Logic Games

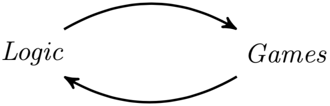
Tu Zeng

Department of Philosophy, PKU

Dec, 17th, 2019

game logic VS logic game

- Game logic: capture essential aspects of game structure, reasoning about, or inside, games.
- Logic as game: the study of logic by means of games.



Dual view

Moreover, the cycles invite spinning round in a spiral, or a carousel, and one can look at game logics via associated logic games, or at logic games in terms of matching game logics. This lecture focuses on logic as game.

Agenda

- Some basics of game theory
- Evaluation Game ($M, s \vDash \phi$?)
- Modal comparison game (Is M similar as N?)
- Modal building game($\exists M(M \vDash \phi)$)
- Dialogue game (ϕ?)

Normal-form game

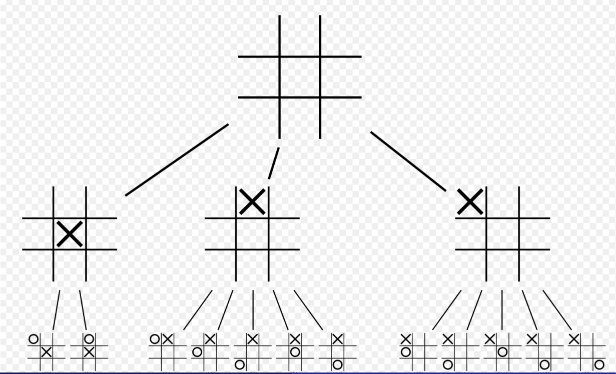
- Normal-form game represent the game by matrix. The below matrix demonstrates the unique Nash equilibrium of this game is (*Defect*, *Defect*).

Player 2 / Player 1	***Cooperate***	***Defect***
Cooperate	–1, –1	–5, 0
Defect	0, –5	–2, –2

The Prisoner's Dilemma

Extensive form game

Extensive form game uses a complete game tree to represent a game.
Let's see what is a game tree:
In game theory, a game tree is a directed graph whose nodes are positions in a game and whose edges are moves. The complete game tree for a game is the game tree starting at the initial position and containing all possible moves from each position.



extensive game

An extensive game G is a complete tree with auxiliary information. We express it by a tuple (I, A, H, t).
(a)I: a set I of players;
(b)A: a set of actions or moves(edges of tree);
(c)H: a set of sequences of successive actions from A(histories);(path of tree)
(d)t: a turn function t mapping each non-terminal history having a proper continuation in H to a unique player whose turn it is;

Turn-based 2-player games: basics

- **Sum-0**: In game theory and economic theory, a zero-sum game is a mathematical representation of a situation in which each participant's gain or loss of utility is exactly balanced by the losses or gains of the utility of the other participants. If the total gains of the participants are added up and the total losses are subtracted, they will sum to zero. Thus, cutting a cake, where taking a larger piece reduces the amount of cake available for others, is a zero-sum game if all participants value each unit of cake equally(if one gains, another loses).

Some notion

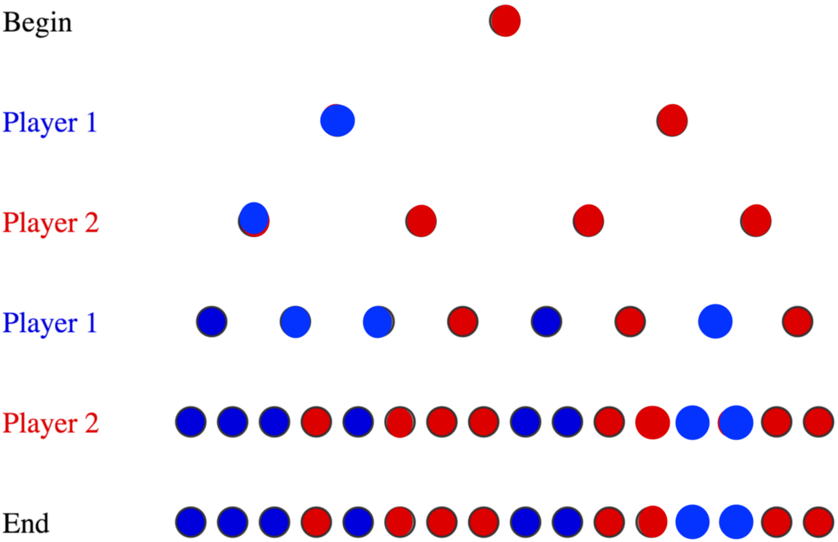
- **Perfect information game**:if each player, when making any decision, is perfectly informed of all the events that have previously occurred.
- **Strategy**:a rule (function) prescribing for each play, such that the given player is to move from the last configuration, a legitimate move for that player.
- **Winning strategy**: a strategy that guarantees the winning condition for the player to be satisfied by every play.
- **Determinacy**: a game in which one of the players has a non-losing strategy.

Theorem in game theory

- **Theorem**: If the game cannot end in a draw,for all zero-sum two-player games of fixed finite depth are determined.
- *Proof* We provide a simple bottom-up algorithm determining the player having the winning strategy at any given node of a game tree of this finite sort. First, color those end nodes black that are wins for player A, and color the other end nodes white, being the wins for E. Then extend this coloring stepwise as follows. If all children of node n have been colored already, do one of the following:
  (a) If player A is to move, and at least one child is black: color n black; if all children are white, color n white,
  (b) If player E is to move, and at least one child is white: color n white; if all children are black, color n black.

Zermelo theorem

This procedure colors all nodes black where player A has a winning strategy, while coloring those where E has a winning strategy white. The key to the adequacy of the coloring can be proved by induction: a player has a winning strategy at a turn iff this player can make a move to at least one daughter node where there is again a winning strategy.



Begin
Player 1
Player 2
Player 1
Player 2
End

Logic games

- Evaluation games: Verifier (V) and Falsifier (F). $M, s \vDash \phi$?
- Model Comparison games: Duplicator (D) and Spoiler (S). M similiar as N ?
- Model Construction games: Builder (B) and Critic (C). $\exists M(M \vDash \phi)$?
- Argumentation games:Proponent (P) and Opponent (O). is $A_1, \ldots, A_k \rightarrow B$ valid?

Evaluation game for predicate logic

Definition

game(ϕ, M, s):Two parties disagree about a proposition ϕ in some situation M,s, where s is an assignment of the variables in ϕ: verifier V claims that it is true, falsifier F that it is false.

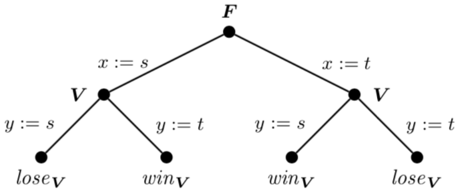
Rules

atoms $Pd, Rde, \ldots$	$\boldsymbol{V}$ wins if the atom is true, $\boldsymbol{F}$ if it is false
disjunction $\varphi \vee \psi$	$\boldsymbol{V}$ chooses which disjunct to play
conjunction $\varphi \wedge \psi$	$\boldsymbol{F}$ chooses which conjunct to play
negation $\neg\varphi$	role switch between the two players, play continues with respect to φ

Next, the quantifiers make players look inside $\boldsymbol{M}$'s domain of objects:

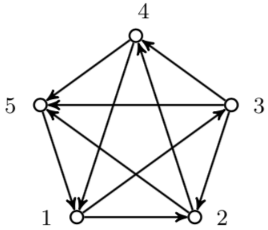
existential $\exists x \varphi(x)$	$\boldsymbol{V}$ picks an object d, play continues with $\varphi(d)$
universal $\forall x \varphi(x)$	the same move, but now for $\boldsymbol{F}$

- Example: Formulas and schedule of play
  Consider a model M with two objects s,t. Here is a game for $\forall x \exists y x \neq y$, pictured as a tree of moves, with the scheduling from top to bottom:



  We interpret this as a game of perfect information: players know throughout what has happened. Falsifier starts, and verifier must respond. There are four possible plays, with two wins for each player. But verifier has a winning strategy, in the standard sense of our earlier chapters.

Example (Notes all the nodes are self-loops).



The formula $\forall x \forall y(Rxy \vee \exists z(Rxz \wedge Rzy))$ says that every two nodes in this network can communicate in at most two steps. Here is a run of the evaluation game:

player	*move*	*next formula*
$\boldsymbol{F}$	picks 2	$\forall y(R\boldsymbol{2}y \vee \exists z(R\boldsymbol{2}z \wedge Rzy))$
$\boldsymbol{F}$	picks 1	$R\boldsymbol{2}\boldsymbol{1} \vee \exists z(R\boldsymbol{2}z \wedge Rz\boldsymbol{1})$
$\boldsymbol{V}$	chooses	$\exists z(R\boldsymbol{2}z \wedge Rz\boldsymbol{1})$
$\boldsymbol{V}$	picks 4	$R\boldsymbol{2}\boldsymbol{4} \wedge R\boldsymbol{4}\boldsymbol{1}$
$\boldsymbol{F}$	chooses	$R\boldsymbol{4}\boldsymbol{1}$
test	$\boldsymbol{F}$ loses	

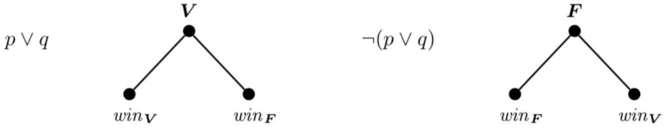
Falsifier started with a threat by picking object 2,but then picked 1.Verifier chose the true right conjunct, and picked the witness 4.Now, falsifier loses with either choice. Still,falsifier could have won, by choosing object 3 that 2 cannot reach in ≤ 2 steps.Falsifier even has another winning strategy,namely,$x = 5, y = 4$

success lemma

$M, s \vDash \phi$ iff V has winning strategy in $game(\phi, M, s)$.
Same as: $M, s \nvDash \phi$ iff F has winning strategy in $game(\phi, M, s)$.

The proof is a direct induction on formulas:
The steps show the close analogy between logical operators and ways of combining strategies.The following typical cases will give the idea. (a) If $\phi \vee \psi$ is true, then at least one of ϕ or ψ is true, say, ϕ. By the inductive hypothesis, V has a winning strategy δ for ϕ. But then V has a winning strategy for the game $\phi \vee \psi$: the first move is left, after which the rest is the strategy δ. (b) If $\phi \vee \psi$ is false, both ϕ and ψ are false, and so by the inductive hypothesis, F has winning strategies δ and τ for ϕ and ψ, respectively. But then the combination of an initial wait-and-see step plus these two is a winning strategy for F in the game $\phi \vee \psi$. If V goes left in the first move, then F should play δ, while, if V goes right, F should play strategy τ. (c) If the formula ϕ is a negation $\neg\psi$ we use a role switch.

Role switch:Consider the game for a formula $p \vee q$ in a model where p is true and q is false, as well as its dual game $\neg(p \vee q)$, that switches all turns and win markings:



Thus, strategies for V in a game for $\neg\psi$ are strategies for F in the game for ψ, and vice versa. Now we prove case (c). Suppose that $\neg\psi$ is true. Then ψ is false, and by the inductive hypothesis, F has a winning strategy in the ψ-game forcing an outcome in the set of F's winning positions. But this is a strategy for V in the $\neg\psi$-game, and indeed one forcing a set of winning positions for V . The other direction is similar.

Modal comparison game: Some theorems

For convenience, in this part, we only use a first-order logic whose vocabulary has finitely many predicate letters and individual constants.

- For all models M and N, if $M \cong N$, then $M \equiv N$.
  Proof An easy induction on first-order formulas ϕ.
- For all finite models, $M \cong N$ iff $M \equiv N$.
  Proof of $\Leftarrow$:
  First, write a first-order sentence δ^M describing M. Consider a model only contain binary relation R and no function.
  There are at least n elements: $\phi_1 = \bigwedge \neg(x_i = x_j)(i \neq j)$and $(i, j \leq n)$.
  There are at most n elements: $\phi_2 = \forall y \bigvee(x_i = y)(i \leq n)$.
  State every element of relation R:$\phi_3 = \bigwedge Rx_i, x_j(a_i, a_j \in R)$.
  State every non-element of R:$\phi_4 = \bigwedge \neg Rx_i x_j(a_i, a_j \notin R)$
  $\phi = \exists x_1 \dots \exists x_n(\phi_1 \wedge \phi_2 \wedge \phi_3 \wedge \phi_4)$.

Continue to prove: $M \equiv N \Rightarrow M \cong N$ Let there be k objects. Then quantify existentially over $x_1, \ldots, x_k$ eumberate all true atomic statements about these in M,plus the true negations of atoms, and state that no other objects exist. Since N satisfies δ^M, it can be enumerated just like M. By definition, the isomorphism is immdiate.

Model comparison game(finite)

Playing the game Consider two models M and N. A player called "duplicator" claims that M and N are similar, while a player called "spoiler" maintains that they are different. Players agree on some finite number k of rounds for the game.

Comparison games

game works as follows, packing two moves into one round. Spoiler (also written S for brevity) chooses one of the models, and picks an object d in its domain. Duplicator (also written D for brevity) then chooses an object e in the other model, and the pair (d, e) is added to the current list of matched objects. After k rounds, the object matching is inspected. If it is a partial isomorphism, duplicator wins; otherwise, spoiler does. Here, a "partial isomorphism" is an injective partial map f between models M and N that is an isomorphism between its own domain and range.

Example

Example Comparing $(Z, <)$ and $(Q, <)$

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

By choosing objects well, duplicator has a winning strategy for the game over two rounds. But spoiler can always win the game in three rounds.

Difference formulas and spoiler's strategies

Distinguish

Winning strategies for spoiler are correlated with first-order formulas ϕ. $\phi = \exists x \exists y (x < y \wedge \neg \exists z (x < z \wedge z < y))$
$Z \vDash \phi$,not $Q \nvDash \phi$.

S choose a witness $d \in \mathbb{Z}$:
$\exists y (d < y \wedge \neg \exists z (d < z \wedge z < y))$
S choose a witness $e \in \mathbb{Z}$:
$d < e \wedge \neg \exists z (d < z \wedge z < e)$
S choose a witness $g \in \mathbb{Q}$:
$d < e \wedge \neg (d < g \wedge z < g)$

D choose a witness $d' \in \mathbb{Q}$:
$\exists y (d' < y \wedge \neg \exists z (d' < z \wedge z < y))$
D choose a witness $e' \in \mathbb{Q}$:
$d' < e' \wedge \neg \exists z (d' < z \wedge z < e')$
Duplicator can't respond:
lose

The quantifier syntax of ϕ triggers the moves for spoiler. Looks like the winning strategy for spoiler over n rounds have correspond with number of quantifiers in difference formula.

Success lemma

We write $WIN(D, M, N, k)$ for: duplicator has a winning strategy against spoiler in k-round comparison game between the models M and N.

Success lemma

For all models M and N, and $k \in N$, the following two assertions are equivalent:
(a)WIN(D,M,N,k):duplicator has a winning strategy in the k-round game.
(b)M and N agree on all first-order sentences up to quantifier depth k.

Proof of $a \Rightarrow b$ is an induction on k. $k = 0$, the initial match of objects must have been a partial isomorphism for D to win. So M and N agree on all atomic sentences, and hence on their Boolean combinations, the formulas of quantifier depth 0.

We proceed with the inductive step. The inductive hypothesis says that, for any models, if D can win their comparison game over k rounds, the models agree on all first-order sentences up to quantifier depth k.Now let D have a winning strategy for the k+1 round game on M and N.Consider any first-order sentence ϕ of quantifier depth k+1. Such a ϕ is equivalent to a Boolean combination of (i)atoms, (ii)sentences of the form $\exists\psi$ with ψ of quantifier depth at most k. The booleans case is trivial. It suffices to show that M and N agree on the latter forms.

Let $M \vDash \exists\psi$. Then for some object d, we get $M, d \vDash \psi$. Think of (M,d) as a model we assign d to x. So $M, d \vDash \psi(\underline{d})$.Now D's winning strategy has a response for whatever S can do in the k+1-round game.If S choose object d from M. Then D has a response e in N. This yield two new model (N,e) and (M,d). By induction hypothesis, we get $N, e \vDash \psi(\underline{d})$. Thus $N \vDash \exists\psi$.

Proof from (b) to (a) requires another induction on k. We need to introduce a lemma:

Finiteness Lemma

Fix variables $x_1, \ldots, x_m$. Up to logical equivalence, there are only finitely many first-order formulas $\phi(x_1, \ldots, x_m)$ of quantifier depth$\leq k$

Proof: k=0, the base step is trivial.As for inductive step, in k+1 round, say let S choose d from M. Now D looks at the set of first-order formulas which is true if d is the witness for $\exists$. By Finite lemma, we can write a formula $\exists x\psi^d$ summarizes all the information. By hypothesis, $M \vDash \exists x\psi^d$ and $N \vDash \exists x\psi^d$. Thus, D can choose a witness e in N responds to d. D has winning strategy in k+1 rounds by $M \cong N$ iff $M \equiv N$.

Theorem

(a) Winning strategies for S in the k-round comparison game for M and N.
(b) There is a first-order sentences ϕ of quantifier depth k with $M \vDash \phi$, not $N \vDash \phi$.

Proof We first look at the direction from (b) to (a). Every ϕ of quantifier depth k induces a winning strategy for S in a k-round game between any two models. Each round $k - m$ starts with a match between objects linked so far that differ on some subformula ψ of ϕ with quantifier depth $k - m$. By Boolean analysis, S then finds some existential subformula $\exists x \alpha$ of ψ with a formula α of quantifier depth $k - m - 1$ on which the models disagree. S's next choice is a witness in that model of the two where $\exists x \alpha$ holds.

Our next direction is from (a) to (b). Any winning strategy δ for S induces a distinguishing formula of proper quantifier depth. To obtain this, let S make the first choice d in model M according to δ, and now write down an existential quantifier for that object. Our formula will be true in M and false in N. We know that each choice of D for an object e in N gives a winning position for 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 exist. Some of these will be true in M (say $A_1 \dots A_r$), and others in N (say $B_1, \dots, B_s$). The total difference formula for strategy δ is then the M-true assertion:

$\exists x(A_1 \wedge \cdots \wedge A_r \wedge \neg B_1 \wedge \cdots \wedge \neg B_s)$

whose appropriateness is easy to check.

Modal comparison game(infinite)

Success lemma for infinite Model comparison game

D has winning strategy iff $M \cong_f N$

Proof

EF Theorem

$M \cong_f N$ iff $M \equiv N$

$M \equiv N$ iff D has winning strategy.

The games in practice

- The rationals$(\mathbb{Q}, <)$ are elementarily equivalent to the reals$(\mathbb{R}, <)$.
  It suffices to show that D can win the comparison game for every k. All choices of spoiler can be countered using the unboundness and density of the orders.
- Even or odd are not first-order definable on the finite models.
  Suppose the even size has a first-order definition on finite models, of quantifier depth k. Then imagine any two finite models for which duplicator can win the k-round comparison game. Such two model can not be distinguished by any formula of which rank is k, otherwise the spoiler wins. So they are both of even size, or both of odd size. This is refuted by any two finite models with k versus k+1 objects in their domains.

Model Construction

The next task asks for finding models that make given assertion true. First we will introduce a semantic tableaus of Beth(1955) for testing existence of models. And then we turn the tableau method into a game between a "builder" **B** and a "critic" **C** disagreeing about a construction making certain assertion true, and other false.

Learning tableaus by example

EXAMPLE 16.2 $A \wedge (B \vee C)$ implies $(A \wedge B) \vee C$
We write the tableau as a tree of unfolding potential counterexamples, noting the rules. Formulas to be true stand on the left, those on the right are to be false:

$$A \wedge (B \vee C) \quad \bullet \quad (A \wedge B) \vee C$$

First, conjunctions on the left are just unpacked, as both conjuncts must be true:

$$A, (B \vee C) \quad \bullet \quad (A \wedge B) \vee C$$

Next, disjunctions on the right are also unpacked, as both disjuncts must be false:

$$A, (B \vee C) \quad \bullet \quad A \wedge B, C$$

Learning tableaus by example

Again, we split the disjunction to the left, as at least one disjunct must be true:

$$A, B \quad \bullet \quad A \wedge B, C \qquad\qquad A, C \quad \bullet \quad A \wedge B, C$$

The node to the right closes, as C occurs on both sides. No counterexample lies that way. Finally, as with the left-split for disjunctions, we split the conjunction to the right in our remaining case, since at least one conjunct must be false:

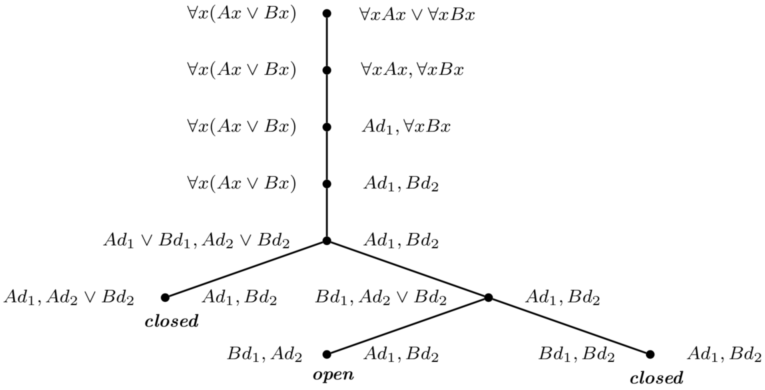
$$A, B \quad \bullet \quad A, C \qquad\qquad A, B \quad \bullet \quad B, C$$

Both of these close, because some formula occurs on both sides. ■

But tableau branches may also remain open, leading to counterexamples.

Example 16.5 A counterexample to a quantifier inference

The following table refutes the inference from $\forall x(Ax \vee Bx)$ to $\forall x Ax \vee \forall x Bx$:



The open branch in the middle describes the simplest counterexample to the above rule. It has two objects, one of which satisfies B only, and the other A only. ■

Tableau, some general features

(a) and (b) are equivalent

(a) The set of formulas$\phi_1, \ldots, \phi_k, \neg\psi_1, \ldots, \neg\psi_m$ is satisfiable.
(b)There is an open tableau with top node $\phi_1 \ldots, \phi_k \bullet \psi_1, \ldots, \psi_m$.

Proof From (a) to (b). Any model M for the given set induces an open branch β_M in the tableau. At propositional splits, a choice is made by checking which disjunct is true(or conjunct false) in the model,taking objects as required from the model.This cannot lead to closure.

Proof From (b) to (a). Any open branch β induces a model M whose domain consists of all objects introduced on β, where we make all atomic statements to the left true, and those on the right false.For other atoms, the valuation is free. Using the tableau decomposition rules, that all the formulas to the left on the branch β are true in branch M_β, while all those to the right are false.

Tableau, some general features

- Closed tableaus are finite.
  Proof We introduce *König*'s lemma to prove this.

König's lemma

Every finitely branching infinite tree has an infinite branch.(We will prove it later).

It is immediately follows that an infinite tableau cannot be closed.

König's lemma proof

Proof of König's lemma:Let T be a rooted tree with an infinite number of nodes, each with a finite number of children.Then T has a branch of infinite length.
We will show that we can choose an infinite sequence of nodes $t_0, t_1, \ldots$ of T such that:
1,t_0 is the root node;
2,$t_n + 1$ is child of t_n;
3,Each t_n has infinitely many descendants.
Then such sequence $t_0, t_1, \ldots$ is such a branch of infinite length.
Take the root node t_0. By definition, it has a finite number of children. t_0 has at least one child with descendants. By axiom of countable choice, we could pick t_1 as any one of those children.Now suppose node t_k has infinitely many descendants. As t_k has finite number of children, by the same argument as above, t_k has at least one child with infinitely many descendants.Thus we may pick t_{k+1} which has infinitely many descendants.

Tableau, some general features

- Closed tableaus $\phi_1, \ldots, \phi_k; \psi_1, \ldots, \psi_m$ correspond to proofs of the initial implication $\bigwedge\{\phi_1, \ldots, \phi_k\}$ to $\bigvee\{\psi_1, \ldots, \psi_m\}$.

Two faces of tableaus

Two faces of tableaus. Read top-down, they are attempts at finding countermodel, bottom-up(when closed), they are proofs.

Model construction games

Definition Model construction games

There are two such sets: the Yes set (the set of true sentences) and the No set (the set of false sentences). Some rounds are automatic.At each stage of play of the game a position is reached. Critic selects a formula to be handled (either from the Yes or No set), after which Builder responds according to the rules listed below.

game rules

- if $\neg\varphi$ is in one box, it changes to φ in the other box
- if $\varphi \wedge \psi$ is in *YES*, it is replaced by φ, ψ separately
- if $\varphi \vee \psi$ is in *NO*, it is replaced by φ, ψ separately
- if $\exists x\varphi$ is in *YES*, it is replaced by $\varphi(d)$ for some new object d not yet used in any formula in *YES* or *NO*
- if $\forall x\varphi$ is in *NO*, it is replaced by $\varphi(d)$ for some new object d not yet used in any formula in *YES* or *NO*

Next come active rounds in which critic schedules some formula for treatment.

(a) Critic can schedule a disjunction in *YES* or a conjunction in *NO*, and builder must choose a subformula replacing that formula.

(b) For existential formulas $\exists x\varphi$ in *NO*, critic mentions some object d in the domain under construction so far, and adds $\varphi(d)$ to the *NO* box. For universal formulas $\forall x\varphi$ in *YES*, critic mentions some object d in the domain under construction so far, and adds $\varphi(d)$ to the *YES* box.

Winning convention

Here is the winning convention. A stage is a loss for builder if some formula occurs in both boxes, while builder wins a run of the game if no such loss occurs at any stage. Note that no model can make a formula both true and false: these are indeed conflicting tasks.

Model construction game example

Example Here is an example which illustrates the correlation between a winning strategy for Builder and the existence of a model (valuation) for the initial formula. The initial position is:
$\{((s \vee r) \wedge (q \vee p)) \wedge \neg(q \vee (\neg p \vee r))\}; \emptyset$
Here is a winning strategy for Builder. After the automatic moves, the players reach the position:
$\{p, (s \vee r), (q \vee p)\}; \{q, r\}$.
If Critic schedules $(s \vee r)$, the players go to:
$\{p, (s \vee r)^{*}, (q \vee p); \{q, r\}\}$
(the asterix indicates the formula scheduled by Critic) and if he schedules $(q \vee p)^{*}$, the players reach the position:
$\{p, (s \vee r), (q \vee p)^{*}; \{q, r\}\}$

In the first case, let Builder choose s, and after Critic scheduling $(q \vee p)$, let her choose p. The play ends up with the position $\{p, s\}$;$\{q, r\}$which is a win for Builder. In the second case, let Builder choose p, and after Critic scheduling $(s \vee r)$, let her choose s. The play ends up with the same position as in the previous case which is a win for Builder. From this position we get a valuation for the initial sentence, by assigning True to the symbols in Yes and False to the symbols in No.

Model construction game example

Example This game illustrates the correlation between a winning strategy for Critic in the model construction game with φ in the No box, and the logical validity of φ. Let φ be $\exists x(A \vee B) \to (\exists xA \vee \exists xB)$ that we rewrite as $\neg\exists x(A \vee B) \vee (\exists xA \vee \exists xB)$. We now take the root of the game-tree to be $\varnothing; \{\varphi\}$. The idea is that from the assumption that φ is false, a contradiction follows, hence a winning strategy for Critic. An automatic move leads to

$$\{A(c_1) \vee B(c_1)\}; \{\exists xA, \exists xB\}.$$

Let Critic schedule the formula in the Yes box. Now, if Builder chooses $A(c_1)$, then let Critic schedule $\exists xA$ and then choose the constant c_1. Analogously, if Builder selects $B(c_1)$, then let Critic schedule $\exists xB$ and then choose the constant c_1.

Success lemma 1

Success lemma 1

The following are equivalent for first-order logic:
1.The set of formulas $\phi_1, \dots, \phi_n, \neg\psi_1, \dots, \neg\psi_m$ is satisfiable.
2,Builder has a winning strategy in the construction game with which starts with $\phi_1, \dots, \phi_n;\ \psi_1, \dots, \psi_m$.

Proof The direction from (1) to (2) gives an explicit correspondence between models and winning strategies for Builder. In fact if $\phi_1, \dots, \phi_n, \neg\psi_1, \dots, \neg\psi_m$ is satisfiable, starting from the root, in every immediate extension there is an open branch. That is, the moves for the conjunction and the universal quantifier do not close the branch. As for the disjunction and the existential quantifier, they preserve the satisfiability, so Builder is guaranteed to have a winning strategy.

Proof of success lemma 1

Proof The direction from (2) to (1) gives an explicit correspondence between winning strategy of Builder and models. Namely, if Builder has a winning strategy, then in all the histories in which Builder follows it, no contradiction appears. From the atomic formulas which appear in the Yes and No boxes (not all of them need to be taken into account), a Hintikka set is formed, from which a model for $\phi_1, \ldots, \phi_n, \neg\psi_1, \ldots, \neg\psi_m$ can be built by well known methods.

Success lemma 2

3.The sentence $\phi_1 \wedge \cdots \wedge \phi_n \to \psi_1 \vee \cdots \vee \psi_m$is a logical truth (i.e. it is provable).
4.Critic has a winning strategy in the construction game which starts with $\phi_1, \ldots, \phi_n;\psi_1, \ldots, \psi_m$

Proof The equivalence between (3) and (4) is also straightforward. A winning strategy for the Critic allows him, for every possible move by Builder, to reach a position which is a win for Critic. Notice, however, and this is one of the main differences with tableaus, that not all branches of the game tree are closed, but only those where the Critic's winning strategy is followed. Critic's winning strategies are explicitly correlated with proofs. To keep things simple, suppose $n = 0$ and $m = 1$. In the classical tableaus method one argues first that if ψ_1 is a first-order valid sentence, then there is a closed tableau starting with $\emptyset;\psi_1$ The tableau closes after finitely many steps.

Success lemma 2 proof

An explanation of this is given using Koning's Lemma (each finitely branching infinite tree has an infinite branch): a closed infinite tableau is impossible because if the tableau is closed then every branch of it must be finite, hence the tableau must be finite. In the present case Koning's Lemma is not needed: tableaux are finite, hence a winning strategy for Critic is a finite objects. The connection with proof comes quite naturally by tableaus theorem:Closed tableaus correspond to proofs of the initial implication $\bigwedge\{\phi_1, \ldots, \phi_k\}$ to $\bigvee\{\psi_1, \ldots, \psi_m\}$.

Conclusion

We have turned the well-known method of testing satisfiability via semantic tableaus into a model construction game. Doing so provides a unified framework for two basic logical tasks that seem very different, but are in fact intuitively intertwined: finding proofs, and constructing models.

Argumentation and Dialogue

Dialogue game A formal debate takes place between a proponent P who defends a claim against an opponent O who grants initial concessions. Moves are attacks and defenses on assertions according to logical and procedural rules. Logical rules involve choices, switches, and picking instances, but no external world determines who wins or loses, only internal criteria such as consistency. Likewise, in actual debate, people often lose by incoherent positions, rather than the judgment of an external arbiter. Procedural conventions are real, too: they can be observed in a court of law.

Definition Dialogue rules of attack and defense

Here are the above rules summarized in a table:

Operator	*Attack*	*Defense*
$A \wedge B$	$?L$	A
	$?R$	B
$A \vee B$	$?$	A, B
$\neg A$	A	$-$
$A \rightarrow B$	A	B
$\exists x \varphi$	$?$	$\varphi(d)$
$\forall x \varphi$	d	$\varphi(d)$

Each move is either a defense to a preceding attack, or an attack on something the other player has asserted. ■

- **Procedural conventions** The scheduling of the dialogue is : players move in turn, as in earlier logic games. Next, we constrain debate by stipulating the rights and duties of players:
  (a) Proponent may only assert an atomic formula after opponent has asserted it.
  (b) If one responds to an attack, this has to be to the latest still open attack.
  (c) An attack may be answered at most once.
  (d) An assertion made by proponent may be attacked at most once.
- **Winning and losing** A player loses if there is nothing legitimate left to say at that player's turn and no attack or answer is available and the last position is a atomic formula.

Example: Defending $p \wedge \neg(p \wedge q) \rightarrow \neg q$

Here is a fairly typical dialogue for a classically valid implication:

1	$\boldsymbol{P}$	$p \wedge \neg(p \wedge q) \rightarrow \neg q$	
2	$\boldsymbol{O}$	$p \wedge \neg(p \wedge q)$	[A, 1]
3	$\boldsymbol{P}$	$\boldsymbol{P}$ can respond to the first attack, or counterattack. Say:	
	$\boldsymbol{P}$	$?L$	[A, 2]
4	$\boldsymbol{O}$	p	[D, 3]
		this is the only thing $\boldsymbol{O}$ can do at this stage	
5	$\boldsymbol{P}$	$?R$	[A, 2]
6	$\boldsymbol{O}$	$\neg(p \wedge q)$	[D, 5]
7	$\boldsymbol{P}$	$\neg q$	[D, 2]
8	$\boldsymbol{O}$	$\boldsymbol{O}$ has no further attacks to respond to, so $\boldsymbol{O}$ must attack:	
	$\boldsymbol{O}$	q	[A, 7]
9	$\boldsymbol{P}$	$p \wedge q$	[A, 6]
10	$\boldsymbol{O}$	$?L$	[A, 9]
11	$\boldsymbol{P}$	p	[D, 10]
		this is admissible, as p has been asserted before by $\boldsymbol{O}$	
12	$\boldsymbol{O}$	$?R$	[A, 10]
13	$\boldsymbol{P}$	q	[D, 12]
		this is admissible, as q has been asserted before by $\boldsymbol{O}$	

$\boldsymbol{O}$ has nothing legitimate to say at this stage, and loses. ∎

Excluded middle

The games show some striking differences with classical logic:

1	$\boldsymbol{P}$	$q \vee \neg q$	
2	$\boldsymbol{O}$	?	[A, 1]
3	$\boldsymbol{P}$	$\neg q$	[D, 2]
4	$\boldsymbol{O}$	q	[A, 3]

This time, $\boldsymbol{P}$ has nothing legitimate to say at this stage, and loses.

If we want $\boldsymbol{P}$ to win this game for a classical law after all, we have to change the procedural rules stated above, and allow $\boldsymbol{P}$ to reply again to an earlier attack:

5	$\boldsymbol{P}$	q	[A, 3]

Now $\boldsymbol{O}$ is the one who has nothing legitimate left to say, and loses. ■

Theorem

The following are equivalent for first-order formulas $A_1 \dots A_k, B$:
(a) P has a winning strategy in the dialogue game for $A_1 \wedge \cdots \wedge A_k \to B$
(b) $A_1 \wedge \cdots \wedge A_k \to B$ is valid in intuitionistic logic.

We have introduced the major varieties of logic games: for evaluating formulas in models, comparing models, constructing models, and for engaging in proof dialogues. In each case, we defined the games, studied the structure of their strategies, and developed some basic theory about winning strategy corresponds to some assertion in logic. *"For us, these games were not just tools for business as usual, even though it is quite true that game talk is a powerful metaphor for standard logic, packaging complex intuitions in a helpful concrete manner. But going beyond that, we have also presented the games as a novel way of thinking about logic as a family of dynamic multi-agent activities. Thus, to us, logic as games is a multi-faceted enterprise, going beyond what is sometimes called game semantics for logical languages."*—Johan Van Benthem

Reference

- Johan van Benthem: Logic in Games, MIT Press, 2014
- Felscher, W. (2001). Dialogues as a foundation for intuitionistic logic.
- Wilfrid Hodges: Logic and Games, Stanford Encyclopedia of Philosophy, Sections 1-3.