

CS F214 Logic in Computer Science

MODULE: PROPOSITIONAL LOGIC

Proof System - Natural Deduction

Propositional Logic

- Propositional Logic is essentially Boolean Logic: i.e.
 - it is a logic of *formulas* (i.e. *propositions*)
 - that evaluate to TRUE or FALSE
 - made from *atomic propositions* and *logical operations* (AND, OR, NOT ...)
 - e.g. raining AND sunny
 - e.g. raining AND sunny OR NOT humid AND NOT hot



Propositional Logic - Propositions

- An atomic proposition is a statement that is atomic (i.e. understood as is without structure):
 - It may be true or false,
 - but its truth or falsity is
 - not based on any structure
 - nor is it derived from something else
 - e.g. (from the formulas in the last slide):
 - raining, sunny, humid, hot
- In Boolean Logic <u>atomic propositions</u> are referred to as <u>variables</u>.



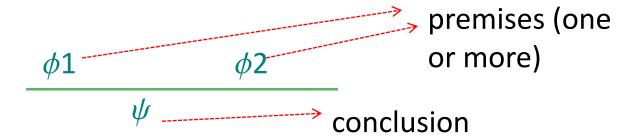
Proof Rules

- Proofs in (propositional) logic can be written down in various formats:
 - one of them is (Gentzen's) *natural deduction*
- Proofs are based on rules (of inference):
 - i.e. rules that state "one can infer S₁ from S₂"



Proof Rules – Generic Format

• In natural deduction, rules are written in the form:

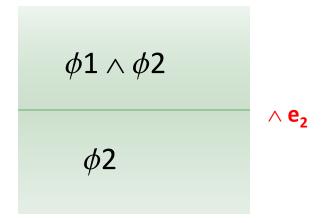


- Each premise is a (class of) propositional formula(s).
- •Similarly a *conclusion* is a (class of) propositional formula(s)



Proof Rules - Format - Example

Consider this rule (labeled \(\Lambda \):



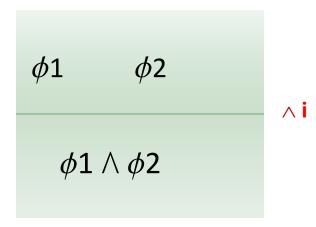
This rule is read as:

given the premise
$$\phi 1 \land \phi 2$$
 one can conclude $\phi 2$



Proof Rules – Different Kinds

- In natural deduction, typically, <u>for each operation there are</u> <u>two kinds of rules</u>:
 - one kind for *elimination*
 - i.e. to prove a conclusion by <u>eliminating the operation</u>
 - e.g. see previous slide (\(\simes \) elimination)
 - one kind for *introduction*
 - i.e. to prove a conclusion by introducing the operation
 - e.g.





Proof Technique: Natural Deduction: Sequents

- In natural deduction, the statement (intended to be proved) is referred to as *a sequent*:
 - ϕ 1, ϕ 2, ..., ϕ n |-- ψ

which is read as

- (the set of premises) " ϕ 1, ϕ 2, ..., and ϕ n" *entails* (the conclusion) " ψ "
- e.g. (sequent):
 - $p \land q, r \mid --q \land r$
- <u>To prove the sequent</u> one has to apply (appropriate) proof rules.
 - Q: Which rule(s) is/are required to prove the example sequent given above?



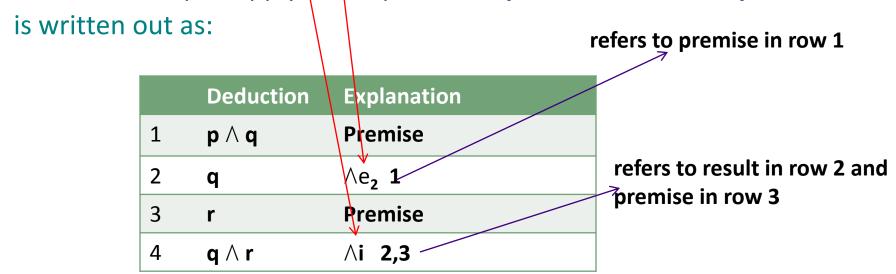
Proofs in Natural Deduction

- Typically a (deduction) step in a proof involves:
 - applying a rule on one or more premises to derive a conclusion:
 - for instance, from applying the rule $\wedge \mathbf{e_2}$ on the premise $\mathbf{p} \wedge \mathbf{q}$ results in the conclusion, say, \mathbf{q}
- Typically, then, a (natural deduction) proof is a sequence of such (deduction) steps:
 - e.g. proof for the sequent $\mathbf{p} \wedge \mathbf{q}$, $\mathbf{r} \mid --\mathbf{q} \wedge \mathbf{r}$
 - Step 1: apply $\wedge \mathbf{e_2}$ on <u>premise $\mathbf{p} \wedge \mathbf{q}$ to result in \mathbf{q} </u>
 - Step 2: apply \wedge **i** on <u>inference **q**</u> and <u>premise **r**</u> to result in **q** \wedge **r**



Proofs in Natural Deduction

- Proofs are written in a particular format:
 - as a table with one column of premises and results along with corresponding explanations in the next column
- For instance, the 2-step proof from the previous slide i.e.
 - Step 1: apply $\wedge \mathbf{e_2}$ on $\mathbf{p} \wedge \mathbf{q}$ to result in \mathbf{q}
 - Step 2: apply \wedge i on premises q and r to result in $\mathbf{q} \wedge \mathbf{r}$



Note that the explanations may refer to the entries in the deduction column by using the row number.



CS/IS F214 Logic in Computer Science

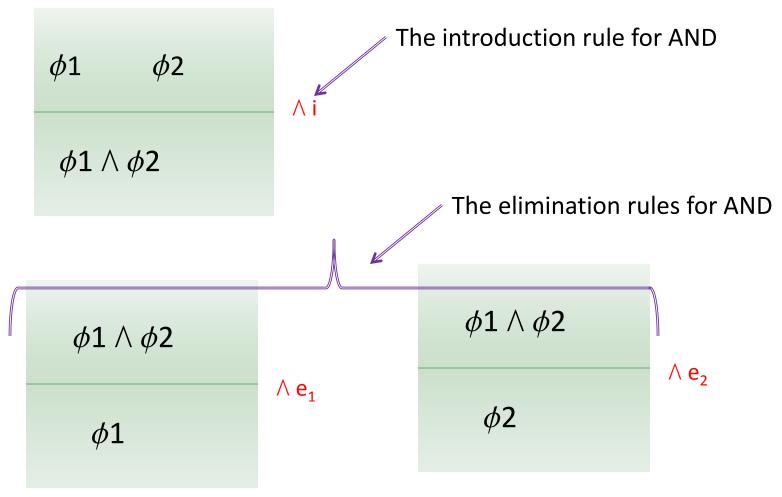
MODULE: PROPOSITIONAL LOGIC

Natural Deduction: Rules for Conjunction

18-08-2018 Sundar B. CS&IS, BITS Pilani 10

ND: Rules for Conjunction

• Conjunction refers to the operation "AND":





ND: Rules for Conjunction – Example 1

- Prove the following sequent:
 - hot ∧ humid, sleepy ∧ dull |-- hot ∧ sleepy
- The proof evolves via the following steps:

	Deduction	Explanation
1	hot \wedge humid	Premise
2	hot	∧e ₁ 1

The first ∧ -elimination rule is applied on the premise in line 1 i.e. on hot ∧ humid to obtain the result hot



ND: Rules for Conjunction – Example 1

(continued)

	Deduction	Explanation
1	$\mathbf{hot}\ \land \mathbf{humid}$	Premise
2	hot	^e ₁ 1
3	dull \wedge sleepy	Premise
4	sleepy	∧e ₂ 3

The second ∧-elimination rule is applied on premise in line 3

i.e. on dull \land sleepy to obtain the result sleepy



ND: Rules for Conjunction – Example 1

(continued)

	Deduction	Explanation
1	hot \land humid	Premise
2	hot	^e₁ 1
3	dull ∧ sleepy	Premise
4	sleepy	∧e ₂ 3
5	hot ∧ sleepy	∧i 2,4

The \land -introduction rule is applied on <u>results in</u> <u>line 2 and 4</u> to obtain the result *hot* \land *sleepy*



Natural Deduction: Proofs

Observations (about proofs):

Typically the <u>first step of a proof</u> is <u>a premise</u> from the given sequent

and the last step is the conclusion from the given sequent.

i.e. if the sequent (to be proved) is of the form

$$\phi_1, \phi_2, ... \phi_n \mid -- \psi$$

then the proof will (typically) be of the form

 $\mathbf{b}_{\mathbf{k}}$

Premise

•••

Ψ

Conclusion



Natural Deduction: Proofs

Observations (about proofs):

- Typically the rule (to be) applied is
 - (i) an elimination rule for an operation if

 the (desired) result should be devoid of the operation

 and one of the premises contains the operation.
 - (ii) an introduction rule for an operation if

 the (desired) result should involve the operation
 and the premises may not



Conjunction - Exercises

- Prove the following sequent:
 - english_summer \(\) green_top \(\) fast_bowling, indian_batsmen_fail|-
 - english_summer ∧ fast_bowling ∧ indian_batsmen_fail



CS/IS F214 Logic in Computer Science

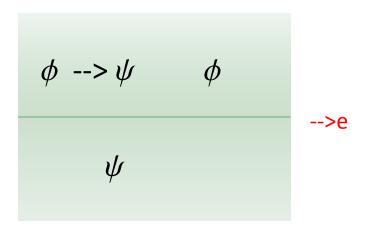
MODULE: PROPOSITIONAL LOGIC

Proof System - Natural Deduction - Implication Rules

18-08-2018 Sundar B. CS&IS, BITS Pilani 18

Natural Deduction: Proof Rules: Implication

Modus Ponens (Implication Elimination)



This rule is read as:

if we know ϕ implies ψ and we know ϕ is true then we can conclude ψ



Prove using "modus ponens":

rains, rains-->wet, wet-->slippery |-- slippery

The proof proceeds as follows:

	Deduction	Explanation	7	>elimination
1	rains	Premise		on premises in row 1 and 2
2	rains>wet	Premise		
3	wet	>e 1,2		
		•••	1 st	step



[continued]

Proof using "modus ponens":

rains, rains-->wet, wet-->slippery |-- slippery

	Deduction	Explanation
1	rains	Premise
2	rains>wet	Premise
3	wet	>e 1,2
4	wet>slippery	Premise
5	slippery	>e 3,4

2nd step

-->-elimination on premises in row 3 and 4

Prove:

rains, rains-->wet, wet-->slippery |-- wet ∧ slippery

	Deduction	Explanation	>-elimination on premises in
1	rains	Premise	row 1 and 2
2	rains>wet	Premise	>-elimination
3	wet	>e 1,2	on premises in
4	wet>slippery	Premise	row 1 and 4
5	slippery	>e 1,4	
6	wet ∧ slippery	∧i 3,5 <u></u>	
			↑ -introductionon premises inrow 3 and 5



Natural Deduction – Proofs

- Questions:
 - How do you identify the next step in a proof?
 - How do you identify which rule is to be applied?
- Observations:
 - An ND proof is usually driven bottom up i.e.
 - one identifies the steps (deductions) of the proof by starting with the conclusion
 - The structure of the conclusion (i.e. the top level operation(s) used) leads to the identification of the appropriate rule
 - for instance the choice of &i rule vs. the choice of
 -->e rule in the two proofs in the last two slides.



ND – Structure of Proofs - Example

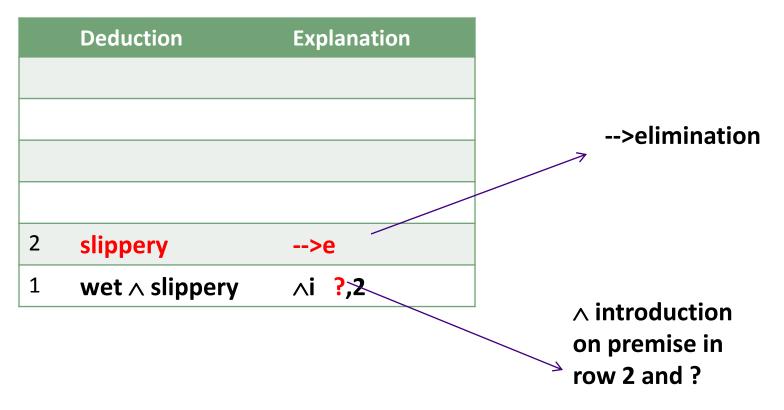
Prove: rains, rains-->wet, wet-->slippery |-- wet ∧ slippery

Deduction	Explanation	
wet ∧ slippery	^i	



Prove:

rains, rains-->wet, wet-->slippery |-- wet ∧ slippery





Prove:

rains, rains-->wet, wet-->slippery |-- wet ∧ slippery

	Deduction	Explanation	
4	wet	?	>-elimination on Premise in
3	wet> slippery	Premise	row 3 and resu
2	slippery	>e 3,4	in row 4
1	wet ∧ slippery	∧i ?,2	△ -introduction
			on premise in row 2 and ?

Prove:

rains, rains-->wet, wet-->slippery |-- wet ∧ slippery

	Deduction	Explanation
6	rains	Premise
5	rains>wet	Premise
4	wet	>e 5,6
3	wet> slippery	Premise
2	slippery	>e 3,4
1	wet ∧ slippery	∧i ?,2

-->-elimination on Premises in rows 5 and 6



Prove:

rains, rains-->wet, wet-->slippery |-- wet ∧ slippery

	Deduction	Explanation
6	rains	Premise
5	rains>wet	Premise
4	wet	>e 5,6
3	wet> slippery	Premise
2	slippery	>e 3,4
1	wet ∧ slippery	∧i 4,2

-->-elimination on Premise in row 4 and result in row 3

> ∧ -introduction on premise in row 2 and result in row 4

ND: Proof Rules: Implication Introduction

A "Proof" introduces implication

Assume ϕ 1

.

 $\phi 1 --> \phi 2$

i.e. if we can assume $\phi 1$ and prove $\phi 2$, then $\phi 1 --> \phi 2$ can be inferred

Question:

How is this rule different from the ones seen before?

[Hint: *Note that there is a sub-proof*. End of Hint.]



29

ND: Proofs using Implication Introduction

- Example:
 - Prove the sequent $|--p \land q --> q$
 - Proof:

	Deduction	Explanation
1	p∧q	Assumption
2	q	∧e ₂ - 1
3	p ∧ q> q	>i - 1-2

These two rows
constitute a proof
of the sequent
p \ q |-- q

Observation:

Proof of a sequent of the form $\phi \mid -- \psi$ can be treated as the proof of a sequent of the form $\mid -- \phi \mid --> \psi$



ND: Proofs using Implication Introduction

Observation:

- Proof of a sequent of the form $\phi \mid -- \psi$ can be treated as the proof of a sequent of the form $\mid -- \phi --> \psi$
- Inductively, proof of $\phi 1$, $\phi 2$, ... $\phi n \mid --\psi$ is proof of $\mid --\phi 1 --> (\phi 2 --> (... --> (\phi n --> \psi)...))$

Exercise:

Prove the following sequent:

and thereby prove the following sequent, using the observation mentioned above:

