Interpretations as coalgebra morphisms

Manuel A. Martins
Dep. Mathematics
University of Aveiro
martins@ua.pt

Alexandre Madeira
Dep. Informatics, U. Minho
Dep. Mathematics, U. Aveiro
Critical Software S.A.
madeira@ua.pt

Luis S. Barbosa
Dep. Informatics & CCTC
University of Minho
lsb@di.uminho.pt

1 Motivations

The concept of *logic or deductive system* is transversal in computer science. Several notions have been proposed on the literature, some of them, highly abstract in order to capture a wide class of logics. An interesting one is the formalization of a logic as an algebra together with a closure operator over that universe (or, equivalently, as an algebra together with a closure system over its universe (eg. [2])). Palmigiano in [6] related this *theory of abstract logic* with the *theory of coalgebras*, another central field in computer science. She shows that an abstract logic can be represented as a $\mathscr{C}$-coalgebra over **Set** with $\mathscr{C}$ the *closure system contravariant functor*; moreover the coalgebraic morphisms are the strict morphisms between the correspondent original logics, i.e., algebraic morphisms such that the pre-images of closed sets of the target logic are exactly the closed sets of the source logic.

In the context of algebraic specification refinement, we came recently interested in relating "equivalent logics" by maps which fail to be signature morphisms, in particular, in [3] and [4], we have studied how refinements can be witnessed by logical interpretation. A logical interpretation is a multifunction f that preserves consequence in the following sense: given two logics $\mathscr{A} = \langle A, C_A \rangle$ and $\mathscr{B} = \langle B, C_B \rangle$, for all $\{x\} \cup X \subseteq A$, $x \in C_A([X])$ iff $f(x) \subseteq C_B(f[X])$. A paradigmatic example of an interpretation, is multifunction $\tau : \mathrm{Fm(Bool)} \to \mathrm{Fm(CPC)}$ from boolean equations to propositional terms defined, for any $\varphi \approx \varphi' \in \mathrm{Fm(Bool)}$, by $\tau(\varphi \approx \varphi') = \{\varphi \to \varphi', \varphi' \to \varphi\}$. The aim of the present work is to frame logics and interpretations along the lines of the coalgebraic perspective forward in [6].

2 The category of logics and interpretations and category $Coalg(\mathscr{C})$

The *category of logics and interpretations* is defined as $\langle \mathbf{Log}, \mathbf{Int}, \mathbf{i}, \circ \rangle$, where **Log** is the class of abstract logics, **Int** the class of its interpretations, **i** the class of identical maps (for each abstract logic $\mathscr{A}$, a multifunction $i_{\mathscr{A}}(x) = \{x\}$) and $\circ$ is the usual composition of multifunctions. We show that this category has a strong relation with the category $Coalg(\mathscr{C})$ on **Pw**, in which the objects are $\{\mathscr{P}(X) | X \in Obj(\mathbf{Set})\}$ and, the arrows, the functions between them. Namely, all abstract logics are $\mathscr{C}$-coalgebras and, its interpretations the respective coalgebraic morphisms.

A *minimal logic of* $\mathscr{A} = \langle A, C_A \rangle$ is defined as the logic $\mathscr{A}^* = \langle A_\sim, C_{\sim_{\mathscr{A}}} \rangle$ where $\sim$ is the equivalence relation which identifies elements with the same consequence, and $C_{\sim_{\mathscr{A}}}$, the closure operator coinduced by C_A. Interesting considerations can be proved concerning these minimizations: canonical projections with respect to $\sim$ are interpretations; moreover, given two abstract logics, there exists an interpretation between them if and only if there is an interpretation between the respective minimizations. This construction may overcame the pairwise redundancy on the axiomatizations of logics, since two properties with equal consequence are identified, via the canonical projection, in its minimization. This aspect seems to be useful for the software specification process where, for instance, the redundancy of proof obligations is naturally undesirable. Similar results are proved in [1] in the context of the theory of the conservative translations of logics.

3 Categorical approach to refinement via interpretation

Stepwise refinement of algebraic specifications is a well known formal methodology for software development. However, traditional notions of refinement based on signature morphisms are often too rigid to capture a number of relevant transformations in the context of software design, reuse, and adaptation. Based on this assumptions the authors suggested in [3, 4] an alternative notion of specification refinement, called *refinement via interpretation* where translations induced by signature morphisms are replaced by logical interpretations. The idea of this formalization is based on the existence of a largest interpretation: given a multifunction suitable to interpret a specification, there is a largest interpretation of that specification (i.e., a specification with the largest class of models) which may be defined as the largest specification whose models satisfy the translation of all properties that the refined specification satisfies. Based on this fact, we take as admissible refinements all the strict refinements of those interpretations.

Considering specifications as abstract logics and approaching consequences as coalgebras, an elegant view of specification refinement can be achieved. Namely, the *category of specifications and strict refinements* is defined and a categorical formalization of refinement via interpretation is presented. Morphisms of this category, the strict refinements, are *forward morphisms* in sense of [5], more precisely, inclusion forward morphisms with respect to the set-theoretic inclusion. In this context we characterize refinement via interpretation as follows: specification SP' refines SP if there is a commuting diagram

$$\begin{array}{ccccccccc} A & \xrightarrow{int'} & \cdots & \xrightarrow{int} & B & \xrightarrow{ref} & \cdots & \xrightarrow{ref'} & B \\ \downarrow{\scriptstyle SP} & & & & \downarrow & & & & \downarrow{\scriptstyle SP'} \\ \mathscr{C}A & \xleftarrow[\mathscr{C}(int')]{} & \cdots & \xleftarrow[\mathscr{C}(int)]{} & \mathscr{C}B & \xleftarrow[\mathscr{C}(ref)]{} & \cdots & \xleftarrow[\mathscr{C}(ref')]{} & \mathscr{C}B \end{array}$$

where the left and right squares commute in the category of specifications and interpretations and in the category of specifications and refinements respectively. This result naturally holds since, the morphisms of the mentioned categories are interpretations and strict refinements.

References

[1] H. Feitosa. *Traduções Conservativas*. PhD thesis, Universidade Federal de Campinas,Instituto de Filosofia e Ciências Humanas, 1997.

[2] R. Jansana and J. Font. *A general Algebraic Semantics for Sentential Logics , 2nd edition*. Lecture Notes in Logic.

[3] M. A. Martins, A. Madeira, and L. S. Barbosa. Refinement by interpretation in a general setting. *Electron. Notes Theor. Comput. Sci.*, 259:105–121, 2009.

[4] M. A. Martins, A. Madeira, and L. S. Barbosa. Refinement via interpretation. In D. V. Hung and P. Krishnan, editors, *SEFM*, pages 250–259. IEEE Computer Society, 2009.

[5] S. Meng and L. S. Barbosa. Components as coalgebras: The refinement dimension. *Theor. Comput. Sci.*, 351(2):276–294, 2006.

[6] A. Palmigiano. Abstract logics as dialgebras. *Electr. Notes Theor. Comput. Sci.*, 65(1), 2002.