Game Quantification on Automatic Structures

Łukasz Kaiser
kaiser@informatik.rwth-aachen.de

Mathematische Grundlagen der Informatik
RWTH Aachen

Isaac Newton Institute Workshop
Games and Verification
Overview

Introduction

Logic with Game Quantifier

Hierarchical Model Checking Games

Conclusions and Future Work
Game Quantification

Build formulas using infinite string of (alternating) quantifiers

\[(\exists x_1 \forall y_1 \exists x_2 \forall y_2 \ldots) R(x_1, y_1, x_2, y_2, \ldots)\]

- $R$ is a set of infinite sequences, classically open or closed
  \[R = \bigvee_i R_i(x_1, y_1, \ldots, x_i, y_i)\]
- semantics given using Gale-Stewart games, first player wins $G(\exists \forall, R)$
- duality under negation follows from determinacy for Borel $R$ (Martin)
- traditionally compared to infinitary or second-order logic

Game quantification on structures where the universe is a finite set of letters $\Sigma$ and $R$ is given by an automaton.
We are working on structures on finite and infinite words which are presentations of $\omega$-automatic structures

$$(\Sigma^{\leq \omega}, R_1, \ldots, R_K)$$

Each $R_i$ is recognised by a Muller automaton over $(\Sigma \cup \{\square\})^\text{arity}(R_i)$. Finite words are encoded by adding $\square^\omega$ suffix.

$$w^1 \otimes \ldots \otimes w^k = \begin{bmatrix} x_1^1 \\ \vdots \\ x_k^1 \\ \vdots \\ x_1^k \\ \vdots \\ x_k^k \\ \vdots \\ x_i^k \\ \vdots \\ x_i^k \\ \vdots \\ x_{i+1}^k \end{bmatrix} \ldots \begin{bmatrix} \square \\ \vdots \\ \square \end{bmatrix} \begin{bmatrix} x_1^1 \\ \vdots \\ x_1^k \\ \vdots \\ x_i^k \\ \vdots \\ x_{i+1}^k \end{bmatrix} \ldots$$
Overview

Introduction

Logic with Game Quantifier

Hierarchical Model Checking Games

Conclusions and Future Work
$\exists x y \, \varphi(x, y)$
Game Quantifier

\[ \exists x y \varphi(x, y) \]

Player I \quad x = a

Player II \quad y =
$\exists x y \varphi(x, y)$

**Player I** \hspace{1cm} \( x = a \)

**Player II** \hspace{1cm} \( y = b \)
\( \exists x y \, \varphi(x, y) \)

**Player I**

\[ x = a b \]

**Player II**

\[ y = b \]
Game Quantifier

\[ \exists x y \varphi(x, y) \]

**Player I** \[ x = a b \]

**Player II** \[ y = b b \]
\( \exists x y \varphi(x, y) \)

**Player I** \( x = a \ b \ b \)

**Player II** \( y = b \ b \)
\( \exists x y \, \varphi(x, y) \)

Player I  \( x = abba \)

Player II  \( y = bbaa \)
\( \exists xy \varphi(x, y) \)

Player I \( x = a \ b \ b \ a \)

Player II \( y = b \ b \ a \)
\[ \exists x y \, \varphi(x, y) \]

**Player I**  \[ x = a b b a \]

**Player II**  \[ y = b b a b \]
\[ \exists x y \varphi(x, y) \]

**Player I** \[ x = a \ b \ b \ a \ a \]

**Player II** \[ y = b \ b \ a \ b \]
\( \exists x y \varphi(x, y) \)

**Player I** \( x = a b b a a \)

**Player II** \( y = b b a b b \)
\[ \exists x y \varphi(x, y) \]

**Player I** \( x = a b b a a \ldots \)

**Player II** \( y = b b a b b \)
\(\exists x y \varphi(x, y)\)

**Player I** \(x = a \ b \ b \ a \ a \ldots\)

**Player II** \(y = b \ b \ a \ b \ b \ldots\)
\( \exists x y \varphi(x, y) \)

Player I \( x = a \ b \ b \ a \ a \ldots \)

Player II \( y = b \ b \ a \ b \ b \ldots \)

Can Player I play so that however Player II plays \( \varphi(x, y) \) holds?
Game Quantifier Formally

\( \exists xy \, \varphi(x, y) \iff \) 

(\exists \text{ well-formed } f : \Gamma^* \times \Gamma^* \to \Gamma)

(\forall \text{ well-formed } g : \Gamma^* \times \Gamma^* \to \Gamma) \, \varphi(x_{fg}, y_{fg}),

\( \Gamma = \Sigma \cup \{\square\}, \) \( x_{fg} \) and \( y_{fg} \) are constructed inductively using \( f \) and \( g \)

\( x_{fg}[n] = f(x_{fg}|_{n-1}, y_{fg}|_{n-1}) \)

\( y_{fg}[n] = g(x_{fg}|_{n}, y_{fg}|_{n-1}) \)

\textbf{well-formed} \( f \) : if \( f \) outputs \( \Box \) then it continues to output \( \Box \) infinitely

Coincides with the classical definition

\( \exists xy \, \varphi(x, y) \iff (\exists x_1 \forall y_1 \exists x_2 \forall y_2 \ldots) \, \varphi(x_1 x_2 \ldots, y_1 y_2 \ldots) \)
Game Formula Example

\[ R(u, w, s, t) := \exists xy \ (y = u \rightarrow x = s) \land (y = w \rightarrow x = t) \]
$R(u, w, s, t) := \exists xy \ (y = u \rightarrow x = s) \land (y = w \rightarrow x = t)$

**common prefix** of $s$ and $t$ is **longer** than the **common prefix** of $u$ and $w$

$R(u, w, s, t) \equiv |s \cap t| > |u \cap w| \ (s \neq t, u \neq w)$
Game Formula Example

\[ R(u, w, s, t) := \forall xy (y = u \rightarrow x = s) \land (y = w \rightarrow x = t) \]

**common prefix** of \( s \) and \( t \) is **longer** than the **common prefix** of \( u \) and \( w \)

\[ R(u, w, s, t) \equiv |s \cap t| > |u \cap w| \quad (s \neq t, u \neq w) \]

(\( \Leftarrow \)) **assume** \( |s \cap t| > |u \cap w| \)

**Player II** will have to choose \( y = u \) or \( y = w \) before **Player I** chooses if \( x = s \) or if \( x = t \).
Game Formula Example

\[ R(u, w, s, t) := \exists xy (y = u \rightarrow x = s) \land (y = w \rightarrow x = t) \]

common prefix of \(s\) and \(t\) is longer than the common prefix of \(u\) and \(w\)

\[ R(u, w, s, t) \equiv |s \cap t| > |u \cap w| \quad (s \neq t, u \neq w) \]

(\(\Leftarrow\)) assume \(|s \cap t| > |u \cap w|\)

Player II will have to choose \(y = u\) or \(y = w\) before Player I chooses if \(x = s\) or if \(x = t\).

(\(\Rightarrow\)) in the other case

Player II knows if \(x = s\) or if \(x = t\) before choosing whether \(y = u\) or \(y = w\) and can therefore win.
Determinacy of FO+⊤

Negating game quantifier reverses move order

\[ \mathcal{A}, \bar{z} \models \neg \exists x y \varphi(\bar{x}, \bar{y}, \bar{z}) \iff \mathcal{A}, \bar{z} \models \forall y x \neg \varphi(\bar{x}, \bar{y}, \bar{z}) \]

**Negation normal form** for FO+⊤.

Follows from determinacy of finitely coloured Muller games. Alternatively can be proved using the general theorem of Martin.
Decidability of FO+Ω

Game quantifier makes automata alternating
Decidability of FO+$\bigcirc$

Game quantifier makes automata alternating

Lemma

If $R(\overline{x}, \overline{y}, \overline{z})$ is $\omega$-regular then $\bigcirc xy\ R(\overline{x}, \overline{y}, \overline{z})$ is $\omega$-regular as well.
Game quantifier makes automata alternating

**Lemma**

If $R(\overline{x}, \overline{y}, \overline{z})$ is $\omega$-regular then $\exists \overline{x} \overline{y} \ R(\overline{x}, \overline{y}, \overline{z})$ is $\omega$-regular as well.

**Proof method**

$$\delta(\overline{q}, \overline{z}) = \bigvee_{\overline{x} \in \Gamma^k} \bigwedge_{\overline{y} \in \Gamma^l} \delta_R(\overline{q}, \overline{x} \otimes \overline{y} \otimes \overline{z})$$

where $k$ is the size of $\overline{x}$, $l$ the size of $\overline{y}$, $\delta$ are transition functions.
Game quantifier makes automata alternating

Lemma

If $R(\bar{x}, \bar{y}, \bar{z})$ is $\omega$-regular then $\exists \bar{x} \bar{y} \ R(\bar{x}, \bar{y}, \bar{z})$ is $\omega$-regular as well.

Proof method

$$\delta_{\exists}(q, \bar{z}) = \bigvee_{\bar{x} \in \Gamma^k} \bigwedge_{\bar{y} \in \Gamma^l} \delta_R(q, \bar{x} \otimes \bar{y} \otimes \bar{z})$$

where $k$ is the size of $\bar{x}$, $l$ the size of $\bar{y}$, $\delta$ are transition functions.

Alternating automata can be determinized with double exponential blowup.
Expressive power of FO$^+$

We have already defined $|s \sqcap t| > |u \sqcap w|$
We have already defined $|s \sqcap t| > |u \sqcap w|$
use boolean combinations for $<, \leq, \geq, =$
We have already defined \( |s \cap t| > |u \cap w| \)

use boolean combinations for \(<, \leq, \geq, =\)

\[ x \sqsubseteq y \equiv (\forall z \neq x) \ |x \cap z| \leq |x \cap y| \]
We have already defined $|s \sqcap t| > |u \sqcap w|$

use boolean combinations for $<, \leq, \geq, =$

$x \sqsubseteq y \equiv (\forall z \neq x) |x \sqcap z| \leq |x \sqcap y|$  
$|x| \leq |y| \equiv (\forall x' \neq x) (\exists y' \neq y) |x \sqcap x'| \leq |y \sqcap y'|$
We have already defined $|s \cap t| > |u \cap w|$ use boolean combinations for $<, \leq, \geq, =$

$x \sqsubseteq y \equiv (\forall z \neq x) \ |x \cap z| \leq |x \cap y|$

$|x| \leq |y| \equiv (\forall x' \neq x) \ (\exists y' \neq y) \ |x \cap x'| \leq |y \cap y'|$

Game quantifier gives us prefix and equal length.
We have already defined $|s \cap t| > |u \cap w|$
use boolean combinations for $<, \leq, \geq, =$

$x \sqsubseteq y \equiv (\forall z \neq x) |x \cap z| \leq |x \cap y|$

$|x| \leq |y| \equiv (\forall x' \neq x) (\exists y' \neq y) |x \cap x'| \leq |y \cap y'|$

Game quantifier gives us prefix and equal length.

$\text{FO}^+\exists$ can define all regular relations

- on the binary tree with successor relations
- on binary coded numbers with number equality
Definition

The bijection $\pi : \Sigma^{\leq \omega} \to \Sigma^{\leq \omega}$ is inductive when there is a family of permutations of $\Sigma \{\pi_w\}_{w \in \Sigma^*}$ so that for each word $u$

$$\pi(u)[n] = \pi_{u|_{n-1}}(u[n]).$$
Invariance under Inductive Automorphisms

Definition

The bijection $\pi : \Sigma^{\leq \omega} \rightarrow \Sigma^{\leq \omega}$ is inductive when there is a family of permutations of $\Sigma \{\pi_w\}_{w \in \Sigma^*}$ so that for each word $u$

$$\pi(u)[n] = \pi_{u|_{n-1}}(u[n]).$$

Theorem

If $\phi$ is an inductive automorphism of $\mathfrak{A} = (\Sigma^{\leq \omega}, R_1, \ldots, R_k)$ and $R$ is definable in $FO^{+\Box}$ on $\mathfrak{A}$, then

$$R(\overline{x}) \iff R(\overline{\phi(x)}).$$
Invariance under Inductive Automorphisms

**Definition**

The bijection \( \pi : \Sigma^{\leq \omega} \rightarrow \Sigma^{\leq \omega} \) is **inductive** when there is a family of permutations of \( \Sigma \ \{ \pi_w \}_{w \in \Sigma^*} \) so that for each word \( u \)

\[
\pi(u)[n] = \pi_{u|_{n-1}}(u[n]).
\]

**Theorem**

If \( \phi \) is an inductive automorphism of \( \mathcal{A} = (\Sigma^{\leq \omega}, R_1, \ldots, R_k) \) and \( R \) is definable in \( FO^{\oplus} \) on \( \mathcal{A} \), then

\[
R(\overline{x}) \iff R(\overline{\phi(x)}).
\]

**Corollary**

The word \( a^\omega \) is not definable in \( FO^{\oplus} \) only with equality.
Overview

Introduction

Logic with Game Quantifier

Hierarchical Model Checking Games

Conclusions and Future Work
Model Checking Game Example

Take the formula \( \exists x \ (R_1(x) \land R_2(x)) \)

\[ R_1 = \{ a^\omega \} \]

\[ R_2 = \{ a, b \}^\omega \setminus \{ a^\omega \} \]
Take the formula $\exists x \ (R_1(x) \land R_2(x))$

$R_1 = \{ a^\omega \}$

$R_2 = \{ a, b \}^\omega \setminus \{ a^\omega \}$

$x$ must be given before $\land$ branch is chosen

$\land$ on a **higher level of information** than $\exists x$
Take the formula $\exists x \left( R_1(x) \land R_2(x) \right)$

$R_1 = \{ a^\omega \}$

$R_2 = \{ a, b \}^\omega \setminus \{ a^\omega \}$

$x$ must be given before $\land$ branch is chosen

$\land$ on a higher level of information than $\exists x$
Take the formula \( \exists x (R_1(x) \land R_2(x)) \)

\[ R_1 = \{ a^\omega \} \]

\[ R_2 = \{ a, b \}^\omega \setminus \{ a^\omega \} \]

\( x \) must be given before \( \land \) branch is chosen \( \land \) on a **higher level of information** than \( \exists x \)
Model Checking Game Example

Take the formula $\exists x \ (R_1(x) \land R_2(x))$

$R_1 = \{a^\omega\}$

$R_2 = \{a, b\}^\omega\setminus\{a^\omega\}$

$x$ must be given before $\land$ branch is chosen

$\land$ on a higher level of information than $\exists x$
Model Checking Game Example

Take the formula $\exists x \ (R_1(x) \land R_2(x))$

$R_1 = \{ a^\omega \}$

$R_2 = \{ a, b \}^\omega \setminus \{ a^\omega \}$

$x$ must be given before $\land$ branch is chosen

$\land$ on a **higher level of information** than $\exists x$

- quantifier alternation $\leadsto$ different levels of information
- game quantifier $\leadsto$ opposing players on the same level of information
Hierarchical Muller Games

- Two coalitions $\mathcal{I}$ and $\mathcal{II}$ on $N$ levels of information, two players on each level ($2N$ players).

  On level $i$ players **see moves on levels** $j \leq i$
  but **can not see moves on levels** $j > i$
Hierarchical Muller Games

- Two coalitions I and II on N levels of information, two players on each level (2N players).

  On level i players see moves on levels $j \leq i$ but can not see moves on levels $j > i$

- usually winning defined when there exists a winning strategy good for all counter-strategies but this contradicts information advantage, here strategies must be given level-by-level,
Hierarchical Muller Games

- Two coalitions \( I \) and \( II \) on \( N \) levels of information, two players on each level (\( 2N \) players).

On level \( i \) players see moves on levels \( j \leq i \) but can not see moves on levels \( j > i \)

- usually winning defined when there exists a winning strategy good for all counter-strategies
  but this contradicts information advantage, here strategies must be given level-by-level, i.e. \( I \) wins when
  - there exists a strategy for \( I \) on level 1
Hierarchical Muller Games

- Two coalitions $I$ and $II$ on $N$ levels of information, two players on each level ($2N$ players).

On level $i$ players see moves on levels $j \leq i$ but cannot see moves on levels $j > i$

- usually winning defined when there exists a winning strategy good for all counter-strategies
  but this contradicts information advantage, here strategies must be given level-by-level, i.e. $I$ wins when
    - there exists a strategy for $I$ on level 1
    - so that for all strategies of $II$ on level 1
Hierarchical Muller Games

- Two coalitions $I$ and $II$ on $N$ levels of information, two players on each level ($2^N$ players).

On level $i$ players see moves on levels $j \leq i$ but cannot see moves on levels $j > i$.

- Usually winning defined when there exists a winning strategy good for all counter-strategies but this contradicts information advantage, here strategies must be given level-by-level, i.e. $I$ wins when
  - there exists a strategy for $I$ on level 1
  - so that for all strategies of $II$ on level 1
  - there exists a strategy for $I$ on level 2
Hierarchical Muller Games

- Two coalitions I and II on N levels of information, two players on each level (2N players).

On level i players see moves on levels \( j \leq i \) but can not see moves on levels \( j > i \)

- usually winning defined when there exists a winning strategy good for all counter-strategies
  but this contradicts information advantage, here strategies must be given level-by-level, i.e. I wins when
  - there exists a strategy for I on level 1
  - so that for all strategies of II on level 1
  - there exists a strategy for I on level 2
  - so that for all strategies of II on level 2
Hierarchical Muller Games

- Two coalitions I and II on N levels of information, two players on each level (2N players).

On level i players see moves on levels j ≤ i but can not see moves on levels j > i

- usually winning defined when there exists a winning strategy good for all counter-strategies but this contradicts information advantage, here strategies must be given level-by-level, i.e. I wins when
  - there exists a strategy for I on level 1
  - so that for all strategies of II on level 1
  - there exists a strategy for I on level 2
  - so that for all strategies of II on level 2
  - ...

/Lukasz Kaiser (RWTH Aachen) Game Quantifier on Automatic Structures/
Hierarchical Muller Games

- Two coalitions $I$ and $II$ on $N$ levels of information, two players on each level ($2N$ players).

On level $i$ players see moves on levels $j \leq i$ but cannot see moves on levels $j > i$.

- usually winning defined when there exists a winning strategy good for all counter-strategies
  but this contradicts information advantage, here strategies must be given level-by-level, i.e. $I$ wins when
  - there exists a strategy for $I$ on level 1
  - so that for all strategies of $II$ on level 1
  - there exists a strategy for $I$ on level 2
  - so that for all strategies of $II$ on level 2
  - ... 
  - there exists a strategy for $I$ on level $N$
Hierarchical Muller Games

- Two coalitions I and II on N levels of information, two players on each level (2N players).

On level $i$ players see moves on levels $j \leq i$ but can not see moves on levels $j > i$

- usually winning defined when there exists a winning strategy good for all counter-strategies
but this contradicts information advantage, here strategies must be given level-by-level, i.e. I wins when
  - there exists a strategy for I on level 1
  - so that for all strategies of II on level 1
  - there exists a strategy for I on level 2
  - so that for all strategies of II on level 2
  - ... 
  - there exists a strategy for I on level $N$
  - so that for all strategies of II on level $N$ I wins the resulting play
Alternating hierarchical Muller games are model checking games for $\text{FO}+\exists^\omega$ on $\omega$-automatic structures.
Alternating hierarchical Muller games are model checking games for FO+$\bigcirc$ on $\omega$-automatic structures.

It can be checked that in the constructed model checking game players alternate their moves.
Alternating hierarchical Muller games are model checking games for $\text{FO}^+ \ominus$ on $\omega$-automatic structures.

It can be checked that in the constructed model checking game players alternate their moves.

**Theorem**

For any alternating hierarchical Muller game $G$ coalition $I$ wins $G$ starting from $\nu_0$ exactly if in $(\Sigma^\omega, W_I^G, \nu_0)$ holds

$$\bigcirc x_1 y_1 \ldots \bigcirc x_N y_N \ W_I^G, \nu_0(x_1, y_1, \ldots, x_N, y_N)$$

$W_I^G, \nu_0(\bar{x}) \equiv \text{plays respecting } \bar{x} \text{ starting from } \nu_0 \text{ are winning for } I$
Overview

Introduction

Logic with Game Quantifier

Hierarchical Model Checking Games

Conclusions and Future Work
Conclusions and Future Work (1)

FO+□ is a natural extension of first-order logic on structures on words

- preserves regularity
- more expressive than FO on weaker automatic structures
- invariant under inductive automorphisms
- what can be expressed using one, two, three □ quantifiers?
- when can formulas that use □ be written in FO?
- how does a relation defined in FO+□ depend on the presentation?
We defined a class of powerful Muller games with information levels
- when players alternate moves, games are determined and decidable
- expressive power equal to FO+□, non-elementary complexity
- long definition with alternation, can we do better?
- can these games be used or extended beyond automatic structures?
We defined a class of powerful Muller games with information levels

- when players alternate moves, games are **determined** and **decidable**
- expressive power equal to \( \text{FO}^{+\exists\forall} \), **non-elementary complexity**
- long definition with alternation, can we do better?
- can these games be used or extended **beyond automatic structures**?
  - these games capture automata determinisation in an abstract way
  - tree-automatic relations can be captured, something more?
  - are these still model checking games when the arena is infinite?
Thank You