

**Al-Sibahi, Ahmad Salim; Jensen, Thomas; Møgelberg, Rasmus Ejlers; Wąsowski, Andrzej**  
**Galois connections for recursive types.** (English) [[Zbl 1440.68042](#)]

Di Pierro, Alessandra (ed.) et al., From lambda calculus to cybersecurity through program analysis. Essays dedicated to Chris Hankin on the occasion of his retirement. Cham: Springer. Lect. Notes Comput. Sci. 12065, 105-131 (2020).

Summary: Building a static analyser for a real language involves modeling of large domains capturing the many available data types. To scale domain design and support efficient development of project-specific analyzers, it is desirable to be able to build, extend, and change abstractions in a systematic and modular fashion. We present a framework for modular design of abstract domains for recursive types and higher-order functions, based on the theory of solving recursive domain equations. We show how to relate computable abstract domains to our framework, and illustrate the potential of the construction by modularizing a monolithic domain for regular tree grammars. A prototype implementation in the dependently typed functional language Agda shows how the theoretical solution can be used in practice to construct static analysers.

For the entire collection see [[Zbl 1435.68026](#)].

#### MSC:

- [68N18](#) Functional programming and lambda calculus
- [06A15](#) Galois correspondences, closure operators (in relation to ordered sets)
- [68N15](#) Theory of programming languages
- [68N30](#) Mathematical aspects of software engineering (specification, verification, metrics, requirements, etc.)
- [68Q42](#) Grammars and rewriting systems

#### Software:

[Agda](#)

**Full Text:** [DOI](#)

#### References:

- [1] Abbott, M.G., Altenkirch, T., Ghani, N.: Containers: constructing strictly positive types. *Theoret. Comput. Sci.* 342(1), 3-27 (2005) · [Zbl 1077.68015](#) · [doi:10.1016/j.tcs.2005.06.002](#)
- [2] Abramsky, S., Jung, A.: Domain theory. In: Abramsky, S., Gabbay, D.M., Maibaum, T.S.E. (eds.) *Handbook of Logic in Computer Science*, vol. 3, pp. 1-168. Clarendon Press, Oxford (1994)
- [3] Aiken, A., Murphy, B.R.: Implementing regular tree expressions. In: Hughes, J. (ed.) *FPCA 1991*. LNCS, vol. 523, pp. 427-447. Springer, Heidelberg (1991). [https://doi.org/10.1007/3540543961\\_21](https://doi.org/10.1007/3540543961_21) · [doi:10.1007/3540543961\\_21](#)
- [4] Awodey, S.: *Category Theory*. Oxford University Press, Oxford (2011) · [Zbl 1100.18001](#)
- [5] Benton, P.N.: Strictness properties of lazy algebraic datatypes. In: Cousot, P., Falaschi, M., Filé, G., Rauzy, A. (eds.) *WSA 1993*. LNCS, vol. 724, pp. 206-217. Springer, Heidelberg (1993). [https://doi.org/10.1007/3-540-57264-3\\_42](https://doi.org/10.1007/3-540-57264-3_42) · [doi:10.1007/3-540-57264-3\\_42](#)
- [6] Bodin, M., Gardner, P., Jensen, T., Schmitt, A.: Skeletal semantics and their interpretations. *PACMPL* 3(POPL), 44:1-44:31 (2019)
- [7] Chang, B.E., Rival, X.: Relational inductive shape analysis. In: Necula, G.C., Wadler, P. (eds.) *POPL 2008*, pp. 247-260. ACM (2008) · [Zbl 1295.68081](#)
- [8] Chang, B.-Y.E., Rival, X., Necula, G.C.: Shape analysis with structural invariant checkers. In: Nielson, H.R., Filé, G. (eds.) *SAS 2007*. LNCS, vol. 4634, pp. 384-401. Springer, Heidelberg (2007). [https://doi.org/10.1007/978-3-540-74061-2\\_24](https://doi.org/10.1007/978-3-540-74061-2_24) · [doi:10.1007/978-3-540-74061-2\\_24](#)
- [9] Cousot, P.: The calculational design of a generic abstract interpreter. In: Broy, M., Steinbrüggen, R. (eds.) *Calculational System Design*. NATO ASI Series F. IOS Press, Amsterdam (1999) · [Zbl 0945.68032](#)
- [10] Cousot, P., Cousot, R.: Abstract interpretation: a unified lattice model for static analysis of programs by construction or approximation of fixpoints. In: Graham, R.M., Harrison, M.A., Sethi, R. (eds.) *POPL 1977*, pp. 238-252. ACM (1977)
- [11] Cousot, P., Cousot, R.: Invited talk: higher order abstract interpretation (and application to compartment analysis generalizing

- strictness, termination, projection, and PER analysis. In: Bal, H.E. (ed.) Proceedings of the IEEE Computer Society 1994 International Conference on Computer Languages, Toulouse, France, 16-19 May 1994, pp. 95-112. IEEE Computer Society (1994)
- [12] Cousot, P., Cousot, R.: Formal language, grammar and set-constraint-based program analysis by abstract interpretation. In: FPCA 1995, pp. 170-181. ACM (1995)
- [13] Cousot, P., Cousot, R.: Modular static program analysis. In: Horspool, R.N. (ed.) CC 2002. LNCS, vol. 2304, pp. 159-179. Springer, Heidelberg (2002). [https://doi.org/10.1007/3-540-45937-5\\_13](https://doi.org/10.1007/3-540-45937-5_13) · Zbl 1051.68624 · doi:10.1007/3-540-45937-5\_13
- [14] Darais, D., Labich, N., Nguyen, P.C., Horn, D.V.: Abstracting definitional interpreters (functional pearl). PACMPL 1(ICFP), 12:1-12:25 (2017)
- [15] Jensen, T.P.: Disjunctive program analysis for algebraic data types. ACM Trans. Program. Lang. Syst. 19(5), 751-803 (1997) · doi:10.1145/265943.265966
- [16] Journault, M., Miné, A., Ouadjaout, A.: Modular static analysis of string manipulations in C programs. In: Podelski, A. (ed.) SAS 2018. LNCS, vol. 11002, pp. 243-262. Springer, Cham (2018). [https://doi.org/10.1007/978-3-319-99725-4\\_16](https://doi.org/10.1007/978-3-319-99725-4_16) · doi:10.1007/978-3-319-99725-4\_16
- [17] Keidel, S., Poulsen, C.B., Erdweg, S.: Compositional soundness proofs of abstract interpreters. PACMPL 2(ICFP), 72:1-72:26 (2018)
- [18] Midtgaard, J., Jensen, T.: A calculational approach to control-flow analysis by abstract interpretation. In: Alpuente, M., Vidal, G. (eds.) SAS 2008. LNCS, vol. 5079, pp. 347-362. Springer, Heidelberg (2008). [https://doi.org/10.1007/978-3-540-69166-2\\_23](https://doi.org/10.1007/978-3-540-69166-2_23) · Zbl 1149.68359 · doi:10.1007/978-3-540-69166-2\_23
- [19] Nielson, F., Nielson, H.R.: The tensor product in wadler's analysis of lists. Sci. Comput. Program. 22(3), 327-354 (1994) · Zbl 0810.68092 · doi:10.1016/0167-6423(94)00009-3
- [20] Nielson, F., Nielson, H.R., Hankin, C.: Principles of Program Analysis. Springer, Heidelberg (1999). <https://doi.org/10.1007/978-3-662-03811-6> · Zbl 0932.68013 · doi:10.1007/978-3-662-03811-6
- [21] Norell, U.: Dependently typed programming in agda. In: Koopman, P., Plasmeijer, R., Swierstra, D. (eds.) AFP 2008. LNCS, vol. 5832, pp. 230-266. Springer, Heidelberg (2009). [https://doi.org/10.1007/978-3-642-04652-0\\_5](https://doi.org/10.1007/978-3-642-04652-0_5) · Zbl 1263.68038 · doi:10.1007/978-3-642-04652-0\_5
- [22] Rival, X., Toubhans, A., Chang, B.-Y.E.: Construction of abstract domains for heterogeneous properties (position paper). In: Margaria, T., Steffen, B. (eds.) ISoLA 2014. LNCS, vol. 8803, pp. 489-492. Springer, Heidelberg (2014). [https://doi.org/10.1007/978-3-662-45231-8\\_40](https://doi.org/10.1007/978-3-662-45231-8_40) · doi:10.1007/978-3-662-45231-8\_40
- [23] Streicher, T.: Domain-Theoretic Foundations of Functional Programming. World Scientific Publishing Company, Singapore (2006) · Zbl 1111.68020 · doi:10.1142/6284
- [24] Tarski, A.: A lattice-theoretical fixpoint theorem and its applications. Pac. J. Math. 5(2), 285-309 (1955) · Zbl 0064.26004 · doi:10.2140/pjm.1955.5.285
- [25] Toubhans, A. · Zbl 1426.68066 · doi:10.1007/978-3-642-35873-9\_23

This reference list is based on information provided by the publisher or from digital mathematics libraries. Its items are heuristically matched to zbMATH identifiers and may contain data conversion errors. It attempts to reflect the references listed in the original paper as accurately as possible without claiming the completeness or perfect precision of the matching.