Formula Evaluation

Abraham Hinteregger

Vienna University of Technology

2016-06-07

Chapter

Motivation

- Given is a formula φ in a model M with a variable setting s
- In model-theoretic semantics the question was whether this formula is true in this model with the setting ($M, s \models \varphi$) or not ($M, s \not\models \varphi$)
- If one person thinks the formula is true and another person doubts that an obvious game arises:
  - The first person (verifier, **V**) tries to verify that the formula is true
  - the second person (falsifier, **F**) tries to do the opposite

First order predicate logic (reminder[1])

- A formula is built from formulas $(A, B, \ldots)$ and operators:
  - Constants $\top, \bot$
  - Unary negation operator $\neg A$
  - Binary operators $\wedge, \vee$: $A \circ B$
  - Quantifiers $\exists, \forall$: $\exists x A(x), \forall x B(x)$
  - Predicates, e.g. $A(x), B(x, y)$
- A model is a set of objects, also called the universe/domain
  - Constants are mapped to an object in the domain
  - Functions map one (or more) objects in the domain to another object in the domain
  - Predicates are mapped to a subset of the domain

[1] in the spirit of CI

Evaluation games for predicate logic

atoms $P(x), R(x,y), \top, \bot$	**V** wins if atom is true, else **F** wins
disjunction $\varphi \vee \psi$	**V** chooses which disjunct to play
conjunction $\varphi \wedge \psi$	**F** chooses which conjunct to play
negation $\neg\varphi$	role switch of the two players
existential quantifier $\exists x \varphi(x)$	**V** picks an object d from the domain
universal quantifier $\forall x \varphi(x)$	**F** picks an object d from the domain

Evaluation games for modal logic

- Rules from evaluation games for predicate logic
- Additional rules that accomodate for the modal operators $\Box$ and $\Diamond$ (or indexed versions $\Box_i, \Diamond_i$):

necessity $\Box P$	**F** chooses a successor of the current world
possibility $\Diamond P$	**V** chooses a successor of the current world

  Failure to choose a successor means a loss for either player
- Game state consists not only of setting s and current formula φ but also the current world w.

Example of evaluation games for predicate logic

Example of evaluation games for predicate logic

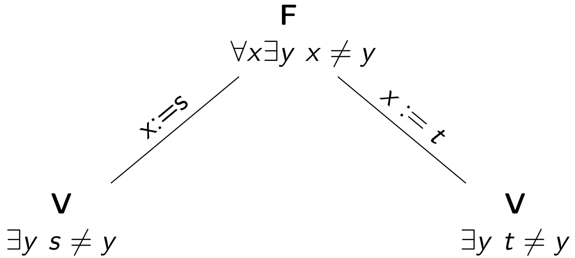
(domain consists of two objects, s and t)

F

$\forall x \exists y \; x \neq y$

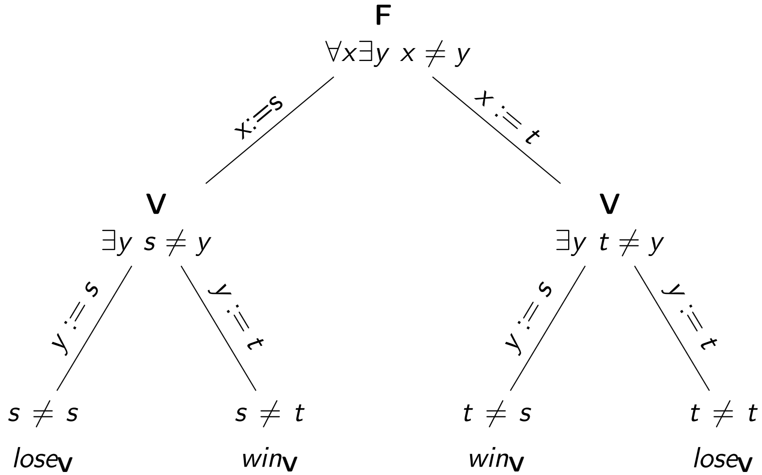
Example of evaluation games for predicate logic

(domain consists of two objects, s and t)



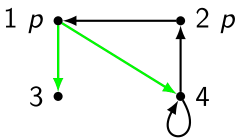
Example of evaluation games for predicate logic

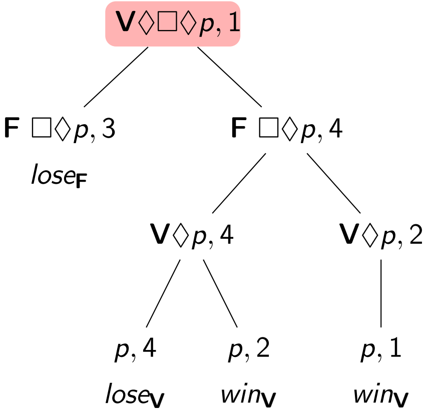
(domain consists of two objects, s and t)



Example of evaluation game for modal logic

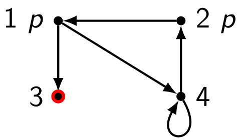
- If **V** chooses world 3 **F** can't move and loses
- Else **V** can choose either world 1 or 2 and wins no matter what **F** does

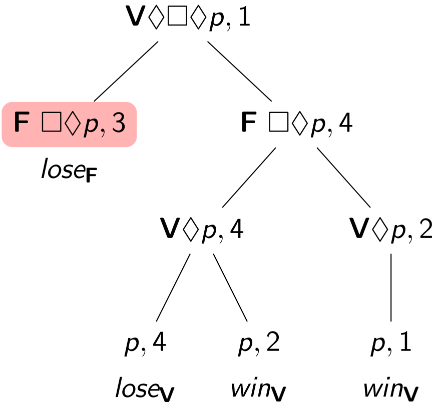




Example of evaluation game for modal logic

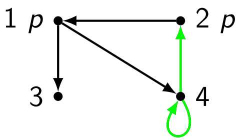
- If **V** chooses world 3 **F** can't move and loses
- Else **V** can choose either world 1 or 2 and wins no matter what **F** does

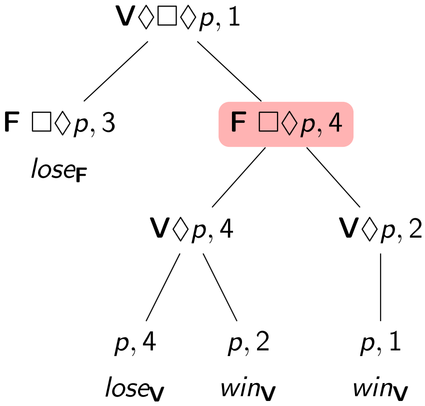




Example of evaluation game for modal logic

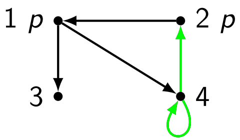
- If **V** chooses world 3 **F** can't move and loses
- Else **V** can choose either world 1 or 2 and wins no matter what **F** does

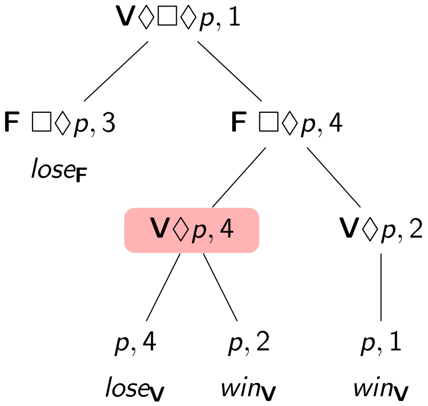




Example of evaluation game for modal logic

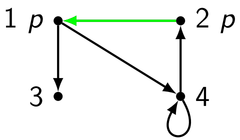
- If **V** chooses world 3 **F** can't move and loses
- Else **V** can choose either world 1 or 2 and wins no matter what **F** does

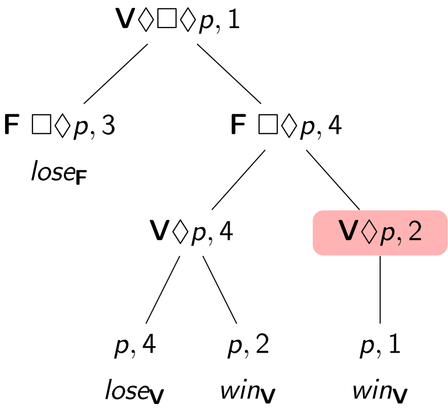




Example of evaluation game for modal logic

- If **V** chooses world 3 **F** can't move and loses
- Else **V** can choose either world 1 or 2 and wins no matter what **F** does





Some observations

- Both players can choose between two options (assigning either s or t to x, y or deciding a successor for the current world)

Some observations

- Both players can choose between two options (assigning either s or t to x, y or deciding a successor for the current world)
- Both players can win and lose but only **V** does influence the outcome of the games (**F** cannot even force a loss)

Some observations

- Both players can choose between two options (assigning either s or t to x, y or deciding a successor for the current world)
- Both players can win and lose but only **V** does influence the outcome of the games (**F** cannot even force a loss)
- **V** has a winning strategy ("don't assign the same thing as **F**", "go to world 1 or 2")

Some observations

- Both players can choose between two options (assigning either s or t to x, y or deciding a successor for the current world)
- Both players can win and lose but only **V** does influence the outcome of the games (**F** cannot even force a loss)
- **V** has a winning strategy ("don't assign the same thing as **F**", "go to world 1 or 2")
- In the second game **V** has even two strategies for winning ("pick 3", "pick 4 and then pick either 2 or 1")

Determinacy[2]

Success Lemma

$M, s \models \varphi \iff$ **V** has a winning strategy in $game(\varphi, M, s)$
$M, s \not\models \varphi \iff$ **F** has a winning strategy in $game(\varphi, M, s)$
Proof by induction on formulas:

- If $v_M(\varphi \vee \psi) = 1$: w.l.o.g. $v_M(\varphi) = 1$. By inductive hypothesis **V** has a winning strategy for $game(M, s, \varphi)$ and thus also for $game(M, s, \varphi \vee \psi)$
- If $v_M(\neg\varphi) = 1$: $\implies v_M(\varphi) = 0$ and by IH **F** has a winning strategy for $game(M, s, \varphi)$. Player switch yields a winning strategy for **V** for $game(M, s, \neg\varphi)$.
- ...

[2]Hintikka referred to this as "determinateness" [Hintikka, 1982]

Determinacy II

- Complexity of formula strictly decreases as game continues until only atomic formula is left
- At every branching the active player has a winning strategy if there is a winning strategy available for at least one of the branches
- As this propagates to the root there's a winning strategy for at least (and at most) one of the two players.
- This theory of truth coincides with Tarski's truth definition

Chapter

Inductive definition of the game

A game "$game(M, s, \varphi)$" is defined as a tree where every node is a pair (s, ψ) where s is an M-assignment and ψ is a subformula of φ.

φ	Interpretation of $game(M, s, \varphi)$
atomic	one node game where **V** wins iff $M, s \models \varphi$
$\exists x$	game where **V** picks any available move $s[x := d]$ and wins
$\phi \vee \psi$	game is disjoint union of two games and it's **V**'s turn
$\phi \wedge \psi$	same as above but it's **F**'s turn
$\neg\phi$	$game(M, s, \phi)$ with win markings reversed
$\phi; \psi$	Game arising by taking $game(M, s, \phi)$ with assignment t at end states and continuing with $game(M, t, \psi)$

Extended syntax of formulas

The changed quantifier and composition rules allow formulas such as:

- $\exists x$: **V** chooses new assignment $s[x := d]$ and wins.

Extended syntax of formulas

The changed quantifier and composition rules allow formulas such as:

- $\exists x$: **V** chooses new assignment $s[x := d]$ and wins.
- $P(x); \exists x$: Test whether $P(s(x))$ holds and then assign a new value to x.

Additional moves[3]

Until now the model M was fixed and remained unchanged. The game could be extended by allowing moves that manipulate the model in some way:

- Adding or removing objects from the domain
- Changing the interpretation

[3] Chapter 16, [Van Benthem, 2014]

Refined semantics[4]

- Difference between the existence of one winning strategy and many of them.
- How many moves does a strategy need to win?
- Is it possible to lose on purpose?

[4] Chapter 15, [Van Benthem, 2014]

Chapter

References I

Hintikka, J. (1982).
Game-theoretical semantics: insights and prospects.
Notre Dame J. Formal Logic, 23(2):219–241.

Hodges, W. (2013).
Logic and games.
In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Spring 2013 edition.

Keiff, L. (2011).
Dialogical logic.
In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Summer 2011 edition.

References II

Knaster-Tarski.
Knaster–Tarski theorem.
Wikipedia, the free encyclopedia.
Page Version ID: 723193149.

Van Benthem, J. (2014).
Logic in games.
MIT press.

Venema, Y. (2007).
Lectures on the modal μ-calculus.
University of Amsterdam.

Modal μ-calculus[6]

Adds two additional operators to propositional (multi-) modal logic:

- Least fixpoint operator $\mu p : \varphi(p)$
- Greatest fixpoint operator $\nu p : \varphi(p)$

Note that p occurs positively in $\varphi(p)$, meaning that there's an even amount of $\neg$ in front of every occurence of p in φ[5]

[5] The positive syntactic occurence of p implies monotonicity concerning inclusion $\implies$ least and greatest fixpoint exist [Knaster-Tarski,].

[6] [Venema, 2007]

Examples for modal μ-calculus

Formula	Interpretation
$\mu p : (q \vee \Diamond p)$	

Examples for modal μ-calculus

Formula	Interpretation
$\mu p : (q \vee \Diamond p)$	Set of all worlds w where a world v s.t. $M, v \models q$ is reachable with finite path

Examples for modal μ-calculus

Formula	Interpretation
$\mu p : (q \vee \Diamond p)$	Set of all worlds w where a world v s.t. $M, v \models q$ is reachable with finite path
$\nu p : (q \wedge \Box_a p)$	

Examples for modal μ-calculus

Formula	Interpretation
$\mu p : (q \vee \Diamond p)$	Set of all worlds w where a world v s.t. $M, v \models q$ is reachable with finite path
$\nu p : (q \wedge \Box_a p)$	Set of all worlds w where $M, w \models q$ on every a-path

Examples for modal μ-calculus

Formula	Interpretation
$\mu p : (q \vee \Diamond p)$	Set of all worlds w where a world v s.t. $M, v \models q$ is reachable with finite path
$\nu p : (q \wedge \Box_a p)$	Set of all worlds w where $M, w \models q$ on every a-path
$\nu p : (\Diamond \top \wedge \Box p)$	

Examples for modal μ-calculus

Formula	Interpretation
$\mu p : (q \vee \Diamond p)$	Set of all worlds w where a world v s.t. $M, v \models q$ is reachable with finite path
$\nu p : (q \wedge \Box_a p)$	Set of all worlds w where $M, w \models q$ on every a-path
$\nu p : (\Diamond \top \wedge \Box p)$	Set of all worlds that have outgoing transitions and don't have a path to a world without outgoing transitions

Examples for modal μ-calculus

Formula	Interpretation
$\mu p : (q \vee \Diamond p)$	Set of all worlds w where a world v s.t. $M, v \models q$ is reachable with finite path
$\nu p : (q \wedge \Box_a p)$	Set of all worlds w where $M, w \models q$ on every a-path
$\nu p : (\Diamond \top \wedge \Box p)$	Set of all worlds that have outgoing transitions and don't have a path to a world without outgoing transitions

Examples for modal μ-calculus

Formula	Interpretation
$\mu p : (q \vee \Diamond p)$	Set of all worlds w where a world v s.t. $M, v \models q$ is reachable with finite path
$\nu p : (q \wedge \Box_a p)$	Set of all worlds w where $M, w \models q$ on every a-path
$\nu p : (\Diamond \top \wedge \Box p)$	Set of all worlds that have outgoing transitions and don't have a path to a world without outgoing transitions

ν means unfolding, μ means finite unfolding

Evaluation game for μ-calculus

- Rules from modal evaluation game
- If fixed point formula $\mu p : \varphi(p)$ or $\nu p : \varphi(p)$ is reached, the game proceeds with $\varphi(p)$

[7] if there is more than one such variable the one with highest rank (contains the others as subformulas) counts

Evaluation game for μ-calculus

- Rules from modal evaluation game
- If fixed point formula $\mu p : \varphi(p)$ or $\nu p : \varphi(p)$ is reached, the game proceeds with $\varphi(p)$
- p is not an atom but a bound variable and instead of testing the atom the original fixed point formula is substituted back in.

---

[7] if there is more than one such variable the one with highest rank (contains the others as subformulas) counts

Evaluation game for μ-calculus

- Rules from modal evaluation game
- If fixed point formula $\mu p : \varphi(p)$ or $\nu p : \varphi(p)$ is reached, the game proceeds with $\varphi(p)$
- p is not an atom but a bound variable and instead of testing the atom the original fixed point formula is substituted back in.
- If the evaluation loops (node visited multiple times) **V** wins if the infinitely many times substituted variable is bound to a ν-formula and **F** wins if it's bound to a μ- formula[7]

[7] if there is more than one such variable the one with highest rank (contains the others as subformulas) counts