1 Introduction

About this lecture

- Refutation tree and valid argument
- Refutation Tree Rules
- Course homepages: http://mathsci.kaist.ac.kr/~schoi/logic.html and the moodle page http://moodle.kaist.ac.kr
- Grading and so on in the moodle. Ask questions in moodle.

Some helpful references

- Richard Jeffrey, Formal logic: its scope and limits, Mc Graw Hill
- A mathematical introduction to logic, H. Enderton, Academic Press.
- Whitehead, Russel, Principia Mathematica (our library). (This could be a project idea.)
- http://plato.stanford.edu/contents.html has much resource.
- http://ocw.mit.edu/OcwWeb/Linguistics-and-Philosophy/24-241Fall-2005/CourseHome/ See also "The Search-for-Counterexample Test for Validity" This a slightly different one.

2 Refutation tree and valid argument

Refutation tree and valid argument

- Recall the valid argument
- To check, we need to show the premises T, T,..,T imply that the conclusion is T always.
- Or conversely, if the conclusion is F, then there is at least one F in the premises
- That is if the conclusion is negated, then not all premises are T.
- Start by negating the premise. Then show that premises and the negated conclusion cannot be all true at the same time. Then this is a valid argument.
- If the premises and the negated conclusions are all true in some way, then the argument is invalid.
- Note that I did not supply proof that this works always.

Refutation tree example

- We break the statements down to atomic items and see if there can be all true instances or not. (invalid case)
- The aim is to obtain paths of atomic statements.
- $P \wedge Q \vdash P$.
- $P \wedge Q$, $\neg P$.
- ✓ $P \wedge Q$, P, Q, $\neg P$.
- The nonchecked atomic items cannot all be true.
- Valid

Refutation tree example

- $P \vee Q, \neg P \vdash Q$.
- ✓ $P \vee Q$, $\neg P$, $\neg Q$, (i) P or (ii) Q.
- (i) $\neg P$, $\neg Q$, P (X)
- (ii) $\neg P$, $\neg Q$, Q. (X)
- The nonchecked atomic items cannot all be true.
- Thus valid.

3 Refutation Tree Rules

Refutation Tree Rules

- Negation $\neg$: If any open path contains both a formula and its negation, place X. (This path is now closed)
- Negated negation $\neg\neg$; In any open path, check any unchecked $\neg\neg\phi$ and write ϕ at the bottom of every path containing it.
- Conjunction $\wedge$: In any open path, check any unchecked $\phi \wedge \psi$ and write ϕ and ψ at the bottom of every path containing it. (same path)
- Disjunction $\vee$: If an open path contain unchecked $\phi \vee \psi$, then check it and the split the bottom of every path containing it into two with one ϕ added and the other ψ added.
- Conditional $\rightarrow$. Unchecked $\phi \rightarrow \psi$. Check it and branch every path containing it into two (i) $\neg\phi$ (ii) ψ.

- Biconditional $\leftrightarrow$. Unchecked $\phi \leftrightarrow \psi$. Check it and branch every path containing it into two (i) $\neg\phi, \neg\psi$ and (ii) ϕ, ψ.
- A path is finished (or closed) if X appears.
- See 3.27 and 3.28.

Refutation Tree Rules

- Negated conjunction $\neg\wedge$: Unchecked $\neg(\phi \wedge \psi)$. Check it and split the bottom of every open path containing it into two (i) add $\neg\phi$ (ii) add $\neg\psi$.
- Negated disjunction $\neg\vee$: unchecked $\neg(\phi \vee \psi)$ and write $\neg\phi$ and $\neg\psi$ at the bottom of every path containing it.
- Negated conditional $\neg \rightarrow$: In any open path, check any unchecked $\neg(\phi \rightarrow \psi)$ and write ϕ and $\neg\psi$ at the bottom of every path containing it. (same path)
- Negated biconditional $\neg \leftrightarrow$: In any open path, check any unchecked $\neg(\phi \leftrightarrow \psi)$ and branch the bottom of every path containing it into two write ϕ and $\neg\psi$ at one (i) and write $\neg\phi$ and ψ (ii)

Example

- 1. $B \rightarrow \neg A$ 2 $\neg B \rightarrow C$. Conclusion $A \rightarrow C$.
- 1. $B \rightarrow \neg A$ 2. $\neg B \rightarrow C$, 3. $\neg(A \rightarrow C)$.
- 1. $B \rightarrow \neg A$ 2. $\neg B \rightarrow C$, 3. check $\neg(A \rightarrow C)$. 4 A, 5 $\neg C$.
- check 1. $B \rightarrow \neg A$, 2. $\neg B \rightarrow C$, check 3. $\neg(A \rightarrow C)$. 4 A, 5 $\neg C$ 6 (i) $\neg B$ (ii) $\neg A$ (X) from 1.
- check 1. $B \rightarrow \neg A$, check 2. $\neg B \rightarrow C$, check 3. $\neg(A \rightarrow C)$. 4 A, 5 $\neg C$ 6 (ii) $\neg A$ (X) (i) $\neg B$ from 1 (i)(i) $\neg\neg B$ (X) (i)(ii) C (X) from 2.
- Now complete. valid

Open tree case

- If open path arises without X, then invalid.
  - 1. $A \rightarrow B$ 2. $\neg A$ 3. $\vdash B$.
  - 1. $A \rightarrow B$ 2. $\neg A$ 3. $\neg B$.
  - check 1. $A \rightarrow B$ 2. $\neg A$ 3. $\neg B$. (i) $\neg A$ (ii) B. (X).
  - (i) is still alive.
  - Invalid case: $\neg A, \neg B$ is the counter example.

Tautology Rules

- A wff ϕ is a tautology if and only if $\neg\phi$ is truth-functionally inconsistent.
- ϕ is a tautology if and only if all path in the finished tree are closed.
- Examples: to be filled....

Some helpful remarks

- Do not apply rules to subformulas. (Confusing)
- The order of rules applied does not make any difference. It is more efficient to apply nonbranching rules first.
- The process eventually terminates. (not go forever). Decidability.
- Soundness of the test: If we can validity from the test, then we can trust it.
- Completeness of the test: If we can invalidity from the test, then we can trust it: we even get counter-examples.
- We need proof: Omit proof in R. Jeffery, Formal logic page 34.