Collection of Combinational Arithmetic Miters Submitted to the SAT Competition 2016

Armin Biere
Institute for Formal Models and Verification
Johannes Kepler University Linz

***Abstract*—In this short note we present a collection of benchmarks submitted to the SAT Competition 2016. Most of them stem from other sources, some crafted ones are new, but all present equivalence checking problems (miters) for arithmetic circuits, such as multipliers.**

INTRODUCTION

Two invited talks by Anna Slobodova and Aaron Tomb, as well as a tutorial by Priyank Kalla in Austin as part of SAT'16 and FMCAD'15 argued, that checking arithmetic miters is still a challenge, both in hardware and software verification, even after more than 20 years after the Pentium FDIV bug. As a consequence even today, verifying arithmetic circuits requires cumbersome manual case splitting or simply gives up on obtaining a formal proof and uses simulation instead.

The reason is that these circuits do not have internal equivalence points, i.e., in essence only the outputs are pair-wise equivalent. It is further conjectured that resolution is not strong enough to obtain polynomial proofs even for such simple tasks as checking commutativity of bit-vector multiplication after bit-blasting and CNF encoding.

In order to help trying to attack this challenge we collected existing arithmetic miters and also generated some new crafted benchmarks. All the submitted benchmarks are published at http://fmv.jku.at/datapath. The README files available there give more information on how exactly the benchmarks were derived. The benchmark archive also contains structural versions for some of the benchmarks in various formats beside CNF in DIMACS format.

CRAFTED MITERS

The CRAFTED benchmark set contains the old subset LINVRINV, which was suggested by Stephen Cook during his invited talk at SAT'04, for which we previously already submitted a C generator to the competition. Pre-generated CNFs up to square matrix size 7 are included, which are still considered to be really challenging. The structure of the propositional arithmetic in this benchmark subset has some flavor or multiplier miters, but might need even more powerful reasoning. The remaining benchmark subsets in the CRAFTED set, check simple properties of bit-vector multiplication for various bit-widths, more precisely, commutativity $x \cdot y = y \cdot x$, associativity $x \cdot (y \cdot z) = (x \cdot y) \cdot z$, and distributivity $x \cdot (y + z) = x \cdot y + x \cdot z$, as well as the property $x \cdot (x + 1) = x \cdot x + x$.

Supported by FWF, NFN Grant S11408-N23 (RiSE).

We consider these problems as crafted, since bit-vector rewriting can prove them trivially. However, disabling rewriting and bit-blasting them to AIGs with Boolector [1], then encoding them into CNF, produces pretty challenging benchmarks too. We included SMT, AIG and of course CNF versions of these benchmarks up to the bit-width, for which we do not know of any known technique which can solve the CNF versions in a reasonable amount of time (16 bits for commutativity and associativity, 12 bits for distributivity and 24 bits for the last property).

Note however, that these benchmarks, as well as probably most of the benchmarks in this submission, have nice linear parallel speed-ups using cube-and-conquer solving [2]. So we expect Treengeling [3] to be able to go a few bits further than other solvers, particularly sequential ones, depending on the number of processor cores.

EPFL MITERS

The benchmark set EPFL was generated by Mathias Soeken using ABC [4]. These 10 miters check correctness of the smallest optimized variant of circuits in the "The EPFL Combinational Benchmark Suite" [5]. Only arithmetic circuits were used for generating miters in this submission. A few benchmarks are considered trivial, most of them challenging. This original set of optimized circuits is still evolving and might be good a source for more miter benchmarks.

MULTIPLIER MITERS BY MATTI JÄRVISALO

The benchmark set JARVISALO was submitted to the SAT Competition 2007 before by Matti Järvisalo [6] and has been used in the competition for quite some time (file name prefix "eqatreebraun"). It consists of miters for checking equivalence of one particular optimized multiplier architecture against a reference multiplier. We only include these because they model the same problem as other benchmarks in this submission.

I. FMCAD 2015 EXAMPLE BY PRIYANK KALLA

Syntactically different polynomials modulo 2^n might still represent the same function. One of them might have less coefficient bits. This in turn might yield a more compact circuit implementation. Checking equivalence of the original circuit versus the optimized implementation again produces a miter. The single benchmark we have in this benchmark set KALLAFMCAD15 is from an example given by Priyank Kalla in his Tutorial at FMCAD'16 [7] on implementing

$F = 1/2\sqrt{a^2 + b^2}$ by the polynomial of its Taylor expansion on $x = a^2 + b^2$, where x is a bit-vector of size 16.

This setting might yield more interesting benchmarks and the same applies to similar problems in the context of verifying arithmetic circuits for signal processing, such as considering Galois field multipliers.

II. Multiplier Miters from Arist Kojevnikov

There exists a generator suite to produce an actually quite large set of multiplier miters, which was published already in 2005 by Arist Kojevnikov. This was used for developing and benchmarking a boolean algebraic solver [8]. These generator scripts produce ISCAS miters, which we translated to AIGs and then to CNF. We generated benchmarks for bit-widths 4,8,9-16,32,64,128, and generated 144 miters per bit-width. We also included buggy multipliers, which yield satisfiable miters all having "bg" in their file name. Some of the miters compare structurally very similar (or even identical) circuits. Those are then much simpler. The other correct miters with high bit-widths are a real challenge.

III. Miters from Kaiserslautern

The first benchmark set WEDLER from the group of Wolfgang Kunz in Kaiserslautern, is based on miters in SMT format as used in an ASPDAC'08 paper on bit-level arithmetic circuit verification [9]. We obtained the actual SMT files from Markus Wedler. These only include miters for their own generated multipliers and not the industrial IBM multipliers, which were used in addition in that paper. These generated benchmarks have further been used already in many papers on arithmetic circuit verification. The 108 SMT files have different combinations of operand size and output sizes, and also differ w.r.t. signedness and whether Booth encoding was used. The SMT files were bit-blasted again with Boolector [1] to obtain CNF files.

The other benchmark set WIELAND from Kaiserslautern is related to their CAV'08 paper [10], which uses algebraic word-level techniques. These benchmarks were submitted to the SMT-LIB [11], and we simply bit-blasted them with Boolector [1]. This set consists of three generic miters over the bit-widths 4,8,16,32,48,64, thus 18 benchmarks altogether.

IV. Conclusion

We consider the effort of collecting a meaningful set of arithmetic problems encoded in CNF as not finished yet. For instance, we tried to obtain some more multiplier miters used in a recent DATE'16 paper [12], but the original multiplier designs are not publicly available. Furthermore, some of the sources of benchmarks used above might yield more benchmarks. Then there are these challenges mentioned in his invited talk by Aaron Tomb last year, and already discussed above, in the context of verifying correctness of the implementation of cryptographic functions. Some of them are already available as SMT-LIB [11] benchmarks. Finally, benchmarks in the context of verifying floating point operations might be interesting too.

References

[1] A. Niemetz, M. Preiner, and A. Biere, "Boolector 2.0," *JSAT*, vol. 9, pp. 53–58, 2015.

[2] M. Heule, O. Kullmann, S. Wieringa, and A. Biere, "Cube and conquer: Guiding CDCL SAT solvers by lookaheads," in *Hardware and Software: Verification and Testing - 7th International Haifa Verification Conference, HVC 2011, Haifa, Israel, December 6-8, 2011, Revised Selected Papers*, ser. Lecture Notes in Computer Science, K. Eder, J. Lourenço, and O. Shehory, Eds., vol. 7261. Springer, 2011, pp. 50–65.

[3] A. Biere, "Lingeling and friends entering the SAT Race 2015," Johannes Kepler University, Linz, Austria, FMV Report Series Technical Report 15/2, April 2015.

[4] R. K. Brayton and A. Mishchenko, "ABC: an academic industrial-strength verification tool," in *Computer Aided Verification, 22nd International Conference, CAV 2010, Edinburgh, UK, July 15-19, 2010. Proceedings*, ser. Lecture Notes in Computer Science, T. Touili, B. Cook, and P. Jackson, Eds., vol. 6174. Springer, 2010, pp. 24–40.

[5] L. Amarú, P.-E. Gaillardon, and G. De Micheli, "The epfl combinational benchmark suite," in *Proceedings of the 24th International Workshop on Logic & Synthesis (IWLS)*, no. EPFL-CONF-207551, 2015.

[6] M. Järvisalo, "Equivalence checking hardware multiplier designs," 2007, sAT Competition 2007 benchmark description. Available at http://www.satcompetition.org/2007/contestants.html.

[7] P. Kalla, "Formal verification of arithmetic datapaths using algebraic geometry and symbolic computation," in *Formal Methods in Computer-Aided Design, FMCAD 2015, Austin, Texas, USA, September 27-30, 2015.*, R. Kaivola and T. Wahl, Eds. IEEE, 2015, p. 2.

[8] E. Hirsch, D. Itsykson, A. Kojevnikov, A. Kulikov, and S. Nikolenko, "Report on the mixed boolean-algebraic solver," Citeseer, Tech. Rep., 2005.

[9] U. Krautz, M. Wedler, W. Kunz, K. Weber, C. Jacobi, and M. Pflanz, "Verifying full-custom multipliers by boolean equivalence checking and an arithmetic bit level proof," in *Proceedings of the 13th Asia South Pacific Design Automation Conference, ASP-DAC 2008, Seoul, Korea, January 21-24, 2008*, C. Kyung, K. Choi, and S. Ha, Eds. IEEE, 2008, pp. 398–403.

[10] O. Wienand, M. Wedler, D. Stoffel, W. Kunz, and G. Greuel, "An algebraic approach for proving data correctness in arithmetic data paths," in *Computer Aided Verification, 20th International Conference, CAV 2008, Princeton, NJ, USA, July 7-14, 2008, Proceedings*, ser. Lecture Notes in Computer Science, A. Gupta and S. Malik, Eds., vol. 5123. Springer, 2008, pp. 473–486.

[11] C. Barrett, P. Fontaine, and C. Tinelli, "The SMT-LIB Standard: Version 2.5," Department of Computer Science, The University of Iowa, Tech. Rep., 2015, available at `www.SMT-LIB.org`.

[12] A. A. R. Sayed-Ahmed, D. Große, U. Kühne, M. Soeken, and R. Drechsler, "Formal verification of integer multipliers by combining gröbner basis with logic reduction," in *2016 Design, Automation & Test in Europe Conference & Exhibition, DATE 2016, Dresden, Germany, March 14-18, 2016*, L. Fanucci and J. Teich, Eds. IEEE, 2016, pp. 1048–1053.