Concurrent Games with Arbitrarily Many Players

Nathalie Bertrand
University of Rennes, Inria, CNRS, IRISA, France

Abstract

Traditional concurrent games on graphs involve a fixed number of players, who take decisions simultaneously, determining the next state of the game. With Anirban Majumdar and Patricia Bouyer, we introduced a parameterized variant of concurrent games on graphs, where the parameter is precisely the number of players. Parameterized concurrent games are described by finite graphs, in which the transitions bear finite-word languages to describe the possible move combinations that lead from one vertex to another.

We report on results on two problems for such concurrent games with arbitrary many players. To start with, we studied the problem of determining whether the first player, say Eve, has a strategy to ensure a reachability objective against any strategy profile of her opponents as a coalition. In particular Eve's strategy should be independent of the number of opponents she actually has. We establish the precise complexities of the problem for reachability objectives.
Second, we considered a synthesis problem, where one aims at designing a strategy for each of the (arbitrarily many) players so as to achieve a common objective. For safety objectives, we show that this kind of distributed synthesis problem is decidable.

2012 ACM Subject Classification Theory of computation → Verification by model checking

Keywords and phrases concurrent games, parameterized verification

Digital Object Identifier 10.4230/LIPIcs.MFCS.2020.1

Category Invited Talk

Acknowledgements This paper is based on partly published results obtained in a collaboration with Patricia Bouyer and Anirban Majumdar.

1 Motivation

We introduce and study concurrent games in which the number of players is *a priori* unknown. Games with arbitrarily many players seem particularly relevant to model modern distributed systems. A first typical situation is the one of a global server, answering requests from an arbitrary number of clients. One can also think of a fleet of drones trying to cooperate to achieve a common goal. Wireless sensors networks and ant colonies are more application examples of games with arbitrarily many players.

2 Games with arbitrarily many players

Starting from concurrent games with a fixed number of players [1, 2], a natural idea is to define concurrent with arbitrarily many players by equipping edges of the arena with languages of finite words. For instance, an edge from vertex v to vertex v' can be labelled with a language L, representing the situation where, if there are k players, and in v, player i chooses action a_i, and $a_1 \cdots a_k \in L$, then the next vertex will be v'. As an example, L can be the regular language described by the regular expression $a(\Sigma^2)^* + (bb)^*$; with six players that all choose b, or with seven players if the first one chooses a while the choices of all others are arbitrary, will lead to v'.

45th International Symposium on Mathematical Foundations of Computer Science (MFCS 2020).
Editors: Javier Esparza and Daniel Král'; Article No. 1; pp. 1:1–1:8
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Note that the number of players k is unknown to them, but fixed all along the play. Choosing k, and resolving the nondeterminism is performed by the (adversarial) environment. Since the number of players is a parameter, we refer to the arenas as parameterized arenas, that we now formally define.

▶ **Definition 1** (Parameterized arena). *A* parameterized arena *is a tuple* $\mathcal{A} = (V, \Sigma, \Delta)$ *with*
- V *is a finite set of vertices;*
- Σ *is a finite alphabet of actions;*
- $\Delta : V \times V \to 2^{\Sigma^+}$ *is a partial transition function.*

▶ **Example 2.** The notion of parameterized arena is illustrated on an example, depicted in Figure 1. Here, the alphabet of actions is $\Sigma = \{a, b\}$, and for instance, the language that labels the edge from v_3 to v_4 is $ba + abab^+$. Thus, when in vertex v_3, if either there are two players and Player 1 plays b while Player 2 plays a, or if there are at least 4 players and their actions form a word in $abab^+$, then the game moves to v_4. Note that this arena is nondeterministic: for instance, from vertex v_2, any word in $a(aa)^*$ can lead to v_0, v_3 and v_5. Also, in this example, all languages are regular, and are thus denoted by regular expressions.

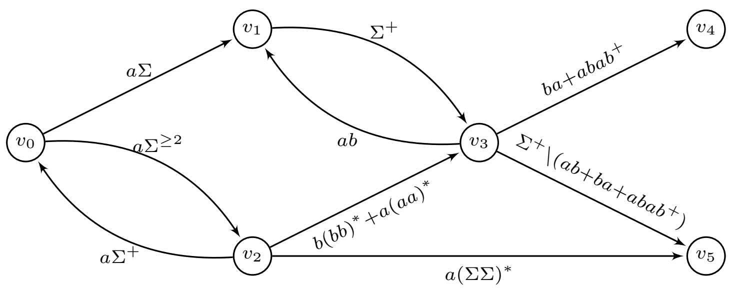

Figure 1 An example of a parameterized arena.

In the whole paper, we assume that arenas are *complete*: from any vertex v, for any non-empty word $w \in \Sigma^+$, there exists an edge $v \xrightarrow{L} v'$ with $w \in L$. For conciseness, the examples –as above– might depict incomplete arenas; a sink state can be added so as to obtain complete arenas.

▶ **Definition 3** (Strategies and induced plays). *A strategy for Player i in the arena* $\mathcal{A} = (V, \Sigma, \Delta)$ *is a function* $\sigma_i : V^+ \to \Sigma$.

An infinite strategy profile $\pi = (\sigma_1, \sigma_2, \cdots)$ *induces the plays:*

$$\begin{aligned}\mathsf{Plays}_{\mathcal{A}}(\pi) &= \bigcup_k \mathsf{Plays}_{\mathcal{A}}(\sigma_1, \sigma_2, \cdots, \sigma_k) \\ &= \bigcup_k \{v_0 v_1 v_2 \cdots \mid \forall j \geq 0,\ \sigma_1(v_0 \cdots v_j) \cdots \sigma_k(v_0 \cdots v_j) \in \Delta(v_j, v_{j+1})\}\end{aligned}$$

In words, a strategy dictates which action to play depending on the sequence of vertices seen so far. The plays induced by a strategy profile are formed of induced k-*plays* for each possible number of players k. An induced k-play satisfies that at each step j, the word, obtained by concatenating the actions prescribed by the strategies for players from 1 to k, belongs to the language labelling the edge from v_j to v_{j+1}. The initial choice of k, and the resolution of nondeterminism during the play are taken care of by an adversarial environment.

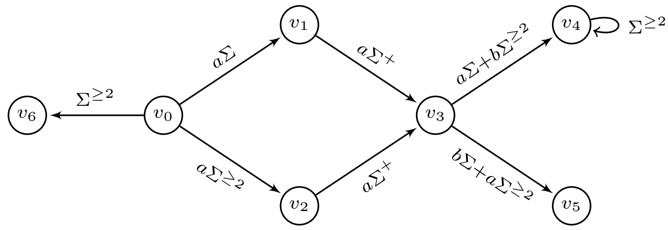

Figure 2 A simple concurrent parameterized arena.

▶ **Example 4.** Let us illustrate the notions of strategies and plays on Figure 2. Notice that in this arena, the actions of all players but Player 1 are irrelevant: indeed the languages are particularly simple $a\Sigma$, $a\Sigma^+$, etc.

Assuming the game starts in v_0, an example of strategy for Player 1 is the following:

- $\sigma_1(v_0) = \sigma_1(v_0v_1) = \sigma_1(v_0v_2) = \sigma_1(v_0v_1v_3) = a$
- $\sigma_1(v_0v_2v_3) = b$

Examples of plays consistent with σ_1 are $v_0v_1v_3v_4$ if $k = 2$ and $v_0v_2v_3v_4$ if $k = 3$. Indeed, if we annotate plays with actions of each players, these plays are for instance obtained by $v_0 \xrightarrow{aa} v_1 \xrightarrow{ab} v_3 \xrightarrow{aa} v_4$ and $v_0 \xrightarrow{aab} v_2 \xrightarrow{abb} v_3 \xrightarrow{baa} v_4$, respectively.

Even if the actions of players 2 to k are irrelevant, the arena is nondeterministic: in the very first step, under σ_1, the environment resolves the nondeterminism between going to v_6, or progressing to v_2 or v_3 (depending on the number of opponents). For a fixed number of players k, there are thus two plays induced by σ_1.

▶ **Definition 5** (Game). *A game* $\mathcal{G} = (\mathcal{A}, \mathsf{Win})$ *is an arena equipped with a set of infinite plays:* $\mathsf{Win} \subseteq V^\omega$.

Typical examples of winning conditions that we use in this paper are

- Reachability: For a target set $T \subseteq V$, $\mathsf{Win} = \{v_0v_1 \cdots | \exists i : \ v_i \in T\}$;
- Safety: For a safe set $S \subseteq V$, $\mathsf{Win} = \{v_0v_1 \cdots | \forall i : \ v_i \in S\}$.

Outline

Parameterized arenas raise many interesting problems, and in this article we focus on two of them. In the first problem, Player 1 is distinguished, and she aims at achieving an objective independently of the number of opponents she has and of the strategies they play. In the second problem, all players try to achieve an objective as a coalition, not knowing however *a priori* how many they are. In the next two sections, we formalize these problems, and give decidability and complexity results.

3 One player against all

We first consider a setting in which player 1 aims at achieving an objective independently on the number of her adversaries, and whatever strategy they play. This situation can be motivated for instance by a scenario in which a server aims at answering requests by arbitrarily many clients. Formally

Eve vs rest of the world
Input: A parameterized arena $\mathcal{A}$ and a winning condition Win.
Question: $\exists \sigma_1 \ \forall k \forall \sigma_2 \cdots \sigma_k \ \mathsf{Plays}_{\mathcal{A}}(\sigma_1, \sigma_2, \cdots, \sigma_k) \subseteq \mathsf{Win}$?

▶ **Theorem 6.** *When* Win *is a reachability objective,* Eve vs rest of the world *is a* PSPACE-*complete problem.*

The PSPACE-hardness (see [3]) is obtained via a natural reduction from QBF [11], and we focus now on sketching the proof of PSPACE membership.

Recall that Eve must win against her opponents playing as a coalition and also against the environment that chooses the number of players before the play starts, and resolves potential nondeterminism. It seems thus quite natural to reduce to a 2-player game (see for instance [7, Chapter 12] for a gentle introduction to such games). As we have seen in Example 4, in order to win, Eve must gain information on the number of opponents she actually has. We thus build the *knowledge game*, a 2-player turn-based game tracking Eve's knowledge. The knowledge game starts in a state $(v_0, \mathbb{N})$ reflecting that the parameterized game is in v_0, and Eve has no information on the number of her opponents. The construction of the knowledge

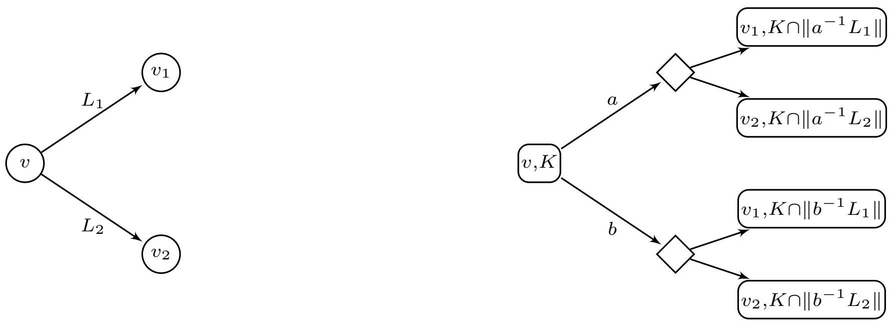

Figure 3 Construction of the knowledge game.

game is depicted in Figure 3: on the left-hand side is a subgame of the parameterized arena, and on the right-hand side part of the corresponding knowledge game. If $K \subseteq \mathbb{N}$ is the current knowledge, from a vertex (v, K) of the existential player (represented with rounded rectangles), she chooses an action in Σ, leading to a vertex of the universal player (represented with diamonds). The universal player resolves nondeterminism (if any), leading to a vertex (v_i, K_i), where K_i represents the updated knowledge. Here $\|L\|$ denotes the set of lengths of words in L: $\|L\| = \{|w| \mid w \in L\}$. Assuming that she played a, and the play moves to v_i, Eve updates the actual number of opponents to $K \cap \|a^{-1}L_i\|$. While building the knowledge game, only existential vertices (v_i, K_i) with non-empty K_i are constructed. Also, only universal vertices corresponding to a feasible action of Eve are built.

Figure 4 provides the knowledge game for the example from Figure 2.

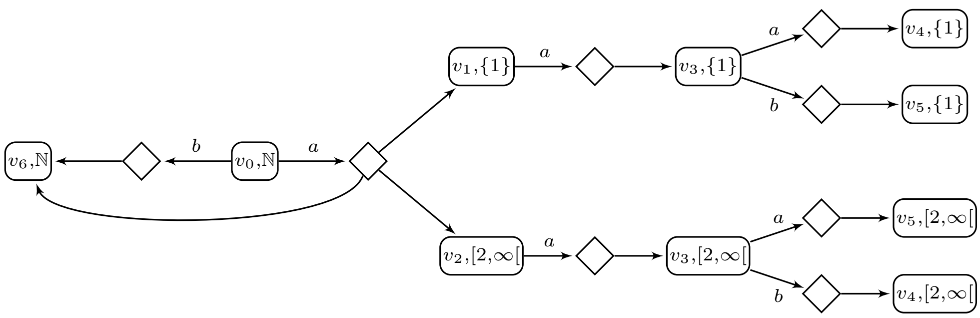

Figure 4 Knowledge game for the arena from Figure 2.

If we transfer the winning condition Win to the knowledge game, one can show that Eve has a winning strategy in the parameterized arena if and only if the existential player wins the knowledge game.

▶ **Example 7.** Back to the parameterized arena of Figure 2, consider the objective to reach the set $\{v_4, v_6\}$. On the corresponding knowledge game (see Figure 4) the objective of the existential player is thus to reach any vertex (v, K) with $v \in \{v_4, v_6\}$. An obvious winning strategy to do so is to play b in the first step. However, even if a is the first action, the existential player can ensure reaching the target set: the chosen action should be a after going through v_1, and b after going through v_2. Thus, the strategy σ_1 described in Example 4 ensures Eve to reach $\{v_4, v_6\}$ independently of the number of her opponents.

To solve Eve vs rest of the world, it thus suffices to solve the 2-player turn-based knowledge game. Starting with a parameterized arena $\mathcal{A}$ with regular languages on the edges, the corresponding knowledge game is at most exponential in $|\mathcal{A}|$. This is due to the fact that vertices in the knowledge game encode the knowledge Eve has, which is obtained by intersecting lengths of words in the initially given languages. These subsets of $\mathbb{N}$ are all semilinear [9], and there can be at most exponentially many, corresponding to the possible combinations of intersections. Constructing the knowledge game, and solving it for a reachability objective would thus yield an exponential time algorithm.

To obtain the announced PSPACE complexity upper bound, we show that storing the whole knowledge game is not necessary. In constrast, taking a dynamic programming approach, it is sufficient to store only subgames that are polynomial in the size of the parameterized arena. Each subgame is rooted at some existential vertex (v, K) and stops as soon as, either the target set T is reached (with arbitrary knowledge), or the knowledge changes to some $K' \subsetneq K$. In such a subgame, there are at most polynomially many vertices, and the objective of the existential player is to reach vertices (v', K') that are winning. Such winning vertices are computing recursively, and the recursion depth is polynomially bounded. The interested reader can find more details on this polynomial space procedure in [3].

4 Strategy synthesis for a coalition

After the one player against all setting, we now consider the case where agents want to collectively achieve a goal, independently of the number they actually are. Formally

Synthesis for arbitrarily-large coalition
Input: A parameterized arena $\mathcal{A}$ and a winning condition Win.
Question: $\exists \sigma_1, \sigma_2 \cdots \forall k$ $\mathsf{Plays}_{\mathcal{A}}(\sigma_1, \sigma_2, \cdots, \sigma_k) \subseteq$ Win?

▶ **Theorem 8.** *When* Win *is a safety objective,* Synthesis for arbitrarily-large coalition *is in* EXPSPACE *and* PSPACE*-hard.*

The PSPACE-hardness proof for Eve vs rest of the world can be adapted to obtain the same lower-bound here. In the sequel, we explain how we establish the EXPSPACE complexity upper bound. Closing the complexity gap is currently on our agenda.

▶ **Example 9.** Consider the nondeterministic parameterized arena from Figure 5, and assume the winning objective Win is to avoid the sink vertices v_2 and v_3. Assuming the game starts at v_0, to ensure this safety condition as a coalition, the players can apply the following memoryless strategy profile: in v_0, all players but Player 1 play a, and in v_1, all players play a. Under this strategy profile, for any number of players $k \geq 1$, all consistent plays avoid v_2 and v_3.

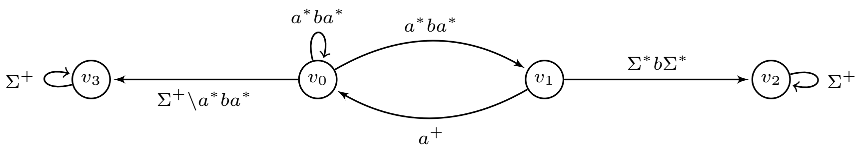

Figure 5 Parameterized arena for the synthesis of a coalition strategy.

In the sequel, we focus on safety objectives specified by a safe set of vertices $S \subseteq V$, and sketch how to decide whether there exists a strategy profile $\pi = (\sigma_1, \sigma_2, \cdots)$ that ensures to stay within S independently of the number of players.

Observe that a strategy profile for arbitrarily many players can equivalently be seen as a map $\pi : V^+ \to \Sigma^\omega$ from sequences of vertices to infinite words. Such a profile operates as follows: after observing a history $v_0v_1v_2 \cdots v_j$, if there are k players, the effect of profile π is the prefix of length k of $\pi(v_0v_1v_2 \cdots v_j)$.

As a first step towards deciding the Synthesis for arbitrarily-large coalition problem for safety objectives, we first unfold the arena into a tree $\mathcal{T}$, and stop a branch as soon as either it reaches a node labelled with an unsafe vertex $v \notin S$, or it reaches a node labelled with a vertex v that has an ancestor with same label. This construction is illustrated in Figure 6 on the example of Figure 5. The two left-most branches are stopped because of the repetition of the label v_0, and the two right-most branches are stopped because of the labelling by an unsafe vertex (v_2 or v_3). The size (i.e., number of nodes) of the tree unfolding $\mathcal{T}$ can be

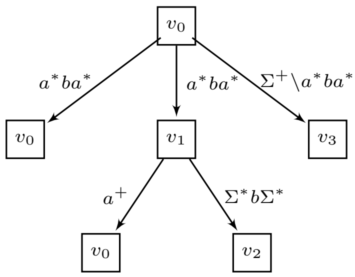

Figure 6 Finite unfolding of the arena from Figure 5 with $S = \{v_0, v_1\}$.

exponential in the number of vertices of the parameterized arena $\mathcal{A}$, however, its branching degree and height are linear in the size of $\mathcal{A}$.

The unfolding $\mathcal{T}$ can itself be seen as a parameterized arena, in which the objective of the coalition is to avoid unsafe branches that end in a node labelled with an unsafe vertex. Strategies of the coalition in $\mathcal{T}$ map inner nodes (nodes that are not leaves) to ω-words. One can show that the coalition has a winning strategy profile in $\mathcal{A}$ for the safety objective defined by S, if and only if it has a winning strategy profile in $\mathcal{T}$ to avoid unsafe branches. Intuitively, from a winning strategy profile in the tree, one can build a winning strategy profile in the arena with finite-memory bounded by the height of the tree. On our example, once we show there is a winning strategy profile $\pi_\mathcal{T}$ in the tree, one can define a winning strategy profile $\pi_\mathcal{A}$ in the arena by: $\pi_\mathcal{A}(V^+v_0) = \pi_\mathcal{T}(v_0)$ and $\pi_\mathcal{A}(V^+v_1) = \pi_\mathcal{T}(v_0v_1)$. Clearly enough, this justifies stopping a branch as soon as there is a repetition.

The second and most involved step is to characterize, at the tree unfolding level, the winning strategy profiles. If m is the number of inner nodes of $\mathcal{T}$, we show that one can effectively build a deterministic safety automaton $\mathcal{B}$ over Σ^m (thus reading one letter of the prescribed strategy at each inner node simultaneously) that accepts all infinite words in

$(\Sigma^m)^\omega$ that correspond to winning strategies in $\mathcal{T}$. Altogether, the Synthesis for arbitrarily-large coalition problem reduces to checking non-emptiness of $\mathcal{B}$. The latter being at most doubly exponential in the size of $\mathcal{A}$, we obtain an EXPSPACE complexity upper-bound.

5 Discussion

In this article, we reported on recent results about a new model of concurrent games, where the number of players is arbitrary [3]. Concurrent games with arbitrarily many players extend 2-player concurrent games, and more generally concurrent games with a fixed number of players. The edges in the arenas are equipped with languages over finite words. Such parameterized arenas can represent at once a denumerable number of standard arenas, each with a fixed number of players. They enable the definition and study of a number of parameterized game-theoretic problems.

We first considered a setting in which one player, say Eve, wants to achieve an objective independently of how many opponents she has, and whatever strategy they choose. We show that, for reachability objectives, deciding the existence of a uniform winning strategy for Eve against the rest of the world, is a PSPACE-complete problem. Second, we started to explore a synthesis problem, in which all players want to achieve an objective as a coalition. The difficulty here lies in the fact that they do not know *a priori* how many they are. For safety objectives, the existence of a coalition strategy is in EXPSPACE and PSPACE-hard.

We believe our preliminary work on parameterized arenas opens up many research paths. On the theoretical side, we currently put our effort on the coalition synthesis problem for reachability objectives. As an example, consider the parameterized arena from Figure 7, and assume the objective is to reach vertex v_1. Without knowing *a priori* how many they are,

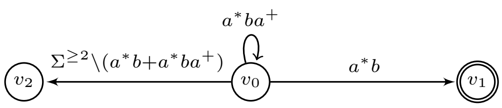

Figure 7 Synthesis for a reachability objective.

the players can collectively achieve this objective with the following profile: as long as the play is in v_0, at step i, Player i plays b while all other players play a. Under this strategy for the coalition, if there are k players, at step k the play moves from v_0 to v_1. In constrast to safety objectives, synthesizing such a symbolic strategy profile (or even deciding its existence) calls for more involved techniques.

Studying other solution concepts [8] such as Nash equilibria [12, 4], secure equilibria [6] and subgame perfect equilibria [10, 5] is also on our agenda.

On the practical side, we believe parameterized arenas could be used to represent a variety of distributed systems. We would be quite interested in exploring potential applications of this model that we find fascinating.

References

1 Luca de Alfaro, Thomas A. Henzinger, and Orna Kupferman. Concurrent reachability games. In *Proceedings of the 39th Annual Symposium on Foundations of Computer Science (FOCS'98)*, pages 564–575. IEEE Computer Society, 1998. `doi:10.1109/SFCS.1998.743507`.

2 Rajeev Alur, Thomas A. Henzinger, and Orna Kupferman. Alternating-time temporal logic. *Journal of the ACM*, 49:672–713, 2002. `doi:10.1145/585265.585270`.

3 Nathalie Bertrand, Patricia Bouyer, and Anirban Majumdar. Concurrent parameterized games. In *39th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'19)*, volume 150 of *LIPIcs*, pages 31:1–31:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. `doi:10.4230/LIPIcs.FSTTCS.2019.31`.

4 Patricia Bouyer, Romain Brenguier, Nicolas Markey, and Michael Ummels. Pure Nash equilibria in concurrent games. *Logical Methods in Computer Science*, 11(2:9), 2015. `doi:10.2168/LMCS-11(2:9)2015`.

5 Thomas Brihaye, Véronique Bruyère, Julie De Pril, and Hugo Gimbert. On subgame perfection in quantitative reachability games. *Logical Methods Computer Science*, 9(1), 2012. `doi:10.2168/LMCS-9(1:7)2013`.

6 Krishnendu Chatterjee, Thomas A. Henzinger, and Marcin Jurdzinski. Games with secure equilibria. *Theoretical Computer Science*, 365(1-2):67–82, 2006. `doi:10.1016/j.tcs.2006.07.032`.

7 Erich Grädel, Wolfgang Thomas, and Thomas Wilke, editors. *Automata, Logics, and Infinite Games: A Guide to Current Research*, volume 2500 of *Lecture Notes in Computer Science*. Springer, 2002. `doi:10.1007/3-540-36387-4`.

8 Erich Grädel and Michael Ummels. Solution concepts and algorithms for infinite multiplayer games. In *New perspectives on Games and Interaction*, volume 4 of *Texts in Logic and Games*. Amsterdam University Press, 2008.

9 Rohit Parikh. On context-free languages. *Journal of the ACM*, 13(4):570–581, 1966. `doi:10.1145/321356.321364`.

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

11 Larry J. Stockmeyer and Albert R. Meyer. Word problems requiring exponential time (preliminary report). In *Proceedings of the 5th Annual ACM Symposium on Theory of Computing (STOC'73)*, pages 1–9. ACM, 1973. `doi:10.1145/800125.804029`.

12 Michael Ummels and Dominik Wojtczak. The complexity of Nash equilibria in limit-average games. In *Proceedings of the 22nd International Conference on Concurrency Theory (CONCUR'11)*, volume 6901 of *Lecture Notes in Computer Science*, pages 482–496. Springer, 2011. `doi:10.1007/978-3-642-23217-6_32`.