Decisions

We agreed i) to have weights on actions and a note/remark about the extension to weights on states or edges; ii) to focus on NE for arbitrary strategies and, if we can prove something useful/interesting, ϵ-NE for finite-memory strategies; iii) we probably have enough material with mean-payoff and can consider max for a journal version; iv) it is probably enough to focus on the following decision problems: NE-emptiness, E-Nash, NE-membership.

Todos

1. Settle on structure of paper
2. Fill in the details of the proof sketches already in the draft. Note that bisimilar CGSs may not preserve existence of NE if histories are sequences of states. However, if histories alternate states and actions, then indeed they do. So, we must carefully check the papers we cite/use.
3. Decide if we want LTL or Lex formulas Φ for E-Nash. Question: how does one motivate E-Nash with LEX objectives for the government? Potential answer: social welfare. Technically, add an extra component to each tuple of actions in the CGS that belongs to the government.
4. Instead of reducing E-Nash to NE-emptiness it seems that one can simply adapt the proof that NE-emptiness is decidable for Lex Games to proving that E-Nash is decidable for Lex Games. Indeed, instead of looking for a secure path in G', look for a secure path in G' that satisfies the given formula Φ. Also, Giuseppe checked that E-Nash reduces to NE-emptiness using the usual matching-pennies gadget. We should decide whether to present NE-emptiness or E-nash "first".
5. Discuss "motivating examples"/"applications". Example scenario: i) one wholesaler and a bunch of retailers, each trying to maximise profit (this needs to be formalised); ii) taxation schemes. Perhaps look at motivating examples from papers on mean-payoff games for ideas.
6. If we need extra: If we need more dec probs, we might look at strong NE; other restrictions on strategy memory, myopic.
7. Absorb definition and techniques for ϵ-Nash equilibria.

Nash Equilibria in Games with Lexicographic Goals

XXX

1 Introduction

2 Preliminaries

In this section we define our game model: our games are played on finite graphs (rather than games in extensive-form or normal-form), games are multi-player, agents move concurrently (which includes the special case that they move sequentially), and agents play deterministic (rather than randomised) and memoryfull (rather than memoryless or finite-memory) strategies, and agents are trying to maximise their payoffs which are given as a lexicographic combination of a qualitative LTL formula and a quantitative long-term average of the rewards of its actions.

We fix some notation. If X is a set, then X^ω is the set of all infinite sequences over X. If X and Y are sets then X^Y is the set of all functions $f : Y \to X$.

2.1 Lexicographic Games

A *weighted arena* is a tuple $A = \langle Ag, Act, Ap, W, St, \iota, tr, \lambda, \kappa \rangle$ where

1. Ag is a finite non-empty set of *agents* (write $N = |Ag|$);
2. Act is a finite non-empty set of the *actions*;
3. Ap is a finite non-empty set of *atoms*;
4. $W \subset \mathbb{Z}$ is a finite non empty-set of integer *weights*;
5. St is a finite non-empty set of *states* and ι is the *initial state*;
6. $tr : St \times Act^{Ag} \to St$ is a *transition function* mapping each pair, consisting of a state and an action for each agent, to a successor state;
7. $\lambda : St \to 2^{Ap}$ is the *labeling function*;
8. $\kappa : Act \to W$ is a *weight function* mapping each action to an integer.

Executions. An *execution* $\pi = s_0 d_0 s_1 d_1 \cdots$ is an infinite sequence over $St \times Act^{Ag}$ such that $s_0 = \iota$ and $tr(s_i, d_i) = s_{i+1}$ for all i. In particular, $d_i(a)$ is the action of agent a in step i. Let Exec denote the set of all executions. Extend λ and κ to executions: i.e., each execution $\pi = s_0 d_0 s_1 d_1 \cdots$ induces

1. a sequence $\lambda(\pi) = \lambda(s_0)\lambda(s_1)\cdots$ of labels; and
2. for each agent a, a sequence $\kappa_a(\pi) = \kappa(d_0(a))\kappa(d_1(a))\cdots$.

Lexicographic games. For a sequence $\alpha \in \mathbb{R}^\omega$, let $\mathsf{mp}(\alpha)$ be the *mean-payoff* of α, i.e.,

$$\mathsf{mp}(\alpha) = \lim \inf_{n \to \infty} \frac{1}{n} \Sigma_{j=0}^{n-1} \alpha_j .$$

A $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$ *game* is a tuple $G = \langle A, \Psi \rangle$ where A is a weighted arena and $\Psi = (\varphi_a)_{a \in Ag}$ is a tuple of LTL formulas. The *payoff for agent a of an execution* π is the pair $pay_a(\pi) = (x, y) \in \{0,1\} \times \mathbb{R}$ where $x = 1$ iff $\lambda(\pi) \models \varphi$, and $y = \mathsf{mp}(\kappa_a(\pi))$. We define a total ordering on payoffs: $(x, y) \prec_{lex} (x', y')$ iff, either $x < x'$, or $x = x'$ and $y < y'$. Each agent is trying to maximise its payoff. In other words, agent a's primary goal is to make φ_a true, and its secondary goal is to maximise it's reward $\mathsf{mp}(\kappa_a(\cdot))$.

Remark 1. *We consider weights to be rewards to be maximised. One may, instead, consider them costs to be minimised. All our results hold for such cost-games. Indeed, given a weighted arena A with weights κ, form a weighted arena A' in which all weights are replaced by their negations. Then, an agent, say a, that maximises its payoff in A' has the primary goal of making φ_a true and the secondary goal of maximising* $\mathsf{mp}(-\kappa_a(\cdot))$. *But maximising* $\mathsf{mp}(-\kappa_a(\cdot)) = -\mathsf{mp}(\kappa_a(\cdot))$ *is the same as minimising* $\mathsf{mp}(\kappa_a(\cdot))$.

Strategies and Nash Equilibria. A *history* is a finite sequence $s_0 d_0 \cdots s_{n-1} d_{n-1} s_n$ such that $s_0 = \iota$ and $tr(s_i, d_i) = s_{i+1}$ for $i < n$. A *strategy profile* is a function $\sigma : Ag \to (\mathsf{Hst} \to Act)$. A strategy profile σ induces a unique execution π_σ, i.e., $\pi_\sigma = s_0 s_1 \cdots$ where $s_0 = \iota$ and $s_{i+1} = tr(s_i, d)$ where $d(a) = \sigma(a)(s_0 s_1 \cdots s_i)$ for $i > 0$. A strategy profile σ is a *Nash-Equilibrium* if for every strategy profile σ' that disagrees with σ on exactly one component, say $\sigma'(a) \neq \sigma(a)$ and $\sigma'(b) = \sigma(b)$ for all $b \neq a$, we have that $pay_a(\pi_{\sigma'}) \preceq_{lex} pay_a(\pi_\sigma)$.

Turn-based Arenas. Arenas in which every state is "controlled" by a single agent are called turn-based (sometimes called sequential). We define such arenas as a special case of concurrent arenas. Formally, a weighted weighted arena A is called *turn-based* if for every $s \in St$ there exists an agent $a \in Ag$ such that $tr(s, d) = tr(s, d')$ for all $d, d' \in Act^A g$ such that $d(b) = d'(b)$ for all $b \neq a$. A game is called *turn-based* if its arena is turn-based.

2.2 Decision Problems

The *NE-language* $\mathsf{NE}(G)$ of a game G is the set of strategy-profiles that are Nash-Equilibria. We consider the following decision problems:

1. *NE-membership* is the problem of deciding, given G and a finite-state profile σ, if $\sigma \in \mathsf{NE}(G)$.
2. *NE-emptiness* is the problem of deciding, given G, if $\mathsf{NE}(G)$ is empty.
3. *E-Nash* is the problem of deciding, given G and an LTL formula φ, if there exists $\sigma \in \mathsf{NE}(G)$ such that $\lambda(\sigma) \models \varphi$.

2.3 Justification of our Model

In this section we justify lexicographic games and our focus on pure memoryfull strategies.

First, $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$-games generalise a number of important models, i.e., LTL-games (let all weights be 0) and mp-games (let all formulas be true). Note that LTL-games generalise safety and reachability games. mention more papers, e.g., taxation model

Second, we focus on pure strategies instead of mixed strategies. This is because $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$-games need not have mixed NE (this is true even for two-player zero-sum games with reachability objectives [12]), and it is undecidable whether or not a given game has a mixed NE (this is true even for mean-payoff games [12]).

Third, we focus on decision problems for memoryfull strategies. This is because pure NE need not exist, e.g., matching pennies is a reachability game with no pure NE. Moreover, even if a pure NE exists, it need not have finite-memory NE. This can be seen by considering the following one-player game with LTL objective $\mathsf{G}\,\mathsf{F}\,p$: $Act = \{l, r\}$, $Ap = \{p\}$, $W = \{0, 10\}$, $St = \{\iota, v_1, v_2\}$, $tr(\iota, l) = v_1$, $tr(\iota, r) = v_2$, $tr(v_1, x) = tr(v_2, x) = \iota$ for $x \in Act$, $\lambda(v_1) = p$, $\kappa(\iota) = \kappa(v_1) = 10$ and $\kappa(v_2) = 0$. It is shown in [2, Example 1] that there is a pure strategy that achieves payoff $(1, 10)$, but every finite-state strategy achieves less.

Finally, we focus on decision problems for concurrent games rather than turn-based games because, as we now show, turn-based games always have NE.

Theorem 1. *Turn-based* $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$*-games always have NE.*

Sketch. The proof is in three steps: first, push the weights to the states and replace the LTL formulas by parity conditions on the states, to get a $\mathsf{Lex}(\mathsf{parity}, \mathsf{mp})$ game with the same set of NE. Second, prove that the following games are determined, i.e., they have a value: two-player zero-sum games with a $\mathsf{Lex}(\mathsf{parity}, \mathsf{mp})$ objective (i.e., one player is trying to maximise $pay(\pi)$ while the other is trying to minimise this same payoff). Third, use the existence of optimal strategies (from the second item) to prove the existence of a NE in multiplayer $\mathsf{Lex}(\mathsf{parity}, \mathsf{mp})$ games. This last step uses the fact that $\mathsf{Lex}(\mathsf{parity}, \mathsf{mp})$ is prefix-independent, i.e., if player a prefers execution π' to π then, for all histories h, player a also prefers $h \cdot \pi'$ to $h \cdot \pi$ (assuming these are also executions).

Here are the details. First, push the weights to the states. This is done by replacing St by $St \times Act^{Ag}$ and defining the weight of (s, d) to be $\kappa(d)$. Now translate each LTL formula φ_i into a deterministic parity word automaton D_i. Form the product weighted arena $A' = A \times \prod_i D_i$ where each state is labeled by both the weight from A and a tuple of priorities $\overline{p}$ from the D_is. The payoff for player i is the pair $(x, y) \in \{0, 1\} \times \mathbb{R}$ where $x = 1$ iff the smallest priority in co-ordinate i that occurs infinitely often is even, and y is the mean-payoff of the weights. The resulting $\mathsf{Lex}(\mathsf{parity}, \mathsf{mp})$ game A has the same NE as the original $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$ game.

Second, let A_i be the two-player game, based on the weighted arena A, in which the first player is trying to maximise player i's $\mathsf{Lex}(\mathsf{parity}, \mathsf{mp})$ objective, and the adversary is trying to minimise this same objective (thus each state is labeled by a single weight and a single priority). We are required to show that such games are determined. To this end, consider the zero-sum game played on weighted arena A in which player i is trying to achieve his parity objective *and* maximise his mean-payoff (and the opponent is adversarial). These are called mean-payoff parity games, and are known to be determined [6]. Let $v_i \in \mathbb{R} \cup \{-\infty\}$ be the value of this game. Let w_i be the value of the zero-sum game played on weighted arena A in which player i is trying to maximise his mean-payoff (i.e., he ignores his parity condition). The value of A_i is equal to v_i if $v_i \neq -\infty$, and equal to w_i if $v_i = -\infty$.

Third, the promised reduction works as follows: player i plays optimally in A_i (thus ensuring at least v_i), but the moment a player, say j, deviates from this strategy all the other players punish j by ensuring that player j receives no more than v_i. To define this profile formally, let σ_i be an optimal strategy for player i in A_i (note that it is also a strategy in A), let σ^{-i} be an optimal strategy for the opponent in A_i, and let $val_i(v)$ be the value of the game A_i starting in vertex v. These exist by determinacy. Let ρ_j^{-i} be the strategy for player j in A derived from σ^{-i}. Let π be the path induced by the strategy profile $(\sigma_1, \cdots, \sigma_N)$. Since σ_j is optimal, we get that $v_j \leq pay_j(\pi)$.

Define a strategy profile $\bar{\xi}$ where ξ_i follows σ_i as long as the play stays on π; the moment the play deviates, say in state v on player j's move, the strategy ξ_i switches to ρ_j^{-i} forever. To see that $\bar{\xi}$ is a NE reason as follows: let τ_j be any strategy for agent j different from ξ_j and consider the play π' induced by $(\xi_1, \cdots, \xi_{j-1}, \tau_j, \xi_{j+1}, \cdots, \xi_N)$ starting from initial state ι. Let v be the vertex of player j where τ_j first deviates from ξ_j. By construction of $\bar{\xi}$, we have that $pay_j(\pi'') \leq val_j(v)$ where π'' is the suffix of π' starting with the deviation. Since the objectives are prefix-independent, we get that $pay_j(\pi') = pay_j(\pi'')$. Since player j played optimally until v, we get that $val_j(v) = val_j(\iota)$. Thus, $pay_j(\pi') \leq pay_j(\pi)$, as required.

The only thing left to check is that $\mathsf{Lex}(\mathsf{parity}, \mathsf{mp})$ is prefix-independent. This follows from the definition of the lexicographic order and the fact that both the parity condition and the mean-payoff condition are prefix-independent, i.e., if the smallest priority occuring infinitely often in π (resp. π') is even (resp. odd), then for every h, the smallest priority occuring infinitely often in $h \cdot \pi$ (resp. $h \cdot \pi'$) is even (resp. odd); and if $\mathsf{mp}(\kappa(\pi)) < \mathsf{mp}(\kappa(\pi'))$ then, for every h, $\mathsf{mp}(\kappa(h \cdot \pi)) < \mathsf{mp}(\kappa(h \cdot \pi'))$. $\square$

3 Decision Procedures

Theorem 2. *NE-emptiness of* $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$*-games is decidable. Todos: 1) debug, 2) polish, 3) calculate complexity of* *the algorithm*

Sketch. We prove this in three steps. In the first step we push the weights of G into the states and translate the result into a $\mathsf{Lex}(\mathsf{parity}, \mathsf{mp})$ game G' such that $\mathsf{NE}(G) = \mathsf{NE}(G')$. Here parity is an aggregation function that maps a sequence of priorities (integer weights) to 0 if the smallest priority occuring infinitely often is even, and to 1 otherwise (the priorities are given as a second weight function, in addition to κ). In the second step we show how to reduce NE-emptiness of G' to the problem of solving two-player zero-sum games H in which player 0 has a $\mathsf{Lex}(\mathsf{parity}, \mathsf{mp})$ objective. We do this by adapting the proof that shows how to decide NE-emptiness of mean-payoff games [13]. In the third step we reduce H to solving mean-payoff parity games. These are two-player zero-sum games in which the objective of player 0 is to ensure both the parity condition holds and that the mean-payoff is maximised [6]. Formally: the payoff is $-\infty$ if the smallest priority occuring infinitely often is even, otherwise it is the mean-payoff of the weights.

Replace LTL by Parity. For the first step, push the weights to the states. This is done by replacing St by $St \times Act^{Ag}$ and defining the weight of (s, d) to be $\kappa(d)$. Convert each LTL formula φ_a into a deterministic parity automaton D_a of size double-exponential in φ_a [**?**]. To build the game G' form the product weighted arena $St' = St \times \prod_a D_a$. Each agent has a pair of weights associated with each state, i.e., a priority (coming from D_a) and the integer weight (coming from κ). Given a play, the payoff for an agent is (x, y) where $x = 0$ iff the smallest priority occuring infinitely often on the play is even, and y is the meanpayoff of the integer weights. By construction we have that $\mathsf{NE}(G) = \mathsf{NE}(G')$.

Reduce NE to path-finding. For the second step, we adapt the proof in Section 6 of [13] that shows how to decide NE-emptiness for mean-payoff games. For $a \in Ag$ and $s \in St$ define the *punishing value* $p_a(s)$ to be the $\prec_{lex}$-largest (x, y) that player a can achieve from state s by "going it alone", i.e., by playing against the coalition $Ag \setminus \{a\}$. We will show how to compute these values in the third step.

Definition: For an agent a and $\bar{z} \in \mathbb{R}^{|Ag|}$, a pair $(s, d) \in St \times Act^{Ag}$ is *$\bar{z}$-secure for a* if $p_a(tr(s, d')) \leq z_a$ for every $d' \in Act^{Ag}$ that agrees with d except possibly at a.

Claim: $\mathsf{NE}(G')$ is non-empty iff there exists $\bar{z}$ where $z_a \in \{p_a(s) : s \in St\}$ and there exists an execution $\pi = s_0 d_0 s_1 d_1 \cdots$ in G' such that for every agent a, i) $z_a \leq pay_a(\pi)$ and ii) for all $i \in \mathbb{N}$, the pair (s_i, d_i) is $\bar{z}$-secure for a.

Sketch proof of Claim: Suppose $\mathsf{NE}(G')$ is non-empty. Let π be the execution resulting from some Nash-profile. Let $z_a = \max\{p_a(\delta(s_n, d'_n)) : n \in \mathbb{N}, \wedge_{b \neq a} d'_n(b) = d_n(b)\}$, i.e., z_a is the largest value player a can get by deviating from π. Clearly (s_n, d_n) is $\bar{z}$-secure for a. Moreover, $z_a \leq pay_a(\pi)$: indeed, if $z_a = p_a(\delta(s_n, d'_n)) > pay_a(\pi)$ then player a would deviate at step n by playing d'_n and achieve a higher payoff, contradicting the choice of π as the execution of a Nash-profile.

For the converse direction, let $\bar{z}$ and π be given with the stated properties. The following is a Nash-profile: every agent plays "follow π" until, and if, a single player, say a, deviates, say at state s; in that case the other agents play a punishing strategy, i.e., a strategy that ensures player a achieves no more than he possibly can given the current history. It can be shown that this value is $p_a(s)$ for $LEX(\mathsf{parity}, \mathsf{mp})$ payoff functions (since such payoff functions are prefix-independent). This completes the proof of the claim.

Note that $p_a(s)$ is the value of the two-player zero-sum game $H_a(s)$ in which the first player is trying to maximise a's lexicographic payoff $pay_a(\cdot)$ and the second player is adversarial (i.e., trying to minimise $pay_a(\cdot)$). Thus, we have reduced NE-emptiness of $\mathsf{Lex}(\mathsf{parity}, \mathsf{mp})$ games to the problem of computing the values of these games $H_a(s)$.

Reduce to solving mean-payoff parity games. In the third and final step we show how to reduce the games $H = H_a(s)$ to solving mean-payoff parity games. We consider two games J and K, both on the same weighted arena as H, but with different objectives: the first player's objective in J is the mean-payoff parity objective, and the first player's objective in K is the mean-payoff objective. Let j be the value of J and k the value of K. It is easy to see that the value of H is $(1, j)$ if $j \neq -\infty$ and $(0, k)$ otherwise. □

Objectives	Complexity	Reference
$\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$	2EXPTIME-complete	This paper
LTL	2EXPTIME-complete	[10]
mean-payoff (mp)	NP-complete	[12]
reachability	NP-complete	[3]
lexicographic reachability	PSPACE-complete	[3]
Büchi	PTIME-complete	[3]
lexicographic Büchi	PTIME-hard, in NP	[3]

Table 1: Complexity of NE-emptiness for multiplayer concurrent-games with objectives from various classes.

Theorem 3. *E-NASH of* $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$*-games is decidable.*

Proof. Check that previous works with: $\mathsf{NE}(G') \subseteq \Phi$ iff there exists $\bar{z}$ where $z_a \in \{p_a(s) : s \in St\}$ and there exists an execution $s_0 d_0 s_1 d_1 \cdots$ in G' **satisfying** Φ such that each transition $tr(s_i, d_i) = s_{i+1}$ (for $i < \omega$) is $\bar{z}$-secure. □

4 Related Work

NE-emptiness is the main decision problem studied in this paper, i.e., decide if a given multiplayer concurrent game has a Nash Equilibrium or not. We studied this problem for $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$-games, i.e., graph-games in which agents are trying to maximise their lexicographic payoff whose first co-ordinate is an LTL formula and whose second co-ordinate is the mean-payoff of its weights. We proved the problem is 2EXPTIME-complete.

Other objectives. NE-emptiness has been studied for graph-games with other objectives, a sample of which are given in Table 4. $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$ generalises both mean-payoff games (simply set every formula to be true) and LTL games (simply set all the weights of the arena to be 0). That NE-emptiness is decidable in 2EXPTIME for LTL games follows from a more general result, i.e., that model checking *strategy logic* on concurrent game structures (i.e., arenas without the weights) is decidable [10]. To get the 2EXPTIME upper bound one writes the existence of a NE in a fragment of strategy logic (i.e., nested-goal with alternation number 1) whose model checking problem is decidable in 2EXPTIME. We cannot apply strategy logic to $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$ games since the latter have a quantitative component while strategy logic, being built on top of LTL, can only express qualitative objectives. Instead, as discussed in Section 3, we generalised the proof in [12] of the fact that NE-emptiness is in NP for mean-payoff games.

We remark that the hardness results for lexicographic reachability and lexicographic Büchi are for lexicographic orders on an arbitrary (but finite) number of co-ordinates.

Turn-based. Very simple concurrent games, such as matching pennies, do not have NE. However, there are many classes of turn-based graph-games that always have NE. For instance, turn-based mean-payoff games [1], turn-based games with ω-regular objectives (which include LTL objectives) [7], turn-based quantitative-reachability games [5]. Actually, this phenonemon has a general explanation: under certain conditions the existence of (finite-memory) NE for a class of objectives follows from the (finite-memory) determinacy of two-player zero-sum games with those objectives [11; 5]. Although the latter citations do not cover the case of turn-based $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$-games, we use a similar strategy in Theorem 1 to prove that these too always have NE. Determinacy (and the complexity of determining the winner) of two-player zero-sum games has been studied for a variety of objectives, i.e., mean-payoff [8; 14], ω-regular (which include LTL-objectives) [9].

Randomised Strategies. We have considered deterministic strategies. Randomised strategies are functions that map a history to a probability distribution over actions. NE-emptiness is undecidable for randomised strategies on games with reachability objectives for each player [4]. Since reachability objectives are a special case of $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$ objectives, we have undecidability of NE-emptiness for randomised strategies on $\mathsf{Lex}(\mathsf{LTL}, \mathsf{mp})$ games.

IMPORTANT RELATED WORK THAT NEEDS TO BE SUMMARISED: citeDBLP:journals/corr/abs-1303-0789; as well as work on secure-Equilibria

5 Conclusion

Future Work.

1. Decide if there is a unique NE (cite GSL paper).
2. The constrained NE-emptiness problem, i.e., does there exist a NE whose value (for each player) is in some given interval? This is studied for mean-payoff games in [12] where it is shown that constrained NE-emptiness is NP-complete for pure NE.

References

[1] S. Alpern. Cycles in extensive form perfect information games. *Journal of Mathematical Analysis and Applications*, 159(1):1 – 17, 1991.

[2] A. Bohy, V. Bruyère, E. Filiot, and J. Raskin. Synthesis from LTL specifications with mean-payoff objectives. *CoRR*, abs/1210.3539, 2012.

[3] P. Bouyer, R. Brenguier, N. Markey, and M. Ummels. Pure nash equilibria in concurrent deterministic games. *Logical Methods in Computer Science*, 11(2), 2015.

[4] P. Bouyer, N. Markey, and D. Stan. Mixed nash equilibria in concurrent terminal-reward games. In V. Raman and S. P. Suresh, editors, *34th International Conference on Foundation of Software Technology and Theoretical Computer Science, FSTTCS 2014, December 15-17, 2014, New Delhi, India*, volume 29 of *LIPIcs*, pages 351–363. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2014.

[5] T. Brihaye, J. De Pril, and S. Schewe. *Multiplayer Cost Games with Simple Nash Equilibria*, pages 59–73. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[6] K. Chatterjee, T. A. Henzinger, and M. Jurdzinski. Mean-payoff parity games. In *20th IEEE Symposium on Logic in Computer Science (LICS 2005), 26-29 June 2005, Chicago, IL, USA, Proceedings*, pages 178–187. IEEE Computer Society, 2005.

[7] K. Chatterjee, R. Majumdar, and M. Jurdzinski. On nash equilibria in stochastic games. In J. Marcinkowski and A. Tarlecki, editors, *Computer Science Logic, 18th International Workshop, CSL 2004, 13th Annual Conference of the EACSL, Karpacz, Poland, September 20-24, 2004, Proceedings*, volume 3210 of *Lecture Notes in Computer Science*, pages 26–40. Springer, 2004.

[8] A. Ehrenfeucht and J. Mycielski. Positional strategies for mean payoff games. *International Journal of Game Theory*, 8(2):109–113, 1979.

[9] E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logics, and Infinite Games: A Guide to Current Research [outcome of a Dagstuhl seminar, February 2001]*, volume 2500 of *Lecture Notes in Computer Science*. Springer, 2002.

[10] F. Mogavero, A. Murano, G. Perelli, and M. Y. Vardi. Reasoning about strategies: On the model-checking problem. *ACM Trans. Comput. Log.*, 15(4):34:1–34:47, 2014.

[11] S. L. Roux and A. Pauly. Extending finite memory determinacy: General techniques and an application to energy parity games. *CoRR*, abs/1602.08912, 2016.

[12] M. Ummels and D. Wojtczak. The complexity of nash equilibria in limit-average games. *CoRR*, abs/1109.6220, 2011.

[13] M. Ummels and D. Wojtczak. The complexity of nash equilibria in limit-average games. In J. Katoen and B. König, editors, *CONCUR 2011 - Concurrency Theory - 22nd International Conference, CONCUR 2011, Aachen, Germany, September 6-9, 2011. Proceedings*, volume 6901 of *Lecture Notes in Computer Science*, pages 482–496. Springer, 2011.

[14] U. Zwick and M. Paterson. The complexity of mean payoff games on graphs. *Theoretical Computer Science*, 158(1&2):343–359, 1996.