

fl.1 Introduction

nml:fil:int:
sec

One important question about a logic is always whether it is decidable, i.e., if there is an effective procedure which will answer the question “is this formula valid.” Propositional logic is decidable: we can effectively test if a formula is a tautology by constructing a truth table, and for a given formula, the truth table is finite. But we can’t obviously test if a modal formula is true in all models, for there are infinitely many of them. We can list all the finite models relevant to a given formula, since only the assignment of subsets of worlds to propositional variables which actually occur in the formula are relevant. If the accessibility relation is fixed, the possible different assignments $V(p)$ are just all the subsets of W , and if $|W| = n$ there are 2^n of those. If our formula φ contains m propositional variables there are then 2^{nm} different models with n worlds. For each one, we can test if φ is true at all worlds, simply by computing the truth value of φ in each. Of course, we also have to check all possible accessibility relations, but there are only finitely many relations on n worlds as well (specifically, the number of subsets of $W \times W$, i.e., 2^{n^2}).

If we are not interested in the logic **K**, but a logic defined by some class of models (e.g., the reflexive transitive models), we also have to be able to test if the accessibility relation is of the right kind. We can do that whenever the frames we are interested in are definable by modal formulas (e.g., by testing if T and 4 valid in the frame). So, the idea would be to run through all the finite frames, test each one if it is a frame in the class we’re interested in, then list all the possible models on that frame and test if φ is true in each. If not, stop: φ is not valid in the class of models of interest.

There is a problem with this idea: we don’t know when, if ever, we can stop looking. If the formula has a finite countermodel, our procedure will find it. But if it has no finite countermodel, we won’t get an answer. The formula may be valid (no countermodels at all), or it may have only an infinite countermodel, which we’ll never look at. This problem can be overcome if we can show that every formula that has a countermodel has a finite countermodel. If this is the case we say the logic has the *finite model property*.

But how would we show that a logic has the finite model property? One way of doing this would be to find a way to turn an infinite (counter)model of φ into a finite one. If that can be done, then whenever there is a model in which φ is not true, then the resulting finite model also makes φ not true. That finite model will show up on our list of all finite models, and we will eventually determine, for every formula that is not valid, that it isn’t. Our procedure won’t terminate if the formula is valid. If we can show in addition that there is some maximum size that the finite model our procedure provides can have, and that this maximum size depends only on the formula φ , we will have a size up to which we have to test finite models in our search for countermodels. If we haven’t found a countermodel by then, there are none. Then our procedure will, in fact, decide the question “is φ valid?” for any formula φ .

A strategy that often works for turning infinite structures into finite structures is that of “identifying” elements of the structure which behave the same

way in relevant respects. If there are infinitely many worlds in \mathfrak{M} that behave the same in relevant respects, then we may be able to collect *all* worlds in finitely many (possibly infinite) “classes” of such worlds. In other words, we should partition the set of worlds in the right way, i.e., in such a way that each partition contains infinitely many worlds, but there are only finitely many partitions. Then we define a new model \mathfrak{M}^* where the worlds are the partitions. Finitely many partitions in the old model give us finitely many worlds in the new model, i.e., a finite model. Let’s call the partition a world w is in $[w]$. We’ll want it to be the case that $\mathfrak{M}, w \Vdash \varphi$ iff $\mathfrak{M}^*, [w] \Vdash \varphi$, since we want the new model to be a countermodel to φ if the old one was. This requires that we define the partition, as well as the accessibility relation of \mathfrak{M}^* in the right way.

To see how this would go, first imagine we have no accessibility relation. $\mathfrak{M}, w \Vdash \Box\psi$ iff for some $v \in W$, $\mathfrak{M}, v \Vdash \Box\psi$, and the same for \mathfrak{M}^* , except with $[w]$ and $[v]$. As a first idea, let’s say that two worlds u and v are equivalent (belong to the same partition) if they agree on all **propositional variables** in \mathfrak{M} , i.e., $\mathfrak{M}, u \Vdash p$ iff $\mathfrak{M}, v \Vdash p$. Let $V^*(p) = \{[w] : \mathfrak{M}, w \Vdash p\}$. Our aim is to show that $\mathfrak{M}, w \Vdash \varphi$ iff $\mathfrak{M}^*, [w] \Vdash \varphi$. Obviously, we’d prove this by induction: The base case would be $\varphi \equiv p$. First suppose $\mathfrak{M}, w \Vdash p$. Then $[w] \in V^*(p)$ by definition, so $\mathfrak{M}^*, [w] \Vdash p$. Now suppose that $\mathfrak{M}^*, [w] \Vdash p$. That means that $[w] \in V^*(p)$, i.e., for some v equivalent to w , $\mathfrak{M}, v \Vdash p$. But “ w equivalent to v ” means “ w and v make all the same **propositional variables** true,” so $\mathfrak{M}, w \Vdash p$. Now for the inductive step, e.g., $\varphi \equiv \neg\psi$. Then $\mathfrak{M}, w \Vdash \neg\psi$ iff $\mathfrak{M}, w \not\Vdash \psi$ iff $\mathfrak{M}^*, [w] \not\Vdash \psi$ (by inductive hypothesis) iff $\mathfrak{M}^*, [w] \Vdash \neg\psi$. Similarly for the other non-modal operators. It also works for \Box : suppose $\mathfrak{M}^*, [w] \Vdash \Box\psi$. That means that for every $[u]$, $\mathfrak{M}^*, [u] \Vdash \psi$. By inductive hypothesis, for every u , $\mathfrak{M}, u \Vdash \psi$. Consequently, $\mathfrak{M}, w \Vdash \Box\psi$.

In the general case, where we have to also define the accessibility relation for \mathfrak{M}^* , things are more complicated. We’ll call a model \mathfrak{M}^* a *filtration* if its accessibility relation R^* satisfies the conditions required to make the inductive proof above go through. Then any filtration \mathfrak{M}^* will make φ true at $[w]$ iff \mathfrak{M} makes φ true at w . However, now we also have to show that there *are* filtrations, i.e., we can define R^* so that it satisfies the required conditions. In order for this to work, however, we have to require that worlds u, v count as equivalent not just when they agree on all **propositional variables**, but on all **sub-formulas** of φ . Since φ has only finitely many sub-formulas, this will still guarantee that the filtration is finite. There is not just one way to define a filtration, and in order to make sure that the accessibility relation of the filtration satisfies the required properties (e.g., reflexive, transitive, etc.) we have to be inventive with the definition of R^* .

Photo Credits

Bibliography