2.3. Failed proofs and counterexamples

2.3.0. Overview

Derivations can also be used to tell when a claim of entailment does not follow from the principles for conjunction.

2.3.1. When enough is enough
A derivation is stopped only when no more rules can be applied. When that is so, any open gap has reached a dead end.

2.3.2. Dead ends and counterexamples
The active resources of any dead-end gap can be separated from its goal. To put it another way, we can run out of ways to develop an open gap only when there is a counterexample to its proximate argument.

2.3.3. Validity through the generations
If we describe as descendents of a gap the gaps that result from developing and perhaps dividing it, the validity of the proximate argument of a gap rests on the validity of the proximate arguments of its descendents.

2.3.4. Sound and safe rules
The derivation rules are designed so that, if a gap has a counterexample, so does at least one descendent at every stage and, moreover, each of its ancestors.

2.3.5. Confirming counterexamples
Because we have enough rules and the ones we have are well-behaved, any gap that reaches a dead end shows us how to separate the premises of the initial argument from its conclusion.

2.3.6. Reaching decisions
A derivation will always reach a point where we must stop either because all gaps are closed or because there is an open gap to which no more rules can be applied.

2.3.7. Soundness and completeness
The properties of this system of derivations combine to show that it does not indicate validity for any argument that is not valid and does indicate validity for every argument that is valid.

2.3.8. Formal validity
The sort of validity we test with derivations is the general validity of arguments with a given form. An argument that is not valid in virtue of a given form could be valid nonetheless due to features not captured by that analysis.

2.3.1. When enough is enough

So far we have seen only derivations whose gaps all close, derivations which show that arguments are valid. But not all arguments are valid, so there ought to be derivations whose gaps do not all close. And if there is no point at which the gaps of a derivation all close, we will eventually have to give up work on it even though it still has open gaps. So we should ask if there can be a reason for giving up work and what, if anything, we can conclude about an argument if we have grounds for giving up work on a derivation for it.

The short answer to the first of these two questions is that we must give up on a derivation when we run out of rules to apply, either to develop a gap or close it. Here's a simple example of a derivation for which that has happened.

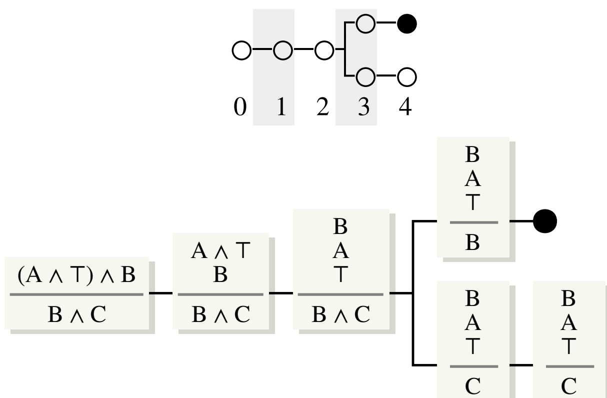
	$(A \wedge \top) \wedge B$	1
1 Ext	$A \wedge \top$	2
1 Ext	B	(4)
2 Ext	A	
2 Ext	$\top$	
	●	
4 QED	B	3
	○	$B, A, \top \nvDash C$
	C	3
3 Cnj	$B \wedge C$	

The gap that is marked with the empty circle ○ has C as its goal, and we currently have no rule to plan for such a goal. There are conjunctions among the available resources of the gap; but they were exploited in the course of developing this gap, so they are no longer active. Also, none of the rules for closing gaps apply here: not QED because the goal is not one of available resources, not EFQ because ⊥ is not a resource, and not ENV because the goal is not ⊤. In short, no rule of any of the three sorts can be applied at this point.

The resources added by exploiting $A \wedge \top$ at stage 2 were never used later (hence there are no line numbers to their right). As a result, this exploitation could have been postponed the end. But, once it is done, there is no more to do. However, although it is easy to see that it will not lead the gap to close, the resource $A \wedge \top$ would have to be exploited before we could claim to have ended because no more that can be done. Until it is exploited, there is a way of developing the derivation further. One thing that needn't have been done is closing the first gap at stage 4. As long as one gap has reached a point where no more can be done to close it, there is reason to stop because all gaps must close before the derivation is complete.

We will describe an open gap to which no more rules apply as a *dead-end gap*. (Although the qualification dead-end will be reserved for open gaps—indeed, a gap that has been closed is in one sense no longer a gap—we will often speak somewhat redundantly of “dead-end open gaps.”) In these terms, we can say that we are forced to abandon a derivation when every open gap has reached a dead end although we *may* abandon a derivation as soon as one open gap has reached a dead end. As in the example above, we will use the empty circle to mark open gaps that have reached a dead end and are thus permanently open. And, also as is done in that example, to the right of this sign, we will use the sign ⊭ (*negated double right turnstile*) to say that, with respect to the analysis of them displayed in the derivation, the active resources do not entail the goal. (The reason for qualifying this by reference to the displayed analysis will be discussed in 2.3.8.)

The way the gaps have developed in this derivation is shown in the following skeleton tree with the full argument tree below it:



The gap that remains open at the end had reached a dead end at stage 3, but it is shown to continue at the next stage because it remains open as the derivation develops elsewhere. Although work may be stopped as soon as a dead end is reached, there is nothing wrong with continuing as long as there are rules to be applied to other gaps, and that will often be done in examples. In general we will not assume that work on a derivation stops as soon as there is a dead-end gap, so to say that gap has reached a dead end is not to say that the gap does not continue at later stages; it is to say rather that it cannot be developed further.

We will now turn to considering the significance of dead end gaps. We will look first at what reaching a dead end tells us about the proximate argument of the gap that has stopped developing and then consider the connection between the validity of the ultimate argument of a derivation and the existence of dead-end gaps. In terms of argument trees, this means we will look first at the tips of unclosed branches and then ask about the connection between the tips of branches and the root of the tree.

2.3.2. Dead ends and counterexamples

Now, let's look more closely at what we can say in general about the significance of dead-end open gaps. First of all, recall what led us to conclude that the gap in the example of the last section could not be developed further. A dead-end gap must not have a conjunction either as its goal or among its active resources, for otherwise we could apply the rules Cnj or Ext. Moreover, it must not have ⊤ as a goal or ⊥ as a resource, or else we could apply the rules ENV or EFQ. Finally, its goal must not be among its resources because then we could apply the rule QED. So the active resources of dead-end gaps are limited to unanalyzed components and ⊤ and their goals are limited to unanalyzed components and ⊥; and no dead-end gap can contain an unanalyzed component both as an active resource and as its goal.

This means that we can assign truth-values to the unanalyzed components appearing in a dead-end gap in a way that makes its active resources true and its goal false. Since no unanalyzed component appears both as a resource and as the goal, we can make any that appears as a resource **T** and any that appears as the goal **F**. While we are not free to assign values to ⊤ and ⊥, the first can appear only as a resource and the second only as the goal so they will not keep us from having true resources and a false goal. In short, we can assign truth values in a way that separates the active resources of a gap from its goal. That is, there is a counterexample to the gap's proximate argument.

In noting this, we described an assignment of truth values to unanalyzed sentences. This is an extensional interpretation in the sense discussed in 2.1.8, and it can be presented in a table. The following table displays the interpretation defined by the dead-end gap of the example we have been considering.

A	B	C	B,	A,	⊤	/	C
T	T	F	Ⓣ	Ⓣ	Ⓣ		Ⓕ

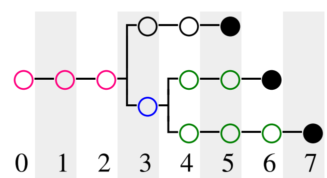
The extensional interpretation of unanalyzed components appears on the left of the table. On the right are the resulting truth values of resources and goals of the gap, which mainly just repeat the assignments. (No value is assigned to ⊤ on the left because its truth value is stipulated by the meaning of the sign. That is, the sentence ⊤ has its value by general stipulation rather than by assignment in specific interpretations.)

Although we have more to show before we know that the system of derivations does what it is supposed to, we can say already that it has enough rules in a certain sense, for we know that, whenever the proximate argument of a gap is valid, some rule can be applied to either develop or close the gap. For if there is no rule allowing us to develop the gap, it has reached a dead end, and we have just seen that the proximate argument of a dead-end gap is not valid. We will indicate this sort of completeness in our rules by saying that a system of derivations is *sufficient* when every dead-end open gap has its active resources separated from its goal by some extensional interpretation. Of course, in saying that our system is sufficient, we do not say that every gap whose proximate argument is invalid has already reached a dead end. We would not expect this to be true since it would mean that we would never need to apply any rules at all in the case of an invalid argument. Indeed, one of the things we have yet to show is that any gap whose proximate argument is not valid will eventually reach a dead end.

2.3.3. Validity through the generations

The connection between the proximate arguments of dead-end gaps and the ultimate argument of a derivation lies in the properties of the rules for developing and closing gaps. We will begin to look at these properties in this section and then look at them more closely in the next.

It will help to have some ways of talking about the relations between gaps at various stages of a derivation. It is common to extend some genealogical vocabulary from family trees to trees in general. In our use of this vocabulary, we will say that any gap that results from applying a rule is a *child* of the gap to which the rule is applied and that the latter gap is its *parent*. It will be convenient to apply the same terminology to gaps that continue unchanged while others develop: a gap at one stage that is open but unchanged at the next stage is understood to have a single child. Looking farther up or down a line of descent, we will say that some gaps are *ancestors* or *descendants* of others. So in the tree of gaps associated with the derivation discussed in 2.2.6,



the lower gap at stage 3 has the gap at stage 2 as its parent and both that and the two earlier gaps as ancestors. Its children are the lower two gaps at stage 4 and its further descendants are the gaps to their right. The line of gaps at the top are neither ancestors or descendants of the gap in question.

In this terminology, the initial gap of a derivation is an ancestor of all gaps of all gaps at each later stage in its development; and they are all its descendants. Only open gaps will be part of these genealogies, so a gap that is closed at the next stage of its development has no children. Dead-end open gaps continue to have children if the derivation is continued at later stages (remember it need not be), yet they have reached a dead end in the sense that these children are always identical to their parents.

Next, let us develop a way of speaking about the effect of derivation rules on the distribution of valid and invalid arguments in the argument tree of a derivation. In the case of QED, we will initially limit ourselves to its use to close a gap whose goal is also among the active resources. (The wider use of QED to close gaps whose goals are among their available but inactive resources will be considered later.)

The derivation rules Ext and Cnj are based on principles of entailment which give necessary and sufficient conditions for an entailment to hold. That is, each principle gives a list of conditions all of which must hold if a given entailment is to hold and which together are enough to insure that it holds. It may seem odd to say the same about the unconditional claims of entailment that lie behind the rules QED, ENV, and EFQ; but, by asserting an entailment unconditionally, they say that an empty list of conditions is sufficient for its truth (and, since an empty list cannot have a member that fails to hold, satisfying the list is trivially necessary since it is bound to be satisfied).

Phrased in terms of arguments, each principle thus tells us that a certain sort of argument is valid if and only each member of a (perhaps empty) list of arguments is valid. When the corresponding rule is applied to a gap, the gap is provided with children whose proximate arguments are those on the list if the list is not empty, and the gap is closed if the list is empty. This is shown for individual rules in the following table:

rule	*prox. arg. of parent*	*prox. args. of children*
Cnj	Γ / $\varphi \wedge \psi$	Γ / φ Γ / ψ
Ext	$\Gamma, \varphi \wedge \psi$ / χ	Γ, φ, ψ / χ
QED	Γ, φ / φ	(none)
ENV	Γ / $\top$	(none)
EFQ	$\Gamma, \bot$ / φ	(none)

In general, we can that the proximate argument of a gap to which a rule is applied is valid if and only if all the proximate arguments of any children given by the rule are valid. When a parent gap acquires a child due to development of another gap, it then acquires only one child and the proximate argument of this child is the same as the parent's, so in this case, too, the proximate argument of the parent if and only if the proximate argument of each child is valid. Putting these two cases together, we can say this:

> For any pair of immediately successive stages of a derivation, a gap at the first stage has a valid proximate argument if and only if every child of it at the next stage has a valid proximate argument.

Here the claim every child … should be understood to be true when the gap has no children to provide counterexamples to this generalization. And this is the reason for the limitation to cases of a pair of successive stages, the fact that a stage has no children tells us nothing about its validity if it has no children merely because the derivation hasn't yet been developed beyond that point. On the other hand, there is no need to limit this claim stages that are *immediately* successive. For what we have seen about children applies equally to grandchildren, great-grandchildren, and so on.

It may be easier to see that if we turn things around and look the conditions under which proximate arguments *fail* to be valid. In order to have a more compact way to talk about that, let us say that an interpretation that is a counterexample to a gap's proximate argument *lurks* in the gap and that the gap has a *lurking counterexample*. So the proximate argument of a gap fails to be valid just in case there is a counterexample lurking in the gap. And the principle above then comes to the same thing as the following:

> For any pair of immediately successive stages of a derivation, a gap at the first stage has a counterexample lurking in it if and only if some child of it at the next stage has a counterexample lurking in it.

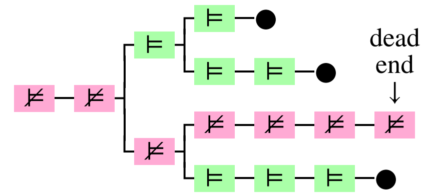
This is equivalent to the earlier principle because saying that some child has a lurking counterexample is the same as denying that every child has a valid proximate argument.

Now suppose a gap is followed by two successive further stages. What we have said regarding any children at the first of these applies to grandchildren at the second. For, if a counterexample lurks in the gap, we have seen that a counterexample must lurk in some child and, for the same reason, in some child of that child. And if a counterexample lurks in a grandchild, one must lurk in a child, and therefore one must lurk in the gap itself. The same argument applies to further succeeding stages, so we can say this:

> For any pair of stages, one earlier than the other, a gap at the first earlier has a counterexample lurking in it if and only if some descendent of it at the later stage has a lurking counterexample.

We still speak only of gaps for which there is a succeeding stage, but that is enough to tie the validity of the initial gap's proximate argument with the state of the derivation after all work is done. And, when all work is done, we know from the last subsection that any remaining open gaps, which will have reached dead ends, must have counterexamples lurking in them.

The diagram below shows how we can put these ideas together. It displays a sort of schematic argument tree that does not indicate actual proximate arguments, only their validity or invalidity—that is, whether or not there is a counterexample lurking in the gap. It is intended to depict a derivation that has come to an end, so the one gap that remains open at the right is understood to be a dead end.



We can distinguish three sorts of cases in this tree. First of all, we know from the sufficiency of the rules that the dead-end gap has a counterexample lurking in it. It has no descendent with a lurking counterexample, but that doesn't conflict with the principle above because there is no later stage. Next, all ancestors of the dead-end gap, right down to the root of the tree, must have lurking counterexamples because each has a descendant that does. And finally, in the case of any of the other gaps—i.e., the ones whose proximate arguments are valid—there is a following stage (the last stage of the derivation if not an earlier one) at which the gap has no descendant at all, and so certainly has no descendant with a lurking counterexample. Also, notice that, at stages where such a gap does have descendants, all its descendents have valid proximate arguments. (There is a fourth sort of case that does not appear here, a gap that has no descendants but has not been closed and is not at a dead end, but this case will appear only in the last stage of an incomplete derivation.)

We now know that the way we have taken the results of a derivation is correct. If there is a dead-end gap—and thus, by sufficiency, a gap with a lurking counterexample—the initial gap must have a counterexample lurking in it, so the ultimate argument is invalid. On the other hand, if all gaps close, there is a stage (the one at which the last gap closes) at which the initial gap has no descendants, so it must have no lurking counterexample and the ultimate argument must be valid. Although the principle we have been using does represent an important property of the system of derivations, we will not label this property (in the way we have labeled the property of sufficiency) because we will go on in the next section to look further at the basis for it and state (and label) some related properties that can be applied to a wider range of rules, including the extended use of QED that we excluded from consideration here.

2.3.4. Sound and safe rules

The necessary and sufficient conditions for the validity of proximate arguments that were developed in the last section were based on connections between the presence of lurking counterexamples at successive stages. In this section, we will look more closely at the rules and consider not merely how the fact that counterexamples lurk in gaps is preserved as we develop a derivation but indeed how any lurking counterexamples are themselves preserved. This closer look at the effect of rules will enable us to give an account of a wider range of possible rules, including the extended use of QED that was not covered in our discussion in the last section.

We begin by considering two properties a rule *R* might have:

R is *strict*	*when*	any interpretation of the derivation that is a counterexample lurking in a gap to which the rule *R* is applied also lurks in some child of the gap
R is *safe*	*when*	any interpretation of the derivation that is a counterexample lurking in a child of a gap to which the rule *R* has been applied also lurks in the parent gap

When a rule is strict we never lose any lurking counterexamples as we apply the rule. When it is safe, we never gain any lurkers. Both definitions are stated for interpretations of the whole derivation because an interpretation that lurks in a child gap need not assign truth values to enough sentences to count as a lurking counterexample of the parent. However, every way of the interpreting the vocabulary of the proximate argument of a gap can be found in some interpretation of the derivation as a whole, so the restriction to interpretations of the whole derivation does not really limit the scope of the generalizations.

Although their association with the necessity and sufficiency of the same condition suggests a kind of parallel between them, these two properties do not have the same importance. Although we will see that strictness is a little more than we need to ask, any serious departure from strictness would undermine the central function of proofs: to establish validity. For then all gaps of a derivation might close even though the original argument was invalid. An unsafe rule would analogously undermine the use of derivations to establish invalidity because it would introduce the possibility that a derivation for a valid argument could lead us to a dead end. But the role of derivations in establishing invalidity is less central, and their full use in that way depends also on a property (discussed in 2.3.7) that will fail for rules to be considered in the last two chapters. This means that safety is dispensible, but no viable system of proof could completely dispense with strictness.

Moreover, moves corresponding to unsafe rules are an important part of explicit deductive reasoning. For example, a natural approach when we seek a way to prove a mathematical result is to introduce a lemma (in the sense is discussed in 1.4.7) as a stepping stone to a final result. If the lemma represents a significant step beyond the premises, it may be no more obviously a valid conclusion from the premises than is the final conclusion we hope to establish. The introduction of such a lemma can be described as a conjecture, and this conjecture may be wrong: the lemma may not be a valid conclusion from our premises even when the final conclusion is valid. In short, by seeking to reach our conclusion by way of this lemma, we may be entering a blind alley. This is just the sort of thing that would appear in the context of derivations as a dead-end open gap in a derivation whose initial argument is valid. So conjecturing a lemma can be thought of as a step in discovering a proof that is valuable but unsafe.

Another step in a proof that can be valuable but is unsafe is a decision to focus on only some of the information in one's premises. This might seem quite different from a conjecture; but, combined with rules we will consider in the next chapter, a rule allowing us to conjecture a conclusion could lead us into a situation in which the active resources entailed less than did the resources at an earlier stage with the same goal. Intuitively, to focus on only part of one's premises is to guess that this part will be as useful for reaching the conclusion as the whole would be, and this guess amounts to a conjecture.

Our interest in deductive reasoning is somewhat different from a mathematician's. We are aiming not at new and surprising conclusions but instead at fuller understanding of the steps by which deductive conclusions are reached. Consequently, we will not be considering the large deductive steps for which conjecturing lemmas is the only practical approach. We will make use of lemmas—and we will look at rules for doing so in 2.4—but the chief value of lemmas for us lies in a restricted range of cases where we can be sure that they are safe.

Earlier, we set aside uses of QED in which the goal of the gap we close is among its available resources but not among the active ones. To discuss such uses of QED, we need to consider a requirement that is less unyielding than strictness. The following property of a rule *R* is the one we will employ:

R is *sound*	*when*	any interpretation that it is a counterexample lurking both in a gap to which the rule *R* is applied and in all ancestors of this gap also lurks in some child of the gap

The difference lies in the added phrase `and all ancestors of this gap`. The addition makes soundness apparently weaker than strictness because, for soundness, we do not require that an interpretation lurk in a child gap whenever it lurks the parent but instead only when it also lurks in all ancestors of the parent. However, when all rules are safe, a rule that is sound is also strict. For, when all rules are safe, an interpretation that is a lurking counterexample for a gap will also lurk in all ancestors of the gap. Thus, when there is a difference between soundness and strictness, it lies in their handling of the spurious lurking counterexamples introduced by unsafe rules: with a strict rule, such an interpretation will continue to lurk in descendants while, with a sound rule, it might not. In particular, a strict rule would force us to bear the burden of proving an unsafe conjecture while a sound rule might allow us to substitute a different way of reaching our initial goal.

And even when not all rules are safe, soundness is enough to insure that the ultimate argument of a derivation is valid whenever all gaps close. For, if all rules are sound, we can be sure that any counterexample lurking in a gap and in all its ancestors will lurk also in some child and in all ancestors of this child (since these are just the parent and its ancestors). But any counterexample to the ultimate argument of a derivation also lurks in any ancestor to the initial gap (since it has none), so if all rules are sound, this interpretation will also lurk in some child and all its ancestors—and so on. That is, as with strictness, when all rules are sound, any counterexample to the ultimate argument must lurk in some descendant at each stage; therefore, if all gaps close, there can be no counterexample to the ultimate argument. In short, if a sound rule ignores any lurking counterexample, this counterexample is an interpretation which shows that some risky conjecture does not follow from the initial premises, not one that shows that the initial conclusion was invalid.

Now, for a gap-closing rule to be sound, it is enough that there be no interpretation that makes the goal of the gap it closes false while making true all active resources of the gap *and all active resources of the gap's ancestors*. This means that it is enough for us to soundly close a gap that its goal be entailed by its active resources together the active resources of its ancestors. With the rules we have so far, all available resources are included if we take the active resources of a gap together with the active resources of its ancestors. So it is sound to close a gap when the goal is among the available resources, and our extended use of QED is sound.

But we can be even more generous since, by the law for lemmas, adding to a collection of resources something that is entailed by them will not change what they entail. In short, we can state rules for closing gaps and have them be sound if the conclusion of the gap is among its active resources, is among the active resources of its ancestors, *or is something entailed by these resources*. The available resources of a gap always include its active resources and the active resources of its ancestors, but in 2.4.4 we will consider rules which add to the available resources certain conclusions entailed by these resources. And we have just seen that this sort of addition will not undermine the soundness of the extended use of QED.

Although we will sometimes need to distinguish soundness and safety (or even consider strictness) in later discussions, most often we will not. We will say that a system is *conservative* when its rules are all safe and sound (which, remember, comes to the same thing as being all safe and strict). So in a conservative system, lurking counterexamples are neither added nor lost as we develop a derivation, though they may be spread out among an increasing number of descendant gaps, something we will see illustrated in the next section's example.

2.3.5. Confirming counterexamples

A dead-end open gap is always has a counterexample lurking in it, and any counterexample lurking in it lurks already in the ultimate argument of the derivation. We will finish off derivations by exhibiting a counterexample lurking in a dead-end open gap and calculating the truth values of the original premises and conclusions on that interpretation. In the case of the example discussed in 2.3.1 , this calculation is shown in the following table:

A	B	C	(A	∧	⊤)	∧	B	/	B	∧	(⊤	∧	C)
T	T	F		T	T	Ⓣ				Ⓕ	T	F	

Here the values of unanalyzed components have not been repeated on the right, but they are used to calculate the values of compounds containing them, with the order of calculation being guided by parentheses. In performing this calculation we are confirming that the counterexample lurking in the dead-end gap really does constitute a counterexample to the ultimate argument; and we will say that, in constructing the table, we are *confirming a counterexample*. It will be our standard way of concluding the treatment of an argument whose derivation fails.

It is not always the case that all unanalyzed components of the ultimate argument all appear among the resources and goal of a dead-end gap. When unanalyzed components do not appear there, values must still be assigned to them in order for a truth value to be defined for each sentence in the ultimate argument; but it will not matter what value we assign to these further unanalyzed components. If an interpretation separates the active resources of a gap from its goal, any way we choose to extend it to unanalyzed components not appearing in the gap's proximate argument will still be a counterexample to that proximate argument and therefore also to the ultimate argument.

The example below is designed to illustrate this. Of the three interpretations shown, the first is a counterexample lurking in only the first dead-end gap (since it assigns the value **T** to the goal of the second dead-end gap), and the last lurks only in the second open gap (for a similar reason); but the middle one lurks in both open gaps. With 4 unanalyzed components, there are $2 \times 2 \times 2 \times 2 = 2^4 = 16$ possible interpretations, so there are 13 interpretations that do not lurk in either gap. The soundness and safety of our rules insures that the 3 interpretations shown below constitute counterexamples to the ultimate argument and that the other 13 do not.

	A ∧ B	1
1 Ext	A	
1 Ext	B	(4)
	○	A, B ⊭ C
	C	2
	●	
4 QED	B	3
	○	A, B ⊭ D
	D	3
3 Cnj	B ∧ D	2
2 Cnj	C ∧ (B ∧ D)	

A	B	C	D	A	∧	B	/	C	∧	(B	∧	D)	
T	T	F	T		Ⓣ				Ⓕ		T		lurks in first dead-end gap
T	T	F	F		Ⓣ				Ⓕ		F		lurks in both dead-end gaps
T	T	T	F		Ⓣ				Ⓕ		F		lurks in second dead-end gap

While a dead-end gap specifies just one assignment of truth values to the vocabulary actually appearing in its proximate argument, this assignment may be provided by more than one interpretation of the derivation as a whole if the derivations contains further vocabulary. That happens in both gaps here, and it also happens that a single interpretation of the whole derivation is a counterexample lurking in both of the gaps. That's why we end up with 3 interpretations all told.

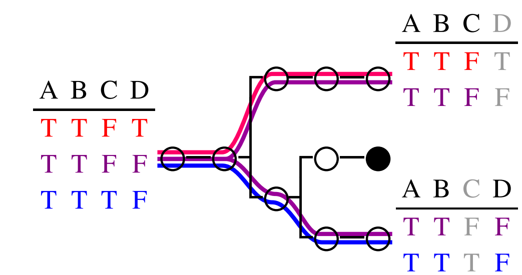


Fig. 2.3.5-1. Counterexamples lurking in the dead-end gaps of the example above.

Since each of these interpretations lurks in all ancestors of the dead-end gap or gaps in which it lurks, any one of the three is enough to provide a counterexample to the ultimate argument. Beginning with chapter 6, it will prove to be most convenient to assign **F** to an unanalyzed component whenever we have a choice, and here that would lead us to the middle interpretation in the case of both gaps. But, for now, when an unanalyzed component does not appear in the proximate argument of a dead-end gap, the choice of the value to assign to it is entirely arbitrary.

2.3.6. Reaching decisions

We know that if a system of derivations has individual rules that are both sound and safe and is, as a whole, sufficient, it will never give us an incorrect answer regarding the validity of an argument. But it is entirely possible that such a system will give us no answer at all. Of course, if we ever run out of rules to apply, we will have an answer. For then either all gaps will have closed or we will have an open gap that has reached a dead end, and both results provide an answer. However, without some guarantee that we will eventually run out of rules, we have no guarantee that we will eventually have an answer. And such a guarantee is not trivial; in fact, once we get to the last two chapters, we will be working in a system some of whose derivations do go on forever.

We will say that a system is *decisive* when we always reach a point where either all gaps are closed or there is a dead-end open gap. It should be clear that our system so far is decisive. The rules Ext and Cnj replace conjunctions among the resources and goals of a gap by simpler sentences and must therefore eventually eliminate all conjunctions. And when the proximate argument of a gap contains no conjunctions, the only rules that might apply are QED, ENV, and EFQ. Each of these closes a gap and there will be only a limited number of gaps to close, so we must eventually run out of things to do.

But we will go on to consider further rules, and some of these will be sufficiently differently from those we have considered so far that, even when a system is decisive, it may not be as easy to see that it is. So let's look at some questions that arise in making this judgment. As we do this, it is worth remembering that, in assessing decisiveness, we are not really interested in whether a system reaches some valuable goal, only in whether we are bound to run out of things to do when we apply its rules.

One way to judge whether that is so is to provide some count of how much there is that might be done, and see whether each rule of the system reduces that count. However, it is not always easy to describe a single quantity that is always reduced, and the reason can be seen even with our current system. The rules QED, ENV, and EFQ reduce the number of open gaps, and that is certainly a relevant quantity. The rules Ext and Cnj, on the other hand, reduce the complexity of proximate arguments, something else that cannot go on for ever. While complexity may seem too abstract to be reduced to a single number, the simple expedient of counting the number of connectives in a proximate argument actually provides a useful quantity in the present setting. So far, so good, but the real problem arises in putting these two numbers together.

This problem is easiest to see by considering Cnj. While the proximate arguments of both its children are simpler than that of their parent, it adds to the total number of open gaps. It is tempting to say that this is acceptable because the increase in the number of open gaps is no greater than the decrease in the complexity, so the sum of the two is not increased. But this would be wrong on two counts. First, it is not enough that we avoid increasing the quantity we are watching: rules that merely kept it the same might go on for ever doing that. Second, our system would still be decisive if Cnj added 10, 100, or even a million new gaps when it eliminated a single connective. For, in the absence of a rule that added connectives, it would eventually run out of connectives to eliminate, and we would be forced to use other rules which did reduce one quantity without increasing the other.

This is not to say that there is no way of putting the number of open gaps and the complexity of proximate arguments together to produce a useful quantity, but any way of doing that must recognize their asymmetry: we can add gaps as we reduce the number of connectives but only provided we add no new connectives when we close gaps. However, we will not look at ways of actually combining these quantities. We will simply employ the abstract idea of a rule moving things along. We will call a rule that does this *progressive*, understanding that whether a rule is progressive depends not only on what quantities it might reduce but also on what other rules are present. The common idea associated with our various uses of this term progressive will be that, if all our rules are progressive, each moves us far enough along that we can never apply them more than a limited (though perhaps very large) number of times before we run out of things to do.

So a system all of whose rules are progressive will be decisive; that is, we will always reach a point at which no more rules can be applied. At that point, any gap that is left open will have reached a dead end, and the derivation will have provided an answer about the validity of the original system. And we saw earlier that if a system is sufficient and conservative, the existence or non-existence of an open gap when no more rules apply provides a correct answer regarding validity of the ultimate argument. A system that always eventually provides an answer and a correct one, can be said to provide a *decision procedure* for validity.

2.3.7. Soundness and completeness

Our current system is sufficient, conservative, and decisive, and it therefore provides a decision procedure. But we can cut up its properties in another way. Because it is decisive as well as accurate in its answers, we can say both of the following about any derivation:

(1) The ultimate argument of a derivation is valid if and only if at some stage all gaps have closed.

(2) The ultimate argument of a derivation is invalid if and only if eventually we reach a dead-end open gap.

The if parts of these together say that the system is accurate, and we have seen that they follow from its conservativeness (along with sufficiency in the case of the second statement). The only if parts follow from the if parts given decisiveness. (For example, if the ultimate argument is valid, it must be the case that all gaps close because otherwise, given decisiveness, we would reach a dead-end gap and the ultimate argument would not be valid.) Moreover, the only if parts of the two claims above together imply decisiveness because an argument will always be either valid or invalid, so they tell us that eventually either all gaps close or we reach a dead-end gap.

But these two claims, like the properties of soundness and safety, are not of equal importance. The first is closely tied to the use of derivations to establish validity while the second is similarly related to their use to find counterexamples and establish invalidity. The first is of special interest also because it can be established in some cases where decisiveness fails, and we will take it as the key property of our system of derivations in chapters 7 and 8 when we must abandon decisiveness.

It is standard to give different names to the two parts of the first statement:

(1a) The ultimate argument of a derivation is valid *if* at some stage all gaps have closed

(1b) The ultimate argument of a derivation is valid *only if* at some stage all gaps have closed

When we can be sure that (1a) is true, we say that the system is *sound*. We have seen that a system will be sound in this sense if all its rules are sound. When we can be sure that (1b) is true, we say the system is *complete* because such a system provides a proof for each valid argument.

We can show that a system is complete if we know (i) that its rules are safe and the system as whole is sufficient and we know also that (ii) any derivation whose ultimate argument is valid eventually reaches an end. Property (ii) is not full decisiveness since it applies only to derivations whose ultimate argument is valid. This sort of partial decisiveness is something we will be able to establish for the systems of chapters 7 and 8, for which full decisiveness does not hold. And, because this partial decisiveness is enough to provide completeness, all systems that we will study in the course are both sound and complete.

Glen Helman 11 Jul 2012

2.3.8. Formal validity

As was noted earlier, the use of the term valid in connection with derivations requires some qualification. In the context of derivations, as in the context of analyses, Roman capital letters are used to stand for particular sentences that are not analyzed further, and such sentences need not be logically independent. That means that a given extensional interpretation of unanalyzed sentences need not be realized in any possible world. So, in the example of 2.3.1, even though the appearance of a dead-end gap leads us to write "B, A, ⊤ ⊭ C", it might be that the particular sentences A and B do together entail the particular sentence C, and it could even be that C is tautology or that A and B are mutually exclusive. In short, knowing that there is an extensional interpretation of analyzed sentences that assigns them certain truth values does not show that it is logically possible for the sentences to have those truth values.

On the other hand, our interest in derivations is as a way of applying general principles of formal logic. And, even though these principles are applied to particular sentences, their application depends only on the features of these sentences that are displayed in symbolic analyses. In particular, the use of derivation rules does not depend on the specific identity of unanalyzed components. This means that when the gaps of a derivation do all close we know not only that its premises entail its conclusion but also that the same is true for any argument having the same form. One way of putting this is to say that we know the argument to be *formally valid* or, more precisely, to be valid in virtue of the form exhibited in the particular analysis we have used. Since formal validity is a stronger property than simple validity, knowing that an argument is formally valid is enough to tell us it is valid; and we will usually drop the qualification formal for this reason. But it is important to remember that when an argument is labeled "invalid" on the basis of a derivation, this judgment is relative to a particular analysis of it. Indeed, if this were not so, we could stop after studying conjunction: the point of considering further logical forms is to recognize the validity of arguments that count as formally invalid when considered solely in terms of conjunction.

The idea of validity in virtue of form can itself be spelled out by saying that an argument is formally valid with respect to a given analysis when any way of associating sentences with its unanalyzed components produces a valid argument. So when the derivation of 2.2.5 showed us that (A ∧ B) ∧ C, D ⊨ C ∧ (A ∧ D), this told us something not only about the specific sentences (A ∧ B) ∧ C, D, and C ∧ (A ∧ D) but about any sentences that are related in the way indicated by these analyses—that is, about the sentences could be formed in these ways from any choice of sentences, A, B, C, and D. Such choice of actual sentences, one for each of a group of unanalyzed components, is an intensional interpretation in the sense discussed in 2.1.8, so we can say that an analyzed argument is formally valid when every intensional interpretation of it is valid.

When a derivation leads to a dead-end gap, what we know, speaking most strictly, is that its ultimate argument is not formally valid. That is because one test of formal validity is whether there is an extensional interpretation of the argument that separates its premises from it conclusion. And we will look more closely at why that is so.

First, if there is an extensional intepretation that provides a counterexample to an argument, we can construct an intensional interpretation by assigning to each component an actual sentence with the truth assigned by the extensional interpretation, and this interpretation will yield an actual argument having the same form as the original one but with actually true premises and an actually true conclusion. In example from 2.3.1, the counterexample given by the dead-end gap assigns **T** to A and B and **F** to C. So we might associate English sentences with these unanalyzed components as follows:

A: Atlanta is in Georgia
B: Boston is in Massachusetts
C: Chicago is in Massachusetts

If so, the proximate argument of the dead-end gap will be

Boston is in Massachusetts
Atlanta is in Georgia
$\top$

Chicago is in Massachusetts

and the ultimate argument of the derivation will be

Atlanta is in Georgia and $\top$; moreover, Boston is in Massachusetts
---
Boston and Chicago are both in Massachusetts

To get something completely in English, we can replace $\top$ by any tautology. If we use Atlanta is Atlanta, we get

Atlanta is in Georgia and is Atlanta; moreover, Boston is in Massachusetts
---
Boston and Chicago are both in Massachusetts

Each of these particular arguments has a false conclusion along with true premises not merely in *some* possible world but in the actual world, so they are certainly invalid. Because the latter two have the same form as the ultimate argument of the derivation, that ultimate argument is not valid *with respect to the form displayed in its analysis*. If in that argument, the unanalyzed A, B, and C happen to be sentences such that A, B $\vDash$ C, the argument will in fact be valid. For example, it might be

All humans are mortal and are human; moreover, Socrates is human
---
Socrates is both human and mortal

But it will remain true that it is not valid with respect to the form displayed in the symbolic analysis, and we have shown it is not by giving another interpretation of this form that is not valid.

We have seen that an argument whose premises are separated from its conclusion by an extensional interpretation is not formally valid. The converse is also true. That is, if an argument is not formally valid, its premises are separated from its conclusion by some extensional interpretation. The claim that an argument is formally valid is a generalization about both intensional interpretations and possible worlds, and a counterexample to this generalization is provided an intensional interpretation and a possible world with the property that the actual argument that results from the intensional interpretation has its premises separated from its conclusion by the possible world. But any intensional interpretation and possible world will determine an assignment of truth values to the unanalyzed components of the argument. In the example above the value **T** is assigned to the unanalyzed component A by associating the sentence Atlanta is in Georgia with A and considering the truth value of this sentence in the actual world. Since any intensional interpretation and possible world will determine an extensional interpretation in this way, any counterexample to the formal validity of a symbolic argument will provide an extensional interpretation that separates its premises from its conclusion.

This means that even if we do not define formally validity directly in terms of inseparability of premises from conclusion by extensional interpretations but instead in terms of validity under any intensional interpretation, it will still be true that an argument is formally valid if and only if no extensional interpretation separates its premises from its conclusion.

2.3.s. Summary

1 When a derivation is constructed for an invalid argument, eventually we will find that an open gap has reached a dead end without closing. We mark such a gap with a empty circle ○ and write its active resources and goal with the sign ⊭ between to indicate that they do not form a valid argument. And we will see that the invalidity of the proximate argument of a dead-end gap implies the invalidity of the ultimate argument for which the derivation is constructed.

2 We will often be concerned with formal validity, so we extend to assignments of truth values the idea of constituting a counterexample to an argument. The fact that any dead-end open gap has its active resources separated from its conclusion by some interpretation indicates that our system is sufficient in the sense of having enough rules to close all dead-end gaps whose proximate arguments are valid.

3 When speaking of the tree structure of the gaps of a proof, it is convenient to use a genealogical metaphor and to speak of a gap at one stage as the parent of the gaps that derive from it at the next stage, gaps that are its children. Children of a gap's children, their children, and so on are descendants of the gap, and it is an ancestor of them. We can state a necessary and sufficient condition for a counterexample to its proximate argument lurk in a gap in terms of the existence of lurking counterexamples at later stages.

4 We can be sure that a counterexample to the proximate argument of a dead-end gap is a counterexample to the derivation's ultimate argument provided all our rules are safe in the sense of never introducing new lurking counterexamples. When the converse is true, when we our rules never allow us to ignore lurking counterexamples, they are strict. Since our real interest is in the ultimate argument of a derivation, it is really enough to attend to lurking counterexamples when they also lurk in all ancestors of a gap. Rules that insure that we do this are sound; when all rules are safe, sound rules are also strict. The idea of soundness enables us to justify the use of available but inactive resources (to, for example, close gaps) even when not all rules are safe. A system whose rules are all sound and also safe is conservative.

5 When an interpretation is a counterexample to the proximate argument of a dead-end open gap, this interpretation is also a counterexample to the ultimate argument of the derivation, and we will confirm such a counterexample as a way of finishing off a derivation that fails.

6 A system will be decisive (in the sense that any derivation will always come to an end) provided its rules are all progressive (in the sense of always leading us closer to a point where no more can be done). Many rules are progressive because they either close a gap or replace a goal or active resource by one or more simpler sentences. A decisive system which is sufficient and conservative (and is therefore correct in the answers it gives) provides a decision procedure for formal validity.

7 Not all systems we consider will provide decision procedures but all will be sound in the sense of providing proofs only for valid arguments and complete in the sense of leading us to a proof whenever an argument is formally valid.

8 An argument that is valid may have a form that is invalid in the sense that some intensional interpretation of the unanalyzed components appearing in the form—i.e., some way of associating actual sentences with them—yields an invalid argument. Formal validity implies validity, so a derivation that succeeds shows both, but one that fails only shows formal invalidity.

2.3.x. Exercise questions

Use the basic system of derivations to check each of the claims below. If a derivation indicates that a claim fails, confirm a counterexample. That is, give an interpretation that separates the active resources of an open gap from its goal and calculate truth values for the premises and conclusion from it—as is done in the example in 2.3.5 (though you need only provide a single counterexample even when the derivation leads you to several):

1. A ⊨ A ∧ B

2. A ∧ B ⊨ A ∧ (B ∧ A)

3. B ∧ E, C ∧ ⊤ ⊨ (A ∧ B) ∧ (C ∧ D)

4. A ∧ B, B ∧ C, C ∧ D ⊨ A ∧ D

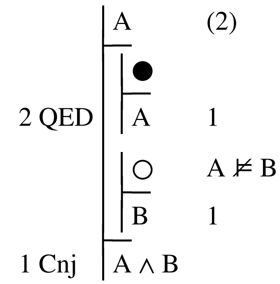
5. A, B ∧ A, D ⊨ B ∧ ((C ∧ A) ∧ D)

For more exercises, use the exercise machine.

Glen Helman 11 Jul 2012

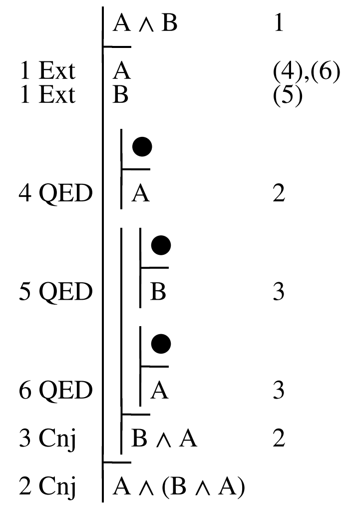
2.3.xa. Exercise answers

1.



A	B	A	/	A	∧	B
T	F	Ⓣ			Ⓕ	

2.



3.

	B ∧ E	1
	C ∧ ⊤	2
1 Ext	B	(5)
1 Ext	E	
2 Ext	C	(7)
2 Ext	⊤	
	○	B, C, E, ⊤ ⊭ A
	A	4
	●	
5 QED	B	4
4 Cnj	A ∧ B	3
	●	
7 QED	C	6
	○	B, C, E, ⊤ ⊭ D
	D	6
6 Cnj	C ∧ D	3
3 Cnj	(A ∧ B) ∧ (C ∧ D)	

A	B	C	D	E	B	∧	E	,	C	∧	⊤	/	(A	∧	B)	∧	(C	∧	D)
F	T	T	F	T		Ⓣ				Ⓣ	T			F		Ⓕ		F	

This derivation could have been ended after stage 4 when the first open gap has reached a dead end. Often answers will show a derivation continued further than necessary in order to show how the further steps would have worked out. The counterexample confirmed here lurks in both dead-end gaps; there are others that lurk in only one of the two. Notice that ⊤ is not assigned a value at the left of the table. Since its value is fixed by the stipulation that it is a tautology, a value need not and cannot be assigned to it as part of an extensional interpretation.

4.

	A ∧ B	1
	B ∧ C	2
	B ∧ D	3
1 Ext	A	(5)
1 Ext	B	
2 Ext	B	
2 Ext	C	
3 Ext	B	
3 Ext	D	(6)
	●	
5 QED	A	4
	●	
6 QED	D	4
4 Cnj	A ∧ D	

Clearly, there is redundancy in the active resources of the gaps after stage 3 of this derivation. Since both gaps close, the exploitation of the second premise at stage 2 is not necessary (though it would be necessary before any gap could reach a dead end). It would be possible to state rules so that the resource B was not repeated at stages 2 and 3, but such repetition does not ordinarily enlarge derivations significantly and makes it easier to check whether rules have been applied fully and correctly.

5.

	A	(6)
	B ∧ A	1
	D	(7)
1 Ext	B	(5)
1 Ext	A	
	●	
5 QED	B	2
	○	A,B,D ⊭ C
	C	4
	●	
6 QED	A	4
4 Cnj	C ∧ A	3
	●	
7 QED	D	3
3 Cnj	(C ∧ A) ∧ D	2
2 Cnj	B ∧ ((C ∧ A) ∧ D)	

A	B	C	D	A	,	B	∧	A	,	D	/	B	∧	((C	∧	A)	∧	D)
T	T	F	T	Ⓣ			Ⓣ			Ⓣ			Ⓕ		F		F	

Glen Helman 15 Jul 2012