Australian National University

Finite Model Theory

Qing Wang
qing.wang@anu.edu.au

Logic and Computation
Research School of Computer Science

Overview of Topics

- **Part 2**: – **Ehrenfeucht-Fraïssé Games**
  1. Elementary equivalence and isomorphism
  2. Ehrenfeucht-Fraïssé (EF) games
     - Rules
     - Winning strategies
  3. Partial isomorphism
  4. Equivalence relation
  5. EF theorem
  6. EF applications

Recall - Inexpressibility Proofs

- *How can one prove that a property P is inexpressible in a logic L on a class C of structures?*
  - To prove that P is expressible, one needs to find a formula of L that defines P on C.
  - To prove that P is not expressible, one has to show no formula of L that defines P on C.
- Common techniques used for inexpressibility proofs in first-order logic:
  - **Compactness theorem**
    $\hookrightarrow$ fails over finite structures.
  - **Ehrenfeucht-Fraïssé games**
    $\hookrightarrow$ used as a central tool on classes of finite structures.

Elementary Equivalence and Isomorphism

- Elementary equivalence, formulated by Alfred Tarski, is an important model-theoretic notion.
- Two models $\mathfrak{A}$ and $\mathfrak{B}$ over the same vocabulary are **elementarily equivalent** if, for every first-order sentence φ, $\mathfrak{B} \models \varphi$ iff $\mathfrak{A} \models \varphi$.

  That is, if two models are elementarily equivalent, then they cannot be distinguished by any first-order sentence.
- The notion of elementary equivalence is important to establishing inexpressibility results.
  - First, prove that two models are elementarily equivalent.
  - Then, show that a property P that can distinguish the two models.
  - Thus, the property P is not definable.

Elementary Equivalence and Isomorphism

- Two models $\mathfrak{A}$ and $\mathfrak{B}$ over the same vocabulary are **isomorphic** if there is a bijective mapping $h : A \to B$ preserving relations and constants.
- In general, two isomorphic models must be elementarily equivalent, but two elementarily equivalent models are not necessarily isomorphic.

Elementary Equivalence and Isomorphism

- Two models $\mathfrak{A}$ and $\mathfrak{B}$ over the same vocabulary are **isomorphic** if there is a bijective mapping $h : A \to B$ preserving relations and constants.
- In general, two isomorphic models must be elementarily equivalent, but two elementarily equivalent models are not necessarily isomorphic.
- In the case of finite structures, elementary equivalence is however uninteresting. Finite structures can be characterized up to isomorphism by **single FO sentence**.

elementary equivalence $\Leftrightarrow$ isomorphism

Elementary Equivalence and Isomorphism

- **Theorem**

  For every finite structure $\mathfrak{A}$, there is a first-order sentence φ such that $\mathfrak{B} \models \varphi$ iff an arbitrary structure $\mathfrak{B}$ is isomorphic to $\mathfrak{A}$.

  Proof

  - Assume w.l.o.g. that $\mathfrak{A}$ is a graph (V, E) where $V = \{a_1, \ldots, a_n\}$.
  - Define φ as

$$\begin{aligned}\exists x_1 \ldots \exists x_n (& (\bigwedge_{i \neq j} \neg(x_i = x_j)) \\ & \wedge (\forall y \bigvee_i y = x_i) \\ & \wedge (\bigwedge_{(a_i, a_j) \in E} E(x_i, x_j)) \\ & \wedge (\bigwedge_{(a_i, a_j) \notin E} \neg E(x_i, x_j)))\end{aligned}$$

  - We have $\mathfrak{A} \models \varphi$. If $\mathfrak{B} \models \varphi$, then $\mathfrak{B}$ is isomorphic to $\mathfrak{A}$.

Methodology for Inexpressibility Proofs

- Thus, for finite structures, the notion of elementary equivalence is **too strong** to establishing inexpressibility results.
- One way to solve this is to weaken the relation of elementary equivalence by **stratifying formulas in a logic**.

Methodology for Inexpressibility Proofs

- To prove that a property P is not expressible in a logic L over finite structures, we can do the following:
  - Partition the set of all formulas of L into countably many classes, i.e., $L[0]$, $L[1]$,..., $L[k]$,...;
  - Find two families of structures $\{\mathfrak{A}_k | k \in \mathbb{N}\}$ and $\{\mathfrak{B}_k | k \in \mathbb{N}\}$ such that
    1. $\mathfrak{A}_k \models \varphi$ iff $\mathfrak{B}_k \models \varphi$ for every sentence φ in $L[k]$; and
    2. $\mathfrak{A}_k$ has property P, but $\mathfrak{B}_k$ does not.

Methodology for Inexpressibility Proofs

- To prove that a property P is not expressible in a logic L over finite structures, we can do the following:
  - Partition the set of all formulas of L into countably many classes, i.e., $L[0]$, $L[1]$,..., $L[k]$,...;
  - Find two families of structures $\{\mathfrak{A}_k | k \in \mathbb{N}\}$ and $\{\mathfrak{B}_k | k \in \mathbb{N}\}$ such that
    1. $\mathfrak{A}_k \models \varphi$ iff $\mathfrak{B}_k \models \varphi$ for every sentence φ in $L[k]$; and
    2. $\mathfrak{A}_k$ has property P, but $\mathfrak{B}_k$ does not.
- But...
  - How to partition FO into such classes?
  - How to show that two families of structures agree on classes of FO?

Methodology for Inexpressibility Proofs

- To prove that a property P is not expressible in a logic L over finite structures, we can do the following:
  - Partition the set of all formulas of L into countably many classes, i.e., $L[0]$, $L[1]$,..., $L[k]$,...;
  - Find two families of structures $\{\mathfrak{A}_k | k \in \mathbb{N}\}$ and $\{\mathfrak{B}_k | k \in \mathbb{N}\}$ such that
    1. $\mathfrak{A}_k \models \varphi$ iff $\mathfrak{B}_k \models \varphi$ for every sentence φ in $L[k]$; and
    2. $\mathfrak{A}_k$ has property P, but $\mathfrak{B}_k$ does not.

- But...
  - How to partition FO into such classes?
    $\hookrightarrow$ Quantifier rank
  - How to show that two families of structures agree on classes of FO?
    $\hookrightarrow$ Partial isomorphism

Quantifier Rank

- The **quantifier rank** of a formula φ, written as $qr(\varphi)$, is its depth of quantifier nesting, i.e.,
  - If φ is atomic, then $qr(\varphi) = 0$.
  - $qr(\varphi_1 \wedge \varphi_2) = qr(\varphi_1 \vee \varphi_2) = max(qr(\varphi_1), qr(\varphi_2))$.
  - $qr(\neg\varphi) = qr(\varphi)$.
  - $qr(\exists x \varphi) = qr(\forall x \varphi) = qr(\varphi) + 1$.
- **Example**: What is the quantifier rank of d_k? What is the total number of quantifiers in d_k?
  - $d_0(x, y) = E(x, y)$
  - ...
  - $d_k = \exists z d_{k-1}(x, z) \wedge d_{k-1}(z, y)$
- The set of all FO-formulas is partitioned into many classes, denoted as $FO[k]$, each having all formulas of quantifier rank up to k.

Equivalence Relation

- We write $\mathfrak{A} \equiv_k \mathfrak{B}$ for two structures $\mathfrak{A}$ and $\mathfrak{B}$ iff the following equivalence holds for all sentences $\varphi \in FO[k]$:

$$\mathfrak{A} \models \varphi \Leftrightarrow \mathfrak{B} \models \varphi,$$

  i.e., $\mathfrak{A}$ and $\mathfrak{B}$ cannot be distinguished by FO sentences with $qr(\varphi) < k$.
- Let $\bar{a}$ and $\bar{b}$ be two tuples from $\mathfrak{A}$ and $\mathfrak{B}$, respectively. We write $(\mathfrak{A}, \bar{a}) \equiv_k (\mathfrak{B}, \bar{b})$ iff the following equivalence holds for all formulas $\varphi \in FO[k]$, where

$$\mathfrak{A} \models \varphi[\bar{a}] \Leftrightarrow \mathfrak{B} \models \varphi[\bar{b}]$$

- Note that,
  - $\mathfrak{A} \equiv_k \mathfrak{B}$ is a weakening of elementary equivalence by only considering the class of FO sentences/formulas of quantifier rank up to k.
  - $\equiv_k$ has finitely many equivalence classes, each of which is FO-definable.

Partial Isomorphism

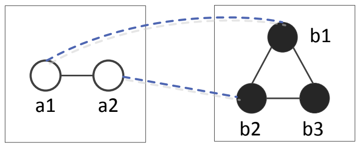
- Recall that all finite structures are relational (no function symbols).
- Let $\mathfrak{A}|_{A'}$ be the substructure of $\mathfrak{A}$ to the subdomain $A' \subseteq A$, i.e., for each relation R:

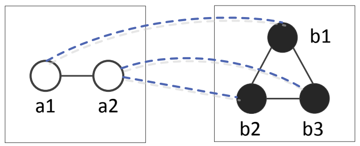
$$R^{\mathfrak{A}|_{A'}} := \{(a_1, \ldots, a_n) \in R^{\mathfrak{A}} | a_1, \ldots, a_n \in A'\}.$$

- A partial function $\zeta : |A| \to |B|$ is a **partial isomorphism** between $\mathfrak{A}$ and $\mathfrak{B}$ if ζ is an isomorphism between $R^{\mathfrak{A}|_{dom(\zeta)}}$ to $R^{\mathfrak{B}|_{rng(\zeta)}}$.

Partial Isomorphism

- Are they partial isomorphisms?





EF Games

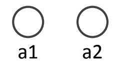
- **Ehrenfeucht-Fraïssé (EF) games**:
  - Fraïssé was the first to find a purely structural necessary and sufficient condition for two structures to be elementarily equivalent (1954).
  - Ehrenfeucht reformulated this condition in terms of games (1961).
- **One of the few model-theoretic techniques** that apply to finite structures as well as infinite ones
  - The infinite case: a number of more powerful tools available
  - The finite case: a central tool for describing expressiveness of logics, e.g., measure the expressive power of database query languages
- Variations for capturing different logics/describing different equivalences

EF Games - Rules

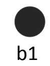
- Two structures $\mathfrak{A}$ and $\mathfrak{B}$ over the same vocabulary.
- Two players: **Spoiler**, **Duplicator**.
  - **Spoiler** tries to show that $\mathfrak{A}$ and $\mathfrak{B}$ are different.
  - **Duplicator** tries to show that $\mathfrak{A}$ and $\mathfrak{B}$ are the same.
- The players play a fixed number of rounds, each having three steps:
  1. **Spoiler** picks a structure ($\mathfrak{A}$ or $\mathfrak{B}$).
  2. **Spoiler** makes a move by picking an element of that structure.
  3. **Duplicator** responds by picking an element in the other structure.
- After n rounds, two sequences have been chosen:
  - $(a_1, \ldots, a_n)$ from $\mathfrak{A}$;
  - $(b_1, \ldots, b_n)$ from $\mathfrak{B}$.

EF Games - Examples

- Consider the following two structures:



$\mathfrak{A} = \langle \{a_1, a_2\}, \emptyset \rangle$



$\mathfrak{B} = \langle \{b_1\}, \emptyset \rangle$

- Some plays:

A 2-round play

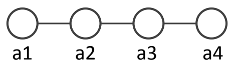
Player	Choice
Spoiler	a_1
Duplicator	b_1
Spoiler	a_2
duplicator	b_1

A 3-round play

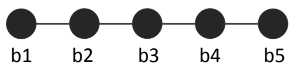
Player	Choice
Spoiler	a_1
Duplicator	b_1
Spoiler	b_1
duplicator	a_1
Spoiler	a_2
duplicator	b_1

EF Games - Examples

- Consider the following two structures:



$\mathfrak{A} = \langle \{a_1, \ldots, a_4\}, \{E\} \rangle$



$\mathfrak{B} = \langle \{b_1, \ldots, b_5\}, \{E\} \rangle$

- Some plays:

A 3-round play

Player	Choice
Spoiler	a_1
Duplicator	b_1
Spoiler	b_4
Duplicator	a_4
Spoiler	b_5
Duplicator	a_3

A 3-round play

Player	Choice
Spoiler	b_3
Duplicator	a_2
Spoiler	a_1
Duplicator	b_2
Spoiler	b_1
Duplicator	a_3

EF Games - Winning Strategies

- How can **Spoiler** or **Duplicator** win in a game?

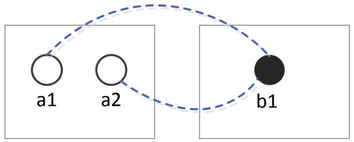
EF Games - Winning Strategies

- **Winning position**: **Duplicator** wins a run of the game if the mapping between elements of the two structures defined by the game run is a partial isomorphism. Otherwise, **Spoiler** wins.
- A player has an **n-round winning strategy** if s/he can play in a way that guarantees a winning position after n rounds, no matter how the other player plays.
- There is always either a winning strategy for **Spoiler** or for **Duplicator**.
- **Notation**:
  - $\mathfrak{A} \sim_n \mathfrak{B}$: if there is an n-round winning strategy for **Duplicator**.
  - $\mathfrak{A} \not\sim_n \mathfrak{B}$: if there is an n-round winning strategy for **Spoiler**.

  Easy to see that $\mathfrak{A} \sim_n \mathfrak{B}$ implies $\mathfrak{A} \sim_k \mathfrak{B}$ for every $k \leq n$.

EF Games - Examples

- Consider the EF game on $\mathfrak{A} = \langle \{a_1, a_2\}, \emptyset \rangle$ and $\mathfrak{B} = \langle \{b_1\}, \emptyset \rangle$.



- Is it a partial isomorphism?

A 2-round play

Player	Choice
Spoiler	a_1
Duplicator	b_1
Spoiler	a_2
Duplicator	b_1

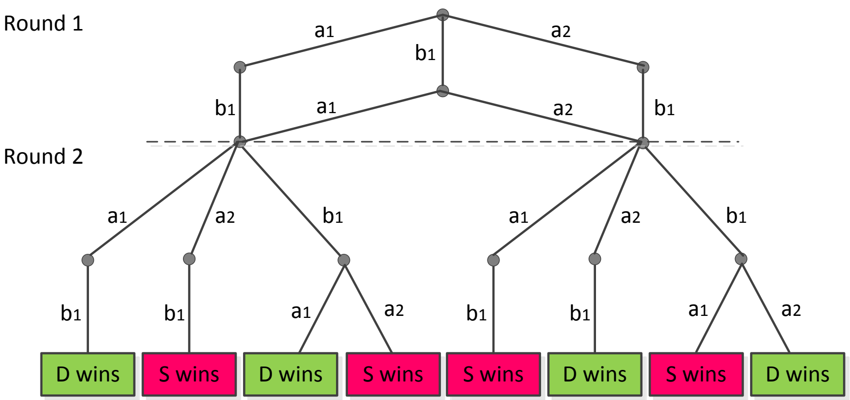
A 3-round play

Player	Choice
Spoiler	a_1
Duplicator	b_1
Spoiler	b_1
Duplicator	a_1
Spoiler	a_2
Duplicator	b_1

- Who wins the plays?

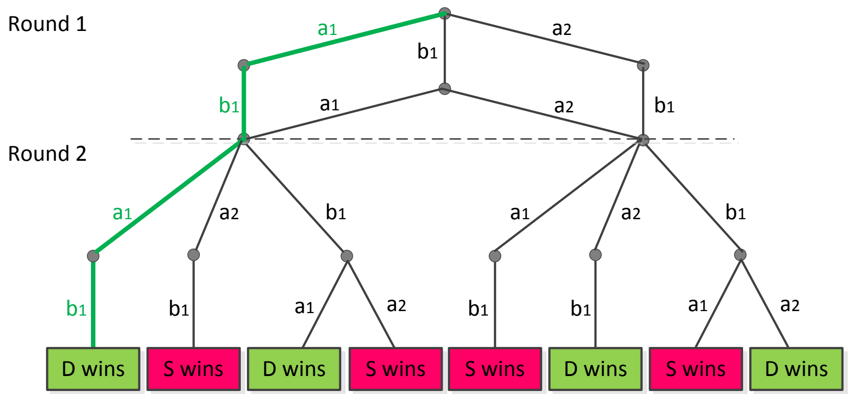
EF Games - Examples

- Consider only 2 rounds of the EF game on $\mathfrak{A} = \langle \{a_1, a_2\}, \emptyset \rangle$ and $\mathfrak{B} = \langle \{b_1\}, \emptyset \rangle$.



EF Games - Examples

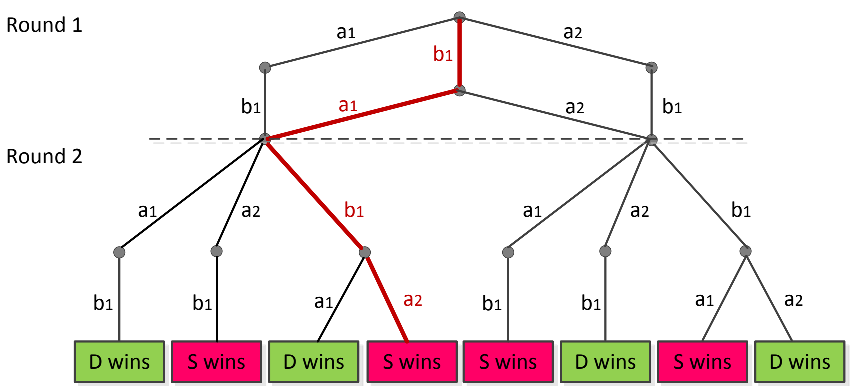
- Consider only 2 rounds of the EF game on $\mathfrak{A} = \langle \{a_1, a_2\}, \emptyset \rangle$ and $\mathfrak{B} = \langle \{b_1\}, \emptyset \rangle$.



- Duplicator has a winning position if $(S \hookrightarrow a_1, D \hookrightarrow b_1, S \hookrightarrow a_1, D \hookrightarrow b_1)$.

EF Games - Examples

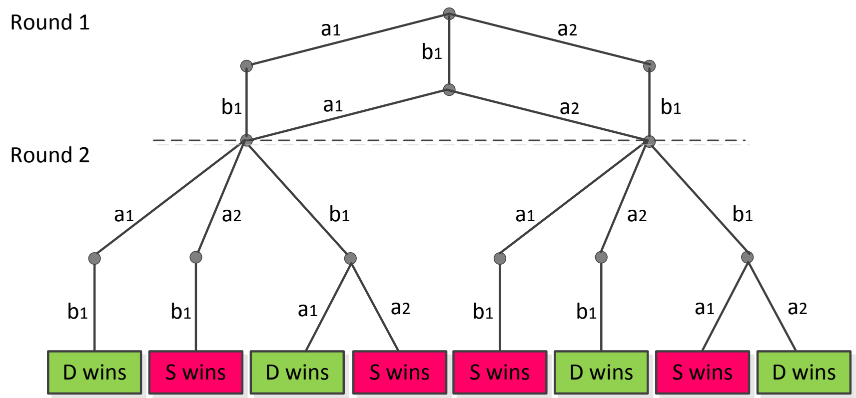
- Consider only 2 rounds of the EF game on $\mathfrak{A} = \langle\{a_1, a_2\}, \emptyset\rangle$ and $\mathfrak{B} = \langle\{b_1\}, \emptyset\rangle$.



- Spoiler has a winning position if ($S \hookrightarrow b_1$, $D \hookrightarrow a_1$, $S \hookrightarrow b_1$, $D \hookrightarrow a_2$).

EF Games - Examples

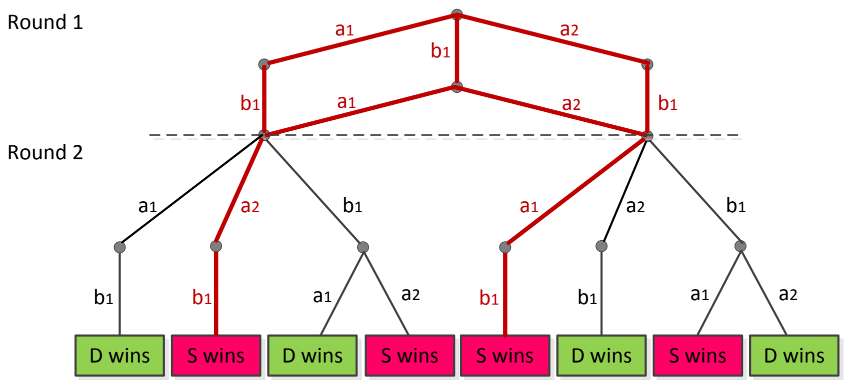
- Consider only 2 rounds of the EF game on $\mathfrak{A} = \langle\{a_1, a_2\}, \emptyset\rangle$ and $\mathfrak{B} = \langle\{b_1\}, \emptyset\rangle$.



- Who has a 2-round winning strategy?

EF Games - Examples

- Consider only 2 rounds of the EF game on $\mathfrak{A} = \langle\{a_1, a_2\}, \emptyset\rangle$ and $\mathfrak{B} = \langle\{b_1\}, \emptyset\rangle$.



- Who has a 2-round winning strategy? Spoiler!

EF Games on Sets

- Let $\sigma = \emptyset$, and $\mathfrak{A}$ and $\mathfrak{B}$ be two sets of size at least n, i.e., $|A|, |B| \geq n$.

- Is it true that $\mathfrak{A} \sim_n \mathfrak{B}$?

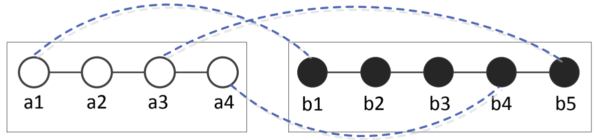
EF Games on Sets

- Let $\sigma = \emptyset$, and $\mathfrak{A}$ and $\mathfrak{B}$ be two sets of size at least n, i.e., $|A|, |B| \geq n$.
- Is it true that $\mathfrak{A} \sim_n \mathfrak{B}$?
- **Winning strategy** for **Duplicator**:
  - Suppose that the position is $((a_1, \ldots a_i), (b_1, \ldots, b_i))$.
  - **Spoiler** picks an element $a_{i+1} \in A$:

$$\begin{cases} \textbf{Duplicator} \text{ picks } b_{i+1} = b_j & \text{if } a_{i+1} = a_j \text{ for } j \leq i \\ \textbf{Duplicator} \text{ picks } b_k \in B - \{b_1, \ldots, b_i\} & \text{otherwise} \end{cases}$$

EF Games - Examples

- Consider 3 rounds of the EF game on $\mathfrak{A} = \langle \{a_1, \ldots, a_4\}, \{E\} \rangle$ and $\mathfrak{B} = \langle \{b_1, \ldots, b_5\}, \{E\} \rangle$.



- Is it a partial isomorphism?

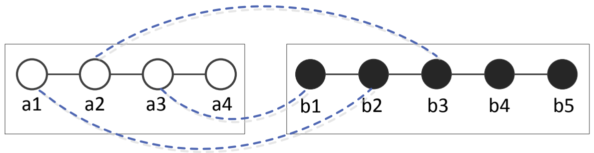
A 3-round play

Player	Choice
Spoiler	a_1
Duplicator	b_1
Spoiler	b_4
duplicator	a_4
Spoiler	b_5
duplicator	a_3

- Who wins the play?

EF Games - Examples

- Consider 3 rounds of the EF game on $\mathfrak{A} = \langle \{a_1, \ldots, a_4\}, \{E\} \rangle$ and $\mathfrak{B} = \langle \{b_1, \ldots, b_5\}, \{E\} \rangle$.



- Is it a partial isomorphism?

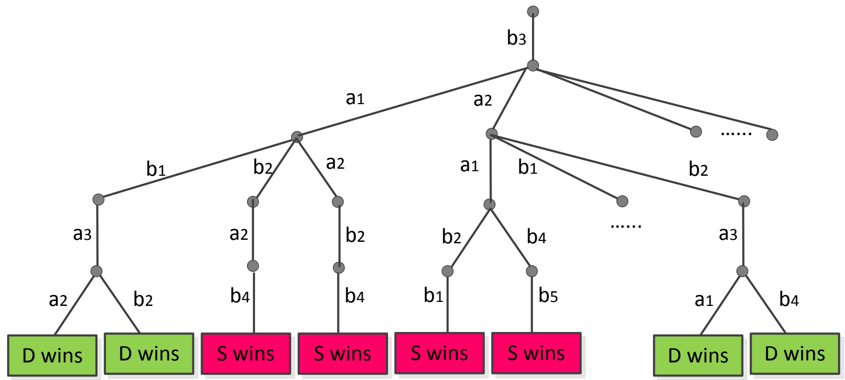
A 3-round play

Player	Choice
Spoiler	b_3
Duplicator	a_2
Spoiler	a_1
duplicator	b_2
Spoiler	b_1
duplicator	a_3

- Who wins the play?

EF Games - Examples

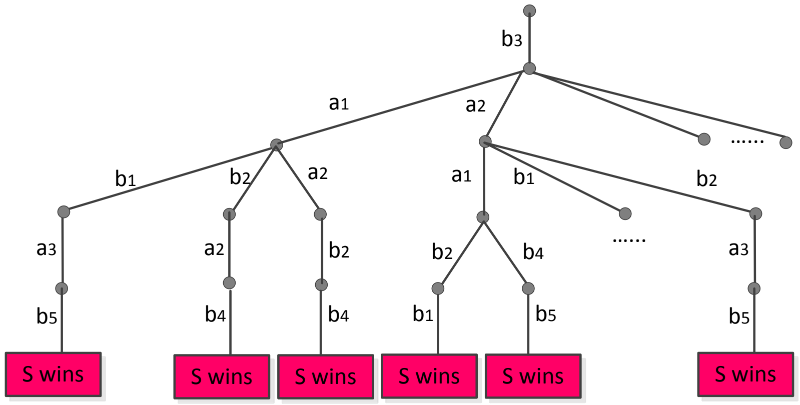
- Consider 3 rounds of the EF game on $\mathfrak{A} = \langle \{a_1, \ldots, a_4\}, \{E\} \rangle$ and $\mathfrak{B} = \langle \{b_1, \ldots, b_5\}, \{E\} \rangle$.



- Who has a 3-round winning strategy?

EF Games - Examples

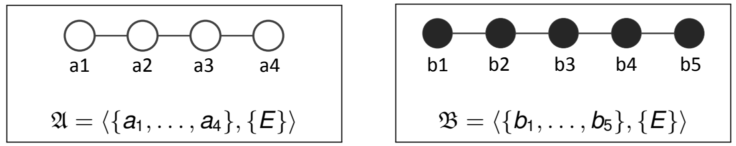
- Consider 3 rounds of the EF game on $\mathfrak{A} = \langle \{a_1, \ldots, a_4\}, \{E\} \rangle$ and $\mathfrak{B} = \langle \{b_1, \ldots, b_5\}, \{E\} \rangle$.



- Who has a 3-round winning strategy? Spoiler!

EF Games - Examples

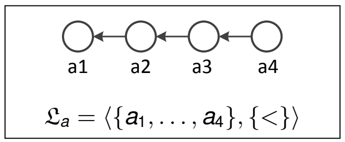
- Consider the EF game on $\mathfrak{A} = \langle \{a_1, \ldots, a_4\}, \{E\} \rangle$ and $\mathfrak{B} = \langle \{b_1, \ldots, b_5\}, \{E\} \rangle$ again.

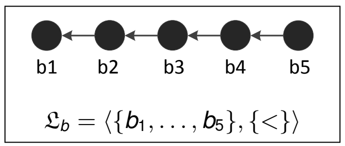


- We know that Spoiler has a 3-round winning strategy now, but
  - Who has a 1-round winning strategy?
  - Who has a 2-round winning strategy?

EF Games - Examples

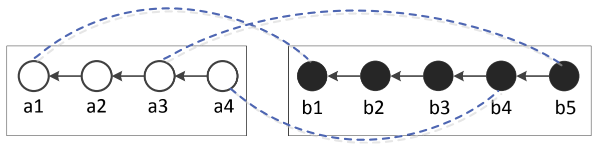
- If we change $\sigma = \{E\}$ to $\sigma = \{<\}$ where $<$ is interpreted as a linear order, and consider the following two structures:





EF Games - Examples

- Consider 3 rounds of the EF game on $\mathfrak{L}_a = \langle \{a_1, \ldots, a_4\}, \{<\} \rangle$ and $\mathfrak{L}_b = \langle \{b_1, \ldots, b_5\}, \{<\} \rangle$.



- Is it a partial isomorphism?

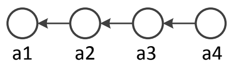
A 3-round play

Player	Choice
Spoiler	a_1
Duplicator	b_1
Spoiler	b_4
duplicator	a_4
Spoiler	b_5
duplicator	a_3

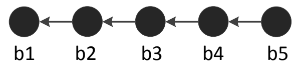
- Who wins the play?

EF Games - Examples

- Consider the following two structures:



$\mathfrak{L}_a = \langle \{a_1, \ldots, a_4\}, \{<\} \rangle$

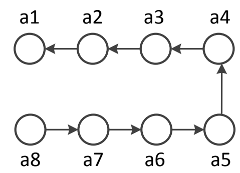


$\mathfrak{L}_b = \langle \{b_1, \ldots, b_5\}, \{<\} \rangle$

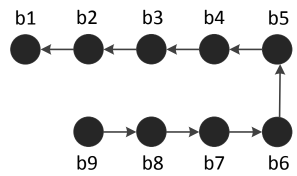
- Who has a winning strategy for 3 rounds of the EF game on $\mathfrak{L}_a$ and $\mathfrak{L}_b$?

EF Games - Examples

- Consider the following two structures:



$\mathfrak{L}_a = \langle \{a_1, \ldots, a_8\}, \{<\} \rangle$



$\mathfrak{L}_b = \langle \{b_1, \ldots, b_9\}, \{<\} \rangle$

- Who has a winning strategy for 3 rounds of the EF game on $\mathfrak{L}_a$ and $\mathfrak{L}_b$?

EF Games on Linear Orders

- Let $\sigma = \{<\}$, and $\mathfrak{L}_a$ and $\mathfrak{L}_b$ be two linear orders of length n and $n+1$, respectively, i.e., $\mathfrak{L}_a = \langle \{1, \ldots, n\}, \{<\} \rangle$ and $\mathfrak{L}_b = \langle \{1, \ldots, n+1\}, \{<\} \rangle$.
- Is it true that $\mathfrak{L}_a \sim_k \mathfrak{L}_b$ for any $k \leq n$?
- Is there a winning strategy for **Duplicator** if the lengths of $\mathfrak{L}_a$ and $\mathfrak{L}_b$ are much larger than the number of rounds?

EF Games on Linear Orders

- **Theorem**: Let $k > 0$, and $\mathfrak{L}_a$ and $\mathfrak{L}_b$ be linear orders of length at least 2^k. Then $\mathfrak{L}_a \sim_k \mathfrak{L}_b$.
- Examples:
  - If $|L_a| = 5$ and $|L_a| = 6$, then $\mathfrak{L}_a \sim_2 \mathfrak{L}_b$ but $\mathfrak{L}_a \not\sim_3 \mathfrak{L}_b$.
  - If $|L_a| = 8$ and $|L_a| = 9$, then $\mathfrak{L}_a \sim_3 \mathfrak{L}_b$ but $\mathfrak{L}_a \not\sim_4 \mathfrak{L}_b$.
- **Duplicator** needs to use the following strategy after r rounds of a EF game, where $1 \leq i < j \leq r$:
  - if $d(a_i, a_j) < 2^{k-r}$, then $d(a_i, a_j) = d(b_i, b_j)$;
  - if $d(a_i, a_j) \geq 2^{k-r}$, then $d(b_i, b_j) \geq 2^{k-r}$;
  - $a_i \leq a_j \Leftrightarrow b_i \leq b_j$;

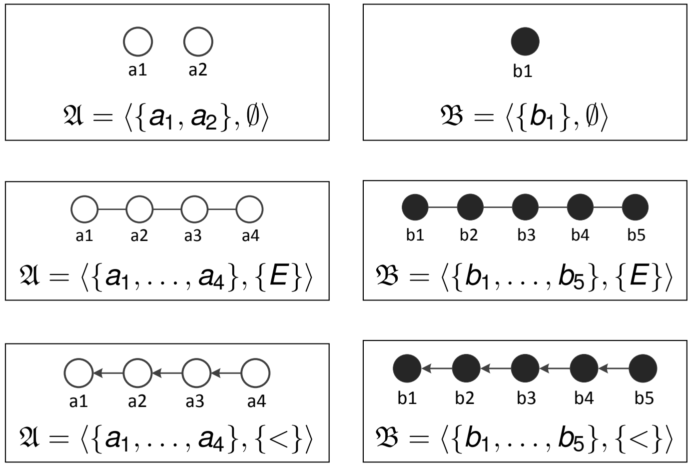
  where $d(x, y)$ denotes the distance between x and y.

Recap

Example 1

Example 2

Example 3



EF Games and FO

- How does EF games relate to FO?

EF Theorem

- **Theorem** (Fraïssé 1954, Ehrenfeucht 1961)

  Given two structures $\mathfrak{A}$ and $\mathfrak{B}$. Then the following are equivalent for every integer k:

  1. $\mathfrak{A} \equiv_k \mathfrak{B}$, i.e., $\mathfrak{A}$ and $\mathfrak{B}$ cannot be distinguished by sentences in $FO[k]$.
  2. $\mathfrak{A} \sim_k \mathfrak{B}$, i.e., **Duplicator** has a winning strategy for the k-round EF game.

- This provides a combinatorial characterization of first-order logic:
  - $\mathfrak{A} \equiv_k \mathfrak{B}$ is defined in terms of logic;
  - $\mathfrak{A} \sim_k \mathfrak{B}$ is defined in terms of games.

EF Theorem - Proof

Proof: $\mathfrak{A} \sim_k \mathfrak{B} \Rightarrow \mathfrak{A} \equiv_k \mathfrak{B}$

- We need to show that: if there is a FO sentence φ with $qr(\varphi) \leq k$ that can distinguish $\mathfrak{A}$ and $\mathfrak{B}$, i.e.

  $$\mathfrak{A} \models \varphi \text{ and } \mathfrak{B} \not\models \varphi,$$

  then **Spoiler** has a winning strategy in the k-round EF games on $\mathfrak{A}$ and $\mathfrak{B}$.
- **Key ideas**:
  - W.l.o.g., assume that all negations are only in front of atomic formulas (i.e., negation normal form).
  - By induction on the quantifier rank, we show that: for φ with $qr(\varphi) \leq k$ and free variables $\{x_1, \ldots, x_n\}$, and two tuples $\bar{a} = (a_1, \ldots, a_n)$ and $\bar{b} = (b_1, \ldots, b_n)$ from $\mathfrak{A}$ and $\mathfrak{B}$ respectively, if

    $$\mathfrak{A} \models \varphi[\bar{a}] \text{ and } \mathfrak{B} \not\models \varphi[\bar{b}],$$

    then **Spoiler** has a winner strategy in the k-round EF game that starts from the moves $(a_1, \ldots, a_n)$ and $(b_1, \ldots, b_n)$.

EF Theorem - Proof

Proof: $\mathfrak{A} \sim_k \mathfrak{B} \Rightarrow \mathfrak{A} \equiv_k \mathfrak{B}$

- By induction on the quantifier rank $qr(\varphi) = k$ of a formula φ with

$$\mathfrak{A} \models \varphi[\bar{a}] \text{ and } \mathfrak{B} \not\models \varphi[\bar{b}].$$

  - If $qr(\varphi) = 0$, i.e.,φ is a quantifier-free formula, then the map from $\bar{a}$ to $\bar{b}$ is not a partial isomorphism.
  - If $\varphi = \exists x \psi$, **Spoiler** chooses an element a_1 for x from $\mathfrak{A}$ s.t.

$$\mathfrak{A} \models \psi[\bar{a}a_1] \text{ and } \mathfrak{B} \not\models \psi[\bar{b}b_1] \text{ for any } b_1 \text{ from } \mathfrak{B}.$$

  - If $\varphi = \forall x \psi$, then $\mathfrak{B} \models \exists x \neg\psi$ and **Spoiler** chooses an element b_2 for x from $\mathfrak{B}$ s.t.

$$\mathfrak{A} \models \psi[\bar{a}a_2] \text{ and } \mathfrak{B} \not\models \psi[\bar{b}b_2] \text{ for any } a_2 \text{ from } \mathfrak{A}.$$

EF Theorem - Proof

Proof: $\mathfrak{A} \equiv_k \mathfrak{B} \Rightarrow \mathfrak{A} \sim_k \mathfrak{B}$

- Given a winning strategy for **Spoiler**, we construct a sentence $\varphi \in FO[k]$ that can distinguish $\mathfrak{A}$ and $\mathfrak{B}$, s.t.

$$\mathfrak{A} \models \varphi \text{ and } \mathfrak{B} \not\models \varphi,$$

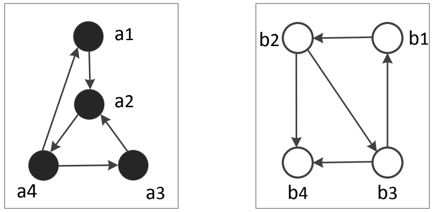
where $\mathfrak{A}$ is the structure from which **Spoiler** chooses an element in the first round, and $\mathfrak{B}$ is the other structure.

EF Games and FO definability

- **Corollary**: A property P is **definable** in FO iff there exists **some** $k \in \mathbb{N}$ such that for **every** two finite structure $\mathfrak{A}$ and $\mathfrak{B}$,
  - $\mathfrak{A} \not\sim_k \mathfrak{B}$, i.e., **Spoiler** has a winning strategy for k-round EF games, and
  - $\mathfrak{A}$ has the property P, but $\mathfrak{B}$ does not.
- If $\mathfrak{A} \not\sim_k \mathfrak{B}$, then a winning strategy for **Spoiler** can be described by a sentence $\in FO[k]$, which is true in exactly one of $\mathfrak{A}$ and $\mathfrak{B}$, and vice versa.

FO Definable Properties

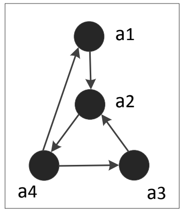
- Can you find a FO definable property in only one of the following directed graphs?

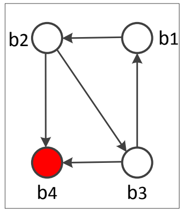


FO Definable Properties

- Consider the following property. Can you construct a winning strategy for **Spoiler**?

$$\exists x \forall y \neg E(x, y)$$

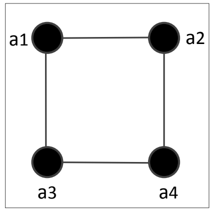


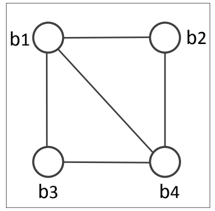


- By EF Theorem, $\mathfrak{A} \not\sim_2 \mathfrak{B}$.

FO Definable Properties

- Can you find a winning strategy for **Spoiler** in the following undirected graph?

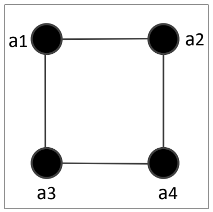


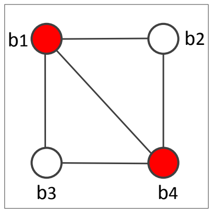


FO Definable Properties

- Given a winning strategy for **Spoiler**: $\{S \hookrightarrow b_1, D \hookrightarrow a_1, S \hookrightarrow a_4, D \hookrightarrow \dots\}$ The following property can be constructed.

$$\exists x \forall y x = y \vee E(x, y)$$

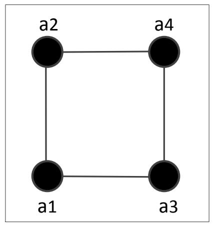


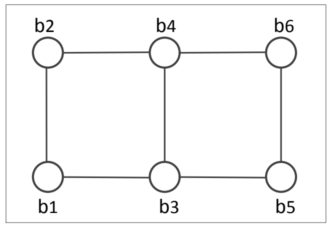


- By EF Theorem, $\mathfrak{A} \not\sim_2 \mathfrak{B}$.

FO Definable Properties

- Find a FO definable property in only one of the following undirected graphs, or find a winning strategy for **Spoiler**.

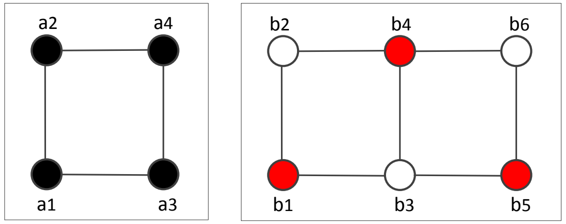




FO Definable Properties

- Consider the following property:

$$\exists x \exists y \exists z (x \neq y \wedge y \neq z \wedge z \neq x \wedge \neg E(x,y) \wedge \neg E(y,z) \wedge \neg E(z,x))$$

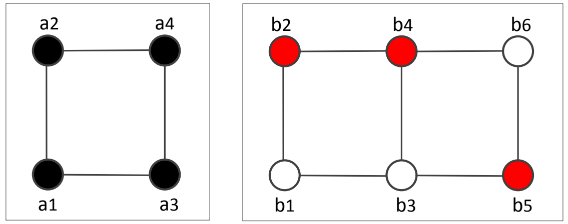


- By EF Theorem, $\mathfrak{A} \not\sim_3 \mathfrak{B}$.

FO Definable Properties

- Consider another property:

$$\exists x \exists y \exists z (x \neq y \wedge y \neq z \wedge z \neq x \wedge E(x,y) \wedge \neg E(y,z) \wedge \neg E(x,z))$$



- By EF Theorem, $\mathfrak{A} \not\sim_3 \mathfrak{B}$.

EF Games and FO Inexpressibility

- How is EF Theorem useful for proving inexpressibility results over finite models?
- **Corollary**: A property P is not expressible in FO if for **every** $k \in \mathbb{N}$, there **exist two** finite structures $\mathfrak{A}$ and $\mathfrak{B}$ s.t.
  - $\mathfrak{A} \sim_k \mathfrak{B}$, i.e., **Duplicator** has a winning strategy for k-round EF games, and
  - $\mathfrak{A}$ has the property P, but $\mathfrak{B}$ does not.
- But finding such structures $\mathfrak{A}_k$ and $\mathfrak{B}_k$ is challenging...

Evenness over Unordered Sets

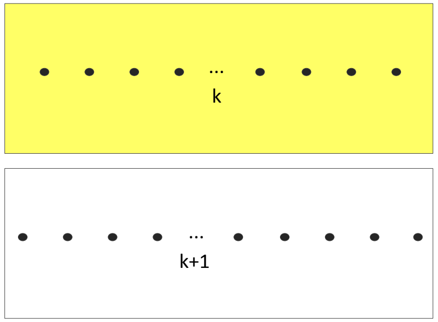
- **Evenness is not expressible over unordered, finite sets in FO.**

Evenness over Unordered Sets

- **Evenness is not expressible over unordered, finite sets in FO**.

  Proof:

  - Pick $\mathfrak{A}$ to be a structure containing k elements, and $\mathfrak{B}$ a structure containing $k+1$ elements.
  - We have $\mathfrak{A} \sim_k \mathfrak{B}$.



Evenness over Linear Order

- **Evenness is not expressible over linearly ordered, finite sets in FO**.

  Hints:

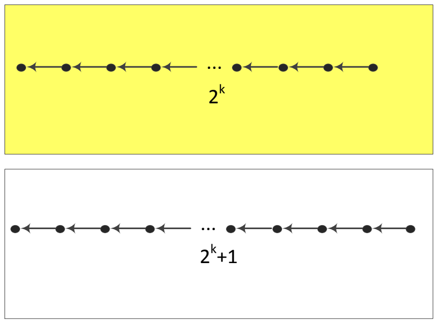
  Theorem: Let $k > 0$, and $\mathfrak{L}_a$ and $\mathfrak{L}_b$ be linear orders of length at least 2^k. Then $\mathfrak{L}_a \sim_k \mathfrak{L}_b$.

Evenness over Linear Order

- **Evenness is not expressible over linearly ordered, finite sets in FO**.

  Proof:

  - Pick $\mathfrak{A}_k$ to be a linear order of length 2^k, and $\mathfrak{B}_k$ to be a linear order of length $2^k + 1$.
  - We have $\mathfrak{A}_k \sim_k \mathfrak{B}_k$.



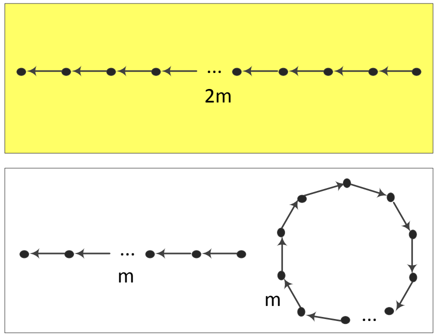
Acyclicity

- A graph is **acyclic** if it does not contain any cycles.
- **Acyclicity of finite graphs is not expressible in FO**.

Acyclicity

- **Acyclicity of finite graphs is not expressible in FO.**

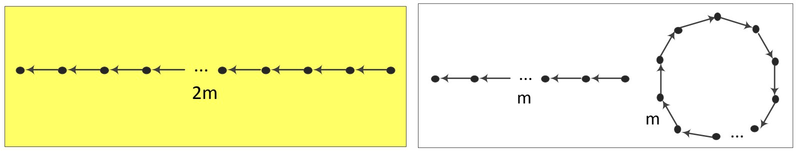
  Proof:
  - Let m depend only on k, and be sufficiently large.
  - Assume that the game starts in a position where two special nodes (i.e., the start and end nodes of the success relation) have been played.



Acyclicity

- **Acyclicity of finite graphs is not expressible in FO.**

  Proof (continue):
  - Let $d(a_j, a_i)$ denote the distance between a_j and a_i, i.e., the length of the shortest path between them.
  - **Duplicator** maintains the following conditions after each round r:
    - if $d(a_j, a_i) \leq 2^{k-r}$, then $d(b_j, b_i) = d(a_j, a_i)$.
    - if $d(a_j, a_i) > 2^{k-r}$, then $d(b_j, b_i) > 2^{k-r}$.
  - By choosing m “very large”, if r rounds have been played, there is a node at a distance greater than $2^{k-(r+1)}$ from all the played nodes.



2-colorability

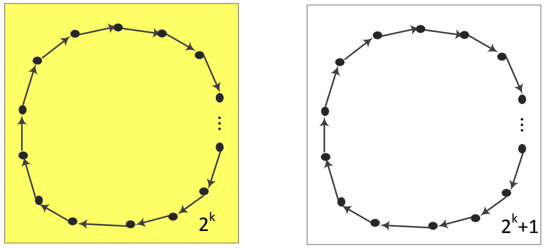
- A graph is called **2-colorable** if one can color each node in either red or green such that no two adjacent nodes have the same color.
- **2-colorability of finite graphs is not expressible in FO.**

  Hint: A cycle of length n is 2-colorable iff n is even.

2-colorability

- A graph is called **2-colorable** if one can color each node in either red or green such that no two adjacent nodes have the same color.
- **2-colorability of finite graphs is not expressible in FO.**

  Hint: A cycle of length n is 2-colorable iff n is even.

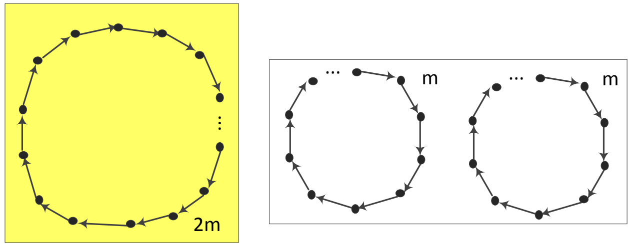


Connectivity

- A graph is **connected** if there exists a path between any two nodes of the graph.
- **Connectivity of finite graphs is not expressible in FO**.

Connectivity

- A graph is **connected** if there exists a path between any two nodes of the graph.
- **Connectivity of finite graphs is not expressible in FO**.



Conclusions

- In general, finding families of structures $\{\mathfrak{A}_k | k \in \mathbb{N}\}$ and $\{\mathfrak{B}_k | k \in \mathbb{N}\}$ is hard.
- In addition to this, it is also hard to prove that $\mathfrak{A}_k \sim_k \mathfrak{B}_k$.
- The complexity of proofs using EF games can quickly increase as the structures become complicated.
- To avoid complicated combinatorial arguments, it is possible to use simple sufficient conditions that guarantee a winning strategy for the duplicator, i.e., build a library of winning strategies.
- For FO, most such conditions are based on the idea of locality.
- EF games can be modified to provide methodologies for other logical languages.