Chapter 2 MODEL COMPARISON GAMES

When interpreted semantically, logical formulas express properties of situations of some appropriate type. Of course, different languages may have different expressive strength over the same class of situations. To measure this expressive strength, one must step outside the semantic format $\boldsymbol{M},\ s \models \phi$ with its single models, and look at powers of distinction between different models. For instance, a poor language with only "Yes" and "No" can distinguish very few moods, a rich language can distinguish a whole spectrum, like sadness, melancholy, and so on. Model comparison can be cast vividly as a game, played between a *Duplicator* **D** who claims that two given models **M**, **N** are similar, and a *Spoiler* **S** who claims the opposite: that they are different. In this chapter, we

- explain the workings of model comparison games for first-order logic,
- prove the adequacy of the method with respect to first-order equivalence
- analyze explicit correspondences between players' winning strategies and logical 'difference formulas' and 'potential isomorphisms'
- discuss some general game-theoretic aspects of the games
- show how to create variations and extensions for many languages
- identify some general themes that will return in the rest of these lectures

Model comparison games go back to Fraïssé 1954 and Ehrenfeucht 1957 – whence the widespread name 'Ehrenfeucht games' or 'Ehrenfeucht-Fraïssé games'. They have also become a tool for many languages in computer science (Thomas 1997). For a lucid modern introduction with a more mathematical slant, cf. Doets 1996.

2.1 Isomorphism and first-order equivalence

Expressive power and invariances The expressive power of a language shows in its power of distinction between different situations. Since the rise of the paradigm of transformations and invariants in 19th century geometry and its further repercussions in science, it has been possible to make precise mathematical sense of this. Two things are needed here. One is an independent semantic relation of structural *invariance* between models, the other a suitable *language* expressing properties of those models. The analysis works if it can be shown that the invariance matches just those differences between models 'which the language cannot detect'. About the most important structural invariance relation is the following.

<u>Definition</u> 2.1 Isomorphism

Two models ***M***, ***N*** are *isomorphic* if there exists a bijection *f* between the objects in their domains which preserves all the relevant structure on these: atomic properties, relations, distinguished objects, and operations. E.g., we want that

$R^M de$ iff $R^N f(d) f(e)$, for all binary predicates *R*, and objects *d, e* in ***M***
$f(G^M(d)) = G^N(f(d))$, for all unary functions *G*, and objects *d* in ***M***

The notation for isomorphism between models is $\boldsymbol{M} \cong \boldsymbol{N}$.

But there are also much coarser invariants which may be just the right comparison level for some other purpose. An earlier example was *bisimulation* between process models (cf. Section I.5), which measures just comparability of available action steps. This will be defined later in this chapter. A general discussion of possible invariances and logical definability in matching languages is found in van Benthem 1996, 2003.

First-order expressiveness To make all this more concrete, consider the expressive power of the language of first-order logic. First, we state a restriction:

For convenience, in this chapter, we will only use a vocabulary with finitely many predicate letters and individual constants.

Here is the obvious 'linguistic' notion of model comparison for this language.

<u>Definition</u> 2.2 Elementary equivalence

Two models ***M, N*** are *elementarily equivalent* if they both verify the same first-order sentences. The notation for this relation between models is $\boldsymbol{M} \equiv \boldsymbol{N}$.

How close are the structural and the linguistic notions of similarity between models? Here is one general implication:

<u>Proposition</u> 2.3 Isomorphism Lemma

For all models ***M, N***, if $\boldsymbol{M} \cong \boldsymbol{N}$, then $\boldsymbol{M} \equiv \boldsymbol{N}$.

<u>Proof</u> An easy induction on the construction of first-order formulas ϕ shows that, for all tuples of objects ***a*** in ***M***, and any isomorphism *f* sending the latter to ***N***:

$\boldsymbol{M} \models \phi[\boldsymbol{a}]$ iff $\boldsymbol{N} \models \phi[f(\boldsymbol{a})]$ ■

This result holds also for second-order logic, infinitary logic, and indeed, any well-behaved logical language. The converse is by no means true for first-order logic. What does hold is the following special case:

Proposition 2.4 Finite Harmony

For all *finite* models, the following two assertions are equivalent:

(a) ***M*** is isomorphic to ***N***

(b) ***M, N*** satisfy the same first-order sentences

Proof *From (a) ⇒ (b)* is the Isomorphism Lemma. *From (b) to (a).* If the language has a finite vocabulary as above, write a first-order sentence δ^{M} about ***M*** describing its *atomic diagram*. That is, if ***M*** has *k* objects, say there are different $x_1, .., x_k$ and no more objects, and after this quantifier prefix, enumerate all true atomic statements between these objects in ***M***, plus the negations of all the others. Since ***N***, too, satisfies the sentence δ^{M}, its domain can be enumerated in just the same pattern as ***M***. But then, the isomorphism between the two models is immediate. ■

This proof may be reworked to deal with an infinite vocabulary. But typically, it does not extend to *infinite* models. This has to do with expressive weaknesses of first-order logic. E.g., this language cannot define *finiteness* of domains of objects or related properties. Likewise, the first-order language cannot tell the rational numbers ***Q*** apart from the reals ***R*** in their natural ordering <. It cannot even distinguish between the natural numbers ***N*** and the intriguing model ***N+Z*** which continues after the naturals with one copy of the integers, the so-called 'supernatural numbers':

N	versus	***N+Z***
0, 1, 2, ….		**0, 1, 2, … … ∞–1, ∞, ∞+1, …**

Remark 2.5 Language relativity

With a *richer vocabulary*, a language may see differences that were invisible before. E.g., first-order logic does distinguish ***Q*** and ***R*** usinmg multiplication. E.g., $\sqrt{2}$ is irrational, so it only exists in the reals. This is expressed by the formula $\exists x: x \cdot x = 2$.

Incidentally, weak expressive power can also be a good thing, as it implies *transfer* of properties across different situations. In non-standard arithmetic, one computes in the structure ***N+Z*** using the infinite numbers to simplify calculations, and then transfers the outcome back to ***N***, provided it is a first-order statement about <.

2.2 Ehrenfeucht-Fraïssé comparison games

To bring out the fine-structure of the above invariance analysis, we need to play a certain type of logic games. These will work for any models, finite or not.

Playing the game Consider any two models ***M, N***. Player ***D*** (*Duplicator*) claims that ***M***, ***N*** are similar, while ***S*** (*Spoiler*) maintains that they are different. Players agree on some finite number *k* of rounds for the game, 'the severity of the probe'.

<u>Definition</u> 2.6 Comparison games
A *comparison game* works as follows, packing two moves into one round:

> ***S*** chooses one of the models, and picks an object *d* in its domain.
> ***D*** then chooses an object *e* in the other model, and the pair *(d, e)*
> is added to the current list of matched objects. At the end of the *k*
> rounds, the total object matching obtained is inspected. If this is
> a 'partial isomorphism', ***D*** wins; otherwise, ***S*** has won the game.

Here, a *partial isomorphism* is an injective partial function *f* between models ***M***, ***N***, which is an isomorphism between its own domain and range seen as submodels.

The alternating schedule of the form '$(\mathbf{DS})^*$' is typical for many games. Here are some possible runs for models with relations only, illustrating players' strategies. As before, in practice, players may play badly and lose, even when they have a winning strategy. We look at a first-order language with a binary relation symbol *R* only, mostly disregarding identity atoms with = for the sake of illustration.

<u>Example</u> 2. 7 Playing between graphs
Our first match is 'Pin' versus 'Dot'. We discuss one run and its implications.

In the first round, ***S*** chooses *a* in ***M***, and ***D*** must choose *c* in ***N***. If we stopped after one round, ***D*** would win. There is no detectable difference between single objects in these models. They are all irreflexive, and that's it. But now take a second round. Let

S choose b in ***M***. Then ***D*** must again choose c in ***N***. Now ***S*** wins, as the map *{(a, c), (b, c)}* is not a partial isomorphism. Clearly, the order does not match.

Our second match is between a '3-Cycle' and a '4 Cycle':

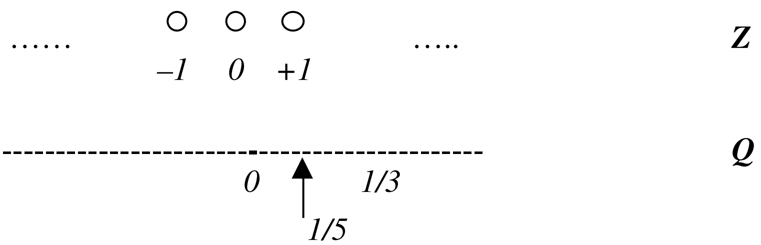
We just display a little table of one possible 'intelligent run':

Round 1	***S*** chooses *1* in ***M***	***D*** chooses *i* in ***N***
Round 2	***S*** chooses *2* in ***M***	***D*** chooses *j* in ***N***
Round 3	***S*** chooses *3* in ***M***	***D*** chooses *k* in ***N***

S wins, as *{(1, i), (2, j), (3, k)}* is not a partial isomorphism. But he can do better:

S has a winning strategy in two rounds, first picking *i* in ***N***, and then taking *k* in the next round. No such pattern occurs in ***M***, so ***D*** is bound to lose.

The third and final match is 'Integers' ***Z*** versus 'Rationals' ***Q***. These two linear orders have obviously different first-order properties: the latter is dense, the former discrete. The only question is how soon this will surface in the game:



By choosing his objects well, ***D*** has a winning strategy here for the game over two rounds. But ***S*** can always win the game in three rounds. Here is a typical play:

Round 1	***S*** chooses *0* in ***Z***	***D*** chooses *0* in ***Q***
Round 2	***S*** chooses *1* in ***Z***	***D*** chooses *1/3* in ***Q***
Round 3	***S*** chooses *1/5* in ***Q***	any response for ***D*** is losing

Difference formulas and Spoiler's strategies In actual play of these games, you will notice more detailed phenomena. Winning strategies for ***S*** are correlated with specific first-order formulas ϕ that bring out a difference between the models. And this correlation is tight. The quantifier syntax of ϕ triggers the moves for ***S***.

<u>Example</u> 2. 7, continued Exploiting definable differences

'Pin versus Point'. An obvious difference between the two in first-order logic is

$$\exists x\ \exists y\ Rxy$$

Two moves were used by **S** to exploit this, staying inside the model where it holds.

'3-Cycle versus 4-Cycle'. The first ***S***-play exploited the formula

$$\exists x\ \exists y\ \exists z\ (Rxy\ \&\ Ryz\ \&\ Rxz)$$

which is true only in ***M***, taking three rounds. The second play, which had only two rounds, used the following first-order formula, which is true only in the model ***N***:

$$\exists x\ \exists y\ (\neg\ Rxy\ \&\ \neg Ryx\ \&\ \neg x{=}y)$$

'Integers versus Rationals'. ***S*** might use the definition of density for a binary order

$$\forall x \forall y\ (x<y \rightarrow \exists z(\ x<z\ \&\ z<y))$$

to distinguish ***Q*** from ***Z***. We spell this out, to show how the earlier spontaneous play for this example has an almost algorithmic derivation from a first-order difference formula. For convenience, we use density in a form with existential quantifiers only. The idea is for ***S*** to maintain a difference between the two models, of stepwise decreasing syntactic depth. ***S*** starts by observing that

$\exists x \exists y\ (x<y\ \ \&\ \neg\ \exists z(x<z\ \&\ z<y))$ is true in ***Z***, but false in ***Q*** #

He then chooses an integer witness d for $\exists x$, making $\exists y\ (d<y\ \&\ \neg \exists z(d<z\ \&\ z<y))$ true in ***Z***. ***D*** can then take any object d' she likes in ***Q***: $\exists y(d'<y\ \&\ \neg \exists z(d'<z\ \& z<y))$ will always be false for it, by #:

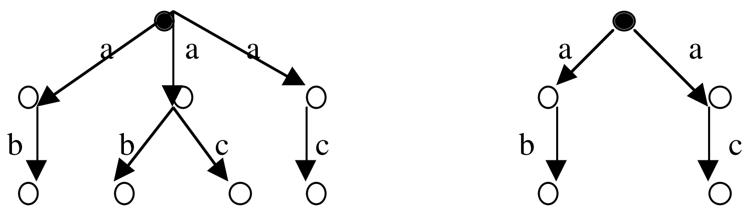
$$\boldsymbol{Z} \models \exists y(d<y\ \&\ \neg \exists z(d<z\ \&\ z<y))\ ,\ \ not\ \boldsymbol{Q} \models \exists y(d'<y\ \&\ \neg \exists z(d'<z\ \&\ z<y))$$

In the second round, ***S*** continues with a witness e for the new outermost quantifier $\exists y$ in the true existential formula in ***Z***: making $d<e\ \&\ \neg \exists z(d<z\ \&\ z<e)$ true there. Again, whatever object e' ***D*** now picks in ***Q***, the formula $d'<e'\ \&\ \neg \exists z(d'<z\ \&\ z<e')$ is false there. In the third round, ***S*** analyses the mismatch in truth value. If ***D*** kept

d'<'e true in ***Q***, then, as *¬∃z(d<z & z<e)* holds in ***Z***, *∃z(d'<z & z<e')* holds in ***Q***.
S then *switches* to ***Q***, chooses a witness for the existential formula, and wins.

Thus, even the right model switches for ***S*** are encoded in the difference formulas. These are mandatory whenever there is a syntactic switch in 'polarity' from one outermost quantifier to a lower one. With the first two finite model pairs, of course, the finite model descriptions of Proposition 2.4 will also define winning strategies for Spoiler. But the latter are clumsy, and may take more rounds than needed.

Excursion 2.8 Wrong and right syntactic measures
Our examples may suggest the correlation: 'winning strategy for ***S*** over *n* rounds' ~ 'difference formula with *n* quantifiers altogether'. But that is not the right measure:



The difference between these two models may be expressed by means of

$$\exists y\ (R_a xy\ \&\ (\exists z\ R_b yz\ \&\ \exists z\ R_c yz))$$

But it can be shown that two-quantifier formulas do not distinguish these situations. The correct syntactic correlation for the number of rounds needed to win is syntactic *quantifier depth*, being *the maximum length of a quantifier nesting in a formula.*

2.3 Adequacy and strategies

As with evaluation games, the interesting information about model comparison is in players' strategies. Here is what lies behind the preceding observations. In the result to follow, it is easier to think of winning strategies for Duplicator – though Spoiler's point of view on the situation will certainly return later. Let us write

*WIN (**D**, **M**, **N**, k)* ***D*** has a *winning strategy* against ***S*** in the *k*-round comparison game between the models ***M*** and ***N***.

Comparison games can start from any given 'handicap', i.e., an initial matching of objects in ***M*** and ***N***. In particular, if models have distinguished objects named by individual constants, then these must be matched *automatically* at the start of the game. In the proofs to follow, for convenience, we will think of all 'initial matches' in the latter way. Now here is the analogue of the Success Lemma in Chapter 1.

<u>Theorem</u> 2.9 Adequacy Theorem

For all models ***M, N***, and $k \in \mathbf{N}$, the following two assertions are equivalent:

(a) *WIN (**D, M, N**, k)* : ***D*** has a winning strategy in the *k*-round game

(b) ***M, N*** agree on all first-order sentences up to quantifier depth *k*.

This improves on the Isomorphism Lemma in two ways. The Adequacy Theorem matches up a language-dependent and a language-independent comparison relation. And it provides fine-structure not available before, which helps in applications.

<u>Proof</u> The direction from (a) to (b) is an induction on *k*. *Base step.* With 0 rounds, the initial match between distinguished objects must have been a partial isomorphism for ***D*** to win. This implies that ***M, N*** agree on all atomic sentences, and so they agree also on the latter's Boolean combinations, i.e., precisely the formulas of quantifier depth 0. *Inductive step.* The inductive hypothesis says that, *for any two models*, if ***D*** can win their comparison game over *k* rounds, then the models agree on all first-order sentences up to quantifier depth *k*. Now assume that, for some models ***M, N***, ***D*** has a winning strategy for the game over $k+1$ rounds. Consider any first-order sentence ϕ of quantifier depth $k+1$. Clearly, looking outside in at its syntax, any such sentence must be equivalent to one of the form

a Boolean combination of (i) atoms, (ii) sentences of
the form $\exists x\ \psi$, with ψ of quantifier depth at most *k*.

Thus, it suffices to show that ***M, N*** agree on the latter forms. (They do on the atoms: as ***D*** can certainly win over 0 rounds). So let $\boldsymbol{M} \models \exists x\ \psi$. Then for some object *d*, $\boldsymbol{M}, d \models \psi(x)$. For convenience, we now think of *(**M**, d)* as an expanded model with a distinguished object *d*, to which we assign a new proper name <u>*n*</u> in our language. We can say then that *(**M**, d)* verifies the sentence $\psi(\underline{n})$. Now, ***D***'s given winning strategy has a response for whatever ***S*** might do in the $k+1$-round game. In particular, let ***S*** start by selecting ***M*** and object *d* in it. Then ***D*** has a response *e* in ***N*** such that her remaining strategy still gives her a win in the *k*-round game played from the given link *d – e*. Equivalently, consider the expanded model *(**N**, e),* with *e* as its interpretation of the proper name n. The remainder is then an ordinary k-round game starting from the expanded models *(**M**, d)* and *(**N**, e)*. By the inductive hypothesis, these models agree on all sentences up to quantifier depth k: and hence also on $\psi(\underline{n})$. Therefore, $N, e \models \psi(\underline{n})$, and hence $N \models \exists x\ \psi$. The core of this argument is a break-down of an existential formula into "one object choice" + "body of the formula", and the matching break-down of a strategy into "first move" + "the remaining strategy".

Next, the direction from (b) to (a) requires another induction on k. This time we need a small auxiliary result about first-order logic in a finite relational vocabulary.

Lemma 2.10 Finiteness Lemma

Fix any set $x_1, .., x_m$. Up to logical equivalence, with these free variables, there are only finitely many first-order formulas of quantifier depth $\leq k$.

Proof The argument for this is again by induction on k, using the preceding analysis of formulas of quantifier depth $k+1$, plus the fact that the Boolean combinations of any finite set of formulas form a finite set modulo logical equivalence.

Now we can do the inductive proof from (b) to (a). The *base step* is trivial as before: 'doing nothing' is a winning strategy for ***D***. As for the *inductive step*, we describe the first move in ***D***' s strategy. Let ***S*** choose one of the models, say ***M***, plus some object d in it. Now, ***D*** looks at the set of first-order formulas that hold of d in ***M*** – which may refer also to all distinguished objects available through their names in the language. This set is finite modulo logical equivalence, by Lemma 2.9, and hence one existential formula $\exists x\ \psi^d$ true in ***M*** summarizes all this information. As ***M**, **N*** agree on all first-order sentences of depth $k+1$, and $\exists x\ \psi^d$ is such a sentence, it also holds in ***N***. So, ***D*** can choose a witness e for it in ***N***. Then the expanded models *(**M**, d), (**N**, e)* agree on all sentences up to quantifier depth k, and by the inductive hypothesis, ***D*** has a winning strategy in the remaining k-round game between them. Her initial response plus the latter gives her over-all strategy over $k+1$ rounds. ■

2.4 An explicit version: the logic content of strategies

The Adequacy Theorem as stated still leaves out the most exciting feature of our original examples, viz. the precise correlation between Spoiler's winning strategies and the structure of first-order difference formulas. Thus, it displays a common disease of many formulations of results in logic, which may be called

'∃-sickness'

This affliction consists in hiding crucial information under an existential quantifier. Sure symptoms are the use of indefinite articles "a", or affixes like "-ility". We see this with phrases like "*having a* strategy" ("winnab–ility") in earlier results. Another case of ∃-sickness is how a standard completeness theorem relates *provability* to validity, instead of a more informative match from *proofs* to semantic 'verifications'. More domestically, e.g., temporal logic reads the past tense in "Lida fell down the stairs" as "at *some time* in the past", whereas we usually have one particular episode in mind. Fortunately, this disease can often be cured! For instance, properly viewed, the proof of the Adequacy Theorem really provides more concrete information:

Theorem 2.11 Adequacy Explicitized

There exists an *explicit correspondence* between

(a) winning strategies for $\boldsymbol{S}$ in the k-round comparison game for $\boldsymbol{M}, \boldsymbol{N}$

(b) first-order sentences ϕ of quantifier depth k with $\boldsymbol{M} \models \phi$, *not* $\boldsymbol{N} \models \phi$

Proof *From (b) to (a).* As was illustrated in Example 2.7, every formula ϕ of quantifier depth k defines a uniform winning strategy for Spoiler in a k-round game between arbitrary models. Each round $k–m$ starts with a match between linked objects chosen so far which differ on some subformula ψ of ϕ with quantifier depth $k–m$. By Boolean analysis, $\boldsymbol{S}$ then finds some existential subformula $\exists x \bullet \alpha$ of ψ with a matrix formula α of quantifier depth $k–m–1$ on which the two models disagree. $\boldsymbol{S}$'s next choice is a witness in that model of the two where $\exists x \bullet \alpha$ holds.

From (a) to (b). Each winning strategy σ for Spoiler induces a distinguishing formula of proper quantifier depth. To obtain this, let $\boldsymbol{S}$ make his first choice d in model $\boldsymbol{M}$ according to σ – and write down an existential quantifier for that object. Our formula will be true in $\boldsymbol{M}$, and false in $\boldsymbol{N}$. We know that each choice of Duplicator for a corresponding object e in $\boldsymbol{N}$ gives a winning position for $\boldsymbol{S}$ in all remaining $k–1$-round games starting from an initial match $d–e$. By the inductive hypothesis, these induce distinguishing formulas of depth $k–1$. By the Finiteness Lemma, only finitely many such formulas are available. Some of these will start with 'their' first quantifier in $\boldsymbol{M}$ (say $A_1, .., A_r$) – others in $\boldsymbol{N}$ (say $B_1, .., B_s$). The total distinguishing formula for strategy σ is then the $\boldsymbol{M}$-true assertion

$$\exists x \bullet (A_1 \ \& \ .. \ \& \ A_r \ \& \ \neg B_1 \ \& \ ... \ \& \ \neg B_s) \qquad \blacksquare$$

Thus, Spoiler's winning strategies in a comparison game correspond to formulas, logical objects of independent interest. A similar match exists for Duplicator, but it is a bit harder to say what logical 'objects' correspond to her winning strategies. One might call them 'analogies', of some finite quality measured by the number k. Technically, they are cut-off versions of the *potential isomorphisms* in Section 2.6.

Caveat The formulation of the new theorem is still ∃-sick! Can you cure it?

Finally, 'explicitness' is not the same as *effectiveness*. The above gives a definable matching of strategies with other objects, which need not be computable in an effective sense. And while we are at it, the strategies themselves employed in evaluation or comparison games need not be effective. They come in many degrees of difficulty: from simple 'history-free' (with all next moves read off from the current game state), to dependence on a complete record of the game so far. The strategic invariants in the next section give concrete examples of the kind of memory to be maintained. Chapter 5 has further details on the complexity of logic games.

2.5 Using comparison games in practice

In practice, comparison games involve not just logic, but also hard work: non-trivial combinatorial analysis of the models involved. Here are some positive examples.

Fact 2.12

The rationals $(\boldsymbol{Q}, <)$ are elementarily equivalent with the reals $(\boldsymbol{R}, <)$

To prove this, it suffices to show that $\boldsymbol{D}$ can win the comparison game for every k. One useful method for this is to identify a suitable *invariant for **D** to maintain* at all intermediate game states. In this particular case, she just needs to make sure that all matches so far from a *finite partial isomorphism*. Further choices of $\boldsymbol{S}$ can always be countered, using the two-way unboundedness and the density of the two orderings. More complicated invariants may depend on the number of rounds to go. Recall that

Fact 2.13

$(\boldsymbol{N}, <)$ is elementarily equivalent with $(\boldsymbol{N}+\boldsymbol{Z}, <)$

Again, $\boldsymbol{D}$ must be able to win all k-round comparison games. This time, at least if the length of the game is known in advance, $\boldsymbol{D}$ can still counter nasty choices of $\boldsymbol{S}$ from the supernatural numbers by matching them with large enough natural numbers in $\boldsymbol{N}$.

The precise invariant involves maintaining suitable distances between objects, measured by an exponential in the length of the remaining game:

Duplicator must make sure, that with k rounds to go, the two sequences $d_1, \ldots, d_m$ in $\boldsymbol{N}$ and $e_1, .., e_m$ in $\boldsymbol{N+Z}$ chosen so far have the properties:

(a) $d_i < d_j$ iff $e_I < e_j$

(b) if d_i , d_j have distance $< 2^k - 1$, then $distance(e_i , e_j)$ is the same; else, d_i , d_j and e_i , e_j both have distance $\geq 2^k - 1$ (finite or infinite).

Invariants may be viewed as perspicuous decidable descriptions of those positions from which a player has a winning strategy – which were only given abstractly in the proofs of Chapter 1 for the Zermelo and Gale-Stewart Theorems. More abstract general descriptions of invariants will be found in the game logics of Chapters 6, 7.

Comparison games also work on model classes where standard properties of first-order logic fail – such as the compactness theorem. A well-known example are the *finite models*. Here is an example of a 'negative' use of games. We know that

<u>Fact</u> 2.14

"Even" or "odd" are not first-order definable as domain sizes.

The usual proof for this non-definability uses a compactness argument. But the same fact holds restricted to the universe of finite models. This may be shown as follows. Suppose that "even size" had a first-order definition on finite models of quantifier depth k. Then any two finite models for which $\boldsymbol{D}$ can win the k-round comparison game are both of even size, or both of odd size. But this can be refuted simply by looking at any two finite models with k versus $k+1$ objects in their domains.

2.6 Determinacy, finite and infinite games

Comparison games as defined so far are two-player zero-sum games of some finite depth k. Thus, *Zermelo's Theorem* still applies: either $\boldsymbol{D}$ or $\boldsymbol{S}$ has a winning strategy:

<u>Fact</u> 2.15

Model comparison games are *determined*.

But these games also have a natural version which goes on *forever*, say over ω rounds. This is the preferred interpretation when we think of games as establishing some kind of desired behaviour, such as a server's granting wishes to clients, or the recurrence of spring. One natural winning convention for this extension is:

> ***D*** wins in case she does not lose at any finite stage,
> i.e., she can maintain a partial isomorphism all the time.

This is like a long-term safety property for an operating system. Note that this is stronger than being able to win all finite games. E.g., the earlier models ***N*** and ***N+Z*** *can* be distinguished by Spoiler in an infinite game: he should just start with a supernatural number and keep descending all the way… On the other hand, comparing ***Q*** with ***R***, Duplicator could hold out indefinitely. For infinite games, ***D***'s winning strategies do correspond to a logical notion with an independent origin:

Definition 2.16 Potential isomorphism
A *potential isomorphism* between models ***M, N*** is a non-empty family *I* of finite partial isomorphisms between ***M, N*** with the following Back & Forth Property:

(a) if $f \in I$ and $a \in \boldsymbol{M}$, then there exists a $b \in \boldsymbol{N}$ such that $f \cup \{(a, b)\} \in I$

(b) and the same must happen vice versa.

Following our earlier analysis, the following correspondence is easy to see:

Proposition 2.17
> The potential isomorphisms between two models correspond with Duplicator's winning strategies in their infinite comparison game.

Potential isomorphism implies elementary equivalence. If ***D*** can win the never-ending game, she can win every finite-length one, and by the Adequacy Theorem, the models must satisfy the same first-order sentences. But the pair ***N*** versus ***N+Z*** refuted the converse. It is also easy to show directly that the partial isomorphisms in a potential isomorphism *I* satisfy the same first-order formulas – and even all those in the extended language allowing *infinite* conjunctions and disjunctions. Here is the true language–simulation match:

Proposition 2.18
> Two models are potentially isomorphic if and only if they satisfy the same sentences in infinitary first-order logic.

More information on infinitary first-order logic and its games can be found in Barwise & van Benthem 1999. Finally, the infinite comparison game still falls under earlier results. Since the winning set for ***S*** is *open* (failure of partial isomorphism occurs by some finite stage, and then persists), the Gale-Stewart Theorem applies:

Fact 2.19 The infinite comparison game is determined, too.

This time, it is Spoiler's winning strategies which lack an established counterpart. They are ways of blocking each attempt at establishing potential isomorphism at some finite stage – but they need not be finite or effective methods for doing this.

2.7 Modifications and extensions

As with evaluation games, one attraction of comparison games is their flexibility. By varying the rules, one can investigate and characterize a wide variety of languages.

No switches One can restrict ***S*** to choices *in one model*. This measures equivalence with respect to purely *universal* first-order sentences. For instance, using such a clipped game, one can show that the universal first-order theories of the discrete linear order ***Z*** and the dense linear order ***Q*** are the same.

Pebble games To make memory a concern, one can let objects be brought into play by using a finite 'resource' that has been supplied to the players, namely marking them with one of a finite set of *pebbles* (cf. Immerman & Kozen 1987). An easy modification of the earlier adequacy arguments shows that

Fact 2.20 Adequacy for pebble games

*WIN(**D**, **M**, **N**, k, m),* with *m* the number of pebble pairs, holds iff
M**, **N agree on all first-order sentences of quantifier depth $\leq k$
which use at most the variables $x_1, .., x_m$ *(free or bound).*

An important motivation for this parametrization of the games is as follows. 3 pebbles suffice for winning all comparison games over *linear orders* with monadic predicates. The sublanguages of first-order logic involved, with just some finite set of variables available, free or bound, are called *finite-variable fragments*. They have been used – amongst other purposes – in computer science to 'calibrate' the time-complexity of answering first-order definable queries in finite data bases.

Example 2.21 Defining with few variables
Stating that there exist four objects in a row is usually written as

$$\exists x\, \exists y\, \exists z\, \exists u\ \ (x<y\ \&\ y<z\ \&\ z<u).$$

But in fact, even *two* variables would do the job: $\exists x\, (\exists y\, y<x\ \&\ \exists y\, x<y\ \&\ \exists x\, y<x).$

Modal logic and guarded choices Another important modification of the game is *restricted selection of objects*, e.g., to relational successors of objects already matched. This leads to comparison games for *modal logic* (Blackburn, de Rijke & Venema 2001) and the *guarded fragment* of first-order logic (Andréka, van Benthem & Németi, 1998) The expressive power of the modal language is measured by a notion of structural invariance already mentioned in the Introduction (Section I.5):

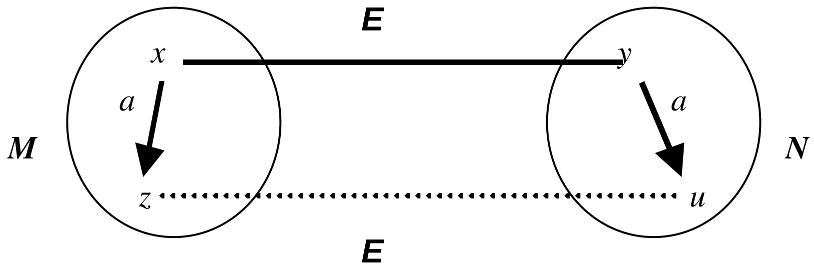
Definition 2.22 Bisimulation

A *bisimulation* is a binary relation **E** between states of models ***M, N*** with binary transition relations R_a, such that, whenever x **E** y, then we have 'atomic harmony', plus two-way zigzag clauses for all relations a:

(a) x, y verify the same proposition letters

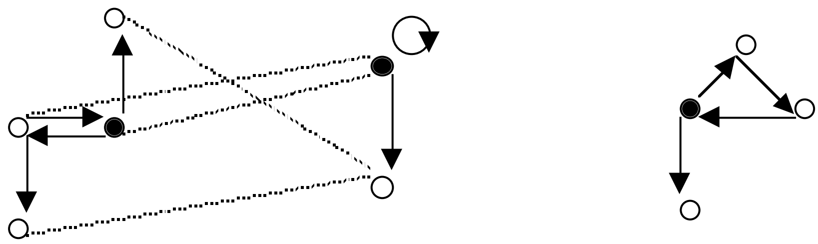
(2a) if $x R_a z$, then there exists u in ***N*** with $y R_a u$ and z **E** u

(2b) vice versa. ■



Example 2.23 Bisimulation between process graphs

The two black states in ***M, N*** are connected by the bisimulation given by the dotted lines – but no bisimulation includes a match between the black worlds in ***N*** and ***K***:



M ***N*** ***K***

Typically, modal formulas are *invariant for bisimulation:*

Proposition 2.24 Invariance Lemma

If **E** is a bisimulation between two graphs ***M*** and ***N***,
and m **E** n, then m, n satisfy the same modal formulas.

Thus, we can see the failure of bisimulation in Example 2.22 by noting that the model in the middle satisfies the modal formula $\langle\rangle\langle\rangle[]\bot$ in its root, whereas the one on the right does not. General features are just as for first-order logic in Section 2.6. Proposition 2.23 becomes an equivalence for a modal language with *arbitrary infinite* conjunctions and disjunctions – and for the plain modal language over *finite* models. In Chapters 6, 12 we will even need stronger definability results like the following:

<u>Proposition</u> 2.25 Definability Lemma

For any model (***M**, s)* with a designated state *s*, there is an infinitary modal formula $\phi^{M, s}$ true in exactly those models (***N**, t)* which are bisimilar to (***M**, s)*: i.e., some bisimulation between ***M***, ***N*** links *t* to *s*.

A proof for this result may be found in Barwise & Moss 1996.

The fine-structure of bisimulation suggests games between Duplicator and Spoiler, comparing successive pairs *(m, n)* in two models ***M**, **N***:

> In each round Spoiler chooses a state *x* in one model which is a successor of the current *m* or *n*, and Duplicator responds with a matching successor *y* in the other model. If *x, y* differ in their atomic properties, Spoiler wins – if Duplicator cannot find a matching successor: likewise.

<u>Theorem</u> 2.26 Adequacy for modal games

(a) Spoiler's winning strategies in a *k*-round game between *(**M**, s), (**N**, t)* match the modal formulas of *operator depth k* on which *s, t* disagree.

(b) Duplicator's winning strategies over an *infinite* round game between *(**M**, s), (**N**, t)* match the bisimulations between them linking *s* to *t*.

Here are some illustrations matching those given earlier for first-order logic.

<u>Example</u> 2.27

Spoiler can win the game between the following models from their roots. He needs two rounds – and different strategies do the job. One stays on the left, exploiting the difference formula $\langle a\rangle(\langle b\rangle T \wedge \langle c\rangle T)$ of depth 2, with three existential modalities. Another winning strategy switches models, but it needs a smaller formula $[a]\langle b\rangle$T:

In the non-bisimulation pair ***N, K*** of Example 2.22, starting from a match between the two black worlds, Spoiler needs 3 rounds to win: forcing Duplicator in two rounds into a match where one world has no successor, while the other does. One winning strategy for this exploits the earlier modal difference formula $\langle\rangle\langle\rangle[]\bot$. ■

As modal logic is our lingua franca, the preceding observations return in Chapters 5 on complexity analysis of logical tasks, and the game logics of Chapters 6, 7, 10. In particular, bisimulation suggests several notions of game equivalence. Finally, we will meet modal bisimulation games with a geometric flavor in Chapter 7, as a way of testing equivalence of players' powers for determining outcomes across two games.

Other languages One can also design comparison games for other languages, such as first-order logic with generalized quantifiers (Westerståhl 1989), or first-order logic with fixed-point operators. Sometimes, this raises interesting open questions. E.g., the comparison game for first-order fixed-point logic in Ebbinghaus & Flum 1995 is a direct transcription of the obvious finite game for the standard second-order definition of fixed-points. It is not known if there is also a nice model comparison analogue of the elegant infinite evaluation games defined in Chapter 1.

2.8 Connections between logic games

Several general game-theoretic themes arising from evaluation in Chapter 1 return with model comparison, though in a different guise. We conclude this chapter with three technical observations that firm up connections between the two sorts of game. The reader can skip this section without loss of continuity.

Game operations The operations of choice, switch, and composition found inside first-order evaluation games may be seen at work inside model comparison games. But they do not seem to make much sense as external operations combining these into new ones. Nevertheless, comparison games suggest new global operations of their own, including parallel compositions. As we will show in Chapter 5,

model comparison games are interleaved evaluation games.

Another source of operations in this vein is the Stage-Comparison Theorem for inductive definitions (Moschovakis 1974) which involves combining two evaluation games by switching between their models, and seeing which one ends first.

Model comparison as evaluation Testing for similarity reduces sometimes to evaluating some logical formula. E.g., through their definition, bisimulations $\boldsymbol{E}$ are non-empty *greatest fixed-points* for a first-order operator on binary relations between models ***M, N***. More precisely, they satisfy a fixed-point equation

$$\boldsymbol{E}\,xy \quad \leftrightarrow \quad (Px \leftrightarrow Py)\ \&\ \forall z\,(R_a xz \rightarrow \exists u\,(R_a yu\ \&\ \boldsymbol{E}\,zu))$$
$$\&\ \forall u\,(R_a yu \rightarrow \exists z\,(R_a xz\ \&\ \boldsymbol{E}\,zu)),$$

with the right-hand side a conjunction over all relevant atomic predicates P and atomic actions R_a . Thus, existence of a bisimulation between two states s–t amounts to the truth of some formula in a *first-order fixed-point language* over the disjoint sum of ***M, N***. This can be checked by a fixed-point evaluation game as in Section 1.6. The latter could be *infinite* – but then, so can model comparison games testing for the existence of a complete bisimulation. This connection between comparison and evaluation games hold more generally. Another example are so-called k-bisimulations appropriate to *k-variable fragments* of first-order logic (cf. Section 2.7 on pebble games, as well as Barwise 1975, van Benthem 1991).

Game equivalence The logical literature often switches between what are called 'equivalent' formulations of games, even though the precise equivalence is left open. Here is how Barwise & van Benthem 1999 define infinite model comparison games. They start with one finite partial isomorphism between two models. Each round then inverts the earlier schedule, letting ***D*** choose some family F of partial isomorphisms, followed by a selection by ***S*** of one f in F. In the next round, ***D*** must select a set F^+ again, satisfying the following back-and-forth property:

> for every object a in one model, there exists an object b in the other
> such that $f \cup \{(a, b)\} \in F^+$ – and likewise in the other direction.

S then chooses a partial isomorphism in F^+ again, and so on. Why is this equivalent to standard comparison games? Well, in each round ***D*** offers ***S*** a complete panorama of all choices he could make, plus her own responses to them. ***S*** makes a choice of his own move plus ***D***'s pre-packaged response – thereby setting the new stage.

A technical analogy here is with transformation into Skolem form, as in Section 1.5. In human terms, ***D*** behaves like an esteemed colleague of mine at Amsterdam, who at department meetings prevents you from speaking by saying: "Now you're going to say *A*, and I will say *B* – or, you're going to say *C*, and then I will say *D* – etcetera."

References

H. Andréka, J. van Benthem & I. Németi, 1998, 'Modal Logics and Bounded Fragments of Predicate Logic', *Journal of Philosophic Logic* 27:3, 217–274.

P. Blackburn, M. de Rijke & Y. Venema, 2001, *Modal Logic*, Cambridge University Press, Cambridge.

J. Barwise,1975, *Admissible Sets and Structures*, Springer, Berlin.

J. Barwise & J. van Benthem,1999, 'Interpolation, Preservation, and Pebble Games', *Journal of Symbolic Logic* 64:2, 881–903.

J. Barwise & L. Moss, 1996, *Vicious Circles, on the mathematics of non-well-founded phenomena*, CSLI Publications, Stanford.

J. van Benthem, 1991, *Language in Action: categories, lambdas and dynamic logic*, North-Holland, Amsterdam.

J. van Benthem 1996, *Exploring Logical Dynamics*, CSLI Publications, Stanford.

J. van Benthem, 2003, 'Logical Constants, the Variable Fortunes of an Elusive Notion', in C. Talcott et al., eds., *Essays In Honour of Sol Feferman*, AMC Publications, Providence.

H.C. Doets 1996,1996, *Basic Model Theory*, CSLI Publications, Stanford.

H-D Ebbinghaus & J. Flum, 1995, *Finite Model Theory*, Springer, Berlin.

A. Ehrenfeucht,1961, 'An Application of Games to the Completeness Problem for Formalized Theories', *Fundamenta Mathematicae* 49, 129–141.

R. Fraïssé,1954, 'Sur Quelques Classifications des Systèmes de Relations', *Publications des Sciences de l'Université d' Algérie*, Série A, 1, 35–182.

N. Immerman & D. Kozen, 1987, 'Definability With Bounded Number of Bound Variables', *Proceedings 2d IEEE Symposium on Logic in Computer Science*, 236-244.

Y. Moschovakis, 1974, *Elementary Induction on Abstract Structures*, North-Holland, Amsterdam.

W. Thomas, 1997, 'Ehrenfeucht games, the composition method, and the monadic theory of ordinal words', In J. Mycielski et al., eds, *Structures in Logic and Computer Science*, A Selection of Essays in Honor of A. Ehrenfeucht, Springer Lecture Notes in CS 1261, Berlin, 118-143.