

**Pun, Ka I; Steffen, Martin; Stolz, Volker**

**Effect-polymorphic behaviour inference for deadlock checking.** (English) Zbl 1392.68148  
*J. Log. Algebr. Methods Program.* 85, No. 6, 1234-1267 (2016).

**Summary:** We present a constraint-based effect inference algorithm for deadlock checking. The static analysis is developed for a concurrent calculus with higher-order functions and dynamic lock creation, where the locks are summarised based on their creation-site. The analysis is context-sensitive and the resulting effects can be checked for deadlocks using state space exploration. We use a specific deadlock-sensitive simulation relation to show that the effects soundly over-approximate the behaviour of a program, in particular that deadlocks in the program are preserved in the effects.

**MSC:**

- 68N30 Mathematical aspects of software engineering (specification, verification, metrics, requirements, etc.)
- 68Q85 Models and methods for concurrent and distributed computing (process algebras, bisimulation, transition nets, etc.)

**Keywords:**

deadlock detection; simulation; type and effect system; concurrency; formal method

**Software:**

Eraser; Featherweight Java; Goblint; LOCKSMITH; ML ; RacerX

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

**References:**

- [1] Agarwal, R.; Bensalem, S.; Farchi, E.; Havelund, K.; Nir-Buchbinder, Y.; Stoller, S. D.; Ur, S.; Wang, L., Detection of deadlock potentials in multithreaded programs, *IBM J. Res. Dev.*, 54, 5, 3:1-3:15, (2010)
- [2] Agarwal, R.; Sasturkar, A.; Stoller, S. D., Type discovery for parameterized race-free Java, (2004), Dept. of Computer Science, SUNY, Stony Brook, Tech. rep. DAR-04-16
- [3] Agarwal, R.; Stoller, S. D., Type inference for parameterized race-free Java, (Steffen, B.; Levi, G., Proceedings of the 5th International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI), Lecture Notes in Computer Science, vol. 2937, (2004), Springer), 149-160
- [4] Agarwal, R.; Wang, L.; Stoller, S. D., Detecting potential deadlocks with static analysis and run-time monitoring, (Ur, S.; Bin, E.; Wolfsthal, Y., Proceedings of the Haifa Verification Conference 2005, Lecture Notes in Computer Science, vol. 3875, (2006), Springer), 191-207
- [5] Amtoft, T.; Nielson, H. R.; Nielson, F., Type and effect systems: behaviours for concurrency, (1999), Imperial College Press
- [6] Bensalem, S.; Havelund, J.; Havelund, K.; Mounier, L., Confirmation of deadlock potentials detected by runtime analysis, (Ur, S.; Farchi, E., Proceedings of the 4th Workshop on Parallel and Distributed Systems: Testing, Analysis, and Debugging, (2006), ACM), 41-50
- [7] Bensalem, S.; Havelund, K., Dynamic deadlock analysis of multi-threaded programs, (Ur, S.; Bin, E.; Wolfsthal, Y., Proceedings of the Haifa Verification Conference 2005, Lecture Notes in Computer Science, vol. 3875, (2006), Springer), 208-223
- [8] Boudol, G., A deadlock-free semantics for shared memory concurrency, (Leucker, M.; Morgan, C. C., Proceedings of the 6th International Colloquium on Theoretical Aspects of Computing (ICTAC), Lecture Notes in Computer Science, vol. 5684, (2009), Springer), 140-154 · [Zbl 1250.68207](#)
- [9] Boyapati, C.; Lee, R.; Rinard, M., Ownership types for safe programming: preventing data races and deadlocks, (Proceedings of 17th ACM SIGPLAN Conference on Object Oriented Programming: Systems, Languages, and Applications (OOPSLA), (2002), ACM), 211-230
- [10] Clarke, E.; Grumberg, O.; Jha, S.; Lu, Y.; Veith, H., Counterexample-guided abstraction refinement, (Emerson, E. A.; Sistla, A. P., Proceedings of the 12th International Conference on Computer-Aided Verification (CAV), Lecture Notes in Computer Science, vol. 1855, (2000), Springer), 154-169 · [Zbl 0974.68517](#)
- [11] Coffman, E. G.; Elphick, M.; Shoshani, A., System deadlocks, *Comput. Surv.*, 3, 2, 67-78, (1971) · [Zbl 0226.68015](#)
- [12] Corbett, J., Evaluating deadlock detection methods for concurrent software, *IEEE Trans. Softw. Eng.*, 22, 3, 161-180, (1996)

- [13] Damas, L., Type assignment in programming languages, (1985), Laboratory for Foundations of Computer Science, University of Edinburgh, Ph.D. thesis
- [14] Damas, L.; Milner, R., Principal type-schemes for functional programming languages, (Proceedings of the 9th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL), (1982), ACM), 207-212
- [15] de Boer, F. S.; Bravetti, M.; Grabe, I.; Lee, M.; Steffen, M.; Zavattaro, G., A Petri net based analysis of deadlock for active objects and futures, (Pasareanu, C. S.; Salaün, G., Revised Selected Papers of the 9th International Workshop on Formal Aspects of Component Software, FACS 2012, Lecture Notes in Computer Science, (2013), Springer), 110-127
- [16] Dijkstra, E. W., Cooperating sequential processes, (1965), Technological University Eindhoven, Tech. rep. EWD-123
- [17] Engler, D. R.; Ashcraft, K., Racerx: effective, static detection of race conditions and deadlocks, (Proceedings of the 19th ACM Symposium on Operating Systems Principles, (2003), ACM), 237-252
- [18] Flanagan, C.; Sabry, A.; Duba, B. F.; Felleisen, M., The essence of compiling with continuations, (Proceedings of the 1993 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI), (1993), ACM), 237-247
- [19] Gerakios, P.; Papaspyrou, N.; Sagonas, K., A type and effect system for deadlock avoidance in low-level languages, (Proceedings of the 6th ACM SIGPLAN International Workshop on Types in Languages Design and Implementation (TLDI), (2011), ACM), 15-28
- [20] Giachino, E.; Grazia, C. A.; Laneve, C.; Lienhardt, M.; Wong, P. Y., Deadlock analysis of concurrent objects: theory and practice, (Johnsen, E. B.; Petre, L., Proceedings of the 11th International Conference on Integrated Formal Methods (iFM), Lecture Notes in Computer Science, vol. 7940, (2013), Springer), 394-411
- [21] Giachino, E.; Laneve, C., Analysis of deadlocks in object groups, (Bruni, R.; Dingel, J., Formal Techniques for Distributed Systems (FMOODS-FORTE), Lecture Notes in Computer Science, vol. 6722, (2011), Springer), 168-182
- [22] Giachino, E.; Laneve, C., Deadlock detection in linear recursive programs, (Bernardo, M.; Damiani, F.; Hähnle, R.; Johnsen, E. B.; Schaefer, I., Advanced Lectures from the 14th International School on Formal Methods for the Design of Computer, Communication, and Software Systems (SFM), Lecture Notes in Computer Science, vol. 8483, (2014), Springer), 26-64 · [Zbl 1445.68052](#)
- [23] Gordon, S. C.; Ernst, M. D.; Grossman, D., Static lock capabilities for deadlock freedom, (Proceedings of the 8th ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI), (2012), ACM), 67-78
- [24] Gosling, J.; Joy, B.; Steele, G. L.; Bracha, G., The Java language specification, (2000), Addison-Wesley
- [25] Harrow, J., Runtime checking of multithreaded applications with visual threads, (Havelund, K.; Penix, J.; Visser, W., Proceedings of the 7th International SPIN Workshop on SPIN Model Checking and Software Verification, (2000), Springer), 331-342 · [Zbl 0976.68568](#)
- [26] Havelund, K., Using runtime analysis to guide model checking of Java programs, (Havelund, K.; Penix, J.; Visser, W., Proceedings of the 7th International SPIN Workshop on SPIN Model Checking and Software Verification, (2000), Springer), 245-264 · [Zbl 0976.68576](#)
- [27] Hindley, J. R., The principal type-scheme of an object in combinatory logic, *Trans. Am. Math. Soc.*, 146, 29-60, (1969) · [Zbl 0196.01501](#)
- [28] Igarashi, A.; Pierce, B. C.; Wadler, P., Featherweight Java: a minimal core calculus for Java and GJ, (Proceedings of 14th ACM SIGPLAN Conference on Object Oriented Programming: Systems, Languages, and Applications (OOPSLA), (1999), ACM), 132-146
- [29] Joshi, P.; Naik, M.; Sen, K.; Gay, D., An effective dynamic analysis for detecting generalized deadlocks, (Proceedings of the 18th ACM SIGSOFT International Symposium on Foundations of Software Engineering (FSE), (2010), ACM), 327-336
- [30] Kidd, N.; Reps, T. W.; Dolby, J.; Vaziri, M., Finding concurrency-related bugs using random isolation, *Softw. Tools Technol. Transf.*, 13, 6, 495-518, (2011)
- [31] Kobayashi, N., A partially deadlock-free typed process calculus, *ACM Trans. Program. Lang. Syst.*, 20, 2, 436-482, (1998)
- [32] Kobayashi, N., Type-based information flow analysis for the  $\pi$ -calculus, *Acta Inform.*, 42, 4-5, 291-347, (2005) · [Zbl 1081.68061](#)
- [33] Kobayashi, N., A new type system for deadlock-free processes, (Baier, C.; Hermanns, H., Proceedings of the 17th International Conference on Concurrency Theory (CONCUR), Lecture Notes in Computer Science, vol. 4137, (2006), Springer), 233-247 · [Zbl 1151.68537](#)
- [34] Leino, K. R.M.; Müller, P., A basis for verifying multi-threaded programs, (Castagna, G., Proceedings of the 18th European Symposium on Programming Languages and Systems (ESOP), Lecture Notes in Computer Science, vol. 5502, (2009), Springer), 378-393 · [Zbl 1234.68078](#)
- [35] Milner, R., A theory of type polymorphism in programming, *J. Comput. Syst. Sci.*, 17, 3, 348-375, (1978) · [Zbl 0388.68003](#)
- [36] Mossin, C., Flow analysis of typed higher-order programs, (1997), DIKU, University of Copenhagen, Ph.D. thesis, Technical report DIKU-TR-97/1 · [Zbl 0891.68025](#)
- [37] Naik, M.; Park, C.-S.; Sen, K.; Gay, D., Effective static deadlock detection, (Proceedings of the 31st International Conference on Software Engineering (ICSE), (2009), IEEE), 386-396
- [38] Nielson, H. R.; Nielson, F., Type and effect systems, (Olderog, E.-R.; Steffen, B., Correct System Design - Recent Insights and Advances, Lecture Notes in Computer Science, vol. 1710, (1999), Springer), 114-136
- [39] Nielson, H. R.; Nielson, F.; Amtoft, T., Polymorphic subtyping for effect analysis: the static semantics, (Dam, M., Proceedings of the 5th LOMAPS Workshop, Lecture Notes in Computer Science, vol. 1192, (1997), Springer), 141-171
- [40] Pratikakis, P., Sound, precise, and efficient static races detection for multi-threaded programs, (2008), University of Maryland,

Ph.D. thesis

- [41] Pratikakis, P.; Foster, J. S.; Hicks, M. W., LOCKSMITH: context-sensitive correlation analysis for race detection, (Proceedings of the 27th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI), (2006), ACM), 320-331
- [42] Pratikakis, P.; Foster, J. S.; Hicks, M. W., LOCKSMITH: practical static race detection for C, *ACM Trans. Program. Lang. Syst.*, 33, 1, 3:1-3:55, (2011)
- [43] Pun, K. I.; Steffen, M.; Stolz, V., Deadlock checking by a behavioral effect system for lock handling, *J. Log. Algebraic Program.*, 81, 3, 331-354, (2012) · [Zbl 1246.68097](#)
- [44] Rehof, J.; Fähndrich, M., Type-based flow analysis: from polymorphic subtyping to CFL-reachability, (Proceedings of the 28th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL), (2001), ACM), 54-66 · [Zbl 1323.68226](#)
- [45] Reps, T.; Horwitz, S.; Sagiv, M., Precise interprocedural data-flow analysis via graph reachability, (Proceedings of the 22nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL), (1995), ACM), 49-61
- [46] Rose, J.; Swamy, N.; Hicks, M., Dynamic inference of polymorphic lock types, *Sci. Comput. Program.*, 58, 3, 366-383, (2005) · [Zbl 1105.68014](#)
- [47] Savage, S.; Burrows, M.; Nelson, G.; Sobalvarro, P.; Anderson, T., Eraser: a dynamic data race detector for multithreaded programs, *ACM Trans. Comput. Syst.*, 15, 4, 391-411, (1997)
- [48] Sterling, N., Warlock: a static race analysis tool, (USENIX Winter Technical Conference, (1993), USENIX Association), 97-106
- [49] Suenaga, K., Type-based deadlock-freedom verification for non-block-structured lock primitives and mutable references, (Ramalingam, G., Proceedings of the 6th Asian Symposium Programming Languages and Systems (APLAS), Lecture Notes in Computer Science, vol. 5356, (2008), Springer), 155-170
- [50] Talpin, J.-P.; Jouvelot, P., Polymorphic type, region and effect inference, *J. Funct. Program.*, 2, 3, 245-271, (1992) · [Zbl 0817.68099](#)
- [51] Vasconcelos, V.; Martins, F.; Cogumbreiro, T., Type inference for deadlock detection in a multithreaded polymorphic typed assembly language, (Beresford, A. R.; Gay, S. J., Proceedings of the 2nd International Workshop on Programming Language Approaches to Concurrency and Communication-cEntric Software (PLACES), Electronic Proceedings in Theoretical Computer Science (EPTCS), vol. 17, (2009)), 95-109
- [52] Vojdani, V.; Vene, V., Goblint: path-sensitive data race analysis, (Proceedings of the 10th Symposium on Programming Languages and Software Tools, (2007), Eötvös Lorand Univ.), 130-141 · [Zbl 1199.68107](#)

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.