Solving the Binary Puzzle

P. Utomo[1], R. Makarim[2]

[1] *Eindhoven University of Technology, Netherlands, p.h.utomo@tue.nl*
[2] *Mathematical Institute, University Leiden and Centrum Wiskunde en Informatica, Netherlands makarim@cwi.nl*

Binary puzzle is a sudoku-like puzzle with values in each cell taken from the set $\{0,1\}$. We look at the mathematical theory behind it. A solved binary puzzle is an $n \times n$ (n even) binary array that satisfies the following conditions:

1. No three consecutive ones and also no three consecutive zeros in each row and each column,
2. Every row and column is balanced, that is the number of ones and zeros must be equal in each row and in each column,
3. Every two rows and every two columns must be distinct.

Figure 1 is an example of initial setting of a binary puzzle. There is only one solution satisfying all three conditions which can be seen in Figure 2. Binary puzzles can be seen as constrained arrays [3]. One can also see this array from an erasure correcting point of view [2].

This paper focuses on solving binary puzzles. Solving binary puzzle is proven to be an NP-complete problem[1]. We devise and compare three approaches for finding its solution. The first solves straightforwardly by means of exhaustive search. The second idea, transforms the problem into a SAT problem, then we solve using a SAT solver. The third approach construct a set of polynomial equations over $\mathbb{F}_2$ representing the three conditions for a solved binary puzzle. The variables in the system of equations correspond to all cells in the puzzle. Hence

	0						
			1		1		0
		0					
	1						
				1			
	0				1		
	0			0			
				0		0	

Figure 1: Unsolved Puzzle

1	0	1	0	1	0	1	0
0	1	0	1	0	1	1	0
1	0	0	1	1	0	0	1
0	1	1	0	0	1	1	0
0	1	0	1	1	0	0	1
1	0	1	0	1	1	0	0
1	0	0	1	0	0	1	1
0	1	1	0	0	1	0	1

Figure 2: Solved Puzzle

the solution for the equation system is a solution for the puzzle and it can be obtained by computing its Gröbner basis.

From a complexity point of view, solving the puzzle straightforwardly is more efficient in terms of execution time. The comparison between the three methods in solving the puzzle of various sizes is given in the Table 1. All the computation is done in SageMath 7.0.

Table 1: Comparison of execution time (in seconds) for each method.

Size	SAT		Goebner basis		Exhaustive search
	Pre-comp.	Solver	Pre-comp.	Solver	
4×4	0.01	0.05	0.02	0.05	0.01
6×6	0.14	0.26	0.11	0.06	0.16
8×8	1.58	2.20	0.53	0.13	0.12
10×10	12.31	16.45	3.30	8.69	0.48
12×12	85.43	107.80	47.80	4.55	3.89
14×14	-	-	-	-	94.32

References

[1] M. De Biasi, Binary puzzle is NP-complete, `http://nearly42.org`

[2] P. Utomo and R. Pellikaan, Binary Puzzles as an Erasure Decoding Problem. *Proc. 36th WIC Symp. on Information Theory in the Benelux*, pp. 129–134 May 2015. `http://www.w-i-c.org/proceedings/proceedings_SITB2015.pdf.`

[3] P. Utomo and R. Pellikaan, On The Rate of Constrained Arrays. *Proc. IndoMS International Conference on Mathematics and Its Applications (IICMA) 2015*, 3 November 2015. `http://www.win.tue.nl/~ruudp/paper/75.pdf`