

### 3.3.2. Using lemmas to complete *reductios*

Now that we have IP, we are in a position to provide a proof for any argument whose validity depends only on the properties of  $\top$ ,  $\perp$ , conjunction, and negation. However, to do this using only the rules we have so far, we would often need to use LFR—or, in simpler cases, Adj—to make use of negative resources. This poses no problems when we construct derivations for valid arguments, but it makes it difficult to show that any argument is not valid. LFR does not itself exploit resources, so negated compounds remain as active resources until a gap is closed. In order to count an open gap as having reached a dead end, we would need some description of the conditions under which LFR had been used often enough. Such a description could certainly be given; and, in the last two chapters, we will need to take an analogous approach in the case of one of the rules for quantifiers. But, in the case of negation, it is possible to keep track of the use of resources by way of a genuine exploitation rule.

The basis for such an approach was the third lesson drawn from the law of negation as a premise: if a *reductio* that has  $\neg \varphi$  as a premise is valid—that is, if  $\Gamma, \neg \varphi \Rightarrow \perp$ —then  $\varphi$  is a valid conclusion from the premises other than  $\neg \varphi$  (i.e.,  $\Gamma \Rightarrow \varphi$ ). And  $\varphi$  is just the lemma we need in order to use the premise  $\neg \varphi$  to complete the *reductio*. That means not only that it is safe to introduce  $\varphi$  as a lemma but also that the gap in which we establish the lemma need not contain  $\neg \varphi$  among its active resources. Of course,  $\neg \varphi$  is needed along with the lemma to reach the goal  $\perp$ , but there is no need to introduce a second gap in which we try to reach this goal (as would be done with LFR) because such a gap would close immediately by Nc. And, indeed, the law for negation as a premise tells us not only that we can reach the needed lemma (provided that the *reductio* is valid to begin with) but also that reaching this lemma is all that we need to do, for it says both that the *reductio* argument  $\Gamma, \neg \varphi / \perp$  is valid only in cases when the argument  $\Gamma / \varphi$  is valid and that  $\Gamma, \neg \varphi / \perp$  is valid in all such cases.

We will call the rule that implements these ideas **Completing a Reductio** (CR).

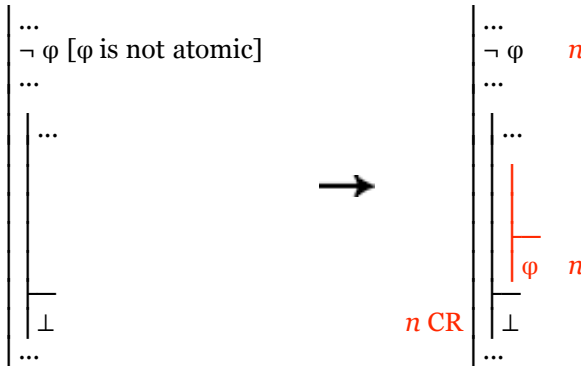


Fig. 3.3.2-1. Developing a derivation by exploiting a negated compound at stage  $n$ .

The motivation for this rule lies in its use with the negations of non-atomic sentences; and, in fact, we must limit its use to such sentences. It is sound and safe in the case of atomic sentence, but it would not be progressive in that case (given the way we are measuring distance from a dead end) because it would replace a resource that we never exploit by a goal that we could go on to plan for by IP; that is, it would provide new opportunities for developing a derivation and thus send us farther from reaching a dead end. Both IP and CR carry us between gaps whose proximate arguments have the forms  $\Gamma, \neg \varphi / \perp$  and  $\Gamma / \varphi$ ; but they carry us in opposite directions, so, if there is any overlap in the sentences  $\varphi$  to which they apply, a derivation could move back and forth between the two arguments forever. We block such circles by limiting IP to cases where  $\varphi$  is atomic and limiting CR to cases where  $\varphi$  is non-atomic.

The following derivations show, on the left, the use of CR in a derivation for the argument from [3.2.2](#) that was used as an illustration in the last subsection and, on the right, an analogous use of LFR:

$\frac{\frac{\frac{\neg((A \wedge B) \wedge \neg C) \quad 2}{A \quad (3)}{B \quad (3)}}{\neg C \quad (4)}}{\frac{\frac{\frac{A \wedge B \quad X,(4)}{(A \wedge B) \wedge \neg C \quad X,(5)}}{\bullet}}{(A \wedge B) \wedge \neg C \quad 2}}{\perp \quad 1}}{C}$ <p style="margin-left: 20px;">3 Adj 4 Adj  5 QED 2 CR 1 IP</p>	$\frac{\frac{\frac{\frac{\neg((A \wedge B) \wedge \neg C) \quad (6)}{A \quad (3)}{B \quad (3)}}{\neg C \quad (4)}}{\frac{\frac{\frac{A \wedge B \quad X,(4)}{(A \wedge B) \wedge \neg C \quad X,(5)}}{\bullet}}{(A \wedge B) \wedge \neg C \quad 2}}{\frac{\frac{(A \wedge B) \wedge \neg C \quad (6)}{\bullet}}{\perp \quad 2}}{\perp \quad 1}}{C}$ <p style="margin-left: 20px;">3 Adj 4 Adj  5 QED  6 Nc 2 LFR 1 IP</p>
---	--

The most obvious difference between the two is an extra argument in the second in which the lemma is  $(A \wedge B) \wedge \neg C$  is used explicitly. But the more important difference is that, while the first premise is exploited at stage 2 in the left-hand derivation, it remains unexploited in the second. It is true that the absence of the second gap introduced with LFR makes the first derivation shorter, but that is not its chief virtue. Derivations are not designed to be the most efficient way of reaching a conclusion, and the left-hand derivation is a little longer than the one at the end of [3.3.1](#). The extra length in comparison with that derivation is the result of introducing  $(A \wedge B) \wedge \neg C$  as an explicit goal. The derivation using CR is shorter than the one using LFR because CR applies only in cases where we know exactly how to use the lemma to complete the *reductio*. It is longer than the derivation using just Adj because it provides us with a stage at which we can mark its negation as exploited and makes explicit the resource aimed at by the two uses of Adj.