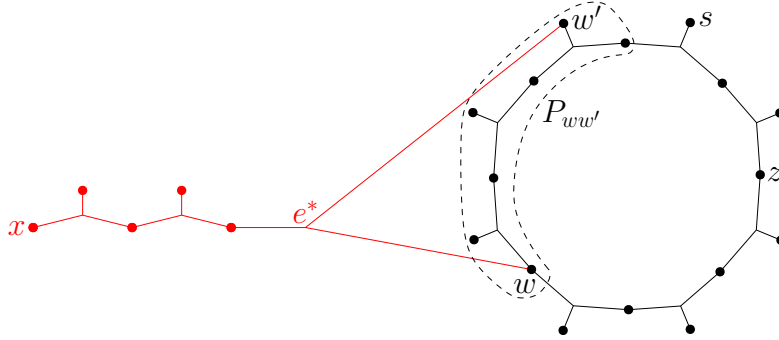


Figure II.24: The cycle C (on the far right) and the chain P_{xw} . In this drawing, we have $w \in \text{inn}(C)$ and $w' \in \text{out}(C)$.

Claim 11. $w \in \text{out}(C)$ or $w' \in \text{out}(C)$.

Proof of Claim 11. Suppose for a contradiction that $w, w' \in \text{inn}(C)$. Write $\overrightarrow{zC} = (z, e_1, \dots, e_L, z)$. Since $L \geq 3$ and $w, w' \in \text{inn}(C) \setminus \{z\}$, there exist $1 \leq i \neq i' \leq L-1$ such that $e_i \cap e_{i+1} = \{w\}$ and $e_{i'} \cap e_{i'+1} = \{w'\}$. Since w and w' have symmetrical roles, assume $i < i'$. Let $1 \leq j \leq L$ be the unique index such that $s \in e_j$.

Since e_1 and e_L are the only edges of C containing z , the ww' -chain represented by the walk $(w, e_{i+1}, \dots, e_{i'}, w')$ does not contain z , so it is necessarily $P_{ww'}$ according to the unicity statement of Substructure Lemma I.3.27. Since $s \notin V(P_{ww'})$, this yields $1 \leq j \leq i$ or $i' + 1 \leq j \leq L$: by symmetry, assume $i' + 1 \leq j \leq L$. Then $(z, e_1, \dots, e_i, w) \oplus \overrightarrow{wP_{ww'}w'} \oplus (w', e^*, w)$ represents a z -tadpole not containing e_j i.e. not containing s , a contradiction which concludes the proof of the claim. \square

Using Claim 11, assume $w' \in \text{out}(C)$ by symmetry. This ensures that $V(P_{xw}) \cap \text{inn}(C) \subseteq \{w\}$. In conclusion, we can apply Proposition II.1.39 to the vertex w , with: the z -cycle C containing w , the xw -chain P_{xw} which satisfies $V(P_{xw}) \cap \text{inn}(C) \subseteq \{w\}$, and the w -cycle C' which does not contain s . We get $I_{H+w}(w\mathcal{D}_1(H)) = \emptyset$, contradicting property $J_1(\mathcal{D}_1, H)$. This ends the proof of Lemma II.1.33 when D is of type (2).

II.1.7.4 Finishing the proof when D is of type (1)

We now suppose that D is of type (1). By definition (see Proposition II.1.32), this means D contains:

- a z -cycle C such that $x \notin V(C)$;
- an xw -chain P_{xw} for some $w \in \text{out}(C)$ such that $V(P_{xw}) \cap V(C) = \{w\}$;

- some $K \in \mathcal{O}_{x,z}(D)$ such that $V(K) \cap V(P_{xw}) \neq \emptyset$ and $\{w_1, w_2\} \not\subseteq V(K)$ where $e = \{w, w_1, w_2\}$ denotes the unique edge of C containing w .

Recall that C is of length at least 3 by Proposition II.1.43 and that $s \in \text{out}(C)$ by Proposition II.1.42.

Since C contains w and $V(P_{xw}) \cap \text{inn}(C) = \emptyset \subseteq \{w\}$, the only subhypergraph in D that we are missing to apply Proposition II.1.39 is a w -tadpole that does not contain s . The rest of the proof consists in finding a w -cycle in D that does not contain s .

Claim 12. There exists a w -chain P_w in D such that:

- The only edge of P_w that intersects $V(C) \setminus \{w\}$ is $e^* := \text{end}(\overrightarrow{wP_w})$. In particular: $|V(P_w) \cap (V(C) \setminus \{w\})| = |e^* \cap (V(C) \setminus \{w\})| \in \{1, 2\}$.*
- $\{w_1, w_2\} \not\subseteq V(P_w)$.*
- $s \notin V(P_w)$.*

Proof of Claim 12. Since $V(K) \cap V(P_{xw}) \neq \emptyset$, the projection $\mathbf{P}_{V(P_{xw})}(z, K)$ is well defined. There is no x -tadpole in $D \supseteq P_{xw} \cup \mathbf{P}_{V(P_{xw})}(z, K)$ by Proposition II.1.26(f), so Union Lemma I.3.38 with $a = w$, $b = x$ and $c = z$ ensures that $P_{xw} \cup \mathbf{P}_{V(P_{xw})}(z, K)$ contains a wz -chain P_{wz} . Now, since $V(P_{wz}) \cap (V(C) \setminus \{w\}) \supseteq \{z\} \neq \emptyset$, the projection $P_w := \mathbf{P}_{V(C) \setminus \{w\}}(w, P_{wz})$ is well defined.

Define $e^* := \text{end}(\overrightarrow{wP_w})$: by definition of a projection, the edge e^* is the only edge of P_w that intersects $V(C) \setminus \{w\}$, and $|e^* \cap (V(C) \setminus \{w\})| \in \{1, 2\}$, hence item (a).

Since $P_w \subseteq P_{wz} \subseteq K \cup P_{xw}$, where $\{w_1, w_2\} \not\subseteq V(K)$ by definition of K and $\{w_1, w_2\} \cap V(P_{xw}) = \emptyset$, we have $\{w_1, w_2\} \not\subseteq V(P_w)$ i.e. item (b).

Finally, suppose for a contradiction that $s \in V(P_w)$:

- First suppose $s = w$. By Substructure Lemma I.3.26, C^{-s} is a w_1w_2 -chain. By Union Lemma I.3.38 with $a = w_1$, $b = w_2$ and $c = w = s$, $C^{-s} \cup P_w$ contains an sw_1 -chain or an sw_2 -chain. Take a shortest sw_1 -chain or sw_2 -chain in $C^{-s} \cup P_w$: by symmetry, assume it is an sw_1 -chain P_{sw_1} . The minimality of the length ensures that either $w_2 \notin V(P_{sw_1})$ or $w_2 = o(w_1, \overleftarrow{sP_{sw_1}w_1})$.
 - If $w_2 \notin V(P_{sw_1})$, then $P_{sw_1} \cup e$ is an s -cycle in D , contradicting Proposition II.1.42.
 - If $w_2 = o(w_1, \overleftarrow{sP_{sw_1}w_1})$, then $\text{end}(\overrightarrow{sP_{sw_1}w_1})$ is an edge of $C^{-s} \cup P_w$ containing both w_1 and w_2 . However, there is no such edge in C^{-s} because C^{-s} is a w_1w_2 -chain of length at least 2 (indeed, recall that C is of length at least 3), and there is no such edge in P_w because $\{w_1, w_2\} \not\subseteq V(P_w)$ by item (b). We have a contradiction.
- Now suppose $s \neq w$. By item (a), $e^* := \text{end}(\overrightarrow{wP_w})$ is the only edge of P_w that intersects $V(C) \setminus \{w\}$, so P_w is a ws -chain and either $V(P_w) \cap V(C) = \{w, s\}$ or $V(P_w) \cap V(C) = \{w, s, o(s, \overrightarrow{sP_w w})\}$.
 - If $V(P_w) \cap V(C) = \{w, s\}$, then let P_{ws} be a ws -chain in C (which exists by Substructure Lemma I.3.28): since P_w and P_{ws} are both ws -chains and $V(P_w) \cap V(P_{ws}) = \{w, s\}$, we get an s -cycle $P_w \cup P_{ws}$, contradicting Proposition II.1.42.
 - If $V(P_w) \cap V(C) = \{w, s, t\}$ where $t := o(s, \overrightarrow{sP_w w}) \in e^*$, then let P_{st} be an st -chain in C that does not contain w (which exists by Substructure Lemma I.3.27 because $w \in \text{out}(C)$): since $V(P_{st}) \cap e^* = \{s, t\}$, we get an s -cycle $P_{st} \cup e^*$, contradicting Proposition II.1.42.

We have a contradiction in all cases, hence item (c). \square

From now on, the action takes place in $C \cup P_w$ exclusively: we are going to exhibit a w -cycle in $C \cup P_w$ that does not contain s . The idea is simply to get such a cycle by using P_w to go from w to C and then rejoining w by rotating along C in the correct direction so as to avoid s

(for instance, see Figure II.25, left and middle). This is always possible, unless this direction is blocked by a cycle of length 2, which cannot happen because there would then be an s -tadpole, contradicting Proposition II.1.42 (for instance, see Figure II.25, right). We now carry out the rigorous proof of this, distinguishing between two cases.

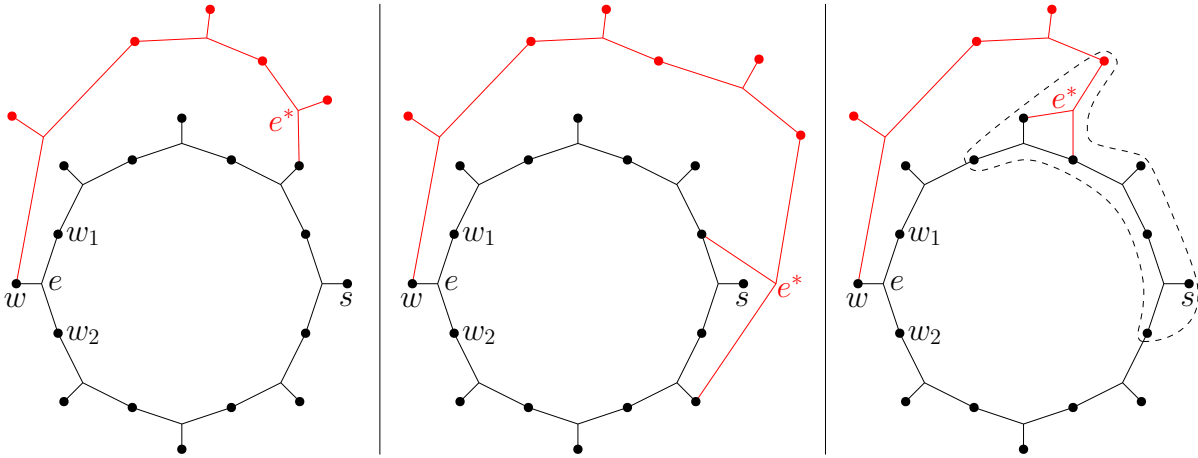


Figure II.25: Represented here are C and P_w . In the left and middle examples, there is a w -cycle not containing s . In the right example, there is none but there is an s -tadpole (highlighted).

1) Case 1: $w_1 \in V(P_w)$ or $w_2 \in V(P_w)$.

By symmetry, assume $w_1 \in V(P_w)$. By Claim 12(b), we have $\{w_1, w_2\} \not\subseteq V(P_w)$ hence $w_2 \notin V(P_w)$. Therefore, P_w is a ww_1 -chain that does not contain w_2 , so $C' := P_w \cup e$ is a w -cycle. Moreover, $s \notin V(C') = V(P_w) \cup \{w_2\}$: indeed, we have $s \notin V(P_w)$ by Claim 12(c), and $s \in \text{out}(C)$ whereas $w_2 \in \text{inn}(C)$. Therefore, C' is the desired cycle.

2) Case 2: $w_1, w_2 \notin V(P_w)$.

By Substructure Lemma I.3.26, C^{-w} is a w_1w_2 -chain. Write $w_1C^{-w}w_2 = (w_1, e_1, \dots, e_L, w_2)$. We have $s \in \text{out}(C)$, moreover $s \neq w$ by Claim 12(c), so there exists a unique index $1 \leq i \leq L$ such that $s \in e_i$. If $i \neq 1$, define s_1 as the only vertex in $e_{i-1} \cap e_i$ and $P_1 := [(w_1, e_1, \dots, e_{i-1}, s_1)]$, otherwise define $s_1 = w_1$ and $P_1 := [(w_1)]$. Similarly, if $i \neq L$, define s_2 as the only vertex in $e_i \cap e_{i+1}$ and $P_2 := [(w_2, e_L, e_{L-1}, \dots, e_{i+1}, s_2)]$, otherwise define $s_2 = w_2$ and $P_2 := [(w_2)]$. For all $j \in \{1, 2\}$, P_j is a $w_j s_j$ -chain in C , and $V(P_1) \cap V(P_2) = \emptyset$. These notations are summed up in Figure II.26.

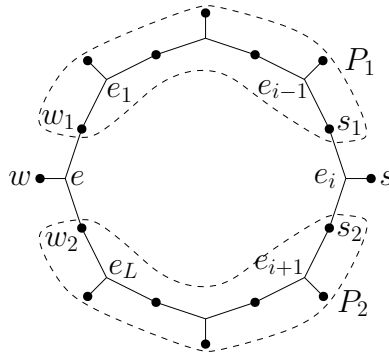


Figure II.26: The cycle C .

By Claim 12(a), we have $|V(P_w) \cap (V(C) \setminus \{w\})| \in \{1, 2\}$. Note that $V(C) \setminus \{w\} = V(P_1) \cup V(P_2) \cup \{s\}$. Since $s \notin V(P_w)$ by Claim 12(c), we obtain that $|V(P_w) \cap V(P_1)| \in \{1, 2\}$ or $|V(P_w) \cap V(P_2)| \in \{1, 2\}$. By symmetry, assume that $|V(P_w) \cap V(P_1)| \in \{1, 2\}$.

- First suppose $|V(P_w) \cap V(P_1)| = 1$. Let u be the only vertex in $V(P_w) \cap V(P_1)$: in particular, P_w is a wu -chain. Recall that $w_2 \notin V(P_w)$ by assumption, moreover $w_2 \notin V(P_1)$ by definition of P_1 . Therefore, the walk $(w, e, w_1) \oplus \overrightarrow{w_1 P_1|_{\{u\}}} \oplus \overrightarrow{u P_w w}$ represents a w -cycle in $C \cup P_w$, which does not contain s since $s \notin e \cup V(P_1) \cup V(P_w)$. This is the desired cycle.
- Now suppose $|V(P_w) \cap V(P_1)| = 2$. Since $V(P_1) \cap V(P_2) = \emptyset$, this yields $V(P_w) \cap V(P_2) = \emptyset$ by Claim 12(a). In particular $s_2 \notin V(P_w)$, so there cannot be an s_1 -tadpole T in $P_w \cup P_1$: indeed, since $s, s_2 \notin V(P_w) \cup V(P_1)$, the walk $(s, e_i, s_1) \oplus \overrightarrow{s_1 T}$ would otherwise represent an s -tadpole in D , contradicting Proposition II.1.42. Therefore, Union Lemma I.3.38 with $a = w_1$, $b = s_1$ and $c = w$ ensures that $P_1 \cup P_w$ contains a ww_1 -chain P_{ww_1} . Since $w_2 \notin V(P_1)$, and $w_2 \notin V(P_w)$ by assumption, the walk $\overrightarrow{w P_{ww_1} w_1} \oplus (w_1, e, w)$ represents a w -cycle, which does not contain s since $s \notin e \cup V(P_{ww_1})$. This is the desired cycle.

In conclusion, Proposition II.1.39 applies and yields $I_{H+w}(w\mathcal{D}_1(H)) = \emptyset$, contradicting property $J_1(\mathcal{D}_1, H)$. This ends the proof of Lemma II.1.33, so that all results of this section are now proved.

II.1.8 Conclusion and prospects

Section I.2 introduced a toolkit for a structural study of the Maker-Breaker game, based on dangers and their intersections. In this section, we have applied these ideas in 3-uniform marked hypergraphs i.e. hypergraphs of rank 3. We have shown that Breaker wins if and only if he can destroy the \mathcal{D}_0^{*2} -dangers in the first round, and more precisely, that Breaker's first pick is a winning move if and only if it destroys the \mathcal{D}_0^{*2} -dangers at Maker's first pick. This is equivalent to saying that Breaker can destroy the \mathcal{D}_0 -dangers during three rounds i.e. can ensure that no nunchaku or necklace is present at the end of each of the first three rounds.

In particular, all specific instances of games with winning sets of size at most 3 which were studied on their own fall within the scope of our results. An example is the H -game on the edge set of a general graph G in the case $|E(H)| \leq 3$, like the triangle game or the P_4 -game for instance. Regarding the latter, [DGM⁺23] provides a different structural characterization of the outcome which is read on the graph G directly. Another example is the total domination game on a general graph G of maximum degree at most 3, notably cubic graphs, for which [FM22] gave a partial structural characterization of the outcome.

Can we find new classes of hypergraphs such that property $J_r(\mathcal{F}, \cdot)$ characterizes Breaker wins, for some well-chosen family of dangers \mathcal{F} and some constant number of rounds r ? Hypergraphs of rank 4 immediately come to mind, possibly with dangers based on 4-uniform forcing paths, the same way \mathcal{D}_0 was based on nunchakus. However, the proof in the 3-uniform case heavily relied on our unions lemmas from Subsection I.3.2, whose 4-uniform counterpart would likely be too complicated to be exploitable. Instead of raising the rank, another avenue may be to introduce a bias, and look at the biased game on 3-uniform marked hypergraphs from a danger-based point of view.

Subsection II.1.1 mentioned a more general version of the game, played on a CNF formula instead of a hypergraph. The concept of danger and the results from Subsection I.2.2 should translate well to this version, except that there would be two types of dangers at x depending on which value False must assign to x . Property $J_r(\cdot, \cdot)$ should then be checked for both types of dangers. It is possible that our proofs generalize to this version, in which case this would show that, apart from some trivial cases, True wins if and only if he can break any manriki that appears during the first three rounds of play, thus fully validating Rahman and Watson's

conjecture [RW20]. However, we have not looked into it. For now, we have only proved that this conjecture is true for positive 3-CNF formulas.

Finally, our structural result suggests an algorithm solving the Maker-Breaker game on hypergraphs of rank 3. Indeed, we now know that this problem reduces to the problem of detecting nunchakus and necklaces in a 3-uniform marked hypergraph, which itself boils down to determining whether there exists an ab -chain for given vertices a and b in a 3-uniform hypergraph. Therefore, the chain existence problem, which has not been studied before, now becomes a main preoccupation for us. We thus proceed with Section II.2, in which we explore the structure of the connected components associated with chains in a 3-uniform hypergraph (actually generalized to k -uniform hypergraphs). The algorithmic rewards will be reaped in Chapter III.

II.2 Structure of a $(k - 2)$ -linear connected component[†]

II.2.1 Presentation of the problem and state of the art

With motivation coming from its crucial importance in the Maker-Breaker game on hypergraphs of rank 3, which we have just established in Section II.1, we are interested in the chain existence problem in 3-uniform hypergraphs. The tools that we are going to develop actually generalize to the $(k - 2)$ -linear path existence problem in k -uniform hypergraphs. In other words, we forbid edge intersections that are maximum i.e. of size $k - 1$.

Studies around the existence of paths with restriction on the size of edge intersections do exist in the hypergraph theory literature, however these are all extremal results. Let us mention some of them. Linear simple paths in k -uniform hypergraphs are the most studied [FJS14][GLS20], with particular emphasis on $k = 3$ i.e. the case of chains [OS14][Jac15][JPR16][WP21]: for instance, [JPR16] determines the Turán number of the chain of length 3, so that a 3-uniform hypergraph on $n \geq 8$ vertices with at least $\binom{n-1}{2}$ edges necessarily contains a chain of length 3. Simple paths where any two consecutive edges must intersect on at least t vertices [GKL10], or exactly t vertices [Tom12][DLM⁺17], have also been studied.

In general hypergraphs, which are potentially sparse, the existence of such paths cannot be determined using extremal results. The goal of this section is to initiate this study by giving a fine description of the connected components associated to $(k - 2)$ -linear paths: how they are structured internally, and how they interact with the rest of the hypergraph.

Definition II.2.1. Let H be a hypergraph and let $\lambda \geq 1$ be an integer. For all $x^* \in V(H)$, we define the λ -linear connected component of x^* in H as:

$$CC_H^\lambda(x^*) := \{y \in V(H) \text{ such that there exists a } \lambda\text{-linear } x^*y\text{-path in } H\}.$$

By extension, when it comes to structural considerations, the term may refer to the induced subhypergraph $H[CC_H^\lambda(x^*)]$. In the absence of ambiguity regarding the considered k -uniform hypergraph H and in the case $\lambda = k - 2$, as will be the case throughout this section, we will simply write $CC(x^*)$ to alleviate notations.

It should be noted that these particular connected components do not form a partition of the vertex set. Indeed, $(k - 2)$ -linear connectivity defines a relation that is reflexive and symmetric but not transitive: for $k = 3$, we have already seen that the union of an ab -chain and a bc -chain

[†]These results are part of the article [GGS22a], which has been submitted to *Discrete Mathematics & Theoretical Computer Science*.

does not necessarily contain an ac -chain, and it is not difficult to generalize the counterexample to $k \geq 4$. This is why our problem is nontrivial from an algorithmic point of view: we will address algorithmic questions in Section III.1.

Also recall that, by Proposition I.3.11, any $(k - 2)$ -linear x^*y -path contains a $(k - 2)$ -linear x^*y -simple-path. Therefore, it suffices to consider simple paths, which are more natural.

II.2.2 The archipelago structure

From now on, we fix an integer $k \geq 3$, a k -uniform hypergraph H , and a vertex $x^* \in V(H)$ whose $(k - 2)$ -linear connected component we are going to study.

II.2.2.1 An overview

Let $A := H[CC(x^*)]$. This segment aims at presenting the challenges at hand and providing an intuition of the "archipelago" structure of A that we are going to establish. Since the case $k = 3$ encapsulates the idea and makes it easier to visualize things, assume $k = 3$: we are then interested in linear simple paths i.e. chains.

Our problem is about chains between given vertices in H . As we know, a major difficulty is that the union of an ab -chain and a bc -chain is not necessarily an ac -chain, even if the edges incident to b in the respective chains are the only ones that intersect: indeed, they may intersect on two vertices, as in Figure II.27.

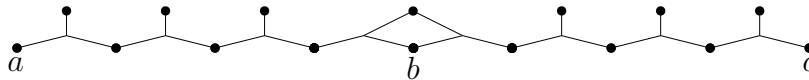


Figure II.27: An ab -chain and a bc -chain do not necessarily form an ac -chain.

Therefore, just as important as the notion of chain connecting two vertices, is the more general notion of chain connecting two sets of vertices X and Y of size 1 or 2, where if X (resp. Y) is a pair then the chain must contain one of its vertices and avoid the other. This way, even an edge containing all of X (resp. all of Y) would prolong the chain in a linear manner. Using our vocabulary around walks, the existence of such a chain for given X and Y can be expressed as follows: there exists a walk \vec{W} , whose elements are edges of H , such that $(X) \oplus \vec{W} \oplus (Y)$ is a linear simple walk. Indeed, the fact that this is a walk means that the chain $P := [\vec{W}]$ connects X and Y , while linearity and simplicity mean that P only contains one vertex of X and one vertex of Y . Such a walk \vec{W} will be deemed (X, Y) -compatible. See Figure II.28 for an illustration.

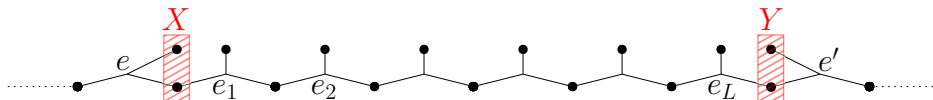


Figure II.28: The walk (e_1, \dots, e_L) is (X, Y) -compatible, so that the edges e and e' prolong the induced chain in a linear manner.

Since we are considering the linear connected component of a fixed vertex x^* , the case $X = \{x^*\}$ is of natural interest. The singletons $Y = \{y\}$ such that there exists an (x^*, y) -compatible walk are precisely the elements of $CC(x^*)$. The pairs Y such that there exists an (x^*, Y) -compatible walk are those that can be "separated", or to state the negative: the pairs Y such that there does not exist an (x^*, Y) -compatible walk are those that cannot be separated from the point of

view of x^* , in the sense that when following an x^* -chain, it is impossible to reach one vertex of Y without reaching the other at the same time. Let $\varepsilon_2, \dots, \varepsilon_N$ denote all the pairs Y in A such that there does not exist an (x^*, Y) -compatible walk. For reasons that will become clearer later, we start the indexing at 2 to make way for the special $\varepsilon_1 := \{x^*\}$.

The importance of these "inseparable pairs" is apparent when looking at the interface between A and the rest of the hypergraph. Indeed, consider a "cut" edge i.e. an edge $e \in E(H)$ intersecting both $V(A)$ and $V(H) \setminus V(A)$. Let $y \in e \setminus V(A)$. Since $y \notin V(A) = CC(x^*)$, there does not exist an x^*y -chain in H , so it is impossible to prolong an x^* -chain in A with the edge e to reach y . This means that the other two vertices of e are in A and form an inseparable pair. Therefore, all cut edges are of the form $\varepsilon_i \cup \{y\}$ for some $2 \leq i \leq N$ and some $y \notin V(A)$. See Figure II.29.

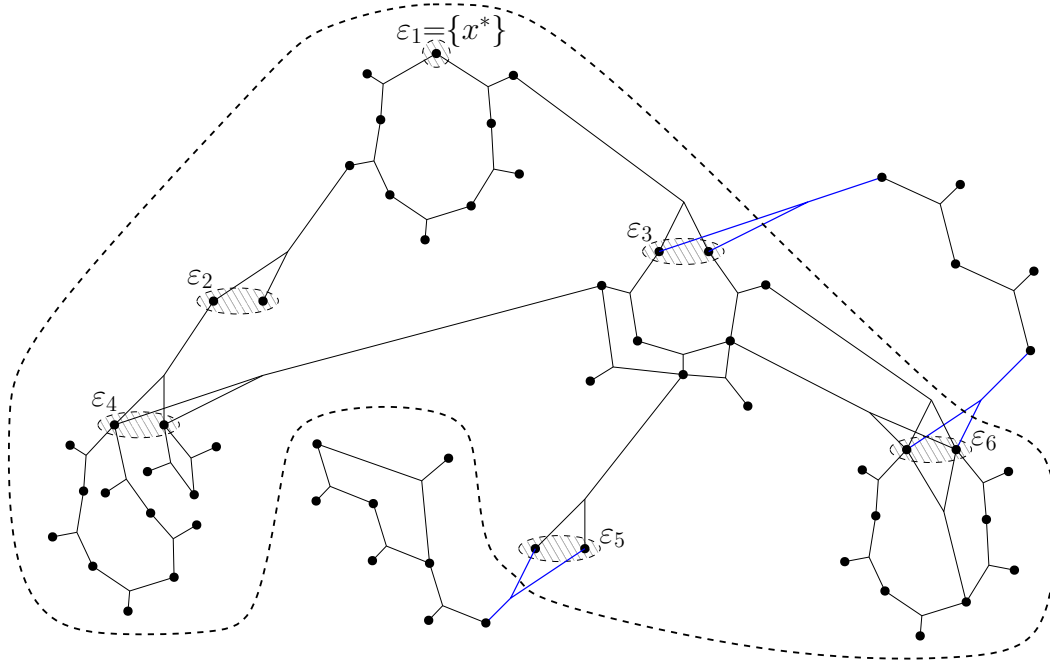


Figure II.29: An example hypergraph H where $k = 3$. Inside the dashed line is A . The blue edges are the cut edges.

As for the inside structure of A , we will see that it is articulated around the ε_i :

- First of all, there is the subhypergraph I_1 containing all that can be reached from $\varepsilon_1 := \{x^*\}$ and without going through any of the other ε_i . Inside I_1 , there exist (x^*, Y) -compatible walks for all Y . This convenient property defines what we will call an *island with entry ε_1* .
- On the other hand, some vertices are only reachable from x^* by going through some of the ε_i ($2 \leq i \leq N$) on the way. For example, consider an (x^*, Y) -compatible walk (e_1, \dots, e_L) that goes through ε_2 but does not intersect $\varepsilon_3, \dots, \varepsilon_N$, as in Figure II.30. Let j be the index such that $\varepsilon_2 \subseteq e_j$. All edges before e_j are in I_1 . All edges after e_j are in the subhypergraph I_2 containing all that can be reached after arriving at ε_2 and without going through any other ε_i ($i \geq 3$). Inside I_2 , there exist (ε_2, Y) -compatible walks for all Y . This convenient property defines what we will call an *island with entry ε_2* .

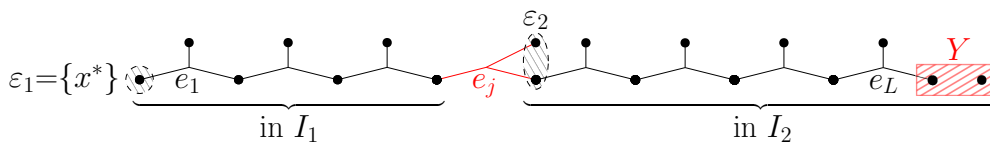


Figure II.30: An (x^*, Y) -compatible walk going through ε_2 .

- All in all, A can be divided into islands I_1, \dots, I_N , with respective entries $\varepsilon_1, \dots, \varepsilon_N$, plus *crossing edges* allowing to jump from one island to the entry of another (in the previous example, the edge e_j allowed to jump from I_1 to I_2 for instance). We will say that A is an *archipelago*. For the hypergraph from Figure II.29, the decomposition into islands is shown in Figure II.31.

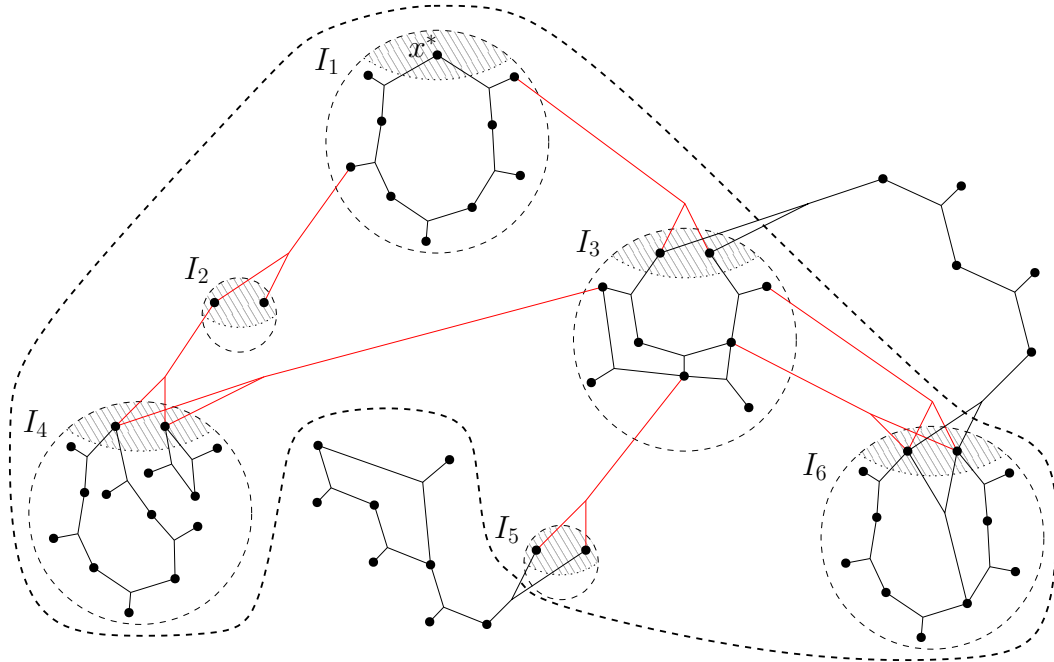


Figure II.31: Decomposition of the archipelago A from Figure II.29. The red edges will always represent the crossing edges. The grey hatched area at the top of an island will always represent its entry.

We now proceed with the formal definitions.

II.2.2.2 Compatible walks

The above principles generalize to any $k \geq 4$. Linearity is replaced by $(k - 2)$ -linearity. The key ability to separate pairs of vertices becomes the ability to separate sets of vertices X and Y of size $k - 1$: if a path connects X and Y while avoiding at least one vertex in each, then it can be prolonged at both ends with edges containing X and Y respectively while maintaining the $(k - 2)$ -linearity. Therefore, the relevant definition of (X, Y) -compatibility is the following:

Definition II.2.2. Let H' be a k -uniform hypergraph, and let $X, Y \subseteq V(H')$ (if $X = \{x\}$ is a singleton then we may write $X = x$, and same goes for Y). An (X, Y) -compatible walk in H' is a walk \vec{W} whose elements are edges of H' and such that $(X) \oplus \vec{W} \oplus (Y)$ is a $(k - 2)$ -linear simple walk.

Remark. Note that the empty walk $()$ is (X, Y) -compatible if and only if $1 \leq |X \cap Y| \leq k - 2$ (in which case it is the only one). In particular, if X and Y are equal and of size $k - 1$, then there can exist no (X, Y) -compatible walk.

As always, we have the choice between walks or paths (or a bit of both) to describe structures. Throughout this section, the sequence of edges will be more relevant than the induced hypergraph itself, so we will express everything in terms of compatible walks. This is possible because all existence questions of $(k - 2)$ -linear simple paths boil down to the existence of compatible walks:

Proposition II.2.3. *Let H' be a k -uniform hypergraph. For all disjoint $X, Y \subseteq V(H')$, the following two assertions are equivalent:*

- (i) *There exists an (X, Y) -compatible walk in H' .*
- (ii) *There exists a $(k - 2)$ -linear xy -simple-path P in H' , for some $x \in X$ and $y \in Y$, such that: $\text{start}(\overrightarrow{xPy})$ is the only edge of P intersecting X and does so on at most $k - 2$ vertices, and $\text{end}(\overrightarrow{xPy})$ is the only edge of P intersecting Y and does so on at most $k - 2$ vertices.*

In particular, for all $x, y \in V(H)$, there exists an (x, y) -compatible walk in H' if and only if there exists a $(k - 2)$ -linear xy -simple-path in H' .

Proof. Let $X, Y \subseteq V(H')$ be disjoint.

- (i) \implies (ii). Let $\overrightarrow{W} = (e_1, \dots, e_L)$ be an (X, Y) -compatible walk in H' i.e. (X, e_1, \dots, e_L, Y) is a $(k - 2)$ -linear simple walk. Since X and Y are disjoint, we know $L \geq 1$. Let $x \in X \cap e_1$ and let $y \in Y \cap e_L$. Clearly, (x, e_1, \dots, e_L, y) is also a $(k - 2)$ -linear simple walk, so by definition it induces a $(k - 2)$ -linear xy -simple-path P . By $(k - 2)$ -linearity and simplicity of (X, e_1, \dots, e_L, Y) : e_1 is the only edge of P intersecting X and does so on at most $k - 2$ vertices, and e_L is the only edge of P intersecting Y and does so on at most $k - 2$ vertices.
- (ii) \implies (i). Let x, y, P as in (ii), and write $\overrightarrow{xPy} = (x, e_1, \dots, e_L, y)$. Since X and Y are disjoint, we know $L \geq 1$. Since the walk \overrightarrow{xPy} is simple by definition, the fact that e_1 (resp. e_L) is the only edge of P intersecting X (resp. Y) ensures that the walk (X, e_1, \dots, e_L, Y) is also simple. Since the walk \overrightarrow{xPy} is $(k - 2)$ -linear, the fact that $|e_1 \cap X| \leq k - 2$ and $|e_L \cap Y| \leq k - 2$ ensures that the walk (X, e_1, \dots, e_L, Y) is also $(k - 2)$ -linear. ■

Corollary II.2.4. *Let H' be a k -uniform hypergraph. For all $x, y \in V(H')$, there exists an (x, y) -compatible walk in H' if and only if there exists a $(k - 2)$ -linear xy -simple-path in H' . In particular, $CC(x^*) = \{y \in V(H) \text{ such that there exists an } (x, y)\text{-compatible walk in } H'\}$.*

Proof. Let $x, y \in V(H)$. If $x = y$, then $()$ is an (x, y) -compatible walk in H' , and the lone vertex $x = y$ constitutes an xy -simple-path in H' . If $x \neq y$, then Proposition II.2.3 with $X = \{x\}$ and $Y = \{y\}$ concludes. ■

We now present three useful "compatibility lemmas", which are useful to deduce the existence of compatible walks from some already known ones. They are illustrated in Figures II.32, II.33 and II.34 respectively.

Lemma II.2.5. *Let H' be a k -uniform hypergraph. Let $\alpha, \beta \subseteq V(H')$ and let \overrightarrow{W} be an (α, β) -compatible walk in H' .*

- *If $\beta' \subseteq V(H')$ satisfies $\beta \subseteq \beta'$ and $\beta' \cap ((\alpha \cup V(\overrightarrow{W})) \setminus \beta) = \emptyset$, then \overrightarrow{W} is an (α, β') -compatible walk in H' .*
- *If $\alpha' \subseteq V(H')$ satisfies $\alpha \subseteq \alpha'$ and $\alpha' \cap ((V(\overrightarrow{W}) \cup \beta) \setminus \alpha) = \emptyset$, then \overrightarrow{W} is an (α', β) -compatible walk in H' .*

Lemma II.2.6. *Let H' be a k -uniform hypergraph. Let $\alpha, \beta \subseteq V(H')$ and let \overrightarrow{W} be an (α, β) -compatible walk in H' .*

- *If $e \in E(H')$ satisfies $\beta \subseteq e$ and $e \cap ((\alpha \cup V(\overrightarrow{W})) \setminus \beta) = \emptyset$, then for all $\gamma \subseteq e \setminus \beta$ such that $1 \leq |\gamma| \leq k - 2$: $\overrightarrow{W} \oplus (e)$ is an (α, γ) -compatible walk in H' .*
- *If $e \in E(H')$ satisfies $\alpha \subseteq e$ and $e \cap ((V(\overrightarrow{W}) \cup \beta) \setminus \alpha) = \emptyset$, then for all $\gamma \subseteq e \setminus \alpha$ such that $1 \leq |\gamma| \leq k - 2$: $(e) \oplus \overrightarrow{W}$ is a (γ, β) -compatible walk in H' .*

Lemma II.2.7. *Let H' be a k -uniform hypergraph. Let $\alpha, \beta, \gamma, \delta \subseteq V(H')$, let \vec{W} be an (α, β) -compatible walk in H' , and let \vec{W}' be a (γ, δ) -compatible walk in H' . We assume that $\alpha \cup V(\vec{W}) \cup \beta$ and $\gamma \cup V(\vec{W}') \cup \delta$ are disjoint. If $e \in E(H')$ satisfies $\beta \subseteq e$, $e \cap ((\alpha \cup V(\vec{W})) \setminus \beta) = \emptyset$, $\gamma \subseteq e$ and $e \cap ((V(\vec{W}') \cup \delta) \setminus \gamma) = \emptyset$, then $\vec{W} \oplus (e) \oplus \vec{W}'$ is an (α, δ) -compatible walk in H' .*

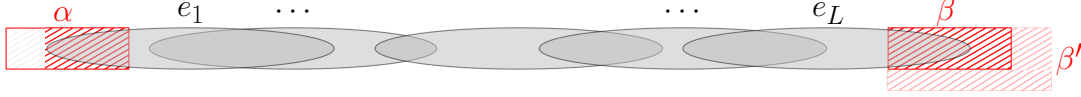


Figure II.32: Illustration of Lemma II.2.5.

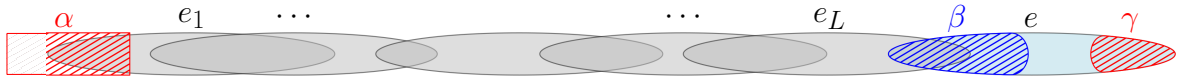


Figure II.33: Illustration of Lemma II.2.6.

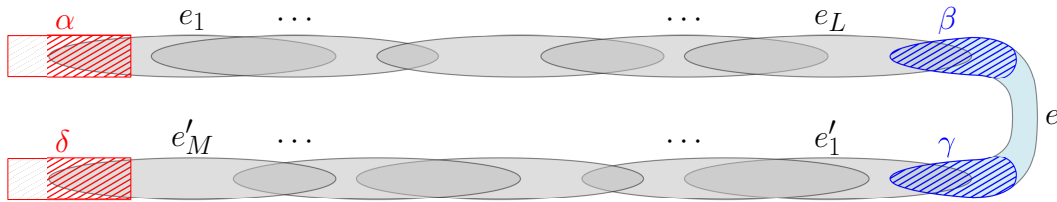


Figure II.34: Illustration of Lemma II.2.7.

Proof of Lemma II.2.5. By symmetry, proving the first assertion is sufficient. Write $\vec{W} = (e_1, \dots, e_L)$ with $L \geq 0$. The assumption on β' ensures that $\alpha \cap \beta' = \alpha \cap \beta$ and $e_i \cap \beta' = e_i \cap \beta$ for all $1 \leq i \leq L$. Therefore, the fact that $(\alpha) \oplus \vec{W} \oplus (\beta)$ is a $(k-2)$ -linear simple walk immediately implies that $(\alpha) \oplus \vec{W} \oplus (\beta')$ is a $(k-2)$ -linear simple walk as well. ■

Proof of Lemma II.2.7. Write $\vec{W} = (e_1, \dots, e_L)$ and $\vec{W}' = (e'_1, \dots, e'_M)$ with $L, M \geq 0$. We know that $(\alpha) \oplus \vec{W} \oplus (\beta)$ and $(\gamma) \oplus \vec{W}' \oplus (\delta)$ are both $(k-2)$ -linear simple walks and have disjoint vertex sets. Therefore, to show that $(\alpha) \oplus \vec{W} \oplus (e) \oplus \vec{W}' \oplus (\delta)$ also is a $(k-2)$ -linear simple walk, the only intersections that need to be checked are those involving the middle edge e . Since $\beta \subseteq e$ and $e \cap ((\alpha \cup V(\vec{W})) \setminus \beta) = \emptyset$, we have $\alpha \cap e = \alpha \cap \beta$ and $e_i \cap e = e_i \cap \beta$ for all $1 \leq i \leq L$, therefore the fact that $(\alpha) \oplus \vec{W} \oplus (\beta)$ is a $(k-2)$ -linear simple walk ensures that all intersections between e and an element from $(\alpha) \oplus \vec{W}$ are good. Similarly, since $\gamma \subseteq e$ and $e \cap ((V(\vec{W}') \cup \delta) \setminus \gamma) = \emptyset$, we have $e \cap e'_j = \gamma \cap e'_j$ for all $1 \leq j \leq M$ and $e \cap \delta = \gamma \cap \delta$, therefore the fact that $(\gamma) \oplus \vec{W}' \oplus (\delta)$ is a $(k-2)$ -linear simple walk ensures that all intersections between e and an element from $\vec{W}' \oplus (\delta)$ are good. ■

Proof of Lemma II.2.6. By symmetry, proving the first assertion is sufficient. Since $1 \leq |\gamma| \leq k-2$, we know the empty walk $()$ is a (γ, γ) -compatible walk. Applying Lemma II.2.7 with $\delta = \gamma$ and $\vec{W}' = ()$ yields Lemma II.2.6. ■

II.2.2.3 Islands and archipelagos

Definition II.2.8. Let I be a k -uniform hypergraph, and let $\varepsilon \subseteq V(I)$ such that $1 \leq |\varepsilon| \leq k - 1$. We say I is an *island with entry* ε if, for all $X \subseteq V(I)$ satisfying $1 \leq |X| \leq k - 1$ (and $X \neq \varepsilon$ if $|\varepsilon| = k - 1$), there exists an (ε, X) -compatible walk in I .

Example. The *deserted island with entry* ε , where $1 \leq |\varepsilon| \leq k - 1$, is the island I with entry ε defined by $V(I) = \varepsilon$ and $E(I) = \emptyset$. It is an island because, for all $X \subseteq V(I)$ satisfying $1 \leq |X| \leq k - 1$ (and $X \neq \varepsilon$ if $|\varepsilon| = k - 1$), $\vec{W} = ()$ is an (ε, X) -compatible walk in I . This example is illustrated at the far left of Figure II.35.

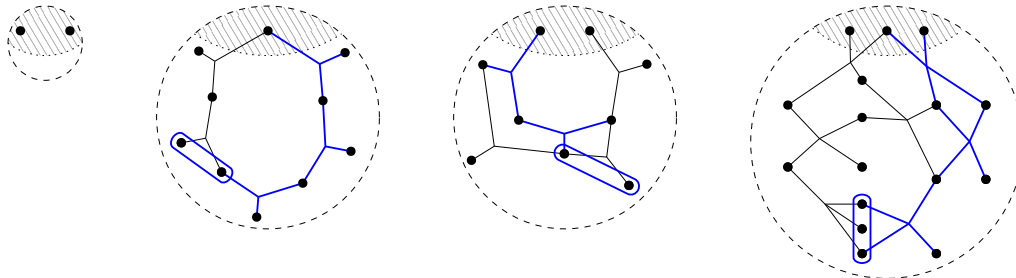


Figure II.35: Some islands for $k = 3$, except the far right one where $k = 4$ (with the same "claw" representation for edges). For three of them, we show an (ε, X) -compatible walk (in blue) for some X of size $k - 1$ (circled in blue).

Definition II.2.9. Let I and I' be disjoint islands in H , where I' has an entry ε of size $k - 1$. An edge $e \in E(H)$ of the form $e = \{x\} \cup \varepsilon$ for some $x \in V(I)$ is called a *crossing edge* from I to I' . We denote by $C(I, I') \subseteq E(H)$ the set of all crossing edges from I to I' in H . If A is a subhypergraph of H containing I and I' , we use the notation $C_A(I, I') := C(I, I') \cap E(A)$.

Remark. The above definition depends on the choice of ε (an island might have several possible entries suiting the definition). However, we will always specify the entries when defining islands and therefore consider crossing edges for those specific entries.

Definition II.2.10. An x^* -*archipelago* is a subhypergraph A of H such that there exist subhypergraphs I_1, \dots, I_N of A that are pairwise disjoint islands with respective entries $\varepsilon_1, \dots, \varepsilon_N$ satisfying the following properties:

- $\varepsilon_1 = \{x^*\}$.
- $|\varepsilon_i| = k - 1$ for all $2 \leq i \leq N$.
- $V(A) = V(I_1) \cup \dots \cup V(I_N)$.
- All edges in $E(A) \setminus (E(I_1) \cup \dots \cup E(I_N))$ are crossing edges between some of the I_i , such that the digraph G defined by $V(G) = \{I_1, \dots, I_N\}$ and $E(G) = \{(I_i, I_j), C_A(I_i, I_j) \neq \emptyset\}$ contains a spanning arborescence rooted at I_1 .

Since x^* is fixed, we usually call A an *archipelago* for short.

Remark. By definition of a crossing edge, there cannot exist a crossing edge from some I_i to I_1 in an archipelago since $|\varepsilon_1| = 1 \neq k - 1$. In other words, I_1 has in-degree zero in G .

Therefore, an archipelago is a union of pairwise disjoint islands and crossing edges between some of them, satisfying specific properties. See Figure II.36 for an example (for clarity, all figures will illustrate the case $k = 3$ from now on). We will later see that an archipelago has a

unique decomposition into islands, but for now we have to give ourselves islands and entries suiting the definition whenever we consider an archipelago.

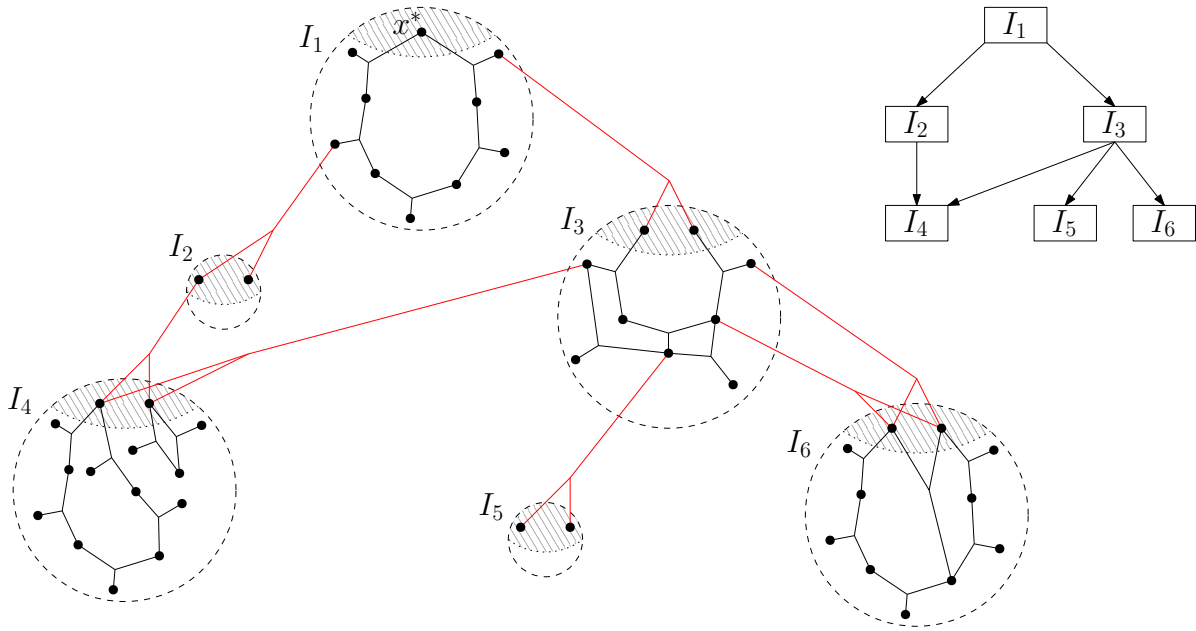


Figure II.36: An archipelago, with the digraph G on the right.

II.2.2.4 Archipelago properties

The next two results show how compatible walks in A are related to directed paths in the digraph G . Obviously, by definition of an archipelago, an (x^*, x) -compatible walk necessarily visits successive islands, using crossing edges to jump from one island to another. The following proposition states that, additionally, a crossing edge can only be used in one direction which is given by the digraph G , therefore each island is entered through its entry (hence the terminology) and it is impossible to reenter an island after leaving it.

Definition II.2.11. Let G be a digraph and let $v, v' \in V(G)$. A *dipath* (short for *directed path*) from v to v' in G is a sequence denoted by $v = v_0 \rightarrow v_1 \rightarrow \dots \rightarrow v_l = v'$ ($l \geq 0$) where $v_0, \dots, v_l \in V(G)$ are pairwise distinct and $(v_i, v_{i+1}) \in E(G)$ for all $0 \leq i \leq l-1$.

Proposition II.2.12. Let A be an archipelago, with $I_1, \dots, I_N, \varepsilon_1, \dots, \varepsilon_N, G$ suiting the definition. Let $x \in V(I_i)$ for some $1 \leq i \leq N$, and let \vec{W} be an (x^*, x) -compatible walk in A . Then the islands visited by \vec{W} form a dipath $I_1 = I_{i_1} \rightarrow \dots \rightarrow I_{i_M} = I_i$ in G , and \vec{W} is of the form $\vec{W} = \vec{W}_1 \oplus (e_{1,2}) \oplus \vec{W}_2 \oplus (e_{2,3}) \oplus \dots \oplus \vec{W}_{M-1} \oplus (e_{M-1,M}) \oplus \vec{W}_M$ where:

- For all $1 \leq p \leq M$: $E(\vec{W}_p) \subseteq E(I_{i_p})$.
- For all $2 \leq p \leq M$: $e_{p-1,p} \in C_A(I_{i_{p-1}}, I_{i_p})$.

In particular, if $\vec{W} \neq ()$ i.e. $x \neq x^*$, then for all $1 \leq p \leq M$ there is an edge of \vec{W} that contains ε_{i_p} .

Proof. That last assertion is clear: for $p = 1$ we have $\varepsilon_{i_p} = \varepsilon_1 = \{x^*\} \subseteq e_1$, and for $p \geq 2$ we have $\varepsilon_{i_p} \subseteq e_{p-1,p}$ by definition of $C_A(I_{i_{p-1}}, I_{i_p})$. Let us now prove the main assertion. Write $\vec{W} = (e_1, \dots, e_L)$.

We proceed by induction on L . The case $L = 0$ is trivial: we have $x = x^*$ so we can set $M = 1$ and $\vec{W}_1 = \vec{W} = ()$. Let $L \geq 1$ and assume the result to be true for walks with less edges than

\vec{W} . The idea is to separate two simple cases: either we are currently visiting the island I_i (case $e_L \in E(I_i)$) or we have just jumped onto I_i from another island (case $e_L \notin E(I_i)$).

Let $y \in e_{L-1} \cap e_L$ if $L \geq 2$, or define $y = x^*$ if $L = 1$, so that in both cases $\vec{U} := (e_1, \dots, e_{L-1})$ is an (x^*, y) -compatible walk in A . We have $y \in V(I_j)$ for some $1 \leq j \leq N$. By the induction hypothesis, there exists a dipath $I_1 = I_{i_1} \rightarrow \dots \rightarrow I_{i_M} = I_j$ in G such that we can write $\vec{U} = \vec{U}_1 \oplus (e_{1,2}) \oplus \vec{U}_2 \oplus (e_{2,3}) \oplus \dots \oplus \vec{U}_{M-1} \oplus (e_{M-1,M}) \oplus \vec{U}_M$ where $E(\vec{U}_p) \subseteq E(I_{i_p})$ for all $1 \leq p \leq M$ and $e_{p-1,p} \in C_A(I_{i_{p-1}}, I_{i_p})$ for all $2 \leq p \leq M$.

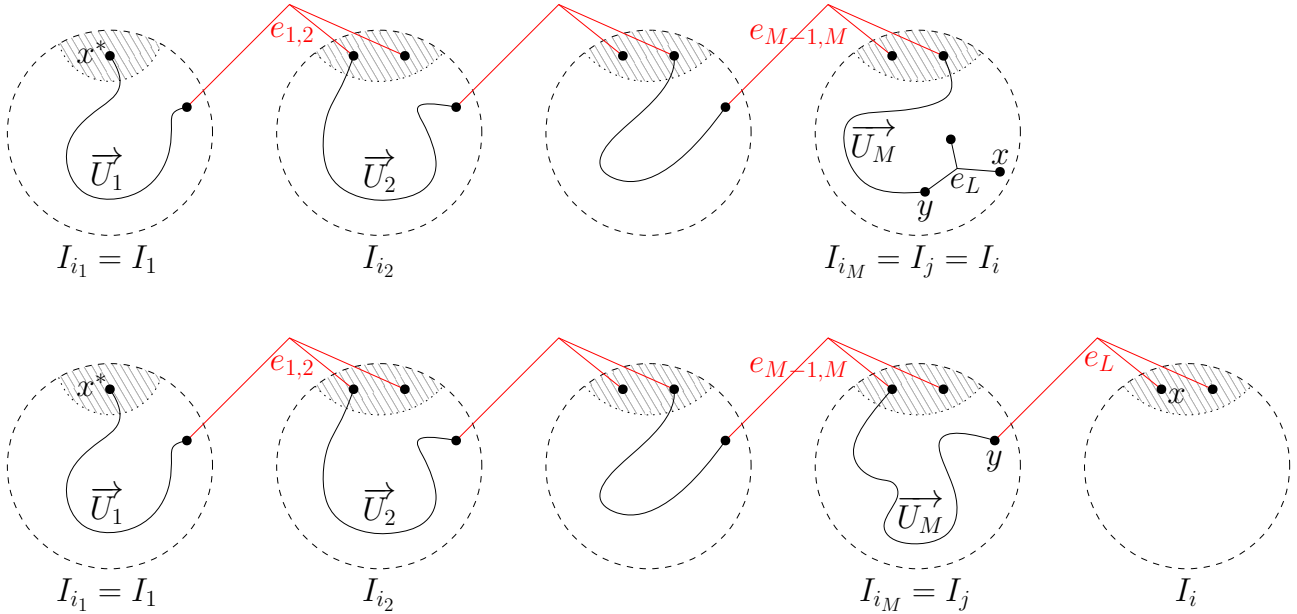


Figure II.37: Top: $e_L \in E(I_i)$. Bottom: $e_L \notin E(I_i)$.

- First suppose that $e_L \in E(I_i)$ (see Figure II.37, top). Since $y \in e_L$, this implies $i = j$, so $\vec{W}_M := \vec{U}_M \oplus (e_L)$ satisfies $E(\vec{W}_M) \subseteq E(I_i)$. Therefore, the following writing of \vec{W} completes the proof: $\vec{W} = \vec{U} \oplus (e_L) = \vec{U}_1 \oplus (e_{1,2}) \oplus \vec{U}_2 \oplus (e_{2,3}) \oplus \dots \oplus \vec{U}_{M-1} \oplus (e_{M-1,M}) \oplus \vec{W}_M$.
- Now suppose $e_L \notin E(I_i)$ (see Figure II.37, bottom), then by definition of an archipelago we have either $e_L \in C_A(I_i, I_j)$ or $e_L \in C_A(I_j, I_i)$. Suppose for a contradiction that $e_L \in C_A(I_i, I_j)$ i.e. $e_L = \{x\} \cup \varepsilon_j$: in particular $j \neq 1$ (and $|\varepsilon_j| = k - 1$), so the fact that $\varepsilon_j \subseteq e_{M-1,M}$ contradicts the $(k - 2)$ -linearity of \vec{W} since $\varepsilon_j \subseteq e_L$. Therefore $e_L \in C_A(I_j, I_i)$. In particular $i \neq 1$ (and $|\varepsilon_i| = k - 1$), so it is impossible that I_i has been visited before: if we had $i \in \{i_1, \dots, i_M\}$ then some edge of \vec{U} would contain ε_i which would contradict the $(k - 2)$ -linearity of \vec{W} once again. Setting $i_{M+1} := i$, this ensures that the islands visited by \vec{W} form a dipath $I_1 = I_{i_1} \rightarrow \dots \rightarrow I_{i_M} = I_j \rightarrow I_{i_{M+1}} = I_i$ in G , and we can write $\vec{W} = \vec{U} \oplus (e_{M,M+1}) \oplus \vec{W}_{M+1}$ where $e_{M,M+1} := e_L \in C_A(I_{i_M}, I_{i_{M+1}})$ and $\vec{W}_{M+1} := ()$, which concludes. \blacksquare

Conversely, dipaths in G yield compatible walks in A . The following proposition is a generalization to archipelagos of the property that defines an island.

Proposition II.2.13. *Let A be an archipelago, with $I_1, \dots, I_N, \varepsilon_1, \dots, \varepsilon_N, G$ suiting the definition. Let $X \subseteq V(A)$ such that $1 \leq |X| \leq k - 1$ and $X \not\subseteq \{\varepsilon_2, \dots, \varepsilon_N\}$. For all $1 \leq j \leq N$ and for every dipath $I_j = I_{i_1} \rightarrow \dots \rightarrow I_{i_M}$ in G satisfying $X \cap V(I_{i_M}) \neq \emptyset$ and $X \cap V(I_{i_p}) = \emptyset$*

for all $1 \leq p \leq M - 1$, there exists an (ε_j, X) -compatible walk \vec{W} in A of the form $\vec{W} = \vec{W}_1 \oplus (e_{1,2}) \oplus \vec{W}_2 \oplus (e_{2,3}) \oplus \dots \oplus \vec{W}_{M-1} \oplus (e_{M-1,M}) \oplus \vec{W}_M$ where:

- For all $1 \leq p \leq M$: $E(\vec{W}_p) \subseteq E(I_{i_p})$.
- For all $2 \leq p \leq M$: $e_{p-1,p} \in C_A(I_{i_{p-1}}, I_{i_p})$.

Proof. We proceed by induction on M .

- First suppose $M = 1$: we need to show that if $X \cap V(I_j) \neq \emptyset$ then there exists an (ε_j, X) -compatible walk in I_j . This is basically the definition of an island, except that X is not necessarily entirely included in $V(I_j)$. This is not a problem: since $X \notin \{\varepsilon_2, \dots, \varepsilon_N\}$ by assumption, there exists an $(\varepsilon_j, X \cap V(I_j))$ -compatible walk \vec{W} in I_j by definition of an island, and \vec{W} is also (ε_j, X) -compatible by Compatibility Lemma II.2.5.
- Now suppose $M \geq 2$ and assume the result to be true for all shorter dipaths in G . We build the desired (ε_{i_1}, X) -compatible walk by assembling three parts:

- (1) By the induction hypothesis, there exists an (ε_{i_2}, X) -compatible walk \vec{W}' in A of the form $\vec{W}' = \vec{W}_2 \oplus (e_{2,3}) \oplus \vec{W}_3 \oplus (e_{3,4}) \oplus \dots \oplus \vec{W}_{M-1} \oplus (e_{M-1,M}) \oplus \vec{W}_M$ where $E(\vec{W}_p) \subseteq E(I_{i_p})$ for all $2 \leq p \leq M$ and $e_{p-1,p} \in C_A(I_{i_{p-1}}, I_{i_p})$ for all $3 \leq p \leq M$.
- (2) Let $e_{1,2} \in C_A(I_{i_1}, I_{i_2})$, which exists since $(I_{i_1}, I_{i_2}) \in E(G)$: we have $e_{1,2} = \{x\} \cup \varepsilon_{i_2}$ for some $x \in V(I_{i_1})$.
- (3) Finally, by definition of an island, there exists an (ε_{i_1}, x) -compatible walk \vec{W}_1 in I_{i_1} .

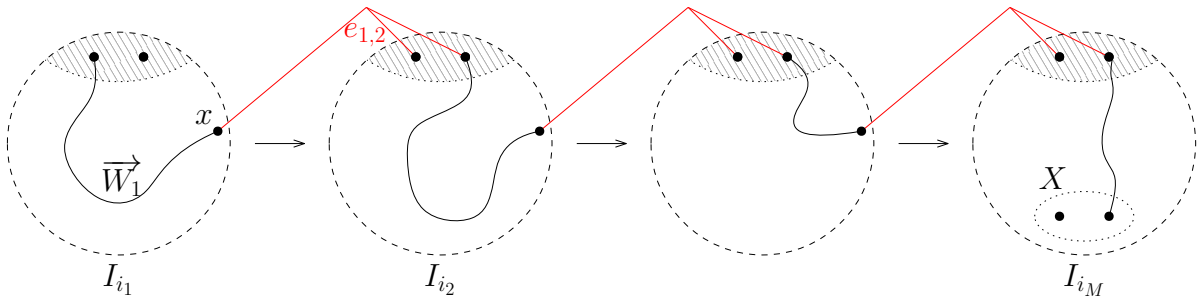


Figure II.38: An (ε_{i_1}, X) -compatible walk in an archipelago.

Let $\vec{W} := \vec{W}_1 \oplus (e_{1,2}) \oplus \vec{W}'$ as in Figure II.38. Compatibility Lemma II.2.7 applied to $H' = A$, $\alpha = \varepsilon_{i_1}$, $\beta = \{x\}$, $\gamma = \varepsilon_{i_2}$ and $\delta = X$ ensures that \vec{W} is an (ε_{i_1}, X) -compatible walk in A . ■

We get the following characterization for the entries of an archipelago:

Proposition II.2.14. *Let A be an archipelago, with $I_1, \dots, I_N, \varepsilon_1, \dots, \varepsilon_N$ suiting the definition. Let $X \subseteq V(A)$ such that $1 \leq |X| \leq k - 1$. There exists an (x^*, X) -compatible walk in A if and only if $X \notin \{\varepsilon_2, \dots, \varepsilon_N\}$.*

Proof. We distinguish both cases:

- Suppose $X = \varepsilon_i$ for some $2 \leq i \leq N$. Suppose for a contradiction that there exists an (x^*, ε_i) -compatible walk $\vec{W} = (e_1, \dots, e_L)$ in A . Let $x \in e_L \cap \varepsilon_i$: in particular, \vec{W} is an (x^*, x) -compatible walk in A , so some edge of \vec{W} contains ε_i by Proposition II.2.12. Therefore, the walk $(x^*) \oplus \vec{W} \oplus (\varepsilon_i)$ is not $(k - 2)$ -linear, contradicting the fact that \vec{W} is (x^*, ε_i) -compatible.

- Suppose $X \notin \{\varepsilon_2, \dots, \varepsilon_N\}$. Out of all the dipaths in G from I_1 to one of the islands intersecting X (recall that G contains a spanning arborescence rooted at I_1 , so there exists at least one), consider a shortest one, so that X only intersects the last island of that path. We can now apply Proposition II.2.13: there exists an (ε_1, X) -compatible walk in A , which concludes since $\varepsilon_1 = \{x^*\}$. ■

Corollary II.2.15. *Let A be an archipelago in H . For all $x \in V(A)$, there exists an (x^*, x) -compatible walk in A . In particular, $V(A) \subseteq CC(x^*)$.*

Proof. It suffices to apply Proposition II.2.14 with $X = \{x\}$. ■

Finally, we show that an archipelago has a unique decomposition.

Proposition II.2.16. *Any archipelago A has unique islands and entries suiting the definition.*

Proof. Let $\varepsilon_1, \dots, \varepsilon_N$ be entries suiting the definition: we have $\varepsilon_1 = \{x^*\}$, moreover $\{\varepsilon_2, \dots, \varepsilon_N\}$ is exactly the set of all subsets $X \subseteq V(A)$ such that $1 \leq |X| \leq k - 1$ and there exists no (x^*, X) -compatible walk in A by Proposition II.2.14, so these entries are unique. Suppose for a contradiction that $\{I_1, \dots, I_N\}$ and $\{I'_1, \dots, I'_N\}$ are two distinct sets of islands suiting the definition, where I_i and I'_i have the same entry ε_i for all $1 \leq i \leq N$. Since islands are induced subhypergraphs of A , $\{V(I_1), \dots, V(I_N)\}$ and $\{V(I'_1), \dots, V(I'_N)\}$ are two distinct partitions of $V(A)$, so there exists $1 \leq i \leq N$ such that $V(I_i) \not\subseteq V(I'_i)$. Let $x \in V(I_i) \setminus V(I'_i)$, and let $1 \leq j \leq N$ be the unique index such that $x \in V(I'_j)$, so that $x \in V(I_i) \cap V(I'_j)$ where $i \neq j$.

- Using the first decomposition, there exists an (ε_i, x) -compatible walk $\vec{W} = (e_1, \dots, e_L)$ in I_i by definition of an island. For all $2 \leq l \leq N$, no edge of \vec{W} contains ε_l : if $l = i$ then this comes the definition of an (ε_i, x) -compatible walk, and if $l \neq i$ then this is obvious since $V(I_i)$ is disjoint from ε_l .
- Using the second decomposition, since $x \in V(I'_j)$ and ε_i is disjoint from $V(I'_j)$, we can define $r := \inf\{1 \leq p \leq L, e_p \not\subseteq V(I'_j)\}$. We have $e_r \not\subseteq V(I'_j)$, however e_r intersects $V(I'_j)$ by minimality of r , therefore e_r is necessarily a crossing edge for the second decomposition. This means that $\varepsilon_l \subseteq e_r$ for some $2 \leq l \leq N$, which contradicts what we have just established. ■

Notation II.2.17. Let A be an archipelago. Proposition II.2.16 allows us to define without ambiguity:

- $\mathcal{I}(A)$: the set of islands of A .
- $\varepsilon(A)$: the set of entries of the islands of A .
- $G(A)$: the digraph from the definition of an archipelago.

By definition of an archipelago, the digraph $G(A)$ contains a spanning arborescence (rooted at the island containing x^*). We introduce the following terminology for the case where $G(A)$ is exactly an arborescence:

Definition II.2.18. An archipelago A is said to be *arborescent* if $G(A)$ is an arborescence.

II.2.3 Characterization as unique maximal archipelago

We are going to establish the main result of this section: $H[CC(x^*)]$ is an archipelago, moreover it is the unique maximal archipelago in H .

II.2.3.1 A -types and statement of the main result

Definition II.2.19. An x^* -archipelago A in H is said to be maximal if there is no x^* -archipelago in H that has A as a strict subhypergraph.

An archipelago A is maximal if and only if there exists no edge $e \in V(H) \setminus V(A)$ such that $A \cup e$ is an archipelago. The nature of $A \cup e$ depends on how A and e interact with each other. We distinguish five situations:

Definition II.2.20. Let A be an archipelago. An edge $e \in E(H) \setminus E(A)$ is of one of five A -types:

1. "exterior": $|e \cap V(A)| = 0$.
2. "new crossing": $|e \cap V(A)| = 1$.
3. "crossing": e is a crossing edge between two islands of A .
4. "cut": e is of the form $e = \varepsilon \cup \{x\}$, where ε is an entry of A of size $k-1$ and $x \in V(H) \setminus V(A)$.
5. "other": e is none of the above.

Those are well defined because the islands and entries of an archipelago are unique by Proposition II.2.16. The five A -types are illustrated in Figure II.39.

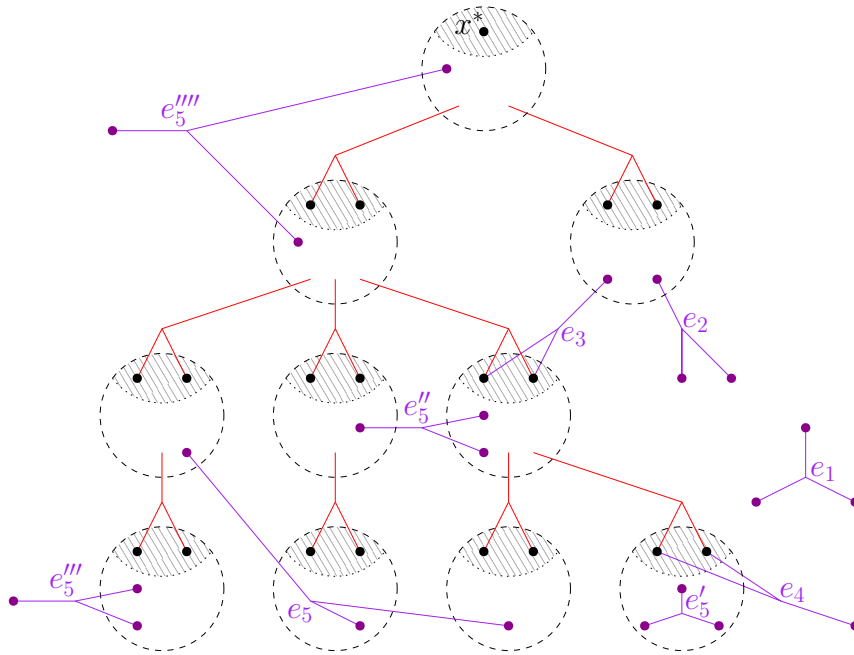


Figure II.39: An arborescent archipelago A (the inside of the islands is not detailed), and some edges in $E(H) \setminus E(A)$ (in purple). The names of the edges follow the numbering from Definition II.2.20: e_1 is of A -type "exterior", e_2 is of A -type "new crossing", etc.

We can now state the main result of this section:

Theorem II.2.21. Let A be an x^* -archipelago. The following three assertions are equivalent:

- (i) $A = H[CC(x^*)]$.
- (ii) A is maximal.
- (iii) All edges in $V(H) \setminus V(A)$ are of A -type "cut" or "exterior".

The difficult implication is (ii) \implies (iii). We are going to establish the contrapositive: any edge that is of A -type "new crossing", "crossing" or "other" can be added to A while maintaining an archipelago structure. This point of view suggests an incremental algorithm to build $H[CC(x^*)]$,

consisting in adding edges of these three A -types until reaching maximality. We will come back to these algorithmic considerations in Section III.1.

II.2.3.2 Augmenting archipelagos

We now show why $A \cup e$ is still an archipelago when e is of A -type "crossing", "new crossing", or "other" (in that order: easiest to hardest). We must exhibit a suitable decomposition of $A \cup e$ into islands for each case. Let us fix an archipelago A with islands I_1, \dots, I_N and entries $\varepsilon_1, \dots, \varepsilon_N$ (with $\varepsilon_1 = \{x^*\}$ as usual).

First of all, the A -type "crossing" is straightforward: e is added as a crossing edge, and the decomposition remains the same as that of A . See $e = e_3$ from Figure II.39 for example. Note that $A \cup e$ might not be arborescent even if A is.

Proposition II.2.22. *Let $e \in V(H) \setminus V(A)$ be of A -type "crossing" i.e. $e \in C(I_i, I_j)$ for some $1 \leq i, j \leq N$. Then $A \cup e$ is an archipelago with:*

- $\mathcal{I}(A \cup e) = \mathcal{I}(A)$.
- $\varepsilon(A \cup e) = \varepsilon(A)$.
- $G(A \cup e)$ the digraph obtained from $G(A)$ by adding an arc (I_i, I_j) if there was not one already.

Proof. This is straightforward. Obviously, $G(A \cup e)$ contains a spanning arborescence rooted at I_1 since $G(A)$ does. ■

The A -type "new crossing" is not much more challenging: a new island is created, with e being the crossing edge that connects it to the rest (see Figure II.40). This time, if A is arborescent then $A \cup e$ also is.

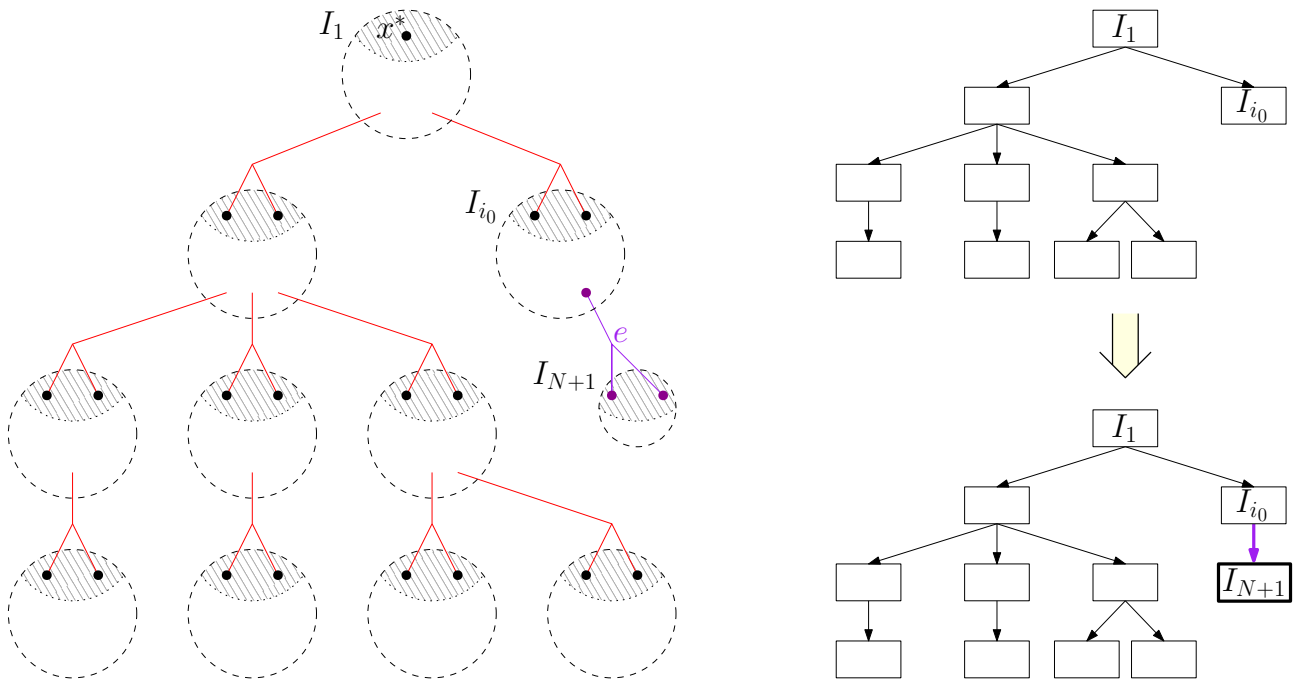


Figure II.40: The archipelago $A \cup e$, where A and $e = e_2$ are as in Figure II.39. On the right: the digraphs $G(A)$ (top) and $G(A \cup e)$ (bottom).

Proposition II.2.23. *Let $e \in V(H) \setminus V(A)$ be of A -type "new crossing". Let $1 \leq i_0 \leq N$ be the index of the only island that intersects e , and let I_{N+1} be the deserted island with entry*

$\varepsilon_{N+1} := e \setminus V(I_{i_0})$. Then $A \cup e$ is an archipelago with:

- $\mathcal{I}(A \cup e) = \mathcal{I}(A) \cup \{I_{N+1}\}$.
- $\varepsilon(A \cup e) = \varepsilon(A) \cup \{\varepsilon_{N+1}\}$.
- $G(A \cup e)$ the digraph obtained from $G(A)$ by adding a new node I_{N+1} and an arc (I_{i_0}, I_{N+1}) .

Proof. Clearly, e is a crossing edge from I_{i_0} to I_{N+1} , hence the new arc in $G(A \cup e)$. Moreover, taking a spanning arborescence of $G(A)$ rooted at I_1 and adding this new arc yields a spanning arborescence of $G(A \cup e)$ rooted at I_1 . ■

Finally, let us consider the case of an edge e of A -type "other". Recall that, by definition, this means that: $|e \cap V(A)| \geq 2$, e is not a crossing edge, and e is not of the form $\varepsilon \cup \{x\}$ where ε is an entry of A of size $k - 1$ and $x \in V(H) \setminus V(A)$.

This case is more complicated. Consider Figure II.39. If e only intersects one island ($e = e'_5$ or $e = e'''_5$ for instance), then it should be easy to show that this island plus e is still an island. If e links several islands however, then the way to redefine islands is not as straightforward, since e is not a crossing edge. Suppose $e = e_5$ for instance, as in Figure II.41. The fact that e acts as a bridge between several islands creates new compatible walks: for example, we have an (x^*, ε_6) -compatible walk in $A \cup e$ (represented schematically in Figure II.41), therefore ε_6 would not be an entry of $A \cup e$ (recall Proposition II.2.14). Actually, it can be shown that the subhypergraph I , formed by the union of $I_2, I_4, I_5, I_6, I_8, I_9$ and the crossing edges between them as well as e , is an island with entry ε_2 . Therefore, $A \cup e$ is an archipelago with five islands: I_1, I_3, I_7, I_{10}, I . On this example, we see how adding an edge can merge islands together. We are now going to generalize this argument for arborescent archipelagos, then address the non-arborescent case.

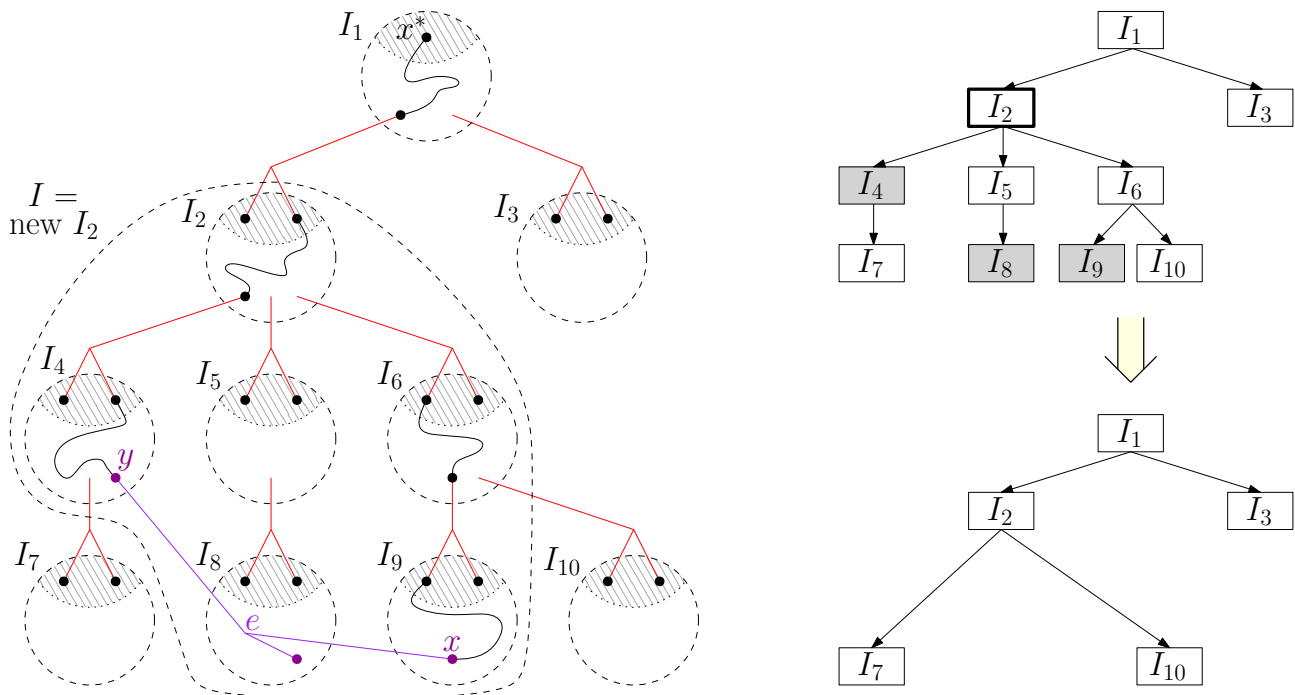


Figure II.41: The archipelago $A \cup e$, where A and $e = e_5$ are as in Figure II.39. On the right: the digraphs $G(A)$ (top) and $G(A \cup e)$ (bottom).

Definition II.2.24. Let G be an arborescence rooted at some $v^* \in V(G)$, and let $U = \{v_1, \dots, v_r\} \subseteq V(G)$. For all $1 \leq i \leq r$, let $v^* = v_{i,1} \rightarrow \dots \rightarrow v_{i,l_i} = v_i$ be the unique dipath from v^* to v_i in G . Define $i_0 := \sup\{1 \leq p \leq \min_{1 \leq i \leq r} l_i \mid v_{1,p} = \dots = v_{r,p}\}$. The *lowest common ancestor of U in G* is defined as $\text{LCA}_G(U) := v_{i_0}$.

Definition II.2.25. Let G be an arborescence and let $v \in V(G)$. For all $1 \leq i \leq r$, let $v = v_{i,1} \rightarrow \dots \rightarrow v_{i,l_i} = v_i$ be a dipath from v to some $v_i \in V(G)$ in G . Let $U := \bigcup_{1 \leq i \leq r} \{v_{i,1}, \dots, v_{i,l_i}\}$ be the set of all nodes on these dipaths. *Merging U into v* means:

- deleting all nodes in $U \setminus \{v\}$;
- deleting all arcs between nodes in U ;
- replacing every arc $(u, w) \in (U \setminus \{v\}) \times (V(G) \setminus U)$ with an arc (v, w) .

Example. Figure II.41 features a merging process on the right. The three considered dipaths are: $I_2 \leftarrow I_4$, $I_2 \leftarrow I_5 \leftarrow I_8$, $I_2 \leftarrow I_6 \leftarrow I_9$. The set $U = \{I_2, I_4, I_5, I_6, I_8, I_9\}$ has been merged into $v = I_2$.

Proposition II.2.26. *Assume A is arborescent. Let $e \in V(H) \setminus V(A)$ be of A -type "other". Define:*

- $J_0 := \{1 \leq i \leq N, V(I_i) \cap e \neq \emptyset\}$, the set of indices of the islands that e intersects.
- i_0 the index such that $I_{i_0} := \text{LCA}_{G(A)}(\{I_i, i \in J_0\})$.
- $J := \bigcup_{i \in J_0} \{1 \leq j \leq N \mid I_j \text{ is on the dipath from } I_{i_0} \text{ to } I_i \text{ in } G(A)\} \supseteq J_0$.
- $I := A[\bigcup_{j \in J} V(I_j)]$, the island that will replace I_{i_0} (with the same entry ε_{i_0}).

Then $A \cup e$ is an arborescent archipelago with:

- $\mathcal{I}(A \cup e) = (\mathcal{I}(A) \setminus \{I_j, j \in J\}) \cup \{I\}$.
- $\varepsilon(A \cup e) = \varepsilon(A) \setminus \{\varepsilon_j, j \in J \setminus \{i_0\}\}$.
- $G(A \cup e)$ the digraph obtained from $G(A)$ by merging $\{I_j, j \in J\}$ into I_{i_0} .

Proof. For visual help, refer to Figure II.41: in this example we have $J_0 = \{4, 8, 9\}$, $i_0 = 2$, $J = \{2, 4, 5, 6, 8, 9\}$. The merging process that defines $G(A \cup e)$ clearly preserves the fact that the digraph is an arborescence. To complete the proof, it remains to show that I is an island with entry ε_{i_0} . Therefore, let $X \subseteq V(I)$ such that $1 \leq |X| \leq k - 1$ (and $X \neq \varepsilon_{i_0}$ if $i_0 \neq 1$): we need to find an (ε_{i_0}, X) -compatible walk in I . As visible in Figure II.39, e might or might not be included in $V(A)$, so in general we have $V(I) = \bigcup_{j \in J} V(I_j) \cup e$. There are four cases, which we address separately.

1) Case 1: $X \subseteq \bigcup_{j \in J} V(I_j)$ and $X \not\subseteq \{\varepsilon_j, j \in J \setminus \{i_0\}\}$.

Of all dipaths in $G(A)$ from I_{i_0} to an island intersecting X , let $I_{i_0} = I_{j_1} \rightarrow \dots \rightarrow I_{j_M}$ be a shortest one, so that $X \cap V(I_{j_p}) \neq \emptyset$ and $X \cap V(I_{j_p}) = \emptyset$ for all $1 \leq p \leq M - 1$. Note that, by definition of J , we have $\{j_1, \dots, j_M\} \subseteq J$, so the islands I_{j_1}, \dots, I_{j_M} are all subhypergraphs of I and all crossing edges between them in A are edges of I . By Proposition II.2.13, there exists an (ε_{i_0}, X) -compatible walk \vec{W} in A such that $E(\vec{W}) \subseteq \bigcup_{p=1}^M E(I_{j_p}) \cup \bigcup_{p=2}^M C_A(I_{j_{p-1}}, I_{j_p}) \subseteq E(I)$, which concludes.

2) Case 2: X intersects both $\bigcup_{j \in J} V(I_j)$ and $e \setminus \bigcup_{j \in J} V(I_j)$.

Define $X' := X \cap \bigcup_{j \in J} V(I_j)$, we have $1 \leq |X'| \leq k - 1$. Case 1 applied to X' gives us an (ε_{i_0}, X') -compatible walk \vec{W} in I , which is also (ε_{i_0}, X) -compatible by Compatibility Lemma II.2.5 applied to $H' = I$, $\alpha = \varepsilon_{i_0}$, $\beta = X'$ and $\beta' = X$.

3) Case 3: $X \subseteq e \setminus \bigcup_{j \in J} V(I_j)$.

Define $X' := e \cap \bigcup_{j \in J} V(I_j)$, we have $2 \leq |X'| \leq k - 1$ hence $1 \leq |X| \leq k - 2$: indeed $|X'| \geq 2$ by definition of the A -type "other", and $|X'| \leq k - 1$ because $e \setminus \bigcup_{j \in J} V(I_j) \supseteq X \neq \emptyset$. Moreover $X' \not\subseteq \{\varepsilon_j, j \in J \setminus \{i_0\}\}$, otherwise e would be of A -type "cut". We can thus apply Case 1 to X' , which gives us an (ε_{i_0}, X') -compatible walk \vec{W} in I . Compatibility Lemma II.2.6 applied to $H' = I$, $\alpha = \varepsilon_{i_0}$, $\beta = X'$ and $\gamma = X$ ensures that $\vec{W} \oplus (e)$ is an (ε_{i_0}, X) -compatible walk in I .

4) Case 4: $X = \varepsilon_j$ for some $j \in J \setminus \{i_0\}$.

In particular $|J| \geq 2$, so e intersects several islands. Note that, since I_{i_0} is a strict ancestor of I_j in $G(A)$, we have $j \neq 1$. Remember our example from Figure II.41: we considered $X = \varepsilon_6$, and the (ε_2, X) -compatible walk was obtained by going from ε_2 to $e \cap V(I_4) = \{y\}$, then using e to jump from I_4 to I_9 , then going from $e \cap V(I_9) = \{x\}$ to X . Let us now build this walk in general.

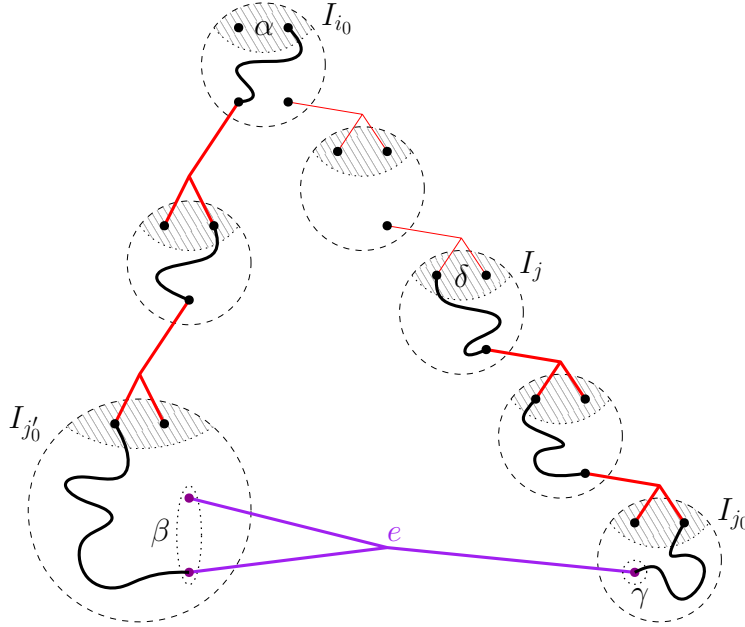


Figure II.42: Illustration of Case 4 from Proposition II.2.26. The bold walks (in red and black) are \vec{W} on the right and \vec{W}' on the left.

- Let $j_0 \in J_0$ such that the dipath $I_j = I_{i_1} \rightarrow \dots \rightarrow I_{i_M} = I_{j_0}$ in $G(A)$ is shortest, so that $i_p \notin J_0$ for all $1 \leq p \leq M-1$. This means $e \cap V(I_{i_M}) \neq \emptyset$ and $e \cap V(I_{i_p}) = \emptyset$ for all $1 \leq p \leq M-1$. Since e intersects several islands, we know $1 \leq |e \cap V(I_{j_0})| \leq k-1$. Moreover the fact that $j \neq 1$ implies that $j_0 \neq 1$, so $e \cap V(I_{j_0}) \neq \varepsilon_{j_0}$, otherwise e would be of A -type "crossing". We can thus apply Proposition II.2.13 and get an $(\varepsilon_j, e \cap V(I_{j_0}))$ -compatible walk \vec{W} in A such that $E(\vec{W}) \subseteq \bigcup_{p=1}^M E(I_{i_p}) \cup \bigcup_{p=2}^M C_A(I_{i_{p-1}}, I_{i_p})$, hence $E(\vec{W}) \subseteq E(I)$ since $\{i_1, \dots, i_M\} \subseteq J$ by definition of J . See Figure II.42 (walk on the right).
- Since the lowest common ancestor of $\{I_i, i \in J_0\}$ is I_{i_0} and not I_j , there exists $j'_0 \in J_0 \setminus \{j_0\}$ such that I_j is not an ancestor of $I_{j'_0}$, so the dipath $I_{i_0} = I_{i'_1} \rightarrow \dots \rightarrow I_{i'_{M'}} = I_{j'_0}$ from I_{i_0} to $I_{j'_0}$ in $G(A)$ satisfies $\{i_1, \dots, i_M\} \cap \{i'_1, \dots, i'_{M'}\} = \emptyset$ (see Figure II.42 for the relative positions of the four islands in play: $I_{i_0}, I_j, I_{j_0}, I_{j'_0}$). As usual, we choose j'_0 so that this dipath is shortest, this way we have $e \cap V(I_{i'_{M'}}) \neq \emptyset$ and $e \cap V(I_{i'_p}) = \emptyset$ for all $1 \leq p \leq M'-1$. Since e intersects several islands, we know $1 \leq |e \cap V(I_{j'_0})| \leq k-1$. Moreover, if $j'_0 \neq 1$ then $e \cap V(I_{j'_0}) \neq \varepsilon_{j'_0}$ otherwise e would be of A -type "crossing". We can thus apply Proposition II.2.13 and get an $(\varepsilon_{j'_0}, e \cap V(I_{j'_0}))$ -compatible walk \vec{W}' in A such that $E(\vec{W}') \subseteq \bigcup_{p=1}^{M'} E(I_{i'_p}) \cup \bigcup_{p=2}^{M'} C_A(I_{i'_{p-1}}, I_{i'_p})$, hence $E(\vec{W}') \subseteq E(I)$ since $\{i'_1, \dots, i'_{M'}\} \subseteq J$ by definition of J . See Figure II.42 (walk on the left).
- Since \vec{W} is an $(\varepsilon_j, e \cap V(I_{j_0}))$ -compatible walk, \overleftarrow{W} is an $(e \cap V(I_{j_0}), \varepsilon_j)$ -compatible walk. Compatibility Lemma II.2.7 applied to $H' = I$, $\alpha = X = \varepsilon_{i_0}$, $\beta = e \cap V(I_{j'_0})$, $\gamma = e \cap$

$V(I_{j_0})$ and $\delta = \varepsilon_j$, whose conditions are fulfilled since $\{i_1, \dots, i_M\} \cap \{i'_1, \dots, i'_{M'}\} = \emptyset$, ensures that $\vec{W}' \oplus (e) \oplus \overleftarrow{W}$ is an $(\varepsilon_{i_0}, \varepsilon_j)$ -compatible walk in I which concludes. ■

The non-arborescent case also features a merging of islands, but its description is not as convenient as what we have just seen in the arborescent case (with the lowest common ancestor). We thus state the following corollary without specifying the exact decomposition.

Corollary II.2.27. *Let $e \in V(H) \setminus V(A)$ be of A -type "other". Then $A \cup e$ is an archipelago.*

Proof. If A is arborescent, then this is simply Proposition II.2.26. Otherwise, let A_0 be an arborescent archipelago obtained from A by removing some crossing edges e_1, \dots, e_r . Since A_0 has the same islands and entries as A , the notions of A -type and A_0 -type coincide, so e_1, \dots, e_r are of A_0 -type "crossing" and e is of A_0 -type "other". In particular, $A_1 := A_0 \cup e$ is an arborescent archipelago by Proposition II.2.26. Now, the edges e_1, \dots, e_r may not all be of A_1 -type "crossing", depending on which islands have been merged together when going from A_0 to A_1 . However, we have $e_1, \dots, e_r \subseteq V(A) = V(A_1)$, so there are only two possible A_1 -types for each e_i : "crossing" or "other".

- If e_1, \dots, e_r are all of A_1 -type "crossing", then $A_1 \cup e_1 \cup \dots \cup e_r = A \cup e$ is an archipelago by successive applications of Proposition II.2.22, so the proof is over.
- If some edge among e_1, \dots, e_r , say e_1 , is of A_1 -type "other", then $A_2 := A_1 \cup e_1$ is an arborescent archipelago by Proposition II.2.26. We then repeat this process: if e_2, \dots, e_r are all of A_2 -type "crossing" then the proof is over, otherwise we add one of A_2 -type "other", etc. until none of the e_i remains. In the end, all e_i will have been put back in the archipelago. ■

The proof of Corollary II.2.27 is instructive: to show that it is possible to add an edge e of A -type "other", we first go backwards by removing some crossing edges to make the archipelago arborescent, then add e , then put the removed edges back in. With the idea in mind to design an efficient incremental algorithm to build a maximal archipelago, this suggests that the edges of type "crossing" should be added last, so that the archipelago remains arborescent for as long as possible. This is exactly what we are going to do in Section III.1, right after we are done proving Theorem II.2.21.

II.2.3.3 Proof of the main result

We can finally prove our main structural result on $(k - 2)$ -linear connected components.

Proof of Theorem II.2.21. We establish (i) \implies (ii) \implies (iii) \implies (i).

- (i) \implies (ii). This is the easiest implication of the three. Suppose $A = H[CC(x^*)]$, and let A' be an archipelago. By Corollary II.2.15, we have $V(A') \subseteq CC(x^*)$, therefore $A' \subseteq H[CC(x^*)] = A$.
- (iii) \implies (i). Suppose that all edges in $V(H) \setminus V(A)$ are of A -type "cut" or "exterior". Since edges of these two A -types cannot have all their vertices in A , we know A is a induced subhypergraph of H . Moreover $V(A) \subseteq CC(x^*)$ by Corollary II.2.15, so it remains to verify that $CC(x^*) \subseteq V(A)$. Suppose for a contradiction that there exists $x \in CC(x^*) \setminus V(A)$. Let $\vec{W} = (e_1, \dots, e_L)$ be an (x^*, x) -compatible walk in H . Since $x \notin V(A)$, we can define $M := \inf\{1 \leq p \leq L, e_p \not\subseteq V(A)\}$. An edge incident to x^* cannot be of A -type "cut" or "exterior", therefore $e_1 \in E(A)$, from which $e_1 \subseteq V(A)$ hence $M \geq 2$. Moreover e_M intersects $e_{M-1} \subseteq V(A)$, so e_M is of A -type "cut": this yields $e_M \cap V(A) = \varepsilon$ for some entry ε of A of size $k - 1$. Let $y \in e_M \cap e_{M-1} \subseteq \varepsilon$: since (e_1, \dots, e_{M-1}) is an (x^*, y) -compatible

walk in A , Proposition II.2.12 ensures that $\varepsilon \subseteq e_{M-1}$. Since $\varepsilon \subseteq e_M$, this contradicts the $(k-2)$ -linearity of \vec{W} .

- (ii) \implies (iii). This is the most difficult implication, but the work has already been done: Propositions II.2.22 and II.2.23 along with Corollary II.2.27 show that archipelagos such that there exists an edge of A -type "crossing", "new crossing", or "other" are not maximal. ■

II.2.4 Conclusion and prospects

We have obtained a fine description of how the $(k-2)$ -linear connected component of a vertex x^* is structured, both in terms of internal organization and interaction with the outside. The structure in question is that of maximal x^* -archipelago. Since this structure actually characterizes the $(k-2)$ -linear connected component of x^* , we will be able to compute this component efficiently, by building a maximal x^* -archipelago.

It is natural to ask the question about λ -linear connected components for values of λ lesser than $k-2$. The linear case $\lambda=1$ is of particular interest, since linear paths appear in numerous other problems. However, it is unclear how to generalize the notion of archipelago. As an illustration of the difficulties that can be encountered, consider $k=4$ and $\lambda=1$. Suppose that we have adapted our structures to this case, settling on some definition of an archipelago as a union of pairwise disjoint islands with crossing edges between them. Say we have a 4-uniform hypergraph H , an x^* -archipelago A in H and an edge $e = \{a, b, c, d\} \in E(H)$ where: $a, b \in V(A)$ are in different islands, and $c, d \notin V(A)$. See Figure II.43. The edge e ensures that $c, d \in CC_H^1(x^*)$, so $A \cup e$ should be an x^* -archipelago, but what would its decomposition be? On the one hand, e forms a bridge between the islands containing a and b , in such a way that e should be part of a new merged island. On the other hand, c and d form an "inseparable" pair (indeed, all linear paths from x^* to c or d in $A \cup e$ end with the edge e , which contains both c and d), so e should be a crossing edge towards a new island with entry $\{c, d\}$. There seems to be a contradiction.

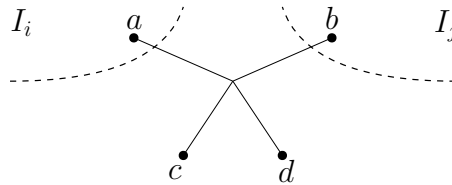


Figure II.43: Illustration of the problematic situation with edges of size 4.

Nevertheless, it might still be possible to define an adequate structure for some values of λ lesser than $k-2$. Such a structure would likely need to contain more information on the existence of certain paths, and would thus be more complex. For instance, there could be overlapping islands carrying different bits of information about the same vertices.

Chapter III

How structure contributes to algorithms and game complexness

This chapter presents results about connectivity problems and the Maker-Breaker game that can be derived from the structural studies performed in the previous chapter, and illustrates how these results compare with other results around the same problems. Most of these consequences come in the form of polynomial-time algorithms. As regards the Maker-Breaker game on hypergraphs of rank 3, the implications go beyond the algorithmic complexity: we also get exact bounds on some hypergraph parameters which help measure how intricate optimal strategies are, so that we get a strong grasp on the complexness of the game as a whole.

III.1 Algorithms for connectivity problems in graphs and hypergraphs[†]

Let us start where we left off in the previous chapter, by considering algorithmic aspects of the λ -linear connectivity problem in hypergraphs.

III.1.1 Presentation of the problem and state of the art

We are interested in the algorithmic complexity of the following decision problem:

HYPCON _{k,λ}
Input : a k -uniform hypergraph H and two distinct vertices x, y of H .
Output : YES if and only if there exists a λ -linear xy -path in H .

In the literature, most algorithmic problems about paths in hypergraphs are based on optimization rather than existence, generalizing known graph problems [GLP93][GZR⁺15][AL17][BCM⁺18]. The few existence problems that have been studied seem to either concern directed hypergraphs [TT09] or be very specific to some applied purpose [GPR⁺14]. To our knowledge, no existence problem on a type of path defined by intersection constraints has been studied before. However, we will see that our problem has links with the following connectivity problem in graphs, which was first introduced in [KSG73]:

[†]These results are part of the article [GGS22a], which has been submitted to *Discrete Mathematics & Theoretical Computer Science*.

PAFP ("Paths Avoiding Forbidden Pairs")
Input : a bicolored graph G (all edges are blue or red), and $x, y \in V(G)$.
Output : YES if and only if there exists a blue induced path between x and y in G .

It has been proved that PAFP (sometimes referred to as PFPP) is NP-complete in general [GMO76]. In fact, unless $P=NP$, there is no linear approximation ratio for the minimum number of red edges induced by a blue path between two given vertices [HKK⁺12]. For the problem on digraphs (the blue edges are directed arcs), which is by far the most studied version in the literature, a few tractable cases are known [Yin97][CKT⁺01][KP09], of which only the first one can be adapted to undirected graphs.

In this section, we provide a polynomial-time algorithm for $\text{HYPCON}_{k,k-2}$ and we explain how this has the potential to unearth new tractable cases for PAFP.

III.1.2 The λ -linear connectivity problem $\text{HYPCON}_{k,\lambda}$

A stronger algorithm for $\text{HYPCON}_{k,\lambda}$ consists in computing the entire λ -linear connected component: indeed, determining $CC_H^\lambda(x^*)$ for some fixed x^* solves the λ -linear x^*y -path existence question for all possibilities of y at once. This is what we are going to do for $\lambda = k - 2$.

III.1.2.1 A polynomial-time algorithm for the case $\lambda = k - 2$

For $\lambda = k - 2$, a polynomial-time incremental algorithm to compute $CC_H^{k-2}(x^*)$ can readily be derived from Theorem II.2.21 and its proof. The algorithm `COMPUTECC` (Algorithm 1) takes a k -uniform hypergraph H and a vertex x^* as an input, and returns $CC_H^{k-2}(x^*)$. Actually, it does more: it even computes the archipelago $H[CC_H^{k-2}(x^*)]$ with its full decomposition. Islands are simply implemented as vertex sets, because their edge sets are never used.

Algorithm 1 `COMPUTECC`(H, x^*)

```

1: initialize  $V(I_1) \leftarrow \{x^*\}$ 
2: define  $\varepsilon_1 \leftarrow \{x^*\}$ 
3: initialize the archipelago  $A$  with:
4:    $E(A) \leftarrow \emptyset$ 
5:    $\mathcal{I}(A) \leftarrow \{V(I_1)\}$ 
6:    $\varepsilon(A) \leftarrow \{\varepsilon_1\}$ 
7:    $G(A) \leftarrow$  a digraph with only one node, labelled  $I_1$ 
8: initialize  $N \leftarrow 1$  (index of the last created island)
9: while there exists  $e \in E(H) \setminus E(A)$  of  $A$ -type "new crossing" or "other" do
10:   if  $e$  is of  $A$ -type "new crossing" then
11:     update  $A$  as  $A \cup e$  by performing ADD_NEWCROSSING
12:   else
13:     update  $A$  as  $A \cup e$  by performing ADD_OTHER
14:   end if
15: end while
16: while there exists  $e \in E(H) \setminus E(A)$  of  $A$ -type "crossing" do
17:   update  $A$  as  $A \cup e$  by performing ADD_CROSSING
18: end while
19: return  $\bigcup_{V(I_i) \in \mathcal{I}(A)} V(I_i)$ 

```

Algorithm 2 ADD_NEWCROSSING

-
- 1: **define** $1 \leq i_0 \leq N$ as the only index such that $e \cap V(I_{i_0}) \neq \emptyset$
 - 2: **initialize** $V(I_{N+1}) \leftarrow e \setminus V(I_{i_0})$
 - 3: **define** $\varepsilon_{N+1} \leftarrow e \setminus V(I_{i_0})$
 - 4: **update** the archipelago A as follows:
 - 5: $E(A) \leftarrow E(A) \cup \{e\}$
 - 6: $\mathcal{I}(A) \leftarrow \mathcal{I}(A) \cup \{V(I_{N+1})\}$
 - 7: $\varepsilon(A \cup e) \leftarrow \varepsilon(A) \cup \{\varepsilon_{N+1}\}$
 - 8: $G(A \cup e) \leftarrow$ the digraph obtained from $G(A)$ by adding a new node labelled I_{N+1} and an arc (I_{i_0}, I_{N+1})
 - 9: $N \leftarrow N + 1$
-

Algorithm 3 ADD_OTHER

-
- 1: **define** $J_0 := \{1 \leq i \leq N, V(I_i) \cap e \neq \emptyset\}$
 - 2: **define** $1 \leq i_0 \leq N$ such that $I_{i_0} = \text{LCA}_{G(A)}(\{I_i, i \in J_0\})$
 - 3: **define** $J := \bigcup_{i \in J_0} \{1 \leq j \leq N \text{ such that } I_j \text{ is on the dipath from } I_{i_0} \text{ to } I_i \text{ in } G(A)\}$
 - 4: $V(I_{i_0}) \leftarrow \bigcup_{i \in J} V(I_i)$
 - 5: **update** the archipelago A as follows:
 - 6: $E(A) \leftarrow E(A) \cup \{e\}$
 - 7: $\mathcal{I}(A) \leftarrow \mathcal{I}(A) \setminus \{V(I_j), j \in J \setminus \{i_0\}\}$.
 - 8: $\varepsilon(A) \leftarrow \varepsilon(A) \setminus \{\varepsilon_j, j \in J \setminus \{i_0\}\}$.
 - 9: $G(A) \leftarrow$ the digraph obtained from $G(A)$ by merging the nodes $\{I_j, j \in J\}$ into the node I_{i_0} .
-

Algorithm 4 ADD_CROSSING

-
- 1: **update** the archipelago A as follows:
 - 2: $E(A) \leftarrow E(A) \cup \{e\}$
-

Let us explain the algorithm. At the start, the archipelago A consists of the deserted island with entry $\{x^*\}$. We then augment A one edge at a time, by adding firstly the edges of A -type "new crossing" or "other" and then the edges of A -type "crossing". The procedures ADD_NEWCROSSING, ADD_OTHER and ADD_CROSSING (Algorithms 2, 3 and 4) are nothing but algorithmic translations of Propositions II.2.23, II.2.26 and II.2.22 respectively.

- Throughout the first While loop, A is an arborescent archipelago, as guaranteed by Propositions II.2.23 and II.2.26. The arborescence $G(A)$ is kept updated, as it is needed to merge islands in ADD_OTHER. It is important to understand that, every time A is augmented in that loop, the islands and entries of A may change, so the A -types of the remaining edges may change as well: the A -types of the edges in $E(H) \setminus E(A)$ must be redetermined at each iteration of that loop.
- Throughout the second While loop, A is an archipelago, as guaranteed by Proposition II.2.22. The digraph $G(A)$ is not kept updated as it is not useful anymore. This time, the decomposition in islands remains the same during the entire loop (we are adding crossing edges between already existing islands) so the A -types of the remaining edges do not change.

That last remark proves that, after the two While loops, all edges in $E(H) \setminus E(A)$ are of A -type either "cut" or "exterior" (the A -types "new crossing" and "other" have not reappeared during the second While loop). By Theorem II.2.21, this means maximality has been reached and A

coincides with $H[CC_H^{k-2}(x^*)]$, ensuring that the algorithm is correct.

Finally, let us determine the time complexity of COMPUTECC. Set $n = |V(H)|$ and $m = |E(H)|$.

- Let us first consider the three procedures ADD_NEWCROSSING, ADD_OTHER and ADD_CROSSING, to figure out how much time each update of A takes. Since basic operations on data structures can be language-dependent, let us clarify: when we use a list, what matters is the ability to remove the current element in $O(1)$ time; when we use an array, what matters is the ability to access and modify any element in $O(1)$ time.
 - $E(H) \setminus E(A)$ can be implemented as a list. Indeed, it is sensible to store $E(H) \setminus E(A)$ rather than $E(A)$ since this is the set in which edges are searched for throughout. Each update consists in removing the current edge which is done in $O(1)$ time.
 - $\mathcal{I}(A)$ can be implemented as an array of size n which contains, for each vertex $x \in V(H)$, the index of the island containing x (or 0 if $x \notin V(A)$). Each update requires going through the array once and is therefore done in $O(n)$ time.
 - $\varepsilon(A)$ can be implemented as an array of size n which contains, for each vertex $x \in V(H)$, a 1 if x is in an entry of A or a 0 otherwise. Each update requires going through the array once and is therefore done in $O(n)$ time.
 - $G(A)$ is an arborescence on the set of islands for the entire time that it is kept updated. Since $O(\frac{n}{k})$ islands are created in total (a new island can only be created during ADD_NEWCROSSING, and this requires $k-1$ previously undiscovered vertices), $G(A)$ can be implemented as an array of size $O(\frac{n}{k})$ containing the parent of each island, i.e. for all index $i \neq 1$ it contains the only index j such that $(I_j, I_i) \in E(G(A))$. In ADD_NEWCROSSING, updating $G(A)$ is clearly done in $O(1)$ time. In ADD_OTHER, updating $G(A)$ is done in $O(n)$ time: indeed, computing $|J_0| \leq k$ dipaths from the root takes $O(k \times \frac{n}{k}) = O(n)$ time, going through them a second time to compute i_0 and J takes $O(k \times \frac{n}{k}) = O(n)$ time again, and finally the merging process is performed in $O(\frac{n}{k})$ time since it only requires going through the array once.

All in all, performing ADD_CROSSING, ADD_NEWCROSSING or ADD_OTHER once is done in $O(n)$ time.

- Determining the A -type of a given edge e is easily done in $O(k)$ time since it boils down to determining, for all $x \in e$, which island/entry (if any) contains x .
- We can now conclude on the time complexity of COMPUTECC:
 - The initializations before the While loop are done in $O(m+n)$ time.
 - During the first While loop, finding an edge of A -type "new crossing" or "other" and then adding it takes $O(mk+n)$ time: indeed, at most m edges are gone through (with the A -type being determined for each one in $O(k)$ time as we have just seen) before finally finding one of A -type "new crossing" or "other" which is added in $O(n)$ time as shown above. Since at most m edges of A -type "new crossing" or "other" are added in total, the While loop ends in $O(m(mk+n)) = O(m^2k+mn)$ time.
 - During the second While loop, no A -types need to be redetermined, and each update of A is done in $O(1)$ time so that this loop ends in $O(m)$ time.

All in all, COMPUTECC runs in $O(m^2k+mn)$ time. Since the $(k-2)$ -linear connected component is a subset of the connected component, it is reasonable to assume that H is connected, which implies that $m \geq \frac{n-1}{k-1}$. Therefore, we can simplify $O(m^2k+mn)$ as $O(m^2k)$, so that our main algorithmic result on $(k-2)$ -linear connectivity can be stated as follows:

Theorem III.1.1. *There exists an algorithm that computes the $(k-2)$ -linear connected component of a given vertex in a k -uniform hypergraph H , and thus solves $\text{HYPCON}_{k,k-2}$, in $O(m^2k)$ time where $m = |E(H)|$. ■*

Remark. We end with two observations about the above algorithm:

- Note that the running time remains polynomial even if k is considered as part of the input. Therefore, this algorithm is also efficient in $k(n)$ -uniform hypergraphs on n vertices where $k(n) \rightarrow \infty$.
- It is not difficult to tweak the algorithm so as to also return an (x^*, x) -compatible walk for each $x \in CC_H^{k-2}(x^*)$. Indeed, it suffices, throughout the algorithm, to store in memory an (x^*, X) -compatible walk in A for each $X \subseteq V(A)$ such that $1 \leq |X| \leq k-1$ and $X \notin \varepsilon(A)$, which is possible by following the construction given in the proof of Proposition II.2.26. This version would also run in polynomial time, unless k is considered as part of the input.

III.1.2.2 Reduction to uniform hypergraphs

It should be noted that, even though we have defined $\text{HYPCON}_{k,\lambda}$ for a k -uniform input, non-uniform hypergraphs are not more difficult:

Proposition III.1.2. *The λ -linear connectivity problem on hypergraphs of rank k admits a polynomial reduction to the same problem on k -uniform hypergraphs i.e. to $\text{HYPCON}_{k,\lambda}$.*

Proof. Let H be a hypergraph of rank k , and let H' be the k -uniform hypergraph obtained from H by adding $k - |e|$ new vertices to each edge e . Since edge intersections are unchanged, it is clear that there exists a λ -linear xy -path in H if and only if there exists one in H' . ■

Therefore, Theorem III.1.1 actually holds for all hypergraphs of rank k .

III.1.3 The "Paths Avoiding Forbidden Pairs" problem PAFP

We now explore the strong link between $\text{HYPCON}_{k,\lambda}$ and PAFP.

III.1.3.1 Reducing $\text{HYPCON}_{k,\lambda}$ to PAFP

Suppose one is only interested in the algorithmic aspect of λ -linear connectivity, and does not possess the structural results from Section II.2. How to tackle the algorithmic complexity of $\text{HYPCON}_{k,\lambda}$? A reasonable attempt is to think in terms of line graphs, in which case PAFP appears naturally through the following reduction. For clarity, since we are manipulating graphs and hypergraphs simultaneously, the word *hyperedge* may be used to refer to edges of hypergraphs as opposed to graphs.

Notation III.1.3. Let $\varphi_{k,\lambda}$ be the function that associates to a k -uniform hypergraph H the bicolored graph G defined by:

- $V(G) = E(H)$;
- For all distinct $e_1, e_2 \in V(G)$, there is a blue (resp. red) edge between e_1 and e_2 in G if and only if $1 \leq |e_1 \cap e_2| \leq \lambda$ (resp. if and only if $|e_1 \cap e_2| > \lambda$).

Therefore, G is simply the line graph of H with added colors on its edges that carry information on the size of hyperedge intersections. See Figure III.1 for an example.

Proposition III.1.4. *For all $k \geq 3$ and $1 \leq \lambda \leq k-2$, $\text{HYPCON}_{k,\lambda}$ admits a polynomial reduction to PAFP.*

Proof. By definition, a sequence (e_1, \dots, e_L) of hyperedges of H is a λ -linear simple walk if and only if it is a blue induced path in $\varphi_{k,\lambda}(H)$. Indeed, "blue" means two consecutive hyperedges

intersect on between 1 and λ vertices (this defines a " λ -linear walk"), while "induced" means two non-consecutive hyperedges do not intersect (this defines "simple"). Therefore, there exists a λ -linear xy -simple-path in H ($x \neq y$) if and only if there exist hyperedges $e_x \ni x$ and $e_y \ni y$ in H such that there exists a blue induced path between e_x and e_y in $\varphi_{k,\lambda}(H)$. ■

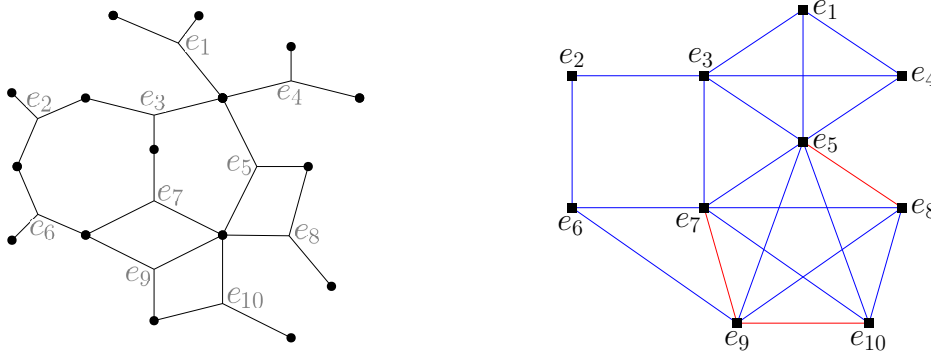


Figure III.1: On the left: a 3-uniform hypergraph H . On the right: the bicolored graph $G = \varphi_{3,1}(H)$.

Unfortunately, as we have mentioned when introducing this problem, PAFP is known to be NP-complete in general. The only notable tractable case in undirected graphs (stated in digraphs, but adaptable) is that of [Yin97]: it states that PAFP is solved in polynomial time on bicolored graphs where the red edges form a matching and a *skew symmetry condition* is satisfied. However, this result is of no help here, since general elements of $\text{Im}(\varphi_{k,\lambda})$ do not satisfy these conditions and nor do they easily reduce to bicolored graphs that do. Therefore, the previous reduction does not lead anywhere.

III.1.3.2 Reducing some instances of PAFP to $\text{HYPCON}_{k,\lambda}$

Instead, now that we know $\text{HYPCON}_{k,k-2}$ is solvable in polynomial time for all $k \geq 3$, it is interesting to turn the tables and examine the implications on PAFP:

Theorem III.1.5. *PAFP is tractable on bicolored graphs in $\bigcup_{k \geq 3} \text{Im}(\varphi_{k,k-2})$ for which a preimage can be computed in polynomial time.*

Proof. Let $G = \varphi_{k,k-2}(H)$ for some k -uniform hypergraph H , and let $e, e' \in V(G) = E(H)$ be distinct. As we have seen before, the blue induced paths between e and e' in G are exactly the $(k-2)$ -linear simple walks ($e = e_1, \dots, e_L = e'$) in H . Since $\text{HYPCON}_{k,k-2}$ requires a start vertex and an end vertex in its input, define, for all $(x, y) \in e \times e'$, the hypergraph $H_{x,y}$ obtained from H by removing all hyperedges incident to x and y other than e and e' , so that any $(k-2)$ -linear xy -simple-path in $H_{x,y}$ necessarily starts with e and ends with e' . There exists a blue induced path between e and e' in G if and only if there exists $(x, y) \in e \times e'$ such that there is a $(k-2)$ -linear xy -simple-path in $H_{x,y}$, which concludes since $\text{HYPCON}_{k,k-2}$ is solvable in polynomial time according to Theorem III.1.1. ■

Thanks to Theorem III.1.5, any sufficient condition for a bicolored graph G to be in $\text{Im}(\varphi_{k,k-2})$ for some $k \geq 3$, if it can be checked in polynomial time and comes with a way to reconstruct a preimage hypergraph in polynomial time, would add to the very short list of known tractable cases for PAFP.

For standard (i.e. non-colored) line graphs, the recognition problem has been studied extensively. Line graphs of graphs are characterized by a finite list of forbidden induced subgraphs ("FIS") [Bei70]. Line graphs of hypergraphs, on the other hand, are notoriously difficult to recognize.

There is no finite FIS characterization for line graphs of k -uniform hypergraphs if $k \geq 3$ [Lov77], and this recognition problem is even known to be NP-complete for $k = 3$ [PRT81]. However, adding information about the size of the pairwise intersections of hyperedges, instead of simply telling which ones are non-empty, changes the problem. For example, if all these sizes are given and in $\{0, 1\}$ (which is equivalent to asking that the hypergraph is linear) then, while remaining NP-complete for $k = 3$ [PRT81][HK97], the problem becomes easier in some cases:

- For $k = 3$, there is a finite FIS characterization for line graphs of 3-uniform linear hypergraphs if the minimum vertex-degree of the graph is at least 69, as well as a polynomial time algorithm to reconstruct the hypergraph in the positive case [NRS⁺82]. This bound has since been improved from 69 to 16 for the finite FIS characterization and 10 for the tractability of the recognition problem [SST09]. No analogous result can exist for $k \geq 4$, no matter what constant lower bound is put on the minimum vertex-degree [MT97].
- For any $k \geq 3$, there is a finite FIS characterization for line graphs of k -uniform linear hypergraphs if the minimum edge-degree of the graph is at least $f(k)$, where f is a polynomial function, as well as a polynomial-time (where the power increases with k) algorithm to reconstruct the hypergraph in the positive case [NRS⁺82]. This result has been generalized by replacing the linearity of the hypergraph with any constant upper bound on its *multiplicity* [BGM⁺21].

III.1.4 Conclusion and prospects

These last few results bring some hope of a finite FIS characterization for bicolored line graphs under some similar restriction on the minimum vertex-degree or edge-degree of the graph, and of a way to reconstruct a preimage in polynomial time, which would mean that PAFP is solvable in polynomial time on such bicolored graphs. Their proof uses the characterization of line graphs of hypergraphs by a Krausz partition in cliques [NRS⁺82], which generalizes to the bicolored version. Figure III.2 features some induced subgraphs that cannot appear in a bicolored graph from $\bigcup_{k \geq 3} \text{Im}(\varphi_{k, k-2})$. The case $k = 3$ is the most promising because the exact size of each intersection is also given (in $\{0, 1, 2\}$: 0 = no edge, 1 = blue edge, 2 = red edge), although it is NP-complete in general since instances with all blue edges correspond to the 3-uniform linear case for standard line graphs which we know is NP-complete. Looking beyond applications to PAFP, a general *weighted line graph* recognition problem, where each edge of the graph would wear a number between 1 and $k - 1$ indicating the exact size of the corresponding intersection, seems interesting in itself.

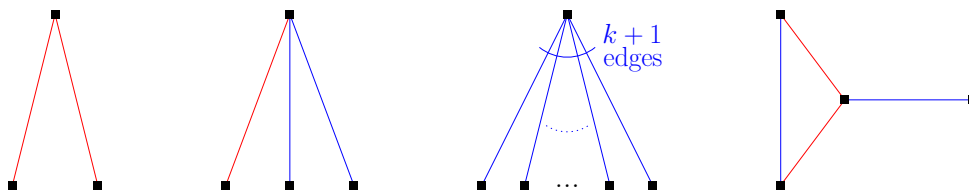


Figure III.2: Some induced subgraphs that cannot appear in $G \in \text{Im}(\varphi_{k, k-2})$. For instance, the left one is impossible because, in a k -uniform hypergraph with $k \geq 3$, if $|e_1 \cap e_2| = |e_2 \cap e_3| = k - 1$ then $|e_1 \cap e_3| \geq k - 2 > 0$.

As for the λ -linear connectivity problem, our polynomial-time algorithm for $\lambda = k - 2$ relies on the archipelago structure, of which we have mentioned in Subsection II.2.4 that it does not seem to generalize to $\lambda < k - 2$. Another approach would likely be needed if ones wishes to extend tractability to smaller values of λ . The linear case $\lambda = 1$ is of particular interest, as

linear paths appear a lot in the literature, however it might be difficult since we have seen that all values of λ reduce to this case. Finally, there is the question of the best possible algorithm for $\lambda = k - 2$. Our time complexity in $O(m^2k)$ might be optimal, since it seems difficult to avoid an "accept or put aside" process on the edges where each edge is potentially examined $O(m)$ times, and the mere computation of the intersection of two edges is in $O(k)$ time.

III.2 The Maker-Breaker game: complexity and bounds on optimization parameters

In this section, we explore some consequences of the structural study from Section II.1 regarding the unbiased Maker-Breaker game on hypergraphs of rank 3, and we look at what happens for higher ranks in comparison.

III.2.1 Presentation of the problem and state of the art

The algorithmic complexity of the Maker-Breaker game is vastly studied in the literature. The question can be asked for any particular class of hypergraphs. We are interested in the complexity depending on the size of the edges:

MAKERBREAKER_k
Input : a hypergraph H of rank k .
Output : YES if and only if H is a Maker win.

We have already mentioned the previously known results on this problem in Section I.1:

Theorem. [RW21] *MAKERBREAKER₆ is PSPACE-complete.* ■

Theorem. [Kut05] *MAKERBREAKER₃ is solvable in polynomial time when restricted to linear hypergraphs.* ■

Rahman and Watson [RW20] have since conjectured tractability without this linearity constraint, even for the more general game played on 3-CNF formulas which we mentioned in Subsection II.1.1. In particular:

Conjecture. *MAKERBREAKER₃ is solvable in polynomial time.*

Kutz proved his result on linear hypergraphs of rank 3 by providing an exact structural description of Breaker wins which is recognizable efficiently. Similarly, we are going to derive a polynomial-time algorithm on general hypergraphs of rank 3 from our structural result from Section II.1, combined with the chain recognition algorithm from Section III.1. This validates the above conjecture the exact way that Rahman and Watson suspected it could be proved. Overall complexity of the Maker-Breaker game does not stop at algorithmic complexity. For instance, it is also interesting, on a given class of hypergraphs, to gauge what optimal strategies look like and how complicated they are. The range of values attained by the hypergraph parameters τ_M and θ_M on said class is a good indicator of this. Recall that $\tau_M(H)$ (resp. $\theta_M(H)$) is defined as the minimum number of rounds (resp. tokens) in a winning Maker strategy on H . Therefore, τ_M measures the minimum length of a winning Maker strategy, whereas θ_M

somewhat measures the quantity of simultaneous threats that Maker must create in order to win. While θ_M had never been introduced prior to the present dissertation, τ_M is frequently studied in the literature. When it comes to hypergraph classes defined by edge size, which is our area of interest, it has been proved for example that the minimum value attained by τ_M over all k -uniform linear hypergraphs is approximately 2^k [EL74][Bec81]. However, we are not looking for asymptotic results in function of k , but rather for results in function of the number n of vertices with fixed $k \geq 3$. Corollary I.1.49 implies that any k -uniform hypergraph H on n vertices satisfies the straightforward inequalities $k \leq \theta_M(H) \leq \tau_M(H) \leq \left\lceil \frac{n}{2} \right\rceil$, where the lower bound k is trivially tight for both parameters. To our knowledge, the question of the maximum value attained by τ_M and θ_M for fixed k as a function of n has not been studied before. We are going to explain how the structural results from Section II.1 allow us to solve this question for $k = 3$, and how these values differ greatly from those that hold for rank $k \geq 4$.

III.2.2 A polynomial-time algorithm for the game on hypergraphs of rank 3[†]

The idea is simple. The algorithm comes as a joint corollary of the structural result on the Maker-Breaker game and the algorithmic result on chains. Indeed, Theorem II.1.12 tells us that it is all a question of whether a nunchaku/necklace appears in the first three rounds. Therefore, up to a factor $O(n^6)$ corresponding to all possibilities of both players' first three picks, MAKERBREAKER_3 reduces to the problem of identifying nunchakus and necklaces: this boils down to $\text{HYPCON}_{3,1}$ which is in polynomial time by Theorem III.1.1. Let us now give the detailed algorithm and its exact complexity.

Theorem III.2.1. *There is an algorithm that solves MAKERBREAKER_3 in $O(\max(n^5 m^2, n^6 \Delta))$ time where $n = |V(H)|$, $m = |E(H)|$ and $\Delta = \Delta_1(H)$.*

Proof. First of all, let us transform the input (non-marked) hypergraph of rank 3 into a (still non-marked) 3-uniform hypergraph. We can assume that all edges are of size 2 or 3, otherwise we have a trivial Maker win. We use item (ii) of Proposition I.1.22: if there are edges of size 2, then we can add two vertices x and y , and replace each edge e of size 2 with two edges $e \cup \{x\}$ and $e \cup \{y\}$ of size 3. At most, we have added two vertices and doubled the number of edges, so all relevant orders of magnitude are preserved. All in all, up to a preprocessing step in $O(m)$ time, we can assume that H is a non-marked 3-uniform hypergraph.

Since MAKERBREAKER_3 is obviously in $O(1)$ time on hypergraphs with less than 6 vertices, further assume $|V(H)| \geq 6$. By Theorem II.1.12, H is a Maker win if and only if:

$$\exists x_1 \in V(H), \forall y_1 \in V(H^{+x_1}) \setminus M(H^{+x_1}), \exists x_2 \in \dots, \forall y_2 \in \dots, \exists x_3 \in \dots, \forall y_3 \in \dots, \\ H^{+x_1-y_1+x_2-y_2+x_3-y_3} \text{ contains a fully marked edge, a nunchaku or a necklace.}$$

Suppose that we are given some $x_1, y_1, x_2, y_2, x_3, y_3$, and consider the marked hypergraph $H^{+x_1-y_1+x_2-y_2+x_3-y_3}$, in which x_1, x_2, x_3 are the only marked vertices:

- Clearly, $H^{+x_1-y_1+x_2-y_2+x_3-y_3}$ contains a fully marked edge or a nunchaku if and only if it contains a chain between two marked vertices. This can be tested in $O(m^2)$ time: for all $i \in \{1, 2, 3\}$, compute $CC_{H^{-y_1-y_2-y_3}}^1(x_i)$ using Algorithm COMPUTECC and check whether it contains some $x_j \neq x_i$.
- If $H^{+x_1-y_1+x_2-y_2+x_3-y_3}$ contains no fully marked edge and no nunchaku, then it contains a necklace if and only if it contains some edge $e = \{x, a, b\}$ with x marked such that

[†]These results are part of the article [GGS22b], which has been submitted to *Journal of Combinatorial Theory, Series A*.

there exists an xb -chain P that does not contain a (the necklace is then $P \cup e$). This can be tested in $O(\Delta m^2)$ time: for all $i \in \{1, 2, 3\}$ and for all edge $e = \{x_i, a, b\}$, compute $CC_{H-y_1-y_2-y_3-a}^1(x_i)$ using Algorithm COMPUTECC and check whether it contains b , then repeat when exchanging the roles of a and b .

This yields a $O(n^6(m^2 + \Delta m^2)) = O(n^6 \Delta m^2)$ time algorithm for MAKERBREAKER₃. However, we can easily apply a slight improvement to it. Indeed, since the linear connected components that we require do not depend on the x_i , it is redundant to compute them inside of the 6 nested loops. Instead, it is better to compute all of them once and for all at the beginning of the algorithm, store them all and then access them when needed. Computing $CC_{H-y_1-y_2-y_3}^1(x)$ and $CC_{H-y_1-y_2-y_3-a}^1(x)$ for all possibilities of y_1, y_2, y_3, a, x is done in $O(n^5 m^2)$ time, so the eventual running time is $O(n^5 m^2 + n^6 \Delta)$. ■

III.2.3 When Maker wants to win in a minimum number of rounds

We now look at the maximum value attained by τ_M for a fixed edge size and as a function of the number of vertices.

III.2.3.1 In hypergraphs of rank 3[†]

Theorem II.1.12 states that, if Maker wins, then she can guarantee the appearance of a nunchaku or a necklace after at most three rounds of play. Therefore, it all comes down to determining the value of τ_M on nunchakus (it is the same on necklaces).

The forcing strategy described in Proposition I.3.13 uses $\Theta(n)$ rounds, where n is the number of vertices of the considered forcing path, since Maker makes her way from one end of the path to the other one edge at a time and all vertices end up being picked. However, when $k = 3$, Maker can win faster by using a "dichotomy strategy" which halves the length of the nunchaku each round and yields a win in just $\Theta(\log_2(n))$ rounds for a nunchaku on n vertices. We now prove that this dichotomy strategy is optimal, from which we get the exact value of τ_M on nunchakus. We actually state the result for the more general case of 3-uniform marked hyperforests i.e. 3-uniform marked hypergraphs with no cycle, as it is no more complicated than the case of nunchakus. The idea is that Maker uses the above dichotomy strategy on a shortest nunchaku in the hyperforest.

Notation III.2.2. Let H be a 3-uniform marked hypergraph. We denote by $L(H)$ the length of a shortest nunchaku in H . If H contains no nunchaku, then $L(H) = \infty$ by convention.

Proposition III.2.3. *Let H be a 3-uniform marked hyperforest with no fully marked edge. If H contains a nunchaku then $\tau_M(H) = 1 + \lceil \log_2(L(H)) \rceil$, otherwise $\tau_M(H) = \infty$.*

Proof. Theorem II.1.1 states that H is a Maker win if and only if H contains a nunchaku. Therefore, it remains to show that if H contains a nunchaku then $\tau_M(H) = 1 + \lceil \log_2(L(H)) \rceil$. We proceed by induction on $L(H) \geq 1$. First suppose $L(H) = 1$: since a nunchaku of length 1 consists of a single edge, which contains exactly one non-marked vertex, H is a trivial Maker win hence $\tau_M(H) \in \{0, 1\}$. Moreover H has no fully marked edge, so $\tau_M(H) = 1 = 1 + \lceil \log_2(1) \rceil$. Now suppose $L(H) \geq 2$ (in particular H is not a trivial Maker win) and assume the result holds for all H' with $L(H') < L(H)$. Maker can ensure that the length of a shortest nunchaku is at least halved after one round:

[†] These results are part of the article [GGS22b], which has been submitted to *Journal of Combinatorial Theory, Series A*.

Claim 13. *There exists $x \in V(H) \setminus M(H)$ such that, for all $y \in V(H^{+x}) \setminus M(H^{+x})$, we have $L(H^{+x-y}) \leq \lceil \frac{L(H)}{2} \rceil$.*

Proof of Claim 13. Let N be a shortest nunchaku in H . Let $x \in \text{inn}(N)$ be in the exact middle of N if N is of even length, or as close to the middle as possible if N is of odd length. By picking x , Maker creates two nunchakus of length at most $\lceil \frac{L(H)}{2} \rceil$ whose sole common vertex is x , so Breaker's answer y cannot be contained in both of them at once. Therefore, at least one of these two nunchakus will be present in H^{+x-y} . \square

On the other hand, Breaker can ensure that the length of a shortest nunchaku is not more than halved after one round:

Claim 14. *For all $x \in V(H) \setminus M(H)$, there exists $y \in V(H^{+x}) \setminus M(H^{+x})$ such that $L(H^{+x-y}) \geq \lceil \frac{L(H)}{2} \rceil$.*

Proof of Claim 14. Let $x \in V(H) \setminus M(H)$. Note that, for any y , the nunchakus in H^{+x-y} are exactly the nunchakus in H^{+x} that do not contain y . Therefore, let \mathcal{N} be the collection of all nunchakus in H^{+x} whose length is less than $\lceil \frac{L(H)}{2} \rceil$: proving the claim comes down to showing the existence of some $y \in V(H^{+x}) \setminus M(H^{+x})$ such that all elements of \mathcal{N} contain y . We can assume $\mathcal{N} \neq \emptyset$, otherwise there is nothing to show.

First of all, notice that all elements of \mathcal{N} are x -nunchakus. Indeed, if some element of \mathcal{N} was not an x -nunchaku i.e. did not contain x , then it would be a nunchaku in H , which is impossible since it is of length less than $\lceil \frac{L(H)}{2} \rceil < L(H)$. Therefore, let $N_x \in \mathcal{N}$: we know N_x is an xm -nunchaku for some $m \in M(H)$. We now show that all elements of \mathcal{N} contain $y := o(x, \overrightarrow{xN_x m})$, which is non-marked since $M(N_x) = \{x, m\}$. Suppose for a contradiction that there exists $N'_x \in \mathcal{N}_x$ such that $y \notin V(N'_x)$: we know N'_x is an xm' -nunchaku for some $m' \in M(H)$.

- Suppose $V(N_x) \cap V(N'_x) \neq \{x\}$. Since $y \notin V(N'_x)$, we have $\text{start}(\overrightarrow{xN_x m}) \neq \text{start}(\overrightarrow{xN'_x m'})$, therefore Union Lemma I.3.40 ensures that $N_x \cup N'_x$ contains an x -cycle or an m -tadpole. Both possibilities contradict the fact that H is a hyperforest.
- Suppose $V(N_x) \cap V(N'_x) = \{x\}$. Then $N_x \cup N'_x$ is an mm' -chain in H^{+x} and $M(N_x \cup N'_x) = \{m, m', x\}$. Let N be the same as $N_x \cup N'_x$ except that x is non-marked: since $N_x \cup N'_x$ is a subhypergraph of H^{+x} , N is a subhypergraph of H . Therefore N is an mm' -nunchaku in H , of length equal to the sum of the lengths of N_x and N'_x . By definition of \mathcal{N} , N_x and N'_x are both of length less than $\lceil \frac{L(H)}{2} \rceil$, therefore N is of length less than $L(H)$, contradicting the definition of $L(H)$. \square

Recall that $\tau_M(H) = 1 + \min_{x \in V(H) \setminus M(H)} \max_{y \in V(H^{+x}) \setminus M(H^{+x})} \tau_M(H^{+x-y})$. Let x_0, y_0 min-maxing this formula. Claim 13 and minimality of x_0 ensure that $L(H^{+x_0-y_0}) \leq \lceil \frac{L(H)}{2} \rceil$. Claim 14 and maximality of y_0 ensure that $L(H^{+x_0-y_0}) \geq \lceil \frac{L(H)}{2} \rceil$. All in all, we get $L(H^{+x_0-y_0}) = \lceil \frac{L(H)}{2} \rceil < L(H)$ hence $\tau_M(H^{+x_0-y_0}) = 1 + \lceil \log_2(L(H^{+x_0-y_0})) \rceil = 1 + \lceil \log_2(\lceil \frac{L(H)}{2} \rceil) \rceil = \lceil \log_2(L(H)) \rceil$ by the induction hypothesis applied to $H^{+x_0-y_0}$. Therefore $\tau_M(H) = 1 + \tau_M(H^{+x_0-y_0}) = 1 + \lceil \log_2(L(H)) \rceil$, which concludes. \blacksquare

Corollary III.2.4. *For any nunchaku or necklace N of length L , we have $\tau_M(N) = 1 + \lceil \log_2(L) \rceil$.*

Proof. Nunchakus are hyperforests, so they are covered by Proposition III.2.3. Moreover, Proposition II.1.7 ensures that a necklace of length L and a nunchaku of length L have the same non-marked reduction, hence the result for necklaces by Corollary I.1.54. \blacksquare

We can now conclude for general 3-uniform marked hypergraphs: Maker can make a nunchaku/necklace N appear after at most 3 rounds, and then she can focus exclusively on N to win in a further logarithmic number of rounds. The case of nunchakus attests that the obtained upper bound is basically tight.

Theorem III.2.5. *Let H be a 3-uniform marked hypergraph such that $|V(H) \setminus M(H)| \geq 6$. If H is a Maker win, then $\tau_M(H) \leq 3 + \lceil \log_2(|V(H) \setminus M(H)| - 5) \rceil$. Moreover, for all $n \geq 3$, there exists a 3-uniform marked hypergraph on n vertices that attains this upper bound up to an additive 3 rounds.*

Proof. By Theorem II.1.12, Maker has a strategy ensuring that, after three rounds of play with successive picks $x_1, y_1, x_2, y_2, x_3, y_3$, there is a fully marked edge, a nunchaku or a necklace in $H^{+x_1-y_1+x_2-y_2+x_3-y_3}$. Monotonicity Lemma I.1.50 thus ensures that, to conclude the proof of the first assertion, it suffices to show that any nunchaku or necklace N in $H^{+x_1-y_1+x_2-y_2+x_3-y_3}$ satisfies $\tau_M(N) \leq \lceil \log_2(|V(H) \setminus M(H)| - 5) \rceil$:

- A nunchaku N is of length $\frac{|V(N)|-1}{2}$, so it satisfies $\tau_M(N) = \lceil \log_2(|V(N)| - 1) \rceil$ according to Corollary III.2.4. Moreover, a nunchaku N in $H^{+x_1-y_1+x_2-y_2+x_3-y_3}$ has two marked vertices and all its other vertices are in $V(H) \setminus (M(H) \cup \{x_1, y_1, x_2, y_2, x_3, y_3\})$, so it satisfies $|V(N)| \leq 2 + (|V(H) \setminus M(H)| - 6)$ hence $\tau_M(N) \leq \lceil \log_2(|V(H) \setminus M(H)| - 5) \rceil$.
- A necklace N is of length $\frac{|V(N)|}{2}$, so it satisfies $\tau_M(N) = \lceil \log_2(|V(N)|) \rceil$ according to Corollary III.2.4. Moreover, a necklace N in $H^{+x_1-y_1+x_2-y_2+x_3-y_3}$ has one marked vertex and all its other vertices are in $V(H) \setminus (M(H) \cup \{x_1, y_1, x_2, y_2, x_3, y_3\})$, so it satisfies $|V(N)| \leq 1 + (|V(H) \setminus M(H)| - 6)$ hence $\tau_M(N) \leq \lceil \log_2(|V(H) \setminus M(H)| - 5) \rceil$.

As for the final assertion, it suffices to consider nunchakus. ■

Remark. Let us make two observations on Theorem III.2.5:

- It is not difficult to find a 3-uniform non-marked hypergraph on n vertices that also attains the upper bound up to an additive 3 rounds, for n large enough. For example, consider the union of two x -cycles of equal length whose only common vertex is x .
- We have already seen, through Theorem III.2.1, that the outcome of the Maker-Breaker game on 3-uniform marked hypergraphs can be determined in polynomial time. However, it should be noted that Theorem III.2.5 alone is enough to obtain a *quasi-polynomial* time $n^{O(\log(n))} = e^{O(\log^2(n))}$, without using any of the results about existence of chains from Sections II.2 and III.1. Indeed, it suffices to check all sequences of $5 + 2\lceil \log_2(|V(H) \setminus M(H)| - 5) \rceil$ moves to see if Maker has a winning strategy.

III.2.3.2 In hypergraphs of rank $k \geq 4$

We have just seen that, on 3-uniform marked hypergraphs, the maximum value of τ_M for Maker wins is basically attained by forcing paths i.e. nunchakus, and that this value is roughly $\log_2(n)$ where n is the number of vertices. On k -uniform marked hypergraphs i.e. hypergraphs of rank k where $k \geq 4$, we are about to see that the maximum value of τ_M for Maker wins is also attained by forcing paths, however it coincides with the trivial upper bound $\lceil \frac{n}{2} \rceil$. This comes from the fact that the dichotomy strategy used for $k = 3$ has no equivalent for $k \geq 4$: the forcing strategy happens to be the only winning strategy for Maker, and it uses up all the vertices in presence.

Proposition III.2.6. *Let $k \geq 4$. A k -uniform forcing path P of length $L \geq 1$ satisfies $\tau_M(P) = L$.*

Proof. We proceed by induction on L . The result is obvious for $L = 1$, so assume $L \geq 2$ and the

result to be true for k -uniform forcing paths of length less than L . By definition of a forcing path of length L , we can write: $V(P) = \{a_1, \dots, a_{L+k-2}, b_1, \dots, b_L\}$, $M(P) = \{a_1, \dots, a_{k-2}, a_{L+k-2}\}$ and $E(P) = \{e_1, \dots, e_L\}$ where $e_i = \{a_i, \dots, a_{i+k-2}, b_i\}$ for all $1 \leq i \leq L$.

Claim 15. *The only winning first pick for Maker on P is a_{k-1} .*

Proof of Claim 15. We already know that a_{k-1} is a winning first pick for Maker (forcing strategy). The idea is that, if Maker does not pick a_{k-1} in the first round, then Breaker can pick a_{k-1} himself in the first round and win the game. Therefore, let $x \in V(P) \setminus (M(P) \cup \{a_{k-1}\})$: we want to show that $P^{+x-a_{k-1}}$ is a Breaker win. If $L < k$ then this is trivial since all edges of P contain a_{k-1} , so assume $L \geq k$ hence $E(P^{+x-a_{k-1}}) = \{e_k, \dots, e_L\} \neq \emptyset$. By Proposition I.1.52, it suffices to exhibit a pairing Π which is complete in $P^{+x-a_{k-1}}$. When defining Π , we have to be careful that it does not use the vertex x which is now marked:

- If $x = a_j$ for some $k \leq j \leq L + k - 3$, then define $\Pi := \{\{a_i, b_i\}, k \leq i \leq j - 1\} \cup \{\{a_{i+1}, b_i\}, j \leq i \leq L\}$ (see Figure III.3).
- If $x = b_j$ for some $k \leq j \leq L$, then define $\Pi := \{\{a_i, b_i\}, k \leq i \leq j - 1\} \cup \{a_j, a_{j+1}\} \cup \{\{a_{i+1}, b_i\}, j + 1 \leq i \leq L\}$.
- Otherwise i.e. if x is isolated in $P^{+x-a_{k-1}}$, then define $\Pi := \{\{a_i, b_i\}, k \leq i \leq L\}$.

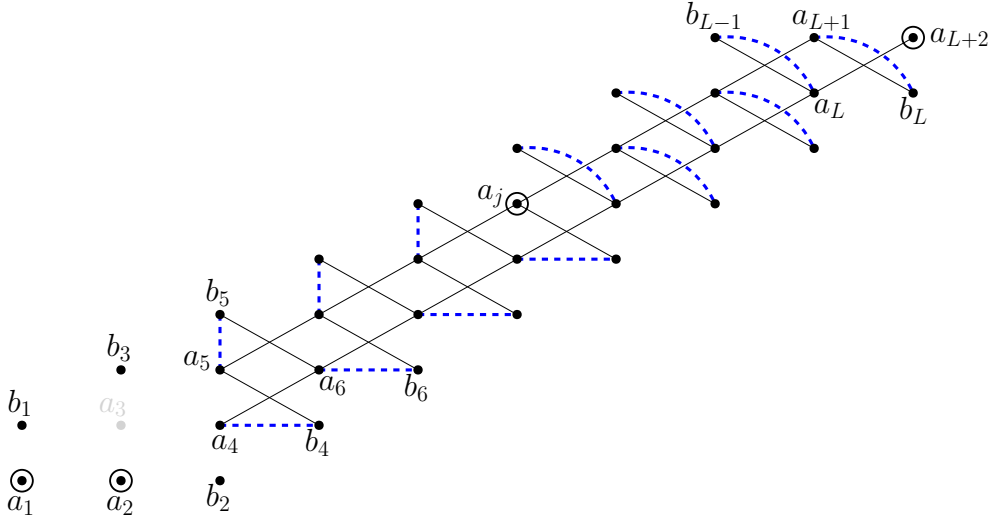


Figure III.3: Definition of Π if $x = a_j$, for $k = 4$.

In all cases, Π is complete in $P^{+x-a_{k-1}}$. □

Therefore, Maker must pick a_{k-1} in the first round, and obviously b_1 is a Breaker answer that maximizes the duration of the game since it is now the only non-marked vertex of e_1 . In conclusion, we have $\tau_M(P) = 1 + \tau_M(P^{+a_{k-1}-b_1})$. Since $P^{+a_{k-1}-b_1}$ consists of a forcing path of length $L - 1$ plus the inconsequential isolated vertex a_1 , the induction hypothesis yields $\tau_M(P^{+a_{k-1}-b_1}) = L - 1$ hence $\tau_M(P) = L$, which concludes. ■

Theorem III.2.7. *For all $k \geq 4$ and for all $n \geq 1$, there exists a non-marked hypergraph H of rank k on n vertices satisfying $\tau_M(H) = \lfloor \frac{n}{2} \rfloor$.*

Proof. Let P be a k -uniform forcing path of length $L := \lfloor \frac{n}{2} \rfloor$. We know that $\tau_M(P) = \lfloor \frac{n}{2} \rfloor$ by Proposition III.2.6. Let H be the non-marked reduction of P , plus an isolated vertex if n is even. Since $|V(P) \setminus M(P)| = (2L + k - 2) - (k - 1) = 2\lfloor \frac{n}{2} \rfloor - 1$, we have $|V(H)| = n$. Moreover, by Corollary I.1.54, we have $\tau_M(H) = \tau_M(P) = \lfloor \frac{n}{2} \rfloor$. ■

Remark. Theorem III.2.7 actually still holds, for n large enough, if we ask for the hypergraph to be k -uniform. First of all, notice that addressing the case $k = 4$ suffices thanks to Proposition I.1.56. Now, for $k = 4$, it is not too difficult to get rid of the three marked vertices in the forcing path. Let us use the same notations as in the proof of Proposition III.2.6, so that $M(P) = \{a_1, a_2, a_{L+1}\}$. The idea is to unmark these three vertices, but create three new vertices $\overline{a_1}, \overline{a_2}, \overline{a_{L+1}}$ and add a few well-chosen edges such that the following action is optimal in the first three rounds: Maker picks a_{L+1} , Breaker picks $\overline{a_{L+1}}$, Maker picks a_1 , Breaker picks $\overline{a_1}$, Maker picks a_2 , Breaker picks $\overline{a_2}$. We claim that this can be achieved by adding the following edges:

- all edges of size 4 containing a_{L+1} and $\overline{a_{L+1}}$;
- all edges of size 4 containing a_{L+1}, a_1 and $\overline{a_1}$;
- the edge $\{a_{L+1}, a_1, a_2, \overline{a_2}\}$.

After that, what remains is a forcing path, so play goes on according to the forcing strategy and all vertices end up being picked.

III.2.4 When Maker wants to win using a minimum number of tokens

We now look at the maximum value attained by θ_M for a fixed edge size and as a function of the number of vertices.

III.2.4.1 In hypergraphs of rank 3

On hypergraphs of rank 3 i.e. 3-uniform marked hypergraphs, a remarkable consequence of our structural result is that if Maker wins then three tokens are always sufficient:

Theorem III.2.8. *Let H be a 3-uniform marked hypergraph. If H is a Maker win, then $\theta_M(H) = 3$. Actually, the following stronger result holds: if H is a Maker win, then Maker has a winning strategy that uses 3 tokens and at most $3 + \lceil \log_2(|V(H) \setminus M(H)| - 5) \rceil$ rounds.*

Proof. By Theorem II.1.12, Maker has a strategy ensuring that, after at most three rounds of play on H , either Maker has won or there is a nunchaku/necklace N . We can assume the latter.

- First suppose N is a vw -nunchaku. Each of v and w was either already marked in H or has a token on it, so assume the worst-case scenario for Maker in terms of token usage, which is that v and w both have tokens on them. We claim that the dichotomy strategy from Theorem III.2.5 still works with only three tokens. Indeed, Maker can place a third token (which was possibly placed already somewhere outside N) on x defined as the centermost inner vertex of N , creating a vx -nunchaku and an xw -nunchaku. At least one of the two, say the former, is left intact by Breaker's answer. This frees up the token on w for Maker to place at the center of the vx -nunchaku, etc.
- Now suppose N is a v -necklace. Maker can place a second token (which was possibly placed already somewhere outside N) on an inner vertex of N other than v , creating two nunchakus. Breaker cannot destroy both, so we are back in the previous case after this round. ■

III.2.4.2 In hypergraphs of rank $k \geq 4$

Since Maker never needs more than three tokens on hypergraphs of rank 3, we ask a natural question: is there, for all k , some constant $f(k)$ so that every Maker win H of rank k satisfies $\theta_M(H) \leq f(k)$? It turns out the answer is negative: for $k \geq 4$, in general, θ_M does not depend solely on the size of the edges, it also depends on the number of vertices. To prove this, we

build an example of a k -uniform hypergraph on which Maker needs a number of tokens that is linear in the number of vertices (recall that $\lceil \frac{n}{2} \rceil$ is the trivial upper bound on n vertices, so a linear dependence is the most we could have got). The difficult case is $k = 4$, from which we easily derive an example for $k \geq 5$.

Notation III.2.9. For any $N \geq 2$, we define the 4-uniform non-marked hypergraph $H_{4,N}$ on $6N + 8$ vertices constructed as follows (see Figure III.4):

1. Start from vertices $v, w, a_1, \dots, a_{2N-1}, b_1, \dots, b_{2N}, c_1, \dots, c_N$ and edges e_1, \dots, e_{2N} defined by:
 - $e_1 = \{v, a_1, b_1, c_1\}$;
 - $e_i = \{a_{i-1}, a_i, b_i, c_i\}$ for all $2 \leq i \leq N$;
 - $e_i = \{a_{i-1}, a_i, b_i, c_{i-N}\}$ for all $N + 1 \leq i \leq 2N - 1$;
 - $e_{2N} = \{a_{2N-1}, b_{2N}, c_N, w\}$.
2. Add vertices $\bar{v}, \bar{w}, \bar{c}_1, \dots, \bar{c}_N, d_1, d_2, d_3, d_4, d_5$ and, for each $u \in \{v, w, c_1, \dots, c_N\}$, add the edges of D_u which is defined by $V(D_u) = \{u, \bar{u}, d_1, d_2, d_3, d_4, d_5\}$ and $E(D_u) = \{\{u, \bar{u}, d_i, d_j\}, 1 \leq i < j \leq 5\}$.

We also define the following pairing, which is incomplete in $H_{4,N}$ as it covers all edges in $H_{4,N}$ apart from e_N (see Figure III.4):

$$\Pi_N := \{\{v, \bar{v}\}, \{w, \bar{w}\}\} \cup \bigcup_{1 \leq i \leq N} \{\{c_i, \bar{c}_i\}\} \cup \bigcup_{1 \leq i \leq N-1} \{\{b_i, a_i\}\} \cup \bigcup_{N \leq i \leq 2N-1} \{\{a_i, b_{i+1}\}\}.$$

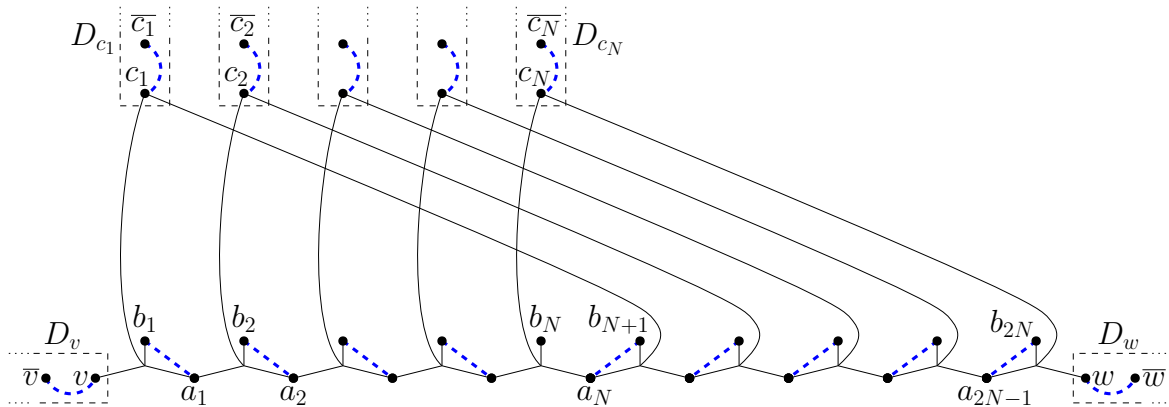


Figure III.4: The hypergraph $H_{4,N}$ and the pairing Π_N (bold dash lines represent the pairs). For clarity, the vertices d_1, d_2, d_3, d_4, d_5 and all edges of the D_u are not represented.

By design, each D_u in the hypergraph $H_{4,N}$ is a danger at u with the following property:

Lemma III.2.10. *Let D be a non-marked hypergraph defined by $V(D) = \{u, \bar{u}, d_1, d_2, d_3, d_4, d_5\}$ and $E(D) = \{\{u, \bar{u}, d_i, d_j\}, 1 \leq i < j \leq 5\}$. Then, for all $1 \leq i \leq 5$, we have $\tau_M(D^{+u-d_i}) = 3$. In other words, playing on D , if Maker picks u and Breaker does not immediately answer by picking \bar{u} , then Maker wins in three further rounds.*

Proof. Suppose that, in the first round of the game played on D , Maker picks u and Breaker picks some d_i . Then Maker picks \bar{u} . Whatever happens next, the four vertices picked by Maker in total will form an edge by definition of D . ■

Now, let us explain the role of the subhypergraphs D_u in the hypergraph $H_{4,N}$.

- If, in place of step 2 in the construction of $H_{4,N}$, we had simply marked the vertices v, w, c_1, \dots, c_N , then we would have obtained a "nunchaku" (with the c_i being marked, it

is as if the edges e_i did not contain them i.e. as if they formed a true nunchaku), on which we know Maker wins.

- Instead of marking $u \in \{v, w, c_1, \dots, c_N\}$, we use the gadget D_u which, to an extent, emulates a token sitting on u . Indeed, suppose that no move has been played inside D_u for some u , and that Maker places a token on u . By Lemma III.2.10, this forces Breaker to pick \bar{u} if he wants to avoid a quick defeat. Now, as a result: there is a token on u , all edges of D_u are destroyed, and it is Maker's turn. In conclusion, everything is as if we had simply marked u in our initial construction of $H_{4,N}$ instead of attaching D_u to it, except for one important thing: this has come to the cost of one token for Maker.

So, how many tokens does Maker need? An easy way to win using $N + 3$ tokens is to start by placing one on each of v, w, c_1, \dots, c_N and then use the extra token to win on the resulting "nunchaku". Actually, as we are about to show comfortably, $N + 2$ tokens suffice with a slight modification of this method. We will then show that $N + 2$ tokens are also necessary for Maker, which is more difficult.

Proposition III.2.11. *For all $N \geq 2$, we have $\theta_M(H_{4,N}) \leq N + 2$.*

Proposition III.2.12. *For all $N \geq 2$, we have $\theta_M(H_{4,N}) \geq N + 2$.*

Corollary III.2.13. *For all $N \geq 2$, we have $\theta_M(H_{4,N}) = N + 2$.*

Proof of Proposition III.2.11. We describe a winning strategy for Maker using only $N + 2$ tokens. She starts by placing a token on v . We can assume that Breaker picks \bar{v} : indeed, if he fails to do so, then Maker uses three tokens other than the one on v (note that $N + 2 \geq 4$) to win in three further rounds as per Lemma III.2.10. Maker then places a second token on c_1 : similarly, if Breaker fails to pick \bar{c}_1 then Maker wins in three further rounds. Continuing so, Maker places tokens on c_2, \dots, c_N and we can assume that Breaker responds by picking $\bar{c}_2, \dots, \bar{c}_N$ respectively. Maker now places a final token on a_1 , threatening to complete e_1 on her next go, thus forcing Breaker to pick b_1 . At this point, Maker has placed a total of $N + 2$ tokens on v, c_1, \dots, c_N, a_1 . She will not use any additional token, so in each round to follow she will move a token that was already placed. For the next $2N - 1$ rounds, Maker moves a token respectively: from v to a_2 , from a_1 to a_3 , from a_2 to a_4, \dots , from a_{2N-3} to a_{2N-1} , from a_{2N-2} to w . This forces Breaker to pick $b_2, b_3, b_4 \dots, b_{2N-1}, b_{2N}$ respectively in order not to lose in the next round each time. Finally, now that Maker has a token on w , she uses three of the other tokens to play her next three moves inside D_w and win as per Lemma III.2.10. ■

Proof of Proposition III.2.12. To alleviate notations, write $H = H_{4,N}$ and $\Pi = \Pi_N$. Suppose that Maker adopts a strategy using at most $N + 1$ tokens, and that Breaker adopts the following two-phase strategy (where it remains to be seen that Phase 2, which wins the game according to Proposition I.1.52, is actually reached):

Phase 1: If Breaker can pick a vertex such that the resulting marked hypergraph admits a complete pairing, then he picks an arbitrary such vertex and switches to Phase 2.

Otherwise, Breaker plays according to the incomplete pairing Π . This means that, if Maker has just placed a token on a vertex of a pair from Π , then Breaker answers by picking the other vertex of that pair. If Maker's pick is in no pair from Π , or if it is but its twin has already been picked by Breaker (which can happen since Maker can vacate a vertex and then re-place a token on it later), then Breaker picks an arbitrary vertex.

Phase 2: Breaker follows a complete pairing until the end.

Now, let the game play out. As we have done before, when Maker moves a token that was already placed, we decompose the move into two consecutive actions: Maker removes a token then places a token, where we consider the "remove" action as part of the previous round. Therefore, the action during round $t \geq 1$ goes as follows, in that order:

1. Maker places a token on some free vertex x_t . (We define $X_t \subseteq \{x_1, \dots, x_t\}$ as the set of all vertices on which Maker has tokens at this point, with $X_0 := \emptyset$.)
2. Breaker deletes some free vertex y_t . (We define $Y_t := \{y_1, \dots, y_t\}$, with $Y_0 := \emptyset$.)
3. Optionally, Maker removes one of her tokens. (We define H_t as the updated marked hypergraph at the end of round t , with $H_0 := H$.)

Suppose for a contradiction that Maker wins. Let T be the duration of the game i.e. Maker completes an edge during round T . Note that x_t, X_t are defined for $0 \leq t \leq T$ while y_t, Y_t, H_t are defined for $0 \leq t \leq T - 1$.

Claim 16. All of Breaker's picks are made according to the incomplete pairing Π and, for all $1 \leq t \leq T$ and for all $y \in V(H_{t-1}^{+x_t}) \setminus M(H_{t-1}^{+x_t})$, the marked hypergraph $H_{t-1}^{+x_t-y}$ admits no complete pairing. Moreover, for all $0 \leq t \leq T - 1$, Π covers all edges in H_t apart from e_N .

Proof of Claim 16. Maker winning means Breaker is stuck in Phase 1 for the whole duration of the game. As for the last assertion, we know it holds for $t = 0$, and the fact that Breaker plays according to Π ensures that it remains true throughout the game. \square

The definition of Π means that the edge completed by Maker is necessarily e_N , and that Maker only places a token on b_N as her very last move:

Claim 17. We have $e_N \subseteq X_T$, moreover $x_1, \dots, x_{T-1} \neq b_N$ and $x_T = b_N$.

Proof of Claim 17. Since Breaker plays according to Π which covers all edges in H apart from e_N , the only edge that Maker can complete is e_N . Now, suppose for a contradiction that $x_t = b_N$ for some $1 \leq t \leq T - 1$. Since $t < T$, there exists $y \in e_N \setminus X_t$. We know Π covers all edges in H_{t-1} apart from e_N , moreover Π does not use b_N and $e_N \notin E(H_{t-1}^{+x_t-y})$, so Π is complete in $H_{t-1}^{+x_t-y}$. This contradicts Claim 16. \square

Let us introduce the following notations, for all $1 \leq i \leq N$ (see Figure III.5):

- $I_i^L := \{a_i, \dots, a_{N-1}\} \cup \{b_i, \dots, b_N\}$;
- $I_i^R := \{a_N, \dots, a_{N+i-1}\} \cup \{b_N, \dots, b_{N+i}\}$;
- $I_i := I_i^L \cup I_i^R$;
- $t(i) := \inf\{1 \leq t \leq T, x_t \in I_i\} = \inf\{1 \leq t \leq T, X_t \cap I_i \neq \emptyset\}$.

Note that $t(i)$ is well defined since $b_N \in I_i$ for all $1 \leq i \leq N$ and $b_N \in X_T$ by Claim 17.

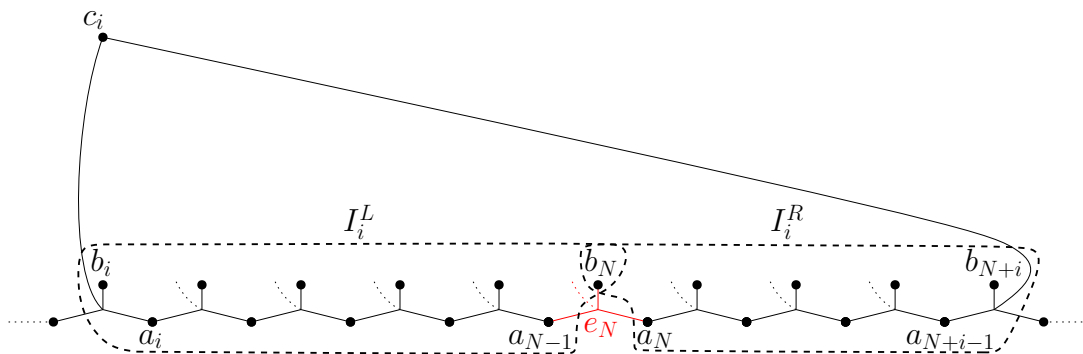


Figure III.5: Definition of I_i^L and I_i^R .

In the proof of Proposition III.2.11, we have seen how controlling the c_i (i.e. having tokens on c_1, \dots, c_N) was key for Maker: it allowed her to force all of Breaker's moves and thus make

progress from left to right until she won the game. We use the same principle here. The idea for the end of the proof is as follows. The first time that Maker plays some x_t inside the "interval" I_i , Maker must control c_i (Claim 18), otherwise Breaker could answer by picking c_i himself which breaks the path on both sides of x_t and creates a complete pairing. Since $b_N \in I_i$ for all i , this means each c_i is controlled by Maker at some point. However, Maker uses at most $N + 1$ tokens in total, therefore she necessarily removes a token from some c_i during the game (Claim 19). The first time this happens, said c_i is freed up to help building a complete pairing since Breaker has picked \bar{c}_i already. We now proceed with the details.

Claim 18. *For all $1 \leq i \leq N$, we have $c_i \in X_{t(i)-1}$.*

Proof of Claim 18. Suppose for a contradiction that $c_i \notin X_{t(i)-1}$ for some $1 \leq i \leq N$. Using the fact that $H_{t(i)-1}$ has no marked vertex in I_i by minimality of $t(i)$, we are going to build a complete pairing in $H_{t(i)-1}^{+x_{t(i)}-c_i}$ from Π by modifying only the pairs inside I_i (see Figure III.6):

- First case: $x_{t(i)} = b_i$ (the case $x_{t(i)} = b_{N+i}$ is analogous). Let Π' be the same as Π except that the pairs inside I_i are replaced by the pairs $\{b_l, a_l\}$ for $i + 1 \leq l \leq N + i - 1$.
- Second case: $x_{t(i)} = b_j$ for some $i < j < N + i$. Let Π' be the same as Π except that the pairs inside I_i are replaced by the pairs $\{a_{j-1}, a_j\}$, $\{a_l, b_{l+1}\}$ for $i \leq l \leq j - 2$, $\{b_l, a_l\}$ for $j + 1 \leq l \leq N + i - 1$.
- Third case: $x_{t(i)} = a_j$ for some $i \leq j \leq N + i - 1$. Let Π' be the same as Π except that the pairs inside I_i are replaced by the pairs $\{a_l, b_{l+1}\}$ for $i \leq l \leq j - 1$, $\{b_l, a_l\}$ for $j + 1 \leq l \leq N + i - 1$.

In all cases, the newly defined pairs cover the edges $e_{i+1}, \dots, e_{N+i-1}$ in $H_{t(i)-1}^{+x_{t(i)}-c_i}$ (note that this includes e_N which was the only edge not covered by Π). Moreover $e_i, e_{N+i} \notin E(H_{t(i)-1}^{+x_{t(i)}-c_i})$ since $c_i \in e_i \cap e_{N+i}$, therefore Π' covers all edges in $H_{t(i)-1}^{+x_{t(i)}-c_i}$. This contradicts Claim 16. \square

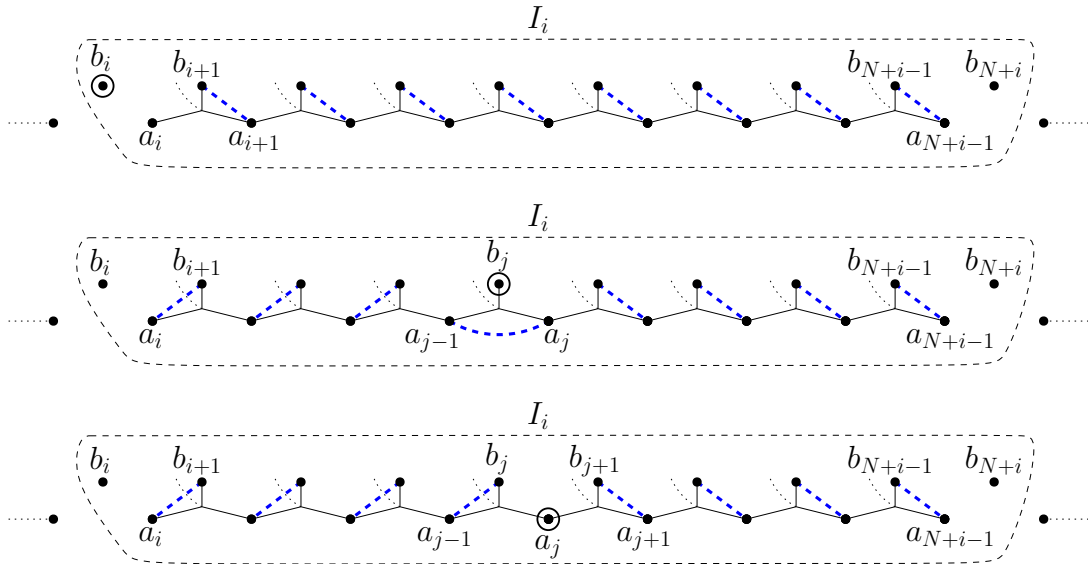


Figure III.6: The pairing used inside I_i in the proof of Claim 18. Three cases from top to bottom: $x_{t(i)} = b_i$, $x_{t(i)} = b_j$ ($i < j < N + i$), $x_{t(i)} = a_j$.

Claim 19. *There exist $0 \leq t \leq T - 1$ and $1 \leq i \leq N$ such that $c_i \in X_t$ and $c_i \notin X_{t+1}$: in other words, at some point during the game, Maker removes a token from some c_i . Moreover, for such (t, i) with t minimal, we have $X_t \cap I_i^L = \emptyset$ or $X_t \cap I_i^R = \emptyset$.*

Proof of Claim 19. Suppose for a contradiction that the first assertion is false: then $c_i \in X_T$ for all $1 \leq i \leq N$ by Claim 18. Since $e_N \subseteq X_T$ by Claim 17, we get $|X_T| \geq N + 3$, contradicting the fact that Maker only uses $N + 1$ tokens.

As for the second assertion, let (t, i) such that $c_i \in X_t$ and $c_i \notin X_{t+1}$, with t minimal. Suppose for a contradiction that there exist $u^L \in X_t \cap I_i^L$ and $u^R \in X_t \cap I_i^R$. Note that $u^L \neq u^R$: indeed, $I_i^L \cap I_i^R = \{b_N\}$ by definition, and Claim 17 ensures that $b_N \notin X_t$ since $t < T$. Now, for all $1 \leq j \leq N$:

- We have $t \geq t(j)$. Indeed, if $j \leq i$ then $I_j \supseteq I_j^L \supseteq I_i^L \ni u^L$, and if $j \geq i$ then $I_j \supseteq I_j^R \supseteq I_i^R \ni u^R$.
- We know $c_j \in X_{t(j)-1}$ by Claim 18. By minimality of t , this token on c_j is not removed before round t , hence $c_j \in X_t$.

In conclusion, we have $\{c_1, \dots, c_N, u^L, u^R\} \subseteq X_t$ hence $|X_t| \geq N + 2$, again contradicting the fact that Maker only uses $N + 1$ tokens. \square

Let (t, i) satisfying Claim 19, with t minimal, and suppose that $X_t \cap I_i^L = \emptyset$ (the case $X_t \cap I_i^R = \emptyset$ is analogous). We are going to build a complete pairing in H_t , using several facts:

- Recall that H_t is the marked hypergraph obtained at the end of round t i.e. just after the token on c_i has been removed, so $M(H_t) = X_t \setminus \{c_i\}$. In particular all vertices in $I_i^L \cup \{c_i\}$ are non-marked in H_t and can thus be used in the pairing.
- Since $c_i \in X_t$, there exists $1 \leq t' \leq t$ such that $x_{t'} = c_i$. This implies $y_{t'} = \bar{c}_i$ because Breaker plays according to Π . As a result, $H_{t'}$ has no edge inside D_{c_i} , and neither does H_t since $t \geq t'$.

Let Π' be the same as Π except that $\{c_i, \bar{c}_i\}$ and the pairs inside I_i^L are replaced by $\{c_i, b_i\}$ and $\{a_l, b_{l+1}\}$ for $i \leq l \leq N - 1$, as in Figure III.7.

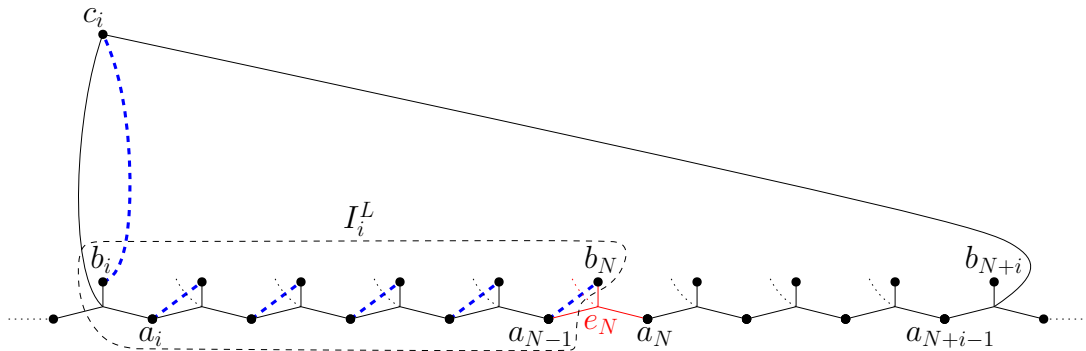


Figure III.7: The pairing used inside $I_i^L \cup \{c_i\}$ if $X_t \cap I_i^L = \emptyset$.

These new pairs cover e_i, \dots, e_N in H_t , so the above facts ensure that Π' is complete in H_t . In particular, if y denotes the twin of x_{t+1} in Π' (or an arbitrary vertex if x_{t+1} is in no pair of Π'), then Π' is also complete in $H_t^{+x_{t+1}-y}$, which contradicts Claim 16. \blacksquare

Notation III.2.14. For any $k \geq 5$ and any $N \geq 2$, we define the k -uniform non-marked hypergraph $H_{k,N}$ built from $H_{4,N}$ through $k - 4$ consecutive applications of the construction from Proposition I.1.56.

Proposition III.2.15. For all $k \geq 4$ and for all $N \geq 2$, we have $|V(H_{k,N})| = 6N + 2k$ and $\theta_M(H_{k,N}) = N + k - 2$.

Proof. By Proposition I.1.56, we have $|V(H_{k,N})| = |V(H_{4,N})| + 2(k - 4) = 6N + 8 + 2(k - 4) = 6N + 2k$ and $\theta_M(H_{k,N}) = \theta_M(H_{4,N}) + (k - 4) = N + 2 + (k - 4) = N + k - 2$ using Corollary III.2.13. \blacksquare

Theorem III.2.16. *For all $k \geq 4$ and for all n large enough, there exists a k -uniform non-marked hypergraph H on n vertices satisfying $\lfloor \frac{n}{6} \rfloor \leq \theta_M(H) \leq \lceil \frac{n}{2} \rceil$.*

Proof. Let $k \geq 4$ and let $n \geq 2k + 12$. Set $N := \lfloor \frac{n-2k}{6} \rfloor \geq 2$, so that $|V(H_{k,N})| = 6N + 2k \leq n$. By Proposition III.2.15, we have $\theta_M(H_{k,N}) = N + k - 2 = \lfloor \frac{n-2k}{6} \rfloor + k - 2 \geq \lfloor \frac{n}{6} \rfloor$. If $|V(H_{k,N})| < n$ then it suffices to add $n - |V(H_{k,N})|$ isolated vertices to get a hypergraph on exactly n vertices. ■

Remark. It is probably possible to make the hypergraph linear and still get a value of θ_M that is $\Omega(n)$ (where the multiplicative constant depends on k), which would show that high values for θ_M can be reached with edge intersections of any size. This would require a direct definition of the hypergraph for all k instead of using Proposition I.1.56 to go from $k = 4$ to general k , since this construction does not preserve linearity. Note that the D_u can be made pairwise vertex-disjoint, by giving each D_u its own vertices d_1, d_2, d_3, d_4, d_5 , which should help. For $k = 4$, it would then suffice to make the D_u linear. For $k \geq 5$, the idea would be to replace c_i with $c_{i,1}, \dots, c_{i,k-3}$ and attach some linear $D_{c_{i,j}}$ to each $c_{i,j}$.

III.2.5 Conclusion and prospects

We have shown that deciding the winner of the unbiased Maker-Breaker game on hypergraphs of rank 3 can be done in polynomial time. Since this problem is known to be PSPACE-complete on 6-uniform hypergraphs, the next question is: what about hypergraphs of rank 4 or 5? An indication on the answer may be provided by the results that we have just established on τ_M and θ_M , which give us Table III.1 (the case $k = 2$ is simply given by Theorem I.1.33).

	θ_M	τ_M
$k = 2$	2	2
$k = 3$	3	$\log_2(n) + O(1)$
$k \geq 4$	$\Omega(n)$	$\lceil \frac{n}{2} \rceil$

Table III.1: Maximum value attained by θ_M, τ_M over all k -uniform non-marked Maker wins on n vertices.

For both τ_M and θ_M , we can observe a significant gap in maximum value between the case $k = 3$ and the case $k \geq 4$. This indicates that Maker's winning strategies become more complex from rank 4, as they can be very long and rely on many simultaneous threats. This jump may translate in terms of algorithmic complexity, which reinforces the commonplace intuition within the community that deciding the winner of the unbiased Maker-Breaker game on hypergraphs of rank 4 should be PSPACE-complete.

Our work on the exact maximum values attained by τ_M and θ_M is basically complete. One thing of note is that the gap between these two parameters can be arbitrarily large, as shown by nunchakus: three tokens are sufficient for Maker whatever the number of vertices n of the nunchaku, whereas the number of rounds that she needs to win is logarithmic in n (note that k -uniform forcing paths are an even more extreme example when $k \geq 4$, since k tokens are sufficient but around $\frac{n}{2}$ rounds are needed). When it comes to remaining open questions, it would be interesting to know if θ_M attains the upper bound $\lceil \frac{n}{2} \rceil$ for $k \geq 4$, as the most

token-consuming example we have found so far only reaches $\frac{n}{6}$.

Chapter IV

From positional games to reconfiguration problems[†]

Consider the unbiased Maker-Breaker game on some hypergraph H , in a version where Maker and Breaker respectively have α and β tokens at their disposal. Assume α is finite. We do not require that β is finite. Actually, the case $\beta = \infty$ corresponds to the study from Subsection III.2.4, except that we fix α here instead of trying to find the minimum value of α such that Maker wins.

Let us look at the state of the game after the first α rounds. Maker has now placed all her tokens: let A be the configuration formed by Maker's tokens at this point i.e. the set of all vertices on which Maker has tokens. From now on, Maker will move one of her tokens in each round, trying to complete an edge. Therefore, she wants her tokens to eventually form some configuration $B \in \mathcal{B}$, where \mathcal{B} is the set of all $B \in \mathbb{P}_\alpha(V(H))$ such that $e \subseteq B$ for some $e \in E(H)$. The game thus obeys the following ruleset template:

- We have a set of positions S .
- We have a starting configuration of tokens $A \subseteq S$.
- We have some target configurations of tokens $B \subseteq S$ with $|B| = |A|$.
- Alice wants to go from A to some target B , moving tokens one at a time.
- There is no restriction for the move's origin: Alice can freely choose the token that she moves.
- However, there is a restriction (R) for the move's destination: Alice cannot move that token anywhere she wants, as some positions are prohibited. These forbidden destinations can evolve during the game, for example they may depend on the current configuration of Alice's tokens.

Each possible restriction (R) defines a different game. For example, if there is a second player alternating moves with Alice who is blocking some positions with tokens of his own, then we get the Maker-Breaker game (with tokens, as above): the forbidden destinations are simply the ones that are already occupied by Bob's tokens. What about other restrictions? If the forbidden destinations, instead of being chosen by a second player as the game goes, obey some preestablished rule fixed before the game begins, then we get a one-player game which classifies as a *reconfiguration problem*.

In general, a reconfiguration problem can be seen as a one-player game played on some discrete structure. A notion of *configuration* on said structure is defined in some way, as well as a rule determining what constitutes an authorized *move* from one configuration to another. Given two configurations A and B , the question is asked whether there exists a finite sequence of moves that leads from A to B . A popular example is the *Rubik's Cube* game, where the player wants to go from a mixed up configuration A to the target configuration B where all colors are sorted.

[†] These results are part of the article [GGS21], which has been accepted in *Games of No Chance*, vol. 6.

Reconfiguration problems on graphs, where configurations and moves are defined according to some graph properties, represent an important part of this field. The literature includes problems where the configurations are vertex-colorings [BCv⁺07], dominating sets [SMN16] or cliques [IOO23] for instance.

In this chapter, we study a specific reconfiguration problem on graphs (more specifically: on the square grid), which satisfies the positional-type ruleset template above, where the restriction (R) is the 2-adjacency rule. This means that a token's destination must have at least two neighbors already occupied by other tokens. For consistency with the existing literature, we hereafter use the "coin" terminology rather than "token".

IV.1 Coin-moving puzzles with 2-adjacency restriction on the square grid

We study a one-player game played on a graph $G = (V, E)$ referred to as the *board*. Throughout the game, there will be *coins* sitting on some of the vertices of G (at most one per vertex). The coins are indistinguishable and define a *configuration*, i.e. a finite subset $C \subseteq V$ where we see each element of C as a coin sitting on the corresponding vertex. A legal *move* consists in moving a single coin to a free vertex so that, after the move, that coin has at least two other coins adjacent to it. This is called the *2-adjacency* restriction. Given two configurations A and B , we want to know whether the *puzzle* $A \stackrel{?}{\rightarrow} B$ is *solvable*: starting from A , is it possible to reach B using only legal moves? In the positive case, we would like an explicit winning sequence of moves.

Instances of this game, or rather a variation with tightly packed coins that can only be slid in the plane without collision, appear in the literature as early as the 1950s in [Lan51] and [Lan53]. Figure IV.1 features a couple of classic puzzles on the triangular grid as well as a rarer puzzle on the square grid. Such examples also appear in [Gar75] and [BCG82, pp.863–864] among others, but it is not until 2002 that general puzzles with these rules have been analyzed, by Demaine et al. [DDV02]. Their study serves as foundation for the master's thesis [Gal19] and for the present chapter. The authors give a full characterization for solvable puzzles on the triangular grid (up to a minor omission which is easily settled: see [Gal19]). Furthermore, they address a large family of puzzles on the square grid. Their proofs are constructive and provide polynomial-time solving algorithms. Note that a version of this game with labelled coins also exists, where each coin must end up at its own specific place: we do not address this version here, but some results are known [DDV02][Gal19].

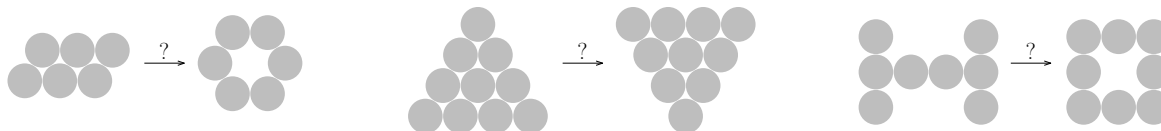


Figure IV.1: The board is the triangular grid in the left and middle puzzles, and the square grid in the right puzzle. The left (resp. middle, resp. right) puzzle is solvable in 2 (resp. 3, resp. 4) moves.

We focus exclusively on the case where G is the square grid. Let us clarify that this refers to the infinite square grid, where each vertex has exactly four neighbors: left, right, top, bottom. This section lists some easy but important notions around this game, most of which are introduced in [DDV02], in which the authors identify that the game subdivides into two main cases as we are about to see. One of these cases is partially investigated in [Gal19], while the other is the

subject of Section IV.2.

IV.1.1 Notations and first observations

We use the (self-)dual grid for graphical representations of the game: each vertex, or *position*, is seen as a square and coins are placed at the center of squares. See Figure IV.2.

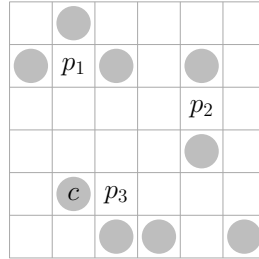


Figure IV.2: A configuration C with 9 coins. From C , a possible move would be $c \mapsto p_1$ or $c \mapsto p_2$ for example. However, c cannot be moved to p_3 , because that position only has one neighboring coin other than c .

Notation IV.1.1. We introduce the following notations:

- Moving a coin c to a valid destination p is denoted by $c \mapsto p$. Note that a coin c is simply an element of V representing its location (since the coins are indistinguishable, a coin c is nothing but an occupied position p). Typically, the notation c is used to designate an occupied position, and the notation p is used for a position that is either unoccupied or not necessarily occupied.
- If there is a single move $c \mapsto p$ from A to B , we write $A \xrightarrow{c \mapsto p} B$ or simply $A \mapsto B$.
- If there exists a sequence of moves from A to B , we write $A \rightarrow B$, otherwise we write $A \not\rightarrow B$. By convention, an empty sequence of moves is allowed so that $A \rightarrow A$.
- If $A \rightarrow B$ and $B \rightarrow A$ then we may write $A \leftrightarrow B$.

Let us start by making two immediate observations:

Proposition IV.1.2. *Let $A \neq B$ be configurations. If $A \rightarrow B$ then there exists a coin in B that has at least two neighboring coins in B .*

Proof. The last moved coin has at least two neighboring coins by the 2-adjacency rule. ■

Proposition IV.1.3. *Let $A \neq B$ be configurations. If there exists a coin $b \in B$ such that $B \setminus \{b\}$ consists of all isolated coins, then $A \rightarrow B$ if and only if $A \mapsto B$.*

Proof. Suppose that $A = A_0 \xrightarrow{c_1 \mapsto p_1} A_1 \xrightarrow{c_2 \mapsto p_2} \dots \xrightarrow{c_T \mapsto p_T} A_T = B$ and that $B \setminus \{b\}$ consists of all isolated coins. In particular all coins in $B \setminus \{b\}$ have at most one neighboring coin in B , so $p_T = b$ by the 2-adjacency rule. Since $A_{T-1} \setminus \{c_T\} = B \setminus \{p_T\}$, this means $A_{T-1} \setminus \{c_T\}$ consists of all isolated coins. In particular all coins in $A_{T-1} \setminus \{c_T\}$ have at most one neighboring coin in A_{T-1} , so $p_{T-1} = c_T$ by the 2-adjacency rule. Continuing so, we get $p_{T-1} = c_T, p_{T-2} = c_{T-1}, \dots, p_1 = c_2$. In conclusion, we have moved the same coin each time, so we could have moved it just once instead and got $A \xrightarrow{c_1 \mapsto p_T} B$. ■

IV.1.2 Picking up and dropping coins

A move can be decomposed into two parts: a coin is picked up, and then dropped back on the board.

Definition IV.1.4. [DDV02] We define the following actions:

- *Pick up* a coin: remove a coin from the board, without any restriction on its position.
- *Drop* a coin: put a previously picked up coin back on the board, with the 2-adjacency restriction.

Picked up coins that have not yet been dropped may be referred to as *coins in hand*.

It is shown in [DDV02] that the game is unchanged if the player is allowed to pick up and drop coins as individual actions:

Proposition IV.1.5. [DDV02] *A puzzle is solvable (by moving coins) if and only if it is solvable by moving, picking up and dropping coins.* ■

Notation IV.1.6. In this variation, the state of the game at any given moment is described by the configuration A on board and the number k of coins in hand: we denote this information by A^{+k} . For example, the notation $A^{+k} \rightarrow B^{+k'}$ means that, from the configuration A with k coins in hand at the start, it is possible (via moving, picking up and dropping coins) to reach the configuration B with $k' = k + |A| - |B|$ coins in hand at the end.

IV.1.3 Span of a configuration

IV.1.3.1 Definition and a key necessary condition

A first natural question is: starting from some configuration A , what positions can we reach? A central observation, which is an immediate consequence of the 2-adjacency restriction, is the following: during the moves, all coins remain inside of the finite set obtained from A by including all vertices that have at least two neighbors in A and iterating this process until no more vertex can be included. We call this the *span* of A (see Figure IV.3 for an example):

Notation IV.1.7. Let C be a configuration, we denote by $\text{Adj}(C) \subseteq V \setminus C$ the set of all positions outside C that have at least two neighbors in C .

Definition IV.1.8. [DDV02] The *span* of a configuration C , denoted by $\text{span}(C)$, is the limit of the non-decreasing sequence of configurations $(C_i)_{i \geq 0}$ defined recursively by $C_0 = C$ and $C_{i+1} = C_i \cup \text{Adj}(C_i)$. In other words, $\text{span}(C)$ is the set of all positions that could be reached from C if we had unlimited coins to add to the board at successive positions satisfying the 2-adjacency rule.

Therefore, any configuration B that we wish to obtain from A must therefore satisfy $B \subseteq \text{span}(A)$, which implies $\text{span}(B) \subseteq \text{span}(A)$. This result is absolutely essential.

Proposition IV.1.9. [DDV02] *The span never increases during moves: if $A \rightarrow B$ then $\text{span}(A) \supseteq \text{span}(B)$.* ■

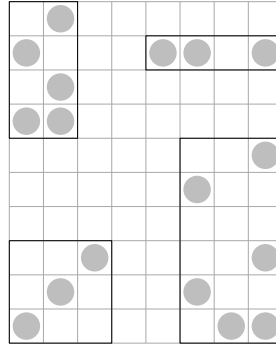


Figure IV.3: The span of a configuration. In this example, there are four components.

IV.1.3.2 Structural properties

Note that the span of any configuration C is finite i.e. the sequence $(C_i)_{i \geq 0}$ from Definition IV.1.8 is eventually constant. For one, the smallest rectangle R enclosing C clearly contains the span, since each square outside R has at most one neighboring square inside R . This is a fundamental difference with the game on the triangular grid for instance, where the span is the entire grid for all nontrivial configurations. More specifically, it is easy to see that the span is always a union of rectangles at distance at least 3 from each other:

Definition IV.1.10. Identifying the square grid as \mathbb{Z}^2 , an $m \times n$ rectangle R is a set of positions of the form $I \times J$ where I and J are intervals of cardinality m and n respectively, so that each row of R contains m positions and each column of R contains n positions. We say R is *even* (resp. *odd*) if its *half-perimeter* $m + n$ is even (resp. odd).

Notation IV.1.11. We consider the usual distance i.e. the geodesic distance in the square grid, denoted by dist .

Proposition IV.1.12. [DDV02] *The span of any configuration C is a union of rectangles, such that the distance between any two of these rectangles is at least 3 (these rectangles are called the components of $\text{span}(C)$).* ■

We add the following structural property of configurations, which will be useful later:

Proposition IV.1.13. *Let C be a configuration and let R be a component of $\text{span}(C)$. Then C contains at least one coin in each of the following: the top row of R , the bottom row of R , the leftmost column of R , the rightmost column of R , any union of two consecutive rows in R , any union of two consecutive columns in R .*

Proof. Using the symmetries, we only address the case of the top row and the union of two consecutive rows. Let $C = C_0, C_1, \dots, C_s = \text{span}(C)$ as in Definition IV.1.8 (indeed, the sequence $(C_i)_{i \geq 0}$ is eventually constant by finiteness of the span).

- Suppose for a contradiction that the top row R' contains no coin in C i.e. $C \cap R' = \emptyset$. Let $0 \leq i \leq s$ be smallest such that $C_i \cap R' \neq \emptyset$: we have $i \geq 1$ and $C_{i-1} \cap R' = \emptyset$. Obviously, any position in R' has at most one neighbor in $R \setminus R'$, so $\text{Adj}(C_{i-1}) \cap R' = \emptyset$. Since $C_i = C_{i-1} \cup \text{Adj}(C_{i-1})$, we get $C_i \cap R' = \emptyset$ which is a contradiction.
- The previous proof still works if we replace R' with any union of two consecutive rows, since the key argument that any position in R' has at most one neighbor in $R \setminus R'$ still holds. ■

IV.1.4 Minimal/minimum configurations

Definition IV.1.14. [DDV02] A configuration C is said to be *minimal* if the removal of any coin in C decreases the span.

Picking up a coin from a minimal configuration decreases the span, and dropping it back afterwards cannot increase the span because of the 2-adjacency restriction. Therefore:

Proposition IV.1.15. [DDV02] *Any move played from a minimal configuration decreases the span.* ■

Definition IV.1.16. A configuration C is said to be *minimum* if there is no configuration C' with same span as C such that $|C'| < |C|$ (in particular, C is then minimal).

As noticed in [DDV02], the cardinality of minimum configurations (as well as a lot of information on their structure) is well known thanks to the following classic problem from folklore. In a rectangular parcel R consisting of small squares arranged in a grid, some squares are initially invaded by weeds. Time passes, and at each time step, any square that is adjacent to at least two weeds-covered squares gets invaded in turn. How many squares need to be covered initially for the entire parcel to be invaded in the end? Since the rule for the propagation of the weeds is exactly the same as for the construction of the span, the answer coincides with the cardinality of a minimum configuration with span R . This problem was first published in [Kva86] for a 10×10 parcel. An elegant solution is obtained via an invariant which is the perimeter of the invaded area:

Proposition IV.1.17. *Let R be an $m \times n$ rectangle and let M be a minimum configuration with span R . We have $|M| = \lceil \frac{m+n}{2} \rceil$. Moreover:*

- *If R is even, then all coins in M are isolated.*
- *If R is odd, then all coins in M are isolated except possibly for a single pair of adjacent coins.*

Proof. Define $\text{Perim}(C)$ as the perimeter of the union of unit squares on board corresponding to all occupied positions in a configuration C . For any configuration C and any $p \notin C$, we have $\text{Perim}(C \cup \{p\}) = \text{Perim}(C) + 4 - 2k$ where k is the number of coins in C that are a neighbor of p : indeed, in terms of the perimeter, each side of p removes 1 if it neighbors a coin in C or adds 1 if it does not. Therefore, any configuration C satisfies $\text{Perim}(C) = 4|C| - 2l$ where l is the number of pairs of adjacent coins in C . Moreover, the perimeter cannot increase during the construction of the span, since each added coin has at least two neighbors amongst the coins that are already present: this implies that any configuration C with span R satisfies $2(m+n) = \text{Perim}(\text{span}(C)) \leq \text{Perim}(C) \leq 4|C|$ hence $|C| \geq \lceil \frac{m+n}{2} \rceil$. Conversely, it is easy to find configurations with span R that have exactly $\lceil \frac{m+n}{2} \rceil$ coins (think of a diagonal of coins if $m = n$, or more generally an ‘L’ shape as in Definition IV.1.20), so the cardinality of minimum configurations with span R is exactly $\lceil \frac{m+n}{2} \rceil$. Finally, let M be a minimum configuration with span R and let l be the number of pairs of adjacent coins in M , we have $2(m+n) \leq \text{Perim}(M) = 4\lceil \frac{m+n}{2} \rceil - 2l$, so in conclusion:

- *If R is even, then $2(m+n) \leq 4\frac{m+n}{2} - 2l$ hence $l = 0$ i.e. all coins in M are isolated.*
- *If R is odd, then $2(m+n) \leq 4\frac{m+n+1}{2} - 2l$ hence $l \in \{0, 1\}$ i.e. all coins in M are isolated except possibly for a single pair of adjacent coins.* ■

IV.1.5 ‘L’s and canonical configurations

In order not to clash with the terminology from the previous chapters, the notion of "chain" defined in [DDV02] will be renamed here as a "string":

Definition IV.1.18. [DDV02, as *chain*] A *string* between some coins c and c' is the configuration denoted by $[c_1, \dots, c_N]$ which is formed by a sequence of coins ($c = c_1, c_2, \dots, c_N = c'$) such that $\text{dist}(c_i, c_{i+1}) \in \{1, 2\}$ for all $1 \leq i \leq N - 1$.

Proposition IV.1.19. [DDV02] *The span of a string coincides with its smallest enclosing rectangle.* ■

In [DDV02], the authors define a reference minimum configuration for a given span, where the coins in each component form an L shape:

Definition IV.1.20. [DDV02] An ‘L’ of size $m \times n$ is a minimum string L between two opposite corners of an $m \times n$ rectangle R and hugging two consecutive sides of R . We say L is *even* (resp. *odd*) if R is even (resp. odd) i.e. if $m + n$ is even (resp. odd). See Figure IV.4.

Remark. In accordance with Proposition IV.1.17, an $m \times n$ ‘L’ has cardinality $\lceil \frac{m+n}{2} \rceil$, and consecutive coins in an even ‘L’ are at distance exactly 2 whereas consecutive coins in an odd ‘L’ are at distance exactly 2 except for a single pair of adjacent coins. An even ‘L’ is entirely defined by its span and orientation, whereas for an odd ‘L’ we also need the localization of the two adjacent coins. By Proposition IV.1.19, the span of an ‘L’ is its smallest enclosing rectangle i.e. R in Definition IV.1.20.

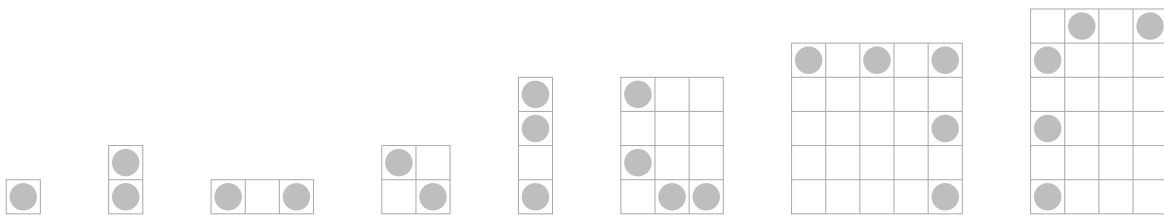


Figure IV.4: Eight examples of ‘L’s with various sizes and orientations.

Definition IV.1.21. [DDV02] Let R be an $m \times n$ rectangle: the *canonical ‘L’ with span R* is the ‘L’ with span R that is oriented like the letter L, with the additional property if R is odd that the two adjacent coins are in the top-left corner (if n is even) or bottom-right corner (if m is even). Let C be a configuration with span $\bigcup_{i=1}^s R_i$ where R_1, \dots, R_s are rectangles at distance at least 3 from each other, as per Proposition IV.1.12: the *canonical configuration associated to C* is the configuration denoted by L_C with same span as C such that, for all $1 \leq i \leq s$, $L_C \cap R_i$ is the canonical ‘L’ with span R_i . See Figure IV.5.

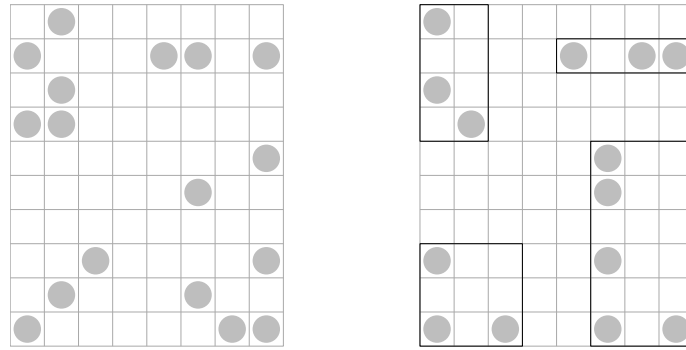


Figure IV.5: Left: a configuration C . Right: the associated canonical configuration L_C (the top-left and bottom-left ‘L’s are even, the top-right and bottom-right ‘L’s are odd).

IV.1.6 Extra coins and redundant coins

IV.1.6.1 Definitions and first observations

Proposition IV.1.9 stated that, for a puzzle $A \stackrel{?}{\rightarrow} B$ to be solvable, it is necessary that $\text{span}(B) \subseteq \text{span}(A)$. In the study that is made in [DDV02], a key information is the number of "extra coins" i.e. coins that can be removed from A while maintaining a span containing that of B . The more extra coins at our disposal, the more flexibility with respect to the span constraint, hence, the easier a puzzle.

Definition IV.1.22. Let A be a configuration and $k \in \mathbb{N}$.

- A set of extra coins in A is a subset $A' \subseteq A$ such that $\text{span}(A \setminus A') = \text{span}(A)$. We say A has k extra coins if it contains a set of k extra coins. For example, A has 1 extra coin if and only if A is not minimal.
- Let B be a configuration. A set of extra coins in A relatively to B is a subset $A' \subseteq A$ such that $\text{span}(A \setminus A') \supseteq \text{span}(B)$. We say A has k extra coins relatively to B if it contains a set of k extra coins relatively to B . If A and B have same span, then this definition coincides with the previous one.

Remark. Notice that the phrase " A has k extra coins" is understood in the "at least" sense as opposed to "exactly".

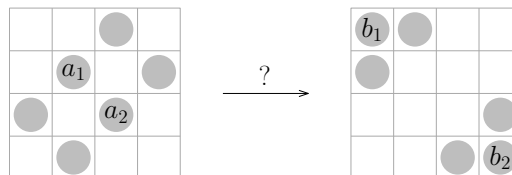


Figure IV.6: An example of a puzzle with 2 extra coins a_1, a_2 (Definition IV.1.22) and 2 redundant coins b_1, b_2 (Definition IV.1.24).

Example. In the puzzle $A \stackrel{?}{\rightarrow} B$ from Figure IV.6, $\{a_1, a_2\}$ is a set of extra coins in A (in particular, it is also a set of extra coins in A relatively to B).

Proposition IV.1.23. Let A and B be distinct configurations. If $A \rightarrow B$ then A has 1 extra coin relatively to B .

Proof. Suppose A does not have 1 extra coin relatively to B , and consider the first move $c \mapsto p$ made from A . This move can be decomposed as follows: first we remove c from the board, then we put it back at p . After removing c from the board, the span is $\text{span}(A \setminus \{c\})$. When we put the coin back, the span stays the same because of the 2-adjacency rule. Therefore the span after the first move is $\text{span}(A \setminus \{c\})$, which does not contain $\text{span}(B)$ because $\{c\}$ is not a set of extra coins relatively to B . By Proposition IV.1.9, the span cannot increase during the moves, hence $A \not\rightarrow B$. ■

We now introduce the notion of redundant coins, which is also present in [DDV02] (though unnamed). They are, to the target configuration B , the relevant analogue of what extra coins are to the starting configuration A .

Definition IV.1.24. Let B be a configuration. A set of redundant coins in B is a subset $B' \subseteq B$ of the form $B' = \{b_1, \dots, b_k\}$ where, for all $1 \leq i \leq k$, b_i has at least two neighbors in $B \setminus \{b_1, \dots, b_{i-1}\}$. We say B has k redundant coins if it contains a set of k redundant coins.

Example. In the puzzle $A \stackrel{?}{\rightarrow} B$ from Figure IV.6, $\{b_1, b_2\}$ is a set of redundant coins in B .

Remark. Note that redundant coins are extra coins, but the converse is not true in general (think of a_1 and a_2 in Figure IV.6).

Proposition IV.1.25. Let A and B be distinct configurations. If $A \rightarrow B$ then B has one redundant coin.

Proof. This is exactly Proposition IV.1.2. ■

Proposition IV.1.26. Let B be a configuration and let $B' = \{b_1, \dots, b_k\}$ be a set of redundant coins in B , ordered as in the definition. Then $B \leftrightarrow (B \setminus B')^{+k}$. In particular, for any configuration A , we have $A \rightarrow B$ if and only if $A \rightarrow (B \setminus B')^{+k}$.

Proof. We perform $B \rightarrow (B \setminus B')^{+k}$ by picking up the coins in B' , and $(B \setminus B')^{+k} \rightarrow B$ by dropping the coins in hand at b_k, b_{k-1}, \dots, b_1 successively (which respects the 2-adjacency rule since b_i has at least two neighbors in $B \setminus \{b_1, \dots, b_{i-1}\}$). ■

Therefore, more redundant coins can only make a puzzle easier. One can think of extra coins in A (or extra coins in A relatively to B) as the first coins that we move, and of redundant coins in B as the last coins that we place.

IV.1.6.2 A central factor for the game

Denoting by e the number of extra coins in the starting configuration relatively to the target configuration, the verdict made in [DDV02] is roughly the following:

- All puzzles with $e = 0$ are unsolvable (this is Proposition IV.1.23).
- Puzzles with $e = 1$ are difficult to assess in general.
- Almost all puzzles with $e \geq 2$ are solvable.

It actually turns out that such a statement for $e = 2$ is overly optimistic. We now address this case in Section IV.2. Before that, let us briefly mention the case $e = 1$. Its study has been initiated in [Gal19], with the additional assumption that A and B have same span and are both *minimum+1* i.e. consist of a minimum configuration plus one extra coin. It is shown that the game then reduces to a *pushing game* where, instead of the 2-adjacency rule, a coin can be slid onto a neighboring square under certain conditions. This pushing game is played directly on the minimum configurations obtained by removing the extra coin in A and B . One result on

the pushing game is that the configurations that can be reached from a minimum string M are exactly all minimum strings between the same vertices as M .

IV.2 The case of two extra coins

IV.2.1 Presentation of the problem and state of the art

The best part of [DDV02] is dedicated to puzzles $A \stackrel{?}{\rightarrow} B$ where A has 2 extra coins relatively to B . The authors design an algorithm that goes from A to B by routing through the canonical configuration: Subsection IV.2.2 will provide details on this procedure, which we are going to use ourselves. Their main result is the following:

Theorem IV.2.1. [DDV02] *Let A and B be configurations such that $|A| = |B|$, and suppose that:*

- (i) $\text{span}(A) = \text{span}(B)$.
- (ii) A has 2 extra coins relatively to B .
- (iii) B has 2 redundant coins.

Then $A \rightarrow B$. ■

Example. The puzzle from Figure IV.6 is solvable according to Theorem IV.2.1, with $\{a_1, a_2\}$ as the set of extra coins in A relatively to B and $\{b_1, b_2\}$ as the set of redundant coins in B .

In fact, a stronger version of this theorem is stated in [DDV02], where condition (i) is omitted altogether. We show in Subsection IV.2.3 that the authors' claim is incorrect, but that an in-between version does hold, where condition (i) is relaxed so as to require that each component of $\text{span}(A)$ contains at most one component of $\text{span}(B)$. Subsection IV.2.4 then investigates what happens when this requirement is not satisfied. We show that, if condition (i) is omitted in Theorem IV.2.1, then the theorem does not hold for any constant number of extra coins or redundant coins that we ask for in conditions (ii) and (iii), let alone just two. On the positive side, we obtain a new sufficient condition for a puzzle to be solvable, using a different algorithm.

IV.2.2 Canonicalization process

Let us start by explaining the algorithm behind Theorem IV.2.1. The key to the proof in [DDV02] is that two coins in hand are enough to turn any 'L' into another 'L' of same span:

Proposition IV.2.2. [DDV02] *Let L_1 and L_2 be two 'L's with the same $m \times n$ span. Then $L_1^{+2} \leftrightarrow L_2^{+2}$ in $O(mn)$ moves.* ■

In particular, two coins in hand are enough to *flip* any 'L', which means turning it into the mirrored 'L' hugging the other two sides of the span. Let us detail the method used for this specific transformation, as we will call on it later. Figure IV.8 explains how to flip an even 'L' with two coins in hand, using the subroutines from Figure IV.7. Figure IV.11 explains how to flip an even 'L' with one coin in hand, using the subroutines from Figure IV.10 as well as the *leapfrog* technique described in Figure IV.9: this means relocating the unique pair of adjacent coins inside an odd 'L'. In all figures throughout this chapter, an encircled coin represents a coin

that has just been dropped, while a crossed out coin represents a coin that we pick up.

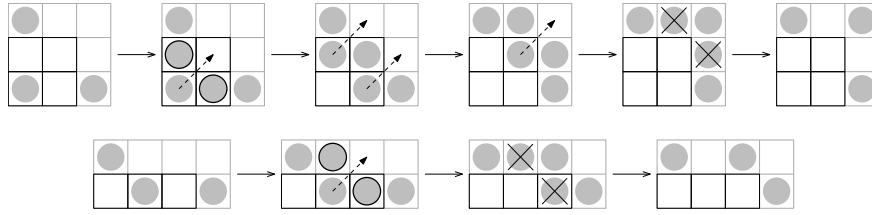


Figure IV.7: Subroutines used to flip an even ‘L’. The bottom subroutine is only used if both sides are even.

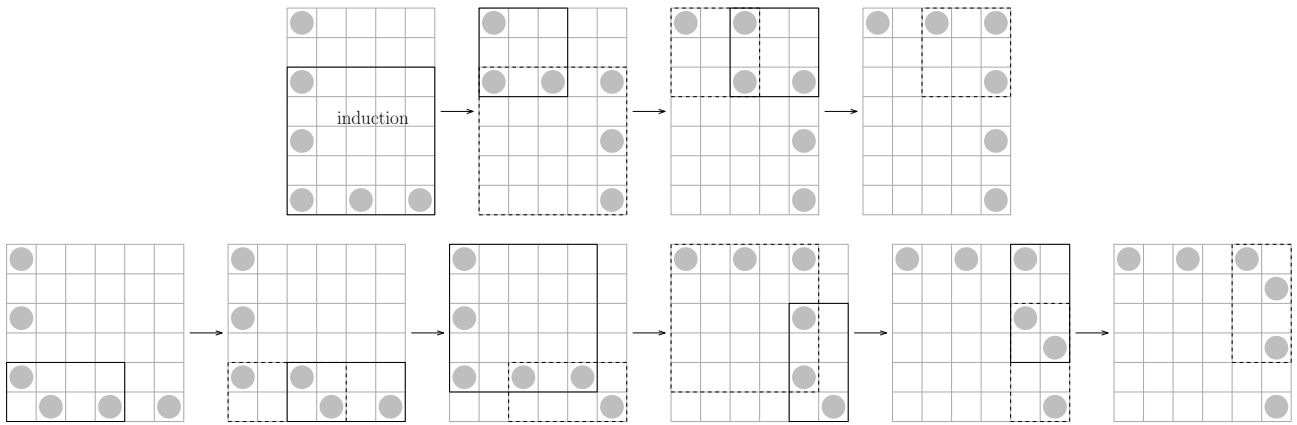


Figure IV.8: Flipping an even ‘L’, with odd sides (top) or even sides (bottom).

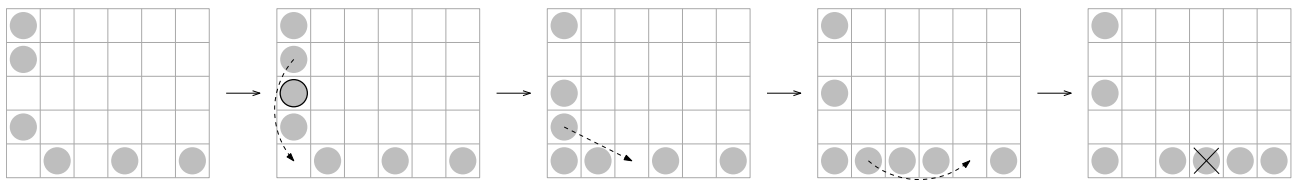


Figure IV.9: A leapfrog. Intermediary states cover all possible locations of the pair of adjacent coins.

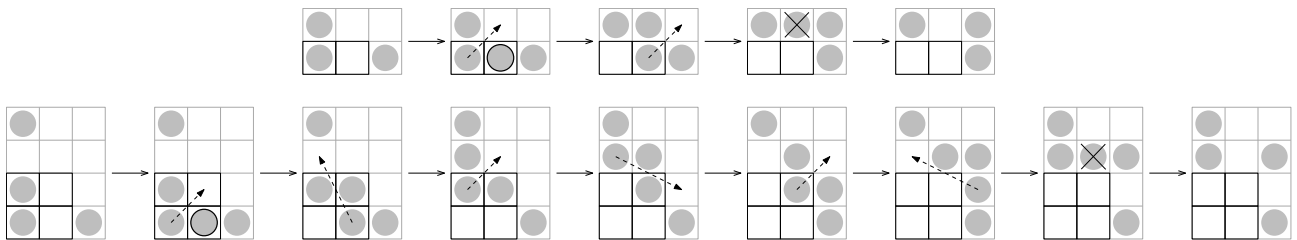


Figure IV.10: Subroutines used to flip an odd ‘L’.

With two coins in hand, using successive transformations of ‘L’s which include reorientations as in Proposition IV.2.2, the authors design a "canonicalization process" that turns any configuration into its associated canonical configuration in a reversible manner:

Lemma IV.2.3. [DDV02] *For any configuration C , we have $C^{+2} \leftrightarrow L_C^{+2+|C|-|L_C|}$ in $O(N^3)$ moves where $N := |C|$.* ■

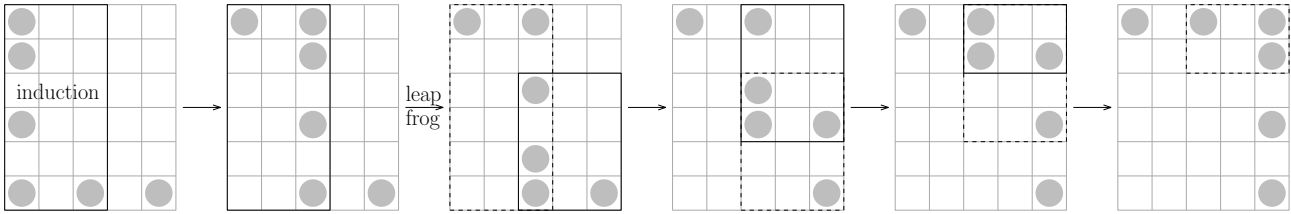


Figure IV.11: Flipping an odd ‘L’.

From there, a method to solve a puzzle $A \stackrel{?}{\rightarrow} B$ would roughly be to: pick up two coins in A ; canonicalize; reverse into B minus two coins; drop two coins to finish B . To do this however:

- We need two coins in A that we can pick up without breaking the inclusion of spans at the start. In other words, we need 2 extra coins in A relatively to B .
- We need two appropriate spots in B to drop our two coins in hand at the end. In other words, we need 2 redundant coins in B .

Moreover, unless the spans are equal once the first two coins $\{a_1, a_2\}$ have been picked up, we need a way to go from $L_{A \setminus \{a_1, a_2\}}$ to L_B . This reasoning can be summed up as follows:

Corollary IV.2.4. *Let A and B be configurations such that $|A| = |B| =: N$, and suppose that:*

- (i) *A has 2 extra coins relatively to B .*
- (ii) *B has 2 redundant coins.*

Let $A_0 := A \setminus \{a_1, a_2\}$ where $\{a_1, a_2\}$ is a set of extra coins in A relatively to B .

If $L_{A_0}^{+|A|-|L_{A_0}|} \rightarrow L_B^{+|B|-|L_B|}$, then $A \rightarrow B$ in $O(N^3)$ moves.

Proof. Let $B_0 := B \setminus \{b_1, b_2\}$ where $\{b_1, b_2\}$ is a set of redundant coins in B . Note that $L_B = L_{B_0}$ since $\text{span}(B) = \text{span}(B_0)$. We go from A to B in five steps:

- We get $A \rightarrow A_0^{+2}$ by picking up a_1 and a_2 .
- Lemma IV.2.3 ensures that $A_0^{+2} \rightarrow L_{A_0}^{+2+|A_0|-|L_{A_0}|}$.
- We have $L_{A_0}^{+2+|A_0|-|L_{A_0}|} = L_{A_0}^{+|A|-|L_{A_0}|} \rightarrow L_B^{+|B|-|L_B|} = L_{B_0}^{+2+|B_0|-|L_{B_0}|}$ by assumption.
- Lemma IV.2.3 ensures that $L_{B_0}^{+2+|B_0|-|L_{B_0}|} \rightarrow B_0^{+2}$.
- Finally, Proposition IV.1.26 yields $B_0^{+2} \rightarrow B$. ■

IV.2.3 A slight improvement on Theorem IV.2.1

The condition that $L_{A_0}^{+|A|-|L_{A_0}|} \rightarrow L_B^{+|B|-|L_B|}$ in Corollary IV.2.4 is what the authors have missed in [DDV02]. If $\text{span}(A) = \text{span}(B)$, then $L_A = L_{A_0} = L_B$, so this condition is trivially satisfied hence why Theorem IV.2.1 is correct. However, for the stronger version that they claim, the authors reduce to the case where $\text{span}(A) = \text{span}(B)$ by picking up all coins in $A \setminus \text{span}(B)$, which is incorrect since this may cause the span to become strictly smaller than that of B . It turns out that, without the added condition (i) in Theorem IV.2.1, some puzzles are solvable and some are not, as illustrated in Figure IV.12. The puzzle on the left is solvable in 12 moves. The puzzle on the right is unsolvable (as we will later prove) and therefore is a counterexample to the version of the theorem in [DDV02]. It is easy to check that no smaller counterexample exists, be it in terms of number of coins or half-perimeter of the starting span. A generalized family of counterexamples will be exhibited in Subsection IV.2.4: in all of them, the problem comes from the fact that $\text{span}(B)$ is split into two far apart components.

So, how can we improve on Theorem IV.2.1 using the same canonicalization process i.e. using

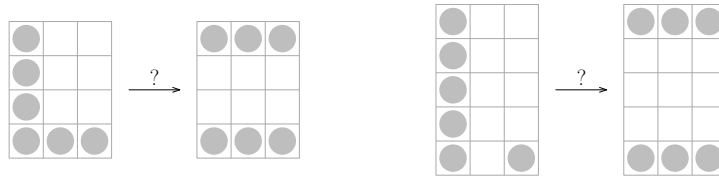


Figure IV.12: Two puzzles satisfying conditions (ii) and (iii) of Theorem IV.2.1 but not (i), since $\text{span}(A) \supsetneq \text{span}(B)$. The left one is solvable but the right one is not.

Corollary IV.2.4? We do not actually need the spans to be equal: it is sufficient that, after picking up two coins, each component of the span contains at most one component of the target span. We can then simply: pick up two adequate coins, reach the canonical configuration, shrink some of the ‘L’s if needed to get the same span as B , and then reverse into B . Indeed, while splitting an ‘L’ into two separate components can be difficult (as we have just seen), shrinking an ‘L’ with two coins in hand is not a problem:

Lemma IV.2.5. [Gal19] *If L_1 and L_2 are canonical ‘L’s with $\text{span}(L_1) \supseteq \text{span}(L_2)$, then $L_1^{+2} \rightarrow L_2^{+2+|L_1|-|L_2|}$ in $O(mn)$ moves, where $m \times n$ is the size of $\text{span}(L_1)$.*

Proof. First of all, we trim L_1 to the right if needed, as follows (see Figure IV.13):

1. If L_1 is odd, we use a leapfrog to put the pair of adjacent coins to the far right.
2. We make sure there is a coin c at the rightmost position that we want to keep, by dropping one there if needed.
3. We finish by simply picking up all coins that are further right than c .

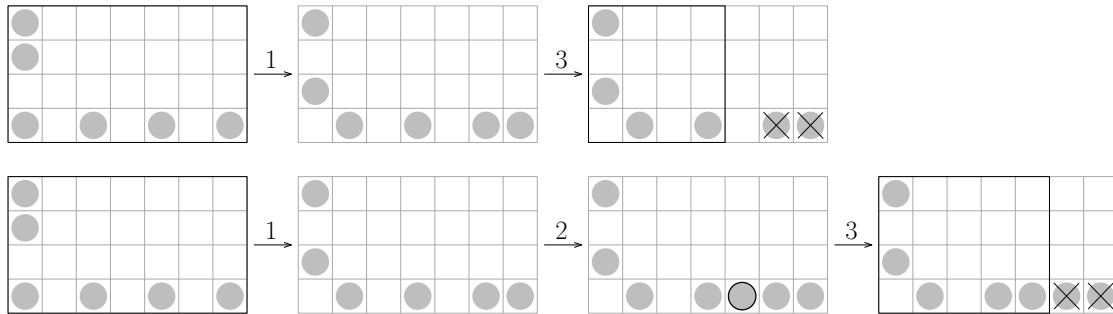


Figure IV.13: Trimming a 7×4 ‘L’ to its right, making it 4×4 (top) or 5×4 (bottom). The numbers above the arrows refer to the three steps.

We then trim our ‘L’ at the top, in analogous fashion. We now flip it, so it is now ready to be trimmed to the left and at the bottom. Once this is done, we flip it back and use a leapfrog if needed to make it canonical. ■

All in all, Theorem IV.2.1 can be improved as follows:

Theorem IV.2.6. [Gal19] *Let A and B be configurations such that $|A| = |B| =: N$, and suppose that:*

- (i) *A has 2 extra coins relatively to B , and more precisely: there exist $a_1 \neq a_2$ in A such that $\text{span}(A \setminus \{a_1, a_2\}) \supseteq \text{span}(B)$ and each component of $\text{span}(A \setminus \{a_1, a_2\})$ contains at most one component of $\text{span}(B)$.*
- (ii) *B has 2 redundant coins.*

Then $A \rightarrow B$ in $O(N^3)$ moves.

Proof. Let $A_0 := A \setminus \{a_1, a_2\}$. By Corollary IV.2.4, it suffices to show that $L_{A_0}^{+|A|-|L_{A_0}|} \rightarrow L_B^{+|B|-|L_B|}$. Since each component of $\text{span}(A_0)$ contains at most one component of $\text{span}(B)$, we can use Lemma IV.2.5 to shrink each ‘L’ in L_{A_0} to the size of the corresponding ‘L’ in L_{B_0} . This is always possible, because we start off with $|A| - |L_{A_0}| \geq |A| - |A_0| = 2$ coins in hand and this number cannot decrease each time we shrink an ‘L’. ■

Note that puzzles satisfying condition (i) of Theorem IV.2.6 but not condition (ii) (while still satisfying the fact that B has 1 redundant coin, otherwise we would be in a trivially unsolvable case by Proposition IV.1.25) also may or may not be solvable as shown in Figure IV.14.



Figure IV.14: Two puzzles satisfying condition (i) of Theorem IV.2.6 but not (ii). The left puzzle is solvable in 4 moves, while the right puzzle is unsolvable.

IV.2.4 The general case with two extra coins

What if some component of $\text{span}(A)$ contains two or more components of $\text{span}(B)$, so that Theorem IV.2.6 does not apply? A first natural guess would be that we then need more extra coins and/or more redundant coins than just two. Nevertheless, we now exhibit a family of unsolvable puzzles which proves that, even with the inclusion of spans, no constant number of extra coins in A (relatively to B or not) or redundant coins in B can guarantee that a puzzle is solvable in general. Next, we present a new sufficient condition for a puzzle to be solvable, which shows in particular that the aforementioned family consists of just about worst-case puzzles.

IV.2.4.1 Worst-case puzzles

Puzzles like those from Figure IV.12 require to split the span, at some point during the moves, in a way that we now prove impossible without a certain total number of coins relative to the size of the rectangles involved.

Definition IV.2.7. Let R_1 and R_2 be rectangles at distance at least 3 from each other, and let A be a configuration such that R_1 and R_2 are included in the same component of $\text{span}(A)$. An (R_1, R_2) -split of A is a sequence of moves $A = A_0 \mapsto A_1 \mapsto \dots \mapsto A_T$ ($T \geq 1$ necessarily) such that R_1 and R_2 are included in two separate components of $\text{span}(A_T)$.

Proposition IV.2.8. Let R_1 and R_2 be rectangles of size $m_1 \times n_1$ and $m_2 \times n_2$ respectively, whose projections on the x axis intersect, and whose projections on the y axis do not intersect with a gap of $h \geq 2$ rows separating them. Let A be a configuration such that R_1 and R_2 are included in the same component of $\text{span}(A)$. If there exists an (R_1, R_2) -split of A , then $|A| \geq \frac{m_1+n_1+m_2+n_2+h-1}{2}$.

Proof. Let $A = A_0 \mapsto A_1 \mapsto \dots \mapsto A_T$ be an (R_1, R_2) -split of A with minimum number of moves, so that R_1 and R_2 are included in the same component R of $\text{span}(A_{T-1})$ but in two separate components R'_1 and R'_2 of $\text{span}(A_T)$ (see Figure IV.15). If R'_1 is of size $m'_1 \times n'_1$ and R'_2 is of size $m'_2 \times n'_2$ with a gap of h' rows separating them, then we have $m'_1 \geq m_1$, $m'_2 \geq m_2$ and $n'_1 + n'_2 + h' \geq n_1 + n_2 + h$ hence $\frac{m'_1+n'_1+m'_2+n'_2+h'-1}{2} \geq \frac{m_1+n_1+m_2+n_2+h-1}{2}$. Therefore, the worst

case for what we want to prove is if $R_1 = R'_1$ and $R_2 = R'_2$, which is what we assume from now on.

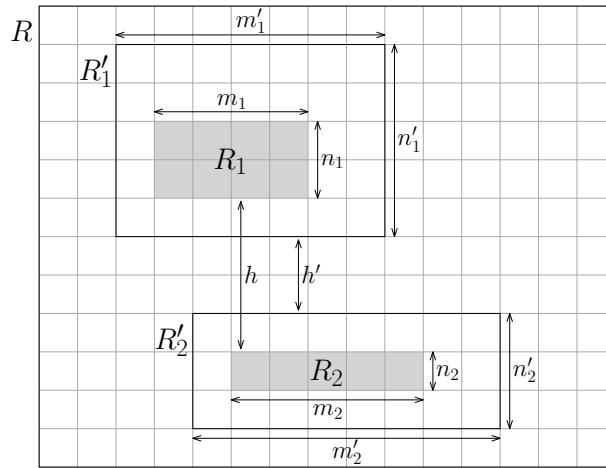


Figure IV.15: Illustration of R , R_1 , R_2 , R'_1 , R'_2 .

We use A_{T-1} to count the coins and carry out the proof. We have:

$$|A| = |A_{T-1}| = |A_{T-1} \cap R_1| + |A_{T-1} \cap R_2| + |A_{T-1} \setminus (R_1 \cup R_2)|. \quad (\text{IV.1})$$

Moreover:

- The only coin in $A_T \cap R_1$ that might not be in $A_{T-1} \cap R_1$ is the coin that has been moved to go from A_{T-1} to A_T , however that coin does not contribute to the span since it has at least two coins adjacent to it in $A_T \cap R_1$. Since $\text{span}(A_T \cap R_1) = R_1$, we thus get $\text{span}(A_{T-1} \cap R_1) = R_1$. In particular, Proposition IV.1.17 yields:

$$|A_{T-1} \cap R_1| \geq \left\lceil \frac{m_1 + n_1}{2} \right\rceil. \quad (\text{IV.2})$$

- Similarly, $\text{span}(A_{T-1} \cap R_2) = R_2$ and:

$$|A_{T-1} \cap R_2| \geq \left\lceil \frac{m_2 + n_2}{2} \right\rceil. \quad (\text{IV.3})$$

- By Proposition IV.1.13, there cannot be two consecutive rows of R without a coin in A_{T-1} . Since there is a gap of h rows between R_1 and R_2 , this yields:

$$|A_{T-1} \setminus (R_1 \cup R_2)| \geq \left\lfloor \frac{h}{2} \right\rfloor. \quad (\text{IV.4})$$

Combining (IV.1) with inequalities (IV.2), (IV.3) and (IV.4), we get:

$$|A| \geq \left\lceil \frac{m_1 + n_1}{2} \right\rceil + \left\lceil \frac{m_2 + n_2}{2} \right\rceil + \left\lfloor \frac{h}{2} \right\rfloor \geq \frac{m_1 + n_1 + m_2 + n_2 + h - 1}{2}. \quad \blacksquare$$

Corollary IV.2.9. *Let $n \geq 9$. We define configurations A_n and B_n as in Figure IV.16:*

- A_n consists of an $n \times n$ ‘L’ with $\left\lfloor \frac{n-2}{2} \right\rfloor$ coins added to the bottom row.
- B_n has the same smallest enclosing rectangle as A_n and contains an $n \times 1$ ‘L’ in both the top and bottom rows with $\left\lfloor \frac{n-5}{2} \right\rfloor$ coins added to the bottom row.

Then $A_n \not\rightarrow B_n$ even though: A_n has $\left\lfloor \frac{n-2}{2} \right\rfloor$ extra coins, B_n has $\left\lfloor \frac{n-5}{2} \right\rfloor$ redundant coins, and $\text{span}(A_n) \supseteq \text{span}(B_n)$.

Proof. Let R_1 (resp. R_2) be the top row (resp. the bottom row) of $\text{span}(A_n)$: R_1 and R_2 are of size $n \times 1$ and separated by a gap of $n - 2$ rows. Solving this puzzle would mean performing an (R_1, R_2) -split of A_n , which is impossible by Proposition IV.2.8 because:

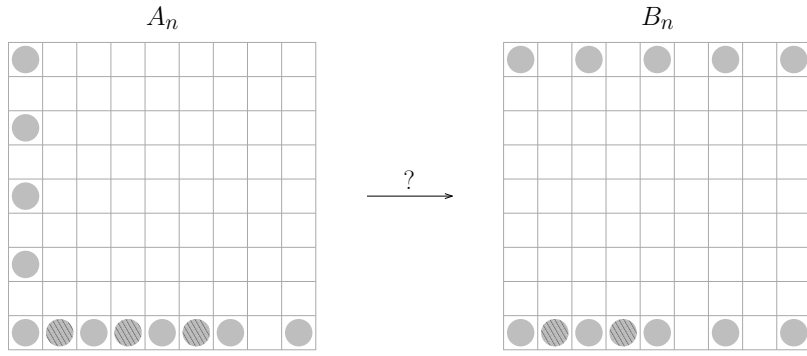


Figure IV.16: Definition of the puzzle $A_n \xrightarrow{?} B_n$ (here $n = 9$). The shaded coins represent the extra/redundant coins.

- If n is odd then $n + \frac{n-3}{2} = |A_n| = |B_n| = 2\frac{n+1}{2} + \frac{n-5}{2} = \frac{3n-3}{2} < \frac{n+1+n+1+(n-2)-1}{2}$.
- If n is even then $n + \frac{n-2}{2} = |A_n| = |B_n| = 2\frac{n+2}{2} + \frac{n-6}{2} = \frac{3n-2}{2} < \frac{n+1+n+1+(n-2)-1}{2}$. ■

Corollary IV.2.10. *For any $k \in \mathbb{N}$, there exist configurations A and B with $|A| = |B|$ such that:*

- $\text{span}(A) \supseteq \text{span}(B)$.
- A has k extra coins.
- B has k redundant coins.
- $A \not\leftrightarrow B$. ■

Proposition IV.2.8 can actually be improved: we now give a refined bound that even applies to the puzzle on the right of Figure IV.12 which, as previously mentioned, is the smallest counterexample to the version of Theorem IV.2.1 in [DDV02].

Proposition IV.2.11. *Let R_1, R_2 and A be as in Proposition IV.2.8. If there exists an (R_1, R_2) -split of A but none in two moves or less, then $|A| \geq \frac{m_1+n_1+m_2+n_2+h+2}{2}$.*

Proof. Let us pick up where the proof of Proposition IV.2.8 ended. Since there exists no (R_1, R_2) -split of A in two moves or less, we have $T \geq 3$. We use the fact that A_{T-1} then has the following properties:

- (i) A_{T-1} contains a coin that is adjacent to at least two other coins.
- (ii) For all $c \in A_{T-1}$, $A_{T-1} \setminus \{c\}$ does not consist of all isolated coins.

Property (i) comes from Proposition IV.1.2 since $T - 1 \geq 1$, and property (ii) comes from Proposition IV.1.3 since $T - 1 \geq 2$ (indeed $A \not\leftrightarrow A_{T-1}$ because our sequence of moves has been chosen shortest). We distinguish between four cases:

- 1) Case 1: R_1 and R_2 are both odd; h is even.

This is the easiest case:

$$\begin{aligned}
 |A| &\geq \left\lceil \frac{m_1 + n_1}{2} \right\rceil + \left\lceil \frac{m_2 + n_2}{2} \right\rceil + \left\lfloor \frac{h}{2} \right\rfloor = \frac{m_1 + n_1 + 1}{2} + \frac{m_2 + n_2 + 1}{2} + \frac{h}{2} \\
 &= \frac{m_1 + n_1 + m_2 + n_2 + h + 2}{2}.
 \end{aligned}$$

- 2) Case 2: At least one of R_1 or R_2 is even (say R_2 is even); h is even.

We just have to find one coin more than what inequalities (IV.2), (IV.3) and (IV.4) give

us combined, because if we do then we can conclude that:

$$\begin{aligned} |A| &\geq \left\lfloor \frac{m_1 + n_1}{2} \right\rfloor + \left\lceil \frac{m_2 + n_2}{2} \right\rceil + \left\lfloor \frac{h}{2} \right\rfloor + 1 \\ &\geq \frac{m_1 + n_1}{2} + \frac{m_2 + n_2}{2} + \frac{h}{2} + 1 = \frac{m_1 + n_1 + m_2 + n_2 + h + 2}{2}. \end{aligned}$$

Therefore, suppose for a contradiction that inequalities (IV.2), (IV.3) and (IV.4) are all tight. For (IV.4), this means that $A_{T-1} \setminus (R_1 \cup R_2)$ consists exactly of one coin every two rows in the gap between R_1 and R_2 . For (IV.2) and (IV.3), this means $A_{T-1} \cap R_1$ and $A_{T-1} \cap R_2$ are both minimum: by Proposition IV.1.17, all coins in $A_{T-1} \cap R_2$ are isolated and all coins in $A_{T-1} \cap R_1$ are isolated except possibly for a single pair of adjacent coins. We can see that the only way to satisfy property (i) is if $A_{T-1} \cap R_1$ contains a pair $\{c_1, c_2\}$ of adjacent coins such that c_1 is adjacent to one of the coins in $A_{T-1} \setminus (R_1 \cup R_2)$, as in Figure IV.17 (left). However, c_1 then violates property (ii), a contradiction.

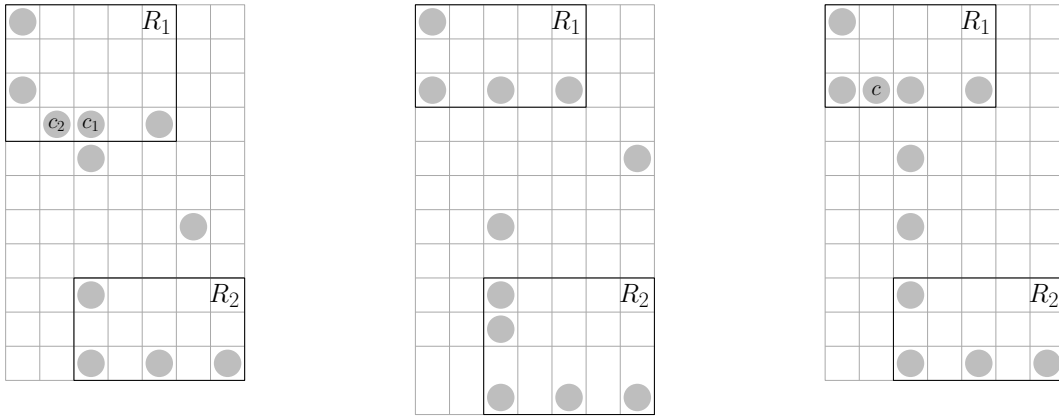


Figure IV.17: The configuration A_{T-1} : a contradiction in Case 2 (left), Case 3 (middle) and Case 4 (right).

- 3) Case 3: At least one of R_1 or R_2 is odd (say R_2 is odd); h is odd.

Again, we just have to find one coin more than what inequalities (IV.2), (IV.3) and (IV.4) give us combined, because if we do then we can conclude that:

$$\begin{aligned} |A| &\geq \left\lfloor \frac{m_1 + n_1}{2} \right\rfloor + \left\lceil \frac{m_2 + n_2}{2} \right\rceil + \left\lfloor \frac{h}{2} \right\rfloor + 1 \\ &\geq \frac{m_1 + n_1}{2} + \frac{m_2 + n_2 + 1}{2} + \frac{h - 1}{2} + 1 = \frac{m_1 + n_1 + m_2 + n_2 + h + 2}{2}. \end{aligned}$$

Therefore, suppose for a contradiction that inequalities (IV.2), (IV.3) and (IV.4) are all tight. For (IV.2) and (IV.3), this means that $A_{T-1} \cap R_1$ and $A_{T-1} \cap R_2$ are both minimum: in particular, neither contains three coins such that one is adjacent to the other two. For (IV.4), this means that $A_{T-1} \setminus (R_1 \cup R_2)$ consists exactly of one coin every two rows in the gap between R_1 and R_2 , none of which is adjacent to R_1 or R_2 since h is odd. See Figure IV.17 (middle). All in all, property (i) is violated, a contradiction.

- 4) Case 4: R_1 and R_2 are both even; h is odd.

In this case, we have to find two coins more than what inequalities (IV.2), (IV.3) and

(IV.4) give us combined, because if we do then we can conclude that:

$$\begin{aligned} |A| &\geq \left\lceil \frac{m_1 + n_1}{2} \right\rceil + \left\lceil \frac{m_2 + n_2}{2} \right\rceil + \left\lfloor \frac{h}{2} \right\rfloor + 2 \\ &\geq \frac{m_1 + n_1}{2} + \frac{m_2 + n_2}{2} + \frac{h - 1}{2} + 2 = \frac{m_1 + n_1 + m_2 + n_2 + h + 3}{2}. \end{aligned}$$

First of all, the same proof as in Case 3 shows that inequalities (IV.2), (IV.3) and (IV.4) cannot all be tight: there needs to be some coin c in A_{t-1} that has (at least) two coins adjacent to it. Now suppose for a contradiction that all three inequalities become tight if we remove c i.e.: $|(A_{T-1} \setminus \{c\}) \cap R_1| = \left\lceil \frac{m_1+n_1}{2} \right\rceil$, $|(A_{T-1} \setminus \{c\}) \cap R_2| = \left\lceil \frac{m_2+n_2}{2} \right\rceil$, $|(A_{T-1} \setminus \{c\}) \setminus (R_1 \cup R_2)| = \left\lfloor \frac{h}{2} \right\rfloor$. See Figure IV.17 (right). If $c \in R_1$ then the two coins adjacent to c in A_{T-1} are inside R_1 as well (indeed, as we have seen in Case 3, the fact that h is odd means that none of the $\lfloor \frac{h}{2} \rfloor$ coins in $A_{T-1} \setminus (R_1 \cup R_2)$ is adjacent to R_1). Therefore $(A_{T-1} \setminus \{c\}) \cap R_1$ has span R_1 , and is minimum since $|(A_{T-1} \setminus \{c\}) \cap R_1| = \left\lceil \frac{m_1+n_1}{2} \right\rceil$. Similarly, $(A_{T-1} \setminus \{c\}) \cap R_2$ has span R_2 and is minimum. By Proposition IV.1.17, all coins in $(A_{T-1} \setminus \{c\}) \cap R_1$ and $(A_{T-1} \setminus \{c\}) \cap R_2$ are thus isolated. Since h is odd, the $\lfloor \frac{h}{2} \rfloor$ coins in $(A_{T-1} \setminus \{c\}) \setminus (R_1 \cup R_2)$ are also isolated. This contradicts property (ii). ■

Corollary IV.2.12. *The puzzle on the right of Figure IV.12 is unsolvable.*

Proof. Let R_1 (resp. R_2) be the top row (resp. the bottom row) of the starting span: R_1 and R_2 are of size 3×1 and separated by a three row gap. Solving this puzzle would mean performing an (R_1, R_2) -split of the starting configuration, which is impossible by Proposition IV.2.11: indeed, we can easily check that it is impossible in two moves or less, and the puzzle contains $6 < \frac{3+1+3+1+3+2}{2}$ coins. ■

IV.2.4.2 A new sufficient condition for a puzzle to be solvable

We now present a result that holds even when some component of $\text{span}(A)$ contains two or more components of $\text{span}(B)$.

Notation IV.2.13. Let C be a configuration. We denote by \min_C the cardinality of minimum configurations with same span as C . Note that, if this span is an $m \times n$ rectangle, then $\min_C = \left\lceil \frac{m+n}{2} \right\rceil$ by Proposition IV.1.17.

Theorem IV.2.14. *Let A and B be configurations such that $|A| = |B| =: N$, and suppose that:*

- (i) $\text{span}(A) \supseteq B$ is a single $m \times n$ rectangle.
- (ii) A has 2 extra coins.
- (iii) B has 2 redundant coins.
- (iv) $N \geq \frac{3}{2} \max(\min_A, \min_B) + 2$.

Then $A \rightarrow B$ in $O(N^3)$ moves.

We would have liked conditions (i) and (ii) to be replaced by the sole condition that A has 2 extra coins relatively to B , however we are not sure how the proof would work in that case. Apart from that, the additional assumption compared to Theorem IV.2.1 is condition (iv), which is not about the quality of the coins involved (extra/redundant) but purely about their quantity, as was suggested by the worst-case puzzles from Corollary IV.2.9. Moreover, these puzzles also show that the bound from condition (iv) is almost tight: indeed, if n is odd for instance, we have $\min_{A_n} = n$ and $\min_{B_n} = n + 1$ so that the puzzle $A_n \xrightarrow{?} B_n$ satisfies $N = \frac{3n-3}{2} = \frac{3}{2} \max(\min_{A_n}, \min_{B_n}) - 3$, just 5 coins away from this bound.

We now proceed with the proof of Theorem IV.2.14. As usual, we are going to route through the canonical configurations, which means the challenge is to go from L_A to L_B . The proof relies on an intuitive way to do so, which consists in forming a wave of coins (by flipping ‘L’s) to sweep across the board while dropping coins at all positions in L_B , as detailed in the proof of the following lemma. Note that this lemma is more general than we use, since the target configuration is not required to be canonical.

Lemma IV.2.15. *Let $m, n \geq 1$ and let L be the canonical ‘L’ of size $m \times n$. Let $k \in \mathbb{N}$. If a configuration $C \subseteq \text{span}(L)$ satisfies $|C| < \min\left(\left\lceil \frac{m+n}{2} \right\rceil - \left\lceil \frac{\min(m,n)}{2} \right\rceil + (k-1), 2(k-1)\right)$, then $L^{+k} \rightarrow C^{+k+|L|-|C|}$ in $O(mn)$ moves.*

Proof. We proceed by induction on the half-perimeter $m+n$. If $m=n=1$, then $C=L$ or $C=\emptyset$ so the result is obvious. Suppose $m+n \geq 3$ and assume the result holds for any half-perimeter lesser than $m+n$. Up to a 90 degree rotation of the board, also assume $m \leq n$. We divide $R := \text{span}(L)$ into two rectangles: a bottom half R_1 and a top half R_2 (if n is even then both halves are equal, otherwise we choose one of them arbitrarily to be bigger than the other by one row). Define $C_1 := C \cap R_1$ and $C_2 := C \cap R_2$. Up to swapping the roles of R_1 and R_2 , assume $|C_1| \leq |C_2|$. Let $n' \in \left\{\left\lfloor \frac{n}{2} \right\rfloor, \left\lceil \frac{n}{2} \right\rceil\right\}$ be the number of rows of R_2 , so that R_2 is of size $m \times n'$. To reach C , we build the bottom half C_1 first, then the top half C_2 .

- 1 We start by building the bottom half of C . We show that $L^{+k} \rightarrow C_0^{+k'}$ where $k' := k+|L|-|C_0|$ and C_0 is the configuration defined as follows: $C_0 \cap R_1 = C_1$, and $C_0 \cap R_2 = L_2$ is the canonical ‘L’ of size $m \times n'$. See Figure IV.18.

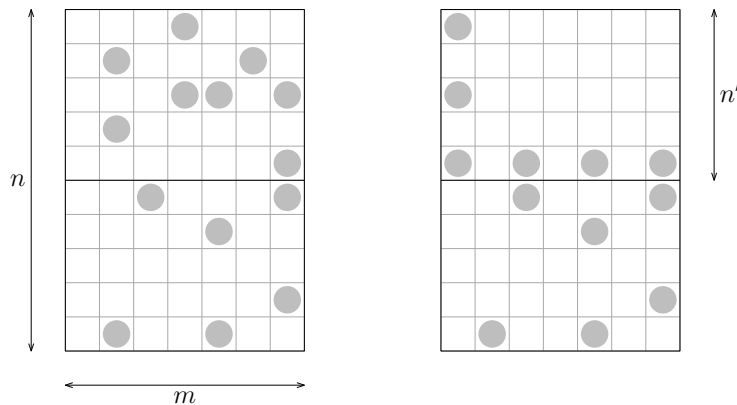


Figure IV.18: An example of a configuration C (left) with its associated configuration C_0 (right).

By our assumption on C , we have $k-1 > \frac{|C|}{2} \geq |C_1|$ (recall that we have assumed without loss of generality that $|C_1| \leq |C_2|$ i.e. C_1 contains at most half of the coins in C) so $k \geq |C_1| + 2$. This allows us to view our k coins in hand as follows:

- We have 2 *supporting coins* that we will use to transform strings.
- We have $|C_1|$ *building coins* that we will drop at the right positions to build C_1 .
- If $k > |C_1| + 2$, the remaining $k - |C_1| - 2$ coins will be kept in hand.

Note that the supporting coins might not remain the same throughout the moves. For example, we might drop a supporting coin, perform some moves, and then pick up a coin: in that case, the picked up coin becomes a supporting coin even if it is not "physically" the same coin that we dropped initially. We proceed in four steps.

- (a) Let p be the bottom-left corner of R_2 : we want to make sure there is a coin at p .

- Case (a1): even R , odd n' . There already is a coin at p .
- Case (a2): even R , even n' . We drop one of our supporting coins at p .
- Case (a3): odd R . If needed, we use a leapfrog to put one of the two adjacent coins at p . Recall that a leapfrog only uses one supporting coin.

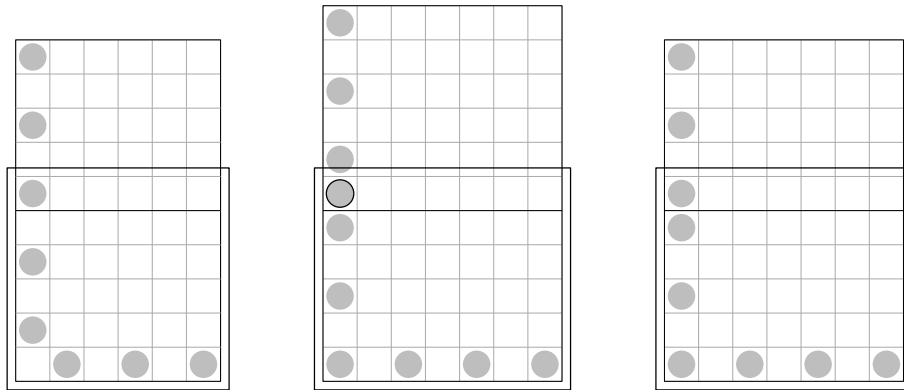


Figure IV.19: The board after step (a). From left to right: case (a1), case (a2), case (a3).

(b) The board now contains an ‘L’ whose extremal coins are at the bottom-left corner of R_2 and the bottom-right corner of R_1 , as highlighted in Figure IV.19. We now use our supporting coin(s) to flip this ‘L’, while building C_1 in the process (see Figure IV.21 for the desired result). To achieve this, we take advantage of the fact that the flip sweeps over the entirety of R_1 . In the subroutines from Figures IV.7 and IV.10, some positions are highlighted by a black outline: whenever one of these positions contains a coin in C_1 , we drop a building coin there at the appropriate moment during the subroutine (one example is detailed in Figure IV.20). Over the flip as a whole, these positions cover all of R_1 except its rightmost column, so that all of C_1 is correctly replicated at the end of this step apart from the coins in the rightmost column. Note that the ‘L’ that we flip is odd in case (a2), so that the lone remaining supporting coin is indeed enough to flip it. See Figure IV.21.

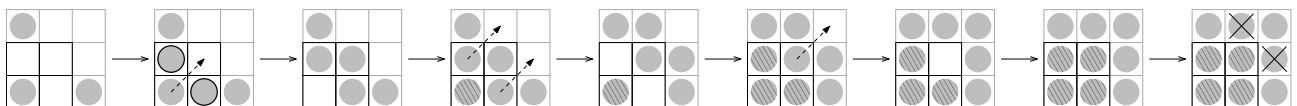


Figure IV.20: How to drop a building coin at any desired position (here we drop four of them, but we can drop less). The building coins are shaded.

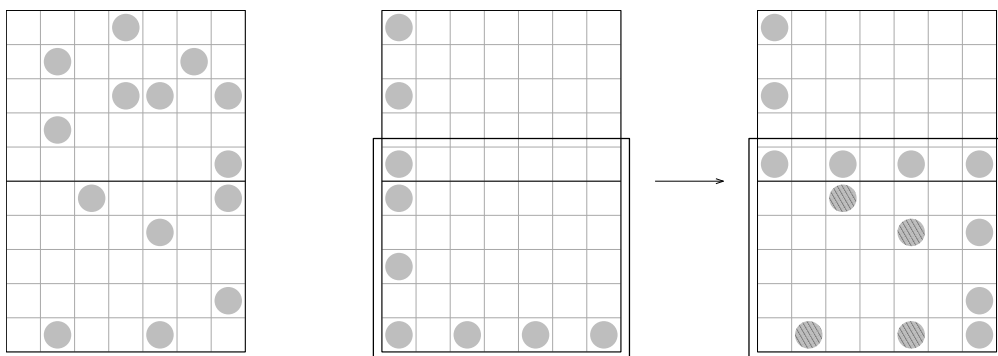


Figure IV.21: Left: a configuration C (the same example as in Figure IV.18). Middle: the board after step (a). Right: the board after step (b).

- (c) We now make sure there is a coin in the bottom-right corner of R_2 . If the newly flipped ‘L’ is odd, this is done with a leapfrog. If it is even, in particular we are not in case (a2), so we can afford to drop one of our two supporting coins at the bottom-right corner of R_2 if needed. In both cases, we still have at least one supporting coin at our disposal. We now correct the rightmost column of R_1 : we drop building coins where they are needed in the holes in-between the coins that are already on board, and then we pick up all coins in that column that are not in C_1 .
- (d) At this point, R_2 either contains an ‘L’ or a string that is almost an ‘L’ apart from the fact it has two pairs of adjacent coins (this can happen if we have dropped a supporting coin at the bottom-right corner of R_2 in step (c)). In this latter case, we use a leapfrog to retrieve the coin in excess so that R_2 contains a true ‘L’. Finally, we leapfrog if needed to make this ‘L’ canonical, so that the full configuration on board is now exactly C_0 . These leapfrogs are always possible because we have at least one supporting coin at our disposal. See Figure IV.22.

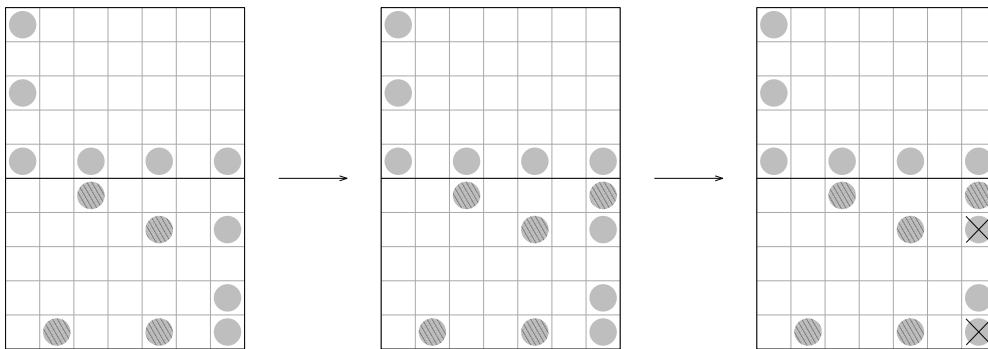


Figure IV.22: Step (d) performed as follow-up to Figure IV.21.

As to the number of moves, since flipping an $m \times n$ ‘L’ is done in $O(mn)$ moves, we have just reached C_0 in $O(mn)$ moves also.

- 2 We now build the top half of C . Since the bottom half is already built, we will not touch it, therefore we want to show that $L_2^{+k'} \rightarrow C_2^{+k'+|L_2|-|C_2|}$ in $O(mn)$ moves. By our induction hypothesis, since R_2 is of half-perimeter $m+n' < m+n$ (recall that $m+n \geq 3$, so $n \geq 2$ hence $n' < n$), it suffices to show that $|C_2| < \min\left(\left\lceil \frac{m+n'}{2} \right\rceil - \left\lceil \frac{\min(m,n')}{2} \right\rceil + (k' - 1), 2(k' - 1)\right)$. We now show both inequalities. Beforehand, since the total number of coins at our disposal is $|L| + k$ and $|C_1| + |L_2|$ of them are on the board at this point, note that:

$$k' = (|L| + k) - (|C_1| + |L_2|) = \left(\left\lceil \frac{m+n}{2} \right\rceil + k\right) - \left(|C_1| + \left\lceil \frac{m+n'}{2} \right\rceil\right). \quad (\text{IV.5})$$

- By our assumption on C , we have $|C_1| = |C| - |C_2| < \left\lceil \frac{m+n}{2} \right\rceil - \left\lceil \frac{\min(m,n)}{2} \right\rceil + (k-1) - |C_2|$. Using equality (IV.5), we get:

$$k' > |C_2| + 1 + \left\lceil \frac{\min(m,n)}{2} \right\rceil - \left\lceil \frac{m+n'}{2} \right\rceil \geq |C_2| + 1 + \left\lceil \frac{\min(m,n')}{2} \right\rceil - \left\lceil \frac{m+n'}{2} \right\rceil,$$

where the last inequality comes from the fact that $n > n'$. We thus get the first desired inequality: $|C_2| < \left\lceil \frac{m+n'}{2} \right\rceil - \left\lceil \frac{\min(m,n')}{2} \right\rceil + (k' - 1)$.

- By our assumption on C , we have $k-1 > \frac{|C|}{2} \geq |C_1|$ hence $k-1 - |C_1| > 0$. Moreover $k-1 - |C_1| = k-1 - |C| + |C_2| > |C_2| - \left\lceil \frac{m+n}{2} \right\rceil + \left\lceil \frac{\min(m,n)}{2} \right\rceil = |C_2| - \left\lceil \frac{m+n}{2} \right\rceil + \left\lceil \frac{m}{2} \right\rceil$. Since all integers $x > y$ with $x > 0$ satisfy $x \geq \frac{y}{2} + 1$, we get $k-1 - |C_1| \geq$

$$\frac{1}{2} \left(|C_2| - \left\lceil \frac{m+n}{2} \right\rceil + \left\lceil \frac{m}{2} \right\rceil \right) + 1.$$

Recall that $k' - 1 = (k - 1 - |C_1|) + \left\lceil \frac{m+n}{2} \right\rceil - \left\lceil \frac{m+n'}{2} \right\rceil$ by equality (IV.5). Therefore:

$$\begin{aligned} k' - 1 &\geq \frac{|C_2|}{2} + \frac{1}{2} \left\lceil \frac{m+n}{2} \right\rceil + \frac{1}{2} \left\lceil \frac{m}{2} \right\rceil - \left\lceil \frac{m+n'}{2} \right\rceil + 1 \\ &\geq \frac{|C_2|}{2} + \frac{m+n}{4} + \frac{m}{4} - \frac{m + \frac{n+1}{2} + 1}{2} + 1 = \frac{|C_2|}{2} + \frac{1}{4} \\ &> \frac{|C_2|}{2}, \end{aligned}$$

from which $|C_2| < 2(k' - 1)$ which concludes. \blacksquare

Proof of Theorem IV.2.14. In fact, we prove a more general result where condition (iv) is replaced by the following double inequality:

$$N > \min_A + \frac{\min_B}{2} + 1 \tag{IV.6}$$

$$N > \min_B + \left\lceil \frac{\min(m, n)}{2} \right\rceil + 1 \tag{IV.7}$$

Let us first check that this assumption is indeed weaker. Suppose that (iv) holds, then:

- $N \geq \min_A + \frac{\min_B}{2} + 2$. Therefore, (IV.6) holds.
- $N \geq \min_B + \frac{\min_A}{2} + 2$. Since N is an integer, this yields $N \geq \min_B + \left\lceil \frac{\min_A}{2} \right\rceil + 2$. Moreover $\left\lceil \frac{\min_A}{2} \right\rceil = \left\lceil \frac{1}{2} \left\lceil \frac{m+n}{2} \right\rceil \right\rceil = \left\lceil \frac{m+n}{4} \right\rceil \geq \left\lceil \frac{\min(m, n)}{2} \right\rceil$, where we have used the fact that any real number x satisfies $\left\lceil \frac{x}{2} \right\rceil = \left\lceil \frac{\lceil x \rceil}{2} \right\rceil$. Therefore, we get $N \geq \min_B + \left\lceil \frac{\min(m, n)}{2} \right\rceil + 2$, so (IV.7) holds.

Assume that conditions (i),(ii),(iii),(IV.6),(IV.7) all hold. As already mentioned, we want to use Corollary IV.2.4 (in this case $L_{A_0} = L_A$ because of our assumption that A has 2 extra coins) so we need to show that $L_A^{+k} \rightarrow L_B^{+k+|L_A|-|L_B|}$ where $k := |A| - |L_A| = N - \min_A = N - \left\lceil \frac{m+n}{2} \right\rceil$. By Lemma IV.2.15, it suffices to show that $\min_B = |L_B| < \min \left(\left\lceil \frac{m+n}{2} \right\rceil - \left\lceil \frac{\min(m, n)}{2} \right\rceil + (k - 1), 2(k - 1) \right)$.

- By (IV.6), we have $\min_B < 2(N - \min_A - 1) = 2(k - 1)$.
- By (IV.7), we have $\min_B < N - \left\lceil \frac{\min(m, n)}{2} \right\rceil - 1 = \left\lceil \frac{m+n}{2} \right\rceil - \left\lceil \frac{\min(m, n)}{2} \right\rceil + (k - 1)$. \blacksquare

IV.2.5 Conclusion and prospects

The case of 2 extra coins turns out to be worse than initially thought. It was believed in [DDV02] that all puzzles with 2 extra coins and 2 redundant coins were solvable, but this is actually not true. The dichotomy between the case of 1 extra coin and the case of 2 extra coins is thus called into question: the case of 3 extra coins, for example, may be just as relevant. Since the canonicalization process only needs 2 extra coins to work, algorithms specializing in the case of 3 or more extra coins likely would not use it, unlike all solving methods that we know of so far: therefore, brand new ideas would be needed to pursue in this direction.

In the case where there are 2 extra coins in A and 2 redundant coins in B , Theorem IV.2.14 gives a sufficient condition that involves the total number of coins and the perimeters of the spans of A and B . The worst-case puzzles described in Corollary IV.2.9 show that the bound from Theorem IV.2.14 is tight in the case of a square span, up to an additive factor $O(1)$. However, for a rectangular $m \times n$ span in general, it looks like discrepancy between m and n tends to make puzzles easier. For instance, and provided we have the inclusion of spans, if $n = 1$

then it seems easy to show that 1 extra coin and 1 redundant coin are always sufficient, and if $n = 2$ then we suspect that 2 extra coins and 2 redundant coins are always sufficient. In general, we think that it could be possible to improve Theorem IV.2.14 by using only $\frac{\min(m,n)}{2} + O(1)$ additional coins.

We now know of two methods to solve general puzzles with 2 extra coins in A and 2 redundant coins in B : the one from Theorem IV.2.1 consists in going from A to 'L's then reverse into B , while the one from Theorem IV.2.14 consists in going from A to an 'L' then sweep across the board and drop coins to create B . It would be good to design other solving methods that would apply to some puzzles that do not meet the conditions of either theorem, such as the puzzle on the left of Figure IV.12.

Conclusion

In this PhD thesis, I have studied several problems around games and hypergraphs, with varying degrees of fulfillment.

First of all, the coin-moving puzzles on the square grid are still far from sorted, not only in the case of one extra coin [Gal19] but also in the case of two extra coins which is much more complicated than what was believed in [DDV02]. We have identified that the difficulty lies in splitting the span, and we have shown how to overcome this in some cases. It turns out the sheer quantity of coins plays a big role in the solvability of puzzles, which came as a surprise and contrasted with my usual structural considerations in games. Overall, several open cases remain for these puzzles, and we lack serious leads as to how to handle them. We thus leave with more questions than we had going into this problem.

On the contrary, the study of the Maker-Breaker game on hypergraphs of rank 3 was much more satisfactory. We managed to obtain the desired structural characterization for the outcome and description of both players' optimal strategies, based on danger intersections. It is unfortunate that the proof is not as beautiful as the result. I cannot exclude the possibility that there exists a shorter and less technical proof, and that I may not have fully grasped the essence of what makes it all work. From this result, we were also able to get the polynomial-time algorithm that we were hoping for. It was not obvious, with 3-SAT in mind for instance, that the problem would remain tractable when making the jump from rank 2 to rank 3. As for future studies, there is a question as to whether the danger approach can be useful for other hypergraph classes. In particular, for $k \geq 4$, is there a constant number of rounds $r(k)$ such that Maker wins on a hypergraph of rank k if and only if she can guarantee the appearance of a k -uniform forcing path during the first $r(k)$ rounds? Since the k -uniform forcing path existence problem is clearly in NP, we know the answer is negative for $k \geq 6$ unless $\text{NP}=\text{PSPACE}$, because deciding the outcome of the Maker-Breaker game is PSPACE-complete in that case [RW21]. The question remains open for $k \in \{4, 5\}$, even though it is believed within the community that these cases are also PSPACE-complete. Other hypergraph classes, not defined by rank, could also be considered.

When it comes to bounds on the hypergraph parameters τ_M and θ_M (optimization of the number of rounds and tokens respectively) depending on the rank k , we basically got exact answers. For $k = 3$, we have shown that a logarithmic number of rounds and just three tokens always suffice for Maker to carry out her winning strategy (which is to make a nunchaku or necklace appear during the first three rounds and then win by dichotomy), and that these bounds are tight. For $k \geq 4$, things are radically different: we have exhibited worst-case hypergraphs for Maker, which show that the trivial bound (number of vertices divided by two) is reached by τ_M and almost reached by θ_M . In a sense, this means Maker's winning strategies are much more complex in the case $k \geq 4$ than they are for $k = 3$, morally comforting the intuition that deciding the outcome of the game is PSPACE-complete for $k \in \{4, 5\}$. A prospect could be to evaluate these bounds in the biased case.

Finally, we have explored some hypergraph problems. We have introduced the notion of λ -linearity to study paths with edge intersections of upper-bounded size. Other paths defined by some restriction on the size of edge intersections existed in the literature, but there only seemed

to be extremal results. Instead, we have inspected the $(k - 2)$ -linear path existence problem in general hypergraphs of rank k . We have shown that the associated connected components are characterized by the maximal archipelago structure, and that they can be computed in polynomial time. In contrast with our study of the Maker-Breaker game on hypergraphs of rank 3, the structures are very visual and the proofs are not too complicated. We have also established a link with the PAFP problem, through a notion of bicolored line graph of a hypergraph. I would be interested in investigating the recognition problem of such line graphs, which carry additional information on the size of (hyper)edge intersections.

I would also like to study some other games related to the Maker-Breaker game on hypergraphs on rank 3.

The Maker-Maker game on hypergraphs of rank 3 is a natural continuation. The algorithmic complexity of this problem is unknown. The forcing technique, which was not vital in the Maker-Breaker convention since it could be supplanted by the dichotomy technique, could prove essential here. Indeed, it allows the player to control exactly which vertices are taken by the opponent, which helps prevent them from creating threats of their own. There seems to be an importance of keeping the initiative in the attack for the first player, and the second player can regain that initiative (and thus draw the game) without necessarily destroying all subhypergraphs that would be dangers in the Maker-Breaker game. This is what happens in 3×3 tic-tac-toe for example. Therefore, there might be a different notion of danger intersection for the Maker-Maker convention, such that picking a vertex that is adjacent to a danger is sometimes enough to destroy it.

The Avoider-Enforcer game on hypergraphs of rank 3 is another topic of interest. Joint work with Valentin Gledel¹ and Aline Parreau² is currently ongoing in the linear case. It turns out chains play a key role in this convention also. Similarly to the Maker-Breaker convention, the non-linear case seems to be more difficult. It is possible that the structural lemmas about chains and cycles from this PhD thesis will end up being useful again.

Finally, I believe our idea of introducing tokens into the game has potential. One could study "token positional games", where both players have a fixed number of tokens to play with (finite or infinite, and not necessarily the same for both). This version is natural, moreover it is in line with the way some positional games were played historically. Along with Nacim Oijid³, I am currently interested in the algorithmic complexity of such games. It is likely that all conventions remain PSPACE-complete in general.

I will conclude this dissertation with a remark on a potential unification and generalization of some positional games.

A *transversal* of an edge set E is a set of vertices T such that $T \cap e \neq \emptyset$ for all $e \in E$. Given an edge set E , let $t(E)$ denote the set of all minimal transversals of E (note that t is an involution). In the Maker-Breaker game on a hypergraph H , Maker's goal is to color some edge of H in red, and Breaker's goal amounts to coloring some transversal of $E(H)$ in blue. Therefore, the denomination of the game is falsely asymmetrical as to the player's roles: both of them are actually "Maker", except they are not making the same edges. A more general game could thus be defined, where: we have two edge sets E_A and E_B on the same vertex set V , Alice colors vertices in red trying to make a monochromatic red edge in E_A , Bob colors vertices in blue trying to make a monochromatic blue edge in E_B , and the first player to accomplish their goal wins. Draws are possible. This would define a unified family of achievement positional games, containing Maker-Maker games, Maker-Breaker games, and everything in between. Maker-Maker

¹ Umeå University, Sweden.

² Centre National de la Recherche Scientifique. LIRIS, University Lyon 1, France.

³ LIRIS, University Lyon 1, France.

games correspond to the particular case $E_A = E_B$. Maker-Breaker games correspond to the particular case $E_A = t(E_B)$ i.e. $E_B = t(E_A)$. Similarly, we can define a unified family of avoidance positional games.

It would be instructive to see which principles, out of all those that hold for Maker-Breaker and Maker-Maker games, are still valid for the general achievement game described above. Indeed, some rely on the specific link between E_A and E_B , whereas some purely stem from the achievement nature of the game. For instance, strategy stealing still applies to show that more moves is always better and that both players prefer to start. Another prospect would be to solve the general achievement game for simple hypergraph classes. I think the unbiased 2-uniform case, meaning all elements of $E_A \cup E_B$ are of size 2, would be a good place to start. At first glance, it looks like matchings and augmenting paths play a significant role.

To generalize even further, one could consider a *vertex-partizan* version where $V = V_A \cup V_B$ (not necessarily a partition). Alice can only pick vertices in V_A , and Alice *completing* an edge $e \in E_A$ means that all vertices in $e \cap V_A$ are colored in red before any vertex in $e \cap V_B$ is colored in blue. The rules are analogous for Bob, and the first player to complete an edge wins, otherwise the game ends in a draw. In particular, this version includes some natural games which are not otherwise considered as positional games. For instance, consider a variation of the Maker-Breaker game where Breaker selects edges rather than vertices. This means that, in each round, Maker picks a vertex (as usual) whereas Breaker removes an edge (but its vertices remain). This game can be modelled as a vertex-partizan game, by adding one vertex per edge which is only playable by Bob, while the rest is only playable by Alice.

Bibliography

- [AL17] G. Ausiello, L. Laura. Directed hypergraphs: introduction and fundamental algorithms — a survey. *Theoret. Comput. Sci.*, 658(B), 2017, pp. 293–306. <http://dx.doi.org/10.1016/j.tcs.2016.03.016>.
- [AMS⁺19] N. Alon, R. Morris, W. Samotij, F. Galliot. Private exchanges, 2019.
- [AvH96] L. V. Alus, H. J. van den Herik, M. P. H. Huntjens. Go-moku solved by new search techniques. *Comput. Intell.*, 12(1), 1996, pp. 7–23. <http://dx.doi.org/10.1111/j.1467-8640.1996.tb00250.x>.
- [BC82] J. Beck, L. Csirmaz. Variations on a game. *J. Comb. Theory, Ser. A*, 33(3), 1982, pp. 297–315. [http://dx.doi.org/10.1016/0097-3165\(82\)90042-5](http://dx.doi.org/10.1016/0097-3165(82)90042-5).
- [BCG82] E. R. Berlekamp, J. H. Conway, R. K. Guy. *Winning Ways for your Mathematical Plays*. Academic Press, London, 2nd ed., 1982. ISBN 1-56881-130-6.
- [BCM⁺18] K. Böhmová, J. Chalopin, M. Mihalák, G. Proietti, P. Widmayer. Sequence hypergraphs: paths, flows, and cuts. In *Adventures Between Lower Bounds and Higher Altitudes: Essays Dedicated to Juraj Hromkovič on the Occasion of His 60th Birthday*. Springer International Publishing, Cham, 2018, pp. 191–215. http://dx.doi.org/10.1007/978-3-319-98355-4_12.
- [BCv⁺07] P. Bonsma, L. Cereceda, J. van den Heuvel, M. Johnson. Finding paths between graph colourings: computational complexity and possible distances. In *Electron. Notes Discrete Math.*, proc. 7th European Conference on Combinatorics, Graph Theory and Applications (EuroComb’07), vol. 29, 2007. pp. 463–469. <http://dx.doi.org/10.1016/j.endm.2007.07.073>.
- [BDD⁺22] G. Bagan, Q. Deschamps, E. Duchêne, B. Durain, B. Effantin, V. Gledel, N. Oijid, A. Parreau. Incidence, a scoring positional game on graphs. 2022. Preprint: <https://arxiv.org/abs/2211.01083>.
- [Bec81] J. Beck. On positional games. *J. Comb. Theory, Ser. A*, 30(2), 1981, pp. 117–133. [http://dx.doi.org/10.1016/0097-3165\(81\)90001-7](http://dx.doi.org/10.1016/0097-3165(81)90001-7).
- [Bec82] J. Beck. Remarks on positional games. *Acta Math. Hungar.*, 40(1-2), 1982, pp. 65–71. <http://dx.doi.org/10.1007/BF01897304>.
- [Bec02] J. Beck. Ramsey games. *Discrete Math.*, 249(1-3), 2002, pp. 3–30. [http://dx.doi.org/10.1016/S0012-365X\(01\)00224-2](http://dx.doi.org/10.1016/S0012-365X(01)00224-2).
- [Bec08] J. Beck. *Combinatorial Games: Tic-Tac-Toe Theory*. Academic Press, Cambridge, 2008. ISBN 978-0-521-46100-9.

- [Bei70] L. W. Beineke. Characterizations of derived graphs. *J. Combin. Theory*, 9(2), 1970, pp. 129–135. [http://dx.doi.org/10.1016/S0021-9800\(70\)80019-9](http://dx.doi.org/10.1016/S0021-9800(70)80019-9).
- [Ber73] C. Berge. *Graphs and Hypergraphs*, vol. 6 of *North-Holland Mathematical Library*. Elsevier, 1973. ISBN 978-0-444-87603-4. [http://dx.doi.org/10.1016/s0924-6509\(09\)x7013-3](http://dx.doi.org/10.1016/s0924-6509(09)x7013-3).
- [BFM⁺23] J. Bensmail, F. Fioravantes, F. Mc Inerney, N. Nisse, N. Oijid. The Maker-Breaker largest connected subgraph game. *Theoret. Comput. Sci.*, 943, 2023, pp. 102–120. <http://dx.doi.org/10.1016/j.tcs.2022.12.014>.
- [BGM⁺21] A. Bhattacharya, A. Godinho, P. Majumder, N. M. Singhi. Reconstruction of hypergraphs from line graphs and degree sequences. 2021. Preprint: <https://arxiv.org/abs/2104.14863>.
- [BH19] K. Burke, R. A. Hearn. PSPACE-complete two-color planar placement games. *Internat. J. Game Theory*, 48, 2019, pp. 393–410. <http://dx.doi.org/10.1007/s00182-018-0628-8>.
- [BL00] M. Bednarska, T. Luczak. Biased positional games for which random strategies are nearly optimal. *Combinatorica*, 20, 2000, pp. 477–488. <http://dx.doi.org/10.1007/s004930070002>.
- [Bys04] J. M. Byskov. Maker-Maker and Maker-Breaker games are PSPACE-complete. *BRICS Report Series*, 11(14). <http://dx.doi.org/10.7146/brics.v11i14.21839>.
- [CE78] V. Chvátal, P. Erdős. Biased positional games. *Ann. Disc. Math.*, 2, 1978, pp. 221–229. [http://dx.doi.org/10.1016/S0167-5060\(08\)70335-2](http://dx.doi.org/10.1016/S0167-5060(08)70335-2).
- [CFK⁺12] D. Clemens, A. Ferber, M. Krivelevich, A. Liebenau. Fast strategies in Maker–Breaker games played on random boards. *Combin. Probab. Comput.*, 21(6), 2012, pp. 897–915. <http://dx.doi.org/10.1017/S0963548312000375>.
- [CKT⁺01] T. Chen, M.-Y. Kao, M. Tepel, J. Rush, G. M. Church. A dynamic programming approach to de novo peptide sequencing via tandem mass spectrometry. *J. Comput. Biol.*, 8(3), 2001, pp. 325–337. <http://dx.doi.org/10.1089/10665270152530872>.
- [DDV02] E. D. Demaine, M. L. Demaine, H. A. Verill. Coin-moving puzzles. In *More Games of No Chance*. Cambridge University Press. ISBN 978-0521808323, pp. 405–431.
- [DGM⁺23] E. Duchêne, V. Gledel, F. Mc Inerney, N. Nisse, N. Oijid, A. Parreau, M. Stojaković. Complexity of Maker-Breaker games on edge sets of graphs. 2023. Preprint: <https://arxiv.org/abs/2302.10972>.
- [DGP⁺20] E. Duchêne, V. Gledel, A. Parreau, G. Renault. Maker-Breaker domination game. *Discrete Math.*, 343(9), 111955. <http://dx.doi.org/10.1016/j.disc.2020.111955>.
- [DLM⁺17] A. Dudek, S. La Fleur, D. Mubayi, V. Rödl. On the size-Ramsey number of hypergraphs. *J. Graph Theory*, 86(1), 2017, p. 104–121. <http://dx.doi.org/10.1002/jgt.22115>.
- [EL74] P. Erdős, L. Lovász. Problems and results on 3-chromatic hypergraphs and some related questions. *Coll. Math. Soc. J. Bolyai*, 10.

- [ES73] P. Erdős, J. Selfridge. On a combinatorial game. *J. Comb. Theory, Ser. A*, 14(3), 1973, pp. 298–301. <http://dx.doi.org/10.1016/0097-3165%2873%2990005-8>.
- [FGM⁺15] S. A. Fenner, D. Grier, J. Messner, L. Schaeffer, T. Thierauf. Game values and computational complexity: an analysis via black-white combinatorial games. In *Algorithms and Computation*, vol. 9472 of *LNCS*. Springer, Berlin, Heidelberg, 2015, pp. 689–699. http://dx.doi.org/10.1007/978-3-662-48971-0_58.
- [FJS14] Z. Füredi, T. Jiang, R. Seiver. Exact solution of the hypergraph Turán problem for k -uniform linear paths. *Combinatorica*, 34(3), 2014, pp. 299–322. <http://dx.doi.org/10.1007/s00493-014-2838-4>.
- [FM22] J. Forcan, M. Mikalački. Maker-Breaker total domination game on cubic graphs. *Discrete Math. Theor. Comput. Sci.*, 24(1). <http://dx.doi.org/10.46298/dmtcs.8529>.
- [Gal19] F. Galliot. A coin-moving game on graphs. 2019. Master’s thesis: <https://dumas.ccsd.cnrs.fr/dumas-03160998>.
- [Gar59] M. Gardner. *Hexaflexagons and Other Mathematical Diversions*, chap. Ticktacktoe. University Of Chicago Press, 2nd ed. ISBN 978-0226282541, 1959. pp. 38–40.
- [Gar61] M. Gardner. *The second scientific american book of mathematical puzzles and diversions*, chap. Recreational Topology. University Of Chicago Press, 2nd ed. ISBN 978-0226282534, 1961. pp. 84–87.
- [Gar75] M. Gardner. *Mathematical Carnival*, chap. Penny Puzzles. Alfred A. Knopf, New York, 1975. pp. 12–26.
- [GGP⁺22] V. Gledel, S. Gravier, A. Parreau, F. Galliot. Private exchanges, 2022.
- [GGS21] F. Galliot, S. Gravier, I. Sivignon. An update on the coin-moving game on the square grid. 2021. Preprint: <https://hal.science/hal-03157069>. Accepted in *Games of No Chance*, vol. 6.
- [GGS22a] F. Galliot, S. Gravier, I. Sivignon. $(k - 2)$ -linear connected components in hypergraphs of rank k . 2022. Preprint: <https://hal.science/hal-03412083>. Submitted to *Discrete Math. Theor. Comput. Sci.*, special issue for the 11th International Colloquium on Graph Theory and combinatorics (ICGT 2022), Montpellier, France.
- [GGS22b] F. Galliot, S. Gravier, I. Sivignon. Maker-Breaker is solved in polynomial time on hypergraphs of rank 3. 2022. Preprint: <https://arxiv.org/abs/2209.12819>. Submitted to *J. Comb. Theory, Ser. A*.
- [GHI⁺20] V. Gledel, M. A. Henning, V. Irić, S. Klavžar. Maker-Breaker total domination game. *Discrete Appl. Math.*, 282, 2020, pp. 96–107. <http://dx.doi.org/10.1016/j.dam.2019.11.004>.
- [GIK19] V. Gledel, V. Irić, S. Klavžar. Maker-Breaker domination number. *Bull. Malays. Math. Sci. Soc.*, 42, 2019, pp. 1773–1789. <http://dx.doi.org/10.1007/s40840-019-00757-1>.
- [GKL10] E. Győri, G. Y. Katona, N. Lemons. Hypergraph extensions of the Erdős-Gallai theorem. *Electron. Notes Discrete Math.*, 36, 2010, pp. 655–662. <http://dx.doi.org/10.1016/j.endm.2010.05.083>.

- [GLP93] G. Gallo, G. Longo, S. Pallottino. Directed hypergraphs and applications. *Discrete Appl. Math.*, 42(2-3), 1993, pp. 177–201. [http://dx.doi.org/10.1016/0166-218X\(93\)90045-P](http://dx.doi.org/10.1016/0166-218X(93)90045-P).
- [GLS20] R. Gu, J. Li, Y. Shi. Anti-Ramsey numbers of paths and cycles in hypergraphs. *SIAM J. Discrete Math.*, 34(1), 2020, pp. 271–307. <http://dx.doi.org/10.1137/19M1244950>.
- [GMO76] H. N. Gabow, S. N. Maheswari, L. J. Osterweil. On two problems in the generation of program test paths. *IEEE Trans. Softw. Eng.*, 2(3), 1976, pp. 227–231. <http://dx.doi.org/10.1109/tse.1976.233819>.
- [GO23] V. Gledel, N. Oijid. Avoidance games are PSPACE-complete. In *40th International Symposium on Theoretical Aspects of Computer Science (STACS 2023)*, vol. 254 of *LIPICs*. Dagstuhl, Germany, 2023, pp. 34:1–34:19. <http://dx.doi.org/10.4230/LIPICs.STACS.2023.34>.
- [GPR⁺14] A. Guzzo, A. Pugliese, A. Rullo, D. Saccà. Intrusion detection with hypergraph-based attack models. In *Graph Structures for Knowledge Representation and Reasoning*, proc. Third International Workshop, GKR 2013, Beijing, China, vol. 8323 of *LNCS*. Springer International Publishing, Cham, 2014, pp. 58–73. http://dx.doi.org/10.1007/978-3-319-04534-4_5.
- [GSZ80] R. K. Guy, J. L. Selfridge, T. G. L. Zetters. S10. *The American Mathematical Monthly*, 87(7), 1980, pp. 575–576. <http://dx.doi.org/10.2307/2321433>.
- [GZR⁺15] J. Gao, Q. Zhao, W. Ren, A. Swami, R. Ramanathan, A. Bar-Noy. Dynamic shortest path algorithms for hypergraphs. *IEEE/ACM Trans. Netw.*, 23(6), 2015, pp. 1805–1817. <http://dx.doi.org/10.1109/TNET.2014.2343914>.
- [Har81] F. Harary. Achievement and avoidance games designed from theorems. *Rendiconti del Seminario Matematico e Fisico di Milano*, 51, 1981, pp. 163–172. <http://dx.doi.org/10.1007/BF02924819>.
- [Har82] F. Harary. Achievement and avoidance games for graphs. *North-Holland Mathematics Studies*, 62, 1982, pp. 111–119. [http://dx.doi.org/10.1016/S0304-0208\(08\)73554-0](http://dx.doi.org/10.1016/S0304-0208(08)73554-0).
- [HJ63] A. W. Hales, R. I. Jewett. Regularity and positional games. *Trans. Amer. Math. Soc.*, 106, 1963, pp. 222–229. http://dx.doi.org/10.1007/978-0-8176-4842-8_23.
- [HK97] P. Hlineny, J. Kratochvil. Computational complexity of the Krausz dimension of graphs. In *Graph-Theoretic Concepts in Computer Science*, proc. 23rd International Workshop, WG'97, Berlin, Germany, vol. 1335 of *LNCS*. Springer, Berlin, 1997, pp. 214–228. <http://dx.doi.org/https://doi.org/10.1007/BFb0024500>.
- [HKK⁺12] M. T. Hajiaghayi, R. Khandekar, G. Kortsarz, J. Mestre. The checkpoint problem. *Theoret. Comput. Sci.*, 452, 2012, pp. 88–99. <http://dx.doi.org/10.1016/j.tcs.2012.05.021>.
- [HKS⁺08] D. Hefetz, M. Krivelevich, M. Stojaković, T. Szabó. Planarity, colorability, and minor games. *SIAM J. Discrete Math.*, 22(1), 2008, pp. 194–212. <http://dx.doi.org/10.1137/060654414>.

- [HKS⁺09] D. Hefetz, M. Krivelevich, M. Stojaković, T. Szabó. Fast winning strategies in Maker-Breaker games. *J. Comb. Theory, Ser. B*, 99(1), 2009, pp. 39–47. <http://dx.doi.org/10.1016/j.jctb.2008.04.001>.
- [HKS⁺14] D. Hefetz, M. Krivelevich, M. Stojaković, T. Szabó. *Positional games*. Springer, Basel, 2014. <http://dx.doi.org/10.1007/978-3-0348-0825-5>.
- [IOO23] T. Ito, H. Ono, Y. Otachi. Reconfiguration of cliques in a graph. *Discrete Appl. Math.*, 333, 2023, pp. 43–58. <http://dx.doi.org/10.1016/j.dam.2023.01.026>.
- [Jac15] E. Jackowska. The 3-color Ramsey number for a 3-uniform loose path of length 3. *Australas. J. Combin.*, 63(2), 2015, pp. 314–320.
- [JPR16] E. Jackowska, J. Polcyn, A. Ruciński. Turán numbers for 3-uniform linear paths of length 3. *Electron. J. Combin.*, 23(2), 2016, p. P2.30. <http://dx.doi.org/10.37236/5320>.
- [KKY20] C. X. Kang, S. Klavžar, I. G. Yero. Maker-Breaker resolving game. *Bull. Malays. Math. Sci. Soc.*, 44, 2020, pp. 2081–2099. <http://dx.doi.org/10.1007/s40840-020-01044-0>.
- [KP09] P. Kolman, O. Pangrac. On the complexity of paths avoiding forbidden pairs. *Discrete Appl. Math.*, 157(13), 2009, pp. 2871–2876. <http://dx.doi.org/10.1016/j.dam.2009.03.018>.
- [Kra42] M. Kraitichik. *Mathematical Recreations*, chap. Positional games. W.W. Norton & Company inc., New York, 2nd ed., 1942. pp. 290–292.
- [KS08] M. Krivelevich, T. Szabó. Biased positional games and small hypergraphs with large covers. *Electron. J. Combin.*, 15, R70. <http://dx.doi.org/10.37236/794>.
- [KSG73] K. Krause, R. Smith, M. Goodwin. Optimal software test planning through automated network analysis. *Proc. IEEE Symp. on Computer Software Reliability*, 1973, pp. 18–22.
- [Kut04] M. Kutz. Weak positional games. In *The angel problem, positional games, and digraph roots*, PhD dissertation, Freie Universität Berlin, 2004.
- [Kut05] M. Kutz. Weak positional games on hypergraphs of rank three. In *Discrete Math. Theor. Comput. Sci.*, proc. 3rd European Conference on Combinatorics, Graph Theory and Applications (EuroComb’05), vol. AE, 2005. pp. 31–36. <http://dx.doi.org/10.46298/dmtcs.3422>.
- [Kva86] Problems from the 49th Moscow mathematical olympiad. *Kvant*, 9, 1986, p. 57. In Russian.
- [Lan51] H. Langman. Curiosa 261: a disc puzzle. *Scr. Math.*, 17(1-2), 1951, p. 144.
- [Lan53] H. Langman. Curiosa 342: easy but not obvious. *Scr. Math.*, 19(4), 1953, p. 242.
- [Leh64] A. Lehman. A solution of the Shannon switching game. *J. Soc. Indust. Appl. Math.*, 12(4), 1964, pp. 687–725. <http://dx.doi.org/10.1137/0112059>.
- [Lov73] L. Lovász. Coverings and colorings of hypergraphs. In *Proc. 4th Southeastern Conf. on Comb*, vol. 8 of *Congressus Numerantium*. Utilitas Mathematica, 1973, pp. 3–12.

- [Lov77] L. Lovász. Problem 9. In *Beiträge zur Graphentheorie und deren Anwendungen*, proc. Internationalen Kolloquium, Oberhof, DDR, 1977. p. 313.
- [Lu92] X. Lu. Hamiltonian games. *J. Comb. Theory, Ser. B*, 55(1), 1992, pp. 18–32. [http://dx.doi.org/10.1016/0095-8956\(92\)90030-2](http://dx.doi.org/10.1016/0095-8956(92)90030-2).
- [Mal20] A. Malekshahian. Strategy stealing in triangle avoidance games. 2020. Preprint: <https://arxiv.org/abs/2001.10116>.
- [MRH74] E. Mead, A. Rosa, C. Huang. The game of Sim: a winning strategy for the second player. *Math. Mag.*, 47(5), 1974, pp. 243–247. <http://dx.doi.org/10.1080/0025570X.1974.11976415>.
- [MT97] Y. Metelsky, R. I. Tyshkevich. On line graphs of linear 3-uniform hypergraphs. *J. Graph Theory*, 25(4), 1997, pp. 243–251. [http://dx.doi.org/10.1002/\(SICI\)1097-0118\(199708\)25:4<243::AID-JGT1>3.0.CO;2-K](http://dx.doi.org/10.1002/(SICI)1097-0118(199708)25:4<243::AID-JGT1>3.0.CO;2-K).
- [NRS⁺82] R. N. Naik, S. B. Rao, S. S. Shrikhande, N. M. Singhi. Intersection graphs of k -uniform linear hypergraphs. *European J. Combin.*, 3, 1982, pp. 159–172. [http://dx.doi.org/10.1016/S0195-6698\(82\)80029-2](http://dx.doi.org/10.1016/S0195-6698(82)80029-2).
- [NSS16] R. Nenadov, A. Steger, M. Stojaković. On the threshold for the Maker-Breaker H -game. *Random Structures Algorithms*, 49(3), 2016, pp. 558–578. <http://dx.doi.org/10.1002/rsa.20628>.
- [OS14] G. R. Omid, M. Shahsiah. Ramsey numbers of 3-uniform loose paths and loose cycles. *J. Combin. Theory Ser. A*, 121, 2014, pp. 64–73. <http://dx.doi.org/10.1016/j.jcta.2013.09.003>.
- [PRT81] S. Poljak, V. Rödl, D. Turzík. Complexity of representation of graphs by set systems. *Discrete Appl. Math.*, 3(4), 1981, pp. 301–312. [http://dx.doi.org/10.1016/0166-218X\(81\)90007-X](http://dx.doi.org/10.1016/0166-218X(81)90007-X).
- [Ram30] F. P. Ramsey. On a problem of formal logic. *Proc. Lond. Math. Soc.*, s2-30(1), 1930, pp. 264–286. <http://dx.doi.org/https://doi.org/10.1112/plms/s2-30.1.264>.
- [RW20] M. L. Rahman, T. Watson. Tractable unordered 3-CNF games. In *LATIN 2020: Theoretical Informatics*, proc. Latin American Symposium on Theoretical Informatics, vol. 12118 of *Lecture Notes in Comput. Sci.* Springer International Publishing, Cham, 2020, pp. 360–372. http://dx.doi.org/10.1007/978-3-030-61792-9_29.
- [RW21] M. L. Rahman, T. Watson. 6-uniform Maker-Breaker game is PSPACE-complete. In *38th International Symposium on Theoretical Aspects of Computer Science (STACS 2021)*, vol. 187 of *LIPICs*. Schloss Dagstuhl, Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2021, pp. 57:1–57:15. <http://dx.doi.org/10.4230/LIPICs.STACS.2021.57>.
- [Sch78] T. J. Schaefer. On the complexity of some two-person perfect-information games. *J. Comput. Syst. Sci.*, 16(2), 1978, pp. 185–225. [http://dx.doi.org/10.1016/0022-0000\(78\)90045-4](http://dx.doi.org/10.1016/0022-0000(78)90045-4).
- [Sim68] G. J. Simmons. The game of SIM. In *Mathematical Solitaires and Games*. Routledge, New York. ISBN 978-0895030177, 1968. <http://dx.doi.org/10.4324/9781315224169-11>.

- [SMN16] A. Suzuki, A. E. Mouawad, N. Nishimura. Reconfiguration of dominating sets. *J. Comb. Optim.*, 32, 2016, pp. 1182–1195. <http://dx.doi.org/10.1007/s10878-015-9947-x>.
- [SS23] M. Stojaković, J. Stratijev. On strong avoiding games. *Discrete Math.*, 346(3), 113270. <http://dx.doi.org/10.1016/j.disc.2022.113270>.
- [SST09] P. V. Skums, S. V. Suzdal, R. I. Tyshkevich. Edge intersection graphs of linear 3-uniform hypergraphs. *Discrete Math.*, 309(11), 2009, pp. 3500–3517. <http://dx.doi.org/10.1016/j.disc.2007.12.082>.
- [Tom12] I. Tomescu. Some results on chromaticity of quasi-linear paths and cycles. *Electron. J. Combin.*, 19(2), P23. <http://dx.doi.org/10.37236/2370>.
- [TT09] M. Thakur, R. Tripathi. Linear connectivity problems in directed hypergraphs. *Theoret. Comput. Sci.*, 410(27-29), 2009, pp. 2592–2618. <http://dx.doi.org/10.1016/j.tcs.2009.02.038>.
- [Wal14] D. G. Walker. *A Book of Historic Board Games*. Lulu, 2014. ISBN 978-1326034801.
- [WP21] B. Wu, Y. Peng. Lagrangian densities of short 3-uniform linear paths and Turán numbers of their extensions. *Graphs Combin.*, 37(3), 2021, pp. 711–729. <http://dx.doi.org/10.1007/s00373-020-02270-w>.
- [Wrz20] W. Wrzos-Kaminska. A simpler winning strategy for Sim. 2020. Preprint: <https://arxiv.org/abs/2001.04024>.
- [Yin97] H. Yinnone. On paths avoiding forbidden pairs of vertices in a graph. *Discrete Appl. Math.*, 74(1), 1997, pp. 85–92. [http://dx.doi.org/10.1016/S0166-218X\(96\)00017-0](http://dx.doi.org/10.1016/S0166-218X(96)00017-0).
- [Zas82] C. Zaslavsky. *Tic Tac Toe: and Other Three-In-A Row Games from Ancient Egypt to the Modern Computer*. Harpercollins, 1982. ISBN 978-0690043167.

