Rational Behaviour and Strategy Construction in Infinite Multiplayer Games

Michael Ummels

Diplomarbeit

vorgelegt der Fakultät für
Mathematik, Informatik und Naturwissenschaften der
Rheinisch-Westfälischen Technischen Hochschule Aachen

angefertigt am
Lehr- und Forschungsgebiet
Mathematische Grundlagen der Informatik
Prof. Dr. Erich Grädel

Juli 2005

Hiermit versichere ich, dass ich die Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Aachen, den 26. Juli 2005

(Michael Ummels)

Contents

Chapter 1

Introduction

After its foundation by von Neumann and Morgenstern [vNM44] in the 1940s, game theory has quickly become a valuable tool for modelling the interaction of several agents with individual and possibly conflicting objectives. It has applications in many scientific fields such as economy, sociology, biology, logic and computer science. Whereas the games used in the former three areas are usually *finite* in the sense that any play of these games ends after a finite number of steps (in fact often after only a single step), the games used in logic and computer science are in general *infinite*.

More precisely, an infinite game is played for an infinite number of rounds. In each round, one player chooses an action depending on the sequence of actions chosen in previous rounds. After infinitely many rounds, an infinite sequence of actions has emerged, called the *outcome* of the game. Each player receives a *payoff* determined by a payoff function mapping each possible outcome to a real number in the interval $[0, 1]$. We will concentrate on games where the possible payoffs are just 0 and 1, in which case a *winning condition*, given as an abstract set of possible outcomes, is used to determine whether a player wins (payoff 0) or loses (payoff 1) an outcome.

In logic, games are used to evaluate a logical formula in a mathematical structure and to compare mathematical structures with respect to their (in)distinguishability by a certain logic. Games of the former kind are called *model checking games*; games of the latter kind are called *model comparison games*. Whereas model checking and model comparison games for plain modal or first-order logic are finite games, infinite games arise as model checking games for fixed point logics and as model comparison games for infinitary logics.

In computer science, games are used for the verification and synthesis of reactive systems, i.e. systems that interact with their environment. As soon as the systems under consideration do not necessarily terminate, these games become infinite. For example, the behaviour of an operating system can be understood as an infinite game between the system and its users.

So far, the games used in logic and computer science have usually been *two-player zero-sum games*, i.e. games with only two players where one player wins if and only if the other player loses. For example, in a model checking game, one player wants to show that the given formula holds in the given structure whereas the other player wants to show that this is not the case. Analogously, in a game used for verification, one player wants to show that the given system is able to react against its environment in such a way that the resulting execution path of the system fulfils the given specification whereas the other player wants to show that this is not the case.

Only in recent years, computer scientists have turned to games with arbitrarily many players that are not necessarily zero-sum. Their motivation has been the verification of multi-component systems. Indeed, the general model of an infinite game offered by game theory seems to be the natural framework for modelling a system with several interacting components, each having its own specification.

Equilibria capture rational behaviour in infinite games with an arbitrary number of players. Intuitively, the game is in an equilibrium state if no player can receive a better payoff by changing her behaviour. Research in computer science [CJM04, CHJ04] has focussed on the concept of a *Nash equilibrium*. However, we will focus on the concept of a *subgame perfect equilibrium* because it seems to be more suitable for infinite games.

Nash or subgame perfect equilibria may not exist in general. However, Chatterjee et al. [CJM04] showed that Nash equilibria exist in games where the winning condition of each player is given by a Borel set in the usual topology on infinite sequences. We extend their result to subgame perfect equilibria. In verification, winning conditions are usually sets of infinite sequences definable in monadic second-order logic (MSO). These sets occur in very low levels of the Borel hierarchy (namely in $\Sigma^0_3 \cap \Pi^0_3$).

The mere existence of a subgame perfect equilibrium is often not satisfactory because they may be too complex to be implemented. In a general framework of games played on a directed graph as an *arena*, we show the existence of subgame perfect equilibria in strategies with bounded memory for games with a finite arena and MSO-definable winning conditions. As an intermediate step, we handle games where the winning condition of each player is a parity condition. These games are a natural generalisation of *parity games*, which have received much interest in recent years.

Computing subgame perfect equilibria in games with MSO-definable winning conditions is another interesting problem. We present two algorithms for this purpose, the first one computing an arbitrary subgame perfect equilibrium and the second one computing an optimal subgame perfect equilibrium. However, the quest for optimality seems to have its price here as the amount of time and space required by the latter algorithm is substantially larger. In fact, there is complexity-theoretical evidence that this is not the fault of our algorithm but of the problem itself.

Outline

In Chapter 2, we introduce the model of an infinite game and discuss the concepts of a Nash and a subgame perfect equilibrium. The last part of Chapter 2 deals with a characterisation of the two notions in the two-player zero-sum case.

Chapter 3 deals with games where the winning condition of each player is a Borel set. After recalling some results for the two-player zero-sum case, we show that subgame perfect equilibria exist in these games.

In Chapter 4, we discuss games played on directed graphs with relatively simple winning conditions. The main part of Chapter 4 is concerned with our results on games with parity winning conditions.

Chapter 5 generalises the results obtained for games with parity winning conditions in Chapter 4 to games with MSO-definable winning conditions. Automata-theoretic methods are needed to establish the reduction.

In Chapter 6, we discuss the problem of deciding whether a game with MSO-definable winning conditions has a subgame perfect equilibrium within given payoff thresholds. We develop an algorithm for the problem, which can also be used to compute an optimal subgame perfect equilibrium. Finally, the last part of Chapter 6 deals with complexity issues.

Acknowledgements

I would like to thank Prof. Erich Grädel for his guidance while I worked on this thesis.

I would like to thank Prof. Wolfgang Thomas for his willingness to co-examine this thesis.

I am very grateful to Vince Bárány, Tobias Ganzow, Łukasz Kaiser, Wong Karianto and Christof Löding for proofreading previous drafts of this thesis and for their valuable comments and suggestions.

Special thanks go to Dietmar Berwanger for suggesting the topic of this thesis and for many inspiring discussions.

Chapter 2

Infinite Games

2.1 Preliminaries

We assume that the reader is familiar with the basic notions of set theory and formal language theory as presented in the textbooks [HH99] and [HU79, PP04], respectively. In the following, we recall some basic definitions of formal language theory to fix our notation.

For a set Σ, a *finite word of length* $n < \omega$ over Σ is a sequence $w = w(0)w(1)\dots w(n-1)$ with $w(i) \in \Sigma$ for all $0 \le i < n$. The length n of a word w is denoted by $|w|$. We write ε for the empty word (i.e. the unique word of length 0). Any $a \in \Sigma$ is identified with the word a of length 1. For $0 \le i \le j \le |w|$, $w[i,j]$ denotes the word $w(i)w(i+1)\dots w(j-1)$ (where $w[i,i]$ is the empty word). Σ^n denotes the set of all words of length n over Σ, and $\Sigma^* = \bigcup_{n<\omega} \Sigma^n$ denotes the set of all finite words over Σ.

An *infinite word* over Σ is an infinite sequence $\alpha = \alpha(0)\alpha(1)\dots$ with $\alpha(i) \in \Sigma$ for all $0 \le i < \omega$. For $0 \le i \le j < \omega$, $\alpha[i,j]$ is defined as in the case of finite words. The infinite word $\alpha(i)\alpha(i+1)\dots$ is denoted by $\alpha[i,\omega]$. By $\mathrm{Occ}(\alpha)$ we denote the set of all $a \in \Sigma$ such that $\alpha(i) = a$ for at least one $i < \omega$, and by $\mathrm{Inf}(\alpha)$ we denote the set of all $a \in \Sigma$ such that $\alpha(i) = a$ for infinitely many $i < \omega$. The set of all infinite words over Σ is denoted by Σ^ω.

For two words $v, w \in \Sigma^*$ of length m and n, respectively, their *concatenation* is the word $v{\cdot}w = v(0)\dots v(m{-}1)w(0)\dots w(n{-}1)$, which we also denote by vw for short. Analogously, the concatenation of a word w of length n with an infinite word α is the infinite word $w \cdot \alpha = w(0)\dots w(n-1)\alpha(0)\alpha(1)\dots$, which we also denote by $w\alpha$ for short. A finite word v is a *prefix* of a finite word w, written $v \preceq w$, if there exists a finite word u with $w = v \cdot u$. A finite word w is a *finite prefix* of an infinite word α, written $w \prec \alpha$, if there exists an infinite word β with $\alpha = v \cdot \beta$. If $V, W \subseteq \Sigma^*$ are sets of finite words, then $V \cdot W$ or VW for short is the set of all words $v \cdot w$ with $v \in V$ and $w \in V$. Analogously, if $W \subseteq \Sigma^*$ and $A \subseteq \Sigma^\omega$, then $W \cdot A$ or WA for short is the set of all words $w \cdot \alpha$ with $w \in W$ and $\alpha \in A$.

A Σ-*tree* is a prefix-closed, non-empty subset T of Σ^*, i.e. $T \neq \emptyset$ satisfies the condition that, if $w \in T$ and $v \preceq w$, then also $v \in T$. T is called *non-terminating* if for any $w \in T$ there exists $a \in \Sigma$ such that $wa \in T$.

2.2 Games

Games represent a model for interaction between decision makers. We study so-called *turn-based games in extensive form with perfect information* where the players have to make decisions again and again. Such a game consists of the following ingredients: a finite set of players, a tree describing which actions the players can take after any initial sequence of actions taken, a function determining for each initial sequence which player takes turn after this sequence and for every player a function giving for every possible play the resulting payoff for this player. For the sake of simplicity, we do not allow dead ends in the game tree.

Definition 2.1. An *infinite game* is a tuple $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (\mu_i)_{i\in\Pi})$ where

(1) Π is a non-empty, finite set of *players*,

(2) Σ is a non-empty set of *actions*,

(3) T is a non-terminating Σ-tree,

(4) λ is a function from T to Π, and

(5) μ_i is a function from Σ^ω to the interval $[0, 1]$ of real numbers.

The tree T is called the *game tree*, λ is called the *move function*, and μ_i is called the *payoff function of player i*. A sequence $h \in T$ is called a *history of $\mathcal{G}$*, and an infinite sequence $\pi \in \Sigma^\omega$ such that every finite prefix of π lies in T is called a *play of $\mathcal{G}$*. For a play π of $\mathcal{G}$, the number $\mu_i(\pi)$ is called the *payoff of π for player i*, and the tuple $(\mu_i(\pi))_{i\in\Pi}$ is called the *payoff of π*.

The idea of an infinite game is the following: Elements of the action set Σ correspond to primitive actions of the game. A play starts with the empty history ε. Then for any history $h \in T$, the player $\lambda(h)$ must choose an action $a \in \Sigma$ such that the resulting sequence ha is again an element of T. In this fashion, an infinite play evolves.

The interpretation of the payoffs is the following: If $\mu_i(\pi_1) \leq \mu_i(\pi_2)$ for two plays π_1 and π_2, then π_2 is at least as good as π_1 for player i.

Example 2.1. In the *prisoners' dilemma*, there are two prisoners 0 and 1 accused of the same crime. Both can either confess or remain silent. If both remain silent, they can only be convicted for a small crime. If one confesses and the other remains silent, the one who remains silent must take on all the burden and is sentenced to a long time in prison whereas the one who confesses is immediately released. Finally, if both confess, then both are sent to prison, but they do not need to stay as long as the one who remains silent if the other one confesses. The game is usually formulated as a so-called strategic game where the players choose actions simultaneously. To use our game model, we adopt the convention that prisoner 0 is the first to be questioned and that prisoner 1 is attendant when prisoner 0 makes his statement (as it would probably be the case in a joint trial against the two). Under this interpretation we refer to the game as the *sequential prisoners' dilemma.*

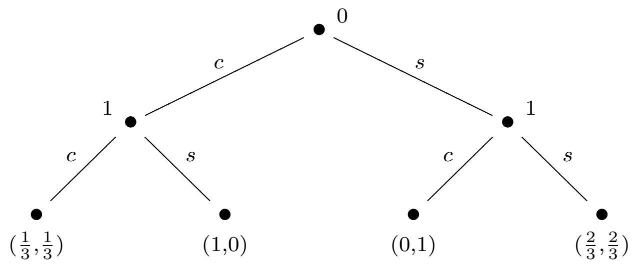

Figure 2.1: The sequential prisoners' dilemma as an "infinite game"

We model the game by an infinite game $\mathcal{G} = (\{0,1\}, \{c,s,x\}, T, \lambda, \mu_0, \mu_1)$ with $T = \{\varepsilon\} \cup \{c,s\} \cup \{c,s\} \cdot \{c,s\} \cdot \{x\}^*$ where

$$\lambda(h) = \begin{cases} 0 & \text{if } h = \varepsilon, \\ 1 & \text{otherwise} \end{cases}$$

for all $h \in T$ and

$$\mu_i(\pi) = \begin{cases} 1 & \text{if } \pi(i) = c \text{ and } \pi(1-i) = s, \\ \frac{2}{3} & \text{if } \pi(i) = s \text{ and } \pi(1-i) = s, \\ \frac{1}{3} & \text{if } \pi(i) = c \text{ and } \pi(1-i) = c, \\ 0 & \text{otherwise} \end{cases}$$

for all $\pi \in \{c,s,x\}^\omega$ and $i \in \Pi$. Obviously, the payoff of a play depends only on the first two actions. We just added "dummy moves" to end up with an infinite game. This gives rise to calling $\mathcal{G}$ a *finite game* despite its definition as an infinite game. Figure 2.1 shows the game tree where a node is labelled with the player who has to move and an edge is labelled with the corresponding action. We prune a subtree if all plays leading in this subtree have the same payoff and label its root with the respective payoff.

We call an infinite game discrete if a player can only lose (payoff 0) or win (payoff 1). An important subclass of discrete infinite games consists of all these games where every play has precisely one winner. We call these games zero-sum.

Definition 2.2. An infinite game $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (\mu_i)_{i\in\Pi})$ is called *discrete* if for each player $i \in \Pi$ we have $\mu_i(\Sigma^\omega) \subseteq \{0, 1\}$. The game $\mathcal{G}$ is called *zero-sum* if $\mathcal{G}$ is discrete and the sets $\mu_i^{-1}(1)$ of winning plays for player i define a partition of Σ^ω. For discrete games we write $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i\in\Pi})$ where $A_i = \mu_i^{-1}(1) \subseteq \Sigma^\omega$. We call the set A_i the *winning condition of player i*.

A strategy is a plan telling a player which action to choose after every history of the game where it is her turn.

Definition 2.3. For an infinite game $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (\mu_i)_{i\in\Pi})$, a *strategy of player i in $\mathcal{G}$* is a partial function $\sigma : T \to \Sigma$ such that

(1) $\sigma(h)$ is defined if and only if $\lambda(h) = i$, and

(2) $h\sigma(h) \in T$ if $\lambda(h) = i$

for all histories $h \in T$. A play π of $\mathcal{G}$ is *consistent with* σ if $\pi(k) = \sigma(\pi[0, k])$ for all $k < \omega$ with $\lambda(\pi[0, k]) = i$.

A *strategy profile of* $\mathcal{G}$ is a tuple $(\sigma_i)_{i\in\Pi}$ of strategies σ_i for each player $i \in \Pi$ in $\mathcal{G}$. A strategy profile $(\sigma_i)_{i\in\Pi}$ uniquely determines a play consistent with each σ_i which we will denote by $\langle(\sigma_i)_{i\in\Pi}\rangle$. This play is called *the outcome of* $(\sigma_i)_{i\in\Pi}$. The *payoff* of a strategy profile is the payoff of its outcome.

Example 2.2. In the game $\mathcal{G}$ from Example 2.1, player 0 has only two different strategies, namely to choose c or s in the beginning of the game. Player 1 has four different strategies, namely to choose c or s after player 0 has chosen c or s. A strategy profile (σ, τ) where σ is a strategy of player 0 and τ is a strategy of player 1 induces a play with payoff

(1) $(\frac{1}{3}, \frac{1}{3})$ if $\sigma(\varepsilon) = c$ and $\tau(s) = c$,

(2) $(1, 0)$ if $\sigma(\varepsilon) = c$ and $\tau(c) = s$,

(3) $(0, 1)$ if $\sigma(\varepsilon) = s$ and $\tau(s) = c$,

(4) $(\frac{2}{3}, \frac{2}{3})$ if $\sigma(\varepsilon) = s$ and $\tau(s) = s$.

The notion of games in extensive form is due to von Neumann and Morgenstern [vNM44] and Kuhn [Kuh50]. In game theory, a different model is widely used, namely the model of *strategic games*. In this model, whole strategies are taken as primitives (cf. [OR94] for a thorough discussion). However, as we are interested in the sequential structure of games, we will focus on infinite games and use game as a synonym for infinite game.

2.3 Equilibria

In the following, we assume that each player acts rationally, i.e. when choosing actions each player aims to maximise her payoff, and that each player has perfect information, i.e. when choosing an action each player knows about the complete history.

Nash [Nas50] introduced the notion of equilibria to model rational behaviour in games. In an equilibrium state, every player's strategy is optimal given the strategies of her opponents. Thus no player has a reason to change her strategy. Nash's notion of equilibria was originally formulated for strategic games, but it can be applied to games in extensive form as well.

Definition 2.4. For a game $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (\mu_i)_{i\in\Pi})$, a strategy profile $(\sigma_i)_{i\in\Pi}$ of $\mathcal{G}$ is called a *Nash equilibrium* if for all players $i \in \Pi$ we have

$$\mu_i(\langle\sigma', (\sigma_j)_{j\in\Pi\setminus\{i\}}\rangle) \leq \mu_i(\langle(\sigma_j)_{j\in\Pi}\rangle)$$

for all strategies σ' of player i in $\mathcal{G}$.

Example 2.3. The unique Nash equilibrium of the game from Example 2.1 is the strategy profile (σ, τ) where σ is the strategy of player 0 with $\sigma(\varepsilon) = c$ and τ is the strategy of player 1 such that $\tau(c) = \tau(s) = c$. (σ, τ) is a Nash equilibrium because, if either one player switches to action s instead of c, then this player will get payoff 0, which is not greater than $\frac{1}{3}$, the common payoff of the equilibrium. It remains to show that there are no other Nash equilibria. Let σ and τ be strategies of player 0 and player 1, respectively:

(1) If $\sigma(\varepsilon) = s$ and $\tau(s) = s$, then player 1 can improve her payoff to 1 by choosing action c instead of s after history s;

(2) If $\sigma(\varepsilon) = s$ and $\tau(s) = c$, then player 0 can improve her payoff to at least $\frac{1}{3}$ by choosing action c instead of s;

(3) If $\sigma(\varepsilon) = c$ and $\tau(c) = s$, then player 1 can improve her payoff to $\frac{1}{3}$ by choosing action c instead of s after history c;

(4) If $\sigma(\varepsilon) = c$, $\tau(c) = c$ and $\tau(s) = s$, then player 0 can improve her payoff to $\frac{2}{3}$ by choosing action s instead of c.

Thus (σ, τ) is a Nash equilibrium only if $\sigma(\varepsilon) = c$ and $\tau(c) = \tau(s) = c$, which proves the claim.

Note that in particular when applying the strategies in the Nash equilibrium both players receive a smaller payoff than if they both choose action s. However, the latter strategy profile is not rational because, if player 0 chooses s in the beginning, then player 1 will choose action c which gives her a payoff of 1 compared to $\frac{2}{3}$ if she chooses action c. Anticipating this behaviour, player 0 will choose action c in the first place to guarantee a payoff of $\frac{1}{3}$ compared to 0 if she chooses action s.

The following proposition shows that already for games with only one player we cannot guarantee the existence of a Nash equilibrium at least if we allow games with infinitely many payoffs. Note that a Nash equilibrium of a one-player game is just a strategy generating a play with a maximal payoff.

Proposition 2.5. *There is a one-player game that has no Nash equilibrium.*

Proof. Let $\mathcal{G} = (\{0\}, \{0,1\}, \{0,1\}^*, \lambda, \mu)$ be the game where $\lambda(h) = 0$ for all $h \in \{0,1\}^*$ and

$$\mu(\pi) = \begin{cases} 1 - \frac{1}{\min\{k<\omega : \pi(k)=0\}+1} & \text{if } \pi \neq 1^\omega, \\ 0 & \text{otherwise} \end{cases}$$

for all $\pi \in \{0,1\}^\omega$.

Assume (σ) is a Nash equilibrium of $\mathcal{G}$. Clearly, $\sigma(h) \neq 1$ for some $h \in \{1\}^*$ because otherwise $\mu(\langle\sigma\rangle) = 0$, but any strategy σ' with $\sigma'(h) = 0$ for some $h \in \{1\}^+$ gives a better payoff. Thus $n = \min\{k < \omega : \sigma(1^k) = 0\}$ exists and we have $\mu(\langle\sigma\rangle) = 1 - \frac{1}{n+1}$. But then any strategy σ' with $\sigma'(1^k) = 1$ for all $k \leq n$ but $\sigma'(1^{n+1}) = 0$ gives payoff $\mu(\langle\sigma'\rangle) = 1 - \frac{1}{n+2} > \mu(\langle\sigma\rangle)$, which yields a contradiction to the assumption that (σ) is a Nash equilibrium. □

On the other hand, any one-player game with only finitely many payoffs has a Nash equilibrium, as there exists a play with maximal payoff. However, already if we allow two players, we can construct a zero-sum game that has no Nash equilibrium. There are trivial examples for strategic games, where the players have incomplete information, but for our model of infinite games we rely on the axiom of choice to define a zero-sum game with no Nash equilibrium. We use a construction due to Gale and Stewart [GS53].

Proposition 2.6 (Gale, Stewart)**.** *There is a two-player zero-sum game that has no Nash equilibrium.*

Proof. Let $\Pi = \Sigma = \{0,1\}$, and $\lambda : \Sigma^* \to \Pi, h \mapsto |h| \mod 2$. By S_i we denote the set of partial functions from Σ^* to Σ which are precisely defined for every $h \in \Sigma^*$ with $\lambda(h) = i$. Clearly, an element σ of S_0 and an element τ of S_1 determine a unique infinite sequence over Σ, which we denote by $\langle\sigma, \tau\rangle$.

Let κ be the cardinality of S_0 and S_1 (obviously $|S_0| = |S_1|$). It is easy to show that $\kappa = |\{0,1\}^\omega| = 2^{\aleph_0}$. By Zermelo's well-ordering theorem, which is equivalent to the axiom of choice, there exists a well-ordering on this set which is isomorphic to $(\kappa, <)$. Thus we can enumerate the elements of S_i by all ordinals below κ, i.e. $S_i = \{\sigma_i^\alpha : \alpha < \kappa\}$.

To define the payoff function of our game, we define inductively the following sets. Let

$$L_0 = M_0 = \emptyset$$

and

$$L_\lambda = \bigcup_{\alpha<\lambda} L_\alpha, \; M_\lambda = \bigcup_{\alpha<\lambda} M_\alpha$$

if $\lambda \leq \kappa$ is a limit ordinal. To define $L_{\alpha+1}$ for some ordinal $\alpha < \kappa$, consider the function $\sigma_1^\alpha \in S_1$. Choose some $\sigma \in S_0$ such that $\langle \sigma, \sigma_1^\alpha \rangle$ is neither an element of L_α nor of M_α. This is possible since $|L_\alpha \cup M_\alpha| \leq 2 \cdot \alpha < \kappa$, but $|\{\langle \sigma, \sigma_1^\alpha \rangle : \sigma \in S_0\}| = \kappa$. We define

$$L_{\alpha+1} = L_\alpha \cup \{\langle \sigma, \sigma_1^\alpha \rangle\}.$$

Analogously, we choose some $\tau \in S_1$ such that $\langle \sigma_0^\alpha, \tau \rangle \notin L_{\alpha+1} \cup M_\alpha$ and define

$$M_{\alpha+1} = M_\alpha \cup \{\langle \sigma_0^\alpha, \tau \rangle\}.$$

Finally, let $A_0 = L_\kappa$ and $A_1 = \Sigma^\omega \setminus L_\kappa$. Obviously, the resulting game $\mathcal{G} = (\Pi, \Sigma, \Sigma^*, \lambda, A_0, A_1)$ is zero-sum and the set of strategies of player $i \in \Pi$ in $\mathcal{G}$ is precisely the set S_i.

We show that $\mathcal{G}$ has no Nash equilibrium. Towards a contradiction, assume that (σ, τ) with $\sigma \in S_0$ and $\tau \in S_1$ is a Nash equilibrium of $\mathcal{G}$. We only discuss the case that $\langle \sigma, \tau \rangle$ is won by player 0, i.e. $\mu_0(\langle \sigma, \tau \rangle) = 1$. As the game is zero-sum, this implies $\mu_1(\langle \sigma, \tau \rangle) = 0$. Now, we have that $\sigma \in S_0$ and thus $\sigma = \sigma_0^\alpha$ for some ordinal $\alpha < \kappa$. By definition of $M_{\alpha+1}$, there is some strategy τ' of player 1, such that $\langle \sigma, \tau' \rangle \in M_{\alpha+1}$. But then the play $\langle \sigma, \tau' \rangle$ is not an element of any of the sets L_β, hence $\langle \sigma, \tau' \rangle \notin L_\kappa$, which implies $\mu_1(\langle \sigma, \tau' \rangle) = 1$. But then, (σ, τ) is not a Nash equilibrium because player 1 can improve her payoff by choosing τ' instead of τ. The case that $\langle \sigma, \tau \rangle$ is won by player 1 is analogous. □

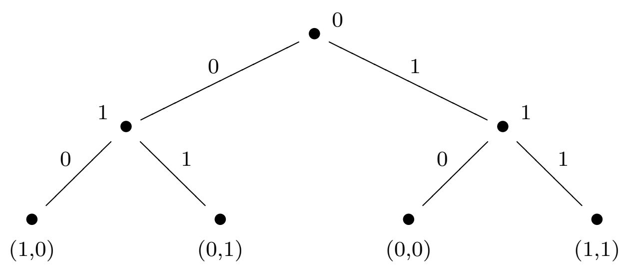

Figure 2.2: A game with an implausible Nash equilibrium.

Example 2.4. Consider the discrete game $\mathcal{G}$ with its pruned game tree depicted in Figure 2.2. For $a, b \in \{0,1\}$ let σ_a be a strategy of player 0 with $\sigma(\varepsilon) = a$ and let τ_{ab} be a strategy of player 1 such that $\tau_{ab}(0) = a$ and $\tau_{ab}(1) = b$. Clearly, (σ_0, τ_{10}) is a Nash equilibrium of $\mathcal{G}$. However, this equilibrium is not rational because, if player 0 chooses action 1 instead of 0, then she can be sure that player 1 will change her strategy and choose action 1 to receive payoff 1 instead of 0 if she chooses 0. But this also gives player 0 a better payoff, namely 1 instead of 0 if she chooses action 0.

The problem of using Nash equilibria for games in extensive form is that they ignore the sequential structure of the game, where it is possible to switch strategies during a play to get a better payoff. As we have seen in the last example, this may make some Nash equilibria implausible. Selten [Sel65] came up with an equilibrium notion which eliminates such equilibria.

Definition 2.7. For a game $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (\mu_i)_{i\in\Pi})$ and a history $h \in T$, the *subgame of* $\mathcal{G}$ *with history* h is the game $\mathcal{G}|_h = (\Pi, \Sigma, T|_h, \lambda|_h, (\mu_i|_h)_{i\in\Pi})$ defined by

(1) $T|_h = \{w \in \Sigma^* : hw \in T\}$,

(2) $\lambda|_h : T|_h \to \Pi : w \mapsto \lambda(hw)$, and

(3) $\mu_i|_h : \Sigma^\omega \to [0,1] : \pi \mapsto \mu_i(h\pi)$ for all $i \in \Pi$.

A strategy σ of player i in $\mathcal{G}$ induces a strategy $\sigma|_h$ of player i in $\mathcal{G}|_h$ defined by $\sigma|_h(w) = \sigma(hw)$ for all $w \in T|_h$.

Definition 2.8. For a game $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (\mu_i)_{i\in\Pi})$, a strategy profile $(\sigma_i)_{i\in\Pi}$ is called a *subgame perfect equilibrium* if for all players $i \in \Pi$ and all histories $h \in T$ we have

$$\mu_i|_h(\langle \sigma'|_h, (\sigma_j|_h)_{j\in\Pi\setminus\{i\}}\rangle) \leq \mu_i|_h(\langle (\sigma_j|_h)_{j\in\Pi}\rangle)$$

for all strategies σ' of player i in $\mathcal{G}$. Equivalently, $(\sigma_i)_{i\in\Pi}$ is a subgame perfect equilibrium of $\mathcal{G}$ if and only if $(\sigma_i|_h)_{i\in\Pi}$ is a Nash equilibrium of $\mathcal{G}|_h$ for all histories $h \in T$.

Example 2.5. Let $\mathcal{G}$ be the game from Example 2.4 together with the strategies σ_a and τ_{ab} for $a, b \in \{0,1\}$ of player 0 and player 1, respectively, as defined there. Then (σ_1, τ_{11}) is a subgame perfect equilibrium of $\mathcal{G}$, but (σ_0, τ_{10}) is not because, given history 1, the choice of action 0 is not optimal for player 1.

We show that there are games that have a Nash equilibrium but no subgame perfect equilibrium. To construct such a game, we just take a game $\mathcal{G}$, say with at least one player 0 and action set $\{0,1\}$, that has no Nash equilibrium and use it as the subgame with history 1. As the subgame with history 0, we take a game in which all plays have payoff 1 for player 0 and payoff 0 for all other players. At the empty history, it is player 0's turn. Then any strategy profile where player 0 has a strategy σ with $\sigma(\varepsilon) = 0$ is a (plausible) Nash equilibrium but, as the subgame with history 1 has no Nash equilibrium at all, this game has no subgame perfect equilibrium. Using the games constructed in Proposition 2.5 and 2.6, we get the following propositions.

Proposition 2.9. *There is a one-player game that has a Nash equilibrium but no subgame perfect equilibrium.*

Proposition 2.10. *There is a two-player zero-sum game that has a Nash equilibrium but no subgame perfect equilibrium.*

One may interpret these propositions as an indication that the notion of a subgame perfect equilibrium is too strong. However, note that the games constructed to show the two propositions are very near to having no Nash equilibrium at all, i.e. if we change the payoff of the plays in the subgame with history 0 to $(1-\varepsilon, 0)$ for any $\varepsilon > 0$, the resulting game has no Nash equilibrium. Thus, on the one hand, the notion of a subgame perfect equilibrium is not much stronger than the notion of a Nash equilibrium, but on the other hand, it is strong enough to exclude implausible equilibria.

2.4 Determinacy

Intuitively, a game is determined if the game admits rational behaviour and every play of the game that emerges from rational behaviour has the same payoff. We distinguish two notions of determinacy depending on the underlying equilibrium notion.

Definition 2.11. A game $\mathcal{G}$ is called *Nash determined* if $\mathcal{G}$ has a Nash equilibrium and any two Nash equilibria of $\mathcal{G}$ have the same payoff. $\mathcal{G}$ is called *subgame perfect determined* if $\mathcal{G}$ has a subgame perfect equilibrium and any two subgame perfect equilibria of $\mathcal{G}$ have the same payoff.

Obviously, if a game is Nash determined and it has at least one subgame perfect equilibrium, then it is also subgame perfect determined. The converse does not hold as demonstrated by the following example.

Example 2.6. The game from Example 2.4 is subgame perfect determined but not Nash determined, as they are two Nash equilibria with different payoffs, whereas the sequential prisoners' dilemma (see Example 2.1) is both Nash and subgame perfect determined[1]. Now consider the game depicted in Figure 2.3. For $a \in \{0, 1\}$, let σ_a be a strategy of player 0 with $\sigma(\varepsilon) = a$

[1]Note that the Nash equilibrium where both players choose to confess is also a subgame perfect equilibrium.

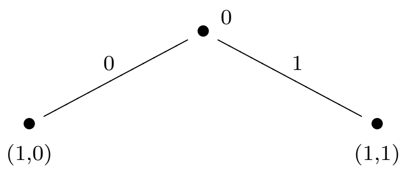

Figure 2.3: A non-determined game.

and let τ be an arbitrary strategy of player 1. Then (σ_0, τ) and (σ_1, τ) are both subgame perfect equilibria of the game. As the two equilibria have a different payoff, this game is neither Nash nor subgame perfect determined.

For two-player zero-sum games, determinacy is usually formulated in terms of the existence of a winning strategy (see [GS53]).

Definition 2.12. For a discrete game $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i\in\Pi})$, a strategy σ of player $i \in \Pi$ is called a *winning strategy* if all plays π consistent with σ are won by player i. A two-player zero-sum game $\mathcal{G}$ is called *determined* if one of the two players has a winning strategy in $\mathcal{G}$.

Obviously, in a zero-sum game at most one player can have a winning strategy, but in general we cannot rule out that none of the players has a winning strategy. We show that a two-player zero-sum game is determined if and only if it is Nash determined in the sense of Definition 2.11 if and only if the game has at least one Nash equilibrium. Thus, by showing that there exists a two-player zero-sum game with no Nash equilibrium (see Proposition 2.6), we have shown that there exists a non-determined two-player zero-sum game. Note that for discrete games $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i\in\Pi})$ we can rephrase the definition of Nash equilibrium as follows: $(\sigma_i)_{i\in\Pi}$ is a Nash equilibrium of $\mathcal{G}$ if and only if for all players $i \in \Pi$ and every strategy σ' of player i in $\mathcal{G}$ we have that $\langle \sigma', (\sigma_j)_{j\in\Pi\setminus\{i\}} \rangle \in A_i$ only if $\langle (\sigma_j)_{j\in\Pi} \rangle \in A_i$.

In the following, when dealing with two-player zero-sum games, we will use $\{0, 1\}$ as the set of players.

Lemma 2.13. *In a two-player zero-sum game, any two Nash equilibria have the same payoff.*

Proof. Let $\mathcal{G} = (\{0, 1\}, \Sigma, T, \lambda, A_0, A_1)$ and let $(\sigma, \tau), (\sigma', \tau')$ be two Nash equilibria of $\mathcal{G}$ with different payoffs. Without loss of generality we can assume that $\langle \sigma, \tau \rangle \in A_0$ and $\langle \sigma', \tau' \rangle \in A_1$. As (σ, τ) is a Nash equilibrium, we have $\langle \sigma, \tau' \rangle \notin A_1$. On the other hand, as (σ', τ') is a Nash equilibrium, we have $\langle \sigma, \tau' \rangle \notin A_0$. Thus $\langle \sigma, \tau' \rangle$ is not won by any player, a contradiction to the fact that $\mathcal{G}$ is zero-sum. □

Lemma 2.14. *For a two-player zero-sum game $\mathcal{G}$ and strategies σ and τ of player 0 and player 1, respectively, in $\mathcal{G}$, the following holds: (σ, τ) is a Nash equilibrium if and only if either σ or τ is winning.*

Proof. ($\Rightarrow$) Assume (σ, τ) is a Nash equilibrium of $\mathcal{G}$. We only discuss the case that $\langle \sigma, \tau \rangle$ is won by player 0. We show that σ is a winning strategy of player 0. Otherwise there would exist a strategy τ' of player 1 such that $\langle \sigma, \tau' \rangle$ is won by player 1. But then (σ, τ) is not a Nash equilibrium, as player 1 can choose τ' instead of τ to get a play won by her. The case that $\langle \sigma, \tau \rangle$ is won by player 1 is analogous.

($\Leftarrow$) Assume without loss of generality that σ is winning. In this case, for any strategy τ' of player 1, we have that $\langle \sigma, \tau' \rangle$ is won by player 0 and thus not won by player 1. Hence, (σ, τ) is a Nash equilibrium of $\mathcal{G}$. □

Theorem 2.15. *For any two-player zero-sum game $\mathcal{G}$, the following statements are equivalent:*

(1) $\mathcal{G}$ is determined.

(2) $\mathcal{G}$ has a Nash equilibrium.

(3) $\mathcal{G}$ is Nash determined.

Proof. (1) $\Rightarrow$ (2): Without loss of generality, we can assume that player 0 has a winning strategy σ in $\mathcal{G}$. Take any strategy τ of player 1 in $\mathcal{G}$. By Lemma 2.14, (σ, τ) is a Nash equilibrium of $\mathcal{G}$.

(2) $\Rightarrow$ (3): This is an immediate consequence of Lemma 2.13.

(3) $\Rightarrow$ (1): Take any Nash equilibrium of $\mathcal{G}$. By Lemma 2.14, one of the two strategies in the equilibrium is winning. □

Our next aim is to give a characterisation of subgame perfect determinacy for two-player zero-sum games. We show that a two-player zero-sum game is subgame perfect determined if and only if every subgame of the game is determined. As the payoff of a Nash equilibrium of a two-player zero-sum game is unique, this is also equivalent to the existence of a subgame perfect equilibrium.

Lemma 2.16. *For any discrete game $\mathcal{G}$ and strategies σ_h of some player i in $\mathcal{G}|_h$ for each history h of $\mathcal{G}$, there exists a strategy σ of player i in $\mathcal{G}$ such that $\sigma|_h$ is winning in $\mathcal{G}|_h$ if σ_h is winning in $\mathcal{G}|_h$.*

Proof. Let $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i \in \Pi})$. Without loss of generality, we can assume that for all $h \in T$ and $a \in \Sigma$ with $ha \in T$, if σ_h is winning and either $\sigma_h(\varepsilon) = a$ or $\lambda(h) \neq i$, then $\sigma_{ha} = \sigma_h|_a$ because, if σ_h is winning in $\mathcal{G}|_h$ and either $\sigma_h(\varepsilon) = a$ or $\lambda(h) \neq i$, then $\sigma_h|_a$ is winning in $\mathcal{G}|_{ha}$.

We define σ by $\sigma(h) = \sigma_h(\varepsilon)$ for all $h \in T$. Clearly, σ is a strategy of player i in $\mathcal{G}$. Let h be any history of $\mathcal{G}$ such that σ_h is winning in $\mathcal{G}|_h$, and let π be a play of $\mathcal{G}|_h$ consistent with $\sigma|_h$. By induction on k, we show that for all $k < \omega$ the following holds:

(1) $\sigma_{h \cdot \pi[0,k]}$ is winning in $\mathcal{G}|_{h \cdot \pi[0,k]}$ and

(2) $\sigma_{h\cdot\pi[0,k]} = \sigma_h|_{\pi[0,k]}$.

For $k = 0$, we have $\sigma_{h\cdot\pi[0,k]} = \sigma_h$ which is winning by assumption. For $k > 0$, $\sigma_{h\cdot\pi[0,k-1]}$ is winning by induction hypothesis. If $\lambda(\pi[0,k-1]) = i$, we have that $\pi(k-1) = \sigma|_h(\pi[0,k-1]) = \sigma(h\cdot\pi[0,k-1]) = \sigma_{h\cdot\pi[0,k-1]}(\varepsilon)$. Thus, in any case, $\sigma_{h\cdot\pi[0,k]} = \sigma_{h\cdot\pi[0,k-1]}|_{\pi(k-1)}$ (see above), which is again winning. By induction hypothesis, we get $\sigma_{h\cdot\pi[0,k-1]} = \sigma_h|_{\pi[0,k-1]}$. Thus, $\sigma_{h\cdot\pi[0,k]} = \sigma_{h\cdot\pi[0,k-1]}|_{\pi(k-1)} = \sigma_h|_{\pi[0,k-1]}|_{\pi(k-1)} = \sigma_h|_{\pi[0,k]}$, which proves the claim.

Now, as π is consistent with $\sigma|_h$, for any $k < \omega$, if $\lambda(\pi[0,k]) = i$, then $\pi(k) = \sigma|_h(\pi[0,k]) = \sigma(h\cdot\pi[0,k]) = \sigma_{h\cdot\pi[0,k]}(\varepsilon) = \sigma_h|_{\pi[0,k]}(\varepsilon) = \sigma_h(\pi[0,k])$. Thus, π is also consistent with σ_h and therefore won by player i. Hence, $\sigma|_h$ is winning in $\mathcal{G}|_h$. □

Theorem 2.17. *For any two-player zero-sum game $\mathcal{G}$, the following statements are equivalent:*

(1) $\mathcal{G}|_h$ is determined for all histories h of $\mathcal{G}$.

(2) $\mathcal{G}$ has a subgame perfect equilibrium.

(3) $\mathcal{G}$ is subgame perfect determined.

Proof. (1) ⇒ (2): By Lemma 2.16, there exist strategies σ and τ of player 0 and player 1, respectively, such that for all histories h of $\mathcal{G}$ either $\sigma|_h$ or $\tau|_h$ is winning in $\mathcal{G}|_h$. Thus, by Lemma 2.14, $(\sigma|_h, \tau|_h)$ is a Nash equilibrium of $\mathcal{G}|_h$ for all histories h of $\mathcal{G}$, i.e. (σ,τ) is a subgame perfect equilibrium of $\mathcal{G}$.

(2) ⇒ (3): This is an immediate consequence of Lemma 2.13.

(3) ⇒ (1): If $\mathcal{G}$ has a subgame perfect equilibrium (σ,τ), then $(\sigma|_h, \tau|_h)$ is a Nash equilibrium of $\mathcal{G}|_h$ for all histories h of $\mathcal{G}$. Thus by Lemma 2.14, either $\sigma|_h$ or $\tau|_h$ is winning in $\mathcal{G}|_h$ for all histories h of $\mathcal{G}$. □

Theorem 2.17 together with Theorem 2.15 shows that, for two-player zero-sum games, the existence of a Nash equilibrium in each subgame implies the existence of a subgame perfect equilibrium. For infinite games with more than two players or non-zero-sum games, this does not seem to be the case.

Chapter 3

Borel Games

In the theory of infinite two-player zero-sum games, the class of Borel games is a class of games with "well-behaved" winning conditions. In particular, every Borel game is determined. The definition of Borel games generalises naturally to non-zero-sum games with arbitrarily many players. We will show that every game of this class admits rational behaviour, i.e. every such game has a subgame perfect equilibrium. The classification is based on topology.

3.1 Topology

For any set Σ, the set Σ^ω of infinite words over Σ is equipped with the topology whose open sets are of the form $W \cdot \Sigma^\omega$ for $W \subseteq \Sigma^*$. The class of Borel sets is the closure of open sets under countable unions and complementation. There is a natural hierarchy measuring the complexity of a Borel set by counting the number of operations "complementation" and "countable unions" needed to create the set. This hierarchy is called the Borel hierarchy.

Definition 3.1. A set $A \subseteq \Sigma^\omega$ is called *open* if $A = W \cdot \Sigma^\omega$ for some $W \subseteq \Sigma^*$, *closed* if the set $\Sigma^\omega \setminus A$ is open and *clopen* if A is both open and closed. The *Borel hierarchy over* Σ^ω is inductively defined by

(1) $\Sigma^0_1(\Sigma^\omega) = \{A \subseteq \Sigma^\omega : A \text{ is open}\}$,

(2) $\Pi^0_\alpha(\Sigma^\omega) = \{\Sigma^\omega \setminus A : A \in \Sigma^0_\alpha(\Sigma^\omega)\}$,

(3) $\Sigma^0_\alpha(\Sigma^\omega) = \{\bigcup_{n<\omega} A_n : A_n \in \bigcup_{\beta<\alpha} \Pi^0_\beta(\Sigma^\omega)\}$ for $\alpha > 1$, and

(4) $\Delta^0_\alpha(\Sigma^\omega) = \Sigma^0_\alpha(\Sigma^\omega) \cap \Pi^0_\alpha(\Sigma^\omega)$,

where α ranges over all countable ordinals. A set $A \subseteq \Sigma^\omega$ is called a *Borel set* if $A \in \Sigma^0_\alpha(\Sigma^\omega)$ for some α.

See [Mos80] for facts about the Borel hierarchy. In particular, we note that for $|\Sigma| \geq 2$ the Borel hierarchy is indeed a strict hierarchy, i.e. for every countable ordinal α we have $\Delta^0_\alpha(\Sigma^\omega) \subsetneq \Sigma^0_\alpha(\Sigma^\omega) \subsetneq \Delta^0_{\alpha+1}(\Sigma^\omega)$ and $\Delta^0_\alpha(\Sigma^\omega) \subsetneq \Pi^0_\alpha(\Sigma^\omega) \subsetneq \Delta^0_{\alpha+1}(\Sigma^\omega)$.

Definition 3.2. A function $f : \Sigma^\omega \to \Gamma^\omega$ is called *continuous* if $f^{-1}(B)$ is open for every open set $B \subseteq \Gamma^\omega$. We say that f is a *continuous reduction* from a set $A \subseteq \Sigma^\omega$ to a set $B \subseteq \Gamma^\omega$, written $f : A \leq B$, if f is continuous and $f^{-1}(B) = A$, i.e. for all $\alpha \in \Sigma^\omega$ we have $\alpha \in A \Leftrightarrow f(\alpha) \in B$. Moreover we write $A \leq B$ if $f : A \leq B$ for some f.

As made precise by the following lemma, continuous reductions are useful for comparing the complexity of two sets of infinite words.

Lemma 3.3. *Let $A \in \Sigma^\omega$ and $B \in \Gamma^\omega$. If $A \leq B$ and $B \in \Sigma^0_\alpha(\Gamma^\omega)$ or $B \in \Pi^0_\alpha(\Gamma^\omega)$, then $A \in \Sigma^0_\alpha(\Sigma^\omega)$ or $A \in \Pi^0_\alpha(\Sigma^\omega)$, respectively.*

Proof. First note that if $f : A \leq B$ and $B \in \Pi^0_\alpha(\Gamma^\omega)$, then $\Gamma^\omega \setminus B \in \Sigma^0_\alpha(\Gamma^\omega)$ and $f : \Sigma^\omega \setminus A \leq \Gamma^\omega \setminus B$, thus the claim for Π^0_α follows from the claim for Σ^0_α. We show the claim for Σ^0_α by induction on α. The claim is trivial for $\alpha = 0$. Thus assume $B \in \Sigma^0_\alpha(\Gamma^\omega)$ for $\alpha > 0$, i.e. $B = \bigcup_{n<\omega} B_n$ with $B_n \in \bigcup_{\beta<\alpha} \Pi^0_\beta(\Gamma^\omega)$, and $f : A \leq B$. Let $A_n := f^{-1}(B_n)$, thus $f : A_n \leq B_n$ and $A_n \in \bigcup_{\beta<\alpha} \Pi^0_\beta(\Sigma^\omega)$ by the induction hypothesis. Now, for any $\alpha \in \Sigma^\omega$, $\alpha \in A$ if and only if $f(\alpha) \in B$ if and only if $f(\alpha) \in B_n$ for some n if and only if $\alpha \in A_n$ for some n. Hence $A = \bigcup_{n<\omega} A_n \in \Sigma^0_\alpha(\Sigma^\omega)$. □

In the following, if the alphabet Σ under consideration is clear from the context, we just write Σ^0_α, Π^0_α and Δ^0_α instead of $\Sigma^0_\alpha(\Sigma^\omega)$, $\Pi^0_\alpha(\Sigma^\omega)$ and $\Delta^0_\alpha(\Sigma^\omega)$, respectively.

3.2 Borel Games

In the theory of two-player zero-sum games, a Borel game is a game where the winning condition of player 0 is a Borel set (which implies that the same holds for the winning condition of player 1). This generalises naturally to our model of discrete infinite games.

Definition 3.4. A discrete game $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i\in\Pi})$ is called a *Borel game* if for all $i \in \Pi$ the set A_i is a Borel set. *Open games, closed games, clopen games, Σ^0_α-games, Π^0_α-games and Δ^0_α-games* are defined analogously.

Note that, if $\mathcal{G}$ is a zero-sum game, then $\mathcal{G}$ is a Σ^0_α-game if and only if $\mathcal{G}$ is a Π^0_α-game if and only if $\mathcal{G}$ is a Δ^0_α-game. Here our notation differs from the the usual notation for two-player zero-sum games where the game is called a Σ^0_α-game if the winning condition of player 0 is in Σ^0_α (which does not imply that the winning condition of player 1 is also in Σ^0_α).

Example 3.1. A game $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (\mu_i)_{i \in \Pi})$ is called *finite* if the payoff of any play π depends only on a finite prefix of π, i.e. for every $\pi \in \Sigma^\omega$ there exists a finite prefix h of π such that for all players $i \in \Pi$ and for all $\pi' \in \Sigma^\omega$ with $h \prec \pi'$ we have $\mu(\pi) = \mu(\pi')$. We show that discrete finite games correspond precisely to clopen games.

Let $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i \in \Pi})$ be a finite discrete game. For any $\pi \in \Sigma^\omega$ fix a finite prefix h_π such that the payoff of π only depends on h_π and let $i \in \Pi$. Then we have $A_i = \{h_\pi : \pi \in A_i\} \cdot \Sigma^\omega$ and $\Sigma^\omega \setminus A_i = \{h_\pi : \pi \in \Sigma^\omega \setminus A_i\} \cdot \Sigma^\omega$. Hence A_i is clopen.

Now let $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i \in \Pi})$ be a clopen game, say $A_i = W_i \cdot \Sigma^\omega$ and $\Sigma^\omega \setminus A_i = V_i \cdot \Sigma^\omega$ with $W_i, V_i \subseteq \Sigma^*$ for any player $i \in \Pi$, and let $\pi \in \Sigma^\omega$. For each player $i \in \Pi$, fix $k_i < \omega$ such that $\pi[0, k_i] \in W_i$ if $\pi \in A_i$ and $\pi[0, k_i] \in V_i$ if $\pi \notin A_i$. Let $k = \max\{k_i : i \in \Pi\}$. Then for any player $i \in \Pi$ we have that $\pi[0, k] \in W_i \cdot \Sigma^*$ or $\pi[0, k] \in V_i \cdot \Sigma^*$. Hence the payoff of π depends only on $\pi[0, k]$.

3.3 Determinacy of Borel Games

There is a long history of determinacy result for two-player zero-sum Borel games. Already in 1913, Zermelo [Zer13] observed that finite two-player zero-sum games are determined. This was then extended to higher levels of the Borel hierarchy until Martin [Mar75] succeeded in showing that all two-player zero-sum Borel games are determined[1]. By Theorem 2.15, this is equivalent to the existence of Nash equilibria in these games.

Theorem 3.5 (Martin)**.** *Two-player zero-sum Borel games are determined.*

For an inductive proof of Martin's theorem, see [Mar85]. As any subgame of a Borel game is again a Borel game, Martin's theorem implies that two-player zero-sum games are subgame perfect determined.

Lemma 3.6. *If $\mathcal{G}$ is a Borel game, then $\mathcal{G}|_h$ is a Borel game for every history h of $\mathcal{G}$.*

Proof. For $A \subseteq \Sigma^\omega$ and $w \in \Sigma^*$, let $A|_w := \{\pi \in \Sigma^\omega : w\pi \in A\}$. By induction on α, we show that $A|_w \in \Sigma^0_\alpha$ (Π^0_α) if $A \in \Sigma^0_\alpha$ (Π^0_α). First note that if $A \in \Pi^0_\alpha$ then $\Sigma^\omega \setminus A \in \Sigma^0_\alpha$ and $\Sigma^\omega \setminus A|_w = (\Sigma^\omega \setminus A)|_w$. Thus the claim for $\Pi^0 \alpha$ follows from the claim for Σ^0_α.

If $A \in \Sigma^0_0$, then $A = W \cdot \Sigma^\omega$ for some $W \subseteq \Sigma^*$. Without loss of generality, we can assume that $|v| \geq |w|$ for all $v \in W$. Then we have $A|_w = \{\pi \in \Sigma^\omega : w\pi \in A\} = \{v \in \Sigma^* : wv \in W\} \cdot \Sigma^\omega \in \Sigma^0_0$.

If $A \in \Sigma^0_\alpha$ for $\alpha > 0$, then $A = \bigcup_{n<\omega} A_n$ for sets $A_n \in \bigcup_{\beta<\alpha} \Pi^0_\beta$. By the induction hypothesis, we have $A_n|_w \in \bigcup_{\beta<\alpha} \Pi^0_\beta$ for all $n < \omega$. Thus $A|_w = \{\pi \in \Sigma^\omega : w\pi \in \bigcup_{n<\omega} A_n\} = \bigcup_{n<\omega} \{\pi \in \Sigma^\omega : w\pi \in A_n\} = \bigcup_{n<\omega} A_n|_w \in \Sigma^0_\alpha$.

[1] See [Gur89] for a survey on determinacy results for Borel games.

Now let $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i\in\Pi})$ be a Borel game and $h \in T$. Then $\mathcal{G}|_h = (\Pi, \Sigma, T, \lambda|_h, (A_i|_h)_{i\in\Pi})$. By what we have just shown, as A_i is Borel for all $i \in \Pi$, so is $A_i|_h$ for all $i \in \Pi$. Hence $\mathcal{G}|_h$ is a Borel game. □

Corollary 3.7. *Two-player zero-sum Borel games are subgame perfect determined.*

Proof. Let $\mathcal{G}$ be a two-player zero-sum Borel game. By Lemma 3.6 and Theorem 3.5, $\mathcal{G}|_h$ is determined for every history h of $\mathcal{G}$. By Theorem 2.15, this implies that $\mathcal{G}$ is subgame perfect determined. □

3.4 Existence of Nash Equilibria

In general, Borel games are not Nash or subgame perfect determined (see Example 2.6 for a finite game that is neither Nash nor subgame perfect determined), but we will show that every Borel game has a Nash equilibrium. The proof is based on [CJM04] and relies on an idea from repeated games where a player is "punished" for deviating from the equilibrium play (see for example [OR94, Chapter 8]).

Theorem 3.8. *Every Borel game has a Nash equilibrium.*

Proof. Let $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i\in\Pi})$ be a Borel game. For each player $i \in \Pi$ let $\mathcal{G}_i = (\{i, \Pi \setminus \{i\}\}, \Sigma, T, \lambda_i, A_i, \Sigma^\omega \setminus A_i)$ be the two-player zero-sum game where player i plays against the coalition $\Pi \setminus \{i\}$ in $\mathcal{G}$, i.e. we have $\lambda_i(h) = i$ if $\lambda(h) = i$ and $\lambda_i(h) = \Pi \setminus \{i\}$ otherwise. As the complement of a Borel set is again a Borel set, these games are Borel games. By Corollary 3.7, each game $\mathcal{G}_i$ is subgame perfect determined. Fix a strategy σ_i of player i and a strategy $\sigma_{\Pi\setminus\{i\}}$ of the coalition such that $(\sigma_i, \sigma_{\Pi\setminus\{i\}})$ is a subgame perfect equilibrium of $\mathcal{G}_i$. For each player $j \in \Pi \setminus \{i\}$, the strategy $\sigma_{\Pi\setminus\{i\}}$ induces a strategy $\sigma_{j,i}$ in $\mathcal{G}$ by restricting its domain to histories h with $\lambda(h) = j$.

In the equilibrium, every player j will use her strategy σ_j as long as no player deviates. If one player i deviates after history h, then every player $j \neq i$ must switch to her counter-strategy $\sigma_{j,i}$ to prevent a play won by player i. Inductively, we define for each history $h \in T$ the player $p(h)$ who has to be "punished" after history h where $p(h) = \bot$ if nobody has to be punished. In the beginning of the game, no player should be punished. Thus we let

$$p(\varepsilon) = \bot.$$

After history ha, the same player has to be punished as after history h as long as player $\lambda(h)$ does not deviate from her prescribed action. Thus for

$h \in T$ and $a \in \Sigma$ with $ha \in T$ we define

$$p(ha) = \begin{cases} \bot & \text{if } p(h) = \bot \text{ and } a = \sigma_{\lambda(h)}(h), \\ p(h) & \text{if } \lambda(h) \neq p(h),\, p(h) \neq \bot \text{ and } a = \sigma_{\lambda(h),p(h)}(h), \\ \lambda(h) & \text{otherwise.} \end{cases}$$

Now, for every player $j \in \Pi$ we define her equilibrium strategy τ_j by

$$\tau_j(h) = \begin{cases} \sigma_j(h) & \text{if } p(h) = \bot \text{ or } p(h) = j, \\ \sigma_{j,p(h)}(h) & \text{otherwise} \end{cases}$$

for all histories $h \in T$.

We show that $(\tau_j)_{j\in\Pi}$ is a Nash equilibrium of $\mathcal{G}$. Consider any strategy τ' of any player $i \in \Pi$, and let $\pi = \langle(\tau_j)_{j\in\Pi}\rangle = \langle(\sigma_j)_{j\in\Pi}\rangle$ and $\pi' = \langle\tau', (\tau_j)_{j\in\Pi\setminus\{i\}}\rangle$. We have to show that $\pi' \in A_i$ implies $\pi \in A_i$. The claim is trivial if $\pi = \pi'$. Hence assume $\pi \neq \pi'$ and let $k < \omega$ be minimal such that $\pi(k) \neq \pi'(k)$. Clearly, $\lambda(\pi[0,k]) = i$ and $p(\pi'[0,l]) = i$ for all $k < l < \omega$. Hence, by the definition of τ_j, every player $j \neq i$ will apply her strategy $\sigma_{j,i}$ for any history with prefix $\pi[0,k]\pi'(k)$. If $\pi' \in A_i$, then $\sigma_{\Pi\setminus\{i\}}|_{\pi[0,k]\pi'(k)}$ is not winning in $\mathcal{G}_i|_{\pi[0,k]\pi'(k)}$. But this implies that $\sigma_i|_{\pi[0,k]\pi'(k)}$ is winning. As $\lambda(\pi[0,k]) = i$, $\sigma_i|_{\pi[0,k]}$ must also be winning and hence $\pi \in A_i$. □

The equilibrium constructed in the proof of Theorem 3.8 is in general not a subgame perfect equilibrium. To see this, consider a subgame $\mathcal{G}|_h$ where h is a history that is not a prefix of the equilibrium play. Then every player $j \neq i$ has to punish some player i who has deviated by applying their strategies $\sigma_{j,i}$. It may be the case that the resulting play is not won by some player j although $\sigma_j|_h$ is winning in $\mathcal{G}|_h$. In this case, $(\tau_i|_h)_{i\in\Pi}$ is not a Nash equilibrium of $\mathcal{G}|_h$ because player j can use $\sigma_j|_h$ to win the game.

3.5 Existence of Subgame Perfect Equilibria

We will now refine the proof of Theorem 3.8 in order to show the existence of subgame perfect equilibria in Borel games. As in the previous proof, if some player has a winning strategy in a subgame, then she should play according to such a strategy. Note that this leads to a new game where all moves that are excluded by the respective winning strategies are eliminated. In this pruned game, players may have a winning strategy in a subgame for which they did not have a winning strategy before. Again, these players should play according to one of their winning strategies, which leads to more moves being eliminated. We do this again and again until we reach a fixed point. We show that, if all players apply their winning strategies in the resulting pruned game and react on deviation with their counter-strategies in the pruned game, then this results in a subgame perfect equilibrium of the original game.

Definition 3.9. Let $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i\in\Pi})$ be a discrete game. We interpret a set $V \subseteq T \times \Sigma$ such that for all $h \in T$ there exists at least one action $a \in \Sigma$ with $(h, a) \in V$ and $ha \in T$ as the set of legal moves in $\mathcal{G}$. For every $h \in T$, V defines the game $\mathcal{G}|_h^V = (\Pi, \Sigma, T|_h^V, \lambda|_h^V, (A_i|_h)_{i\in\Pi})$ where $T|_h^V \subseteq \Sigma^*$ is defined inductively by

(1) $\varepsilon \in T|_h^V$ and

(2) $wa \in T|_h^V$ if $w \in T|_h^V$, $wa \in T|_h$ and $(hw, a) \in V$

and $\lambda|_h^V$ is the restriction of $\lambda|_h$ to $T|_h^V$, i.e. $\mathcal{G}|_h^V$ is the subgame of $\mathcal{G}$ with history h where all illegal moves have been eliminated. As every subgame of a Borel game is a Borel game, each $\mathcal{G}|_h^V$ is a Borel game if $\mathcal{G}$ is a Borel game. We call a strategy σ of player i in $\mathcal{G}$ *V-compliant* if $(h, \sigma(h)) \in V$ for all $h \in T$ with $\lambda(h) = i$. If σ is V-compliant, then let $\sigma|_h^V$ be the induced strategy of player i in $\mathcal{G}|_h^V$, i.e. $\sigma|_h^V$ is $\sigma|_h$ restricted to $T|_h^V$.

Theorem 3.10. *Every Borel game has a subgame perfect equilibrium.*

Proof. Let $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i\in\Pi})$ be a Borel game. Inductively, we define the set V of legal moves in $\mathcal{G}$. In the beginning, we allow any action, i.e.

$$V^0 = T \times \Sigma.$$

If λ is a limit ordinal, we let

$$V^\lambda = \bigcap_{\alpha<\lambda} V^\alpha.$$

To define $V^{\alpha+1}$ for any ordinal α, for each player $i \in \Pi$ we consider the two-player zero-sum Borel game $\mathcal{G}_i$ as defined in the proof of Theorem 3.8. By Corollary 3.7, for every history $h \in T$ the game $\mathcal{G}_i|_h^{V^\alpha}$, which contains only the moves allowed by V^α, has a subgame perfect equilibrium. As a subgame perfect equilibrium induces a subgame perfect equilibrium for every subgame, there exists a V^α-compliant strategy σ_i^α of player i and a V^α-compliant strategy $\sigma_{\Pi\setminus\{i\}}^\alpha$ of the coalition in $\mathcal{G}_i$ such that $(\sigma_i^\alpha|_h^{V^\alpha}, \sigma_{\Pi\setminus\{i\}}^\alpha|_h^{V^\alpha})$ is a subgame perfect equilibrium of $\mathcal{G}_i|_h^{V^\alpha}$. For each player $j \in \Pi \setminus \{i\}$, the strategy $\sigma_{\Pi\setminus\{i\}}^\alpha$ induces a strategy $\sigma_{j,i}^\alpha$ of player j in $\mathcal{G}$ by restricting its domain to histories h with $\lambda(h) = j$. We define

$$V^{\alpha+1} = \{(h, a) \in V^\alpha : \sigma_{\lambda(h)}^\alpha|_h^{V^\alpha} \text{ is not winning or } a = \sigma_{\lambda(h)}^\alpha(h)\}.$$

The sequence $(V^\alpha)_{\alpha\in\mathrm{On}}$ is obviously non-increasing. Thus fix the least ordinal ξ with $V^\xi = V^{\xi+1}$ and let $V = V^\xi$. Moreover, for each player $i \in \Pi$ let $\sigma_i = \sigma_i^\xi$. For all players $j \in \Pi \setminus \{i\}$ let $\sigma_{j,i} = \sigma_{j,i}^\xi$. Note that $\sigma_i|_h^V$ is winning in $\mathcal{G}_i|_h^V$ if $\sigma_i^\alpha|_h^{V^\alpha}$ is winning in $\mathcal{G}_i|_h^{V^\alpha}$ for some ordinal α because, by definition of $V^{\alpha+1}$, if $\sigma_i^\alpha|_h^{V^\alpha}$ is winning, then every play of $\mathcal{G}_i|_h^{V^{\alpha+1}}$ is consistent with $\sigma_i^\alpha|_h^{V^\alpha}$ and therefore won by player i. As $V \subseteq V^{\alpha+1}$, this holds also for $\mathcal{G}_i|_h^V$.

As in the proof of Theorem 3.8, for each history $h \in T$ we define the player $p(h)$ who has to be "punished" after history h by

$$p(\varepsilon) = \bot$$

and

$$p(ha) = \begin{cases} \bot & \text{if } p(h) = \bot \text{ and } a = \sigma_{\lambda(h)}(h), \\ p(h) & \text{if } \lambda(h) \neq p(h),\, p(h) \neq \bot \text{ and } a = \sigma_{\lambda(h),p(h)}(h), \\ \lambda(h) & \text{otherwise} \end{cases}$$

for $h \in T$ and $a \in \Sigma$ with $ha \in T$. For every player $j \in \Pi$, her equilibrium strategy τ_j is defined by

$$\tau_j(h) = \begin{cases} \sigma_j(h) & \text{if } p(h) = \bot \text{ or } p(h) = j, \\ \sigma_{j,p(h)}(h) & \text{otherwise} \end{cases}$$

for all histories $h \in T$.

We show that $(\tau_j|_h)_{j\in\Pi}$ is a Nash equilibrium of $\mathcal{G}|_h$ for every history $h \in T$. Let $h \in T$ and $\pi = \langle(\tau_j|_h)_{j\in\Pi}\rangle$. Furthermore, let τ' be any strategy of some player i in $\mathcal{G}$ and $\pi' = \langle\tau'|_h, (\tau_j|_h)_{j\in\Pi\setminus\{i\}}\rangle$. We have to show that $h\pi \in A_i$ or $h\pi' \notin A_i$. The claim is trivial if $\pi = \pi'$. Thus assume $\pi \neq \pi'$ and fix the least $k < \omega$ such that $\pi(k) \neq \pi'(k)$. Then $\lambda(h \cdot \pi[0,k]) = i$ and $\tau'(h \cdot \pi[0,k]) \neq \tau_i(h \cdot \pi[0,k])$. Without loss of generality, let $k = 0$. We distuingish the following two cases:

(1) $\sigma_i|_h^V$ is winning in $\mathcal{G}_i|_h^V$. By definition of the strategies τ_j, π is a play of $\mathcal{G}_i|_h^V$. We show that π is consistent with $\sigma_i|_h^V$. As $\sigma_i|_h^V$ is winning, this implies that $h\pi \in A_i$. Otherwise fix the least $l < \omega$ such that $\lambda(h \cdot \pi[0,l]) = i$ and $\sigma_i|_h^V(\pi[0,l]) \neq \pi(l)$. As $\sigma_i|_h^V$ is winning, so is $\sigma_i|_h^V|_{\pi[0,l]} = \sigma_i|_{h\cdot\pi[0,l]}^V$. But then $(h \cdot \pi[0,l], \pi(l)) \in V^\xi \setminus V^{\xi+1}$, a contradiction to $V^\xi = V^{\xi+1}$.

(2) $\sigma_i|_h^V$ is not winning in $\mathcal{G}_i|_h^V$. As $(\sigma_i|_h^V, \sigma_{\Pi\setminus\{i\}}|_h^V)$ is a subgame perfect equilibrium of $\mathcal{G}_i|_h^V$, $\sigma_{\Pi\setminus\{i\}}|_h^V$ is winning. As $\tau'(h) \neq \tau_i(h)$, we have $p(h \cdot \pi'[0,l]) = i$ for all $0 < l < \omega$ and therefore $\pi' = \langle\tau'|_h, (\sigma_{j,i}|_h)_{j\in\Pi}\rangle$. We show that π' is a play of $\mathcal{G}_i|_h^V$. As $\sigma_{\Pi\setminus\{i\}}|_h^V$ is winning in $\mathcal{G}_i|_h^V$, this implies that $h\pi' \notin A_i$. Otherwise, let us fix the least $l < \omega$ such that $(h \cdot \pi'[0,l], \pi'(l)) \notin V$ together with the ordinal α such that $(h \cdot \pi'[0,l], \pi'(l)) \in V^\alpha \setminus V^{\alpha+1}$. Clearly $\lambda(h \cdot \pi'[0,l]) = i$. Hence, $\sigma_i^\alpha|_{h\cdot\pi'[0,l]}^{V^\alpha}$ is winning in $\mathcal{G}_i|_{h\cdot\pi'[0,l]}^{V^\alpha}$. But then $\sigma_i|_{h\cdot\pi'[0,l]}^V = \sigma_i|_h^V|_{\pi'[0,l]}$ is winning in $\mathcal{G}_i|_{h\cdot\pi'[0,l]}^V$. As $\pi'[0,l]$ is the prefix of a play of $\mathcal{G}_i|_h^V$ consistent with $\sigma_{\Pi\setminus\{i\}}|_h^V$, this implies that $\sigma_{\Pi\setminus\{i\}}|_h^V$ is not winning in $\mathcal{G}_i|_h^V$, a contradiction.

As $(\tau_j|_h)_{j\in\Pi}$ is a Nash equilibrium of $\mathcal{G}|_h$ for every history $h \in T$, $(\tau_j)_{j\in\Pi}$ is a subgame perfect equilibrium of $\mathcal{G}$. □

With this result, we conclude the game-theoretic analysis of infinite games. In the next chapters, we will look at a more practical model of infinite games and analyse the complexity of strategies realising an equilibrium as well as algorithmical issues in this model.

Chapter 4

Graph Games

Graph games are games played on a directed graph as arena. In the two-player variant, they turned out to be very fruitful for the verification and synthesis of open systems, i.e. systems that interact with the environment. McNaughton [McN65] was the first to apply the theory of two-player zero-sum graph games to infinite input-output behaviour. Büchi and Landweber [BL69] proved the first fundamental result on solving these games (cf. Corollary 5.19). See [ALW89] and [PR89, PR90] for applications to the synthesis of software modules and controllers.

4.1 Graph Games

When modelling an open system as a game, the system is represented as a directed graph where the vertices of the graph stand for the states of the system and the edges for possible state transitions. The vertex set is partitioned into vertices controlled by player 0 (system) and vertices controlled by player 1 (environment). A play is started at some initial vertex. Whenever the play reaches some vertex, the player who controls this vertex has to choose a successor vertex as the next vertex of the play. In this fashion, a possibly infinite play evolves. It is natural to generalise the model to an arbitrary number of players and independent winning conditions.

Definition 4.1. A tuple $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ is called a *multi-player graph game* if

(1) Π is a non-empty, finite set of *players*,

(2) V is a non-empty set of *vertices*,

(3) $(V_i)_{i\in\Pi}$ is a partition of V,

(4) $E \subseteq V \times V$ is a set of *edges* such that for every $v \in V$ the set $vE := \{w \in V : (v,w) \in E\}$ is non-empty, and

(5) $W_i \subseteq V^\omega$ for all $i \in \Pi$.

The directed, labelled graph $(V, (V_i)_{i\in\Pi}, E)$ is called the *arena* of $\mathcal{G}$. For any player i, the set W_i is called the *winning condition* for player i. $\mathcal{G}$ is called *finite* if V is finite. $\mathcal{G}$ is called *zero-sum* if $(W_i)_{i\in\Pi}$ defines a partition of V^ω.

A *play of* $\mathcal{G}$ is an infinite word $\pi \in V^\omega$ such that $(\pi(k), \pi(k+1)) \in E$ for all $k < \omega$. Any finite prefix h of a play π of $\mathcal{G}$ is called a *history of* $\mathcal{G}$. A play π is won by player i if $\pi \in W_i$. The *payoff* of a play π is the vector $(x_i)_{i\in\Pi} \in \{0,1\}^\Pi$ defined by $x_i = 1$ if π is won by player i.

Definition 4.2. For $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ a multiplayer graph game and $v_0 \in V$ a designated initial vertex, we call the pair $(\mathcal{G}, v_0)$ an *initialised multiplayer graph game*. A *play (history) of* $(\mathcal{G}, v_0)$ is a play π (non-empty history h) of $\mathcal{G}$ such that $\pi(0) = v_0$ ($h(0) = v_0$). For an initialised multiplayer graph game $(\mathcal{G}, v_0)$ and a history $hv \in V^*V$ of $(\mathcal{G}, v_0)$, the initialised multiplayer graph game $(\mathcal{G}|_h, v)$ defined by $\mathcal{G}|_h = (V, (V_i)_{i\in\Pi}, E, (W_i|_h)_{i\in\Pi})$ with $W_i|_h = \{\pi \in V^\omega : h\pi \in W_i\}$ is called the *subgame of* $(\mathcal{G}, v_0)$ *with history* hv.

Strategies and equilibria are defined for (initialised) multiplayer graph games in analogy to the model of infinite games introduced in Chapter 2.

Definition 4.3. For $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ a multiplayer graph game, a *strategy* of player i in $\mathcal{G}$ is any function $\sigma : V^*V_i \to V$ such that for all $w \in V^*$ and $v \in V_i$ we have $(v, \sigma(wv)) \in E$. A play π of $\mathcal{G}$ is *consistent with* σ if $\pi(k+1) = \sigma(\pi[0, k+1])$ for all $k < \omega$ with $\pi(k) \in V_i$. A *strategy profile* of $\mathcal{G}$ is a tuple $(\sigma_i)_{i\in\Pi}$ where σ_i is a strategy of player i in $\mathcal{G}$ for all players $i \in \Pi$.

For initialised multiplayer graph games, strategies and strategy profiles are defined accordingly. If $hv \in V^*V$is a history of $\mathcal{G}$ and σ is a strategy of some player i in $\mathcal{G}$, then $\sigma|_h$ defined by $\sigma|_h(w) = \sigma(hw)$ is a strategy of player i in the subgame $(\mathcal{G}|_h, v)$.

A strategy σ of some player i in an initialised multiplayer graph game $(\mathcal{G}, v_0)$ is *winning* if every play of $(\mathcal{G}, v_0)$ consistent with σ is won by player i. A strategy profile $(\sigma_i)_{i\in\Pi}$ of $(\mathcal{G}, v_0)$ uniquely determines a play of $(\mathcal{G}, v_0)$ consistent with each σ_i, which we denote by $\langle(\sigma_i)_{i\in\Pi}\rangle_{v_0}$ or $\langle(\sigma_i)_{i\in\Pi}\rangle$ if v_0 is clear from the context. This play is called the *outcome of* $(\sigma_i)_{i\in\Pi}$. The *payoff* of a strategy profile of an initialised multiplayer graph game is the payoff of its outcome.

Definition 4.4. A strategy profile $(\sigma_i)_{i\in\Pi}$ of an initialised multiplayer graph game $(\mathcal{G}, v_0)$ where $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ is called a *Nash equilibrium* if for all players $i \in \Pi$ we have

$$\langle\sigma', (\sigma_j)_{j\in\Pi\setminus\{i\}}\rangle \in W_i \Rightarrow \langle(\sigma_j)_{j\in\Pi}\rangle \in W_i$$

for all strategies σ' of player i in $(\mathcal{G}, v_0)$. $(\sigma_i)_{i\in\Pi}$ is called a *subgame perfect equilibrium* if $(\sigma_i|_h)_{i\in\Pi}$ is a Nash equilibrium of $(\mathcal{G}|_h, v)$ for every history $hv \in V^*V$ of $(\mathcal{G}, v_0)$.

Note that a strategy of player i is defined for any non-empty finite sequence of vertices ending in a vertex controlled by player i, not only for histories. However, it clearly suffices to define a strategy only for histories. We decided in favour of the former only for the purpose of simplifying notation. In both cases, strategies are infinite objects. Luckily, in many cases strategies with bounded memory suffice.

Definition 4.5. For $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ a multiplayer graph game and $i \in \Pi$, a *strategy automaton for player i in $\mathcal{G}$* is a tuple $\mathcal{A} = (Q, q_0, \delta, \tau)$ where

(1) Q is a non-empty, finite set of *states*,

(2) $q_0 \in Q$ is the *initial state*,

(3) $\delta : Q \times V \to Q$ is the *transition function*, and

(4) $\tau : Q \times V_i \to V$ with $\tau(q, v) \in vE$ for all $q \in Q$ and $v \in V_i$ is the *output function*.

The transition function δ is extended to a function $\delta^* : V^* \to Q$ defined by $\delta^*(\varepsilon) = q_0$ and $\delta^*(wv) = \delta(\delta^*(w), v)$ for all $wv \in V^*V$. $\mathcal{A}$ *computes* the strategy $\sigma_{\mathcal{A}}$ of player i in $\mathcal{G}$ defined by $\sigma_{\mathcal{A}}(wv) = \tau(\delta^*(w), v)$ for all $wv \in V^*V_i$.

A strategy σ of player i in $\mathcal{G}$ is called a *finite-state strategy* if $\sigma = \sigma_{\mathcal{A}}$ for a strategy automaton $\mathcal{A}$ of player i in $\mathcal{G}$. σ is called *positional* if it depends only on the last vertex, i.e. if $\sigma(wv) = \sigma(v)$ for all $wv \in V^*V_i$. A strategy profile is called a *finite-state strategy profile* or *positional* if all contained strategies are finite-state strategies or positional, respectively.

Obviously, any positional strategy can be computed by a strategy automaton with only one state.

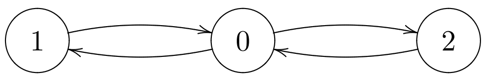

Figure 4.1: Arena of the game from Example 4.1.

Example 4.1. Let $\mathcal{G} = (\{1, 2\}, V, V_1, V_2, E, W_1, W_2)$ be the multiplayer graph game given by the arena depicted in Figure 4.1 where $V_1 = V$ and $V_2 = \emptyset$ and the winning conditions that player i visits vertex i, i.e. $W_i = V^*\{i\}V^\omega$. This game has two possible subgame perfect equilibrium payoffs for 0 as the initial vertex:

(1) Player 1 wins and player 2 loses, realised by player 1's positional strategy σ given by $\sigma(0) = 1$;

(2) Player 1 and player 2 win, realised for example by player 1's finite-state strategy σ given by $\sigma(h0) = 2$ if 1 occurs in h and $\sigma(h0) = 1$ otherwise.

Note that the second payoff is not realised by any positional strategy of player 1.

In analogy to infinite games, we measure the complexity of a multiplayer graph game in terms of the Borel hierarchy on V^ω.

Definition 4.6. A multiplayer graph game $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ is called a *multiplayer Borel game* if for all $i \in \Pi$ the set W_i is a Borel subset of V^ω. *Multiplayer open, closed, clopen,* Σ^0_α*-,* Π^0_α*- and* Δ^0_α*-games* are defined analogously.

A discrete infinite game $\mathcal{G} = (\Pi, \Sigma, T, \lambda, (A_i)_{i\in\Pi})$ as defined in Chapter 2 can be considered as a multiplayer graph game played with the game tree of $\mathcal{G}$ as arena and with its root as designated initial vertex. To define the winning condition of the graph game, we use the "last-letter extraction" function $f : (\Sigma^*)^\omega \to \Sigma^\omega : w_0 w_1 w_2 \ldots \mapsto a_1 a_2 \ldots$ where a_i is the last letter of w_i or $a_i = a$ for some fixed $a \in \Sigma$ if $w_i = \varepsilon$. Then we define the associated multiplayer graph game $\widetilde{\mathcal{G}} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ by

(1) $V = T$,

(2) $V_i = \{v \in V : \lambda(v) = i\}$ for all $i \in \Pi$,

(3) $E = \{(v, w) \in V \times V : w = va \text{ for some } a \in \Sigma\}$, and

(4) $W_i = \{\pi \in (\Sigma^*)^\omega : f(\pi) \in A_i\}$ for all $i \in \Pi$.

The designated initial vertex is ε. It is easy to see that f is continuous with $f : W_i \leq A_i$ for all players $i \in \Pi$. Hence $\widetilde{\mathcal{G}}$ is a Σ^0_α-game (Π^0_α-game) if $\mathcal{G}$ is a Σ^0_α-game (Π^0_α-game).

On the other hand, let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ be a multiplayer graph game with designated initial vertex v_0. In the corresponding infinite game, executing an action v corresponds to moving to vertex v in the graph game. The game tree is just the unravelling of the arena of $\mathcal{G}$ from v_0. To define the winning condition of the infinite game, we use the function $g : V^\omega \to V^\omega : \pi \mapsto v_0\pi$. Formally, we define the associated discrete infinite game $\widetilde{\mathcal{G}} = (\Pi, V, T, \lambda, (A_i)_{i\in\Pi})$ by

(1) $T = \{h \in V^* : v_0 h \text{ is a history of } (\mathcal{G}, v_0)\}$,

(2) $\lambda(h) = i$ if either $h = \varepsilon$ and $v_0 \in V_i$ or $h = h'v$ for some $v \in V_i$,

(3) $A_i = \{\pi \in V^\omega : g(\pi) \in W_i\}$ for all $i \in \Pi$.

Clearly, g is continuous with $g : A_i \leq W_i$ for all $i \in \Pi$. Hence, $\widetilde{\mathcal{G}}$ is a Σ^0_α-game (Π^0_α-game) if $\mathcal{G}$ is a Σ^0_α-game (Π^0_α-game).

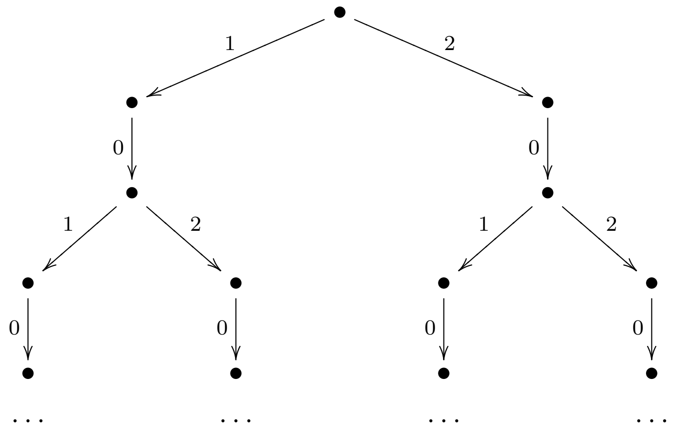

Figure 4.2: Translation of the graph game from Example 4.1

Example 4.2. Consider the initialised multiplayer graph game defined in Example 4.1. The game tree of the corresponding infinite game is depicted in Figure 4.2. The Σ^0_1 winning conditions $V^* \cdot \{i\} \cdot V^\omega$ remain the same.

It should be fairly clear that we can define a strategy in the multiplayer graph game from a strategy in the corresponding infinite game and vice versa such that Nash equilibria (subgame perfect equilibria) are mapped to Nash equilibria (subgame perfect equilibria). Thus the existence of an equilibrium in a certain multiplayer graph game follows from the existence of an equilibrium in the corresponding infinite game. In particular, we have the following theorem as a variant of Theorem 3.10.

Theorem 4.7. *Every initialised multiplayer Borel game has a subgame perfect equilibrium.*

Hence, from a game-theoretic point of view, discrete infinite games and multiplayer graph games are essentially the same. Thus, from now on, we will simply speak of a game when we mean a multiplayer graph game. Often we will also implicitly assume that the game under consideration is initialised. The advantage of graph games is that they are much more flexible in applications. Typical winning conditions in these games can be formalised as a constraint on the set of vertices visited or visited infinitely often.

4.2 Winning Conditions

We consider games where the winning condition is given by an acceptance condition on infinite words over a finite, non-empty set C of *colours* and a *colouring* $\chi : V \to C$ of the (possibly infinite arena) as considered by Zielonka [Zie98]. χ is extended to a function $\chi : V^\omega \to C^\omega$ in the natural way, i.e. $\chi(v_0 v_1 \dots) = \chi(v_0)\chi(v_1)\dots$ for all $v_0 v_1 \dots \in V^\omega$. All presented acceptance conditions will be finitely represented. If the arena of the game and the colouring are also finitely represented, we can use the games as an input for an algorithm. For finite games, one may take V as the set of colours and the identity mapping as colouring. In this chapter, we will discuss the following acceptance conditions.

(1) *Reachability*: Given a set $F \subseteq C$, a word $\alpha \in C^\omega$ is accepted if and only if $\mathrm{Occ}(\alpha) \cap F \neq \emptyset$.

(2) *Safety*: Given a set $F \subseteq C$, a word $\alpha \in C^\omega$ is accepted if and only if $\mathrm{Occ}(\alpha) \subseteq F$.

(3) *Büchi*: Given a set $F \subseteq C$, a word $\alpha \in C^\omega$ is accepted if and only if $\mathrm{Inf}(\alpha) \cap F \neq \emptyset$.

(4) *Co-Büchi*: Given a set $F \subseteq C$, a word $\alpha \in C^\omega$ is accepted if and only if $\mathrm{Inf}(\alpha) \subseteq F$.

(5) *Parity*: Given a *priority function* $\Omega : C \to \omega$, a word $\alpha \in C^\omega$ is accepted if and only if $\min(\Omega(\mathrm{Inf}(\alpha)))$ is even.

Clearly, a safety condition is the negation of a reachability condition, a co-Büchi condition is the negation of a Büchi condition, a Büchi or co-Büchi condition is also a parity condition (for a Büchi condition define $\Omega(c) = 0$ if $c \in F$ and $\Omega(c) = 1$ if $c \notin F$), and the negation of a parity condition is again a parity condition (add 1 to the priority of each colour). The Büchi condition has been introduced by Büchi [Büc62] and the parity condition independently by Emerson and Jutla [EJ91] and Mostowski [Mos84].

We establish the topological complexity of the presented acceptance conditions on C^ω. For a reachability condition given by $F \subseteq C$, the set of accepted words is

$$(C^* \cdot F) \cdot C^\omega \in \Sigma^0_1.$$

For a Büchi condition given by $F \subseteq C$, the set of accepted words is

$$(C^* \cdot F)^\omega = \bigcap_{n<\omega} \big((C^* \cdot F)^n \cdot C^\omega\big) \in \Pi^0_2.$$

As a safety or co-Büchi condition can be written as the negation of a reachability or Büchi condition, respectively, safety and co-Büchi acceptance conditions lie in the classes Π^0_1 and Σ^0_2, respectively. Finally, let $\Omega : C \to \omega$ be a priority function. A word $\alpha \in C^\omega$ is accepted by the associated parity condition if there is an even priority occurring infinitely often in α but no smaller one. Hence, the set of accepted words is

$$\bigcup_{\substack{k\in\Omega(C)\\ k \text{ even}}} \Big[\big(C^* \cdot \Omega^{-1}(k)\big)^\omega \setminus \bigcup_{\substack{l\in\Omega(C)\\ l<k}} \big(C^* \cdot \Omega^{-1}(l)\big)^\omega\Big],$$

a Boolean combination of Σ^0_2-sets and therefore contained in Δ^0_3.

Definition 4.8. Let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ be a game. $\mathcal{G}$ is called a *multiplayer reachability, safety, (co-)Büchi, or parity game* if there exists a colouring $\chi : V \to C$ into a finite, non-empty set C and a reachability, safety, (co-)Büchi, or parity condition $A_i \subseteq C^\omega$, respectively, such that $W_i = \{\pi \in V^\omega : \chi(\pi) \in A_i\}$.

We write $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (F_i)_{i\in\Pi})$ where $F_i \subseteq C$ for multiplayer reachability, safety, Büchi and co-Büchi games. For multiplayer parity games, we write $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\Omega_i)_{i\in\Pi})$ where $\Omega_i : C \to \omega$.

Note that any zero-sum reachability game is also a safety game and vice versa, and that any zero-sum Büchi game is also a co-Büchi game and vice versa. Thus our notation differs from the standard notation for two-player zero-sum graph games, where a game is called a reachability (Büchi) game if the winning condition of player 0 is given by a reachability (Büchi) condition (which does not imply that the winning condition of player 1 can be given by a reachability (Büchi) condition).

For two-player zero-sum parity games, one can choose $C \subset \omega$ and Ω_0 as the identity function on C, i.e. one can identify colours with priorities. In this case, we can further assume that $\Omega_1(k) = k + 1$ for all $k \in C$. This gives *parity games* as considered in the literature [EJ91]. We have decided to uncouple colours and priorities, as we want to allow an independent winning condition for each player.

As for $\chi : V \to C$ the induced function $\chi : V^\omega \to C^\omega$ on words is continuous with $\chi : W_i \leq A_i$ for χ, W_i and A_i as in Definition 4.8, the topological complexity of the presented games can be inferred from the topological complexity of the used acceptance condition, i.e. reachability games are Σ^0_1-games, safety games are Π^0_1-games, Büchi games are Π^0_2-games, co-Büchi games are Σ^0_2-games, and parity games are Δ^0_3-games.

4.3 Game Reductions

We present the notion of a game reduction introduced by Thomas [Tho95]. We use it as a tool to transfer equilibria from one game to another.

Definition 4.9. For two games $\mathcal{G} = (\Pi, V, (V_i), E, (W_i)_{i\in\Pi})$ and $\mathcal{G}' = (\Pi, V', (V'_i), E', (W'_i)_{i\in\Pi})$ with initial vertices v_0 and v'_0, respectively, we write $(\mathcal{G}, v_0) \leq (\mathcal{G}', v'_0)$ if there exists a non-empty set S, $s_0 \in S$ and a function $f : S \times V \to S$ such that

(1) $V' = V \times S$,

(2) $V'_i = V_i \times S$ for all $i \in \Pi$,

(3) $v'_0 = (v_0, s_0)$,

(4) $E' = \{((v,s),(w,t)) \in V' \times V' : (v,w) \in E \text{ and } t = f(s,v)\}$, and

(5) $\pi \in W_i \Leftrightarrow \pi' \in W'_i$ for all plays π of $(\mathcal{G}, v_0)$.

Here, for a play π of $(\mathcal{G}, v_0)$, π' denotes the unique play of $(\mathcal{G}', v'_0)$ where the elementwise projection on the first component is precisely π. In the case that $(\mathcal{G}, v_0) \leq (\mathcal{G}', v'_0)$ we say that $(\mathcal{G}, v_0)$ *reduces to* $(\mathcal{G}', v'_0)$. If the set S in the definition of $(\mathcal{G}, v_0) \leq (\mathcal{G}', v'_0)$ is finite, we write $(\mathcal{G}, v_0) \leq_{\text{fin}} (\mathcal{G}', v'_0)$ and say that $(\mathcal{G}, v_0)$ *finitely reduces to* $(\mathcal{G}', v'_0)$.

We remark that the resulting relations $\leq$ and $\leq_{\text{fin}}$ are reflexive (by identifying V with $V \times \{0\}$) and transitive. The following two lemmata show that game reductions are meaningful because they preserve equilibria.

Lemma 4.10. *If $(\mathcal{G}, v_0) \leq (\mathcal{G}', v'_0)$ for two games $(\mathcal{G}, v_0)$ and $(\mathcal{G}', v'_0)$, then $(\mathcal{G}, v_0)$ has a Nash (subgame perfect) equilibrium with payoff x if and only if $(\mathcal{G}', v'_0)$ has a Nash (subgame perfect) equilibrium with payoff x.*

Proof. Let $(\mathcal{G}, v_0) \leq (\mathcal{G}', v'_0)$ where $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ and $\mathcal{G}' = (\Pi, V', (V'_i)_{i\in\Pi}, E', (W'_i)_{i\in\Pi})$ and fix S, s_0 and f as in Definition 4.9. We extend f to a function $f^* : V^* \to S$ by defining $f^*(\varepsilon) = s_0$ and $f^*(wv) = f(f^*(w), v)$ for all $wv \in V^*V$.

($\Rightarrow$) Assume $(\sigma_i)_{i\in\Pi}$ is a Nash (subgame perfect) equilibrium of $(\mathcal{G}, v_0)$. For each player $i \in \Pi$, define a strategy σ'_i of player i in $(\mathcal{G}', v'_0)$ by

$$\sigma'_i((v_1, s_1) \dots (v_k, s_k)) = (\sigma_i(v_1 \dots v_k), f(s_k, v_k))$$

for all $(v_1, s_1) \dots (v_k, s_k) \in V'^* V'_i$. Then, by condition (5) in Definition 4.9, $\langle(\sigma_i)_{i\in\Pi}\rangle$ and $\langle(\sigma'_i)_{i\in\Pi}\rangle$ have the same payoff. It is straightforward to show that $(\sigma'_i)_{i\in\Pi}$ is a Nash (subgame perfect) equilibrium of $(\mathcal{G}', v'_0)$.

($\Leftarrow$) Assume $(\sigma'_i)_{i\in\Pi}$ is a Nash (subgame perfect) equilibrium of $(\mathcal{G}', v'_0)$. For each player $i \in \Pi$, we define a strategy σ_i of player i in $(\mathcal{G}, v_0)$ by

$$(\sigma_i(v_1 \dots v_k), f^*(v_1 \dots v_k)) = \sigma'_i((v_1, f^*(\varepsilon)) \dots (v_k, f^*(v_1 \dots v_{k-1})))$$

for all $v_1 \dots v_k \in V^* V_i$. Then, by condition (5) in Definition 4.9, $\langle(\sigma'_i)_{i\in\Pi}\rangle$ and $\langle(\sigma_i)_{i\in\Pi}\rangle$ have the same payoff. It is straightforward to show that $(\sigma_i)_{i\in\Pi}$ is a Nash (subgame perfect) equilibrium of $(\mathcal{G}, v_0)$. □

Lemma 4.11. *If $(\mathcal{G}, v_0) \leq_{\text{fin}} (\mathcal{G}', v'_0)$ for two games $(\mathcal{G}, v_0)$ and $(\mathcal{G}', v'_0)$, then $(\mathcal{G}, v_0)$ has a finite-state Nash (subgame perfect) equilibrium with payoff x if and only if $(\mathcal{G}', v'_0)$ has a finite-state Nash (subgame perfect) equilibrium with payoff x.*

Proof. Let $(\mathcal{G}, v_0) \leq_{\text{fin}} (\mathcal{G}', v_0')$ where $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ and $\mathcal{G}' = (\Pi, V', (V_i')_{i\in\Pi}, E', (W_i')_{i\in\Pi})$ and fix S, s_0 and f as in Definition 4.9.

($\Rightarrow$) Let $(\sigma_i)_{i\in\Pi}$ be a finite-state Nash (subgame perfect) equilibrium of $(\mathcal{G}, v_0)$. We show that the Nash (subgame perfect) equilibrium $(\sigma_i')_{i\in\Pi}$ of $(\mathcal{G}', v_0')$ as defined in the proof of Lemma 4.10 is a finite-state strategy profile. Let $i \in \Pi$ and fix a strategy automaton $\mathcal{A} = (Q, q_0, \delta, \tau)$ of player i in $\mathcal{G}$ with $\sigma_{\mathcal{A}} = \sigma_i$. We define a strategy automaton $\mathcal{A}' = (Q, q_0, \delta', \tau')$ of player i in $\mathcal{G}'$ by

$$\delta'(q, (v, s)) = \delta(q, v),$$
$$\tau'(q, (v, s)) = (\tau(q, v), f(s, v))$$

for all $q \in Q$, $v \in V$ and $s \in S$. It is straightforward to verify that $\sigma_{\mathcal{A}'} = \sigma_i'$.

($\Leftarrow$) Let $(\sigma_i')_{i\in\Pi}$ be a finite-state Nash (subgame perfect) equilibrium of $(\mathcal{G}', v_0')$. We show that the Nash (subgame perfect) equilibrium $(\sigma_i)_{i\in\Pi}$ of $(\mathcal{G}, v_0)$ as defined in the proof of Lemma 4.10 is a finite-state strategy profile. Let $i \in \Pi$ and fix a strategy automaton $\mathcal{A}' = (Q, q_0, \delta', \tau')$ of player i in $\mathcal{G}'$ with $\sigma_{\mathcal{A}'} = \sigma_i'$. We define a strategy automaton $\mathcal{A} = (Q \times S, (q_0, s_0), \delta, \tau)$ of player i in $\mathcal{G}$ by

$$\delta((q, s), v) = (\delta'(q, (v, s)), f(s, v)),$$
$$\tau((q, s), v) = w \text{ if } \tau'(q, (v, s)) = (w, t) \text{ for some } t \in S$$

for all $q \in Q$, $s \in S$ and $v \in V$. It is straightforward to verify that indeed $\sigma_{\mathcal{A}} = \sigma_i$. □

Our first example of a game reduction is a reduction of multiplayer reachability and safety games to multiplayer (co-)Büchi games. As multiplayer (co-)Büchi games are a special case of multiplayer parity games, this allows us to concentrate on parity games in our further analysis.

Claim 4.12. *For every multiplayer reachability or safety game $(\mathcal{G}, v_0)$, there exists a multiplayer (co-)Büchi game $(\mathcal{G}', v_0')$ with $(\mathcal{G}, v_0) \leq_{\text{fin}} (\mathcal{G}', v_0')$.*

Proof. Let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (F_i)_{i\in\Pi})$ be a multiplayer reachability game. The multiplayer Büchi game $\mathcal{G}' = (\Pi, V', (V_i')_{i\in\Pi}, E', \chi', (F_i')_{i\in\Pi})$ has vertex set

$$V' = V \times \{0, 1\}^{\Pi}$$

where $(v, x) \in V_i' \Leftrightarrow v \in V_i$. The colouring function $\chi' : V' \to \{0, 1\}^{\Pi}$ maps a vertex (v, x) to x. $(x_i)_{i\in\Pi} \in \{0, 1\}^{\Pi}$ is contained in F_i' if and only if $x_i = 1$. It remains to specify E'. A pair $((v, (x_i)_{i\in\Pi}), (w, (y_i)_{i\in\Pi}))$ is in E' if and only if $(v, w) \in E$ and

$$y_i = 1 \Leftrightarrow (x_i = 1 \text{ or } \chi(v) \in F_i)$$

for all $i \in \Pi$. The initial vertex of $\mathcal{G}'$ is $v_0' = (v_0, (0)_{i\in\Pi})$. It is straightforward to verify that $(\mathcal{G}, v_0) \leq_{\text{fin}} (\mathcal{G}', v_0')$. Moreover, this also holds if we define $\mathcal{G}'$ as a co-Büchi game. The construction for safety games is analogous. □

4.4 Parity Games

We have seen that any of the presented winning conditions is or can be reduced to a parity condition. In Chapter 5, we will show that this holds for a much larger class of winning conditions. Hence, parity games are of central interest in this thesis. In the theory of two-player zero-sum games, there is a second reason for the importance of parity games. They enjoy *positional determinacy*, i.e. either one of the two players not only has a winning strategy (as it is already guaranteed by Martin's theorem), but even a positional one. The result was obtained independently by Emerson and Jutla [EJ91] and Mostowski [Mos91]. Games with generalisations of the parity condition as winning condition lack this nice property in general.

Theorem 4.13 (Emerson, Jutla; Mostowski)**.** *Two-player zero-sum parity games are positionally determined, i.e. in any initialised two-player zero-sum parity game either one of the two players has a positional winning strategy.*

An important consequence of Theorem 4.13 is that the problem of deciding whether player 0 has a winning strategy in an initialised finite two-player zero-sum parity game lies in the complexity class NP ∩ co-NP.

Corollary 4.14. *The decision problem, given an initialised finite two-player zero-sum parity game, decide whether player 0 has a winning strategy, is in* NP ∩ *co*-NP*.*

Proof. We show that the problem is in NP. As player 0 does not have a winning strategy in a two-player zero-sum parity game $(\mathcal{G}, v_0)$ if and only if player 1 has a winning strategy in $(\mathcal{G}, v_0)$, its membership in co-NP follows immediately. By Theorem 4.13, it suffices to show that, given a positional strategy σ of player 0 in $(\mathcal{G}, v_0)$, we can decide in polynomial time whether σ is winning.

Let $\mathcal{G} = (\{0,1\}, V, V_0, V_1, E, \chi, \Omega_0, \Omega_1)$ be a finite two-player zero-sum parity game with initial vertex v_0 and σ a positional strategy of player 0. We consider the *solitaire game* $\mathcal{G}_\sigma = (\{0,1\}, V, V_0, V_1, E_\sigma, \chi, \Omega_0, \Omega_1)$ where E_σ is the set of edges $(v,w) \in E$ such that either $v \in V_1$ or $\sigma(v) = w$, i.e. we fix the edges taken by σ. Then σ is a winning strategy if and only if in (V, E_σ) there is no vertex v reachable from v_0 such that $\Omega_0(\chi(v))$ is odd and there is a path from v to v containing only vertices w with priority $\Omega_0(\chi(w)) \geq \Omega_0(\chi(v))$. This can easily be checked by depth-first search in time $O(d|E|)$ where d is the number of different priorities. □

Jurdziński [Jur98] proved the stronger result that the decision problem is in UP ∩ co-UP, where UP is the class of languages recognised by a nondeterministic Turing machine with at most one accepting computation on every input. Thus, the problem is "not too far away from P". It is indeed a long-standing open problem whether the problem is in P.

By Lemma 2.14, Theorem 4.13 is equivalent to the existence of a positional Nash equilibrium in any initialised two-player zero-sum parity game (choose any positional strategy of the player who has no winning strategy as her equilibrium strategy). We can even show more: Any two-player zero-sum parity game has a *uniform* positional Nash equilibrium, i.e. a positional strategy profile that is a Nash equilibrium for every initial vertex we choose. Note that this is equivalent to the existence of a uniform positional subgame perfect equilibrium because, if $\mathcal{G}$ is a parity game and σ a positional strategy in $\mathcal{G}$, then we have $\mathcal{G}|_h = \mathcal{G}$ and $\sigma|_h = \sigma$ for each history h. We give two proofs of the theorem. The first one is an application of Theorem 4.13 whereas the second one is a direct proof following Grädel [Grä04].

Theorem 4.15. *For any two-player zero-sum parity game $\mathcal{G}$, there exists a positional strategy profile that is a Nash equilibrium of the initialised game $(\mathcal{G}, v)$ for every vertex v.*

The following lemma allows the composition of positional winning strategies and will be needed in both proofs of Theorem 4.15.

Lemma 4.16. *Let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\Omega_i)_{i\in\Pi})$ be a multiplayer parity game, $i \in \Pi$ and $X \subseteq V$. If, for all $v \in X$, player i has a positional winning strategy in the initialised game $(\mathcal{G}, v)$, then she also has a positional strategy in $\mathcal{G}$ that is winning in $(\mathcal{G}, v)$ for all $v \in X$.*

Proof. We choose a well-ordering on X and index X by all ordinals below $\kappa := |X|$, i.e. $X = \{v_\alpha : \alpha < \kappa\}$. For each $\alpha < \kappa$, we choose a winning strategy σ_α of player i in $(\mathcal{G}, v_\alpha)$. Moreover, let $F_\alpha \subseteq X$ be the set of vertices ooccurring in at least one play of $(\mathcal{G}, v_\alpha)$ consistent with σ_α. We define a new positional strategy σ of player i in $\mathcal{G}$ by

$$\sigma(v) = \begin{cases} \sigma_\alpha(v) \text{ for the least } \alpha \text{ such that } v \in F_\alpha & \text{if } v \in X, \\ \sigma_0(v) & \text{otherwise.} \end{cases}$$

σ is well-defined because $v \in X$ implies $v = v_\alpha \in F_\alpha$ for some ordinal $\alpha < \kappa$. σ is a winning strategy in $(\mathcal{G}, v_\alpha)$ for all $\alpha < \kappa$ because, if ρ is a play of $(\mathcal{G}, v_\alpha)$ consistent with σ, then there exist $\beta < \kappa$ and $k < \omega$ such that $\rho(k) \in F_\beta$ and $\rho[k, \omega]$ is consistent with σ_β. As σ_β is winning from every initial vertex $v \in F_\beta$, this implies that ρ is won by player i. □

Proof of Theorem 4.15. Let $\mathcal{G} = (\{0,1\}, V, V_0, V_1, E, \chi, \Omega_0, \Omega_1)$ be a two-player zero-sum parity game and for $i \in \{0,1\}$ let

$$W_i = \{v \in V : \text{player } i \text{ has a positional winning strategy in } (\mathcal{G}, v)\}.$$

By Theorem 4.13, we have $W_0 \cup W_1 = V$. By Lemma 4.16, there exists a positional strategy σ of player 0 and a positional strategy τ of player 1 in $\mathcal{G}$ such that σ and τ are winning strategies for every initial vertex $v \in W_0$ and $v \in W_1$, respectively. By Lemma 2.14, (σ, τ) is a Nash equilibrium of $(\mathcal{G}, v)$ for all $v \in W_0 \cup W_1 = V$. □

Definition 4.17. For a game $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (W_i)_{i\in\Pi})$, a player $i \in \Pi$ and $X \subseteq V$, let $\mathrm{Attr}_i(X)$ be the set of vertices from which player i can force a visit of X, i.e. $\mathrm{Attr}_i(X) = \bigcup_{\alpha\in\mathrm{On}} Z^\alpha$ with

$$\begin{aligned} Z^0 &= X, \\ Z^\lambda &= \bigcup_{\alpha<\lambda} Z^\alpha \text{ if } \lambda \text{ is a limit ordinal, and} \\ Z^{\alpha+1} &= Z^\alpha \cup \{v \in V_i : vE \cap Z^\alpha \neq \emptyset\} \cup \{v \in V \setminus V_i : vE \subseteq Z^\alpha\}. \end{aligned}$$

It is easy to see that player i has a positional strategy σ in $\mathcal{G}$ such that $\mathrm{Occ}(\rho) \cap X \neq \emptyset$ for all plays ρ of $\mathcal{G}$ starting in a vertex of $\mathrm{Attr}_i(X)$ (if the rank of a vertex $v \in \mathrm{Attr}_i(X)$ is the least ordinal α such that $v \in Z^\alpha$, then move from a vertex v with rank α to a vertex $w \in vE$ with rank $\beta < \alpha$). We call any such strategy σ a *positional attractor strategy.*

Direct proof of Theorem 4.15. The proof is carried out by induction on the number $c = |\chi(V)|$ of used colours. If $c = 1$, then either one of the two players wins every play of $\mathcal{G}$ (depending on the priority of the single used colour). Thus every pair of positional strategies of player 0 and player 1, respectively, is a uniform Nash equilibrium of $\mathcal{G}$.

Now assume $c > 1$ and let $m = \min(\Omega_0(\chi(V)))$. Without loss of generality, let m be even (otherwise we can switch players). Let

$$X = \{v \in V : \text{player 1 has a positional winning strategy in } (\mathcal{G}, v)\}.$$

By Lemma 4.16, we can fix a positional strategy τ of player 1 that is winning in $(\mathcal{G}, v)$ for all $v \in X$. We construct a positional strategy σ of player 0 that is winning in $(\mathcal{G}, v)$ for all $v \in V \setminus X$. First observe that, if $v \in V_0 \setminus X$, then $vE \setminus X \neq \emptyset$. Thus we can fix a positional strategy $\sigma_{\overline{X}}$ of player 0 with $\sigma_{\overline{X}}(v) \in V \setminus X$ for all $v \in V_0 \setminus X$. Now let

$$Y = \chi^{-1}(\Omega^{-1}(m)), \; Z = \mathrm{Attr}_0(Y)$$

and fix a corresponding positional attractor strategy σ_Y of player 0. Let $V' := V \setminus (X \cup Z)$. If $V' \neq \emptyset$, then we can restrict the arena of $\mathcal{G}$ to V' and obtain a two-player zero-sum parity game $\mathcal{G}'$ with less than c used colours (if there exists a vertex $v \in V'$ with $vE \subseteq X \cup Z$, then also $v \in X \cup Z$ and therefore $v \notin V'$). By the induction hypothesis, $\mathcal{G}'$ has a uniform positional

Nash equilibrium (σ', τ'). Observe that σ' is winning in $(\mathcal{G}', v)$ for all $v \in V'$ because, if τ' is winning in $(\mathcal{G}', v)$ for some $v \in V'$, then τ'' defined by

$$\tau''(v) = \begin{cases} \tau(v) & \text{if } v \in V', \\ \tau'(v) & \text{otherwise} \end{cases}$$

is winning in $(\mathcal{G}, v)$ and therefore $v \in X$ (otherwise there would exist a vertex $w \in V' \cap V_0$ with $wE \cap Z \neq \emptyset$, but then $w \in Z$ by definition of Z). If $V' = \emptyset$, then let σ' be an arbitrary positional strategy of player 0 in $\mathcal{G}$. We define σ by

$$\sigma(v) = \begin{cases} \sigma'(v) & \text{if } v \in V', \\ \sigma_Y(v) & \text{if } v \in Z \setminus Y, \\ \sigma_{\overline{X}}(v) & \text{otherwise.} \end{cases}$$

It remains to show that σ is winning in $(\mathcal{G}, v)$ for all $v \in V \setminus X$. Let ρ be a play of $(\mathcal{G}, v)$ consistent with σ where $v \in V \setminus X$. Clearly, $\rho(k) \in V \setminus X$ for all $k < \omega$. We consider the following two cases.

(1) $\rho(k) \in Z$ only for finitely many $k < \omega$. Then there exits $l < \omega$ such that $\rho(k) \in V'$ for all $l \leq k < \omega$. Thus, $\rho[l, \omega]$ is a play of $\mathcal{G}'$ consistent with σ' and therefore won by player 0.

(2) $\rho(k) \in Z$ for infinitely many $k < \omega$. By definition of the attractor strategy σ_Y, also $\rho(k) \in Y$ for infinitely many $k < \omega$ and therefore $\min(\Omega_0(\chi(\rho))) = m$, which is even. Hence ρ is won by player 0.

As σ is winning in $(\mathcal{G}, v)$ for all $v \in X$ and τ is winning in $(\mathcal{G}, v)$ for all $v \in V \setminus X$, (σ, τ) is a Nash equilibrium of $(\mathcal{G}, v)$ for all $v \in V$. □

There are numerous sophisticated deterministic algorithms for computing a uniform positional Nash equilibrium of a two-player zero-sum parity game (see [GTW02, Chapters 7–8]). The currently best known one is due to Jurdziński [Jur00]. It needs time $O(dm(\frac{n}{\lfloor d/2 \rfloor})^{\lfloor d/2 \rfloor})$ and space $O(dn \log n)$ for a parity game with n vertices, m edges and $d \geq 2$ different priorities (for player 0).

Theorem 4.18 (Jurdziński)**.** *Computing a uniform positional Nash equilibrium of a two-player zero-sum parity game with at most n vertices, m edges and $d \geq 2$ different priorities can be done in time $O\big(dm(\frac{n}{\lfloor d/2 \rfloor})^{\lfloor d/2 \rfloor}\big)$.*

Note that Jurdziński's algorithm can also be used to determine the winner of an initialised parity game because, given a positional strategy of player 0, we can decide in linear time if it is a winning strategy (see the proof of Corollary 4.14).

We will now turn to multiplayer parity games. By Theorem 4.7, these games are guaranteed to have a subgame perfect equilibrium. Thus our next step is to analyse the complexity such an equilibrium may have. We show that parity games with two players allow a positional subgame perfect equilibrium and that finite parity games with an arbitrary number of players allow a finite-state subgame perfect equilibrium. This raises two questions, namely whether we can guarantee a positional subgame perfect equilibrium also in the case of more than two players, and if this is not true for games with an infinite arena, whether we can guarantee the existence of a finite-state subgame perfect equilibrium in parity games with an infinite arena. We leave these questions as open problems.

To prove our results on multiplayer parity games, we use the same idea as in the proof of Theorem 3.10. We start with the complete arena and check which players have a winning strategy from which initial vertex. By Theorem 4.15, for each player we can choose a uniform positional strategy that is winning from each vertex from which this player has a winning strategy. Then, for any vertex from which the player who controls this vertex has a winning strategy, we delete all those outgoing edges that are not taken by the chosen strategy. This yields a new arena for the game, and we iterate this construction until we reach a fixed point. In the resulting game, we choose an optimal positional counter-strategy for each player. As it turns out, this gives a subgame perfect equilibrium of the original game.

Theorem 4.19. *Any initialised two-player parity game has a positional subgame perfect equilibrium.*

Proof. Let $\mathcal{G} = (\{0,1\}, V, V_0, V_1, E, \chi, \Omega_0, \Omega_1)$ be a two-player parity game. We define a set $E^\alpha \subseteq E$ for each ordinal α beginning with

$$E^0 = E.$$

If λ is a limit ordinal, let

$$E^\lambda = \bigcap_{\alpha<\lambda} E^\alpha.$$

To define $E^{\alpha+1}$ from E^α, we consider the two-player zero-sum parity game $\mathcal{G}_0^\alpha = (\{0,1\}, V, V_0, V_1, E_\alpha, \chi, \Omega_0, \overline{\Omega_0})$ and the two-player zero-sum parity game $\mathcal{G}_1^\alpha = (\{0,1\}, V, V_0, V_1, E_\alpha, \chi, \overline{\Omega_1}, \Omega_1)$ defined by $\overline{\Omega_i} = \Omega_i + 1$, i.e. $\mathcal{G}_i^\alpha$ is the parity game played on the arena of $\mathcal{G}$ with the set of edged restricted to E^α and with the winning condition for player $1-i$ changed to the complement of the winning condition for player i. By Theorem 4.15, we can fix a uniform positional Nash equilibrium $(\sigma_0^\alpha, \tau_1^\alpha)$ of $\mathcal{G}_0^\alpha$ and a uniform positional Nash equilibrium $(\tau_0^\alpha, \sigma_1^\alpha)$ of $\mathcal{G}_1^\alpha$. Let W_i^α be the set of all $v \in V$ such that σ_i^α is winning in $(\mathcal{G}_i^\alpha, v)$. Then we define

$$E^{\alpha+1} = \bigcap_{i\in\Pi} \{(v,w) \in E^\alpha : v \notin V_i \cap W_i^\alpha \text{ or } w = \sigma_i^\alpha(v)\}.$$

The sequence $(E^\alpha)_{\alpha\in\mathrm{On}}$ is obviously non-increasing. Thus fix the least ordinal ξ with $E^\xi = E^{\xi+1}$ and define $\tau_0 = \tau_0^\xi$ and $\tau_1 = \tau_1^\xi$.

We show that (τ_0, τ_1) is a Nash equilibrium of $(\mathcal{G}, v)$ for all $v \in V$. As $\mathcal{G}|_h = \mathcal{G}$, $\tau_0|_h = \tau_0$ and $\tau_1|_h = \tau_1$ for all histories h of $\mathcal{G}$, this implies that (τ_0, τ_1) is a positional subgame perfect equilibrium of $(\mathcal{G}, v_0)$ for any initial vertex v_0. Let $v \in V$ and $\pi = \langle \tau_0, \tau_1\rangle_v$. Furthermore, let τ' be any other strategy of player 1 in $(\mathcal{G}, v)$ and $\pi' = \langle \tau_0, \tau'\rangle_v$. We have to show that π is won by player 1 or that π' is not won by player 1. We distinguish the following two cases:

(1) σ_1^ξ is winning in $(\mathcal{G}_1^\xi, v)$. By definition of the strategies τ_0 and τ_1, π is a play of $(\mathcal{G}_1^\xi, v)$. We show that π is consistent with σ_1^ξ which implies that π is won by player 1. Otherwise fix the least $l < \omega$ such that $\pi(l) \in V_1$ and $\sigma_1^\xi(\pi(l)) \neq \pi(l+1)$. As σ_1^ξ is winning in $(\mathcal{G}_1^\xi, v)$, σ_1^ξ is also winning in $(\mathcal{G}_1^\xi, \pi(l))$. But then $(\pi(l), \pi(l+1)) \in E^\xi \setminus E^{\xi+1}$, a contradiction to $E^\xi = E^{\xi+1}$.

(2) σ_1^ξ is not winning in $(\mathcal{G}_1^\xi, v)$. Then τ_0 is winning in $(\mathcal{G}_1^\xi, v)$. We show that π' is a play of $(\mathcal{G}_1^\xi, v)$ which implies that π' is not won by player 1. Otherwise fix the least $l < \omega$ such that $(\pi'(l), \pi'(l+1)) \notin E^\xi$ together with the ordinal α such that $(\pi'(l), \pi'(l+1)) \in E^\alpha \setminus E^{\alpha+1}$. Clearly $\pi'(l) \in V_1$. Hence, σ_1^α is winning in $(\mathcal{G}_1^\alpha, \pi'(l))$. But then σ_1^ξ is also winning in $(\mathcal{G}_1^\xi, \pi'(l))$. As $\pi'[0, l]$ is the prefix of a play of $(\mathcal{G}_1^\xi, v)$ consistent with τ_0, this implies that τ_0 is not winning in $(\mathcal{G}_1^\xi, v)$, a contradiction.

The reasoning for player 0 is analogous. □

We use the same construction as in the last proof to show that finite parity games with arbitrarily many players have a finite-state subgame perfect equilibrium. As demonstrated in the proof of Theorem 3.10, upon deviation of some player all other players have to switch to an optimal counter strategy against this player. In the case of a finite arena, this behaviour can easily be implemented by a strategy automaton.

Theorem 4.20. *Any initialised finite multiplayer parity game has a finite-state subgame perfect equilibrium.*

Proof. Let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\Omega_i)_{i\in\Pi})$ be a finite multiplayer parity game with initial vertex v_0. We define $E^n \subseteq E$ for each $n < \omega$ beginning with

$$E^0 = E.$$

To define E^{n+1} from E^n, for each player $i \in \Pi$, we consider the two-player zero-sum parity game $\mathcal{G}_i^n = (\{i, \Pi \setminus \{i\}\}, V, V_i, V_{\Pi\setminus\{i\}}, E^n, \chi, \Omega_i, \Omega_{\Pi\setminus\{i\}})$ defined by $V_{\Pi\setminus\{i\}} = V \setminus V_i$ and $\Omega_{\Pi\setminus\{i\}} = \Omega_i + 1$, i.e. in $\mathcal{G}_i^n$ player i plays the parity game $\mathcal{G}$ against the coalition $\Pi \setminus \{i\}$ with the set of edges restricted to E^n. By Theorem 4.15, each of the games $\mathcal{G}_i^n$ has a uniform positional Nash equilibrium $(\sigma_i^n, \sigma_{\Pi\setminus\{i\}}^n)$. Let W_i^n be the set of all $v \in V$ such that σ_i^n is winning in $(\mathcal{G}_i^n, v)$. We delete all edges from E^n that are not taken by a winning strategy σ_i^n, i.e. we define

$$E^{n+1} = \bigcap_{i\in\Pi} \{(v, w) \in E^n : v \notin V_i \cap W_i^n \text{ or } w = \sigma_i^n(v)\}.$$

The sequence $(E^n)_{n<\omega}$ is obviously non-increasing. As E is finite, this implies that the sequence has a fixed point $m < \omega$, i.e. we have $E^m = E^{m+1}$. For each player $i \in \Pi$, let $\sigma_i = \sigma_{i,i} = \sigma_i^m$ and $\sigma_{\Pi\setminus\{i\}} = \sigma_{\Pi\setminus\{i\}}^m$. For all players $j \in \Pi \setminus \{i\}$, let $\sigma_{j,i}$ be the positional strategy of player j in $\mathcal{G}$ defined by $\sigma_{j,i}(v) = \sigma_{\Pi\setminus\{i\}}^m(v)$ for all $v \in V_j$. Note that σ_i is winning in $(\mathcal{G}_i^m, v)$ if σ_i^n is winning in $(\mathcal{G}_i^n, v)$ for some $n < \omega$ because, by definition of E^{n+1}, if σ_i^n is winning in $(\mathcal{G}_i^n, v)$, then every play of $(\mathcal{G}_i^{n+1}, v)$ is consistent with σ_i^n and therefore won by player i. As $E^m \subseteq E^{n+1}$, this is also true for $(\mathcal{G}_i^m, v)$.

Now, for each player $j \in \Pi$, we implement a punishment strategy τ_j by a strategy automaton $\mathcal{A}_j$ for player j in $\mathcal{G}$. $\mathcal{A}_j = (Q, q_0, \delta, \tau)$ is defined by

(1) $Q = \Pi \times V \times (\Pi \,\dot\cup\, \{\bot\})$,

(2) $q_0 = (j, v_0, \bot)$,

(3) $\delta((i, u, p), v) = \begin{cases} (i^*, \sigma_{i^*}(v), \bot) & \text{if } p = \bot \text{ and } u = v, \\ (i^*, \sigma_{i^*,p}(v), p) & \text{if } p \neq \bot \text{ and } u = v, \\ (i^*, \sigma_{i^*,i}(v), i) & \text{otherwise,} \end{cases}$

(4) $\tau((i, u, p), v) = \begin{cases} \sigma_j(v) & \text{if } p = \bot \text{ and } u = v, \\ \sigma_{j,p}(v) & \text{if } p \neq \bot \text{ and } u = v, \\ \sigma_{j,i}(v) & \text{otherwise.} \end{cases}$

In the definition of $\delta((i, c, p), v)$, i^* stands for the player with $v \in V_{i^*}$. The meaning of a state (i, u, p) is that at the last vertex it was player i's turn, the expected next vertex is u, and the player to be punished is p (where $p = \bot$ if no player has to be punished). A new player is punished if the vertex she chose is different from the vertex she should have chosen with her prescribed strategy.

We show that $(\tau_j|_h)_{j\in\Pi}$ is a Nash equilibrium of $(\mathcal{G}|_h, v) = (\mathcal{G}, v)$ for every history $hv \in V^*V$ of $(\mathcal{G}, v_0)$. Let hv be a history of $(\mathcal{G}, v_0)$ and $\pi = \langle(\tau_j|_h)_{j\in\Pi}\rangle_v$. Furthermore, let τ' be any strategy of some player i in $\mathcal{G}$ and $\pi' = \langle\tau'|_h, (\tau_j|_h)_{j\in\Pi\setminus\{i\}}\rangle_v$. We have to show that $h\pi$ is won by player i or that $h\pi'$ is not won by player i. The claim is trivial if $\pi = \pi'$. Thus, assume $\pi \neq \pi'$ and fix the least $k < \omega$ such that $\pi(k+1) \neq \pi'(k+1)$. Then $\pi(k) \in V_i$ and $\tau'(h\cdot\pi[0, k+1]) \neq \tau_i(h\cdot\pi[0, k+1])$. Without loss of generality, let $k = 0$. We distuingish the following two cases:

(1) σ_i is winning in $(\mathcal{G}_i^m, v)$. By definition of the strategies τ_j, π is a play of $(\mathcal{G}_i^m, v)$. We show that π is consistent with σ_i which implies that $h\pi$ is won by player i. Otherwise fix the least $l < \omega$ such that $\pi(l) \in V_i$ and $\sigma_i(\pi(l)) \neq \pi(l+1)$. As σ_i is winning in $(\mathcal{G}_i^m, v)$, σ_i is also winning in $(\mathcal{G}_i^m, \pi(l))$. But then $(\pi(l), \pi(l+1)) \in E^m \setminus E^{m+1}$, a contradiction to $E^m = E^{m+1}$.

(2) σ_i is not winning in $(\mathcal{G}_i^m, v)$. Then $\sigma_{\Pi\setminus\{i\}}$ is winning in $(\mathcal{G}_i^m, v)$. As $\tau'(hv) \neq \tau_i(hv)$, by the definition of $\mathcal{A}_j$, for all players $j \in \Pi \setminus \{i\}$ we have $\pi'(l+1) = \tau_j(h \cdot \pi'[0, l+1]) = \sigma_{j,i}(\pi'(l))$ for all $l < \omega$ such that $\pi'(l) \in V_j$. Hence, $\pi' = \langle \tau'|_h, (\sigma_{j,i})_{\Pi\setminus\{i\}} \rangle_v$. We show that π' is a play of $(\mathcal{G}_i^m, v)$. As $\sigma_{\Pi\setminus\{i\}}$ is winning in $(\mathcal{G}_i^m, v)$, this implies that $h\pi'$ is not won by player i. Otherwise fix the least $l < \omega$ with $(\pi'(l), \pi'(l+1)) \notin E^m$ together with the $n < \omega$ such that $(\pi'(l), \pi'(l+1)) \in E^n \setminus E^{n+1}$. Clearly $\pi'(l) \in V_i$. Hence, σ_i^n is winning in $(\mathcal{G}_i^n, \pi'(l))$. But then σ_i is also winning in $(\mathcal{G}_i^m, \pi'(l))$. As $\pi'[0, l]$ is the prefix of a play of $(\mathcal{G}_i^m, v)$ consistent with $\sigma_{\Pi\setminus\{i\}}$, this implies that $\sigma_{\Pi\setminus\{i\}}$ is not winning in $(\mathcal{G}_i^m, v)$, a contradiction.

As $(\tau_j|_h)_{j\in\Pi}$ is a Nash equilibrium of $(\mathcal{G}|_h, v)$ for every history hv of $(\mathcal{G}, v_0)$, $(\tau_j)_{j\in\Pi}$ is a subgame perfect equilibrium of $(\mathcal{G}, v_0)$. □

We have proven the existence of a finite-state subgame perfect equilibrium in any finite multiplayer parity game. In most cases, this result is not satisfactory because our result makes no statement about the payoff of the finite-state equilibrium shown to exist. This raises the question whether finite-state equilibria suffice in general to realise any subgame perfect equilibrium payoff of a finite parity game, i.e. whether for any subgame perfect equilibrium of a finite multiplayer parity game there exists a finite-state subgame perfect equilibrium of the game with the same payoff. We will answer this question positively in Chapter 6. Note that, as demonstrated in Example 4.1, positional subgame perfect equilibria are in general not sufficient to realise an arbitrary subgame perfect equilibrium payoff. Thus it is clear that finite-state strategies are necessary in general.

When we will have shown in Chapter 6 that every subgame perfect equilibrium payoff is realisable by a finite-state subgame perfect equilibrium, Theorem 4.20 will have become a corollary of this result because we have already proven that any multiplayer Borel game has a subgame perfect equilibrium. We have decided to present a direct proof of the theorem here because the proof is a natural refinement of the proof of Theorem 3.10, and moreover it can immediately be turned into an algorithm for computing a finite-state subgame perfect equilibrium of a finite multiplayer parity game. Algorithm 4.1 is derived from the proof in a straightforward way. Thus its correctness follows immediately.

Algorithm 4.1 Computes a finite-state subgame perfect equilibrium of a multiplayer parity game

input multiplayer parity game $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\Omega_i)_{i\in\Pi})$, $v_0 \in V$
$E_{\text{new}} := E$
repeat
  $E_{\text{old}} := E_{\text{new}}$
  for each $i \in \Pi$ **do**
    Compute a uniform positional Nash equilibrium $(\sigma_i, \tau_{\Pi\setminus\{i\}})$ of
      $\mathcal{G}_i = (\{i, \Pi \setminus \{i\}\}, V, V_i, V_{\Pi\setminus\{i\}}, E_{\text{old}}, \chi, \Omega_i, \Omega_{\Pi\setminus\{i\}})$
    $W_i := \{v \in V : \sigma_i \text{ is winning in } (\mathcal{G}_i, v)\}$
    $E_{\text{new}} := E_{\text{new}} \cap \{(v,w) \in E : v \notin V_i \cap W_i \text{ or } \sigma_i(v) = w\}$
  end for
until $E_{\text{new}} = E_{\text{old}}$
for each $i \in \Pi$ **do**
  Compute the strategy automaton $\mathcal{A}_i$ for the finite-state strategy τ_i of
    player i in $(\mathcal{G}, v_0)$
end for
output $(\mathcal{A}_i)_{i\in\Pi}$

Assume we have an algorithm for computing a uniform positional Nash equilibrium of a two-player zero-sum parity game with worst-case running time bounded by $f(n,m,d) \in \Omega(d(n+m))$ for a game with n vertices, m edges and d different priorities. Consequently, each iteration of the repeat loop in Algorithm 4.1 can be implemented in time $O(k \cdot f(n,m,d))$ where k is the number of players and d is the maximal number of different priorities for each player because, given a positional strategy of player 0 in a two-player zero-sum parity game, we can compute in time $O(d(n+m))$ the set of all vertices from where this strategy is winning (see for example [BG04, Proposition 3.3]). As the number of iterations of the loop is bounded by the number of edges, this implies that the running time of the loop is in $O(km \cdot f(n,m,d))$. Each automaton $\mathcal{A}_i$ can be constructed in time $O(k^2n^2)$. Thus the running time of the algorithm is in $O(km \cdot f(n,m,d) + k^3n^2)$. Hence, if there exists a polynomial-time algorithm for computing a uniform positional Nash equilibrium of a two-player zero-sum parity game, then our algorithm can be made to work in polynomial time as well. Note that it even suffices to have a polynomial-time algorithm for computing a positional winning strategy for one player in an initialised two-player zero-sum parity game because such an algorithm makes it possible to compute a uniform Nash equilibrium of the game in polynomial time via the construction in the first proof of Theorem 4.15. Finally, replacing $f(n,m,d)$ by the worst-case running time of Jurdziński's algorithm (Theorem 4.18), we get the following result.

Theorem 4.21. *Computing a finite-state subgame perfect equilibrium of a multiplayer parity game with k players, n vertices, m edges and at most* $d \geq 2$ *priorities for each player can be done in time* $O\big(kdm^2\big(\frac{n}{\lfloor d/2 \rfloor}\big)^{\lfloor d/2 \rfloor} + k^3 n^2\big)$.

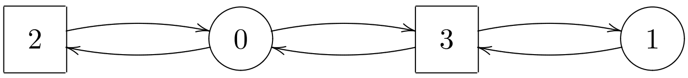

Figure 4.3: Arena of the game from Example 4.3.

Example 4.3. Let $\mathcal{G} = (\{1, 2\}, V, V_1, V_2, E, F_1, F_2)$ be the multiplayer Büchi game with its arena depicted in Figure 4.3. In the figure, circled vertices stand for vertices controlled by player 1, and boxed vertices stand for vertices controlled by player 2. The winning condition of player i is to visit vertex i infinitely often, i.e. $F_i = \{i\}$ for $i = 1, 2$. For every initial vertex v, no player has a winning strategy in $(\mathcal{G}, v)$. Thus Algorithm 4.1 may choose any pair of positional strategies (σ_1, σ_2) for the two players, for example the strategies defined by $\sigma_1(0) = 3$ and $\sigma_2(3) = 0$. The resulting strategy profile has payoff $(0, 0)$. This implies that the the subgame perfect equilibrium computed by the algorithm has payoff $(0, 0)$. However, $(\mathcal{G}, v)$ has a finite-state subgame perfect equilibrium with payoff $(1, 1)$, for example the strategy profile where player 1 moves alternatingly from 0 to 2 and 3 and player 2 moves alternatingly from 3 to 0 and 1.

Example 4.3 shows that Algorithm 4.1 may compute a very poor equilibrium. What we actually want is a maximal subgame perfect equilibrium of the game, i.e. an equilibrium such that there is no subgame perfect equilibrium with a higher payoff. This optimisation problem corresponds to the following decision problem:

> Given a multiplayer graph game $(\mathcal{G}, v_0)$ and a threshold $x \in \{0, 1\}^k$, decide whether $(\mathcal{G}, v_0)$ has a subgame perfect equilibrium with payoff $\geq x$.

If we can solve this decision problem (which should mean that we can also extract a suitable equilibrium from a positive answer), then we can find a maximal subgame perfect equilibrium of $(\mathcal{G}, v_0)$ in the following way. We initialise the threshold with $x = (0)_{i \in \Pi}$. Then for each player i (following any ordering on players), we check if there is an equilibrium with payoff x^i where x^i is defined by $x^i_j = x_j$ for all players $j \neq i$ and $x^i_i = 1$. If the answer is yes, we set x_i to 1 and continue with the next player. In the end, x is the payoff of a maximal subgame perfect equilibrium of $(\mathcal{G}, v_0)$.

Chapter 5

Logical Winning Conditions

We discuss two other formalisms for specifying winning conditions of a graph game. The general idea is to use logical formulae to define winning conditions. The resulting formalisms are very expressive and subsume all previously presented winning conditions.

5.1 Monadic Second-Order Logic

The first formalism we introduce is based on monadic second-order logic, i.e. first-order logic with the ability to quantify over sets. We use it as a framework for specifying properties of infinite words. We assume that the reader is familiar with the basic notions of mathematical logic and go rather quickly through the necessary definitions. For an introduction into mathematical logic, the reader is referred to [EFT84].

Definition 5.1. A *signature* is a finite set of *relational symbols* each one having a certain *arity* $k < \omega$. For a signature $\tau = \{R_1, \dots, R_m\}$, a τ-*structure* is a tuple $\mathfrak{A} = (A, R_1^{\mathfrak{A}}, \dots, R_m^{\mathfrak{A}})$ where A is a non-empty set called the *universe* of $\mathfrak{A}$ and $R_i^{\mathfrak{A}} \subseteq A^k$ if the relation symbol R_i has arity k.

Definition 5.2 (Syntax of MSO)**.** Let $\mathrm{Var} = \{x_0, x_1, \dots\} \dot\cup \{X_0, X_1, \dots\}$ be a fixed, countable set of *variables* consisting of infinitely many *first-order variables* which we write in lower case letters and infinitely many *second-order variables* which we write in upper case letters, and let τ be a signature. The set $\mathrm{MSO}[\tau]$ of *MSO-formulae over τ* is inductively defined by:

(1) If x, y are first-order variables, then $x = y \in \mathrm{MSO}[\tau]$;

(2) if X is a second-order variable and x is a first-order variable, then $Xx \in \mathrm{MSO}[\tau]$;

(3) if $R \in \tau$ is a k-ary relation symbol and $x_1, \dots, x_k$ are first-order variables, then $Rx_1 \dots x_k \in \mathrm{MSO}[\tau]$;

(4) if $\varphi, \psi \in \text{MSO}[\tau]$, then $\neg\varphi \in \text{MSO}[\tau]$ and $(\varphi \vee \psi) \in \text{MSO}[\tau]$;

(5) If $\varphi \in \text{MSO}[\tau]$, x is a first-order variable, and X is a second-order variable, then $\exists x.\varphi \in \text{MSO}[\tau]$ and $\exists X.\varphi \in \text{MSO}[\tau]$.

We omit brackets if they are clear from the context. Moreover, we define 0 as an abbreviation for the formula $\exists x.\neg x = x$, 1 as an abbreviation for $\neg 0$, $(\varphi \wedge \psi)$ as an abbreviation for $\neg(\neg\varphi \vee \neg\psi)$, $(\varphi \rightarrow \psi)$ as an abbreviation for $(\neg\varphi \vee \psi)$, $\forall x.\varphi$ as an abbreviation for $\neg\exists x.\neg\varphi$ and $\forall X.\varphi$ as an abbreviation for $\neg\exists X.\neg\varphi$. Given a finite set $I = \{i_1, \ldots, i_k\}$ and MSO-formulae $\varphi_{i_1}, \ldots, \varphi_{i_k}$, we also write $\bigvee_{i \in I} \varphi_i$ and $\bigwedge_{i \in I} \varphi_i$ for $0 \vee \varphi_{i_1} \vee \ldots \vee \varphi_{i_k}$ and $1 \wedge \varphi_{i_1} \wedge \ldots \wedge \varphi_{i_k}$, respectively.

An occurrence of a variable x or X in a formula φ is called *bound* if it is part of a subformula $\exists x.\psi$ or $\exists X.\psi$, respectively, of φ and *free* otherwise. A *sentence* is a formula in which every occurrence of a variable is bound. We write $\varphi(x_1, \ldots, x_m)$ to indicate that at most the first-order variables $x_1, \ldots, x_m$ occur free in φ.

Definition 5.3 (Semantics of MSO)**.** A pair $(\mathfrak{A}, \beta)$ where $\mathfrak{A}$ is a τ-structure and $\beta : \text{Var} \rightarrow A \cup \mathcal{P}(A)$ is a function such that $\beta(x) \in A$ for first-order variables x and $\beta(X) \in \mathcal{P}(A)$ for second-order variables X is called a τ-*interpretation*. For a τ-interpretation $(\mathfrak{A}, \beta)$, a first-order variable x and an element $a \in A$, let $(\mathfrak{A}, \beta)[x/a]$ be the interpretation $(\mathfrak{A}, \beta')$ with $\beta'(x) = a$ and $\beta'(y) = \beta(y)$ for all $y \neq x$. For a second-order variable X and a subset $P \subseteq A$, the interpretation $(\mathfrak{A}, \beta)[X/P]$ is analogously defined.

Let $(\mathfrak{A}, \beta)$ be a τ-interpretation. Inductively, for every $\varphi \in \text{MSO}[\tau]$ we define whether $(\mathfrak{A}, \beta) \models \varphi$ by

(1) $(\mathfrak{A}, \beta) \models x = y$ if and only if $\beta(x) = \beta(y)$,

(2) $(\mathfrak{A}, \beta) \models Xx$ if and only if $\beta(x) \in \beta(X)$,

(3) $(\mathfrak{A}, \beta) \models Rx_1 \ldots x_k$ if and only if $(\beta(x_1), \ldots, \beta(x_k)) \in R^{\mathfrak{A}}$,

(4) $(\mathfrak{A}, \beta) \models \neg\varphi$ if and only if $(\mathfrak{A}, \beta) \not\models \varphi$,

(5) $(\mathfrak{A}, \beta) \models (\varphi \vee \psi)$ if and only if $(\mathfrak{A}, \beta) \models \varphi$ or $(\mathfrak{A}, \beta) \models \psi$,

(6) $(\mathfrak{A}, \beta) \models \exists x.\varphi$ if and only if $(\mathfrak{A}, \beta)[x/a] \models \varphi$ for some $a \in A$, and

(7) $(\mathfrak{A}, \beta) \models \exists X.\varphi$ if and only if $(\mathfrak{A}, \beta)[X/P] \models \varphi$ for some $P \subseteq A$.

Given a formula $\varphi(x_1, \ldots, x_m) \in \text{MSO}[\tau]$, a τ-structure $\mathfrak{A}$ and elements $a_1, \ldots, a_m \in A$, we write $\mathfrak{A} \models \varphi(a_1, \ldots, a_m)$ if $(\mathfrak{A}, \beta) \models \varphi$ for some (and hereby for all) interpretations $(\mathfrak{A}, \beta)$ with $\beta(x_i) = a_i$ for all $1 \leq i \leq m$. In particular, we write $\mathfrak{A} \models \varphi$ if φ is a sentence and $(\mathfrak{A}, \beta) \models \varphi$ for some interpretation $(\mathfrak{A}, \beta)$. In this case, we say that $\mathfrak{A}$ is a *model of* φ or that φ *holds in* $\mathfrak{A}$.

We use MSO-formulae to define sets of infinite words by assigning to each infinite word over a certain alphabet a structure of an appropriate signature.

Definition 5.4. A signature $\tau = \{S\} \cup \{P_a : a \in \Sigma\}$ where Σ is a non-empty, finite set, S is a binary relation symbol, and each P_a is a unary relation symbol is called a *word signature*.

For a word signature $\tau = \{S\} \cup \{P_a : a \in \Sigma\}$, a *word model* is a τ-structure $\mathfrak{W}$ with universe ω where S is interpreted with the successor relation on ω, i.e. $S^{\mathfrak{W}} = \{(n, n+1) : n < \omega\}$, and the sets $P_a^{\mathfrak{W}}$ define a partition of ω. We identify a word model $\mathfrak{W}$ with the infinite word $\alpha \in \Sigma^\omega$ defined by $\alpha(n) = a$ if and only if $n \in P_a^{\mathfrak{W}}$. For a sentence $\varphi \in \mathrm{MSO}[\tau]$, the set $L(\varphi) = \{\alpha \in \Sigma^\omega : \alpha \models \varphi\}$ is called the *language defined by* φ.

Example 5.1. Let $\tau = \{S\} \cup \{P_a : a \in \Sigma\}$ be a word signature. We use $x \leq y$ as an abbreviation for the formula

$$\forall X.((Xx \wedge \forall u.\forall v.((Xu \wedge Suv) \rightarrow Xv)) \rightarrow Xy)$$

which defines the reflexive, transitive closure of S. Here are some MSO-sentences over τ and their word models.

(1) Given a set $F \subseteq \Sigma$, an infinite word $\alpha \in \Sigma^\omega$ is a model of

$$\bigvee_{a \in F} \exists x.P_a x$$

if and only if $\mathrm{Occ}(\alpha) \cap F \neq \emptyset$.

(2) Given a set $F \subseteq \Sigma$, an infinite word $\alpha \in \Sigma^\omega$ is a model of

$$\bigvee_{a \in F} \forall x.\exists y.(x \leq y \wedge P_a y)$$

if and only if $\mathrm{Inf}(\alpha) \cap F \neq \emptyset$.

(3) Given a priority function $\Omega : \Sigma \rightarrow \omega$, $\alpha \in \Sigma^\omega$ is a model of

$$\bigvee_{\substack{k \in \Omega(\Sigma) \\ k \text{ even}}} \bigvee_{\substack{a \in \Sigma \\ \Omega(a) = k}} \Big(\forall x.\exists y.(x \leq y \wedge P_a y) \wedge \\ \bigwedge_{\substack{l \in \Omega(\Sigma) \\ l < k}} \bigwedge_{\substack{b \in \Sigma \\ \Omega(b) = l}} \exists x.\forall y.(x \leq y \rightarrow \neg P_b y)\Big)$$

if and only if $\min(\Omega(\mathrm{Inf}(\alpha)))$ is even.

(4) For $a \in \Sigma$, an infinite word $\alpha \in \Sigma^\omega$ is a model of

$$\exists X.(\forall x.(\forall y.\neg Syx \rightarrow Xx) \wedge \\ \forall x.\forall y.\forall z.((Xx \wedge Sxy \wedge Syz) \rightarrow (\neg Xy \wedge Xz)) \wedge \\ \forall x.(P_a x \rightarrow Xx))$$

if and only if $\alpha(n) = a$ only for even $n < \omega$.

To use MSO-formulae as winning conditions for games, we use again a colouring of the game graph into a finite set of colours.

Definition 5.5. A game $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ is called a *multiplayer MSO-game* if there exists a function $\chi : V \to C$ for some finite, non-empty set C and for each player $i \in \Pi$ an MSO-sentence φ_i over the word signature $\{S\} \cup \{P_c : c \in C\}$ such that for all players $i \in \Pi$ we have $W_i = \{\pi \in V^\omega : \chi(\pi) \models \varphi_i\}$. In this case, we write $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\varphi_i)_{i\in\Pi})$.

Given the example formulae (1),(2) and (3) from Example 5.1, it is obvious that any multiplayer reachability, Büchi, or parity game is also a multiplayer MSO-game. As MSO-formulae are closed under negation, this also holds for multiplayer safety and co-Büchi games. On the other hand, formula (4) defines a set of words $\alpha \in \Sigma^\omega$ that cannot be defined by a condition on $\mathrm{Occ}(\alpha)$ or $\mathrm{Inf}(\alpha)$. However, we will show that every multiplayer MSO-game can be reduced to a multiplayer parity game, though the reduction involves a blow-up of the game graph.

5.2 Linear Temporal Logic

Another popular logic for specifying properties of infinite sequences is linear temporal logic (LTL) as introduced by Pnueli [Pnu77]. LTL is not as expressive as MSO, but strong enough to express most properties used in verification. It has an easier syntax than MSO which avoids the use of variables and is much better tractable than MSO.

Definition 5.6 (Syntax of LTL)**.** Let $P = \{p_0, \dots, p_{m-1}\}$ be a finite, non-empty set of *propositions*. The set $\mathrm{LTL}[P]$ of *LTL-formulae over* P is inductively defined by:

(1) If $p \in P$, then $p \in \mathrm{LTL}[P]$;

(2) If $\varphi, \psi \in \mathrm{LTL}[P]$, then $\neg\varphi \in \mathrm{LTL}[P]$ and $(\varphi \vee \psi) \in \mathrm{LTL}[P]$;

(3) If $\varphi, \psi \in \mathrm{LTL}[P]$, then $\mathrm{X}\varphi \in \mathrm{LTL}[P]$ and $(\varphi \mathrm{U} \psi) \in \mathrm{LTL}[P]$.

As with MSO, we omit brackets if they are clear from the context and define some abbreviations. We write 0 as an abbreviation for $\neg(p_0 \vee \neg p_0)$, 1 as an abbreviation for $\neg 0$, $(\varphi \wedge \psi)$ as an abbreviation for $\neg(\neg\varphi \vee \neg\psi)$, $\mathrm{F}\varphi$ as an abbreviation for $1\mathrm{U}\varphi$ and $\mathrm{G}\varphi$ as an abbreviation for $\neg\mathrm{F}\neg\varphi$.

Definition 5.7 (Semantics of LTL)**.** We interpret $\mathrm{LTL}[\{p_0, \dots, p_{m-1}\}]$-formulae over ω-words $\alpha \in (\{0,1\}^m)^\omega$ where $\alpha(j)(i) = 1$ has the meaning that proposition p_i is true at position j of α. Let $\alpha \in (\{0,1\}^m)^\omega$ and $j < \omega$. Inductively, we define for every $\varphi \in \mathrm{LTL}[\{p_0, \dots, p_{m-1}\}]$ whether $\alpha, j \models \varphi$ by

(1) $\alpha, j \models p_i$ if and only if $\alpha(j)(i) = 1$,

(2) $\alpha, j \models \neg\varphi$ if and only if $\alpha, j \not\models \varphi$,

(3) $\alpha, j \models (\varphi \vee \psi)$ if and only if $\alpha, j \models \varphi$ or $\alpha, j \models \psi$,

(4) $\alpha, j \models \mathrm{X}\varphi$ if and only if $\alpha, j+1 \models \varphi$, and

(5) $\alpha, j \models (\varphi \mathrm{U} \psi)$ if and only if there exists $k \geq j$ such that $\alpha, k \models \psi$ and $\alpha, l \models \varphi$ for all $j \leq l < k$.

We write $\alpha \models \varphi$ if $\alpha, 0 \models \varphi$. In this case, we say that α is a *model of* φ or that φ *holds in* α. The set $L(\varphi) = \{\alpha \in (\{0,1\}^m)^\omega : \alpha \models \varphi\}$ is called the *language defined by* φ.

Example 5.2. Let $P = \{p_0, \ldots, p_{m-1}\}$. We give some example LTL-formulae and their models.

(1) An infinite word $\alpha \in (\{0,1\}^m)^\omega$ is a model of the formula

$$\mathrm{F}(p_0 \wedge \neg p_{m-1})$$

if and only there exists a position of α where p_0 is true but p_{m-1} is not.

(2) An infinite word $\alpha \in (\{0,1\}^m)^\omega$ is a model of the formula

$$\mathrm{GF}(p_0 \wedge \neg p_{m-1})$$

if and only there exist infinitely many positions of α where p_0 is true but p_{m-1} is not.

(3) An infinite word $\alpha \in (\{0,1\}^m)^\omega$ is a model of the formula

$$p_0 \wedge \mathrm{X}p_0 \wedge \mathrm{XX}(\neg p_0 \mathrm{U} p_{m-1})$$

if and only if p_0 is true at positions 0 and 1 of α and there exists a position $k \geq 2$ where p_{m-1} is true and at all positions inbetween p_0 is not true.

The following proposition shows that LTL can be understood as a fragment of MSO (interpreted over words $\alpha \in (\{0,1\}^m)^\omega$).

Proposition 5.8. *Let* $P = \{p_0, \ldots, p_{m-1}\}$ *be a non-empty, finite set. For every* $\varphi \in \mathrm{LTL}[P]$, *there exists an MSO-sentence* $\widetilde{\varphi}$ *over the word signature* $\{S\} \cup \{P_a : a \in \{0,1\}^m\}$ *such that* $\alpha \models \varphi \Leftrightarrow \alpha \models \widetilde{\varphi}$ *for all* $\alpha \in (\{0,1\}^m)^\omega$.

Proof. Inductively, for every $\varphi \in \mathrm{LTL}[P]$ we define an MSO-formula $\overline{\varphi}(x)$ such that $\alpha, j \models \varphi \Leftrightarrow \alpha \models \overline{\varphi}(j)$ for all $\alpha \in (\{0,1\}^m)^\omega$ and $j < \omega$. Given $\overline{\varphi}$, we can define $\widetilde{\varphi} = \exists x.(\forall y.\neg Syx \wedge \overline{\varphi}(x))$. For propositions p_i, we define

$$\overline{p_i}(x) = \bigvee_{\substack{a \in \{0,1\}^m \\ a(i)=1}} P_a x.$$

Negations and disjunctions are translated directly, i.e. $\overline{\neg\varphi}(x) = \neg\overline{\varphi}(x)$ and $\overline{(\varphi \vee \psi)}(x) = (\overline{\varphi}(x) \vee \overline{\psi}(x))$. For a formula $\mathrm{X}\varphi$, we define

$$\overline{\mathrm{X}\varphi}(x) = \exists y.(Sxy \wedge \overline{\varphi}(y)).$$

Here $\overline{\varphi}(y)$ stands for the formula that is created from $\overline{\varphi}(x)$ by replacing all free occurrences of x by y. Finally, for a formula $(\varphi \mathrm{U} \psi)$ we define

$$\overline{(\varphi \mathrm{U} \psi)}(x) = \exists y.(x \leq y \wedge \overline{\psi}(y) \wedge \forall z.((x \leq z \wedge z < y) \rightarrow \overline{\varphi}(z)))$$

where $\leq$ is defined as in Example 5.1 and $z < y$ is an abbreviation for $z \leq y \wedge \neg z = y$. It is obvious from the semantics of LTL that this definition fulfils the claim. □

To separate LTL from MSO, one introduces the notion of counting. A language $L \subseteq \Sigma^\omega$ is *non-counting* if there exists an $n < \omega$ such that for all $k > n$, all $v, w \in \Sigma^*$ and all $\alpha \in \Sigma^\omega$ we have

$$vw^k\alpha \in L \Leftrightarrow vw^{k+1}\alpha \in L.$$

This means that for $k > n$ either all $vw^k\alpha$ are in L or none is. By induction on φ, it is easy to prove that every language defined by an LTL-formula φ is non-counting. On the other hand, the language $L \subseteq \{0,1\}^\omega$ defined by the MSO-formula from Example 5.1 (4) (with $\Sigma = \{0,1\}$ and $a = 1$) is *counting*, i.e. not non-counting (take $v = \varepsilon$, $w = 0$ and $\alpha = 10^\omega$).

A full characterisation of LTL in MSO was given by Kamp [Kam68]. Observe that the translation from LTL to MSO does not use any second-order quantification except for its implicit usage to define $\leq$. Kamp showed that the converse also holds: Every language definable in first-order logic with the additional binary relation symbol $\leq$ interpreted with the natural well-ordering on ω is also definable in LTL.

To use LTL-formulae as winning conditions for games, we require that the colouring function maps into $\{0,1\}^m$ for some $m < \omega$.

Definition 5.9. A game $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ is called a *multi-player LTL-game* if there exists a function $\chi : V \rightarrow \{0,1\}^m$ for some $m < \omega$ and for each player $i \in \Pi$ a formula $\varphi_i \in \mathrm{LTL}[p_0, \ldots, p_{m-1}]$ such that for all players $i \in \Pi$ we have $W_i = \{\pi \in V^\omega : \chi(\pi) \models \varphi_i\}$. In this case, we write $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\varphi_i)_{i\in\Pi})$.

As with MSO, given the example formulae (1) and (2) from Example 5.2, it should be obvious that any multiplayer reachability, safety, (co-)Büchi, or parity game is also a multiplayer LTL-game. On the other hand, formula (3) defines a set of words $\alpha \in (\{0,1\}^m)^\omega$ that cannot be defined by a condition on $\mathrm{Occ}(\alpha)$ or $\mathrm{Inf}(\alpha)$. Given the translation from LTL to MSO, it is also clear that every multiplayer LTL-game is also a multiplayer MSO-game.

5.3 Automata on Infinite Words

The automata-theoretic approach has been extremely successful when dealing with monadic second-order logic over tree-like structures. In general, it allows us to translate an MSO-formula into an automaton that, given a class of tree-like structures, accepts precisely those structures in the given class that are a model of the formula. In the case of MSO interpreted over word models, one can use usual finite automata with an acceptance condition for infinite words. We use the Büchi and the parity condition for this.

Definition 5.10. A *parity automaton* is a tuple $\mathcal{A} = (Q, \Sigma, q_0, \Delta, \Omega)$ where

(1) Q is a non-empty, finite set of *states*,

(2) Σ is a non-empty, finite set,

(3) $q_0 \in Q$ is the *initial state*,

(4) $\Delta \subseteq Q \times \Sigma \times Q$ is the *transition relation*, and

(5) $\Omega : Q \to \omega$ is the *priority function*.

$\mathcal{A}$ is called a *Büchi automaton* if $\Omega(Q) \subseteq \{0, 1\}$. $\mathcal{A}$ is called *deterministic* if for every pair $(p, a) \in Q \times \Sigma$ there exists precisely one $q \in Q$ with $(p, a, q) \in \Delta$. In this case, we usually replace Δ by a *transition function* $\delta : Q \times \Sigma \to Q$.

A *run of* $\mathcal{A}$ on an infinite word $\alpha \in \Sigma^\omega$ is an infinite word $\rho \in Q^\omega$ such that $\rho(0) = q_0$ and $(\rho(k), \alpha(k), \rho(k+1)) \in \Delta$ for all $k < \omega$. A run ρ is *accepting* if $\min(\Omega(\mathrm{Inf}(\rho)))$ is even. $\mathcal{A}$ *accepts* an infinite word $\alpha \in \Sigma^\omega$ if there exists an accepting run of $\mathcal{A}$ on α. By $L(\mathcal{A})$, we denote the set of all $\alpha \in \Sigma^\omega$ such that $\mathcal{A}$ accepts α. $L(\mathcal{A})$ is called the *language defined by* $\mathcal{A}$.

Büchi showed in his seminal paper [Büc62] that MSO and Büchi automata are expressively equivalent over infinite words. Here, we are only interested in the translation of MSO-formulae into Büchi automata. The translation is *effective* in the sense that there is an algorithm computing the automaton for any given input formula.

Theorem 5.11 (Büchi)**.** *For any MSO-sentence φ over a word signature, one can effectively construct a Büchi automaton $\mathcal{A}$ such that $L(\mathcal{A}) = L(\varphi)$.*

It should be noted that the size of the automaton may be very large. The number of states of the automaton can only be bounded by

$$2^{2^{\cdot^{\cdot^{\cdot^{O(|\varphi|)}}}}}\Big\}q+1$$

where q is the number of quantifier alternations in the given formula φ (see [GTW02, Chapter 11]).

There is one problem left. Büchi automata are in general non-deterministic, but we will need deterministic automata equivalent to MSO-formulae. Unfortunately, deterministic Büchi automata are not able to define all MSO-definable languages, i.e. there is an MSO-definable language L (for example $L = \{0,1\}^*1^\omega$) such that $L(\mathcal{A}) \neq L$ for any deterministic Büchi automaton $\mathcal{A}$. However, it can be shown that Büchi automata, deterministic parity automata and parity automata are able to define precisely the same languages. Thus, our preferred automaton model is that of a deterministic parity automaton.

There are several constructions for determinising parity automata, the best one being a combination of *Safra's construction* [Saf88] and an *index appearance record (IAR)* construction (see [Saf92]). If the given parity automaton has n states and d priorities, then the resulting deterministic parity automaton has $2^{O(nd \log nd)}$ states and $O(nd)$ priorities[1].

Theorem 5.12 (Safra)**.** *Given a parity automaton $\mathcal{A}$ with n states and d priorities, one can effectively construct a deterministic parity automaton $\mathcal{B}$ with $2^{O(nd \log nd)}$ states and $O(nd)$ priorities such that $L(\mathcal{B}) = L(\mathcal{A})$.*

A corollary is that we can transform a Büchi automaton with n states into a deterministic parity automaton with $2^{O(n \log n)}$ states and $O(n)$ priorities. Michel [Mic88] and Löding [Löd99] showed that this bound is optimal.

Corollary 5.13. *Given a Büchi automaton $\mathcal{A}$ with n states, one can effectively construct a deterministic parity automaton $\mathcal{B}$ with $2^{O(n \log n)}$ states and $O(n)$ priorities such that $L(\mathcal{B}) = L(\mathcal{A})$.*

As we know the topological complexity of the parity condition, we can determine the complexity of MSO-definable languages and MSO-games.

Corollary 5.14. *Any language defined by an MSO-formula is contained in Δ^0_3. In particular, multiplayer MSO-games are Δ^0_3-games.*

[1] A parity automaton with n states and d priorities can be transformed into a Büchi automaton with $O(nd)$ states. Safra's construction transforms a Büchi automaton with n states into a deterministic Rabin automaton (see [Rab72]) with $2^{O(n \log n)}$ states and $2n$ accepting pairs. The IAR construction transforms a deterministic Rabin automaton with n states and r accepting pairs into a deterministic parity automaton with $n \cdot 2^{O(r \log r)}$ states and $O(r)$ priorities.

Proof. Let $L = L(\varphi) \subseteq \Sigma^\omega$ for an MSO-sentence φ over a word signature $\tau = \{S\} \cup \{P_a : a \in \Sigma\}$. By Theorem 5.11 and Theorem 5.12, there exists a deterministic parity automaton $\mathcal{A} = (Q, \Sigma, q_0, \delta, \Omega)$ such that $L(\mathcal{A}) = L$. The set A of infinite words $\rho \in Q^\omega$ satisfying the parity condition is a Δ^0_3-set (see above). We extend δ to a function $\delta : \Sigma^\omega \to Q^\omega$ by defining $\delta(a_0 a_1 a_2 \ldots) = q_0 q_1 q_2 \ldots$ where $q_{i+1} = \delta(q_i, a_i)$. Thus, δ maps an infinite word $\alpha \in \Sigma^\omega$ to the unique run of $\mathcal{A}$ on α. Clearly, δ is continuous with $\delta : L \leq A$, and hence $L \in \Delta^0_3$.

Now let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\varphi_i)_{i\in\Pi})$ be a multiplayer MSO-game. By definition of $\mathcal{G}$, the set of plays won by player i is $W_i = \chi^{-1}(L(\varphi_i))$. As χ is continuous with $\chi : W_i \leq L(\varphi_i)$, we have $W_i \in \Delta^0_3$. As this holds for any player i, $\mathcal{G}$ is a Δ^0_3-game. □

The naïve way of translating LTL into Büchi automata would be to write the given formula as an equivalent MSO-formula and to apply the translation for MSO-formula. However, every subformula $\varphi \mathrm{U} \psi$ introduces quantifier alternation. Hence, the translation suffers from the high complexity of translating MSO-formulae into Büchi-automata. The nice thing about LTL is that we can do much better. Vardi et al. [VWS83] showed that a Büchi automaton with exponentially many states in the length of the formula suffices.

Theorem 5.15 (Vardi et al.)**.** *For any LTL-formula φ, one can effectively construct a Büchi-automaton $\mathcal{A}$ with $2^{O(|\varphi|)}$ states such that $L(\mathcal{A}) = L(\varphi)$.*

Using Corollary 5.13 to transform the resulting Büchi automaton into a deterministic parity automaton, we end up with a parity automaton with $2^{2^{O(|\varphi|)}}$ states and $2^{O(|\varphi|)}$ priorities, a result originally due to Emerson and Sistla [ES84].

Corollary 5.16 (Emerson, Sistla)**.** *For any LTL-formula φ, one can effectively construct a deterministic parity automaton $\mathcal{A}$ with $2^{2^{O(|\varphi|)}}$ states and $2^{O(|\varphi|)}$ priorities such that $L(\mathcal{A}) = L(\varphi)$.*

5.4 The Reduction

We are now able to reduce multiplayer MSO-games to multiplayer parity games. The arena of the parity game is the product of the arena of the original MSO-game with the parity automata defining the players' winning conditions.

Claim 5.17. *For any initialised multiplayer MSO-game $(\mathcal{G}, v_0)$, there exists an initialised multiplayer parity game $(\mathcal{G}', v_0')$ with $(\mathcal{G}, v_0) \leq_{\mathrm{fin}} (\mathcal{G}', v_0')$.*

Proof. Let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\varphi_i)_{i\in\Pi})$ where $\chi : V \to C$. By Theorem 5.11 and Theorem 5.12, for each player $i \in \Pi$ there exists a deterministic parity automaton $\mathcal{A}_i = (Q_i, C, q_{0i}, \delta_i, \Omega_i)$ with $L(\mathcal{A}_i) = L(\varphi_i)$. The multiplayer parity game $\mathcal{G}' = (\Pi, V', (V'_i)_{i\in\Pi}, E', \chi', (\Omega'_i)_{i\in\Pi})$ has vertex set

$$V' = V \times \prod_{i\in\Pi} Q_i$$

where $(v,p) \in V'_i \Leftrightarrow v \in V_i$. The colouring function $\chi' : V' \to \prod_{i\in\Pi} Q_i$ maps a vertex (v,p) to p. The priority function Ω'_i maps a tuple $(p_i)_{i\in\Pi}$ to the priority $\Omega_i(p_i)$ of the automaton state p_i. It remains to specify E'. $((v,(p_i)_{i\in\Pi}),(w,(q_i)_{i\in\Pi}))$ is in E' if and only if $(v,w) \in E$ and

$$\delta_i(p_i, \chi(v)) = q_i$$

for all $i \in \Pi$. The initial vertex of $\mathcal{G}'$ is $v'_0 = (v_0, (q_{0i})_{i\in\Pi})$. Thus, in a play π, $\mathcal{G}'$ simulates each parity automaton $\mathcal{A}_i$ on the corresponding play of $\mathcal{G}$ (or more precisely on the corresponding word of colours). It is straightforward to verify that $(\mathcal{G}, v_0) \leq_{\text{fin}} (\mathcal{G}', v'_0)$. □

We can use the reduction to prove the existence of a finite-state subgame perfect equilibrium in any finite multiplayer MSO-game.

Corollary 5.18. *Any initialised finite multiplayer MSO-game has a finite-state subgame perfect equilibrium.*

Proof. This is a direct consequence of Theorem 4.20 together with Claim 5.17 and Lemma 4.11. □

As, by Lemma 2.14, in a Nash equilibrium of a two-player zero-sum game, either one of the two strategies is a winning strategy, this also shows that finite two-player zero-sum MSO-games are finite-state determined, a result due to Büchi and Landweber [BL69].

Corollary 5.19 (Büchi, Landweber)**.** *In any initialised finite two-player zero-sum MSO-game, either one of the two players has a finite-state winning strategy.*

Using Theorem 4.21, the reduction also gives an algorithm for computing a finite-state subgame perfect equilibrium of a finite multiplayer MSO-game. However, the reduction suffers from the high complexity of translating MSO-formulae into parity automata. For multiplayer LTL-games, we can exploit Theorem 5.15 and Corollary 5.16 to construct a parity game with an arena of size linear in the size of the original arena and double-exponential in the size of the LTL-formulae specifying the winning conditions and a maximal number of priorities single-exponential in the size of the formulae. All in all, this gives a double-exponential time bound for the algorithm.

Corollary 5.20. *Computing a finite-state subgame perfect equilibrium of a finite multiplayer LTL-game can be done in double-exponential time.*

Chapter 6

Decision Problems

At the end of Chapter 4, we discussed that a solution of the problem of deciding whether a multiplayer graph game has a subgame perfect equilibrium with payoff at least x for a given threshold x gives us a way to find an optimal subgame perfect equilibrium of the game. In fact, we will study a stronger variant of this decision problem, namely to decide whether a multiplayer graph game has a subgame perfect equilibrium with payoff at least x and at most y for given thresholds x and y. We call this problem the *subgame perfect equilibrium threshold problem.* We will solve the problem for finite multiplayer parity games using a reduction to the non-emptiness problem of a parity tree automaton.

6.1 Tree Automata

We represent the complete binary tree by the set $\{0,1\}^*$. The root of the tree is the empty word ε, the left successor of a tree node w is $w0$, and the right successor of a tree node w is $w1$. The tree nodes are labelled with elements of some alphabet.

Definition 6.1. For any set Σ, a *complete* Σ*-valued binary tree* is a function $t : \{0,1\}^* \to \Sigma$. A *path through* t is an infinite word $\pi \in \{0,1\}^\omega$. For a path π through t, we denote by $t|_\pi$ the infinite word $\alpha \in \Sigma^\omega$ defined by $\alpha(k) = t(\pi[0,k])$ for all $k < \omega$. By $\mathcal{T}_\Sigma^\omega$, we denote the set of all complete Σ-valued binary trees.

Definition 6.2. A *parity tree automaton* is a tuple $\mathcal{A} = (Q, \Sigma, q_0, \Delta, \Omega)$ where

(1) Q is a non-empty, finite set of *states*,

(2) Σ is a non-empty, finite set,

(3) $q_0 \in Q$ is the *initial state*,

(4) $\Delta \subseteq Q \times \Sigma \times Q \times Q$ is the *transition relation*, and

(5) $\Omega : Q \to \omega$ is the *priority function*.

$\mathcal{A}$ is called a *Büchi tree automaton* if $\Omega(Q) \subseteq \{0, 1\}$.

A *run* of $\mathcal{A}$ on a tree $t \in \mathcal{T}^\omega_\Sigma$ is a tree $s \in \mathcal{T}^\omega_Q$ such that $s(\varepsilon) = q_0$ and $(s(w), t(w), s(w0), s(w1)) \in \Delta$ for all $w \in \{0, 1\}^*$. A run s of $\mathcal{A}$ is *accepting* if $\min(\Omega(\mathrm{Inf}(s|_\pi)))$ is even for all paths π through s. $\mathcal{A}$ accepts a tree $t \in \mathcal{T}^\omega_\Sigma$ if there exists an accepting run of $\mathcal{A}$ on t. By $T(\mathcal{A})$, we denote the set of all $t \in \mathcal{T}^\omega_\Sigma$ such that $\mathcal{A}$ accepts t. $T(\mathcal{A})$ is called the *tree language defined by* $\mathcal{A}$. A *tree language* $T \subseteq \mathcal{T}^\omega_\Sigma$ is called *regular* if $T = T(\mathcal{A})$ for some parity tree automaton $\mathcal{A}$.

As proposed by Gurevich and Harrington [GH82], it is convenient to describe the behaviour of a tree automaton on an input tree by a two-player zero-sum game played on the tree. Any play starts at the root of t in the initial state of the automaton. Whenever a play arrives at a node w of the tree and the automaton is in state q, then player 0 has to choose a suitable transition $(q, t(w), p_0, p_1)$. It is then player 1's task to decide whether the play should continue at the left successor of w in state p_0 or at the right successor of w in state p_1. If there is no suitable transition available for player 0, then she must move to a sink vertex $\perp$. Player 0 wins an infinite play of the game if $\perp$ is never visited and the corresponding sequence of automaton states satisfies the acceptance condition of the automaton.

Definition 6.3. Let $\mathcal{A} = (Q, \Sigma, q_0, \Delta, \Omega)$ be a parity tree automaton and t a complete Σ-valued binary tree. We define the two-player zero-sum parity game $\mathcal{G}_{\mathcal{A},t} = (V, V_0, V_1, E, \chi, \Omega_0, \Omega_1)$ as follows. The vertex set of $\mathcal{G}_{\mathcal{A},t}$ is

$$V = ((Q \cup \Delta) \times \{0, 1\}^*) \cup \{\perp\}$$

where the set of vertices controlled by player 0 is

$$V_0 = Q \times \{0, 1\}^*.$$

The colouring function $\chi : V \to Q \mathbin{\dot\cup} \{\perp\}$ is defined by $\chi((q, w)) = q$, $\chi(((q, a, p_0, p_1), w)) = q$ and $\chi(\perp) = \perp$. The priority function of player 0 is defined by $\Omega_0(q) = \Omega(q)$ and $\Omega_0(\perp) = 1$, and the priority function of player 1 is defined by $\Omega_1(q) = \Omega(q) + 1$ and $\Omega_1(\perp) = 0$. It remains to specify the edge relation. E consists of all pairs $((q, w), ((q, t(w), p_0, p_1), w))$ with $(q, t(w), p_0, p_1) \in \Delta$, all pairs $((q, w), \perp)$, the pair $(\perp, \perp)$ and all pairs $(((q, a, p_0, p_1), w), (p_i, wi))$ where $i \in \{0, 1\}$.

The following lemma should be obvious from the definition of $\mathcal{G}_{\mathcal{A},t}$, and we will not prove it here.

Lemma 6.4. *For any parity tree automaton $\mathcal{A} = (Q, \Sigma, q_0, \Delta, \Omega)$ and any tree $t \in \mathcal{T}^\omega_\Sigma$, we have $t \in T(\mathcal{A})$ if and only if player 0 has a winning strategy in the game $(\mathcal{G}_{\mathcal{A},t}, (q_0, \varepsilon))$.*

It is easy to show that the class of regular tree languages is closed under union and projection (i.e. if $T \subseteq \mathcal{T}^{\omega}_{\Sigma\times\Gamma}$ is regular, then so is the tree language $\overline{T} = \{\overline{t} : t \in T\} \subseteq \mathcal{T}^{\omega}_{\Sigma}$ where for a tree $t \in \mathcal{T}^{\omega}_{\Sigma\times\Gamma}$ the tree $\overline{t} : \{0,1\}^* \to \Sigma$ is defined as the elementwise projection of t on the first component). However, the result that this class is also closed under complementation is a highly non-trivial result, which is due to Rabin [Rab69].

Theorem 6.5 (Rabin)**.** *The class of regular tree languages is closed under complementation, i.e. if $T \subseteq \mathcal{T}^{\omega}_{\Sigma}$ is regular, then $\mathcal{T}^{\omega}_{\Sigma} \setminus T$ is also regular.*

We will sketch a proof of Rabin's theorem due to Thomas [Tho97] and Zielonka [Zie98]. A central argument in the proof is positional determinacy of the parity game $\mathcal{G}_{\mathcal{A},t}$ which follows from Theorem 4.13. The use of this argument makes the proof a lot simpler than the original proof by Rabin[1]. The proof of Thomas and Zielonka reduces the problem of complementing a parity tree automaton to the problem of complementing a parity word automaton, which can be achieved by first making the automaton deterministic and then dualising the acceptance condition. This gives rise to the following stronger version of Theorem 6.5.

Theorem 6.6. *Given a parity tree automaton $\mathcal{A}$ with n states and d priorities, one can effectively construct a parity tree automaton $\mathcal{B}$ with $2^{O(nd\log nd)}$ states and $O(nd)$ priorities such that $T(\mathcal{B}) = \mathcal{T}^{\omega}_{\Sigma} \setminus T(\mathcal{A})$.*

Proof. We sketch the proof of Theorem 6.6 following [GTW02, Chapter 8]. Let $\mathcal{A} = (Q, \Sigma, q_0, \Delta, \Omega)$. We have to construct a parity tree automaton $\mathcal{B}$ such that $\mathcal{B}$ accepts some tree $t \in \mathcal{T}^{\omega}_{\Sigma}$ if and only if $\mathcal{A}$ does not accept t. By Lemma 6.4 and the positional determinacy of two-player zero-sum parity games (Theorem 4.13), we can reformulate this in the following way: $\mathcal{B}$ must accept t if and only if player 1 has a positional winning strategy in the initialised parity game $(\mathcal{G}_{\mathcal{A},t}, (q_0, \varepsilon))$.

A positional strategy of player 1 in $\mathcal{G}_{\mathcal{A},t}$ can be encoded as a function $s : \{0,1\}^* \to \{0,1\}^{\Delta}$ determining for every tree node whether to continue the play at the left (0) or the right successor (1) depending on the transition chosen by player 0. Thus, a positional strategy of player 1 in $\mathcal{G}_{\mathcal{A},t}$ is again a complete binary tree s labelled with elements of $\{0,1\}^{\Delta}$. Let $\Sigma' = \Sigma \times \{0,1\}^{\Delta}$ and let $t\hat{\ }s : \{0,1\}^* \to \Sigma'$ be the tree defined by $t\hat{\ }s(w) = (t(w), s(w))$. We construct a parity tree automaton $\mathcal{B}' = (Q', \Sigma', q_0', \Delta', \Omega')$ accepting precisely those trees $t\hat{\ }s \in \mathcal{T}^{\omega}_{\Sigma'}$ over the extended alphabet Σ' such that s is a winning strategy of player 1 in $(\mathcal{G}_{\mathcal{A},t}, (q_0, \varepsilon))$. Then we can define $\mathcal{B} = (Q', \Sigma, q_0', \Delta'', \Omega')$ where $(q, a, p_0, p_1) \in \Delta''$ if and only if $(q, (a, f), p_0, p_1) \in \Delta'$ for some $f \in \{0,1\}^{\Delta}$. Note that $\mathcal{B}$ uses the same state set and the same priority function as $\mathcal{B}'$.

[1]The idea of using determinacy of games in a proof of Rabin's theorem goes back to Gurevich and Harrington [GH82].

We observe that, given a tree $t\hat{\ }s \in \mathcal{T}^{\omega}_{\Sigma'}$, s is a winning strategy of player 1 in $(\mathcal{G}_{\mathcal{A},t}, (q_0, \varepsilon))$ if and only if for all possible moves of player 0 in the game the unique path π through $t\hat{\ }s$ determined by player 0's move sequence violates the parity condition given by Ω. A reformulation of this is that s is a winning strategy of player 1 in $(\mathcal{G}_{\mathcal{A},t}, (q_0, \varepsilon))$ if and only if for all paths $\pi \in \{0,1\}^\omega$ through t we have that, if π is generated by a possible move sequence of player 0 in $(\mathcal{G}_{\mathcal{A},t}, (q_0, \varepsilon))$, then π violates the parity condition given by Ω. This implication can be understood as a condition on infinite words $\alpha \in (\Sigma' \times \{0,1\})^\omega$. We show that it can be checked by a deterministic parity word automaton $\mathcal{W}$.

The set of infinite words $\alpha \in (\Sigma' \times \{0,1\})^\omega$ *not* satisfying the above condition is defined by the parity word automaton $\mathcal{V} = (Q, \Sigma' \times \{0,1\}, q_0, \widetilde{\Delta}, \Omega)$ where $(q, (a, f, i), p) \in \widetilde{\Delta}$ if and only if there exists $\tau = (q, a, p_0, p_1) \in \Delta$ such that $f(\tau) = i$ and $p = p_i$. Note that $\mathcal{V}$ has the same state set and priority function as the given tree automaton $\mathcal{A}$. By Theorem 5.12, there exists a deterministic parity automaton $\mathcal{V}' = (Q', \Sigma' \times \{0,1\}, q_0, \delta, \overline{\Omega})$ with $2^{O(nd \log nd)}$ states and $O(nd)$ priorities such that $L(\mathcal{V}') = L(\mathcal{V})$. Now we can define $\mathcal{W} = (Q', \Sigma' \times \{0,1\}, q_0, \delta, \Omega')$ by $\Omega'(q) = \overline{\Omega}(q) + 1$ for all $q \in Q'$.

Finally, we can define $\mathcal{B}' = (Q', \Sigma', q_0', \Delta', \Omega')$. $\mathcal{B}'$ simulates $\mathcal{W}$ on each path of the input tree, i.e. there is a transition $(q, (a, f), p_0, p_1) \in \Delta'$ if and only if $\delta(q, (a, f, 0)) = p_0$ and $\delta(q, (a, f, 1)) = p_1$. □

Now we turn to the non-emptiness problem for parity tree automata, i.e. the problem to decide whether $T(\mathcal{A}) \neq \emptyset$ for a parity tree automaton $\mathcal{A}$. To solve this problem, we associate another two-player zero-sum parity game $\mathcal{G}_\mathcal{A}$ with any parity tree automaton $\mathcal{A}$ such that $T(\mathcal{A}) \neq \emptyset$ if and only if player 0 has a winning strategy in $\mathcal{G}_\mathcal{A}$ from a given initial vertex.

Definition 6.7. For a parity tree automaton $\mathcal{A} = (Q, \Sigma, q_0, \Delta, \Omega)$ we define the two-player zero-sum parity game $\mathcal{G}_\mathcal{A} = (\{0,1\}, V, V_0, V_1, E, \chi, \Omega_0, \Omega_1)$ as follows. The vertex set of $\mathcal{G}_\mathcal{A}$ is

$$V = (Q \cup \Delta) \mathbin{\dot\cup} \{\bot\}$$

with the set of vertices controlled by player 0 being

$$V_0 = Q.$$

The colouring function $\chi : V \to Q \cup \{\bot\}$ is defined by $\chi(q) = q$, $\chi(\bot) = \bot$ and $\chi((q, a, p_0, p_1)) = q$. The priority function of player 0 is defined by $\Omega_0(q) = \Omega(q)$ and $\Omega_0(\bot) = m$ for some odd $m < \omega$. It remains to specify E. E consists of all pairs $(q, (q, a, p_0, p_1))$ with $(q, a, p_0, p_1) \in \Delta$, all pairs $(q, \bot)$, $(\bot, \bot)$ and all pairs $((q, a, p_0, p_1), p_i)$ with $i \in \{0,1\}$.

Lemma 6.8. *For any parity tree automaton $\mathcal{A}$, player 0 has a winning strategy in $(\mathcal{G}_\mathcal{A}, q_0)$ if and only if $T(\mathcal{A}) \neq \emptyset$.*

Proof. (⇒) Let $\mathcal{A} = (Q, \Sigma, q_0, \Delta, \Omega)$, and assume that player 0 has a (without loss of generality) positional winning strategy σ in $(\mathcal{G}_\mathcal{A}, q_0)$. We define a tree $t \in \mathcal{T}_\Sigma^\omega$ and a tree $s \in \mathcal{T}_Q^\omega$ such that s is an accepting run of $\mathcal{A}$ on t. s and t are defined inductively by $s(\varepsilon) = q_0$ and $\sigma(s(w)) = (s(w), t(w), s(w0), s(w1))$ for all $w \in \{0,1\}^*$. Clearly, s is a run of $\mathcal{A}$ on t. Moreover, the run is accepting because a path π through s not satisfying the parity condition can be used to define a winning strategy of player 1 in $(\mathcal{G}_\mathcal{A}, q_0)$.

(⇐) Let $\mathcal{A} = (Q, \Sigma, q_0, \Delta, \Omega)$ and $t \in T(\mathcal{A})$. Thus we can fix an accepting run s of $\mathcal{A}$ on t. For any tree node $w \in \{0,1\}$, let $\tau(w) = (s(w), t(w), s(w0), s(w1)) \in \Delta$. We define a winning strategy σ of player 0 in $(\mathcal{G}_\mathcal{A}, q_0)$. To achieve this, for any $h \in (Q \cdot \Delta)^* \cdot Q$, we define a tree node $n(h) \in \{0,1\}^*$ by $n(p) = \varepsilon$ for $p \in Q$ and $n(h(q, a, p_0, p_1)p) = n(h)i$ where $i = 0$ if $p = p_0$ and $i = 1$ otherwise. σ is then defined by $\sigma(h) = \tau(n(h))$ for all $h \in (Q \cdot \Delta)^* \cdot Q$. This strategy is winning because the existence of a play of $(\mathcal{G}_\mathcal{A}, q_0)$ consistent with σ but lost by player 0 implies the existence of a path through s not fulfilling the parity condition. □

Lemma 6.8 allows us to reduce the problem of deciding whether $T(\mathcal{A}) \neq \emptyset$ to the problem of deciding whether player 0 has a winning strategy in a finite two-player zero-sum parity game, and we can use Jurdziński's algorithm to solve the latter problem. This leads to the following theorem.

Theorem 6.9. *For a parity tree automaton $\mathcal{A}$ with n states, m transitions and $d \geq 2$ different priorities, whether $T(\mathcal{A}) \neq \emptyset$ can be decided in time $O\big(d(m+n)\big(\frac{m+n+1}{\lfloor d/2 \rfloor}\big)^{\lfloor d/2 \rfloor}\big)$. The decision problem, given a parity tree automaton $\mathcal{A}$, decide whether $T(\mathcal{A}) \neq \emptyset$, is in* NP ∩ *co*-NP*.*

Proof. To decide whether $T(\mathcal{A}) \neq \emptyset$, we construct the parity game $\mathcal{G}_\mathcal{A}$. We observe that $\mathcal{G}_\mathcal{A}$ has $m + n + 1$ states, $3m + n + 1$ transitions and d different priorities (assuming without loss of generality that $\mathcal{A}$ has at least one odd priority). By Lemma 6.8, this describes a polynomial-time reduction from the problem of deciding non-emptiness for parity tree automata to the problem of deciding the winner in an initialised parity game. By Corollary 4.14, this problem is in NP ∩ co-NP. The time bound follows from Jurdziński's algorithm for two-player zero-sum parity games (Theorem 4.18). □

Finally, we show that every non-empty regular tree language contains a tree of a special kind, namely a tree that has only finitely many distinct subtrees, a result also due to Rabin [Rab72].

Definition 6.10. For a tree $t \in \mathcal{T}_\Sigma^\omega$, the *subtree rooted at* $w \in \{0,1\}^*$ is the tree $t|_w$ defined by $t|_w(v) = t(wv)$ for all $v \in \{0,1\}^*$. t is called *regular* if the set $\{t|_w : w \in \{0,1\}^*\}$ of all subtrees of t is finite.

Theorem 6.11 (Rabin)**.** *Every non-empty regular tree language contains a regular tree.*

Proof. Let $\mathcal{A} = (Q, \Sigma, q_0, \Delta, \Omega)$ be a parity tree automaton with $T(\mathcal{A}) \neq \emptyset$. Again we consider the game $\mathcal{G}_\mathcal{A}$. By Lemma 6.8, player 0 has a (without loss of generality) positional winning strategy σ in $(\mathcal{G}_\mathcal{A}, q_0)$. In the proof of the lemma, we constructed from σ a tree $t \in \mathcal{T}^\omega_\Sigma$ together with an accepting run s of $\mathcal{A}$ on t by defining $s(\varepsilon) = q_0$ and $\sigma(s(w)) = (s(w), t(w), s(w0), s(w1))$. Clearly s is regular because, if $s(w) = s(v)$, then $s|_w = s|_v$. Thus t is also regular because, if $s|_w = s|_v$, then $t|_w = t|_v$. □

6.2 Checking Strategies

We will now turn to the subgame perfect equilibrium threshold problem for multiplayer parity games. To use tree automata, we need to choose a suitable encoding of strategy profiles by binary trees.

In this section, we only consider games $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ where each vertex of the game graph has at most two successors, i.e. we require $|vE| \leq 2$ for all $v \in V$. Moreover, we assume that the vertices of the game are implicitly ordered by a linear ordering $<$ on V. Note that we can effectively transform any game $\mathcal{G}$ with n vertices into a game $\mathcal{G}'$ with $O(n^2)$ vertices where every vertex has at most two successors such that equilibria are preserved in both directions, i.e. $\mathcal{G}$ has a Nash (subgame perfect) equilibrium of payoff x if and only $\mathcal{G}'$ has a Nash (subgame perfect) equilibrium of payoff x.

Definition 6.12. Let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, (W_i)_{i\in\Pi})$ be a game, $v_0 \in V$ and $(\sigma_i)_{i\in\Pi}$ a strategy profile of $(\mathcal{G}, v_0)$. We define a tree $t : \{0,1\}^* \to V$ by $t(\varepsilon) = v_0$ and

$$t(wx) = \begin{cases} \min(t(w)E) & \text{if } x = 0, \\ \max(t(w)E) & \text{otherwise} \end{cases}$$

for all $w \in \{0,1\}^*$ and $x \in \{0,1\}$. Note that, by our condition on E, we have $t(w)E = \{t(w0), t(w1)\}$ for all $w \in \{0,1\}^*$. The tree $t_{(\sigma_i)_{i\in\Pi}} : \{0,1\}^* \to V \times \{0,1\}$ is defined by $t_{(\sigma_i)_{i\in\Pi}}(w) = \binom{t(w)}{x}$ for the least $x \in \{0,1\}$ with $t(wx) = \sigma_i(t(\varepsilon)\dots t(w))$ if $t(w) \in V_i$. Any $t_{(\sigma_i)_{i\in\Pi}}$ is called a *strategy tree of* $(\mathcal{G}, v_0)$.

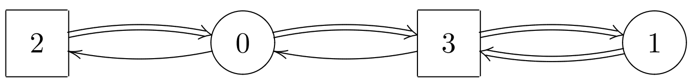

Figure 6.1: Game arena for Example 6.1

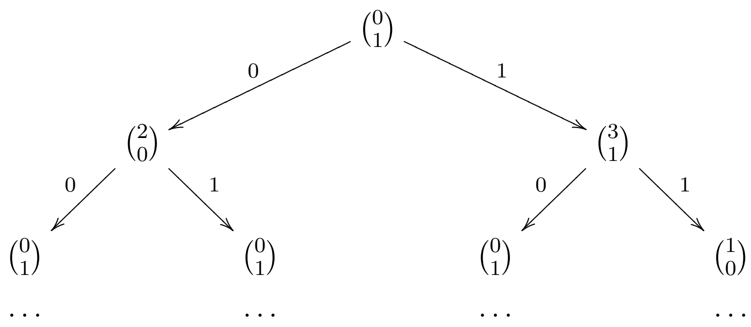

Figure 6.2: Strategy tree of the game from Example 6.1

Example 6.1. Consider again the game $\mathcal{G}$ from Example 4.3 with its arena depicted in Figure 6.1 together with the positional strategy profile (σ_1, σ_2) indicated by the double edges in the figure. The initial vertex is 0. The corresponding strategy tree $t_{(\sigma_1,\sigma_2)}$ of $(\mathcal{G}, 0)$ is depicted in Figure 6.2. Here $<$ is the usual ordering on ω.

To solve the subgame perfect equilibrium threshold problem for multiplayer parity games, we construct two tree automata: one checking whether a tree is a strategy tree of a given initialised finite multiplayer parity game $(\mathcal{G}, v_0)$ and one checking whether the strategy profile encoded by a strategy tree of $(\mathcal{G}, v_0)$ is *not* a subgame perfect equilibrium of $(\mathcal{G}, v_0)$ with payoff at least x and at most y for given thresholds x and y.

Lemma 6.13. *For any initialised multiplayer parity game $(\mathcal{G}, v_0)$ with vertices in a finite set V, there exists a Büchi tree automaton $\mathcal{A}_{\mathcal{G},v_0}$ with $O(|V|)$ states such that a tree $t \in \mathcal{T}^{\omega}_{V\times\{0,1\}}$ is accepted by $\mathcal{A}_{\mathcal{G},v_0}$ if and only if t is a strategy tree of $(\mathcal{G}, v_0)$.*

Proof. Let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\Omega_i)_{i\in\Pi})$. The automaton has to check whether the first component of $t(\varepsilon)$ is v_0, whether for each tree node w the vertices given by the first component of $t(w0)$ and $t(w1)$, respectively, are indeed *the* successors of the vertex given by the first component of $t(w)$ and whether for each tree node w the second component of $t(w)$ is 0 if the vertex given by the first component of $t(w)$ has only one successor. We define $\mathcal{A}_{\mathcal{G},v_0} = (Q, V \times \{0,1\}, q_0, \Delta, \Omega)$ as follows. The automaton has state set

$$Q = V \times \{1,2\},$$

and the initial state is $q_0 = (v_0, |v_0E|)$. The priority $\Omega(q)$ of each state $q \in Q$ is 0. Thus every run of $\mathcal{A}_{\mathcal{G},v_0}$ is accepting. The transition relation Δ consists of all tuples

$$((v,2), \binom{v}{*}, (w_0, |w_0E|), (w_1, |w_1E|))$$

with $vE = \{w_0, w_1\}$, $w_0 < w_1$, $* \in \{0,1\}$ and all tuples

$$((v,1), \binom{v}{0}, (w, |wE|), (w, |wE|))$$

with $vE = \{w\}$. It is easy to see that $t \in T(\mathcal{A}_{(\mathcal{G},v_0)})$ if and only if t is a strategy tree of $(\mathcal{G}, v_0)$. □

Lemma 6.14. *For any finite multiplayer parity game $\mathcal{G}$ with k players and at most d different priorities for each player and any two thresholds $x, y \in \{0,1\}^k$ with $x \leq y$, there exists a parity tree automaton $\mathcal{A}_{\mathcal{G},x,y}$ with $O(kd)$ states and $O(d)$ different priorities such that a strategy tree $t_{(\sigma_i)_{i\in\Pi}}$ of $(\mathcal{G}, v_0)$ is accepted by $\mathcal{A}_{\mathcal{G},x,y}$ if and only if $(\sigma_i)_{i\in\Pi}$ is not a subgame perfect equilibrium of $(\mathcal{G}, v_0)$ with payoff at least x and at most y.*

Proof. Let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\Omega_i)_{i\in\Pi})$ where without loss of generality $\Pi = \{1, \ldots, k\}$ and $\Omega_i(\chi(V)) \subseteq \{0, \ldots, 2d-1\}$ for all $i \in \Pi$. There may be two reasons why a strategy profile $(\sigma_i)_{i\in\Pi}$ of $(\mathcal{G}, v_0)$ is not a subgame perfect equilibrium with payoff between x and y: $\langle(\sigma_i)_{i\in\Pi}\rangle$ may not have a payoff between x and y or there may exist a history hv of $(\mathcal{G}, v_0)$ and a player $i \in \Pi$ such that $\langle(\sigma_j|_h)_{j\in\Pi}\rangle_v$ is not won by player i, but $\langle(\sigma_j|_h)_{j\in\Pi\setminus\{i\}}, \sigma'|_h\rangle_v$ is won by player i for some strategy σ' of player i in $\mathcal{G}$. The automaton $\mathcal{A} = \mathcal{A}_{\mathcal{G},x,y} = (Q, V \times \{0,1\}, q_0, \Delta, \Omega)$ will just check this. $\mathcal{A}$ has state set

$$Q = \{q_0, q_?, q_+\} \cup (\Pi \times \{0, \ldots, 2d-1\} \times \{\text{lose}, \text{win}, \text{guess}\}).$$

Intuitively, if the automaton is in state (i, m, lose), it checks whether player i loses the outcome of the given strategy profile; if it is in state (i, m, win), it checks whether player i wins the outcome of the given strategy profile; if it is in state (i, m, guess), it guesses a strategy of player i that leads to a play won by player i; if it is in state $q_?$, it guesses a subgame; and if it is in state q_+, it will accept the rest of the input tree.

In the initial state, the automaton guesses whether it checks the payoff of the outcome or whether it guesses a subgame for which the given strategy profile is not a Nash equilibrium. Thus Δ contains all transitions of the form

$$(q_0, \binom{v}{0}, (i, \Omega_i(\chi(v)), \text{lose}), q_+), \ (q_0, \binom{v}{1}, q_+, (i, \Omega_i(\chi(v)), \text{lose}))$$

for all i with $x_i = 1$, all transitions of the form

$$(q_0, \binom{v}{0}, (i, \Omega_i(\chi(v)), \text{win}), q_+), \ (q_0, \binom{v}{1}, q_+, (i, \Omega_i(\chi(v)), \text{win}))$$

for all i with $y_i = 0$ and all transitions of the form

$$(q_0, \binom{v}{*}, q_+, q_?), \ (q_0, \binom{v}{*}, q_?, q_+).$$

In a state (i, m, lose) or (i, m, win), the automaton just follows the move given by the input tree. Thus we add all transitions of the form

$$((i, m, \text{lose}), \binom{v}{0}, (i, \Omega_i(\chi(v)), \text{lose}), q_+),$$

$$((i, m, \text{lose}), \binom{v}{1}, q_+, (i, \Omega_i(\chi(v)), \text{lose})),$$

$$((i, m, \text{win}), \binom{v}{0}, (i, \Omega_i(\chi(v)), \text{win}), q_+),$$

$$((i, m, \text{win}), \binom{v}{1}, q_+, (i, \Omega_i(\chi(v)), \text{win})).$$

When guessing a subgame, the automaton non-deterministically guesses whether it should continue guessing the subgame or whether it should start to guess a strategy of the player who controls the current vertex. Thus we add all transitions of the form

$$(q_?, \binom{v}{*}, q_?, q_+), \ (q_?, \binom{v}{*}, q_+, q_?),$$

$$(q_?, \binom{v}{0}, (i, \Omega_i(\chi(v)), \text{lose}), (i, \Omega_i(\chi(v)), \text{guess})),$$

$$(q_?, \binom{v}{1}, (i, \Omega_i(\chi(v)), \text{guess}), (i, \Omega_i(\chi(v)), \text{lose}))$$

where $v \in V_i$. It may also be the case that the automaton needs to start guessing a strategy from the initial state. Thus we also add all transitions of the form

$$(q_0, \binom{v}{0}, (i, \Omega_i(\chi(v)), \text{lose}), (i, \Omega_i(\chi(v)), \text{guess})),$$

$$(q_0, \binom{v}{1}, (i, \Omega_i(\chi(v)), \text{guess}), (i, \Omega_i(\chi(v)), \text{lose}))$$

where $v \in V_i$. If the automaton guesses a strategy of player i, it may guess arbitrary moves at vertices controlled by player i, but it must follow the move given by the input tree at vertices not controlled by player i. Thus we add all transitions of the form

$$((i, m, \text{guess}), \binom{v}{*}, (i, \Omega_i(\chi(v)), \text{guess}), q_+),$$

$$((i, m, \text{guess}), \binom{v}{*}, q_+, (i, \Omega_i(\chi(v)), \text{guess}))$$

where $v \in V_i$ and

$$((i,m,\text{guess}), \binom{v}{0}, (i,\Omega_i(\chi(v)),\text{guess}), q_+),$$

$$((i,m,\text{guess}), \binom{v}{1}, q_+, (i,\Omega_i(\chi(v)),\text{guess}))$$

where $v \notin V_i$. If the automaton is in the accepting state q_+, then it should remain there. Thus we also add all transitions of the form

$$(q_+, \binom{v}{*}, q_+, q_+).$$

This completes the description of the transition relation Δ.

The priority $\Omega(q_+) = \Omega(q_0)$ of the states q_+ and q_0 is 0, and the priority $\Omega(q_?)$ of the state $q_?$ is 1. Finally, the priority $\Omega((i,m,\text{lose}))$ of a state (i,m,lose) is $m+1$, and the priority $\Omega((i,m,\text{win})) = \Omega((i,m,\text{guess}))$ of a state (i,m,win) or (i,m,guess) is m.

It remains to show that, for all strategy trees $t_{(\sigma_i)_{i\in\Pi}}$ of $(\mathcal{G}, v_0)$ (where $v_0 \in V$ is any initial vertex), we have $t_{(\sigma_i)_{i\in\Pi}} \in T(\mathcal{A})$ if and only if $(\sigma_i)_{i\in\Pi}$ is *not* a subgame perfect equilibrium of $(\mathcal{G}, v_0)$ with payoff $\geq x$ and $\leq y$.

($\Rightarrow$) Assume $t = t_{(\sigma_i)_{i\in\Pi}} \in T(\mathcal{A})$, i.e. there exists an accepting run s of $\mathcal{A}$ on t. We distinguish the following three cases:

(1) If $s(z) = (i,\Omega_i(\chi(v_0)),\text{lose})$ and $s(1-z) = q_+$ for some $z \in \{0,1\}$ and some $i \in \Pi$, then $x_i = 1$ and there exists a path π through s such that $s|_\pi(k) = (i,\Omega_i(\chi(\rho(k-1))),\text{lose})$ for all $0 < k < \omega$ where $\rho = \langle(\sigma_j)_{j\in\Pi}\rangle$. As s is accepting, the least priority occurring infinitely often in $s|_\pi$ is even. Thus the least priority of player i occurring infinitely often in ρ is odd, which implies that ρ is not won by player i. Hence the payoff of $(\sigma_i)_{i\in\Pi}$ is not $\geq x$.

(2) Analogously, if $s(z) = (i,\Omega_i(\chi(v_0)),\text{win})$ and $s(1-z) = q_+$ for some $z \in \{0,1\}$ and some $i \in \Pi$, then $y_i = 0$ and $\langle(\sigma_j)_{j\in\Pi}\rangle$ is won by player i. Hence the payoff of $(\sigma_i)_{i\in\Pi}$ is not $\leq y$.

(3) Otherwise let $w \in \{0,1\}^*$ be of maximal length with $s(w) = q_?$ or $s(w) = q_0$. As $s(\varepsilon) = q_0$ and any path through s labelled only with $q_?$ from some point onwards does not fulfil the parity condition, such a w must exist. Let $hv \in V^*V$ be the corresponding history of $(\mathcal{G}, v_0)$, i.e. the elementwise projection of the word $t(\varepsilon)\ldots t(w)$ on the first component. Then there must exist two paths π and π' through s with prefix w such that $s|_\pi(k) = (i,\Omega_i(\chi(\rho(k-1))),\text{lose})$ and $s|_{\pi'}(k) = (i,\Omega_i(\chi(\rho'(k-1))),\text{guess})$ for all $|w| < k < \omega$ where $\rho = h\langle(\sigma_j|_h)_{j\in\Pi}\rangle_v$ and $\rho' = h\langle(\sigma_j|_h)_{j\in\Pi\setminus\{i\}}, \sigma'|_h\rangle_v$ for some strategy σ' of player i in $\mathcal{G}$. As s is accepting, the least priorities occurring infinitely often in $s|_\pi$ and $s|_{\pi'}$, respectively, are even. Thus the play $h\rho'$ is won by player i, but $h\rho$ is not. Hence $(\sigma_i|_h)_{i\in\Pi}$ is not a Nash equilibrium of $(\mathcal{G}|_h, v)$.

In all cases, we have that $(\sigma_i)_{i\in\Pi}$ is not a subgame perfect equilibrium of $(\mathcal{G}, v_0)$ with payoff $\geq x$ and $\leq y$.

($\Leftarrow$) Assume that $(\sigma_i)_{i\in\Pi}$ is not a subgame perfect equilibrium of $(\mathcal{G}, v_0)$ with payoff at least x and at most y, and let $t = t_{(\sigma_i)_{i\in\Pi}}$. We distinguish the following three cases.

(1) If there exists a player $i \in \Pi$ such that $x_i = 1$ but $\rho = \langle(\sigma_j)_{j\in\Pi}\rangle$ is not won by player i, then we can define an accepting run s of $\mathcal{A}$ on t as follows. Let $s(z) = (i, \Omega_i(\chi(v_0)), \text{lose})$ and $s(1-z) = q_+$ if $t(\varepsilon) = \binom{v_0}{z}$. By the definition of Δ, there is precisely one run s of $\mathcal{A}$ on t with this property, and there is precisely one path π through s that is not labelled with q_+ from some point onwards, where $s|_\pi(k) = (i, \Omega_i(\chi(\rho(k-1))), \text{lose})$ for all $0 < k < \omega$. But this path also fulfils the parity condition because the least priority of player i occurring infinitely often in ρ is odd. Hence s is accepting.

(2) If there exists a player $i \in \Pi$ such that $y_i = 0$ but $\rho = \langle(\sigma_j)_{j\in\Pi}\rangle$ is won by player i, then we can define an accepting run s of $\mathcal{A}$ on t as follows. Let $s(z) = (i, \Omega_i(\chi(v_0)), \text{win})$ and $s(1-z) = q_+$ if $t(\varepsilon) = \binom{v_0}{z}$. By the definition of Δ, there is precisely one run s of $\mathcal{A}$ on t with this property, and there is precisely one path π through s that is not labelled with q_+ from some point onwards, where $s|_\pi(k) = (i, \Omega_i(\chi(\rho(k-1))), \text{win})$ for all $0 < k < \omega$. But this also path fulfils the parity condition because the least priority of player i occurring infinitely often in ρ is even. Hence s is accepting.

(3) If the requirements of (1) and (2) are not met, then $(\sigma_i)_{i\in\Pi}$ is not a subgame perfect equilibrium of $(\mathcal{G}, v_0)$ at all. Thus there exists a history hv of $(\mathcal{G}, v_0)$ and a strategy σ' of some player i in $\mathcal{G}$ such that $\rho = \langle(\sigma_j|_h)_{j\in\Pi}\rangle_v$ is not won by player i, but $\rho' = \langle(\sigma_j|_h)_{j\in\Pi\setminus\{i\}}, \sigma'|_h\rangle_v$ is. Without loss of generality, let $\sigma'(hv) \neq \sigma_i(hv)$. We define a path π' through t by $\pi'(k) = 0$ if and only if $t(\pi'[0,k]0) = \binom{(h\rho')(k+1)}{0}$ or $t(\pi'[0,k]0) = \binom{(h\rho')(k+1)}{1}$. Note that this implies that for all $k < \omega$ we have $t|_{\pi'}(k) = \binom{(h\rho')(k)}{0}$ or $t|_{\pi'}(k) = \binom{(h\rho')(k)}{1}$. Now we can define a run s of $\mathcal{A}$ on t by $s(\varepsilon) = q_0$, $s|_{\pi'}(k) = q_?$ for all $0 < k \leq |h|$ and $s|_{\pi'}(k) = (i, \Omega_i(\chi(h\rho'(k-1))), \text{guess})$ for all $|h| < k < \omega$. By the definition of Δ, there is precisely one run s of $\mathcal{A}$ on t with this property, and there is precisely one path $\pi \neq \pi'$ through s that is not labelled with q_+ from some point onwards, where $s|_\pi(k) = (i, \Omega_i(\chi(\rho(k-1))), \text{lose})$ for all $|h| < k < \omega$. Both paths π and π' fulfil the parity condition because the least priority of player i occurring infinitely often in ρ is odd, whereas the least priority of player i occurring infinitely often in ρ' is even.

In all cases, there exists an accepting run s of $\mathcal{A}$ on t. Hence $t \in T(\mathcal{A})$. □

6.3 Putting It All Together

As regular tree languages are closed under complementation and intersection, we have that the set of all subgame perfect equilibria of a given finite multiplayer parity game with a payoff between any two given thresholds is regular. This allows us to infer the existence of a finite-state subgame perfect equilibrium for all payoffs realised by a subgame perfect equilibrium.

Theorem 6.15. *For any initialised finite multiplayer parity game* $(\mathcal{G}, v_0)$ *and any two payoff thresholds* $x, y \in \{0,1\}^\Pi$*, the set of all trees* $t \in \mathcal{T}^\omega_{V\times\{0,1\}}$ *such that* $t = t_{(\sigma_i)_{i\in\Pi}}$ *for a subgame perfect equilibrium* $(\sigma_i)_{i\in\Pi}$ *of* $(\mathcal{G}, v_0)$ *with payoff at least* x *and at most* y *is regular.*

Proof. By Lemma 6.13 and Lemma 6.14, for any $t \in \mathcal{T}^\omega_{V\times\{0,1\}}$, we have $t = t_{(\sigma_i)_{i\in\Pi}}$ for a subgame perfect equilibrium $(\sigma_i)_{i\in\Pi}$ of $(\mathcal{G}, v_0)$ with payoff at least x and at most y if and only if $t \in T(\mathcal{A}_{\mathcal{G},v_0}) \cap (\mathcal{T}^\omega_{V\times\{0,1\}} \setminus T(\mathcal{A}_{\mathcal{G},x,y}))$. Thus the claim follows from the closure of regular tree languages under complementation and intersection (the latter following from the closure under complementation and union). □

We say that $x \in \{0,1\}^k$ is a subgame perfect payoff of a game $(\mathcal{G}, v_0)$ if $(\mathcal{G}, v_0)$ has a subgame perfect equilibrium with payoff x.

Corollary 6.16. *For any initialised finite multiplayer parity game* $(\mathcal{G}, v_0)$ *and any subgame perfect payoff* x *of* $(\mathcal{G}, v_0)$*,* $(\mathcal{G}, v_0)$ *has a finite-state subgame perfect equilibrium with payoff* x*.*

Proof. Let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\Omega_i)_{i\in\Pi})$ and assume that $(\mathcal{G}, v_0)$ has a subgame perfect equilibrium with payoff x. By Theorem 6.15, the set of all strategy trees $t_{(\sigma_i)_{i\in\Pi}}$ where $(\sigma_i)_{i\in\Pi}$ is a subgame perfect equilibrium of $(\mathcal{G}, v_0)$ with payoff x is regular. Moreover, this set is non-empty because $(\mathcal{G}, v_0)$ has a subgame perfect equilibrium with payoff x. By Theorem 6.11, this implies that there exists a regular strategy tree $t = t_{(\sigma_i)_{i\in\Pi}}$ where $(\sigma_i)_{i\in\Pi}$ is a subgame perfect equilibrium of $(\mathcal{G}, v_0)$ with payoff x. Let R be a *subtree representative system* of t, i.e. a set $R \subseteq \{0,1\}^*$ such that $t|_r \neq t|_s$ for all $r \neq s \in R$ and for all $w \in \{0,1\}^*$ there exists $r \in R$ with $t|_w = t|_r$. For each $w \in \{0,1\}^*$ we denote the unique $r \in R$ with $t|_w = t|_r$ by $\overline{w}$. As t is regular, R is finite. For each player $i \in \Pi$, we define a strategy automaton $\mathcal{A}_i = (R, \overline{\varepsilon}, \delta, \tau)$ of player i in $\mathcal{G}$ by

$$\delta(r, v) = \begin{cases} \overline{r0} & \text{if } v = \min(uE), \\ \overline{r1} & \text{otherwise} \end{cases}$$

where $u \in V$ is the unique vertex with $t(r) = \binom{u}{0}$ or $t(r) = \binom{u}{1}$. The output function is defined by

$$\tau(r, v) = \begin{cases} \min(vE) & \text{if } t(r0) = \binom{v}{0} \text{ or } t(r1) = \binom{v}{0}, \\ \max(vE) & \text{otherwise.} \end{cases}$$

It is straightforward to verify that $t = t_{(\sigma_{\mathcal{A}_i})_{i\in\Pi}}$. Hence $(\sigma_{\mathcal{A}_i})_{i\in\Pi}$ is a finite-state subgame perfect equilibrium of $(\mathcal{G}, v_0)$ with payoff x. □

As any multiplayer MSO-game reduces finitely to a multiplayer parity game, the corollary generalises to multiplayer MSO-games.

Corollary 6.17. *For any initialised finite multiplayer MSO-game $(\mathcal{G}, v_0)$ and any subgame perfect payoff x of $(\mathcal{G}, v_0)$, $(\mathcal{G}, v_0)$ has a finite-state subgame perfect equilibrium with payoff x.*

Proof. This is a direct consequence of Corollary 6.16 together with Claim 5.17 and Lemma 4.11. □

Using the algorithms we have for parity tree automata, we can now give a solution to our decision problem.

Theorem 6.18. *The subgame perfect equilibrium threshold problem for finite multiplayer parity games is in* Exptime. *For classes of finite multiplayer parity games with a bounded number of players and priorities, the problem is in* P.

Proof. Let $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\Omega_i)_{i\in\Pi})$ be a finite multiplayer parity game, let $k = |\Pi|$, and let d be the maximal number of different priorities for each player. We need to decide whether the tree language $T := T(\mathcal{A}_{\mathcal{G},v_0}) \cap (\mathcal{T}^{\omega}_{V\times\{0,1\}} \setminus T(\mathcal{A}_{\mathcal{G},x,y}))$ is non-empty. Towards this, we construct the two tree automata $\mathcal{A}_{\mathcal{G},v_0}$ and $\mathcal{A}_{\mathcal{G},x,y}$. We deploy Theorem 6.6 to construct from $\mathcal{A}_{\mathcal{G},x,y}$ a tree automaton $\bar{\mathcal{A}}_{\mathcal{G},x,y}$ with $2^{O(kd^2 \log kd)}$ states and $O(kd^2)$ priorities such that $T(\bar{\mathcal{A}}_{\mathcal{G},x,y}) = \mathcal{T}^{\omega}_{V\times\{0,1\}} \setminus T(\mathcal{A}_{\mathcal{G},x,y})$. Finally, we use a product construction to construct a parity tree automaton $\mathcal{A}$ with $T(\mathcal{A}) = T$. $\mathcal{A}$ has states (q^1, q^2) where q^1 is a state of $\mathcal{A}_{\mathcal{G},v_0}$ and q^2 is a state of $\bar{\mathcal{A}}_{\mathcal{G},x,y}$, and there is a transition $((q^1, q^2), a, (p^1_0, p^2_0), (p^1_1, p^2_1))$ if there is a transition (q^1, a, p^1_0, p^1_1) in $\mathcal{A}_{\mathcal{G},v_0}$ and a transition (q^2, a, p^2_0, p^2_1) in $\bar{\mathcal{A}}_{\mathcal{G},x,y}$. The priority of a state (q^1, q^2) is simply the priority of the state q^2 in $\bar{\mathcal{A}}_{\mathcal{G},x,y}$. It is easy to see that there is an accepting run of $\mathcal{A}$ on a tree $t \in \mathcal{T}^{\omega}_{V\times\{0,1\}}$ if and only if there is a run of $\mathcal{A}_{\mathcal{G},v_0}$ on t and an accepting run of $\bar{\mathcal{A}}_{\mathcal{G},x,y}$ on t. As every run of $\mathcal{A}_{\mathcal{G},v_0}$ is accepting, this implies that $T(\mathcal{A}) = T$. Note that $\mathcal{A}$ has $O(|V|) \cdot 2^{O(kd^2 \log kd)}$ states and $O(kd^2)$ priorities. By Theorem 6.9, we can check whether $T(\mathcal{A}) \neq \emptyset$ in time polynomial in the number of states of $\mathcal{A}$ and exponential in the number of priorities. Thus our algorithm works in exponential time in general and in polynomial time for bounded k and d. □

Now we deploy the previously presented reductions to show decidability of the subgame perfect equilibrium threshold problem for more games.

Corollary 6.19. *The subgame perfect equilibrium threshold problem for finite multiplayer reachability or safety games is in* Exptime. *For classes of these games with a bounded number of players, the problem is in* P.

Proof. By Claim 4.12, any multiplayer reachability or safety game reduces finitely to a multiplayer Büchi game with the same set of players and exponentially many vertices only in the number of players. Thus the claim follows from Theorem 6.18. □

Corollary 6.20. *The subgame perfect equilibrium threshold problem is decidable for finite multiplayer MSO-games. The problem is in* 2EXPTIME *for finite multiplayer LTL-games.*

Proof. By Claim 5.17, every finite multiplayer MSO-game reduces finitely to a multiplayer parity game. Thus decidability follows from Theorem 6.18. Using Corollary 5.16, we can reduce a finite multiplayer LTL-game $\mathcal{G} = (\Pi, V, (V_i)_{i\in\Pi}, E, \chi, (\varphi_i)_{i\in\Pi})$ to a finite multiplayer parity game $\mathcal{G}'$ with $|V| \cdot 2^{2^{O(|\Pi| \max|\varphi_i|)}}$ vertices and at most $2^{O(\max|\varphi_i|)}$ priorities for each player. As the algorithm sketched in the proof of Theorem 6.18 needs exponential time only in the number of players and priorities, the claim for multiplayer LTL-games follows. □

6.4 Complexity

Our algorithm exhibits upper bounds for the complexity of the subgame perfect equilibrium threshold problem. A general lower bound is P, as the problem of deciding the winner in a finite two-player zero-sum reachability game is already P-hard. As every two-player zero-sum reachability game is also a safety game, the result holds for safety games as well.

Theorem 6.21. *The decision problem, given an initialised finite two-player zero-sum reachability game, decide whether player 0 has a winning strategy, is* P*-complete.*

Proof. It is well-known that the problem is in P (see for example [GTW02, Chapter 2]). On the other hand, it is easy to see that there is a logspace reduction from the problem of deciding, given a monotone Boolean circuit C and inputs $x_1, \ldots, x_n$, whether $C(x_1, \ldots, x_n) = 1$, which is known to be P-complete. □

Corollary 6.22. *The subgame perfect equilibrium threshold problem for finite two-player zero-sum reachability games is* P*-complete.*

Proof. By Corollary 6.19, the problem is in P. By Theorem 4.7, every reachability game has a subgame perfect equilibrium. Together with the zero-sum condition, this implies that player 0 has a winning strategy in a two-player zero-sum reachability game $(\mathcal{G}, v_0)$ if and only if $(\mathcal{G}, v_0)$ has a subgame perfect equilibrium with payoff $(1, 0)$. This yields a logspace reduction from the problem of deciding whether player 0 has a winning strategy in a two-player zero-sum reachability game to the subgame perfect equilibrium threshold problem for two-player zero-sum reachability games. By Theorem 6.21, the former problem is P-complete. □

As reachability games easily reduce to Büchi games, we can conclude that the subgame perfect equilibrium threshold problem is also P-complete for two-player zero-sum Büchi games (or equivalently two-player zero-sum co-Büchi games).

Corollary 6.23. *The subgame perfect equilibrium threshold problem for finite two-player zero-sum Büchi games is* P*-complete.*

Proof. By Theorem 6.18, the problem is in P. On the other hand, the game reduction from Claim 4.12 gives a logspace reduction from the subgame perfect equilibrium threshold problem for finite two-player zero-sum reachability games for reachability games to the same problem for finite two-player zero-sum Büchi games. □

It follows that (for any $d \geq 2$) the subgame perfect equilibrium threshold problem for finite two-player zero-sum parity games with at most d different priorities is also P-complete. For two-player zero-sum parity games with an unbounded number of priorities, we do not know whether the problem is in P, but we do have an NP ∩ co-NP upper bound.

Corollary 6.24. *The subgame perfect equilibrium threshold problem for finite two-player zero-sum parity games is in* NP ∩ *co-*NP*.*

Proof. Let $(\mathcal{G}, v_0)$ be a finite two-player zero-sum parity game, and let $x, y \in \{0, 1\}$. The only non-trivial cases for the subgame perfect equilibrium threshold problem are $x = y = (1, 0)$ and $x = y = (0, 1)$. By Theorem 4.7 and the zero-sum condition, $(\mathcal{G}, v_0)$ has a subgame perfect equilibrium with payoff $(1 - i, i)$ if and only player i has a winning strategy in $(\mathcal{G}, v_0)$. By Corollary 4.14, this can be decided in NP ∩ co-NP. □

For finite two-player zero-sum LTL-games, we can use the following result due to Pnueli and Rosner [PR90] to show 2EXPTIME-completeness of the subgame perfect equilibrium threshold problem.

Theorem 6.25 (Pnueli, Rosner)**.** *The decision problem, given an initialised finite two-player zero-sum LTL-game, decide whether player 0 has a winning strategy, is* 2EXPTIME*-complete.*

Corollary 6.26. *The subgame perfect equilibrium threshold problem for finite two-player zero-sum LTL-games is* 2EXPTIME*-complete.*

Proof. The proof is analogous to the proof of Corollary 6.22, replacing P by 2EXPTIME, Corollary 6.19 by Corollary 6.20 and Theorem 6.21 by Theorem 6.25. □

Finally, we turn to MSO-games, where we show a non-elementary lower bound for the subgame perfect equilibrium threshold problem. The lower bound originates in the following theorem due to Meyer [Mey75].

Theorem 6.27 (Meyer)**.** *The decision problem, given an MSO-sentence φ over the signature $\{<\}$, decide whether $(\omega, <) \models \varphi$, is not elementary.*

Corollary 6.28. *The subgame perfect equilibrium threshold problem for finite one-player MSO-games is not elementary.*

Proof. Given an MSO-sentence φ over $\{<\}$, let $\mathcal{G}_\varphi$ be the one-player game with one vertex 0, one edge $(0, 0)$ and winning condition φ. Note that $\mathcal{G}_\varphi$ can be constructed from φ in linear time. Then $(\omega, <) \models \varphi$ if and only if the unique play of $(\mathcal{G}_\varphi, 0)$ is won by the sole player if and only if $(\mathcal{G}_\varphi, 0)$ has a subgame perfect equilibrium with payoff 1. Thus we have a linear-time reduction from the problem of deciding $(\omega, <) \models \varphi$ to the subgame perfect equilibrium threshold problem for one-player MSO-games. By Theorem 6.27, the former problem is not elementary. □

	Reachability/ Safety	Parity, No. priorities bounded	Parity
No. players unbounded	EXPTIME	EXPTIME	EXPTIME
No. players bounded	P-complete	P-complete	EXPTIME
Two-player zero-sum	P-complete	P-complete	NP ∩ co-NP

	LTL/MSO, Formula bounded	LTL	MSO
No. players unbounded	EXPTIME	2EXPTIME-complete	non-elementary
No. players bounded	P-complete	2EXPTIME-complete	non-elementary
Two-player zero-sum	P-complete	2EXPTIME-complete	non-elementary

Table 6.1: Complexity of the subgame perfect equilibrium threshold problem

Table 6.1 summarises the complexity of the subgame perfect equilibrium threshold problem for various classes of games. What remains open is the exact complexity of the problem for multiplayer parity games with a bounded number of players but with an unbounded number of priorities and the complexity of the problem for games with an unbounded number of players. As an indication that the latter problem is substantially harder than the corresponding problem for games with a bounded number of players, we show that the problem is at least NP-hard for reachability games. Moreover, this already holds for the subproblem where the upper payoff threshold is trivial and for the corresponding problem for Nash equilibria. The reduction is based on a similar reduction from [CJM04].

Proposition 6.29. *The decision problem, given an initialised multiplayer reachability game* $(\mathcal{G}, v_0)$ *and a payoff threshold* $x \in \{0,1\}^k$*, decide whether* $\mathcal{G}$ *has a Nash (subgame perfect) equilibrium with payoff at least* x*, is* NP*-hard.*

Proof. We give a polynomial-time reduction from SAT, the problem of deciding whether a given Boolean formula in conjunctive normal form is satisfiable, to our problem. As SAT is well known to be NP-complete, this proves the claim. Given a Boolean formula α with clauses $C_1, \dots, C_k$ over the propositional variables $X_1, \dots, X_m$, we construct a multiplayer reachability game $\mathcal{G}_\alpha$ as follows. The game is played by players $0, \dots, m$ where player 0 is the only player making non-trivial moves. The arena of $\mathcal{G}_\alpha$ has vertex set $\{0\} \cup \{X_j, \neg X_j, j : j = 1, \dots, m\}$ where there are edges from $j-1$ to X_j and $\neg X_j$, from X_j and $\neg X_j$ to j and from m back to m. The resulting graph is depicted in Figure 6.3. Every vertex is controlled by player 0. The winning condition of any player $i \neq 0$ is given by the clause C_i, i.e. player i wins if a literal X_j or $\neg X_j$ occurring in the clause C_i is reached. Player 0 wins every play of $\mathcal{G}_\alpha$. Clearly, $\mathcal{G}_\alpha$ can be constructed from α in polynomial time. We show that $(\mathcal{G}_\alpha, 0)$ has a Nash (subgame perfect) equilibrium with payoff (at least) $(1, \dots, 1)$ if and only if α is satisfiable.

($\Rightarrow$) Assume $(\mathcal{G}_\alpha, 0)$ has a subgame perfect equilibrium where every player wins, and let π be the equilibrium play. Then the interpretation $\mathfrak{I} : \{X_1, \dots, X_m\} \to \{0,1\}$ defined by $\mathfrak{I}(X_j) = 1$ if and only if X_j occurs in π satisfies α because for any clause C_i, as π is won by player i, there exists a literal X_j or $\neg X_j$ occurring in C_i that also occurs in π, and hence we have $\mathfrak{I}(X_j) = 1$ or $\mathfrak{I}(X_j) = 0$, respectively, which shows that the clause C_i is satisfied by $\mathfrak{I}$.

($\Leftarrow$) Assume that α is satisfiable, and let $\mathfrak{I}$ be a satisfying interpretation. Then the strategy of player 0 to move from vertex $j-1$ to vertex X_j if $\mathfrak{I}(X_j) = 1$ and to vertex $\neg X_j$ otherwise generates a play π in $(\mathcal{G}_\alpha, 0)$ won by every player because for each clause C_i there must be a literal X_j or $\neg X_j$ occurring in C_i with $\mathfrak{I}(X_j) = 1$ or $\mathfrak{I}(X_j) = 0$, respectively, which implies that this literal occurs in π. As any strategy profile with payoff $(1, \dots, 1)$ is also a Nash (subgame perfect) equilibrium, this shows that $(\mathcal{G}_\alpha, 0)$ has a Nash (subgame perfect) equilibrium with payoff $(1, \dots, 1)$. □

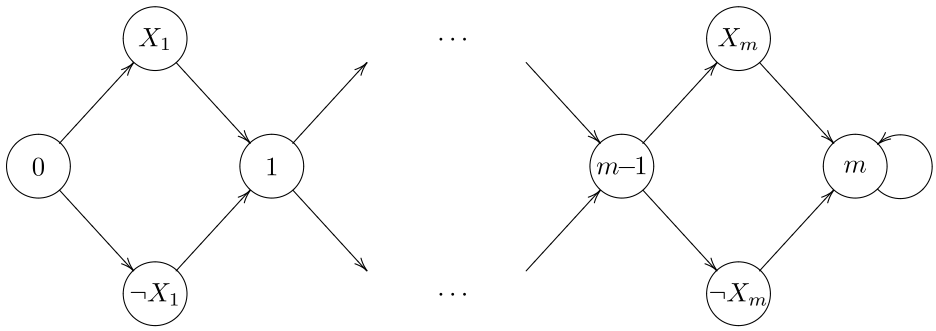

Figure 6.3: Arena of the game $\mathcal{G}_\alpha$.

Bibliography

[ALW89] Martín Abadi, Leslie Lamport and Pierre Wolper. Realizable and unrealizable specifications of reactive systems. In *Proceedings of the 16th International Colloquium on Automata, Languages and Programming, ICALP '89*, volume 372 of *Lecture Notes in Computer Science*, pages 1–17. Springer-Verlag, 1989. [25]

[BG04] Dietmar Berwanger and Erich Grädel. Fixed-point logics and solitaire games. *Theory of Computing Systems*, 37:675–694, 2004. [42]

[BL69] J. Richard Büchi and Lawrence H. Landweber. Solving sequential conditions by finite-state strategies. *Transactions of the American Mathematical Society*, 138:295–311, 1969. [25, 54]

[Büc62] J. Richard Büchi. On a decision method in restricted second order arithmetic. In *International Congress on Logic, Methodology and Philosophy of Science*, pages 1–11. Stanford University Press, 1962. [30, 51]

[CHJ04] Krishnendu Chatterjee, Thomas A. Henzinger and Marcin Jurdziński. Games with secure equilibria. In *Proceedings of the 19th Annual Symposium on Logic in Computer Science, LICS '04*, pages 160–169. IEEE Computer Society Press, 2004. [2]

[CJM04] Krishnendu Chatterjee, Marcin Jurdziński and Rupak Majumdar. On Nash equilibria in stochastic games. In *Proceedings of the 13th Annual Conference of the Europan Association for Computer Science Logic, CSL '04*, volume 3210 of *Lecture Notes in Computer Science*, pages 26–40. Springer-Verlag, 2004. [2, 20, 71]

[EFT84] Heinz-Dieter Ebbinghaus, Jörg Flum and Wolfgang Thomas. *Mathematical Logic*. Undergraduate texts in mathematics. Spinger-Verlag, 1984. [45]

[EJ91] E. Allen Emerson and Charanjit S. Jutla. Tree automata, mu-calculus and determinacy (extended abstract). In *Proceedings of* *the 32nd Annual Symposium on Foundations of Computer Science, FoCS '91*, pages 368–377. IEEE Computer Society Press, 1991. [30, 31, 34]

[ES84] E. Allen Emerson and A. Prasad Sistla. Deciding full branching time logic. *Information and Control*, 61(3):175–201, 1984. [53]

[GH82] Yuri Gurevich and Leo Harrington. Trees, automata and games. In *Proceedings of the 14th Annual ACM Symposium on Theory of Computing, STOC '82*, pages 60–65. ACM Press, 1982. [56, 57]

[Grä04] Erich Grädel. Positional determinacy of infinite games. In *Proceedings of the 21st Annual Symposium on Theoretical Aspects of Computer Science, STACS '04*, volume 2996 of *Lecture Notes in Computer Science*, pages 2–18. Springer-Verlag, 2004. [35]

[GS53] David Gale and Frank M. Stewart. Infinite games with perfect information. In *Contributions to the Theory of Games II*, volume 28 of *Annals of Mathematical Studies*, pages 245–266. Princeton University Press, 1953. [10, 14]

[GTW02] Erich Grädel, Wolfgang Thomas and Thomas Wilke, editors. *Automata, Logics, and Infinite Games*, volume 2500 of *Lecture Notes in Computer Science*. Springer-Verlag, 2002. [37, 52, 57, 68]

[Gur89] Yuri Gurevich. Infinite games. *Bulletin of the European Association for Theoretical Computer Science*, 38:93–100, 1989. [19]

[HH99] Andras Hajnal and Peter Hamburger. *Set Theory*. Cambridge University Press, 1999. [5]

[HU79] John E. Hopcroft and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, 1979. [5]

[Jur98] Marcin Jurdziński. Deciding the winner in parity games is in UP $\cap$ co-UP. *Information Processing Letters*, 68(3):119–124, 1998. [34]

[Jur00] Marcin Jurdziński. Small progress measures for solving parity games. In *Proceedings of the 17th Annual Symposium on Theoretical Aspects of Computer Science, STACS 2000*, volume 1770 of *Lecture Notes in Computer Science*, pages 290–301. Springer-Verlag, 2000. [37]

[Kam68] Johan Anthony Willem Kamp. *Tense Logic and the Theory of Linear Order*. PhD thesis, University of California, Los Angeles, 1968. [50]

[Kuh50] Harold W. Kuhn. Extensive games. *Proceedings of the National Academy of Sciences of the United States of America*, 36:570–576, 1950. [8]

[Löd99] Christof Löding. Optimal bounds for the transformation of omega-automata. In *Proceedings of the 19th Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS '99*, volume 1738 of *Lecture Notes in Computer Science*, pages 97–109. Springer-Verlag, 1999. [52]

[Mar75] Donald A. Martin. Borel determinacy. *Annals of Mathematics*, 102:363–371, 1975. [19]

[Mar85] Donald A. Martin. Purely inductive proof of Borel determinacy. In *Proceedings of Symposia in Pure Mathematics*, volume 42, pages 303–308, 1985. [19]

[McN65] Robert McNaughton. Finite-state infinite games. Technical report, Project MAC, Massachusetts Institute of Technology, USA, 1965. [25]

[Mey75] A. R. Meyer. Weak monadic second order theory of successor is not elementary-recursive. In *Proceedings of the Boston University Logic Colloquium*, pages 132–154. Springer-Verlag, 1975. [70]

[Mic88] Max Michel. Complementation is more difficult with automata on infinite words. Manuscript, CNET, Paris, 1988. [52]

[Mos80] Yiannis N. Moschovakis. *Descriptive Set Theory*. North-Holland, 1980. [18]

[Mos84] Andrzej Wlodzimierz Mostowski. Regular expressions for infinite trees and a standard form of automata. In *Computation Theory*, volume 208 of *Lecture Notes in Computer Science*, pages 157–168. Springer-Verlag, 1984. [30]

[Mos91] Andrzej Wlodzimierz Mostowski. Games with forbidden positions. Technical Report 78, Instytut Matematyki, Uniwersytet Gdański, Poland, 1991. [34]

[Nas50] John F. Nash Jr. Equilibrium points in N-person games. *Proceedings of the National Academy of Sciences of the United States of America*, 36:48–49, 1950. [9]

[OR94] Martin J. Osborne and Ariel Rubinstein. *A Course in Game Theory*. MIT Press, 1994. [8, 20]

[Pnu77] Amir Pnueli. The temporal logic of programs. In *Proceedings of the 18th Annual Symposium on Foundations of Computer Science, FOCS '77*, pages 46–57. IEEE Computer Society Press, 1977. [48]

[PP04] Dominique Perrin and Jean-Eric Pin. *Infinite Words (Automata, Semigroups, Logic and Games)*. Elsevier, 2004. [5]

[PR89] Amir Pnueli and Roni Rosner. On the synthesis of a reactive module. In *Proceedings of the Sixteenth Annual ACM Symposium on Principles of Programming Languages, POPL '89*, pages 179–190. ACM Press, 1989. [25]

[PR90] Amir Pnueli and Roni Rosner. Distributed reactive systems are hard to synthesize. In *Proceedings of the 31st Annual Symposium on Foundations of Computer Science, FoCS '90*, pages 746–757. IEEE Computer Society Press, 1990. [25, 69]

[Rab69] Michael O. Rabin. Decidability of second-order theories and automata on infinite trees. *Transactions of the American Mathematical Society*, 141:1–35, 1969. [57]

[Rab72] Michael O. Rabin. Automata on infinite objects and Church's problem. *American Mathematical Society*, 1972. [52, 59]

[Saf88] Shmuel Safra. On the complexity of omega-automata. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science, FoCS '88*, pages 319–327. IEEE Computer Society Press, 1988. [52]

[Saf92] Shmuel Safra. Exponential determinization for omega-automata with strong-fairness acceptance condition (extended abstract). In *Proceedings of the 24th Annual ACM Symposium on the Theory of Computing, STOC '92*, pages 275–282. ACM Press, 1992. [52]

[Sel65] Reinhard Selten. Spieltheoretische Behandlung eines Oligopolmodells mit Nachfrageträgheit. *Zeitschrift für die gesamte Staatswissenschaft*, 121:301–324 and 667–689, 1965. [12]

[Tho95] Wolfgang Thomas. On the synthesis of strategies in infinite games. In *Proceedings of the 12th Annual Symposium on Theoretical Aspects of Computer Science, STACS '95*, volume 900 of *Lecture Notes in Computer Science*, pages 1–13. Springer-Verlag, 1995. [31]

[Tho97] Wolfgang Thomas. Languages, automata, and logic. In *Handbook of Formal Language Theory*, volume III, pages 389–455. Springer-Verlag, 1997. [57]

[vNM44] John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. John Wiley and Sons, 1944. [1, 8]

[VWS83] Moshe Y. Vardi, Pierre Wolper and A. Prasad Sistla. Reasoning about infinite computation paths. In *Proceedings of the 24th Annual Symposium on Foundations of Computer Science, FOCS '83*, pages 185–194. IEEE Computer Society Press, 1983. [53]

[Zer13] Ernst Zermelo. Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels. In *Proceedings of the 5th International Congress of Mathematicians*, volume II, pages 501–504. Cambridge University Press, 1913. [19]

[Zie98] Wieslaw Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theoretical Computer Science*, 200(1–2):135–183, 1998. [30, 57]