

Logical Semantics (Abstract)

M. Dezani-Ciancaglini Università di Torino - Dipartimento di Informatica
Corso Svizzera 185 - 10149 Torino (Italy) e-mail: `dezani@di.unito.it`

Stone dualities allow to describe special classes of topological spaces by means of (possibly finitary) partial orders. Typically, these partial orders are given by the topology, a basis for it, or a subbasis for it. The seminal result is the duality between the categories of Stone spaces and that of Boolean algebras (see [23]). Other very important examples are the descriptions of *Scott domains as information systems* [27] and the description of *SFP domains as pre-locales* [1]. It is worthwhile to mention also Martin-Löf's domain interpretation of intuitionistic type theory [24].

Intersection types can be viewed also as a restriction of the domain theory in logical form, see [1], to the special case of modelling pure lambda calculus by means of ω -algebraic complete lattices. Intersection types have been used as a powerful tool both for the analysis and the synthesis of λ -models, see *e.g.* [3], [8], [2], [17], [16], [25], [21], [26]. On the one hand, intersection type disciplines provide finitary inductive definitions of interpretation of λ -terms in models. On the other hand, they are suggestive for the shape the domain model has to have in order to exhibit certain properties, [9], [22], [13], [14], [28], [15].

More recently intersection (together with union) types have been used to build *fully abstract* models of extensions of the λ -calculus including parallel features [4], [11], [12], [5], of Higher-Order Processes [19], [20], [18], of the π -calculus [10] and of ambient calculi [7], [6].

References

- [1] Samson Abramsky. Domain theory in logical form. *Ann. Pure Appl. Logic*, 51(1-2):1–77, 1991.
- [2] Fabio Alessi. *Strutture di tipi, teoria dei domini e modelli del lambda calcolo*. PhD thesis, Torino University, 1991.
- [3] Henk Barendregt, Mario Coppo, and Mariangiola Dezani-Ciancaglini. A filter lambda model and the completeness of type assignment. *J. Symbolic Logic*, 48(4):931–940 (1984), 1983.
- [4] Gérard Boudol. Lambda-calculi for (strict) parallel functions. *Inform. and Comput.*, 108(1):51–127, 1994.
- [5] Gérard Boudol, Pierre-Louis Curien, and Carolina Lavatelli. A semantics for lambda calculi with resources. *Math. Struct. Comput. Sci.*, 9(4):483–506, 1999.
- [6] Mario Coppo and Mariangiola Dezani-Ciancaglini. A fully abstract model for higher-order mobile ambients. In *VMCAI'02*, LNCS 2294, pages 255–271. Springer, 2002.
- [7] Mario Coppo and Mariangiola Dezani-Ciancaglini. A fully abstract model for mobile ambients. In *TOSCA'01*, ENTCS 62. Elsevier Science B. V., 2002.
- [8] Mario Coppo, Mariangiola Dezani-Ciancaglini, Furio Honsell, and Giuseppe Longo. Extended type structures and filter lambda models. In *Logic colloquium '82*, pages 241–262. North-Holland, Amsterdam, 1984.
- [9] Mario Coppo, Mariangiola Dezani-Ciancaglini, and Maddalena Zacchi. Type theories, normal forms, and D_∞ -lambda-models. *Inform. and Comput.*, 72(2):85–116, 1987.

- [10] Ferruccio Damiani, Mariangiola Dezani-Ciancaglini, and Paola Giannini. A filter model for mobile processes. *Math. Struct. Comput. Sci.*, 9(1):63–101, 1999.
- [11] Mariangiola Dezani-Ciancaglini, Ugo de’Liguoro, and Adolfo Piperno. Filter models for conjunctive-disjunctive λ -calculi. *Theoret. Comput. Sci.*, 170(1-2):83–128, 1996.
- [12] Mariangiola Dezani-Ciancaglini, Ugo de’Liguoro, and Adolfo Piperno. A filter model for concurrent λ -calculus. *SIAM J. Comput.*, 27(5):1376–1419 (electronic), 1998.
- [13] Mariangiola Dezani-Ciancaglini, Silvia Ghilezan, and Silvia Likavec. Behavioural inverse limit models. *Theoret. Comput. Sci.*, 316(1–3):49–74, 2004.
- [14] Mariangiola Dezani-Ciancaglini, Furio Honsell, and Yoko Motohama. Compositional characterization of λ -terms using intersection types. *Theoret. Comput. Sci.*, 340(3):459–495, 2005.
- [15] Mariangiola Dezani-Ciancaglini and Makoto Tatsuta. A Behavioural Model for Klop’s Calculus. In Flavio Corradini and Carlo Toffalori, editors, *Logic, Model and Computer Science*, volume 169 of *ENTCS*, pages 19–32. Elsevier, 2007.
- [16] Pietro Di Gianantonio and Furio Honsell. An abstract notion of application. In *TLCA’93*, volume 664 of *LNCS*, pages 124–138. Springer, Berlin, 1993.
- [17] Lavinia Egidi, Furio Honsell, and Simona Ronchi Della Rocca. Operational, denotational and logical descriptions: a case study. *Fund. Inform.*, 16(2):149–169, 1992.
- [18] Chrysafis Hartonas and Matthew Hennessy. Full abstractness for a functional/concurrent language with higher-order value-passing. *Inform. and Comput.*, 145(1):64–106, 25 August 1998.
- [19] M. Hennessy. A fully abstract denotational model for higher-order processes. *Inform. and Comput.*, 112(1):55–95, July 1994.
- [20] Matthew Hennessy. Higher-order process and their models. In *ICALP’94*, volume 820 of *LNCS*, pages 286–303. Springer, 1994.
- [21] Furio Honsell and Marina Lenisa. Semantical analysis of perpetual strategies in λ -calculus. *Theoret. Comput. Sci.*, 212(1-2):183–209, 1999.
- [22] Furio Honsell and Simona Ronchi Della Rocca. An approximation theorem for topological lambda models and the topological incompleteness of lambda calculus. *J. Comput. System Sci.*, 45(1):49–75, 1992.
- [23] Peter T. Johnstone. *Stone spaces*. Cambridge University Press, Cambridge, 1986. Reprint of the 1982 edition.
- [24] Per Martin-Löf. Lecture notes on domain interpretation of type theory. Programming Methodology Group, Workshop on the Semantics of Programming Languages, Chalmers University of Technology, 1983.
- [25] Gordon D. Plotkin. Set-theoretical and other elementary models of the λ -calculus. *Theoret. Comput. Sci.*, 121(1-2):351–409, 1993.
- [26] Alberto Pravato, Simona Ronchi, and Luca Roversi. The call-by-value lambda calculus: a semantic investigation. *Math. Struct. Comput. Sci.*, 9(5):617–650, 1999.
- [27] Dana S. Scott. Domains for denotational semantics. In *ICALP’82*, volume 140, pages 577–613. Springer, Berlin, 1982.
- [28] Makoto Tatsuta and Mariangiola Dezani-Ciancaglini. Normalisation is Insensible to Lambda-term Identity or Difference. In Rajeev Alur, editor, *LICS’06*, pages 327–336. IEEE Computer Society, 2006.