Formal Proofs and Boolean Logic II

Extending $\mathcal{F}$ with Rules for ¬

William Starr

02.24.09

Announcements

For 02.24

1. HW5 is due now
2. Grades for HW1 & HW3 are posted in the Sakai gradebook
3. Do the Homework, it's really important!

Outline

Review

Proof by Contradiction

- Last class, we learned how to do formal proofs for ∨ and ∧
  - But what about the rules for ¬?
- That's the topic of Today's class
- Our ¬ **Intro** rule will allow us to prove negated claims
- As always, our formal rule mirrors our informal proofs
- So let's review our informal method for proving negated claims

Proof by Contradiction
Proving a Negated Claim

Proof by Contradiction (Official Version)

1. To prove that P is false, show that a contradiction $\bot$ follows from P
2. To prove that P is true, show that a contradiction $\bot$ follows from $\neg$P

Proving a Negated Claim

To prove $\neg$P, assume P and prove a contradiction $\bot$

- All contradictions are impossible, thus false
- If you can show that P leads to a contradiction, then P must be false
- But if P is false, then $\neg$P must be true

Review
What is a Contradiction Again?

Contradiction

- A contradiction is any sentence that cannot possibly be true, or any group of sentences that cannot all be true simultaneously
- The symbol $\bot$ is often used as a short-hand way of saying that a contradiction has been obtained

- Examples:
  1. $\neg$Cube(a) $\wedge$ Cube(a)
  2. a = b, b = c, a $\neq$ c
  3. Cube(a) $\wedge$ Tet(a)

Proof by Contradiction
A Simple Example

Claim: This argument is valid

$$\begin{array}{|l} \neg\text{SameShape}(a, b) \\ b = c \\ \hline \neg a = c \end{array}$$

Proof: We want to show $\neg$a = c from the premises, so we will use a proof by contradiction

1. Suppose a = c
2. Then, from premise one $\neg$SameShape(c, b) follows by Indiscernibility of Identicals
3. But by premise two, we know SameShape(c, b). This is a contradiction, $\bot$!
4. So our supposition must have been false; that is, $\neg$a = c must be true given the premises

Formal Rules for ¬
Where We Are Going

- The basic idea behind $\neg$ **Intro** is familiar from our informal method of proof by contradiction
  - You can use $\neg$ **Intro** to infer $\neg$P when you have proven that a contradiction $\bot$ follows from P
- So, to formulate this rule, we must think clearly about what exactly should count as proving $\bot$
- When have we satisfactorily shown that a contradiction follows from what we are assuming?
- This suggests that we also need a rule handling the introduction of $\bot$
- The next question is then how to formulate such a rule

Contradictions Again

Varieties Contradiction

- Before looking at ⊥ **Intro** & ⊥ **Elim**, we need to recognize a distinction between two kinds of contradictions
  1. **Boolean Contradictions**
     For example:
     - Cube(a), ¬Cube(a)
     - Tet(a) ∧ ¬Tet(a)
  2. **Analytic Contradictions**
     For example:
     - Large(a), Small(a)
     - FrontOf(a, b), BackOf(a, b)
- With these details fresh in our heads, we'll add two rules to $\mathcal{F}$ for handling contradictions

Contradictions

⊥ Intro

⊥ Intro

	P
	⋮
	¬P
▷	⊥

- If you've proven a sentence and its negation:
  - You can introduce the ⊥ symbol to indicate that a contradiction has been derived
- **Important**: This rule only detects Boolean contradictions
  - If you had FrontOf(a, b) and ¬BackOf(b, a) you could not use this rule to introduce ⊥
- Fitch is more liberal than ⊥ **Intro**
  - It allows Analytic contradictions to justify introduction of ⊥ with **Ana Con**

Boolean v. Analytic Contradictions

Within $\mathcal{F}$

Boolean ⊥ in $\mathcal{F}$

1	Cube(a)	
2	¬Cube(a)	
3	⊥	⊥ **Intro**: 1, 2

- We have P and ¬P
- So ⊥ **Intro** allows us to introduce ⊥

Analytic ⊥ in Fitch

1	Cube(a)	
2	Tet(a)	
3	⊥	**Ana Con**: 1, 2

- Here we do **not** have P and ¬P
- So ⊥ **Intro** does not give us ⊥
- But **Ana Con** does

⊥ Elim

What Should ⊥ Elim Be?

- Remember, all rules come in pairs
- We've stated ⊥ **Intro**, but we haven't said anything about ⊥ **Elim**
- What **should** we be able to infer from a contradiction?
- Let's think about it for a minute

Valid Arguments

What If The Premises are Inconsistent?

Logical Consequence, Validity

C is a logical consequence of $P_1, \ldots, P_n$ if and only if it is impossible for $P_1, \ldots, P_n$ to be true while C false

- What follows from a contradiction?
  - Anything!
- Why?
  - It's impossible for it to be true
- So, it is impossible for it to be true while any conclusion is false!

Contradictions

⊥ Elim

⊥ Elim

| ⊥
| ⋮
▷ | P

- From a contradiction ⊥, any conclusion follows!
- Why again?

- An inference step is valid just in case it cannot lead you from a true premise to a false conclusion
- Since the premise ⊥ in this inference can never be true, the inference can never lead one from a true premise to a false conclusion

Contradictions

Wait, What were We Doing?

- So, now we have two more rules in $\mathcal{F}$: ⊥ **Intro**, ⊥ **Elim**
- But wait, why did we start talking about ⊥ in the first place?
- Because contradictions are essential to formulating our ¬ **Intro** rule
  - Recall that proving ¬P involves assuming P and showing that a contradiction follows from this assumption
- So now we'll learn ¬ **Intro**

¬ Intro

From Informal to Formal Proof

Proving a Negative Claim

- To prove ¬P, assume P and prove a contradiction using this assumption
- This is an example of **Proof by Contradiction**

Example Informal Proof

From a = b and b ≠ c we will prove a ≠ c. We use proof by contradiction. **Proof**: Suppose a = c. Well, b = c follows from this assumption and premise one by Ind. of Id.'s. But, this contradicts premise two, ⊥. So our assumption was wrong, in which case a ≠ c.

¬ Intro

| | P
| | ⋮
| | ⊥
▷ | ¬P

To prove ¬P:

1. Assume P
2. Derive ⊥ (using ⊥ **Intro**)
3. Conclude ¬P
   (Discharging assumption of P)

¬ Intro
An Example

1	$a = b$	
2	$b \neq c$	
3	| $a = c$	
4	| $b = c$	$=$ **Elim**: 3, 1
5	| $\bot$	$\bot$ **Intro**: 2, 4
6	$a \neq c$	¬ **Intro**: 3-5

✓Goal: $a \neq c$

¬ Intro

| P
| ⋮
| ⊥
▷ ¬P

⊥ Intro

P
⋮
¬P
▷ ⊥

Some More Examples

- Let's do a formal proof for **6.25**:

$\neg A \wedge \neg B$

$\neg(A \vee B)$

- Let's also finish the proof from slide 26 of 02.19
  - This will use ⊥ **Elim**

¬ Intro
Another Example

ARGUMENT 1: Analytic ⊥ Revisited

1	¬SameShape(a, b)	
2	$b = c$	
3	| $a = c$	
4	| ¬SameShape(c, b)	$=$ **Elim**: 1,3
5	| $\bot$	**Ana Con**: 2, 4
6	$a \neq c$	¬ **Intro**: 3-5

Goal: $a \neq c$

Informal Proof

We want to show $a \neq c$, so we use proof by contradiction. **Proof**: Suppose $a = c$. From premise one it follows that ¬SameShape(c, b), by Ind. of Id. But this contradicts premise two which requires that *c* **is** *b*. So our assumption ($a = c$) was wrong, hence $a \neq c$ follows.

¬ Intro

| P
| ⋮
| ⊥
▷ ¬P

- There is no rule in $\mathcal{F}$ which justifies line 5
- But this is what we need to prove $a \neq c$!
- So, this proof can't be finished in $\mathcal{F}$
- We *can* finish it in Fitch!

Negation
¬ Elim

¬ Elim

¬¬P
⋮
▷ P

Simple Example

1	¬¬Cube(a)	
2	Cube(a)	¬ **Elim**: 1

- If ¬¬P is true, so is P
- This may seem like an obvious and useless rule, but it is in fact quite useful
- ¬ **Intro** allows us to use proof by contradiction to prove P
  - We simply use ¬ **Intro** to prove ¬¬P and then apply ¬ **Elim**

¬ Elim
An Example

Argument 2

Tet(e) ∨ Cube(a)
¬Tet(e)
Cube(a)

Informal Proof of Argument 2

We will use a proof by contradiction. Suppose ¬Cube(a). This pretty clearly contradicts the first premise. To be sure, we'll take it in cases. In the first case (Tet(e)) the contradiction is clear. In the second case (¬Cube(a)) we also have a contradiction. So our assumption must have been wrong. Hence, Cube(a) must be true given the premises.

Let's make this into a formal proof in Fitch

Subproofs
The Big Picture

- Subproofs correspond to things we do in informal proofs:
  - They correspond to the cases of a proof by cases
  - They correspond to the temporary assumption in a proof by contradiction
- Just like cases and temporary assumptions, there are certain important restrictions on subproofs

Cases
The Constraints

Argument 3

(Cube(c) ∧ Small(c)) ∨ (Tet(c) ∧ Small(c))
Small(c) ∧ Cube(c) ∧ Tet(c)

Pseudo-Proof of Argument 3

We will use a proof by cases based on premise one. **Case 1**: Suppose (Cube(c) ∧ Small(c)). Then Small(c) follows. **Case 2**: Suppose Tet(c) ∧ Small(c). Then Small(c) follows. So, Small(c) follows in either case. But in case 1 we had Cube(c) and in case 2 we had Tet(c), hence our conclusion follows: Small(c) ∧ Cube(c) ∧ Tet(c).

- Why pseudo-proof?
  - Argument 3 is not valid
  - This 'proof' leads us from a possible premise to an impossible conclusion
  - That's exactly what proofs aren't supposed to do

Cases
The Constraint

Pseudo-Proof of Argument 3

We will use a proof by cases based on premise one. **Case 1**: Suppose (Cube(c) ∧ Small(c)). Then Small(c) follows. **Case 2**: Suppose Tet(c) ∧ Small(c). Then Small(c) follows. So, Small(c) follows in either case. But in case 1 we had Cube(c) and in case 2 we had Tet(c), hence our conclusion follows: Small(c) ∧ Cube(c) ∧ Tet(c).

- Where exactly does this proof go wrong?
  - When we reached back into cases that were already finished
  - The assumptions and conclusions of a case are only live within that case, once you are done, you are done
  - What happens in a case, stays in a case
  - Although we can cite the case as a whole, as in proof by cases, we can't pick claims out of it to use later in the proof

Temporary Assumptions
The Constraints

- In proof by contradiction, like in proof by cases, we make a temporary assumption:
  - We assume P and try to show $\bot$
  - But P is a temporary assumption
  - And anything we infer from it is also temporary
  - Once we show $\bot$, we discharge the assumption of P
  - This temporary assumption of P and the things we infer from it corresponds to a subproof
  - Once this assumption is discharged, we can't reach back into the subproof, only cite the subproof as a whole

Subproofs
Drawing the Connections

Proof with A Subproof

$$\begin{array}{l|l} & \vdots \\ & \begin{array}{|l} \underline{A} \\ \vdots \\ B \end{array} \\ & \vdots \\ & C \end{array}$$

- A subproof involves a temporary assumption
  - Just like proof by contradiction
  - Just like proof by cases
- And just like the temporary assumptions of our informal proof methods:
  - Once the temporary assumption has been discharged, you cannot use any of the individual lines of the subproof to justify additional steps

Subproofs
Guidelines for Use

Guidelines for Using Subproofs

1. Once a subproof has ended, you can never cite one of its lines individually for any purpose, although you may cite the subproof as a whole (as in $\lor$ **Elim** & $\neg$ **Intro**)
2. In justifying a step of a proof, you may cite any earlier line of the main proof, or any subproof that has not ended

Let's do exercise 6.17 to solidify these points

Proof Strategies

How to Approach a Formal Proof

1. Understand what the sentences are saying
2. Decide whether you think the conclusion follows from the premises
3. If you don't think so, try to find a counterexample
4. If you do think so, try to give an informal proof
5. Use this informal proof to guide your formal proof
6. If you get stuck try working backwards

Summary
Negation

- We learned two new negation rules: ¬ **Intro**, ¬ **Elim**
- ¬ **Intro** mirrors the proof by contradiction method
- To mimic this method in $\mathcal{F}$ we introduced the ⊥ symbol and two rules for it: ⊥ **Intro**, ⊥ **Elim**
- Proof by contradiction isn't just good for proving negated claims
  - It can also be used to prove positive claims

Summary
Subproofs & Strategy

- Mastering $\mathcal{F}$ involves mastering subproofs
- Just like cases and reductio assumptions, there are constraints on how you can use subproofs
- We learned these constraints and the perils the guard us against
- We also learned how to approach proofs
  - There's strategy to it!
  - Don't just try to shuffle symbols!