1. **Background**

In 2016 a new 3SAT solving methodology, described in two subsequent papers availed on the public domain: [1], [2], has been proposed. The methodology used a novel, dual-nature view of logical variables to construct small FBDDs (Free Binary Decision Diagrams), a task which is, since the 1980th, known to be impossible for some important types of Boolean functions. As the preliminary peer review of the papers hasn’t been showing any major flaws, a more thorough theoretical and formal investigation is under way at the time of writing of this document.

The novel approach has been implemented in a proof-of-concept Prototype-Solver capable of handling small 3SAT-instances (i.e. Number of Variables and Clauses are in hundreds only) and a series of tests using benchmarks from SAT-Competitions and other known hard-formula sources were conducted. A detailed technical white paper (see: [3]) describing performed tests, obtained results as well as explanations thereof was also finalized.

1. **Objective**

Objective of this document is to show for selected, practical areas of interest, using results in [3], and in the most conscience way, evidence supporting claims of the papers.

1. **State of the Art**

Building small FBDDs for all Boolean functions is in direct contradiction with established exponential lower bounds, known since the last 1980th of last century. Prominent results in this respect are those obtained for the multiplication function, notably in [4], and the “blocking-sets-in-finite-projective-planes” function found in [5]. While the latter is merely of theoretical importance, the former is directly related to the crucial one-way-function assumption, since FBDDs may be used in two directions.

On the other hand : Current SAT-Solving technology is focused on DPLL and CDCL methodologies, which don’t construct full Truth Table-equivalent structures and are thus not comparable to any methods attempting to solve SAT/3SAT using BDDs. There are, nevertheless some scarce SAT-Solving attempts which bind Solvers to BDDs (see [6], for example).

Some hardness-phenomena are, nevertheless, known to exist for both DPLL/CDCL and BDD-oriented Solvers. Such a phenomenon is, for example, the easy-hard-easy threshold of DPLL/CDCL methods which is widely believed to occur at Density=4.26 of random 3SAT formulas (the Density of a CNF-Clause Set is the ratio: Number of Clauses M, over Number of Variables N). The same threshold for BDD-based algorithms is expected at densities around 2.0. For such algorithms : Fixing densities in the range between 0.5 and 15, while increasing N has been shown to yield exponential results in both time and space (see [10], page 5)

1. **Selected Areas of investigation and Algorithms used**

As per the previous section and per current technologies:

1. Small FBDDs are **not** expected to be constructed for:
   1. The “blocking-sets-in-finite-projective-planes” function
   2. The multiplication/factorization function. Current results for this function extend also to OBDDs (where only one variable ordering is admitted in the graph)
2. **Exponential behavior** is expected to occur for random 3SAT formulas when densities between 0.5 and 15 are fixed while N is increased.

We have, therefore, chosen three areas as primary areas of investigation on which the new algorithms are applied:

1. Blocking-sets-in-finite-projective-planes: CNF-Source [7]
2. Multiplication/Factorization: CNF-Source [8]
3. Random 3SAT Formulas: CNF-Source [9]

The new algorithms come in this preliminary validation work in two flavors: A version in which the l.o. condition is applied to all the nodes of the constructed graphs, constructing thus an FBDD and a version in which the same or a more relaxed condition (called in [1]: l.o.u) is only applied to the base CNF Clause Set/Formula resulting in an OBDD.

1. **Preliminary Results Summary, explanations and ways of verification**
   1. **Blocking-sets-in-finite-projective-planes (FBDD):** 
      1. This problem area was taken as a challenge in the first as well as the second paper ([1], Section III, page 185) where CNF representations and FBDDs for q=2, q=3 were depicted (see also : Appendix A,B in [1])
      2. **Result:** As per [5], any FBDD constructed for q must possess a complete binary tree for the first root(q) levels. This is not the case for the 3SAT-Solver pattern algorithms shown in [1], since they are guaranteed to reach False nodes in only c=3 steps.
      3. **Explanation:** If f is the function representing the Blocking-Sets-in-projective-planes problem, then: The theoretical investigation in [5] did not take into consideration equi-satisfiable translations f’ of f, which are necessary to formalize the problem for a 3SAT-solver. Such translations introduce many auxiliary variables which become part of f’ and destroy hence the symmetric sub-function property of f which lead originally to the lower bound.
      4. **Verification Method :** Manual (see [1], section III as well as lower bound proofs in [5])
   2. **Multiplication/Factorization (OBDD, l.o.u.):**
      1. **Result :** The CNF-source in [8] guarantees that the length of a 3CNF representation of a multiplication/factorization problem does not exceed O(input-bits3) as seen in the figure of point i. (this has also been verified through email-communication with the author of the CNF generation algorithms). The figure in section ii. suggests that the number of unique nodes for an OBDD generated using pattern Algorithms of [1] (l.o.u. in base set version) is also in O(M2). This makes the number of unique nodes in such an OBDD: O(input-bits6), i.e. polynomial in the input-bits, not exponential as expected.
      2. **Explanation :** The same reason like the one seen in point a.iii. above : Known lower bound for the multiplication (like the ones in [4]) were obtained assuming the use of original kCNF representations of the problem and not equi-satisfiable ones.
      3. **Verification Method :** Program runs repeatable using the l.o.u. version of the pattern algorithms in [1], CNF-formula length verifiable for [8].
   3. **Random 3SAT (FBDD, OBDD both l.o. and l.o.u.):**
      1. **Result:** Fixing densities while increasing the number of variables is supposed for Random 3SAT to produce exponential results. This is clearly not the case for produced FBDDs. Even OBDD construction is not following the expected exponential pattern : The OBDD (l.o. version) shows even a drastic decrease indicating the possibility of a threshold in this region (to be confirmed in future work).
      2. **Explanation:** Random 3SAT formulas are a good measure of the behavior of any SAT-Solving algorithm, because they do not assume any structure in processed CNF-formulas. Results here show runs which never exceeded the O(M3) bound, which is even less than what the O(M4) predicted in theory.
      3. **Verification Method:** All program runs are repeatable
2. **Discussion and conclusions**

The pattern algorithms under investigation have bypassed lower bounds known for the two selected areas: Blocking-sets-in-finite-projective-planes and Multiplication/Factorization as seen in the last section. This can mainly be attributed to the use of a 3SAT-solving methodology which necessitates equi-satisfibale transformations of functions. However: The use of such functions alone is not sufficient to achieve the same results. A CNF-resolution strategy which selects a near-to-optimal ordering as the one described in [1] is also necessary. Using such a strategy for hard random 3SAT formulas is found to be effective strengthening the impression that processing 3SAT-instances in such a way may be polynomial in nature. Finally: All runs of the pattern-algorithms under investigation did not exceed the O(M4) bound predicted in theory (see: [1]).

1. **Tables**
   1. **Test cases for Figure in Section V.b.i:**

|  |  |  |
| --- | --- | --- |
| **Input-bits** | **Actual Clauses** | **O(bits3)** |
| 5 | 73 | 125 |
| 7 | 187 | 343 |
| 8 | 248 | 512 |
| 10 | 434 | 1000 |
| 11 | 519 | 1331 |
| 13 | 777 | 2197 |
| 14 | 886 | 2744 |
| 14 | 886 | 2744 |
| 14 | 886 | 2744 |
| 16 | 1216 | 4096 |

1. **Test cases for Figure in Section V.b.ii:**

|  |  |  |
| --- | --- | --- |
| **Clauses (M)** | **Actual Nodes**  **OBDD (l.o.u.)** | **O(M2)** |
| 73 | 126 | 5329 |
| 187 | 359 | 34969 |
| 248 | 2215 | 61504 |
| 434 | 11469 | 188356 |
| 519 | 7108 | 269361 |
| 777 | 44666 | 603729 |
| 886 | 50863 | 784996 |
| 886 | 50443 | 784996 |
| 886 | 50861 | 784996 |
| 1216 | 142703 | 1478656 |

1. **Test cases for Figure in Section V.c.i (fixed density 4.26) :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N,M / UNodes** | **Normal (CNF clauses are not pre-processed)** | **OBDD (l.o.u.)** | **(OBDD) l.o.** | **FBDD** |
| 10,43 | 75 | 64 | 58 | 48 |
| 15,64 | 1170 | 451 | 269 | 174 |
| 20,85 | 4980 | 2024 | 1401 | 378 |
| 25,107 | 6946 | 8683 | 4198 | 582 |
| 30,128 | 52434 | 22370 | 17879 | 866 |
| 35,149 | 240655 | 107764 | 82855 | 2396 |
| 40,170 | 324921 | 206663 | 468067 | 8289 |
| 45,192 | 6074868 | 1057517 | 4295253 | 18805 |
| 50,213 | 12054732 | 7797056 | 10466313 | 49412 |
| 55,234 | overflow | 8830861 | 4628678 | 58611 |
| 60,256 | overflow | overflow | overflow | 138319 |
| 65,277 | overflow | overflow | overflow | 240888 |
| 70,298 | overflow | overflow | overflow | 568690 |
| 75,320 | overflow | overflow | overflow | 1922224 |

* Cyan indicates UNSAT

|  |  |
| --- | --- |
| **N,M** | **FBDD** |
| 10,43 | 48 |
| 15,64 | 174 |
| 20,85 | 378 |
| 25,107 | 582 |
| 30,128 | 866 |
| 35,149 | 2396 |
| 40,170 | 8289 |
| 45,192 | 18805 |
| 50,213 | 49412 |
| 55,234 | 58611 |
| 60,256 | 138319 |
| 65,277 | 240888 |
| 70,298 | 568690 |
| 75,320 | 1922224 |

1. **Test cases for Figure in Section V.c.ii:**
2. **Literature**

**[1]** [Abdelwahab 2016-2]: N. Abdelwahab, *Constructive Patterns of Logical Truth*, J. Acad. (N.Y.) 2016, Vol. 6, 2:99.

**[2]** [Abdelwahab 2016-1]:N. Abdelwahab, *On the dual Nature of logical Variables and Clause Sets*, J. Acad. (N.Y.) 2016, Vol. 6, 3:202-239.

**[3]** Preliminary Validation Document, White paper, EasyXPS, 1/2017

**[4]** [Bryant 1986]: Randal Bryant, *Graph-Based Algorithms for Boolean Function Manipulation*, IEEE Transactions on Computers, C-35-8, 677-691, August, 1986

**[5]** [Gal 1997]: Anna Gal, *A simple function read-once that requires exponential size branching programs*, Information Processing Letters 62 (1997), 13-16.

**[6]** Wille, Robert & Drechsler, Rolf. (2007). Building Free Binary Decision Diagrams Using SAT Solvers. Facta universitatis - series: Electronics and Energetics. 20. 10.2298/FUEE0703381W.

**[7]** [**http://ericmoorhouse.org/pub/planes/**](http://ericmoorhouse.org/pub/planes/)

**[8]** [**http://cgi.cs.indiana.edu/~sabry/cnf.html**](http://cgi.cs.indiana.edu/~sabry/cnf.html)

**[9]** [**https://toughsat.appspot.com/**](https://toughsat.appspot.com/)

**[10]** <https://www.ics.uci.edu/~dechter/courses/ics-275a/fall-2001/project/vardi.pdf>