Is your automaton good for playing games?

Marc Bagnol, Denis Kuperberg

CNRS, LIP, ENS Lyon

Highlights 2018, Berlin
Solving an $\omega$-regular game

**Input:** $G$ game with $\omega$-regular winning condition $W \subseteq V^\omega$.

**Question:** Who wins $G$? How?

**Solution:**

1. Build Deterministic Parity automaton $A^{\text{Det}}$ for $W$,
2. Solve the parity game $G' = G \circ A^{\text{Det}}$.

**Theorem** $G \circ A^{\text{Det}}$ has the same winner as $G$.

**Problem:** Determinization is expensive. Maybe too strong?

**Definition (Henzinger, Piterman 2006)**

$A$ is Good-for-Games (GFG) if $G \circ A$ has the same winner as $G$, for any game $G$ with winning condition $L(A)$.
Solving an $\omega$-regular game

**Input:** $G$ game with $\omega$-regular winning condition $W \subseteq V^\omega$.

**Question:** Who wins $G$? How?

**Solution:** Turn $G$ into a game with more states but simpler acceptance condition.

1. Build Det Parity automaton $A_{\text{Det}}$ for $W$,
2. Solve the parity game $G' = G \circ A_{\text{Det}}$.

**Theorem**

$G \circ A_{\text{Det}}$ has same winner as $G$. 

**Problem:** Determinization is expensive. Maybe too strong?

**Definition (Henzinger, Piterman 2006)**

$A$ is Good-for-Games (GFG) if $G \circ A$ has same winner as $G$, for any game $G$ with winning condition $L(A)$. 

Solving an $\omega$-regular game

**Input:** $G$ game with $\omega$-regular winning condition $W \subseteq V^\omega$.

**Question:** Who wins $G$? How?

**Solution:** Turn $G$ into a game with more states but simpler acceptance condition.

1. Build Det Parity automaton $A_{Det}$ for $W$,
2. Solve the parity game $G' = G \circ A_{Det}$.

**Theorem**

$G \circ A_{Det}$ has same winner as $G$.

**Problem:** Determinization is expensive. Maybe too strong?
Solving an $\omega$-regular game

**Input:** $G$ game with $\omega$-regular winning condition $W \subseteq V^\omega$.

**Question:** Who wins $G$? How?

**Solution:** Turn $G$ into a game with more states but simpler acceptance condition.

1. Build Det Parity automaton $A_{Det}$ for $W$,
2. Solve the parity game $G' = G \circ A_{Det}$.

**Theorem**

$G \circ A_{Det}$ has same winner as $G$.

**Problem:** Determinization is expensive. Maybe too strong?

**Definition (Henzinger, Piterman 2006)**

$A$ is Good-for-Games (GFG) if $G \circ A$ has same winner as $G$, for any game $G$ with winning condition $L(A)$. 
Definition of GFG via a game

A Parity automaton, we associate to it a **GFG game**: 
Adam plays letters:

Eve: resolves non-deterministic choices for transitions
Definition of GFG via a game

A Parity automaton, we associate to it a GFG game:
Adam plays letters: $a$
Eve: resolves non-deterministic choices for transitions
Definition of GFG via a game

A Parity automaton, we associate to it a GFG game:

Adam plays letters: $a \ a$

Eve: resolves non-deterministic choices for transitions
A Parity automaton, we associate to it a **GFG game**:

**Adam** plays letters: \(a\ a\ b\)

**Eve**: resolves non-deterministic choices for transitions
Definition of GFG via a game

A Parity automaton, we associate to it a **GFG game**:

Adam plays letters: \(a\ a\ b\ c\)

Eve: resolves non-deterministic choices for transitions
Definition of GFG via a game

A Parity automaton, we associate to it a GFG game:
Adam plays letters: $a \ a \ b \ c \ c$
Eve: resolves non-deterministic choices for transitions
Definition of GFG via a game

A Parity automaton, we associate to it a **GFG game**: 
Adam plays letters: $a \ a \ b \ c \ c \ldots = \ w$

**Eve**: resolves non-deterministic choices for transitions

Eve wins if: $w \in L \Rightarrow \text{Run accepting.}$
Definition of GFG via a game

A Parity automaton, we associate to it a **GFG game**:

*Adam* plays letters: \( a \ a \ b \ c \ c \ldots = w \)

*Eve*: resolves non-deterministic choices for transitions

\[
\begin{array}{c}
\text{a, b, c} \\
\text{a} \\
\text{b, c} \\
\text{b} \\
\text{c} \\
\text{a, b, c}
\end{array}
\]

*Eve* wins if: \( w \in L \Rightarrow \text{Run accepting.} \)

\( A \text{ GFG} \iff \text{Eve wins the GFG game on } A. \)
Recognizing GFG automata

GFGness problem: input $A_{ND}$, is it GFG?
Recognizing GFG automata

**GFGness problem**: input $A_{ND}$, is it GFG?

Solve the GFG game?
Recognizing GFG automata

**GFGness problem**: input $A_{\text{ND}}$, is it GFG?

Solve the GFG game?
Acceptance condition of the form $u \in L \Rightarrow \text{run accepting}$
Recognizing GFG automata

**GFGness problem:** input $A_{ND}$, is it GFG?

Solve the GFG game?
Acceptance condition of the form “$u \in L \Rightarrow \text{run accepting}$”

*Upper bound:* EXPTIME
Recognizing GFG automata

**GFGness problem**: input $A_{ND}$, is it GFG?

Solve the GFG game?
Acceptance condition of the form “$u \in L \Rightarrow \text{run accepting}”

*Upper bound*: EXPTIME

**Theorem (Löding)**
The GFGness problem is in $P$ for reachability/safety automata.

**Theorem (K., Skrzypczak 2015)**
The GFGness problem is in $P$ for coBüchi automata.

Parity Games $\equiv$ GFGness for universal Parity automata.
Recognizing GFG automata

**GFGness problem**: input $A_{ND}$, is it GFG?

Solve the GFG game?
Acceptance condition of the form “$u \in L \implies$ run accepting”

*Upper bound*: EXPTIME

**Theorem (Löding)**
The GFGness problem is in $P$ for reachability/safety automata.

**Theorem (K., Skrzypczak 2015)**
The GFGness problem is in $P$ for coBüchi automata.
*Parity Games $\equiv$ GFGness for universal Parity automata.*

**This talk:**
**Theorem (Bagnol, K. (unpublished))**
The GFGness problem is in $P$ for Büchi automata.
Abstracting the GFG game

The game $G_2$:
Adam plays letters:
Eve: moves one token Adam: moves two tokens

Eve wins if:
1. or 2. accepts $\Rightarrow$ accepts.

Goal: Show that Eve wins $G_2 \iff$ Eve wins the GFG game.
Abstracting the GFG game

The game $G_2$:

Adam plays letters: $a$

Eve: moves one token  Adam: moves two tokens

Eve wins if:

1. or

Goal: Show that $Eve$ wins $G_2$ $\iff$ Eve wins the GFG game.
Abstracting the GFG game

The game $G_2$:
- **Adam** plays letters: $a$
- **Eve**: moves one token **Adam**: moves two tokens

Eve wins if:
1. or
2. accepts $\Rightarrow$ accepts.

**Goal**: Show that Eve wins $G_2 \iff$ Eve wins the GFG game.
Abstracting the GFG game

The game $G_2$:
- **Adam** plays letters: $a$, $a$
- **Eve**: moves one token  
- **Adam**: moves two tokens

Eve wins if:
1. or 2. accepts $\Rightarrow$ accepts.

Goal: Show that Eve wins $G_2 \iff$ Eve wins the GFG game.
Abstracting the GFG game

The game $G_2$:

Adam plays letters: $a \ a$

Eve: moves one token Adam: moves two tokens

Eve wins if:

1. or

Goal: Show that Eve wins $G_2 \iff$ Eve wins the GFG game.
Abstracting the GFG game

The game $G_2$:
- Adam plays letters: $a \ a \ b$
- Eve: moves one token  
  Adam: moves two tokens

Eve wins if:
1. or
2. accepts $\Rightarrow$ accepts.

Goal: Show that Eve wins $G_2 \iff$ Eve wins the GFG game.
Abstracting the GFG game

The game $G_2$:

Adam plays letters: $a \ a \ b$

Eve: moves one token Adam: moves two tokens

Goal: Show that Eve wins $G_2$ ⇔ Eve wins the GFG game.
Abstracting the GFG game

The game $G_2$:
Adam plays letters: $a$ $a$ $b$ $c$
Eve: moves one token Adam: moves two tokens

Eve wins if:
1. or
2. accepts $\Rightarrow$ accepts.

Goal: Show that Eve wins $G_2$ $\iff$ Eve wins the GFG game.
Abstracting the GFG game

The game $G_2$:
Adam plays letters: $a \ a \ b \ c$
Eve: moves one token  Adam: moves two tokens

Eve wins if:
1 or 2 accepts $\Rightarrow$ accepts.

Goal: Show that Eve wins $G_2 \iff$ Eve wins the GFG game.
Abstracting the GFG game

The game $G_2$:
Adam plays letters: $a \ a \ b \ c \ \ldots \ = \ w$
Eve: moves one token  Adam: moves two tokens

Eve wins if: $\blacksquare_1$ or $\blacksquare_2$ accepts $\Rightarrow$ $\bigcirc$ accepts.
Abstracting the GFG game

The game $G_2$:

Adam plays letters: $a \ a \ b \ c \ldots = w$

Eve: moves one token  Adam: moves two tokens

Eve wins if: $\blacksquare_1$ or $\blacksquare_2$ accepts $\Rightarrow \bigcirc$ accepts.

Goal: Show that Eve wins $G_2$ $\iff$ Eve wins the GFG game.
Proof sketch

Lemma

Eve wins $G_2 \iff$ Eve wins $G_k$ for all $k$.

Proof sketch: how to win against $k + 1$ tokens:

- play a virtual token $\bigcirc$ against the first $k$ tokens
- play the $G_2$ strategy against the virtual token and the remaining token.
Main proof sketch for $G_2 \Leftrightarrow \text{GFG}$

Assume:
- Adam wins the GFG game with finite-memory strategy $\tau_{\text{GFG}}$.
- Eve wins $G_2 \Rightarrow$ wins $G_k$ with strategy $\sigma_k$, for a big $k$. 

Build strategy for Eve against $\tau_{\text{GFG}}$:
- move $k$ virtual tokens against $\tau_{\text{GFG}}$
- play $\sigma_k$ against these $k$ tokens...
Main proof sketch for $G_2 \Leftrightarrow \text{GFG}$

Assume:
- Adam wins the GFG game with finite-memory strategy $\tau_{\text{GFG}}$.
- Eve wins $G_2 \Rightarrow$ wins $G_k$ with strategy $\sigma_k$, for a big $k$.

Build strategy for Eve against $\tau_{\text{GFG}}$:
- move $k$ virtual tokens $\bigcirc$ against $\tau_{\text{GFG}}$
- play $\sigma_k$ against these $k$ tokens

\[\text{at most } M \text{ steps} \]

\[\text{initial state} \rightarrow \text{Büchi state} \geq N \text{ tokens}\]
Conclusion

Results

▶ Characterisation of Büchi GFG automata via $G_2$.
▶ → Büchi GFGness $\in P$, actually in $O(n^4 m(n + m)|\Sigma|^2)$.

Perspectives

▶ is $G_2$ equivalent to GFG for coBüchi, Parity ?
▶ if Yes, Parity GFGness $\in P$ for any fixed parity condition.
▶ recognizing GFG automata $\rightarrow$ building them.