

ON THE LOGIC OF CATEGORY DEFINITIONS

In their paper on category structures, Gazdar et al. (1988) define a constraint language  $L_C$  for categories and a logic  $\Lambda_C$  of admissible category structures.<sup>1</sup> The intuitive idea is that for a constraint  $\phi$  expressed in  $L_C$ ,  $\phi$  is a **nontrivial** constraint if and only if  $\Lambda_C \not\models \phi$ ; and it is a **satisfiable** constraint if and only if  $\Lambda_C \not\models \neg\phi$ . From a practical point of view it is therefore important to know whether  $\Lambda_C$  is decidable and even better that the decision can be given in a time bounded by a recursive function on the length of  $\phi$ . However, the remarks made in their paper only suffice to show that the modal fragment of  $\Lambda_C$ <sup>2</sup> contains S4.Grz =  $K(\Box p \rightarrow p, \Box p \rightarrow \Box\Box p, \Box(\Box(p \rightarrow \Box p) \rightarrow p) \rightarrow p)$ , which does not show that this fragment is decidable. In this note, I will establish both that the modal fragment of  $\Lambda_C$  and  $\Lambda_C$  itself are decidable, and I will prove it in that order. As a result, I will also axiomatize  $\Lambda_C$ . Thus I show first that the modal reduct of  $\Lambda_C$ , which I call  $\Lambda_M$ , is decidable.

This paper will be rather hardgoing for anyone not acquainted with modal logic. We advise the reader to have Gazdar et al. (1988) at hand while reading this paper, or better still, to read it once through beforehand. For the modal logics we refer the reader to Boolos (1979), Harel (1984), and Segerberg (1971), but in principle any introduction to modal logic will provide enough background to be able to understand the gist of the arguments.

Without going into too many details of the construction, I will show that there is an easy way to give a proof that in fact shows that  $\Lambda_M = \text{S4.Grz}$  using the structure of the models those logics admit. Intuitively, categories correspond to Kripke models. For let  $\alpha$  be a category. Then  $\alpha$  defines a set of categories  $W$ , which is obtained by successively applying type 1<sup>3</sup> features to  $\alpha$ . An accessibility relation  $\triangleleft$  is defined via  $\alpha \triangleleft \beta$  iff  $f(\alpha) = \beta$  for some type 1 feature  $f$ . This accessibility relation is irreflexive, intransitive, finite, and defines a tree-structure on  $W$ . Most importantly, it is cycle free. Thus, if we look at the reflexive and transitive closure  $\triangleleft^+$  of  $\triangleleft$ , it is again finite and has no non-trivial cycles. It therefore is an S4.Grz structure (see, e.g., Boolos 1979). Conversely, an S4.Grz structure  $\langle W, \triangleleft^+ \rangle$  which is a tree can be represented as a category. If we then take a model  $\langle W, \triangleleft^+, val \rangle$  based on that frame, where  $val: X \rightarrow 2^W$  maps a finite set of propositional variables into  $2^W$ , we can code this model by adding a type 0 feature  $f_p$  for each  $p \in X$  that takes values  $\top$  or  $\perp$ . Thus the resulting category  $\alpha$  not only codes the successor function by means of type 1 features, but also the valuation  $val$ .  $W$  is in one-to-one correspondence  $\phi$  with the set  $\Gamma$  of categories generated by  $\alpha$ . We then make the following definitions:

Let  $\beta$  be in  $\Gamma$ :

- i.  $\beta \models f_p : \top$  iff  $\phi(\beta) \in val(p)$  iff  $\phi(\beta) \models p$
- ii.  $\beta \models f_p : \perp$  iff  $\phi(\beta) \in val(p)$  iff  $\phi(\beta) \models \neg p$

It is easy to see that for any modal formula  $P$  with variables in  $X$ , the corresponding translation  $\tau$  induced by  $p \mapsto f_p : \top$  satisfies  $\beta \models \tau(P)$  iff  $\phi(\beta) \models P$ . The logic of  $\langle W, \triangleleft^+ \rangle$  therefore coincides with the logic of all categories that differ from  $\alpha$  only in the assignment of type 0 features. To conclude, the logic of categories as defined in Gazdar et al. (1988) coincides with the logic of all finite, reflexive, transitive trees. It is easily seen that the finite, reflexive, transitive trees generate the class of finite models for S4.Grz. Thus the logic of categories is the logic of the finite models of S4.Grz which, since S4.Grz has the finite model property, is identical to S4.Grz (*end of proof*).

A few remarks are in order:

1. I used a purely semantical argument, which in this case is the most direct way, because it is fairly easy to see why we get just the models we get, though there is some footwork to be done.
2. Alternatively I could have built a canonical model out of a category structure  $\Sigma$ , whose worlds are the categories that  $\Sigma$  admits and whose accessibility relation is as defined above for categories. The proof is essentially the same.
3. The idea of encoding frames and valuations into a single structure has also been explored in Fagin (1985).
4. In Rautenberg (1983) a simple tableau calculus for S4.Grz is given which shows that S4.Grz consistency is effectively decidable, and that the decision procedure is primitive recursive. Furthermore, the size of a tableau is bounded by a function of the number  $\sigma(P)$  of subformulas of  $P$ , or, more precisely, the theoremhood of  $P$  can be decided with a tableau of length  $\leq 2^{7+6\sigma(P)}$ . Given the proof, the same holds for  $\Lambda_M$ , since the translation procedure reduces the size of a formula. So we have the same bound for  $\Lambda_C$ .
5. In Gazdar et al. (1988) another logic is mentioned which arises from restricting the number of type 1 features to 1. The resulting logic is equal to S4.3.Grz =  $\text{S4.Grz}(\Diamond p \wedge \Diamond q. \rightarrow .\Diamond(p \wedge \Diamond q) \vee \Diamond(q \wedge \Diamond p) \vee \Diamond(p \wedge q))$ , the logic of all finite linear orders, as can be seen in the same way. Since finitely generated S4.3.Grz models are finite, this logic is decidable as well.

I will now proceed to the full case. Before I embark on the proof, let me remark on a few things. First, although each particular category structure contains only a finite number of features and values,  $L_C$  contains infinitely many of them. As regards the type 0 features, this causes no problem, since we treat  $f : a$  as a proposition and we allow ourselves infinitely many of those. However, type 1 features will create some problems that are not very serious but have to be dealt with carefully. Second, as we defined a translation of  $L_C$  into modal logic, we will now define a translation of  $L_C$  into elementary propositional dynamic logic (EPDL) so that every type 1 feature has a program associated with it

whose interpretation is an accessibility relation between categories. This translation is harmless and allows us to forget about type 0 features altogether.

- i.  $\tau(f : \alpha) = p_{\langle f, a \rangle}$  where  $f$  is of type 0
- ii.  $\tau(f_i : \phi) = \langle \gamma_i \rangle \phi$  where  $f_i$  is of type 1
- iii.  $\Box \phi = [\alpha] \phi$

$\alpha$  is a program that by definition contains all other programs; that is, if two categories are related by  $\gamma_i$ , they are also related by  $\alpha$ . However, although the intuition is that  $\alpha$  is the reflexive and transitive closure of all the  $\gamma_i$ 's, this fact is not expressible in EPDL nor in  $\Lambda_C$  because it requires a formula of infinite length. But, as it turns out, this is a harmless deficiency of our language. The translation of  $\Lambda_C$  into EPDL will be called  $\Xi$ .

I will now give a full axiomatization of  $\Xi$ . As we saw earlier, the axioms governing the behaviour of  $[\alpha]$  are exactly the axioms for S4.Grz, since  $[\alpha]$  is the old  $\Box$ . Similar reasoning will reveal that the  $[\gamma_i]$  behave alike, and the corresponding logic is the logic otherwise known as **K.Alt<sub>1</sub>**, which stands for ‘‘only one alternative’’. The accessibility relation for **K.Alt<sub>1</sub>** allows a world to have at most one successor. Although it has more models, **K.Alt<sub>1</sub>** is the logic of all irreflexive, linear and finite frames, which shows that it is the logic we are looking for. If we take all this together with the observation that  $\alpha$  includes  $\gamma_i$ , we get the following axioms for  $\Xi$ :

- $Grz_\alpha \Xi \vdash [\alpha]([\alpha](\phi \rightarrow [\alpha]\phi) \rightarrow \phi) \rightarrow \phi$
- $Alt_{1,\gamma} \Xi \vdash \langle \gamma_i \rangle (\phi \wedge \psi) \leftrightarrow (\langle \gamma_i \rangle \phi \wedge \langle \gamma_i \rangle \psi), i \in \omega$
- $Mix \Xi \vdash \langle \gamma_i \rangle \phi \rightarrow \langle \alpha \rangle \phi$

Note that  $\Xi$  is not finitely axiomatizable and so  $\Lambda_C$  isn't either. Note also that if it weren't for the axiom(s) called **Mix**, life would be very easy for us now. Since the axioms for the various programs are independent, the finite model property for each of those programs individually would yield the finite model property for the whole logic by simple induction on the number of programs. Thus let us call the logic without **Mix**  $\Xi^\ominus$ . Also, since we have a tableau calculus for S4.Grz and a tableau calculus for **K.Alt<sub>1</sub>**, we have a calculus for  $\Xi^\ominus$  as well, simply by putting all tableau rules together. The tableau rule for  $\gamma_i$  would look as follows: from  $\Gamma; \langle \gamma_i \rangle \phi$  step to  $\Gamma^\diamond; \phi$ , where  $\Gamma^\diamond := \{\psi \mid \langle \gamma_i \rangle \psi \in \Gamma \text{ or } [\gamma_i]\psi \in \Gamma\}$ . I shall spare the reader an exact specification of the tableau rules and refer him to Rautenberg (1983) again. Note that the length of the tableaux for **K.Alt<sub>1</sub>** is bounded by the size of  $\phi$  so that the actual size of the tableau is at most  $2^{n\sigma(\phi)}$ , where  $n$  is the length of  $\phi$  and  $\sigma(\phi)$  the number of subformulas of  $\phi$ . This bound could be sharpened somewhat but we ignore this point. What I want to show is how the fact that  $\Xi^\ominus$  can be shown to have all those properties can be made to explain why  $\Xi$  must have those properties, too. What the reader should understand at this point is that tableaux are a way of systematically constructing a model for a formula (if it is consistent) and showing inconsistency by exhausting all possible choices, of which there are only finitely many.

What I do now is boost up a model construction procedure for  $\Xi^\ominus$  to a model construction procedure for  $\Xi$ . This is done as follows: Suppose we want to construct a  $\Xi$ -model for  $\phi$ . Since we do not know how to do this we construct a  $\Xi^\ominus$ -model instead. However, this model might be deficient by not respecting **Mix**. Therefore we add a finite set  $\phi^\#$  of instances of **Mix** which will ensure that **Mix** is respected for subformulas of  $\phi$ . The  $\Xi^\ominus$  model can then safely be turned into an  $\Xi$  model.

Let me therefore define the modal degree  $d$  of a formula.

$$d(\phi) = 0, \text{ if } \phi \text{ is a propositional variable or constant}$$

$$d(\neg\phi) = d(\phi)$$

$$d(\phi \wedge \psi) = \max(d(\phi), d(\psi))$$

$$d([\alpha]\phi) = d(\phi) + 1$$

$$d([\gamma_i]\phi) = d(\phi) + 1$$

Furthermore, let  $sf(\phi)$  be the set of subformulas of  $\phi$ . Then define

$$\zeta = \cup \langle \gamma_i \mid \gamma_i \text{ occurs in } \phi \rangle \cup \alpha$$

$$\zeta^1 = \zeta$$

$$\zeta^{n+1} = \zeta; \zeta^n$$

$$\phi^\# = \{[\zeta^{d(\phi)}](\langle \gamma_i \rangle \chi \rightarrow \langle \alpha \rangle \chi) \mid \chi \in sf(\phi)\}$$

Then we say that an  $\Xi$  tableau for  $\phi$  is simply an  $\Xi^\ominus$  tableau for  $\phi; \phi^\#$ . Let us see how  $\phi^\#$  makes everything right for us. The failure of  $\Xi^\ominus$  is to allow models for  $\langle \gamma_i \rangle \psi \wedge [\alpha]\neg\psi$ . For suppose we build a simple  $\Xi^\ominus$  tableau for  $\phi$  and we encounter a line  $\Gamma; \langle \gamma_i \rangle \psi \wedge [\alpha]\neg\psi$ . In the next step we get  $\Gamma; \langle \gamma_i \rangle \psi; [\alpha]\neg\psi$  and then  $\Gamma^\diamond; \psi$ . But if we added  $\phi^\#$ , then  $\Gamma$  would necessarily contain a formula yielding  $\langle \gamma_i \rangle \psi \rightarrow \langle \alpha \rangle \psi$ , which would close this branch of the tableau. Thus an  $\Xi^\ominus$  tableau for  $\phi; \phi^\#$  results in a model  $\langle W, \triangleleft, val \rangle$  in which, though  $\alpha$  is an independent program, for every subformula  $\psi$  of  $\phi$ , if  $s, val \models \langle \gamma_i \rangle \psi$ , then also  $s, val \models \langle \alpha \rangle \psi$ . Thus it is easy to see that if we now reinterpret the  $\alpha$  relation  $\triangleleft_\alpha$  as the reflexive, transitive closure of  $\alpha$  and the  $\gamma_i$ , we get an  $\Xi$  model  $\langle W, \triangleleft, val \rangle$  for  $\phi$ , which obviously is of the same size. Thus, if  $\Xi^\ominus$  has the finite model property,  $\Xi$  has the finite model property, and if the decision procedure for  $\Xi^\ominus$  is bounded a priori by a function on the length of  $\phi$ , the same holds for  $\Xi$ . Obviously, the bound is much higher than for  $\Xi^\ominus$  because of the  $\phi^\#$ , but this is the price we have to pay (*end of proof*).

Given the above proof we can now change the tableau calculus for  $\Xi$  by redefining the rule for  $\gamma_i$  to the following: from  $\Gamma; \langle \gamma_i \rangle \phi$  infer  $\Gamma^\diamond; \Gamma^\square; \phi$  where  $\Gamma^\diamond := \{\psi \mid \langle \gamma_i \rangle \psi \in \Gamma \text{ or } [\gamma_i]\psi \in \Gamma\}$  and  $\Gamma^\square := \{[\alpha]\psi \mid [\alpha]\psi \in \Gamma\} \cup \{\psi \mid [\alpha]\psi \in \Gamma\}$ . The proof method I used for proving decidability for  $\Xi$  ( $\Lambda_C$ ) from the decidability of  $\Xi^\ominus$  is explained in full detail in Kracht (1988). The same method applies to the case when we only allow a single type 1 feature, since it only requires the base logic—in

this case S4.3.Grz—to have the finite model property. The resulting logic  $\Xi.3$  can be (finitely) axiomatized as follows:

$$\begin{aligned} \text{Grz}_\alpha \quad \Xi.3 \vdash & [\alpha]([\alpha](\phi \rightarrow [\alpha]\phi) \rightarrow \phi) \rightarrow \phi \\ .3_\alpha \quad \Xi.3 \vdash & \langle \alpha \rangle \phi \wedge \langle \alpha \rangle \psi \rightarrow \\ & \langle \alpha \rangle (\phi \wedge \langle \alpha \rangle \psi) \vee \langle \alpha \rangle (\psi \wedge \langle \alpha \rangle \phi) \\ & \vee \langle \alpha \rangle (\phi \wedge \psi) \\ \text{Alt}_{1,\gamma} \quad \Xi.3 \vdash & \langle \gamma_1 \rangle \phi \wedge \langle \gamma_1 \rangle \psi \leftrightarrow \langle \gamma_1 \rangle (\phi \wedge \psi) \\ \text{Mix} \quad \Xi.3 \vdash & \langle \gamma_1 \rangle \phi \rightarrow \langle \alpha \rangle \phi \end{aligned}$$

The model construction requires some care since we do not have an unlimited resort of extra features, but it can be done in the same spirit. This shows decidability for  $\Xi.3$  and hence for the corresponding logic mentioned in the paper. An alternative formulation of  $\Xi.3$  could be given with the help of propositional dynamic logic using the star operator  $*$ . We would then simply have an axiom  $\langle \gamma_1^* \rangle \phi \leftrightarrow \langle \alpha \rangle \phi$  that says nothing else, but that  $\alpha$  is the reflexive and transitive closure of  $\gamma_1$ . This trick would also work if we restrict  $\Xi$  to any finite number of features. But the star does not gain us much for  $\Xi$  itself since we still could not express the fact that  $\alpha$  is the reflexive and transitive closure of the  $\gamma_i$ 's because we have infinitely many of them.

Marcus Kracht II  
 Mathematisches Institut  
 FU Berlin  
 1000 Berlin 33, Germany

ACKNOWLEDGMENTS

This paper was written while I was at the Centre for Cognitive Science in Edinburgh. I wish to thank Jaap van der Does for encouraging me

to write this proof down and for proofreading it. I also wish to thank G. Gazdar for remarks on an earlier version of the paper and an anonymous referee for further suggestions.

REFERENCES

Boolos, G. 1979 *The Unprovability of Consistency*. Cambridge University Press, Cambridge, England.  
 Fagin, R. and Vardi, M. 1985 An Internal Semantics for Modal Logics: Preliminary Report, CSLI-Report No. 85-25.  
 Gazdar, G.; Pullum, G.; Carpenter, R.; Klein, E.H.; Hukari, T.E.; and Levine, R.D. 1988 Category Structures, *Computational Linguistics* 14.1: 1-19.  
 Harel, D. 1984 Dynamic Logic. in Gabbay, D. and Guentner, F. (eds.) *Handbook of Philosophical Logic*. Reidel, Dordrecht, Holland.  
 Kracht, M. 1988 Splittings and the Finite Model Property (forthcoming).  
 Rautenberg, W. 1983 Modal Tableau Calculi and Interpolation, in *Journal of Philosophical Logic* 12: 403-423.  
 Segerberg, K. 1971 *An Essay in Classical Modal Logic*. Uppsala.

NOTES

1. Unfortunately, they do not distinguish between the language  $L_C$  and the logic, which defines a subset of that language, namely the set of its theorems. We make this distinction here by calling the logic as well as the set of theorems it defines  $\Lambda_C$ .
2. We define a logic as a set of rules, which are pairs  $\Delta/\phi$ , where  $\Delta$  is the set of premises of that rule and  $\phi$  its consequence. Modus Ponens thus takes the form  $\phi, \phi \rightarrow \psi/\psi$ . Rules are closed under substitution. Axioms are rules because we can take  $\Delta = \emptyset$ . The modal fragment of  $\Lambda_C$  is then simply the subpart of rules that only involve modal formulas, i.e. no type 1 features.
3. Remember that type 1 features take propositions as values, whereas type 0 features only take atoms.