Analysis of Modular Arithmetic *

Markus Müller-Olm
Universität Dortmund FB 4, LS V
44221 Dortmund, Germany
mmo@ls5.cs.uni-dortmund.de

September 12, 2005

In my talk at the IFIP WG 2.2 meeting in Skagen, I mainly reported on sound and complete static analyses in which we consider integer arithmetic modulo a power of 2 as provided by mainstream programming languages like Java or standard implementations of C. A particular new difficulty is that the ring $\mathbb{Z}_m$ of integers modulo $m = 2^w$, $w > 1$, has zero divisors and thus cannot be embedded into a field. Not withstanding that, we have constructed intra- and inter-procedural algorithms for inferring for every program point u, affine relations between program variables valid at u. Our algorithms are not only sound but also *complete* in that they detect *all* valid affine relations in what we call affine programs. Moreover, they run in time linear in the program size and polynomial in the number of program variables and can be implemented by using the same modular integer arithmetic as the target language to be analyzed.

This talk was mainly based on an ESOP 2005 paper [1]. It is also related to our work in [2, 3, 4, 5, 6].

References

[1] M. Müller-Olm and H. Seidl. Analysis of Modular Arithmetic. In *14th Symposium on Programming, ESOP 2005*, LNCS 3444, pages 31-45. Springer, 2005.

[2] M. Müller-Olm and H. Seidl. Precise Interprocedural Analysis through Linear Algebra. In *31st ACM Symp. on Principles of Programming Languages (POPL)*, pages 330–341, 2004.

[3] M. Müller-Olm and H. Seidl. A Generic Framework for Interprocedural Analysis of Numerical Properties. In C. Hankin and I. Siveroni, editors, *SAS 2005 (Static Analysis Symposium)*, volume 3672 of *Lecture Notes in Computer Science*, pages 235–250. Springer, 2005.

*Joint work with Helmut Seidl, Institut für Informatik, I2, Technische Universität München, Boltzmannstr. 3, 85748 Garching, Germany, seidl@in.tum.de

[4] M. Müller-Olm and H. Seidl. A Note on Karr's Algorithm. In *31st Int. Coll. on Automata, Languages and Programming (ICALP)*, pages 1016–1028. Springer Verlag, LNCS 3142, 2004.

[5] M. Müller-Olm and H. Seidl. Polynomial constants are decidable. In M. Hermenegildo and G. Puebla, editors, *SAS 2002 (Static Analysis Symposium)*, volume 2477 of *Lecture Notes in Computer Science*, pages 4–19. Springer, 2002.

[6] M. Müller-Olm and H. Seidl. Computing Polynomial Program Invariants. *Information Processing Letters (IPL)*, 91(5):233–244, 2004.