1 2 3 Ehrenfeucht: Descriptive Games

Neil Immerman

University of Massachusetts, Amherst, USA

people.cs.umass.edu/~immerman
Descriptive Complexity

Input
\[ x_1 \ x_2 \ \cdots \ x_n \]

\[ \mapsto \]

Computation

\[ \mapsto \]

Output
\[ a_1 \ a_2 \ \cdots \ a_i \ \cdots \ a_m \]

Individual bits of the output are decision problems.

Computational Complexity: How hard is it to check if input has property \( S \)?

Descriptive Complexity: How rich a language do we need to describe property \( S \)?

Constructive Isomorphism between these two approaches.

Neil Immerman 1 2 3 Ehrenfeucht: Descriptive Games
Individual bits of the output are decision problems.
Descriptive Complexity

Input \( x_1 \ x_2 \ \cdots \ x_n \) \[\mapsto\] Computation \[\mapsto\] Output \( a_1 \ a_2 \ \cdots \ a_i \ \cdots \ a_m \)

Individual bits of the output are decision problems.

Computational Complexity:
How hard is it to check if input has property \( S \)?
Descriptive Complexity

Individual bits of the output are decision problems.

**Computational Complexity:**
How hard is it to check if input has property $S$?

**Descriptive Complexity:**
How rich a language do we need to describe property $S$?
Individual bits of the output are decision problems.

Computational Complexity:
How hard is it to check if input has property $S$?

Descriptive Complexity:
How rich a language do we need to describe property $S$?

Constructive Isomorphism between these two approaches.
Input is Finite Ordered Structure

\[ H = ( \{ a, b, c \}, \leq^H, E^H ) \]

Graph \[ E^H = \{ (a, b), (b, a), (b, c), (c, b), (c, a), (a, c) \} \]
Input is Finite Ordered Structure

\[ H = (\{a, b, c\}, \leq^H, E^H) \]

\[ \leq^H = \{(a, a), (a, b), (a, c), (b, b), (b, c), (c, c)\} \]

\[ E^H = \{(a, b), (b, a), (b, c), (c, b), (c, a), (a, c)\} \]
First-Order Logic

input symbols: \( E, R, Y, B, \ldots \)
variables: \( x, y, z, \ldots \)
boolean connectives: \( \land, \lor, \neg \)
quantifiers: \( \forall, \exists \)
numeric symbols: \( =, \leq, \text{min}, \text{max} \)

In this setting, with the structure of interest being the finite input, FO is a weak complexity class. It is easy to test if input, \( H \), satisfies \( \alpha | H = \alpha \)
First-Order Logic

input symbols: \( E, R, Y, B, \ldots \)
variables: \( x, y, z, \ldots \)
boolean connectives: \( \land, \lor, \neg \)
quantifiers: \( \forall, \exists \)
numeric symbols: \( =, \leq, min, max \)

In this setting, with the structure of interest being the **finite input**, \( FO \) is a **weak** complexity class.
First-Order Logic

input symbols: $E, R, Y, B, \ldots$
variables: $x, y, z, \ldots$
boolean connectives: $\land, \lor, \lnot$
quantifiers: $\forall, \exists$
numeric symbols: $=, \leq, \text{min}, \text{max}$

In this setting, with the structure of interest being the finite input, FO is a weak complexity class.

It is easy to test if input, $H$, satisfies $\alpha$ \quad ($H \models \alpha$).
First-Order Logic

$$H \quad a \leq b \leq c$$

$$G \quad 1 \leq 2 \leq 3$$

$$\alpha \equiv \forall x \exists y \ E(x, y)$$

$$\beta \equiv \forall xy (\neg E(x, x) \land (E(x, y) \rightarrow E(y, x)))$$

$$\gamma \equiv \forall x ((\forall y x \leq y) \rightarrow R(x))$$
First-Order Logic

\[ H \] \quad a \leq b \leq c

\[ G \] \quad 1 \leq 2 \leq 3

\[ H \models \alpha \land \beta \land \gamma \]

\[ \alpha \equiv \forall x \exists y \ E(x, y) \]

\[ \beta \equiv \forall xy (\neg E(x, x) \land (E(x, y) \rightarrow E(y, x))) \]

\[ \gamma \equiv \forall x ((\forall y x \leq y) \rightarrow R(x)) \]
First-Order Logic

\[ H \models a \leq b \leq c \]

\[ G \models 1 \leq 2 \leq 3 \]

\[ H \models \alpha \land \beta \land \gamma \]

\[ G \models \alpha \land \beta \land \neg \gamma \]

\[ \alpha \equiv \forall x \exists y \ E(x, y) \]

\[ \beta \equiv \forall xy (\neg E(x, x) \land (E(x, y) \rightarrow E(y, x))) \]

\[ \gamma \equiv \forall x ((\forall y x \leq y) \rightarrow R(x)) \]
First-Order Logic

\[ H \quad a \leq b \leq c \]
\[ G \quad 1 \leq 2 \leq 3 \]

\[ H \models \alpha \land \beta \land \gamma \]
\[ G \models \alpha \land \beta \land \neg \gamma \]

\[ \alpha \equiv \forall x \exists y \ E(x, y) \]
\[ \beta \equiv \forall xy (\neg E(x, x) \land (E(x, y) \rightarrow E(y, x))) \]
\[ \gamma \equiv \forall x ((\forall y x \leq y) \rightarrow R(x)) \]

\(\alpha\) and \(\beta\) are order independent; \(\gamma\) is order dependent
\[ \Phi_{3\text{color}} \equiv \exists R^1 \ G^1 \ B^1 \ \forall x \ y \ ((R(x) \lor G(x) \lor B(x)) \land (E(x,y) \rightarrow \\
(\neg (R(x) \land R(y)) \land \neg (G(x) \land G(y)) \land \neg (B(x) \land B(y)))))) \]
Fagin’s Theorem: $\text{NP} = \text{SO} \exists$

$$\Phi_{3\text{color}} \equiv \exists R^1 G^1 B^1 \forall x y ((R(x) \lor G(x) \lor B(x)) \land (E(x, y) \rightarrow \neg(R(x) \land R(y)) \land \neg(G(x) \land G(y)) \land \neg(B(x) \land B(y))))$$
Inductive Definition of Transitive Closure

\[ R^*(x, y) \equiv x = y \lor E(x, y) \lor \exists z (R^*(x, z) \land R^*(z, y)) \]
Inductive Definition of Transitive Closure

\[ R^*(x, y) \equiv x = y \lor E(x, y) \lor \exists z (R^*(x, z) \land R^*(z, y)) \]

\[ \varphi_{tc}(R, x, y) \equiv x = y \lor E(x, y) \lor \exists z (R(x, z) \land R(z, y)) \]
Inductive Definition of Transitive Closure

\[ R^*(x, y) \equiv x = y \lor E(x, y) \lor \exists z(R^*(x, z) \land R^*(z, y)) \]

\[ \varphi_{tc}(R, x, y) \equiv x = y \lor E(x, y) \lor \exists z(R(x, z) \land R(z, y)) \]

\[ \text{LFP}(\varphi_{tc}) = \varphi_{tc}^{[1+\log n]}(\emptyset) = R^* \]
Inductive Definition of Transitive Closure

\[ R^*(x, y) \equiv x = y \lor E(x, y) \lor \exists z (R^*(x, z) \land R^*(z, y)) \]

\[ \varphi_{tc}(R, x, y) \equiv x = y \lor E(x, y) \lor \exists z (R(x, z) \land R(z, y)) \]

\[ \text{LFP}(\varphi_{tc}) = \varphi_{tc}^{\left[1+\log n\right]}(\emptyset) = R^* \]

Next, we’ll sketch that every first-order relational operator such as \( \varphi_{tc} \) is equivalent to a block of restricted quantifiers. Thus the LFP is just the iteration of a quantifier block.
\[ \varphi_{tc}(R, x, y) \equiv x = y \lor E(x, y) \lor \exists z \ (R(x, z) \land R(z, y)) \]

1. Dummy universal quantification for base case:

\[
\varphi_{tc}(R, x, y) \equiv (\forall z. M_1)(\exists z)(R(x, z) \land R(z, y))
\]

\[ M_1 \equiv \neg(x = y \lor E(x, y)) \]
\( \varphi_{tc}(R, x, y) \equiv x = y \lor E(x, y) \lor \exists z (R(x, z) \land R(z, y)) \)

1. Dummy universal quantification for base case:

\[
\varphi_{tc}(R, x, y) \equiv (\forall z. M_1)(\exists z)(R(x, z) \land R(z, y)) \\
M_1 \equiv \neg(x = y \lor E(x, y))
\]

2. Using \( \forall \), replace two occurrences of \( R \) with one:

\[
\varphi_{tc}(R, x, y) \equiv (\forall z. M_1)(\exists z)(\forall uv. M_2)R(u, v) \\
M_2 \equiv (u = x \land v = z) \lor (u = z \land v = y)
\]
\( \varphi_{tc}(R, x, y) \equiv x = y \lor E(x, y) \lor \exists z \ (R(x, z) \land R(z, y)) \)

1. Dummy universal quantification for base case:

\[
\varphi_{tc}(R, x, y) \equiv (\forall z. M_1)(\exists z)(R(x, z) \land R(z, y)) \\
M_1 \equiv \neg(x = y \lor E(x, y))
\]

2. Using \( \forall \), replace two occurrences of \( R \) with one:

\[
\varphi_{tc}(R, x, y) \equiv (\forall z. M_1)(\exists z)(\forall uv. M_2)R(u, v) \\
M_2 \equiv (u = x \land v = z) \lor (u = z \land v = y)
\]

3. Requantify \( x \) and \( y \).

\[
M_3 \equiv (x = u \land y = v)
\]

\[
\varphi_{tc}(R, x, y) \equiv [ (\forall z. M_1)(\exists z)(\forall uv. M_2)(\exists xy. M_3) ] R(x, y)
\]

Every FO inductive definition is equivalent to a quantifier block.
\[
\text{CRAM}[t(n)] = \text{concurrent parallel random access machine; polynomial hardware, parallel time } O(t(n))
\]

\[
\text{IND}[t(n)] = \text{first-order, depth } t(n) \text{ inductive definitions}
\]

\[
\text{FO}[t(n)] = t(n) \text{ repetitions of a block of restricted quantifiers:}
\]

\[
\text{QB} = \left[ (Q_1 x_1. M_1) \cdots (Q_k x_k. M_k) \right]; \quad M_i \text{ quantifier-free}
\]

\[
\varphi_n = \underbrace{[\text{QB}][\text{QB}] \cdots [\text{QB}]}_{t(n)} M_0
\]
Thm. For all constructible, polynomially bounded \( t(n) \),

\[
\text{CRAM}[t(n)] = \text{IND}[t(n)] = \text{FO}[t(n)]
\]
parallel time = inductive depth = QB iteration

**Thm.** For all constructible, polynomially bounded $t(n)$,

$$\text{CRAM}[t(n)] = \text{IND}[t(n)] = \text{FO}[t(n)]$$

**Thm.** For all $t(n)$, even beyond polynomial,

$$\text{CRAM}[t(n)] = \text{FO}[t(n)]$$
Thm. For all constructible, polynomially bounded $t(n)$,

$$\text{CRAM}[t(n)] = \text{IND}[t(n)] = \text{FO}[t(n)]$$

Thm. For all $t(n)$, even beyond polynomial,

$$\text{CRAM}[t(n)] = \text{FO}[t(n)]$$

Thm. For all $t(n)$,

$$\text{CH}[t(n), 2^{n^{O(1)}}] = \text{SO}[t(n)]$$

CH[$t(n)$, $h(n)$] is parallel time $O(t(n))$ on a CRAM with $O(h(n))$ hardware gates.
Neil Immerman

1 2 3 Ehrenfeucht: Descriptive Games
Ehrenfeucht-Fraïssé Game

$G_m^c(G, H)$ $m$ moves, $c$ colors,

$\phi \equiv \exists rbg (E(r, b) \land E(b, g) \land E(g, r)) \quad G \models \phi$; $H \models \neg \phi$.
Ehrenfeucht-Fraïssé Game

$G_m^c(G, H) \ m \text{ moves, } c \text{ colors, } \textbf{Spoiler}: \text{ show difference}$

\[ \phi \equiv \exists rbg(E(r, b) \land E(b, g) \land E(g, r)) \quad G \models \phi; \quad H \models \neg \phi \]
Ehrenfeucht-Fraïssé Game

$G^c_m(G, H)$ $m$ moves, $c$ colors, **Spoiler**: show difference

**Duplicator**: preserve isomorphism of induced substructures

---

$\phi \equiv \exists rbg (E(r,b) \land E(b,g) \land E(g,r))$

$G \models \phi$; $H \models \neg \phi$
Ehrenfeucht-Fraïssé Game

$G_m^c(G, H)$ $m$ moves, $c$ colors,  
**Spoiler**: show difference

**Duplicator**: preserve isomorphism of induced substructures

For all $m$, $D$ wins $G_{2m}(G, H)$; but $S$ wins $G_{33}(G, H)$. 

$\phi \equiv \exists \text{rgb} (E(r, b) \land E(b, g) \land E(g, r)) \ G = \phi \land H = \neg \phi$
Ehrenfeucht-Fraïssé Game

$\mathcal{G}^c_m(G, H)$ $m$ moves, $c$ colors,  

**Spoiler**: show difference

**Duplicator**: preserve isomorphism of induced substructures
Ehrenfeucht-Fraïssé Game

$g^c_m(G, H)$ $m$ moves, $c$ colors,  **Spoiler**: show difference

**Duplicator**: preserve isomorphism of induced substructures

For all $m$, D wins $g^c_m(G, H)$; but S wins $g^c_3(G, H)$. 

$\phi \equiv \exists \text{rgb} \left( E(r, b) \land E(b, g) \land E(g, r) \right) G | = \phi; H | = \neg \phi$
Ehrenfeucht-Fraïssé Game

$G^c_m(G, H)$ $m$ moves, $c$ colors, **Spoiler**: show difference

**Duplicator**: preserve isomorphism of induced substructures

For all $m$, $D$ wins $G^2_m(G, H)$; but $S$ wins $G^3_3(G, H)$. 

$\phi \equiv \exists rbg(E(r, b) \land E(b, g) \land E(g, r)) \models G \wedge \neg \models H$.
Ehrenfeucht-Fraïssé Game

$G_m^c(G, H)$ $m$ moves, $c$ colors, **Spoiler**: show difference

**Duplicator**: preserve isomorphism of induced substructures
Ehrenfeucht-Fraïssé Game

$G^c_m(G, H)$ $m$ moves, $c$ colors, Spoiler: show difference

Duplicator: preserve isomorphism of induced substructures
Ehrenfeucht-Fraïssé Game

$G^c_m(G, H)$ $m$ moves, $c$ colors,  

**Spoiler:** show difference

**Duplicator:** preserve isomorphism of induced substructures

For all $m$, $D$ wins $G^2_m(G, H)$;

1. $G$
2. $H$

Neil Immerman  1 2 3 Ehrenfeucht: Descriptive Games
Ehrenfeucht-Fraïssé Game

$G^c_m(G, H)$ $m$ moves, $c$ colors,  Spoiler: show difference

Duplicator: preserve isomorphism of induced substructures

For all $m$, D wins $G^2_m(G, H)$;
Ehrenfeucht-Fraïssé Game

$G^c_m(G, H)$ $m$ moves, $c$ colors, \textbf{Spoiler}: show difference

\textbf{Duplicator}: preserve isomorphism of induced substructures

For all $m$, D wins $G^2_m(G, H)$;

\[ \varphi \equiv \exists \text{rgb}(E(r, b) \land E(b, g) \land E(g, r)) \]

\[ G \models \varphi \]

\[ H \models \neg \varphi \]
Ehrenfeucht-Fraïssé Game

\[ \mathcal{G}^c_m(G, H) \] \( m \) moves, \( c \) colors,  \textbf{Spoiler}: show difference

\textbf{Duplicator}: preserve isomorphism of induced substructures

For all \( m \), \( D \) wins \( \mathcal{G}^2_m(G, H) \);

\[ \phi \equiv \exists r b g (E(r, b) \land E(b, g) \land E(g, r)) \]

\( G \) \( H \)
Ehrenfeucht-Fraïssé Game

$G^c_m(G, H)$ $m$ moves, $c$ colors,  **Spoiler:** show difference

**Duplicator:** preserve isomorphism of induced substructures

For all $m$, $D$ wins $G^2_m(G, H)$; but $S$ wins $G^3_3(G, H)$.

$\varphi \equiv \exists rbg(E(r, b) \land E(b, g) \land E(g, r)) \quad G \models \varphi; \quad H \models \neg \varphi$
**Notation:** \( G \sim^c_m H \) means that **Duplicator** has a winning strategy for \( G^c_m(G, H) \).
Fundamental Thm of Ehrenfeucht-Fraïssé Games

Notation: \( G \sim^c_m H \) means that Duplicator has a winning strategy for \( G_m^c(G, H) \).

Thm. D has a winning strategy on the \( m \)-move, \( c \)-color game on \( G, H \) iff \( G \) and \( H \) agree on all formulas using \( c \) variables and quantifier depth \( m \),

\[
G \sim^c_m H \iff G \equiv^c_m H
\]
Notation: \( G \sim_{m}^{c} H \) means that Duplicator has a winning strategy for \( G_{m}^{c}(G, H) \).

**Thm.** D has a winning strategy on the \( m \)-move, \( c \)-color game on \( G, H \) iff \( G \) and \( H \) agree on all formulas using \( c \) variables and quantifier depth \( m \),

\[
G \sim_{m}^{c} H \iff G \equiv_{m}^{c} H
\]

Ehrenfeucht-Fraïssé games are **fantastically useful** for determining what is expressible in FO logic in a given quantifier depth and with a given number of variables.
Notation: \( G \sim_{c}^{m} H \) means that Duplicator has a winning strategy for \( G_{c}^{m}(G, H) \).

Thm. \( D \) has a winning strategy on the \( m \)-move, \( c \)-color game on \( G, H \) iff \( G \) and \( H \) agree on all formulas using \( c \) variables and quantifier depth \( m \),

\[
G \sim_{c}^{m} H \iff G \equiv_{c}^{m} H
\]

Ehrenfeucht-Fraïssé games are fantastically useful for determining what is expressible in FO logic in a given quantifier depth and with a given number of variables.

But, as we will see next, Ehrenfeucht-Fraïssé games are not very helpful for proving Descriptive Lower Bounds.
Thm. $L^3_{2+\log n}$ suffices to characterize any property whatsoever over ordered graphs.
Thm. $L^{3}_{[2+\log n]}$ suffices to characterize any property whatsoever over ordered graphs.

Proof: We can name any vertex by number in $L^{3}_{[1+\log n]}$. 
Thm. $\mathcal{L}^3_{2+\log n}$ suffices to characterize any property whatsoever over ordered graphs.

Proof: We can name any vertex by number in $\mathcal{L}^3_{1+\log n}$. We can identify a graph in $\mathcal{L}^3_{2+\log n}$ by asserting for each $i, j \leq n$, whether $E(v_i, v_j)$. 

Ehrenfeucht: Descriptive Games
Thm. \( L^{3}_{[2+\log n]} \) suffices to characterize any property whatsoever over ordered graphs.

Proof: We can name any vertex by number in \( L^{3}_{[1+\log n]} \).

We can identify a graph in \( L^{3}_{[2+\log n]} \) by asserting for each \( i, j \leq n \), whether \( E(v_i, v_j) \).

In \( L^{3}_{[2+\log n]} \), we can identify an arbitrary set of graphs on \( n \) vertices.
Number of Quantifiers game:

- Separation Game: [I81]

[FLVR21]: LICS21, determined exact number of quantifiers to identify a linear order of length $n$.

[CFIKLS23] – next talk by Rik Sengupta

Personal history of my 1980 Ph.D. thesis:

- 1978: Larry Carter sends me via snail mail a hard copy of R. Fagin, "Generalized First-Order Spectra and Polynomial-Time Recognizable Sets."
Number of Quantifiers game:
  ▶ Separation Game: [I81]
  ▶ renamed Multistructural Game: [FLRV21]: LICS21, determined exact number of quantifiers to identify a linear order of length $n$. 
Number of Quantifiers game:

- Separation Game: [I81]
- renamed Multistructural Game: [FLRV21]: LICS21, determined exact number of quantifiers to identify a linear order of length \( n \).
- [FLVW22]
Number of Quantifiers game:
- Separation Game: [I81]
- renamed Multistructural Game: [FLRV21]: LICS21, determined exact number of quantifiers to identify a linear order of length $n$.
- [FLVW22]
- [CFIKLS23] – next talk by Rik Sengupta
Number of Quantifiers game:

- Separation Game: [I81]
- renamed Multistructural Game: [FLRV21]: LICS21, determined exact number of quantifiers to identify a linear order of length $n$.
- [FLVW22]
- [CFIKLS23] – next talk by Rik Sengupta

Personal history of my 1980 Ph.D. thesis:

- 1978: Larry Carter sends me via snail mail a hard copy of R. Fagin, "Generalized First-Order Spectra and Polynomial-Time Recognizable Sets.
Number of Quantifiers game:
- Separation Game: [I81]
- renamed Multistructural Game: [FLRV21]: LICS21, determined exact number of quantifiers to identify a linear order of length \( n \).
- [FLVW22]
- [CFIKLS23] – next talk by Rik Sengupta

Personal history of my 1980 Ph.D. thesis:
- 1978: Larry Carter sends me via snail mail a hard copy of R. Fagin, “Generalized First-Order Spectra and Polynomial-Time Recognizable Sets.”
Number of Quantifiers game:

- Separation Game: [I81]
- renamed Multistructural Game: [FLRV21]: LICS21, determined exact number of quantifiers to identify a linear order of length $n$.
- [FLVW22]
- [CFIKLS23] – next talk by Rik Sengupta

Personal history of my 1980 Ph.D. thesis:

- 1978: Larry Carter sends me via snail mail a hard copy of R. Fagin, “Generalized First-Order Spectra and Polynomial-Time Recognizable Sets.”
Number of Quantifiers game:
- Separation Game: [I81]
- renamed Multistructural Game: [FLRV21]: LICS21, determined exact number of quantifiers to identify a linear order of length $n$.
- [FLVW22]
- [CFIKLS23] – next talk by Rik Sengupta

Personal history of my 1980 Ph.D. thesis:
- 1978: Larry Carter sends me via snail mail a hard copy of R. Fagin, “Generalized First-Order Spectra and Polynomial-Time Recognizable Sets.”
Number of Quantifiers game = Multistructural Game

\[ MS_m(A, B); \ m \text{ moves played on a pair of sets of structures.} \]
Number of Quantifiers game = Multistuctural Game

\[ MS_m(\mathcal{A}, \mathcal{B}); \ m \text{ moves played on a pair of sets of structures.} \]

Spoiler chooses an element of each structure on one side.

Duplicator makes multiple copies of each structure on the other side and then chooses a corresponding element of each structure. Duplicator wins if after each move there is a pair of isomorphic induced substructures, one from each side.

Thm. Spoiler wins \[ MS_m(\mathcal{A}, \mathcal{B}) \] iff there is a formula \[ \varphi \] having at most \[ m \] quantifiers, \[ \mathcal{A}|= \varphi; \mathcal{B}|= \neg \varphi. \]

Cor. Property \( S \) is expressible with \( m(n) \) quantifiers, for inputs of size \( n \) iff Spoiler wins \( MS_m(\mathcal{S}_n, \mathcal{S}_n^c) \) where \( \mathcal{S}_n \) is the set of all ordered structures of size \( n \) satisfying \( S \) and \( \mathcal{S}_n^c \) is the set of all ordered structures of size \( n \) not satisfying \( S \).
$MS_m(A, B);$ $m$ moves played on a pair of sets of structures.

Spoiler chooses an element of each structure on one side.

Duplicator makes multiple copies of each structure on the other side and then chooses a corresponding element of each structure.
$MS_m(A, B)$; $m$ moves played on a pair of sets of structures.

Spoiler chooses an element of each structure on one side.

Duplicator makes multiple copies of each structure on the other side and then chooses a corresponding element of each structure.

Duplicator wins if after each move there is a pair of isomorphic induced substructures, one from each side.
Number of Quantifiers game = Multistructural Game

\[ MS_m(\mathcal{A}, \mathcal{B}); \] \( m \) moves played on a pair of sets of structures.

Spoiler chooses an element of each structure on one side.

Duplicator makes multiple copies of each structure on the other side and then chooses a corresponding element of each structure.

Duplicator wins if after each move there is a pair of isomorphic induced substructures, one from each side.

**Thm.** Spoiler wins \( MS_m(\mathcal{A}, \mathcal{B}) \) iff there is a formula \( \varphi \) having at most \( m \) quantifiers, \( \mathcal{A} \models \varphi; \ \mathcal{B} \models \neg \varphi. \)
$MS_m(\mathcal{A}, \mathcal{B})$; $m$ moves played on a pair of sets of structures.

Spoiler chooses an element of each structure on one side.

Duplicator makes multiple copies of each structure on the other side and then chooses a corresponding element of each structure.

Duplicator wins if after each move there is a a pair of isomorphic induced substructures, one from each side.

**Thm.** Spoiler wins $MS_m(\mathcal{A}, \mathcal{B})$ iff there is a formula $\varphi$ having at most $m$ quantifiers, $\mathcal{A} \models \varphi$; $\mathcal{B} \models \neg \varphi$.

**Cor.** Property $S$ is expressible with $m(n)$ quantifiers, for inputs of size $n$ iff Spoiler wins $MS_m(S_n, \overline{S}_n)$ where $S_n$ is the set of all ordered structures of size $n$ satisfying $S$ and $\overline{S}_n$ is the set of all ordered structures of size $n$ not satisfying $S$. 
Examples: $MS_2(\mathcal{A}, \mathcal{B})$ and $MS_3(\mathcal{A}, \mathcal{B})$ Games

\[ \mathcal{A} = \{L_3\} \quad \mathcal{B} = \{L_2\} \]
Examples: $MS_2(A, B)$ and $MS_3(A, B)$ Games

$A = \{L_3\}$ \hspace{1cm} $B = \{L_2\}$
Examples: $MS_2(\mathcal{A}, \mathcal{B})$ and $MS_3(\mathcal{A}, \mathcal{B})$ Games

\[ \mathcal{A} = \{L_3\} \quad \mathcal{B} = \{L_2\} \]
Examples: $MS_2(A, B)$ and $MS_3(A, B)$ Games

$A = \{L_3\}$ \hspace{2cm} $B = \{L_2\}$

$A_1 = \{(L_3, 2)\}$ \hspace{2cm} $B_1 = \{(L_2, 4), (L'_2, 7)\}$
Examples: $MS_2(\mathcal{A}, \mathcal{B})$ and $MS_3(\mathcal{A}, \mathcal{B})$ Games

$\mathcal{A} = \{L_3\}$  \hspace{1cm} $\mathcal{B} = \{L_2\}$

$\mathcal{A}_1 = \{(L_3, 2)\}$  \hspace{1cm} $\mathcal{B}_1 = \{(L_2, 4), (L'_2, 7)\}$
Examples: $MS_2(\mathcal{A}, \mathcal{B})$ and $MS_3(\mathcal{A}, \mathcal{B})$ Games

Duplicator wins $MS_2(\mathcal{A}, \mathcal{B})$

\[
\mathcal{A} = \{L_3\} \quad \mathcal{B} = \{L_2\}
\]
\[
\mathcal{A}_1 = \{(L_3, 2)\} \quad \mathcal{B}_1 = \{(L_2, 4), (L'_2, 7)\}
\]
\[
\mathcal{A}_2 = \{(L_3, 2, 1)\} \quad \mathcal{B}_2 = \{(L_2, 4, 5), (L'_2, 7, 6)\}
\]
Examples: $MS_2(\mathcal{A}, \mathcal{B})$ and $MS_3(\mathcal{A}, \mathcal{B})$ Games

Duplicator wins $MS_2(\mathcal{A}, \mathcal{B})$ Spoiler wins $MS_3(\mathcal{A}, \mathcal{B})$

\[ \mathcal{A} = \{L_3\} \quad \mathcal{B} = \{L_2\} \]
\[ \mathcal{A}_1 = \{(L_3, 2)\} \quad \mathcal{B}_1 = \{(L_2, 4), (L_2', 7)\} \]
\[ \mathcal{A}_2 = \{(L_3, 2, 1)\} \quad \mathcal{B}_2 = \{(L_2, 4, 5), (L_2', 7, 6)\} \]
Examples: $MS_2(\mathcal{A}, \mathcal{B})$ and $MS_3(\mathcal{A}, \mathcal{B})$ Games

Duplicator wins $MS_2(\mathcal{A}, \mathcal{B})$ Spoiler wins $MS_3(\mathcal{A}, \mathcal{B})$

$\mathcal{A} = \{L_3\} \quad \mathcal{B} = \{L_2\}$

$\mathcal{A}_1 = \{(L_3, 2)\} \quad \mathcal{B}_1 = \{(L_2, 4), (L'_2, 7)\}$

$\mathcal{A}_2 = \{(L_3, 2, 1)\} \quad \mathcal{B}_2 = \{(L_2, 4, 5), (L'_2, 7, 6)\}$

$\varphi \equiv \exists \text{rgb}(E(b, r) \land E(r, g)) \quad \mathcal{A} \models \varphi \quad \mathcal{B} \models \neg \varphi$
Examples: $MS_2(A, B)$ and $MS_3(A, B)$ Games

Duplicator wins $MS_2(A, B)$ Spoiler wins $MS_3(A, B)$

Spoiler wins $G_2^2(L_3, L_2)$

\[
A = \{L_3\} \quad B = \{L_2\}
\]
\[
A_1 = \{(L_3, 2)\} \quad B_1 = \{(L_2, 4), (L'_2, 7)\}
\]
\[
A_2 = \{(L_3, 2, 1)\} \quad B_2 = \{(L_2, 4, 5), (L'_2, 7, 6)\}
\]

\[
\varphi \equiv \exists \text{rg}(E(b, r) \land E(r, g)) \quad A \models \varphi \quad B \models \neg \varphi
\]

\[
\psi \equiv \exists r(\exists b(E(b, r)) \land \exists bE(r, b)) \quad A \models \varphi \quad B \models \neg \varphi
\]
$QVT^c_m(\mathcal{A}, \mathcal{B})$  Spoiler builds formula tree separating $\mathcal{A}, \mathcal{B}$. 

$\mathcal{A} \square \mathcal{B}$
$QVT^c_m(\mathcal{A}, \mathcal{B})$  Spoiler builds formula tree separating $\mathcal{A}$, $\mathcal{B}$. 

\[
\begin{array}{c}
\mathcal{A} \\
\exists r \\
\mathcal{B} \\
\mathcal{A}_1 \\
\mathcal{B}_1
\end{array}
\]
$QVT_m^c(A, B)$  Spoiler builds formula tree separating $A$, $B$. 

\[
\begin{align*}
A & \quad \exists r \quad B \\
\lor & \\
A_1 & \lor B_1 \\
A_{10} & \quad B_1 \\
A_{11} & \quad B_1
\end{align*}
\]
$QVT^c_m(A, B)$ Spoiler builds formula tree separating $A$, $B$. 

\[
\begin{align*}
A & \exists r B \\
A_1 & \lor B_1 \\
A_{10} & \neg B_1 \\
B_1 & A_{10} \\
A_{11} & B_1
\end{align*}
\]
$QVT_m^c(A, B)$  Spoiler builds formula tree separating $A, B$.  

$A \exists r B$

$A_1 \lor B_1$

$A_{10} \neg B_1$

$B_1 A_{11} \alpha B_1$

Neil Immerman  1 2 3 Ehrenfeucht: Descriptive Games
\[ QVT^c_m(A, B) \] Spoiler builds formula tree separating \( A, B \).

\[
\begin{align*}
& A \quad \exists r \quad B \\
& A_1 \quad \lor \quad B_1 \\
& A_{10} \quad \neg \quad B_1 \\
& B_1 \quad \neg \quad A_{10} \\
& A_{11} \quad \alpha \quad B_1
\end{align*}
\]

**Thm.** Spoiler can close the \( QVT^c_m(A, B) \) game tree using \( c \) colors and \( m \) quantifier moves iff there is a formula with \( c \) variables and \( m \) quantifiers separating \( A \) from \( B \).
$QVT^2(L_5, L_4)$

Diagram: 

- $a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow a_5$ 
- $b_1 \rightarrow b_2 \rightarrow b_3 \rightarrow b_4$
$QVT^2(L_5, L_4)$
$QVT^2(L_5, L_4)$

\[
\begin{array}{c}
\begin{array}{c}
\exists r \\
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow a_5 \\
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
\exists r \\
\end{array}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
\exists r \\
\end{array}
\end{array}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
\exists r \\
\end{array}
\end{array}
\end{array}
\end{array}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
b_1 \rightarrow b_2 \rightarrow b_3 \rightarrow b_4 \\
\end{array}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
b_1 \rightarrow b_2 \rightarrow b_3 \rightarrow b_4 \\
\end{array}
\end{array}
\end{array}
\begin{array}{c}
\begin{array}{c}
\begin{array}{c}
b_1 \rightarrow b_2 \rightarrow b_3 \rightarrow b_4 \\
\end{array}
\end{array}
\end{array}
\end{array}
\end{array}
\end{array}
\end{array}
\end{array}
\end{array}

$QVT^2(L_5, L_4)$
$QVT^2(L_5, L_4)$

*Neil Immerman*  
1 2 3 Ehrenfeucht: Descriptive Games
$QVT^2(L_5, L_4)$
$QVT^2(L_5, L_4)$
$QVT^2(L_5, L_4)$

\[
\begin{align*}
\exists r & \quad a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow a_5 \\
\exists b & \quad b_1 \rightarrow b_2 \rightarrow b_3 \rightarrow b_4
\end{align*}
\]

\[
\begin{align*}
\exists r & \quad a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow a_5 \\
\exists b & \quad b_1 \rightarrow b_2 \rightarrow b_3 \rightarrow b_4
\end{align*}
\]

\[
\begin{align*}
\exists r & \quad a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow a_5 \\
\exists b & \quad b_1 \rightarrow b_2 \rightarrow b_3 \rightarrow b_4
\end{align*}
\]

\[
\begin{align*}
\exists r & \quad a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow a_5 \\
\exists b & \quad b_1 \rightarrow b_2 \rightarrow b_3 \rightarrow b_4
\end{align*}
\]

\[
\begin{align*}
\exists r & \quad a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow a_5 \\
\exists b & \quad b_1 \rightarrow b_2 \rightarrow b_3 \rightarrow b_4
\end{align*}
\]
QVT^2(L_5, L_4)
$QVT^2(L_5, L_4)$
$QVT^2(L_5, L_4)$
$QVT^2(L_5, L_4)$

Spoiler wins $QVT^2_5(L_5, L_4)$; Can he do better?
Spoiler wins $QVT_5^2(L_5, L_4)$; Duplicator wins $QVT_4^2(L_5, L_4)$.

\[
\begin{array}{c}
\begin{array}{c}
\makebox[2cm][c]{a_1} \rightarrow \makebox[2cm][c]{a_2} \rightarrow \makebox[2cm][c]{a_3} \rightarrow \makebox[2cm][c]{a_4} \rightarrow \makebox[2cm][c]{a_5} \quad \Box \quad \begin{array}{c}
\makebox[2cm][c]{b_1} \rightarrow \makebox[2cm][c]{b_2} \rightarrow \makebox[2cm][c]{b_3} \rightarrow \makebox[2cm][c]{b_4}
\end{array}
\end{array}
\end{array}
\]
Spoiler wins $QVT^2_5(L_5, L_4)$; Duplicator wins $QVT^2_4(L_5, L_4)$.
Spoiler wins $QVT_5^2(L_5, L_4)$; Duplicator wins $QVT_4^2(L_5, L_4)$.
Spoiler wins $QVT^2_5(L_5, L_4)$; Duplicator wins $QVT^2_4(L_5, L_4)$. 

Neil Immerman 1 2 3 Ehrenfeucht: Descriptive Games
Spoiler wins $QVT^2_5(L_5, L_4)$; Duplicator wins $QVT^2_4(L_5, L_4)$. 
Arithmetic Hierarchy

\( \text{FO}(N) \) r.e. complete
Halt
co-r.e. complete
\( \text{FO} \)-SAT
\( \text{FO} \)-VALID

Recursive

\( \text{Succinct QSAT} \) EXPSPACE complete
\( \text{SO}(\text{PFP}) \) \( \text{SO}[2^{2n^{O(1)}}] \) EXPSPACE

\( \text{Succinct Horn SAT} \) EXPTIME complete
\( \text{SO}(\text{LFP}) \) \( \text{SO}[2^{n^{O(1)}}] \) EXPSPACE

\( \text{QSAT} \) PSPACE complete
\( \text{FO}[2^{n^{O(1)}}] \) \( \text{FO}(\text{PFP}) \)
\( \text{SO}(\text{TC}) \) \( \text{SO}[n^{O(1)}] \) PSPACE

PTIME Hierarchy

\( \text{PTIME} \) \( \text{NP} \cap \text{co-NP} \)

\( \text{NP} \) co-NP complete
\( \text{SAT} \)

\( \text{FO}[n^{O(1)}] \)
\( \text{FO}(\text{LFP}) \) \( \text{SO}(\text{Horn}) \)

\( \text{FO}[(\log n)^{O(1)}] \) \( \text{FO}[(\log n)^{O(1)}] \) “truly feasible” \( \text{AC}^1 \)

\( \text{FO}(\text{CFL}) \)
\( \text{FO}(\text{TC}) \) \( \text{SO}(\text{Krom}) \)
\( \text{2SAT} \) NL comp.
\( \text{2COLOR} \) L comp.

\( \text{FO}(\text{REGULAR}) \)
\( \text{FO}(\text{COUNT}) \)

\( \text{FO} \) \( \text{LOGTIME} \) Hierarchy

\( \text{AC}^0 \)
Let’s learn to play these games better, especially the QVT game: for fun and improving our knowledge.
Let’s learn to play these games better, especially the QVT game: for fun and improving our knowledge.

Let’s look again at some great previous lower bounds including the following, among others, and try hard to reprove them and extend them using the QVT game:

- Grohe Schweikardt: Succinctness [GS05]
- Rossman: Tight Variable Hierarchy [R08]
- Hella Väänänen: Formula Size [HV15]
Let’s learn to play these games better, especially the $QVT$ game: for fun and improving our knowledge.

Let’s look again at some great previous lower bounds including the following, among others, and try hard to reprove them and extend them using the $QVT$ game:

- Grohe Schweikardt: Succinctness [GS05]

I’ll be here the whole weekend; come say, “Hello”; let’s talk about these and related issues.

Thank you!
Let’s learn to play these games better, especially the \textit{QVT} game: for fun and improving our knowledge.

Let’s look again at some great previous lower bounds including the following, among others, and try hard to reprove them and extend them using the \textit{QVT} game:

- Grohe Schweikardt: Succinctness [GS05]
- Rossman: Tight Variable Hierarchy [R08]
Conclusions and Future Directions

- Let’s learn to play these games better, especially the $QVT$ game: for fun and improving our knowledge.
- Let’s look again at some great previous lower bounds including the following, among others, and try hard to reprove them and extend them using the $QVT$ game:
  - Grohe Schweikardt: Succinctness [GS05]
  - Rossman: Tight Variable Hierarchy [R08]
  - Hella Väänänen: Formula Size [HV15]

I’ll be here the whole weekend; come say, “Hello”; let’s talk about these and related issues.

Thank you!
Let’s learn to play these games better, especially the QVT game: for fun and improving our knowledge.

Let’s look again at some great previous lower bounds including the following, among others, and try hard to reprove them and extend them using the QVT game:

- Grohe Schweikardt: Succinctness [GS05]
- Rossman: Tight Variable Hierarchy [R08]
- Hella Väänänen: Formula Size [HV15]

I’ll be here the whole weekend; come say, “Hello”; let’s talk about these and related issues.
Let’s learn to play these games better, especially the QVT game: for fun and improving our knowledge.

Let’s look again at some great previous lower bounds including the following, among others, and try hard to reprove them and extend them using the QVT game:

- Grohe Schweikardt: Succinctness [GS05]
- Rossman: Tight Variable Hierarchy [R08]
- Hella Väänänen: Formula Size [HV15]

I’ll be here the whole weekend; come say, “Hello”; let’s talk about these and related issues.

Thank you!