Ehrenfeucht-Fraïssé Games

Bhumika Mittal

Abstract

This paper presents an exposition on Ehrenfeucht-Fraïssé Games, a tool in finite model theory and logic. Ehrenfeucht-Fraïssé games provide a game-theoretic characterization of first-order logic expressibility, offering an intuitive and elegant way to determine what properties can and cannot be expressed in first-order logic. We explore the definition of these games, their connection to first-order logic through the Ehrenfeucht-Fraïssé theorem, and their application as a methodology for proving non-expressibility results.

1 Introduction

Ehrenfeucht-Fraïssé games, named after Andrzej Ehrenfeucht and Roland Fraïssé, are a fundamental tool in finite model theory that provides a game-theoretic semantics for first-order logic. These games offer an alternative, and often more intuitive, way to understand the expressive power of first-order logic compared to the standard syntactic definitions.

The central idea behind Ehrenfeucht-Fraïssé games is to model the process of distinguishing between two structures as a game between two players: the *Spoiler*, who aims to expose differences between the structures, and the *Duplicator*, who tries to maintain similarity. The number of rounds in the game corresponds to the quantifier rank of first-order formulas. The key insight is that the Duplicator has a winning strategy in an n-round game if and only if the two structures satisfy the same first-order sentences of quantifier rank at most n.

This game-theoretic characterization provides a complete methodology for proving that certain properties are not expressible in first-order logic. If we can construct two structures—one satisfying a property P and the other not—such that the Duplicator wins the n-round game for any n, then P cannot be expressed by any first-order formula, regardless of its quantifier rank.

In this paper, we begin with preliminaries on games on graphs, which provide the mathematical foundation. We then define Ehrenfeucht-Fraïssé games formally, explore winning strategies, and establish the fundamental connection to first-order logic through the Ehrenfeucht-Fraïssé theorem. Finally, we demonstrate how these games serve as a methodology for proving non-expressibility results, with several concrete examples including connectivity, reachability, and parity.

2 Preliminaries: Games on Graphs

Games played on graphs form a powerful tool to reason about various questions arising in Logic and Automata theory. This section introduces the fundamental concepts that form the basis of game-theoretic analysis.

2.1 What is a Game?

The first model of games that we will define are 2-player, zero-sum, turn-based, deterministic, perfect information games.

2.1.1 Players

The term 2-player games means that there are two people that are playing the game. We use various names to refer to the players. Commonly used names are: Eve and Adam, Player 0 and Player 1, Circle and Square corresponding to the graphical representation of games, Even and Odd corresponding to parity based games, Pathfinder and Verifier corresponding to representation using automaton, Min and Max corresponding to quantitative games. We use Eve and Adam to refer to players in qualitative games, and Min and Max to refer to players in quantitative games.

We will also be referring to 1-player games when there is only one player. In stochastic games, we have an additional random player which we refer to as the $\frac{1}{2}$ player. In games involving more than two players, we call them multiplayer games.

2.1.2 Graphs

A (directed) graph is given by a set V of vertices and a set E of edges given by the functions In, Out : $E \rightarrow V$: for an edge e we write $In(e)$ for the incoming vertex and $Out(e)$ for the outgoing vertex. We say that e is an outgoing edge of $In(e)$ and an incoming edge to $Out(e)$. To introduce an edge, it is convenient to write $e = v \rightarrow v'$ to express that $v = In(e)$ and $v' = Out(e)$. A path π is a finite or infinite sequence:

$$\pi = v_0 \rightarrow v_1 \rightarrow v_2 ...$$

We use first(π) to denote the first vertex of the path π and use last(π) to denote the last vertex of the path π . When a path is finite, we say that π starts from first(π) and ends in last(π). We will be using the notation $\pi_{\leq i}$ for the finite path $v_0 \rightarrow v_1 \rightarrow ... \rightarrow v_i$.

We define $Paths(G)$ which denote the set of finite paths in the graph G. To denote paths in a graph G starting from a vertex v, we say $Paths(G, v)$ .We define $Paths_\omega(G)$ which denote the set of infinite paths in a graph G. To denote paths in a graph G starting from a vertex v, we say $Paths_\omega(G, v)$.

We will use standard terminology associated with graphs:

- Successor: a vertex v' is a successor of v if $\exists$ $e \in E$ such that $In(e) = v$ and $Out(e) = v'$.
- Predecessor: a vertex v is a predecessor of v' , a vertex v' is reachable from v if $\exists$ a path starting from v that ends in v'.
- Outdegree: for a vertex v, the outdegree tells us the number of edges leaving that vertex.
- Indegree: for a vertex v, the indegree tells us the number of edges entering that vertex.
- Simple Path: a path where no vertices is repeated or more formally, a path π where $\pi = v_0 \rightarrow v_1 \rightarrow v_2 ... \rightarrow v_i$ such that each of the vertices are unique.
- Cycle: a path where the the first and last vertex coincide or more formally, a path π where $\pi = v_0 \rightarrow v_1 \rightarrow v_2 ... \rightarrow v_i$ such that $v_1 = v_i$.
- Simple cycle: a cycle that strictly does not contain another cycle.
- Self loop: an edge from a vertex to itself.
- Sink: a vertex with only self loops as outgoing edges.

2.1.3 Arenas

The arena is the place where the game is played, they are also called game structures or game graphs. Since, we have defined our model to be a turn-based game, wherein between two consecutive moves of a player, there needs to a be a move made by the opponent. In this turn-based setting, we divide the set of vertices into vertices controlled by each player. Since we are interested in 2-player games, we can define the set of vertices as $V = V_{Eve} \uplus V_{Adam}$ where V_{Eve} is the set of vertices controlled by Eve and V_{Adam} is the set of vertices controlled by Adam. We use circles to represent vertices belonging to V_{Eve} and squares to represent squares belonging to V_{Adam}.

An arena is given by a graph and the sets V_{Eve} and V_{Adam}. In the context of games, vertices are also referred to as positions. We call an arena finite when there are finitely many vertices and edges. An important assumption called *perfect information* means that the players can see everything about how the game is played out, in particular they see the other players' moves. We also assume that all vertices have an outgoing edge.

2.1.4 Play

The interaction between the two players consists in moving a token on the vertices of the arena. Initially, the token is on some vertex, lets call this vertex v. We have learnt that each player controls a set of vertices. WLOG we can say Player 1 controls v and they choose some edge e such that $e = v \to v'$ and pushes the token along this edge. Now, the token is at vertex v'. The outcome of this interaction is a sequence of vertices that were traversed by the token: this is a path. Consequently, a path can also be called a *play* . Similar to paths, plays can be finite or infinite.

2.1.5 Strategies

A strategy (or a *policy*) is the full description of a players moves in all situations. Formally, we define strategy as a function that maps finite plays to edges. We use σ to denote a strategy.

$$\sigma : Paths \to E$$

Notation: We use σ to denote strategies of Eve and Max and τ to denote strategies of Adam and Min.

We say that a play $\pi = v_0 \to v_1 \to v_2 ...$ is consistent with a strategy σ of Eve if $\forall i$ such that $v_i \in V_{Eve}$ we have $\sigma(\pi_{\leq i}) = v_i \to v_{i+1}$ We can use the same idea to represent Adam's strategies as well. Once an initial vertex v and two strategies σ and τ have been fixed, there exists a unique infinite play starting from v and consistent with both strategies, it is written with the following notation $\pi^v_{\sigma,\tau}$. Note that the fact that it is infinite follows from our assumption that all vertices have an outgoing edge.

2.1.6 Winning Conditions

Condition or winning conditions is what Eve wants to achieve. There are two types of winning conditions based on the two types of game qualitative and quantitative conditions.

Qualitative condition is defined as $W \subseteq Paths_\omega$ where W is a set of paths such that if Eve takes a path π such that $\pi \in W$, Eve is taking a winning path (or play).

Quantitative condition is defined as $f : Paths_\omega \to R \cup \pm\infty$ where f assigns a real value to a path that Eve is taking and we want to maiximize the value of this path.

2.1.7 Objectives

2.1.8 Coloring Functions

Colouring function as the name suggests is a mapping between labelling edges of a graph and a set of colours. $c : E \to C$

We can extend the function c to also map a *play* or a path to a sequence of colours. $c : Paths_\omega \to C^\omega$

$$c(e_0 e_1 ...) = c(e_0)c(e_1)...$$

A colouring function along with an objective induces a condition. Based on the type of objective: qualitative or quantitative, corresponding condition is induced. When we have a qualitative objective Ω where $\Omega(c) = \{\pi \in Paths_\omega : c(\pi) \in \Omega\}$and a colouring function c , we have the corresponding qualitative condition given by $\Omega(c)(\pi) = c(\pi) \in \Omega$.

Similarly, for a quantitative objective Φ where $\Phi : C^\omega \to R \cup \{\pm\infty\}$ and a colouring function c, we have the corresponding quantitative condition given by $\Phi(c)$.

$$\Phi(c)(\pi) = \Phi(c(\pi))$$

2.1.9 Games

1. A graph is a tuple $G = (V, E, In, Out)$ where is E is a set of edges and V is a set of vertices and $In, Out : E \to V$ defines the incoming and outgoing vertices of edges.
2. An arena is a tuple $A(G, V_{Eve}, V_{Adam})$ where G is a graph over the set of vertices V and $V = V_{Eve} \uplus V_{Adam}$. For quantitative games, we define $V = V_{Min} \uplus V_{Max}$.
3. A colouring function is a function $c : E \to C$ where C is a set of colours.

3 Ehrenfeucht-Fraïssé Games

3.1 Definition of the Games

Consider the EF game between two players, the *Spoiler* and the *Duplicator*, played on two graphs, denoted by A and B. The Spoiler's objective is to exhibit a structural difference between A and B, whereas the Duplicator aims to demonstrate their similarity.

The game is played over a fixed number of rounds. In each round, the following steps are executed:

1. The Spoiler selects one of the graphs, either A or B.
2. The Spoiler chooses a vertex from the selected graph.
3. The Duplicator responds by choosing a vertex from the other graph.

Let x_i denote the vertex chosen from A and y_i the vertex chosen from B in the i-th round, for $i = 1, \ldots, n$. Define the sets $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$.

The Duplicator wins the n-round EF game if the mapping $f : X \to Y$ defined by $f(x_i) = y_i$ for each i is a *partial isomorphism*. This means that:

1. For any $x_i, x_j \in X$, x_i and x_j are adjacent in A if and only if y_i and y_j are adjacent in B.
2. For any $x_i, x_j \in X$, $x_i = x_j$ if and only if $y_i = y_j$.

In this case, we write $A \sim_n B$. If the conditions are not met, the Spoiler wins.

It is important to note that since the Spoiler chooses the graph in each round, this decision can vary from round to round. Moreover, if both A and B contain more than n vertices, it is strategically redundant for the Spoiler to repeat a vertex selection, as the Duplicator may simply replicate the corresponding move.

3.2 Winning Strategies

A *strategy* for a player is a predefined set of rules that dictate the player's choices based on the history of moves. A strategy is deemed *winning* if it guarantees victory regardless of the opponent's actions. Crucially, only one player can have a winning strategy, since if both did, the game would lead to a contradiction.

3.2.1 Spoiler's Winning Strategy

A winning strategy for the Spoiler can be formulated as follows:

$$\begin{gathered} \exists \, \text{a move for the Spoiler such that } \forall \, \text{responses by the Duplicator,} \\ \exists \, \text{a subsequent move for the Spoiler such that } \forall \, \text{further responses, } \ldots, \end{gathered}$$

ensuring that after n rounds, the Spoiler achieves a win.

3.2.2 Duplicator's Winning Strategy

Conversely, a winning strategy for the Duplicator is described by:

$$\begin{aligned} \forall \, \text{moves made by the Spoiler,} \\ \exists \, \text{a corresponding response by the Duplicator such that} \\ \forall \, \text{subsequent moves by the Spoiler, } \exists \, \text{a valid response, } \ldots, \end{aligned}$$

which guarantees that after n rounds the Duplicator wins. It is also observed that if the Duplicator possesses a winning strategy for the n-round game, this strategy remains effective for any game played with fewer than n rounds.

3.3 First-Order Logic for Graphs

We adapt first order logic (FOL) to the study of graphs. In the language of graphs, the binary relation $E(x, y)$ serves as the edge relation between nodes x and y. The formula $E(x, y)$ is interpreted as true on a graph A if there is an edge from node x to node y; otherwise, it is false.

3.3.1 Atomic Formulas and Logical Connectives

- **Atomic Formulas:** The basic unit is an atomic formula. For instance, $E(1, 2)$ is an atomic formula where 1 and 2 are interpreted as constant symbols representing specific nodes.

- **Logical Connectives:** From atomic formulas, one constructs more complex formulas using:

  - Conjunction: $\varphi \wedge \psi$
  - Disjunction: $\varphi \vee \psi$
  - Negation: $\neg\varphi$

3.3.2 Quantifiers and Variables

FOL extends propositional logic by including quantifiers and variables. In the context of graphs:

- $\exists$ denotes the existential quantifier.
- $\forall$ denotes the universal quantifier.

Variables (typically denoted by letters such as x, y, etc.) range over the nodes of the graph, while constants represent specific nodes. A typical atomic formula such as $E(1, y)$ can be viewed as a relation that is true for those nodes y that are adjacent to the node represented by the constant 1.

The syntax for first order logic, as applied to graphs, can be summarized as follows:

$$\begin{aligned}\varphi, \psi &::= E(t_1, t_2) \mid t_1 = t_2 \mid \neg\varphi \mid (\varphi \wedge \psi) \mid (\varphi \vee \psi) \mid \exists x\, \varphi \mid \forall x\, \varphi \\ t &::= \text{variable} \mid \text{constant}\end{aligned}$$

In this syntax, $E(t_1, t_2)$ represents the edge relation between terms t_1 and t_2, and the equality $t_1 = t_2$ allows for the expression of identity between nodes.

Definition 3.1. An important measure in FOL is the *quantifier rank* of a formula, which is defined as the maximum depth of nested quantifiers. Formally, for a first order formula φ:

- If φ is atomic, then $\mathrm{qr}(\varphi) = 0$.
- If $\varphi = \neg\psi$, then $\mathrm{qr}(\varphi) = \mathrm{qr}(\psi)$.
- If $\varphi = \psi_1 \wedge \psi_2$ or $\varphi = \psi_1 \vee \psi_2$, then $\mathrm{qr}(\varphi) = \max\{\mathrm{qr}(\psi_1), \mathrm{qr}(\psi_2)\}$.
- If $\varphi = \exists x\, \psi$ or $\varphi = \forall x\, \psi$, then $\mathrm{qr}(\varphi) = \mathrm{qr}(\psi) + 1$.

The quantifier rank provides a measure of the complexity of a formula with respect to the nesting of quantifiers.

3.4 EF Games and First-Order Logic

Ehrenfeucht-Fraïssé games offer a semantics for first-order logic that is equivalent to, but more directly applicable than, the standard definitions.

Theorem 3.1. *(Ehrenfeucht-Fraïssé Theorem) For two graphs, A and B, the following are equivalent:*

1. *A and B satisfy the same first-order sentences of quantifier rank n.*
2. *$A \sim_n B$, i.e., the Duplicator has a winning strategy in the n-round Ehrenfeucht-Fraïssé game on A and B.*

Proof. We will prove the theorem by induction on n.

Base Case: For $n = 0$, the 0-round EF game requires no moves, so Duplicator wins trivially. This means that A and B must satisfy the same quantifier-free first-order sentences. Hence, the base case holds.

Inductive Step: Assume the statement is true for $n = k$. We will prove it for $n = k + 1$.
Let φ be a first-order sentence of quantifier rank $k + 1$. We can write φ as $\exists x\psi(x)$, where ψ is a first-order sentence of quantifier rank k. By the induction hypothesis, $A \sim_k B$ if (and only if)

A and B satisfy the same first-order sentences of quantifier rank k. Let $A \sim_k B$. We will show that $A \sim_{k+1} B$.

Consider the Ehrenfeucht-Fraïssé game on A and B with $k + 1$ rounds. Let D be the Duplicator's winning strategy in the k-round game. The Spoiler will choose a vertex from either of the graph in $k+1$-th round. The Duplicator will respond with a vertex from the other graph such that the vertices are indistinguishable.

`Case 1`: Suppose the Spoiler chooses a vertex $a \in A$ in the $(k + 1)$-th round. The Duplicator responds with a vertex $b \in B$ such that $a \sim_k b$. Since $A \sim_k B$, this ensures that a and b satisfy the same sentences of quantifier rank k. In particular, if $A \models \psi(a)$, then $B \models \psi(b)$. We now prove that such a vertex b exists. If no such b exists, we can add a vertex b with the same neighbors as a in B. This will ensure that $a \sim_k b$. Hence, such a vertex b exists.

Alternatively, let's say the the Spoiler chooses a vertex $a \in A$ in the first round. The Duplicator responds with a vertex $b \in B$. Now, we can remove a from A and b from B. We claim that the $A' = A/a$ and $B' = B/b$ satisfy the same first-order formula ψ of the rank k. By the induction hypothesis, $A' \sim_k B'$. This implies that $A \sim_{k+1} B$. We can prove that this claim holds because to show that A and B are partially isomorphic, we need to satisfy the following conditions:

1. $x_i = x_j \iff y_i = y_j$ for all $x_i, x_j \in A$ and $y_i, y_j \in B$.
2. x_i and x_j are adjacent in A if and only if y_i and y_j are adjacent in B.

After removing a and b, the first condition is satisfied. The second condition holds due to the way ψ is modified based on the existential quantifier. When we remove x, we also remove all edges incident on x. Consequently, all such edges are removed from both A and B, ensuring that the second condition is satisfied as well.

`Case 2`: Suppose the Spoiler chooses a vertex $b \in B$ in the $(k + 1)$-th round. The Duplicator responds with a vertex $a \in A$ such that $a \sim_k b$. This case follows symmetrically from the previous argument.

Therefore, $A \sim_{k+1} B$. By induction, this holds for all n.

Now, let us show that if the Duplicator has a winning strategy in the n-round Ehrenfeucht-Fraïssé game on A and B, then A and B satisfy the same first-order sentences of quantifier rank n. We proceed by induction on the structure of formulas (structural induction):

- *Atomic formulas*: For quantifier rank 0, the base case ensures A and B satisfy the same atomic sentences. Since the Duplicator wins the 0-round game, all atomic sentences (if any) are preserved.
- *Negation*: Suppose $\varphi = \neg\psi$. Then ψ has quantifier rank n. By induction, $A \models \psi \iff B \models \psi$. Thus, $A \models \neg\psi \iff B \models \neg\psi$.
- *Conjunction/Disjunction*: Let $\varphi = \psi \wedge \theta$ or $\psi \vee \theta$. Both ψ and θ have quantifier rank $\leq n$. By induction, $A \models \psi \iff B \models \psi$ and $A \models \theta \iff B \models \theta$. Hence, the truth of φ is preserved.
- *Existential Quantifier*: Let $\varphi = \exists x \psi(x)$, where $\psi(x)$ has quantifier rank $n - 1$. If $A \models \varphi$, there exists $a \in A$ such that $A \models \psi(a)$. In the EF game, Spoiler plays a, and Duplicator responds with $b \in B$ such that the remaining $(n - 1)$-round game is a win for Duplicator.

By the induction hypothesis, $(A, a) \sim_{n-1} (B, b)$, so $B \models \psi(b)$, implying $B \models \varphi$. The converse holds symmetrically.

Thus, by induction, A and B satisfy the same first-order sentences of quantifier rank n. □

3.5 Methodology for First-Order Expressibility

Ehrenfeucht-Fraïssé games provide a complete methodology for proving that a query is not first-order. We have already seen that these games are a convenient tool for determining what can be said in first-order logic. This theorem says that if we can show using any method that a query is not first-order expressible, then we can show it using Ehrenfeucht-Fraïssé games.

Lemma 3.2. *There are only finitely many inequivalent first order logic formulas of quantifier rank k.*

Proof. Let Σ be a finite signature (finitely many constant symbols and relation symbols, with no function symbols) and let $r(\varphi)$ denote the quantifier rank of a formula φ. We can show that there are only finitely many inequivalent formulas of quantifier rank k by induction on k.

Base Case: For $k = 1$, a generic form is $Qx\varphi(x)$, where Q is a quantifier $(\exists, \forall)$ and $\varphi(x)$ is a formula of quantifier rank 0. There are only finitely many atomic formulas upto variable renaming because the signature is finite. Boolean combinations of finitely many atomic formulas yield finitely many inequivalent atomic formulas and for each φ, thee are two possibilities $\exists x\varphi(x)$ and $\forall x\varphi(x)$. Hence, the base case holds.

Inductive Hypothesis: Assume that there are only finitely many inequivalent formulas of quantifier rank $\leq k$. We will show the same for rank $k + 1$.

Inductive Step: Let φ be a formula of quantifier rank $k + 1$. Then, φ can be written as $Qx\psi(x)$, where Q is a quantifier and $\psi(x)$ is a formula of quantifier rank k. By the inductive hypothesis, there are only finitely many inequivalent formulas of quantifier rank k. Since there are only finitely many possible formulas $\psi(x)$, there are only finitely many possible formulas of quantifier rank $k + 1$. □

Theorem 3.3. *(Methodology Theorem) Given a property P, there is no first-order logic formula that expresses P if and only if there is a pair of graphs $\mathcal{A}$ and $\mathcal{B}$ such that P holds on $\mathcal{A}$ and P doesn't hold on $\mathcal{B}$, and the Duplicator has a winning strategy in the k-move Ehrenfeucht-Fraïssé game between $\mathcal{A}$ and $\mathcal{B}$.*

Proof. Given there exists such a pair of graphs $\mathcal{A}$ and $\mathcal{B}$, it is straightforward from the EF Theorem that there exists no first-order logic formula that expresses P. Conversely, if there is no first-order logic formula that expresses P, then we show such $\mathcal{A}$ and $\mathcal{B}$ exist. By the lemma, there are finitely many inequivalent quantifier-rank k formulas. Let $\{\varphi_1, \ldots, \varphi_r\}$ list all *complete* formulas of rank k, where each φ_i decides every quantifier-rank k sentence. There could be the following two cases:

Case 1: Suppose for some φ_j, there exist structures $\mathcal{A} \models \varphi_j$ with $P(\mathcal{A})$ and $\mathcal{B} \models \varphi_j$ with $\neg P(\mathcal{B})$. These $\mathcal{A}$ and $\mathcal{B}$ satisfy the theorem's conditions.

Case 2: If no such φ_j exists, then every φ_i is either:

- *Always P:* All models of φ_i satisfy P, or
- *Never P:* All models of φ_i fail P.

Let $Y = \{i \mid \text{models of } \varphi_i \text{ satisfy } P\}$. Then P is defined by $\bigvee_{i \in Y} \varphi_i$, contradicting P's undefinability. Thus, Case 2 is impossible, and Case 1 must hold. For the φ_j in Case 1, $\mathcal{A}$ and $\mathcal{B}$ satisfy the same φ_j. By the EF Theorem, the Duplicator has a winning strategy in the k-move EF game between them. $\square$

Now, let us develop an algorithm that constructs a pair of graphs $\mathcal{A}$ and $\mathcal{B}$ such that $\mathcal{A} \models \phi$ and $\mathcal{B} \not\models \phi$, and the Duplicator has a winning strategy in the k-move Ehrenfeucht-Fraïssé game between $\mathcal{A}$ and $\mathcal{B}$.

Let $\mathcal{A}$ be a graph and $v \in \mathcal{A}$. Let $\text{dist}(v, u)$ denote the number of edges in the shortest path between vertices v and u. The *d-neighborhood* $N(v, d)$ is the induced subgraph on $\{u \in \mathcal{A} \mid \text{dist}(v, u) \leq d\}$. The *$d$-type* of v, denoted $\mathsf{type}_d(v)$, is the isomorphism class of $N(v, d)$.

Let $\mathcal{A}$ and $\mathcal{B}$ be graphs and $k \in \mathbb{N}$. If for every 2^k-type t, the number of vertices with t in $\mathcal{A}$ equals the number in $\mathcal{B}$, then:

$$\mathcal{A} \equiv_k \mathcal{B}$$

i.e., Duplicator has a winning strategy in the k-round Ehrenfeucht-Fraïssé game. We can prove this as follows.

Duplicator maintains the following invariant $\mathcal{I}(i)$ for all $0 \leq i \leq k$:

After i moves, $\mathcal{A}$ and $\mathcal{B}$ have the same number of vertices with each 2^{k-i}-type.

Base case ($i = 0$): By assumption, $\mathcal{A}$ and $\mathcal{B}$ have identical 2^k-type counts. Thus, $\mathcal{I}(0)$ holds.

Inductive step: Assume $\mathcal{I}(i)$ holds for $0 \leq i < k$. On move $i+1$, Spoiler picks $v \in \mathcal{A}$. By $\mathcal{I}(i)$, there exists $v' \in \mathcal{B}$ with $\mathsf{type}_{2^{k-i}}(v) = \mathsf{type}_{2^{k-i}}(v')$. Duplicator plays v'. Let $s = 2^{k-(i+1)}$. The isomorphism $f : N(v, 2^{k-i}) \to N(v', 2^{k-i})$ ensures:

$$\forall u \in N(v, s),\ \mathsf{type}_s(u) = \mathsf{type}_s(f(u)).$$

For $u \notin N(v, 2^{k-i})$, $\mathsf{type}_s(u)$ is preserved because:

$$\text{dist}(u, v) > 2^{k-i} \implies N(u, s) \cap N(v, s) = \emptyset.$$

Thus, $\mathcal{I}(i+1)$ holds.

After k rounds, all chosen pairs have isomorphic 1-neighborhoods, ensuring $\mathcal{A} \equiv_k \mathcal{B}$. From EF theorem, the Duplicator has a winning strategy in the k-move Ehrenfeucht-Fraïssé game between $\mathcal{A}$ and $\mathcal{B}$.

Proposition 3.1. *A linear order is a binary relation $\leq$ on some set A that is reflexive, transitive, antisymmetric, and total. A linear order is said to be "even" if it has an even number of nodes. The class of "even" linear orders is not FO definable.*

Proof. For each n, consider a linear order $A_n = G_{2^n}$ and $B_n = G_{2^n+1}$, where G_n is the linear order on n vertices, depicted below:

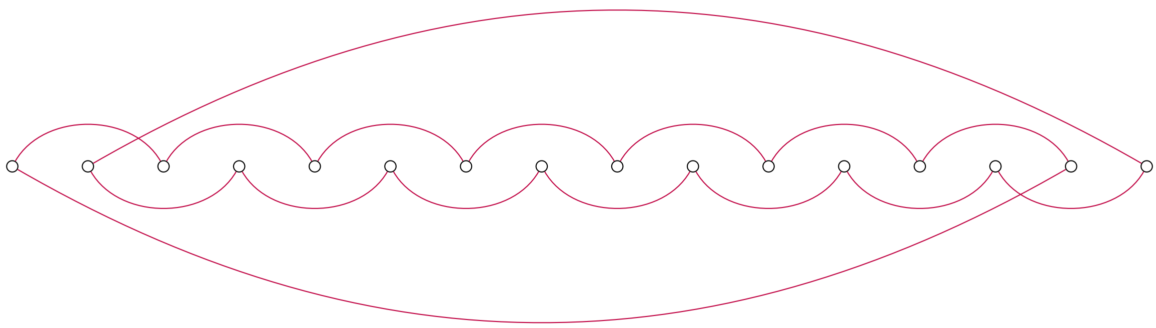
$$\begin{array}{ccccc} a_1 & a_2 & a_3 & & a_n \\ \circ & \circ & \circ & \cdots & \circ \end{array}$$

G_n

The Ehrenfeucht-Fraïssé game is played on A_n and B_n with n rounds. If the Spoiler selects a node a_i in A_n in the i-th round, the Duplicator responds by choosing the corresponding node b_i in B_n. The Duplicator has the following winning strategy is that in round $i \in [n]$, preserve all distances between choosen elements upto 2^{n-i}. Thus, A_n and B_n are indistinguishable by first-order logic with quantifier depth at most n. □

Proposition 3.2. *The class of connected graphs is not first-order definable.*

Proof. If A is a linear order of size n, then let G_A be the graph with n vertices and edges $\{i, i+2 \mod n\} \forall a_i \in A$.



Here, G_A is connected if and only if A is odd. If G_A is first order definable, then A is also first order definable. If φ is a first order formula that defines G_A, then replace each sub-formula $E(x, y)$ with $P(x, y)$ where $P(x, y)$ is a relation that x and y have a cyclic distance of 2 in the linear order A. Formally, $P(x, y) = \exists z(x < z < y \wedge \neq \exists(x < w < z)) \wedge \exists z(y < z < x \wedge \neq \exists(y < w < z))$. This new formula defines the oddness of A. Since, the parity is not first order definable, the class of connected graphs is also not first order definable. □

Proposition 3.3. *Reachability is not definable in first order logic.*

Proof. We will prove this by contradiction. Suppose that reachability is definable in first order logic. Let $\varphi(x, y)$, i.e. y is reachable from x, be a first order formula that defines reachability. Then $\forall x \forall y(\varphi(x, y))$ defines the class of connected graphs. However, the class of connected graphs is not first order definable. Thus, reachability is not definable in first order logic. □

Proposition 3.4. *Let us define natural numbers $\mathbb{N}$ from the Peano axioms restricted to* 0 *and the* Succ*. The relation $<$ is not first order definable in* $(\mathbb{N}, 0, \mathsf{Succ})$*.*

Proof. Assume that $<$ is first order definable in $\mathbb{N}$. Let $\varphi(x, y)$ be a first order formula that defines $x < y$. For all $n \in \mathbb{N}$, let $\mathcal{A}_n$ is a graph with 2^n vertices and edges $\{i, i+1\} \forall i \in [n-1]$. The graph $\mathcal{B}_n$ is a cyclic graph of size 2^n with edges $\{i, i+1 \mod 2^n\} \forall i \in [n-1]$. The Ehrenfeucht-Fraïssé game is played on $\mathcal{A}_n$ and $\mathcal{B}_n$ with n rounds. If the Spoiler selects a node a_i in $\mathcal{A}_n$ in the i-th round, the Duplicator responds by choosing the corresponding node $b_{i \mod 2^n}$ in $\mathcal{B}_n$. The Duplicator has the following winning strategy is that in round $i \in [n]$, preserve all distances between choosen elements upto 2^{n-i}. Thus, $\mathcal{A}_n$ and $\mathcal{B}_n$ are indistinguishable by first-order logic with quantifier depth at most n. □