not be interpreted as necessarily representing the official views, policies or endorsements, either expressed or implied, of the National Science foundation (NSF), any other agency of the U.S. Government, or any of the companies mentioned above.

References

[1] LuaJIT roadmap 2008 - http://lua-users.org/lists/lua-l/2008-02/msg00051.html.

[2] Mozilla — Firefox web browser and Thunderbird email client - http://www.mozilla.com.

[3] SPECJVM98 - http://www.spec.org/jvm98/.

[4] SpiderMonkey (JavaScript-C) Engine - http://www.mozilla.org/js/spidermonkey/.

[5] Surfin' Safari - Blog Archive - Announcing SquirrelFish Extreme - http://webkit.org/blog/214/introducing-squirrelfish-extreme/.

[6] A. Aho, R. Sethi, J. Ullman, and M. Lam. Compilers: Principles, techniques, and tools, 2006.

[7] V. Bala, E. Duesterwald, and S. Banerjia. Dynamo: A transparent dynamic optimization system. In *Proceedings of the ACM SIGPLAN Conference on Programming Language Design and Implementation*, pages 1–12. ACM Press, 2000.

[8] M. Berndl, B. Vitale, M. Zaleski, and A. Brown. Context Threading: a Flexible and Efficient Dispatch Technique for Virtual Machine Interpreters. In *Code Generation and Optimization, 2005. CGO 2005. International Symposium on*, pages 15–26, 2005.

[9] C. Chambers and D. Ungar. Customization: Optimizing Compiler Technology for SELF, a Dynamically-Typed O bject-Oriented Programming Language. In *Proceedings of the ACM SIGPLAN 1989 Conference on Programming Language Design and Implementation*, pages 146–160. ACM New York, NY, USA, 1989.

[10] A. Gal. *Efficient Bytecode Verification and Compilation in a Virtual Machine Dissertation*. PhD thesis, University Of California, Irvine, 2006.

[11] A. Gal, C. W. Probst, and M. Franz. HotpathVM: An effective JIT compiler for resource-constrained devices. In *Proceedings of the International Conference on Virtual Execution Environments*, pages 144–153. ACM Press, 2006.

[12] C. Garrett, J. Dean, D. Grove, and C. Chambers. Measurement and Application of Dynamic Receiver Class Distributions. 1994.

[13] J. Ha, M. R. Haghighat, S. Cong, and K. S. McKinley. A concurrent trace-based just-in-time compiler for javascript. Dept.of Computer Sciences, The University of Texas at Austin, TR-09-06, 2009.

[14] B. McCloskey. Personal communication.

[15] I. Piumarta and F. Riccardi. Optimizing direct threaded code by selective inlining. In *Proceedings of the ACM SIGPLAN 1998 conference on Programming language design and implementation*, pages 291–300. ACM New York, NY, USA, 1998.

[16] A. Rigo. Representation-Based Just-In-time Specialization and the Psyco Prototype for Python. In *PEPM*, 2004.

[17] M. Salib. Starkiller: A Static Type Inferencer and Compiler for Python. In *Master's Thesis*, 2004.

[18] T. Suganuma, T. Yasue, and T. Nakatani. A Region-Based Compilation Technique for Dynamic Compilers. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 28(1):134–174, 2006.

[19] M. Zaleski, A. D. Brown, and K. Stoodley. YETI: A graduallY Extensible Trace Interpreter. In *Proceedings of the International Conference on Virtual Execution Environments*, pages 83–93. ACM Press, 2007.