Strategy Complexity of Zero-Sum Games on Graphs

Pierre Vandenhove\textsuperscript{1,2}

Thesis supervised by Patricia Bouyer\textsuperscript{1} and Mickael Randour\textsuperscript{2}

\textsuperscript{1}Université Paris-Saclay, CNRS, ENS Paris-Saclay, LMF, France
\textsuperscript{2}F.R.S.-FNRS & UMONS – Université de Mons, Belgium

March 14, 2023 – LMF Seminar
Context

- Present work part of my thesis.
- Thesis supervised by...

Patricia Bouyer, Université Paris-Saclay, LMF

Mickael Randour, Université de Mons, Belgium

- Thesis defense in Mons at the end of April.
Problem: **synthesis**

- An (incomplete, reactive) **system**,  
- living in an (uncontrollable) **environment**,  
- with a purpose/**specification**.

⇝ Modeling through a **zero-sum game**.
Zero-sum turn-based games on graphs

- **Colors** $C = \{a, b, c\}$, arena $A = (V_1, V_2, E)$.
- **Two players** $P_1$ (○) and $P_2$ (□).

- **Objective** of $P_1$ is a set $W \subseteq C^\omega$.
- **Zero-sum**: objective of $P_2$ is $C^\omega \setminus W$.

**Strategies**

A **strategy** of a player is a function $\sigma : E^* \rightarrow E$.

A strategy $\sigma$ of $P_1$ is **winning for** $W$ **from** $v \in V$ if all infinite paths from $v$ **consistent with** $\sigma$ induce an infinite word in $W$. 
Zero-sum games

Zero-sum turn-based games on graphs

- **Colors** $C = \{a, b, c\}$, arena $A = (V_1, V_2, E)$.
- **Two players** $P_1$ (○) and $P_2$ (□).

Infinite interaction

\[ \leadsto \text{infinite word } w = b \]

- **Objective** of $P_1$ is a set $W \subseteq C^\omega$.
- **Zero-sum**: objective of $P_2$ is $C^\omega \setminus W$.

**Strategies**

A **strategy** of a player is a function $\sigma : E^* \rightarrow E$.

A strategy $\sigma$ of $P_1$ is **winning for** $W$ **from** $v \in V$ if all infinite paths from $v$ **consistent with** $\sigma$ induce an infinite word in $W$. 
Zero-sum games

Zero-sum **turn-based** games on graphs

- **Colors** $C = \{a, b, c\}$, arena $A = (V_1, V_2, E)$.
- **Two players** $P_1$ (○) and $P_2$ (□).
  - Infinite interaction
    - $\leadsto$ infinite word $w = ba$
- **Objective** of $P_1$ is a set $W \subseteq C^\omega$.
- **Zero-sum**: objective of $P_2$ is $C^\omega \setminus W$.

**Strategies**

A **strategy** of a player is a function $\sigma : E^* \rightarrow E$.

A strategy $\sigma$ of $P_1$ is **winning for** $W$ **from** $v \in V$ if all infinite paths from $v$ **consistent with** $\sigma$ induce an infinite word in $W$. 
Zero-sum games

Zero-sum turn-based games on graphs

- **Colors** $C = \{a, b, c\}$, arena $A = (V_1, V_2, E)$.
- **Two players** $P_1$ (●) and $P_2$ (□).
  
  Infinite interaction
  $\rightsquigarrow$ **infinite word** $w = bab$

- **Objective** of $P_1$ is a set $W \subseteq C^\omega$.
- **Zero-sum**: objective of $P_2$ is $C^\omega \setminus W$.

**Strategies**

A **strategy** of a player is a function $\sigma : E^* \rightarrow E$.

A strategy $\sigma$ of $P_1$ is **winning for** $W$ from $v \in V$ if all infinite paths from $v$ **consistent with** $\sigma$ induce an infinite word in $W$. 
Zero-sum games

Zero-sum turn-based games on graphs

- Colors $C = \{a, b, c\}$, arena $A = (V_1, V_2, E)$.
- Two players $P_1$ (°) and $P_2$ (□).
  Infinite interaction
  $\rightsquigarrow$ infinite word $w = babb$
- Objective of $P_1$ is a set $W \subseteq C^\omega$.
- Zero-sum: objective of $P_2$ is $C^\omega \setminus W$.

Strategies

A strategy of a player is a function $\sigma : E^* \rightarrow E$.
A strategy $\sigma$ of $P_1$ is winning for $W$ from $v \in V$ if all infinite paths from $v$ consistent with $\sigma$ induce an infinite word in $W$. 
Zero-sum games

Zero-sum turn-based games on graphs

- **Colors** $C = \{a, b, c\}$, arena $A = (V_1, V_2, E)$.
- **Two players** $P_1$ (〇) and $P_2$ (□).
  Infinite interaction
  $\leadsto$ **infinite word** $w = babbc \ldots \in C^\omega$.
- **Objective** of $P_1$ is a set $W \subseteq C^\omega$.
- **Zero-sum**: objective of $P_2$ is $C^\omega \setminus W$.

**Strategies**

A **strategy** of a player is a function $\sigma : E^* \to E$.
A strategy $\sigma$ of $P_1$ is **winning for** $W$ from $v \in V$ if all infinite paths from $v$ **consistent with** $\sigma$ induce an infinite word in $W$. 
Strategy complexity

- Given a game and an initial vertex ⇝ **who can win?**
- To decide it, exhibit a **winning strategy** of a player.
- **Issues:**
  - strategies $\sigma : E^* \to E$ may not have a finite representation;
  - there are infinitely many of them.

Given an **objective**, when winning is possible, understand if **simple** strategies suffice to win, or if **complex** strategies are required.

Desirable properties:
- winning strategies can use bounded information (finite representation!);
- computable bounds on this information (finite number of strategies!).
Simple strategies

Memoryless strategies

A strategy is **memoryless** if it makes decisions based only on the **current arena vertex** \((\sigma: V_i \rightarrow E)\).

Finite-memory strategies

A strategy is **finite-memory** if it makes decisions based on the current arena vertex **and** the current state of a **memory structure** \((\sigma: V_i \times M \rightarrow E)\).

Finite **memory structure** \(M = (M, m_{\text{init}} \in M, \alpha_{\text{upd}}: M \times C \rightarrow M)\).

E.g., to remember whether \(a\) or \(b\) was last played:

```
a
\diamond m_1 \quad a
\quad a
m_2 \quad b
\quad b
\diamond
```

Memoryless strategies use **memory structure** \(\rightarrow C\).
Example

\[ C = \{a, b, c\}, \]
\[ W = \{w \in C^\omega \mid a \text{ is seen infinitely often and } b \text{ is seen infinitely often}\} \]

\[ \sigma(v_1, m_1) = \overset{c}{\rightarrow} v_2 \]
\[ \sigma(v_2, m_1) = \overset{b}{\rightarrow} v_2 \]
\[ \sigma(v_2, m_2) = \overset{c}{\rightarrow} v_1 \]
\[ \sigma(v_1, m_2) = \overset{a}{\rightarrow} v_2 \]

\(\rightsquigarrow\) Memoryless strategies do not suffice…
but **two memory states** do!
Example

\[ C = \{a, b, c\}, \]
\[ W = \{w \in C^\omega \mid a \text{ is seen infinitely often and } b \text{ is seen infinitely often}\} \]

\[ \sigma(v_1, m_1) = \rightarrow v_2 \]
\[ \sigma(v_2, m_1) = \rightarrow v_2 \]
\[ \sigma(v_2, m_2) = \rightarrow v_1 \]
\[ \sigma(v_1, m_2) = \rightarrow v_2 \]

\[ \sim \] Memoryless strategies do **not** suffice...
but **two memory states** do!
Example

\[ C = \{a, b, c\}, \]
\[ W = \{w \in C^\omega \mid a \text{ is seen } \infty \text{ly often and } b \text{ is seen } \infty \text{ly often}\} \]

\[ \sigma(v_1, m_1) = \xrightarrow{c} v_2 \]
\[ \sigma(v_2, m_1) = \xrightarrow{b} v_2 \]
\[ \sigma(v_2, m_2) = \xrightarrow{c} v_1 \]
\[ \sigma(v_1, m_2) = \xrightarrow{a} v_2 \]

\[ \leadsto \text{ Memoryless strategies do } \textbf{not} \text{ suffice...} \]
\[ \textbf{but two memory states do!} \]
Example

\[ C = \{ a, b, c \}, \]
\[ W = \{ w \in C^\omega \mid a \text{ is seen } \infty \text{ly often and } b \text{ is seen } \infty \text{ly often} \} \]

\[ \sigma(v_1, m_1) = c \xrightarrow{} v_2 \]
\[ \sigma(v_2, m_1) = b \xrightarrow{} v_2 \]
\[ \sigma(v_2, m_2) = c \xrightarrow{} v_1 \]
\[ \sigma(v_1, m_2) = a \xrightarrow{} v_2 \]

\[ \leadsto \] Memoryless strategies do not suffice…
but two memory states do!
Example

\[ C = \{a, b, c\}, \]
\[ W = \{ w \in C^\omega \mid a \text{ is seen infinitely often and } b \text{ is seen infinitely often} \} \]

\[ \sigma(v_1, m_1) = c \rightarrow v_2 \]
\[ \sigma(v_2, m_1) = b \rightarrow v_2 \]
\[ \sigma(v_2, m_2) = c \rightarrow v_1 \]
\[ \sigma(v_1, m_2) = a \rightarrow v_2 \]

\[ \sim \Rightarrow \text{Memoryless strategies do not suffice...} \]
\[ \text{but two memory states do!} \]
Example

\[ C = \{a, b, c\}, \]
\[ W = \{w \in C^\omega \mid a \text{ is seen } \infty \text{ly often and } b \text{ is seen } \infty \text{ly often}\} \]

\[ \sigma(v_1, m_1) = \xrightarrow{c} v_2 \]
\[ \sigma(v_2, m_1) = \xrightarrow{b} v_2 \]
\[ \sigma(v_2, m_2) = \xrightarrow{c} v_1 \]
\[ \sigma(v_1, m_2) = \xrightarrow{a} v_2 \]

\[ \rightsquigarrow \text{ Memoryless strategies do not suffice...} \]
\[ \text{but two memory states do!} \]
Finite-memory determinacy

Memoryless determinacy

An objective is memoryless-determined if in all arenas, memoryless strategies suffice for both players.

Finite-memory determinacy

An objective is finite-memory-determined if in all arenas, finite-memory strategies suffice for both players.

Various definitions depending on

- the class of arenas considered (finite, infinite, finitely branching...),
- whether we focus on both players or a single player.
State of the art: memoryless determinacy

Many “classical” objectives are memoryless-determined: reachability, Büchi, parity, energy, mean payoff, discounted sum...

Memoryless determinacy is well-understood:

- **Sufficient conditions** for both players,\(^1\) for a single player.\(^2\)
- **Characterizations** for both players over finite\(^3\)/infinite\(^4\) arenas, for a single player over infinite arenas.\(^5\)

---


\(^5\)Ohlmann, “Characterizing Positionality in Games of Infinite Duration over Infinite Graphs”, 2022.
State of the art: finite-memory determinacy

- **Finite-memory determinacy** is understood for specific objectives,\(^6\) but few results of wide applicability.\(^7\)
- Central class: \(\omega\)-regular objectives. Examples with \(C = \{a, b\} \):

  \[
  \omega\text{-regular expressions:} \quad \omega\text{-automata:} \quad \text{Linear temporal logic (LTL)}:
  \]

  \[
  b^* ab^* aC^\omega \quad \begin{array}{c}
  \text{q}_{\text{init}} \\
  \vdash
  \end{array} \quad \begin{array}{c}
  q_a \quad q_{aa} \\
  a \quad a, b
  \end{array} \quad \text{GF} a
  \]

**Theorem**\(^8,9\)

All \(\omega\)-regular objectives are finite-memory-determined.

---


Significance

Consequences of a fine-grained understanding of strategy complexity:

- **Decidability** of logical theories through FM det. (see *monadic second-order logic*, linked to $\omega$-regular objectives).
- Practical **synthesis** problems through FM det. (see, e.g., *LTL specifications*\(^\text{10}\)).
- At the core of algorithms to **solve** games (see, e.g., *parity games*\(^\text{11}\)).
- Controllers as **compact** as possible.

---

\(^{10}\) Pnueli, “The Temporal Logic of Programs”, 1977.

\(^{11}\) Zielonka, “Infinite Games on Finitely Coloured Graphs with Applications to Automata on Infinite Trees”, 1998.
Overview of our contributions

I. General conditions for finite-memory determinacy

- **Arbitrary** objectives
- Algebraic **characterizations** of the sufficiency of a memory structure for both players
- **Theoretical tools** to find memory structures
- Generalizations of memoryless determinacy results

II. Precise memory requirements of classes of objectives

- \( \omega \)-regular objectives
- **Observation**: memory requirements not settled
  1. Regular objectives (\( \approx \) DFAs)
     - Effective characterization of precise memory structures
     - (Computational) complexity
  2. Objectives recognizable by **deterministic Büchi automata**
     - Effective characterization of “no memory for \( \mathcal{P}_1 \)”
     - Complexity
I. General conditions for finite-memory determinacy
I. General conditions
for finite-memory determinacy

- Generalizations of memoryless determinacy results
- One-to-two-player lifts
- Algebraic characterizations of the sufficiency of a memory structure for both players
- Theoretical tools to help find memory structures

II. Precise memory requirements of classes of objectives

- \( \omega \)-regular objectives
- Observation: memory requirements not settled
- Regular objectives (≈ DFAs)
- Effective characterization of precise memory structures
- (Computational) complexity
- Objectives recognizable by deterministic Büchi automata
- Effective characterization of “no memory for player 1”
- Complexity

Strategy Complexity of Zero-Sum Games on Graphs
P. Vandenhove (supervised by P. Bouyer and M. Randour)
One-to-two-player lift

One-to-two-player memoryless lift (finite arenas)$^{12}$

Let $W \subseteq C^\omega$ be an objective. If

- in all one-player arenas of $P_1$, $P_1$ has memoryless winning strategies,
- in all one-player arenas of $P_2$, $P_2$ has memoryless winning strategies,

then both players have memoryless winning strategies in two-player arenas.

Extremely useful in practice. Very easy to recover memoryless determinacy of, e.g., parity$^{13}$ and mean-payoff$^{14}$ games.

What about finite-memory determinacy?

---


$^{13}$Emerson and Jutla, “Tree Automata, Mu-Calculus and Determinacy (Extended Abstract)”, 1991.

$^{14}$Ehrenfeucht and Mycielski, “Positional Strategies for Mean Payoff Games”, 1979.
What about finite-memory determinacy?

- **Counterexample** to a one-to-two-player lift for FM determinacy 😞.
- In the counterexample, the size of the memory depends on the size of the one-player arenas. Motivates the restriction to...

**Arena-independent memory**

An objective has *arena-independent finite-memory winning strategies* if

there exists a memory structure $M$ such that for all arenas $A$, strategies using $M$ suffice to win in $A$.

- Still holds for ω-regular objectives!
- Restriction over finite arenas, not so much over infinite arenas.
- **One-to-two-player lift works!**
# One-to-two-player lifts

When does memory determinacy in two-player zero-sum games reduce to one-player memory determinacy?

<table>
<thead>
<tr>
<th>Arenas</th>
<th>Str. comp.</th>
<th>Memoryless</th>
<th>FM “∃M∀A”</th>
<th>Mildly growing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finite</td>
<td></td>
<td>[GZ05]¹⁵</td>
<td>[BLORV22]¹⁶</td>
<td>[Koz22]¹⁷</td>
</tr>
<tr>
<td>Infinite</td>
<td></td>
<td>[CN06]¹⁸</td>
<td>[BRV23]¹⁹</td>
<td></td>
</tr>
<tr>
<td>Finite stochastic</td>
<td></td>
<td>[GZ09]²⁰</td>
<td>[BORV21]²¹</td>
<td></td>
</tr>
</tbody>
</table>

By-products of algebraic/language-theoretic characterizations.

¹⁸ For prefix-independent objectives; Colcombet and Niwiński, “On the positional determinacy of edge-labeled games”, 2006.
I. General conditions for finite-memory determinacy

- **Arbitrary** objectives
- Algebraic **characterizations** of the sufficiency of a memory structure for both players
- **Theoretical tools** to help find memory structures
  - One-to-two-player lifts
  - **Memory structures**
    - $\sim$ automata for the objectives
- Generalizations of memoryless determinacy results
Let $W \subseteq C^\omega$ be an objective.

**≈ Myhill-Nerode congruence**

For $x, y \in C^*$, $x \sim_W y$ if for all $z \in C^\omega$, $xz \in W \iff yz \in W$.

I.e., $x$ and $y$ have the same winning continuations; as good as each other.

**Properties**

- If $W$ is $\omega$-regular, then $\sim_W$ has finitely many equivalence classes.
- There is a DFA $S_W$ “prefix classifier” associated with $\sim_W$.

Might not “recognize” the language ($\neq$ languages of *finite* words)...
Two examples

...but we noticed a decomposition involving prefix classifiers and memory structures.

Let $C = \{a, b\}$.

<table>
<thead>
<tr>
<th>Objective</th>
<th>Prefix-classifier $S_W$</th>
<th>Sufficient memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>$W = b^* ab^* aC^\omega$</td>
<td>$b$ $a$ $b$ $a$ $b$ $a$</td>
<td>$C$</td>
</tr>
<tr>
<td>$W = \text{“a and b infinitely often”}$</td>
<td>$C$</td>
<td>$b$ $a$ $b$ $a$</td>
</tr>
</tbody>
</table>

Strategy Complexity of Zero-Sum Games on Graphs

P. Vandenhove (supervised by P. Bouyer and M. Randour)
Main result

Let $\mathcal{W} \subseteq \mathcal{C}^\omega$ be an objective.

**Theorem**

If a finite memory structure $\mathcal{M}$ suffices to play optimally in infinite arenas for both players, then

\[ \mathcal{W} \text{ is recognized by a parity automaton } (S_W \otimes \mathcal{M}, p). \]

\[ \Rightarrow \mathcal{W} \text{ is } \omega\text{-regular!} \]

Generalizes [CN06]\(^{22}\) (prefix-independent, memoryless case).

Let $W$ be an objective. $W$ is **finite-memory-determined** over infinite arenas $\iff W$ is $\omega$-regular.

$\iff$ is well-known.\(^{23,24}\)

$\implies$ follows from the previous slide.

---


Part I: Summary

- **Useful notion** of arena-independent FM determinacy.
- General **characterizations** over finite and infinite arenas.
- Theoretical **tools** to determine memory requirements.
- Central place of $\omega$-regular objectives.

**Related publications**

- Bouyer, Randour, V. (STACS’22 & TheoretiCS) “Characterizing Omega-Regularity through Finite-Memory Determinacy of Games on Infinite Graphs”

**Limits**

Wide applicability, but...

- not fully **effective**;
- in general, no tight memory requirements for **each** player.
II. Precise memory requirements of classes of objectives
II. Precise memory requirements of classes of objectives

- \(\omega\)-regular objectives
- Observation: memory requirements not settled

1. Regular objectives (\(\approx\) DFAs)
   - Effective characterization of precise memory structures
   - Existence of small structures is NP-complete

2. Objectives recognizable by deterministic Büchi automata
   - Effective characterization of “no memory for \(P_1\)"
   - Decidable in polynomial time
1 Regular objectives

Well-understood $\omega$-regular objectives: *Muller conditions*, focusing on what is seen **infinitely often**.\(^25, 26\)

E.g., $b^*ab^*aC^\omega$ is not a Muller condition.

### Missing pieces

Orthogonal quest: objectives where "**finite prefixes matter**".

We consider the “simplest” ones.

### Regular objectives

- A **regular reachability objective** is a set $LC^\omega$ with $L \subseteq C^*$ regular.
- A **regular safety objective** is a set $C^\omega \setminus LC^\omega$.

Expressible as standard **deterministic finite automata**.

---


\(^{26}\) Dziembowskii, Jurdziński, and Walukiewicz, “How Much Memory is Needed to Win Infinite Games?”, 1997.
Question

Memory requirements of regular objectives

**Characterize the memory structures** that suffice to make optimal decisions for regular objectives **in any arena**. Compute **minimal** ones.

Idea

- A DFA recognizing the language, taken as a memory structure, always suffices for both players.
- But can be much smaller in general!
- Properties linked to the Myhill-Nerode congruence.
Comparing words

Let $W \subseteq C^\omega$ be an objective.

Comparing prefixes

For $x, y \in C^*$, $x \preceq_W y$ if for all $z \in C^\omega$, $xz \in W \implies yz \in W$.

I.e., $y$ has more winning continuations than $x$; better situation.

Example

Let $W$ be the regular reachability objective induced by this DFA.

E.g., $\varepsilon \prec_W a$, $a \prec_W ab$, $a$ and $b$ are incomparable for $\preceq_W$. 
Necessary condition for the memory

Let $W \subseteq C^\omega$ be an objective.

**Lemma**

A sufficient memory structure $\mathcal{M} = (M, m_{\text{init}}, \alpha_{\text{upd}})$ needs to distinguish incomparable words (for $\preceq_W$), i.e.,

$$\text{if } x, y \in C^* \text{ are incomparable for } \preceq_W, \text{ then } \alpha^*_{\text{upd}}(m_{\text{init}}, x) \neq \alpha^*_{\text{upd}}(m_{\text{init}}, y).$$

Why? (Example) need to make the right decision in this arena.
Characterizations

Theorem

Let $W$ be a regular safety objective.

A memory structure $\mathcal{M}$ suffices in all arenas for $P_1$ if and only if $\mathcal{M}$ distinguishes incomparable words.

Theorem

Let $W$ be a regular reachability objective.

Memory structure $\mathcal{M}$ suffices in all arenas for $P_1$ if and only if $\mathcal{M}$ distinguishes incomparable words and $\mathcal{M}$ distinguishes insufficient progress.
### Decision problems

**Input:** An automaton $D$ inducing the regular **reachability** (or **safety**) objective $W$ and $k \in \mathbb{N}$.

**Question:** $\exists$ a memory structure $M$ with $\leq k$ states that suffices for $W$?

Thanks to the “effectiveness” of the two properties, we showed that:

**Theorem**

These problems are NP-complete.
Implementation

Algorithms that find minimal memory structures for regular objectives, using a **SAT solver**.

\[
D = \text{memReq.smallest_memory_safety}(D)
\]

\[
M = \text{memReq.smallest_memory_safety}(D)
\]
II. Precise memory requirements of classes of objectives

- \( \omega \)-regular objectives
- **Observation:** memory requirements not settled

1. Regular objectives (\( \approx \) DFAs)
   - Effective characterization of precise memory structures
   - Existence of small structures is NP-complete

2. Objectives recognizable by deterministic Büchi automata
   - Effective characterization of “no memory for \( P_1 \)”
   - Decidable in polynomial time
Deterministic Büchi automata

A deterministic Büchi automaton \( B \) on \( C \)

- reads infinite words (in \( C^\omega \)),
- accepts words that see infinitely many Büchi transitions \( \bullet \).

\[
L(B) = \{ w \in \{ a, b \}^\omega \mid w \text{ sees } \infty \text{ly many } a \text{ and } \infty \text{ly many } b \}
\]

Question

Given \( B \), can \( P_1 \) win without memory for objective \( W = L(B) \)?
(Is \( L(B) \) half-positional?)
2 Results

Let $\mathcal{B}$ be a deterministic Büchi automaton.

**Theorem**

For objective $W = \mathcal{L}(\mathcal{B})$, $\mathcal{P}_1$ does not need memory if and only if

- all prefixes are comparable for $\preceq_W$,
- $W$ is progress-consistent, and
- $W$ is recognized by its prefix classifier as a DBA.

**Polynomial-time algorithm**

Can be **decided** in $O(|\mathcal{B}|^4)$ time.
Part II: Summary

• Tools to study memory req. of classes of \( \omega \)-regular objectives.
• Effective **characterizations** for DFAs and DBAs.
• **Decidability** and **complexity** of the related decision problems.

Related publications

• Bouyer, Fijalkow, Randour, V. (Submitted preprint) “How to Play Optimally for Regular Objectives?”
• Bouyer, Casares, Randour, V. (CONCUR’22) “Half-Positional Objectives Recognized by Deterministic Büchi Automata”
Future works

• *(Part I)* General results for arena-dependent memory requirements.
  ▶ Observing *edges* rather than colors in the model.
  ▶ Well-behaved nondeterminism (*history-determinism*).\textsuperscript{27}

• *(Part II)* Automatically *compute minimal memory structures* for all \(\omega\)-regular objectives?

• More expressive *settings* (e.g., stochastic, concurrent,\textsuperscript{28} or timed games).

• More expressive *strategy models* than finite-state machines (e.g., pushdown\textsuperscript{29} or register\textsuperscript{30} automata).

Thanks!

\textsuperscript{29} Walukiewicz, “Pushdown Processes: Games and Model-Checking”, 2001.
\textsuperscript{30} Exibard et al., “Computability of Data-Word Transductions over Different Data Domains”, 2022.