

Problem Set 5

Solutions: Please Never Share or Upload to Internets or I'll have to make many new problems in the future and that would be a BIG BUMMER

1. Consider the argument with premise set $\{\Phi, \Psi\}$ and conclusion Θ . Suppose that you are able to make a tree for this argument in which all branches close, even though you make no use of Φ and Ψ : i.e. you do not resolve these sentences, and when you close a branch, it is never because it contains Φ or because it contains Ψ . What is the most informative thing you can say about Θ ?

Solution:

Θ must be a tautology. Whatever we did in the given tree with Φ , Ψ , and $\sim\Theta$ at its root to make all the branches close, we could have done in a tree with just $\sim\Theta$ at its root—since by assumption Φ and Ψ played no role in the further work we did. So there is a tree with just $\sim\Theta$ at its root in which all branches close (important to note this!). Hence, $\emptyset \vdash_{STD} \Theta$. By soundness, $\emptyset \models \Theta$, so Θ is a semantic tautology.

2. Prove that in a sound tree system, no argument G is both tree-valid and tree-invalid. Let ‘ G ’ be an arbitrary argument with finite premise set Γ and conclusion Θ .

(*Hints*: Show that if an argument is tree-invalid, then it is not tree-valid. To do this, apply a key fact coming from our proof of completeness, and then apply the soundness result. Alternatively, you could do a proof by contradiction, using the same results.)

Solution:

Suppose that argument G is tree-invalid. Then by definition, there exists at least one tree with root $\Gamma \cup \{\sim\Theta\}$ that has a complete open branch. Call this complete open branch ‘Oprah.’ From our proof of completeness, we can use Oprah to define a truth value assignment that satisfies the root. This would be a TVA that makes the premises true but the conclusion false. Hence, the argument G must be semantically invalid, i.e. $\Gamma \not\models \Theta$. Since the system is Sound, we can apply the contrapositive of soundness to conclude that *it is not the case that* G is tree-valid, i.e. $\Gamma \not\vdash_{STD} \Theta$.

Hence, if an argument is tree-invalid, then it is not tree-valid (assuming our system is sound and complete). Hence, no argument is both tree-invalid and tree-valid.

Alternative solution: proof by contradiction:

Assume for contradiction that there is some argument G that is both tree-valid and tree-invalid. By soundness, G is semantically valid. Hence, whenever the premises Γ are true, so is the conclusion Θ . Hence, $\Gamma \cup \{\sim\Theta\}$ is inconsistent.

Yet, if G is tree-invalid, then there exists a tree with root $\Gamma \cup \{\sim\Theta\}$ that has a complete open branch. From our proof of completeness, we saw how to use such a branch to construct a TVA where the root is satisfied. So this TVA would satisfy $\Gamma \cup \{\sim\Theta\}$. But that contradicts that $\Gamma \cup \{\sim\Theta\}$ is unsatisfiable! Hence, G cannot also be tree-invalid.

Pick **one** of questions 3 or 4 to answer:

3. Briefly explain why we could augment our nine tree rules with their ‘syntactic equivalents’ and not get into trouble with our Soundness and Completeness results (e.g. swapping the order of branches in the ‘splitting rules’, or the order of sentences in the ‘stacking rules’). Your brief argument should note both (i) why our system would remain Sound (‘reasoning from the top-down’) and (ii) why our system would remain Complete (‘reasoning from the bottom-up’).

Solution:

A very short ‘brief explanation’: notice that the 20 syntactic variant rules involve either (i) the same child-nodes (possibly permuted) or (ii) permutations of the sentences within a child-node (or both). Yet, neither our soundness nor completeness proof depended on the order of the nodes or the order of the sentences appearing within a child-node.

Providing some more details (since the last claim above might be taken to be exactly what’s in question): In the proof of soundness, we showed that if the sentence being resolved is satisfied, then so is at least one of the child-nodes (reasoning from the ‘top-down’). Since a syntactic variant involves the same child-nodes—just permuted or with sentences within a node permuted (or both)—at least one of the child nodes must remain satisfied whenever the sentence being resolved is satisfied.

In the proof of completeness, we showed that satisfying any one of the child nodes entails that the sentence being resolved is satisfied (reasoning from the ‘bottom-up’). This will also remain true, since our syntactic variants never change the sentences on a child node, just their order and/or the order of the child nodes.

A cheap argument for preservation of completeness: As some of you noted, adding additional rules to a complete system can never break completeness. Hence, to really make the completeness question interesting, I would have had to add some convoluted language about considering variant systems that swap out a syntactic variant for each of the standard nine rules. Thanks to those who respected the spirit of the question!

4. Our textbook's Chapter 5 does not set-up the inductive proof for completeness properly. In lecture, we showed how to properly set-up the proof for soundness. Do the same for completeness noting (i) what you are doing induction over; (ii) the base case(s); (iii) the induction hypothesis; (iv) what you would need to show in the induction step.
NB: I'm not asking you to actually re-do the proof, just to set it up!

Solution:

Background: We assume that the argument from Γ to Θ is not tree-valid, i.e. that $\Gamma \not\vdash_{STD} \Theta$. This means that EVERY tree with root $\Gamma \cup \{\sim\Theta\}$ does not close. So considering an arbitrary such tree, if we complete it, we are guaranteed to have a complete open branch. Call this open branch O . We aim to use this completed open branch O to define a truth value assignment \mathcal{I} that makes true every well-formed formula on O . We will then have shown that the root $\Gamma \cup \{\sim\Theta\}$ is satisfiable, which means that $\Gamma \not\models \Theta$, which is the consequent of our contrapositive of completeness.

(i) We choose to do induction over the number of nodes, treating the last node of our completed open branch as having index $n = 1$. The next higher node has index $n = 2$, and so on to some finite number m which indexes the root (m could be an arbitrarily large finite number in \mathbb{N}).

(ii) **Base case:** we consider the last node, with index $n = 1$. Since the branch is complete, the sentences appearing here must be either atomic sentences or a single negation of an atomic sentence. We partially define an interpretation \mathcal{I} such that it assigns True to any atomic sentence at this last node and False to an atomic sentence that is negated (so that the negation of this sentence is true).

Variant base case: we could avoid a case below by fully defining \mathcal{I} such that it assigns an atomic sentence False if and only if the negation of that sentence appears at some node on the branch (so it assigns an atomic sentence True if and only if that sentence appears at some node on the branch). Note that since the branch is complete and open, we are guaranteed to never have both a sentence and its negation appear on the branch, so \mathcal{I} is well-defined. This is perhaps more elegant, but it also takes us out of the base case node; it's really a mix of using the inductive structure of SL right off the bat.

(iii) **Induction hypothesis:** Assume that the property holds at each node with index n , lower than a given node-index k , i.e. such that $1 \leq n < k$. (remember: we are counting up the tree, letting the bottom node have index 1, so that the root is assigned the largest node index). Then in particular, the property holds for each 'complex sentence' below node k .

(iv) Consider an arbitrary node $k > 1$ on our completed open branch O . Then this node either has one or two sentences S and S' . We'll consider sentence S , since the same considerations will show that S' has the property as well. If S is an atomic sentence or the negation of an atomic sentence not considered in the base case, then we extend our definition of \mathcal{I} such it assigns S true. Note that we are guaranteed to be able to

extend \mathcal{I} in this way, since the branch is assumed complete and open, so we will never have already assigned an atomic S as false if S appears at k (since that would mean that $\sim S$ occurs at a lower node, which would mean the branch closes; contradiction). [note that if we used the variant of the base case above, S would already be handled in the case where it is atomic or the negation of an atomic sentence].

Otherwise, S must have string length > 2 . Since it is by assumption a well-formed formula of sentential logic, we know that there must exist well-formed formulae Φ and Ψ such that one of the following five cases obtains: S either equals (i) $\sim\Phi$ (ii) $\Phi \& \Psi$, (iii) $\Phi \vee \Psi$, (iv) $\Phi \supset \Psi$, or (v) $\Phi \equiv \Psi$.

Let's first consider cases (ii)–(v): since O is a complete open branch, we know that sentence S has been resolved. Hence, if S is of the form (ii)–(v), it has been resolved either by the tree rule for conjunction, disjunction, conditional, or biconditional. Each of these rules results in at least one new node, with what we'll call 'tree-children' of S . Since at least one of these tree-children lies on O at a node below k , it has the property by the induction hypothesis. We would then show that S must also have the property given that at least one of its tree-children has the property.

Second, we consider case (i), where S has the form $\sim\Phi$, where $\sim\Phi$ is not an atomic sentence. In this case (again by implicit induction on sentential logic), Φ is itself either of the form (a) $\sim\Omega$ or else of the form (b) $\Omega \& \Delta$, (c) $\Omega \vee \Delta$, (d) $\Omega \supset \Delta$, or (e) $\Omega \equiv \Delta$.

In case (a), S is of the form $\sim\sim\Omega$ for some double negated sentence Ω . Since S has been resolved, Ω appears on a lower node and hence has the property by the induction hypothesis. Since the interpretation \mathcal{I} assigns true to Ω , it must assign true to $\sim\sim\Omega$, so S has the property.

In cases (b)–(e), S is a negation of a sentence whose main connective is one of our four binary connectives. Hence, S has been resolved either by the tree rule for negated conjunction, negated disjunction, negated conditional, or negated biconditional. Hence, at least one of the tree-children coming from these rules lies on O at a node below k . By the induction hypothesis, this tree-child(s) has the property, i.e. it is assigned true by interpretation \mathcal{I} . We would then show that S must also have the property.

We will then have shown that no matter what, the sentence S at node k has the property. Of course, node k could have an additional sentence S' as well, and our argument applies to S' mutatis mutandis. Hence, we will have shown that the arbitrary sentences at our arbitrary node k have the property, namely they are assigned to true by our interpretation \mathcal{I} .

It would follow by induction on the nodes in our completed open branch that the interpretation \mathcal{I} makes true every well-formed formulae on the open branch O , including those at the root. BOOM!

An Alternative ‘Inspired’ Solution using Induction on SL:

Alternatively, we could make a clever choice of the defining property, and proceed to do induction directly on the recursive structure of SL sentences, thereby avoiding the tedium of induction on tree nodes. This clever choice introduces some ‘degenerate cases’, but they are easy to handle.

Background: As before, we assume that every tree with root $\Gamma \cup \{\sim\Theta\}$ does not close. Hence each such tree has a complete open branch O . Using a particular O , we define an interpretation \mathcal{I} as follows: \mathcal{I} assigns an atomic sentence ‘False’ if and only if its negation appears by itself on a node of O (so all atomic sentences appearing by themselves on O are assigned true.) As before, we aim to show that \mathcal{I} makes true every sentence on O (including, of course, the root, showing it to be satisfiable!).

Say that a sentence of SL is ‘inspired’ if it meets one of three conditions: (i) it appears at a node on our complete open branch O and is true according to interpretation \mathcal{I} , (ii) its negation appears at a node in O and the wff is false according to \mathcal{I} (so its negation is \mathcal{I} -true), or (iii) neither the sentence nor its negation appear on O .

(i) Doing complete induction on the string length of sentences in SL, we will show that every SL sentence is inspired.

(ii) **Base case:** consider an arbitrary atomic sentence B . Then there are three cases to consider, and in each, we show that B is inspired: (i) B appears in O , (ii) $\sim B$ appears in O , (iii) neither B nor $\sim B$ appears in O .

(iii) **Induction Hypothesis:** Assume that the property holds for all sentences of string length n where $1 \leq n < k$, i.e. that all such sentences of SL are inspired.

(iv) Consider an arbitrary sentence Δ of string length $k > 1$. Show that this sentence is inspired. By the recursive structure of SL, we know that there must exist component sentences Φ and Ψ (of length $< k$) such that Δ equals either (a) $\sim\Phi$, (b) $\Phi \& \Psi$, (c) $\Phi \vee \Psi$, (d) $\Phi \supset \Psi$, or (e) $\Phi \equiv \Psi$. By the induction hypothesis, Φ and Ψ are both inspired.

In each of these five cases, there are three sub-cases to consider: (a) neither Δ nor $\sim\Delta$ appears on O , (b) Δ appears on O , or (c) $\sim\Delta$ appears on O . The five (a)-subcases are degenerate (Δ is trivially inspired). One (b)-subcase involves $\sim\Phi$ appearing on O , in which case since Φ is assumed to be inspired, Φ must be false according to \mathcal{I} so that $\sim\Phi$ is \mathcal{I} -true (and hence inspired).

In the remaining nine cases, Δ occurs on O and is resolved according to one of our nine tree rules. We then show that no matter which child node appears on O , Δ must be \mathcal{I} -true given that Φ and Ψ are inspired... *Inspired BOOM!*

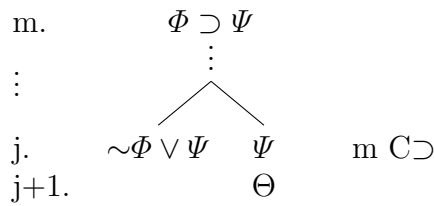
Pick **one** of questions 5 or 6 to answer:

What follows are two modifications to our SL tree system. For each, imagine a system STD^* exactly like our system STD , except for the single indicated change.

(a) **Would the modified tree system be sound?** If so, explain how to extend our inductive soundness proof to a system with this rule; if not, give a tree that is a counterexample to the soundness of STD^* .

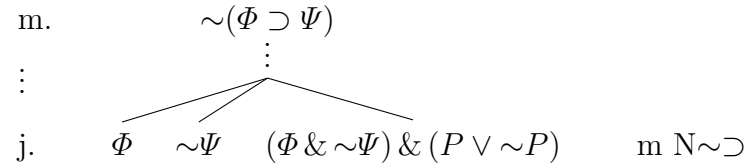
(b) **Would the modified tree system be complete?** If so, explain how to extend our inductive completeness proof to a system with this rule; if not, give a tree that is a counterexample to the completeness of STD^* .

5. *Crunk Conditional* ($\text{C}\supset$)



Note that Θ is an arbitrary wff of SL

6. *Negligent Negated Conditional* ($\text{N}\sim\supset$)



Solution to Number 5:

To test whether a rule preserves soundness, we work ‘from the top down’: does the sentence we are resolving entail the disjunction of the sentences on the node? (I.e. does the sentence ‘up top’ entail the disjunction of the lower sentences?)

Notice that the conditional is logically equivalent to $\sim\Phi \vee \Psi$. So this entails $(\sim\Phi \vee \Psi)$. Since it entails the lower sentence on the left branch, it of course entails the disjunction of the sentences in the two new nodes, i.e. $(\sim\Phi \vee \Psi) \vee (\Psi \& \Theta)$, *regardless* of what sentence Θ is. So far, this is ‘heuristic reasoning’ to be supplemented with a formal extension of our soundness result:

Formally, we consider an interpretation \mathcal{I} that makes $(\Phi \supset \Psi)$ true. Then \mathcal{I} makes either Φ false or Ψ true. Hence, \mathcal{I} makes the left branch sentence $(\sim\Phi \vee \Psi)$ true, so we are guaranteed to have an open branch upon resolving.

To test whether a rule preserves completeness, we work ‘from the bottom up’: does each branch below individually entail the sentence we are resolving? Since the sentence on the left branch is logically equivalent to $(\Phi \supset \Psi)$, we have entailment from the bottom up for this branch: $(\sim\Phi \vee \Psi) \models (\Phi \supset \Psi)$. Turning to the right branch, we notice that the consequent of a conditional entails that conditional: whenever the consequent is true, so is the conditional: $\Psi \models (\Phi \supset \Psi)$. Conjoining this consequent with an arbitrary sentence Θ does not change this, since if the conjunction $\Psi \& \Theta$ is true, both conjuncts are true, and we’ve seen that $\Psi \models (\Phi \supset \Psi)$. If the conjunction is false, then that truth-value assignment can’t be a counterexample to entailment.

Formally, we first consider an interpretation \mathcal{I} that makes $(\sim\Phi \vee \Psi)$ true. Then \mathcal{I} either makes Φ false or Ψ true. Either way, \mathcal{I} assigns true to the sentence we are resolving. Second, we consider an interpretation \mathcal{I}' that makes $\Psi \& \Theta$ true. Then \mathcal{I}' makes true each conjunct. Since it makes Ψ true, \mathcal{I}' must also assign true to the sentence we are resolving (since that sentence is false only when Φ is true but Ψ is false). Hence, no matter which child node lies on our complete open branch, if it is satisfied by our interpretation, then so is the sentence being resolved.

Solution to Number 6:

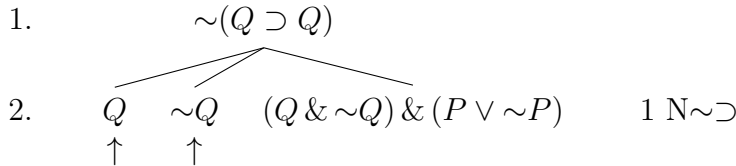
Note that the sentence we are resolving is logically equivalent to $\Phi \& \sim\Psi$. This logically entails $(\Phi \vee \sim\Psi) \vee ((\Phi \& \sim\Psi) \& (P \vee \sim P))$, which is the semantic content of the new node. Hence, the system is sound.

Reframing this in terms of interpretations: if we have an interpretation that assigns True to $\sim(\Phi \supset \Psi)$, then it must assign True to Φ and False to Ψ . Hence, it satisfies each child node (and to be sound, we just need at least one branch below to be satisfiable on the interpretation).

However, completeness fails: neither the left nor middle branch entails $\sim(\Phi \supset \Psi)$. Hence, there exist interpretations where either of these branches is true but $\sim(\Phi \supset \Psi)$ is false. Note that the far right branch is a decoy: it is logically equivalent to $\sim(\Phi \supset \Psi)$.

To **formally show** a failure of completeness, we need to construct a counterexample, namely a case where although $\Gamma \models \Theta$, we have a complete open tree with root $\Gamma \cup \sim\{\Theta\}$ (for if the system *were* complete, then we could use this complete open tree to construct a TVA that satisfies the root, which would contradict the root being unsatisfiable!) Hence, it suffices to construct a tree-invalid argument that is semantically valid.

Below is the simplest counterexample: Let both Φ and Ψ be the atomic sentence ‘ Q .’ We construct a tree with a complete open branch that has an unsatisfiable root. (Note that $(Q \supset Q)$ is a tautology, so its negation is unsatisfiable).



Since the left branch is a complete open branch (likewise for the middle branch), this is a complete open tree. So it is tree-invalid. If the modified system were complete, we could use the leftmost branch to construct a TVA that satisfies the root. This TVA would make Q true, but the root $\sim(Q \supset Q)$ remains false; contradiction! So we can conclude that the modified system is not complete.

Other commonly used counterexamples included the argument from $\sim(A \supset B)$ to conclusion A (so $\sim A$ goes in the root), or from B to $(A \supset B)$ (so $\sim(A \supset B)$ goes in the root).