7.8. Finite & infinite structures

7.8.0. Overview

Many arguments that generate unending derivations have counterexamples that use only finitely many reference values. But, although the rules can be modified to uncover such counterexamples, this is not enough to insure decisiveness.

7.8.1. Finding finite structures

We can search for finite counterexamples by modifying rules to consider old terms along with new ones or to consider the possibility that new terms are co-aliases of old ones.

7.8.2. The failure of decisiveness

We cannot hope to find counterexamples in this way for all invalid arguments because the counterexamples to some invalid arguments always have infinite ranges.

Glen Helman 11 Jul 2012

7.8.1. Finding finite structures

To complete the discussion of the adequacy of the system of derivations for generalizations, we will look a little more closely at the reasons why it is not decisive. There are two aspects to the problem, one concerning universals alone and another concerning their interaction with functors. The infinitely developing derivations displayed earlier are enough to show us that our system is not decisive, but the failure of decisiveness in these derivations does not run very deep and can be overcome by a relatively small adjustment to our rules. Different adjustments are needed to handle universals and functors, and we will consider the case of universals first.

The rule UG directs us to reach a universal goal ∀x θx by trying to close a gap whose goal is an instance θa for some independent term a. Although we need to close such a gap to show that the universal goal can be reached, this gap need not point us toward the only the only sort of counterexample lurking in the original gap. When we are constructing general arguments we are checking for ways of making generalizations false, and that means ways of making instances of the generalizations false. Thus, for a general argument to go through, we must show that there is no false instance of any sort; it is not enough to show that the instances for the things we are already speaking of are not false.

However, to show that an argument for a generalization must fail, a false instance of the generalization for a term of any sort, new or old, will do; and a structure separating resources from an instance of the universal for an old term would be enough to show that the universal is not entailed by those resources. This means that, in a negative use of derivations, there is some reason for considering gaps whose goals are instances for old terms. We can refine our analysis of entailment to take account of this by making the planning rule for universals more elaborate. The alteration makes derivations cumbersome in practice; but, even if we do not put it put into actual use, it can help to focus attention on deeper reasons for failure of decisiveness.

The revised rule is *Supplemented Universal Generalization* (UG+); it is shown in Figure 7.8.1-1. The rule UG+ alters UG by adding further new gaps in which we try to conclude instances of the universal not only for a new term but also for terms already appearing in the gap. Adding these new gaps will certainly make it no easier to show that an entailment holds. And they make it no harder either: anything that can be shown for the independent term a can be shown for any term, so if that gap closes, all the others will, too. The function of the added gaps is instead to help us show that an entailment fails while using as few terms as possible. The new gaps provide new directions in which we may search for a path that not only remains open but reaches a dead end.

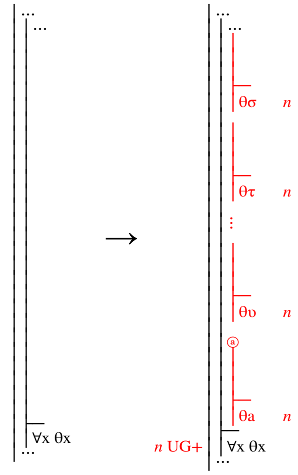
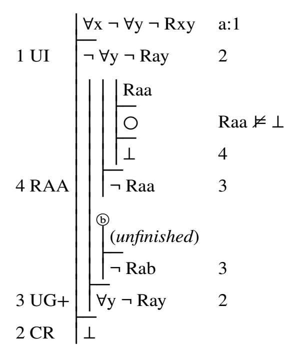
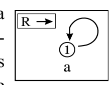


Fig. 7.8.1-1. Developing a derivation at stage *n* by planning for an unrestricted universal; the independent term a is new to the derivation and the terms σ, τ, …, υ include at least one from each current alias set for the gap.

The derivation at the right shows the effect of the first rule when it is applied to one of the examples of 7.7.3. The first gap in this derivation has reached a dead end. Its only active resources are the initial premise and Raa, and neither is exploitable. In the case of the universal premise, this is because it has already been exploited for the term a, which is the only term ever appearing in this gap.



The gap describes a structure whose referential range contains one value, and the



predicate R will be true of the pair consisting of this value and itself. The initial premise—which says that there is no value that is related to nothing by R—is thus true in this structure, showing that the *reductio* entailment ∀x ¬ ∀y ¬ Rxy ⊨ ⊥ fails.

A planning rule for universal goals is one way we can be led to introduce and unending series of terms. Another way we have seen occurs when a universal quantifier binds a variable occurring in a compound term. When such a generalization is instantiated, a new compound terms can be introduced into the derivation, leading to still further instantiation. We could avoid such further instantiation if the new compound term was in the same alias set as a term for which the universal had already been instantiated, so we can investigate the possibility of avoiding an infinitely developing gap by trying to put new compound terms in already existing alias sets.

On this approach, when we introduce a new compound term that does not automatically become part of an already existing alias set, we also look at ways of identifying the new compound term with existing terms, at least one from each alias set. We will say that in doing this we are *securing* the term. Of course, it may be that no identification with existing terms is consistent with our resources. We allow for this possibility by adding a gap in which we make no assumptions about the new term.

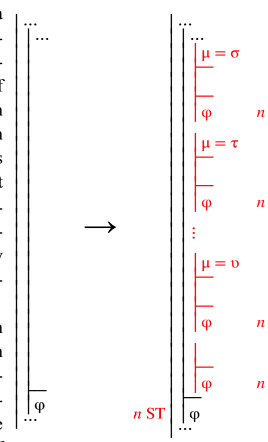
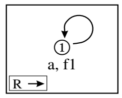


Fig. 7.8.1-2. Developing a derivation at stage *n* by securing a compound term μ; the terms σ, τ, …, υ include at least one from each current alias set for the gap other than the one including μ.

The rule shown in Figure 7.8.1-2 can be used to secure terms. We suppose in turn along new scope lines that a compound term is a co-alias of each of a series of unanalyzed terms already in the gap and also pursue the development of the gap along a new scope line without added assumptions. Fullest investigation of the possibilities comes if we include at least one term from each alias set. We will call this rule *Securing a Term* (ST). Nothing in the statement of this rule requires that the term μ be new, but that is the only use that we are interested in now.

Although the application of ST would often be quite awkward, it makes short work of the first of the examples from 7.7.3 .

```
     │ ∀x Rx(fx)         a:1
1 UI │ Ra(fa)
     │ │ fa = a          a—fa
     │ │ ○               Ra(fa), fa = a ⊭ ⊥
     │ │ ⊥               2
     │
     │ │ (unfinished)
     │ │ ⊥               2
2 ST │ ⊥
```



Having introduced the term fa through the instantiation at stage 1, we have the alias sets {a} and {fa}. We consider securing fa by identifying it with the term a. The first gap has then reached a dead end because the universal has already been exploited for a member of its single alias set. There is a second unfinished gap that merely represents the continuation of the gap after stage 1 with no added assumption about the identity of the term fa. The structure described by the dead-end gap is one whose range has a single member named by the term a, which stands in the relation R to itself. The single reference value of this structure is the only possible input and output for the functor f.

Glen Helman 11 Jul 2012

7.8.2. The failure of decisiveness

The rules UG+ and ST are designed to uncover finite structures whenever possible. We will not prove that they do this; instead we will see why finite structures are not always there to be uncovered.

For example, consider the following pair of sentences:

$$\forall x\ \forall y\ \forall z\ ((Rxy \land Ryz) \rightarrow Rxz)$$
$$\forall x\ \neg\ Rxx$$

The first says that the relation expressed by R obeys a law of transitivity, and the second says that nothing is related to itself by R, which is to say that R is *irreflexive*.

What must a structure be like to make these sentences true? Thinking in terms of the diagrams of 6.4.2 , the claim of irreflexivity tells us that there cannot be any looped arrows. The claim of transitivity tells us that arrows linked head to tail running from object a to object b and from object b to object c can be spanned by an arrow running directly from a to c (see Figure 7.8.2-1).

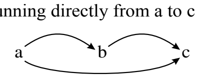


Fig. 7.8.2-1. The arrow spanning two linked arrows that is implied by transitivity.

Now, if we had a circuit of arrows leading from some object back to itself by way of other objects, transitivity would imply that there was a loop leading from the object directly back to itself. Figure 7.8.2-2 illustrates this in a case where we have Rab, Rbc, Rcd, and Rda. Transitivity made be applied three times, first to show that Rac (because Rab and Rbc), then to show that Rad (because Rac and Rcd), and finally to show that Raa (because Rad and Rda).

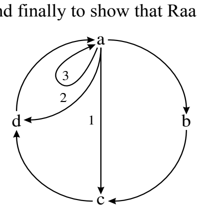


Fig. 7.8.2-2. A circuit from a to a reduced to a looped arrow in three steps by spanning linked arrows.

Irreflexivity would rule out the truth of Raa, so irreflexivity can hold along with transitivity only if there are no loops or circuits of arrows like the one illustrated.

Finally, let us add to the statements of transitivity and irreflexivity either of the sentences ∀x ¬ ∀y ¬ Rxy and ∀x Rx(fx) that we considered in 7.8.1 . Each of the latter sentences tells us, in its own way, that every object is at the tail of some arrow. A little thought (and attempts at diagrams) will show that there is no way to manage this with a finite number of objects unless there is somewhere a loop or a circuit of arrows. So, although the sentences ∀x ¬ ∀y ¬ Rxy and ∀x Rx(fx) can each be made true in a structure with only one reference value if we consider them by themselves, they cannot be true along with claims of transitivity and irreflexivity in any structure with only a finite number of values.

Nevertheless ∀x ¬ ∀y ¬ Rxy and ∀x Rx(fx) are consistent with claims of transitivity and irreflexivity. For example, let us take the positive integers as our referential range and let R express the relation $<$ of one number being less than another. The relation $<$ is transitive and irreflexive. Moreover, each positive integer is less than some positive integer, so there is no positive integer that has the property of being less than no positive integer—and that is what ∀x ¬ ∀y ¬ Rxy says on this interpretation. And, if we interpret the functor f by any function whose output is always larger than its input, ∀x Rx(fx) will also be true.

So there are sets of sentences that are consistent but whose members cannot all be true with only a finite range of reference values. This means that, even if a revised system of derivations using UG+ and ST always succeeds in locating finite structures, it cannot always provide an answer to our questions about entailment. If the entailment holds, it will say so. If the entailment fails and can be shown to fail using a finite structure, it will say so. But, if the entailment fails and can only be shown to fail only by using an infinite structure, it will give no answer because it will never finish describing a structure of the required sort.

Of course, it is possible to describe an infinite structure in a finite space (as we did informally above), so we might hope that a more substantial modification of our system might lead us to descriptions of infinite structures after finitely many stages. But here we must recall the result of Church mentioned in 7.7.1 : although an improved system might provide answers to some further questions about entailment, no system could answer them all correctly. In terms of the present discussion, this implies that no matter what method we choose for describing structures, there are bound to be structures among those we need to describe that our method would not lead us to.

Glen Helman 11 Jul 2012

7.8.s. Summary

1 Our system is not decisive in part because we always look to new independent terms as possible counterexamples to a generalization and assume that terms are not co-aliases unless our resources tell us otherwise. But, while we must consider new terms as possible counterexamples and we must allow for the possibility that terms that have not been made co-aliases refer to different things, we may also consider alternatives that point toward smaller structures. The rule Supplemented Universal Generalization (UG+) leads us to consider instances for old as well as new terms when planning for a generalization. And we can secure new compound terms as co-aliases of terms already present by using the rule Securing a Term (ST) .

2 Even with these rules, we cannot always reach dead-end gaps when derivations fail because dead-end gaps describe finite structures, and invalid arguments do not always have finite counterexamples. That is, there are some sets of sentences whose members can be made all true only with an infinite range of reference values. One example consists of sentences saying that a predicate R expresses a relation that is irreflexive and transitive and is such that each reference value stands in this relation to some reference value. No system like ours could drive a gap to a dead end in such cases and, while a very different system might do better in some of them, it has been shown that no system could do so in all such cases.

Glen Helman 11 Jul 2012

7.8.x. Exercise questions

Use the system of derivations to find structures separating premises from conclusions in the cases below. You will need to use the rule UG+.

1. ∀x ¬ ∀y ¬ Rxy / ∀x ¬ Rxx
2. ∀x ¬ ∀y Rxy / ¬ ∀x Rxa
3. ∀x ¬ ∀y Rxy / ∀x ¬ Rax

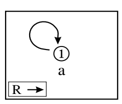
The exercise machine doesn't incorporate the rule UG+, so derivations for arguments where it is needed will never end.

Glen Helman 11 Jul 2012

7.8.xa. Exercise answers

1.

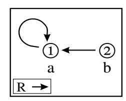
	∀x ¬ ∀y ¬ Rxy	a:3
	ⓐ	
	Raa	
3 UI	¬ ∀y ¬ Ray	4
	Raa	
	○	Raa ⊭ ⊥
	⊥	6
6 RAA	¬ Raa	5
	ⓑ	
	(*unfinished*)	
	¬ Rab	5
5 UG+	∀y ¬ Ray	4
4 CR	⊥	2
2 RAA	¬ Raa	1
1 UG	∀x ¬ Rxx	



From the 1st open gap (the 2nd is not fully developed).

2.

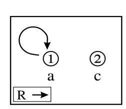
	∀x ¬ ∀y Rxy	a:2, b:8
	∀x Rxa	a:3, b:9
2 UI	¬ ∀y Ray	4
3 UI	Raa	(6)
	●	
6 QED	Raa	5
	ⓑ	
	¬ Rab	
8 UI	¬ ∀y Rby	10
9 UI	Rba	(12)
	●	
12 QED	Rba	11
	¬ Rbb	
	○	Raa,¬ Rab,Rba,¬ Rbb ⊭ ⊥
	⊥	13
13 IP	Rbb	11
	ⓒ	
	(*unfinished*)	
	Rbc	11
11 UG+	∀y Rby	10
10 CR	⊥	7
7 IP	Rab	5
5 UG+	∀y Ray	4
4 CR	⊥	1
1 RAA	¬ ∀x Rxa	



From the 1st open gap (the 2nd is not fully developed)

3.

	∀x ¬ ∀y Rxy	a:3, c:8
	Raa	(6)
3 UI	¬ ∀y Ray	4
	●	
6 QED	Raa	5
	ⓒ	
	¬ Rac	
8 UI	¬ ∀y Rcy	9
	¬ Rca	
	○	Raa,¬ Rac,¬ Rca ⊭ ⊥
	⊥	11
11 IP	Rca	10
	¬ Rcc	
	○	Raa,¬ Rac,¬ Rcc ⊭ ⊥
	⊥	12
12 IP	Rcc	10
	ⓓ	
	(*unfinished*)	
	Rcd	10
10 UG+	∀y Rcy	9
9 CR	⊥	7
7 IP	Rac	5
5 UG+	∀y Ray	4
4 CR	⊥	2
2 RAA	¬ Raa	1
	ⓑ	
	(*unfinished*)	
	¬ Rab	1
1 UG+	∀x ¬ Rax	



From the 1st and 2nd open gaps (the 3rd and 4th are not fully developed)

Glen Helman 19 Jul 2012