

Verification of hardware circuits still remains a challenge to our community. Binary Decision Diagrams (BDDs) and SAT solvers based verification cannot scale-up with the current circuit sizes. In recent years, computer algebra and Gröbner basis based verification have been proposed and improved upon. The circuit is modeled as an ideal of polynomials and its Gröbner basis is computed using the Buchberger’s algorithm. The expensive cost of Buchberger’s algorithm can be avoided by extracting a term order from the circuit’s topology. Verification approaches based on this ordering have been shown to perform well on large Galois field arithmetic circuits [?] [?]. However, using similar approaches for integer arithmetic circuits, results in a term explosion problem. Several heuristics based on the circuit topology and hierarchy have been suggested to avoid this blow-up.

The authors in [?] perform fan-out rewriting to increase the chances of term cancellation before their blow-up. The work in [?] uses a number of heuristics, including fan-outs and levelization, to improve the reduction process. The term explosion in certain multiplier architectures is attributed to vanishing monomials that eventually reduce to zero but result in an intermediate blow-up [?].

For the approaches discussed above, the polynomials are implemented explicitly using symbolic data structures. The sparsity inherent in these polynomials cannot be exploited using these data structures. Zero-Suppressed Binary Decision Diagrams (ZBDDs) [?] find numerous applications in fields requiring set manipulation. Boolean polynomials can be considered as sets of their monomials which can be represented using ZBDDs [?]. As a result, ZBDDs, being an implicit data structure, are better for representing these polynomials as characteristic functions. The approach presented in [?] depends on the structural hierarchy of the circuit. If this information is unavailable or if the netlist is bit-blasted, the heuristic of vanishing monomial cannot be applied. A similar but more generalized heuristic is required for these cases.

*Contributions:* We present an implementation of symbolic computer algebra algorithms, specially targeted for circuit verification, using the unate cube set algebra and ZBDDs. The basic unate cube set algebra operations using ZBDDs are discussed in [?]. We show that the process of reduction can be performed by accessing the subgraphs of the ZBDD and using the modulo-2 sum and product algorithms described in [?]. Based on the term orders derived from the circuits, ZBDDs gives further opportunities to speed up Gröbner basis reduction implicitly by cancelling multiple monomials. As ZBDD is a canonical data structure, the result of the reduction for the output bits of the circuit can be used for verification. The heuristics presented in [?] and [?], can further be extracted from arbitrary bit-blasted netlists using our approach. We present algorithms for performing these computations and experiments to support our implementations.

*Paper Organization:* Section II discusses the related previous work in computer algebra based abstraction and equivalence checking for verification of circuits. Section III covers the preliminary concepts related to Binary decision diagrams, boolean polynomials and Gröbner basis reduction. Section IV describes the use of ZBDDs for polynomial abstraction from circuits including the

algorithm for performing the multi-term cancellation. Section V discusses the heuristics required for speeding up the reduction process and their implementation on ZBDDs. Our experiments on galois field and arithmetic circuits are presented in section VI, and finally, section VII concludes the paper.