# Solving SAT in polynomial time?

Brian Tenneson

April 11, 2023

Tenneson

**Abstract**

Our goal is to shed some light on the $\mathcal{P}$ vs $\mathcal{NP}$ problem by looking at a SAT solver and verify that it runs in polynomial time. Given the $\mathcal{NP}$-completeness of SAT, this would imply $\mathcal{P} = \mathcal{NP}$.

$\mathcal{P}$ vs $\mathcal{NP}$, polynomial time SAT solver, Boolean algebra We would like to acknowledge all of the many people who contributed to this work.

# 1 A SAT solver which may run in polynomial time

## 1.1 Input Boolean wff $\varphi$

Either $\varphi = p_i$ for a propositional variable $p_i$ or $\varphi = \neg\varphi_1$ for some Boolean wff $\varphi_1$, or $\varphi = \varphi_1 \wedge \varphi_2$ for Boolean wffs $\varphi_1$ and $\varphi_2$, or $\varphi = \varphi_1 \vee \varphi_2$ for Boolean wffs $\varphi_1$ and $\varphi_2$.

## 1.2 Compute a polynomial associated with $\varphi$ called $\tau(\varphi)$ as follows:

$$\tau(p_i) := x_i$$

$$\tau(\varphi_1 \wedge \varphi_2) := \tau(\varphi_1)\,\tau(\varphi_2)$$

$$\tau(\varphi_1 \vee \varphi_2) := 1 + (1 + \tau(\varphi_1))\,(1 + \tau(\varphi_2))$$

Calculations are done in $\mathbb{F}_2$.

## 1.3 Let $J = \langle \tau + 1, x_1^2 + x_1, ..., x_n^2 + x_n \rangle$ be an ideal in $\mathbb{F}_2[x_1, ..., x_n]$

Note that $J = \langle \tau + 1 \rangle + \langle x_1^2 - x_1, ..., x_n^2 - x_n \rangle$.

## 1.4 Find a Gröbner basis $G$ for $J$ which might be done in polynomial time using the FGLM algorithm

## 1.5 Check whether $\overline{1}^G = 0$

### 1.5.1 If $\overline{1}^G = 0$, then $\varphi$ is not satisfiable

### 1.5.2 If $\overline{1}^G \neq 0$, then $\varphi$ is satisfiable

# References