EXERCISE 32

BOOLEAN SATISFIABILITY REVIEW

Write your name and answer the following on a piece of paper

Apply DPLL to determine if there is a satisfying assignment to the following Boolean formula

$$(a \vee b) \wedge (a \vee c) \wedge (\neg b \vee \neg c) \wedge (\neg d \vee \neg c) \wedge (\neg d \vee \neg b) \wedge (c)$$

EXERCISE 32 SOLUTION

BOOLEAN SATISFIABILITY REVIEW

EXERCISE 32

BOOLEAN SATISFIABILITY REVIEW

Write your name and answer the following on a piece of paper

Apply DPLL to determine if there is a satisfying assignment to the following Boolean formula

$$(a \vee b) \wedge (a \vee c) \wedge (\neg b \vee \neg c) \wedge (\neg d \vee \neg c) \wedge (\neg d \vee \neg b) \wedge (c)$$

EXERCISE 32 SOLUTION

BOOLEAN SATISFIABILITY REVIEW

$$(a \vee b) \wedge (a \vee c) \wedge (\neg b \vee \neg c) \wedge (\neg d \vee \neg c) \wedge (\neg d \vee \neg b) \wedge (c)$$

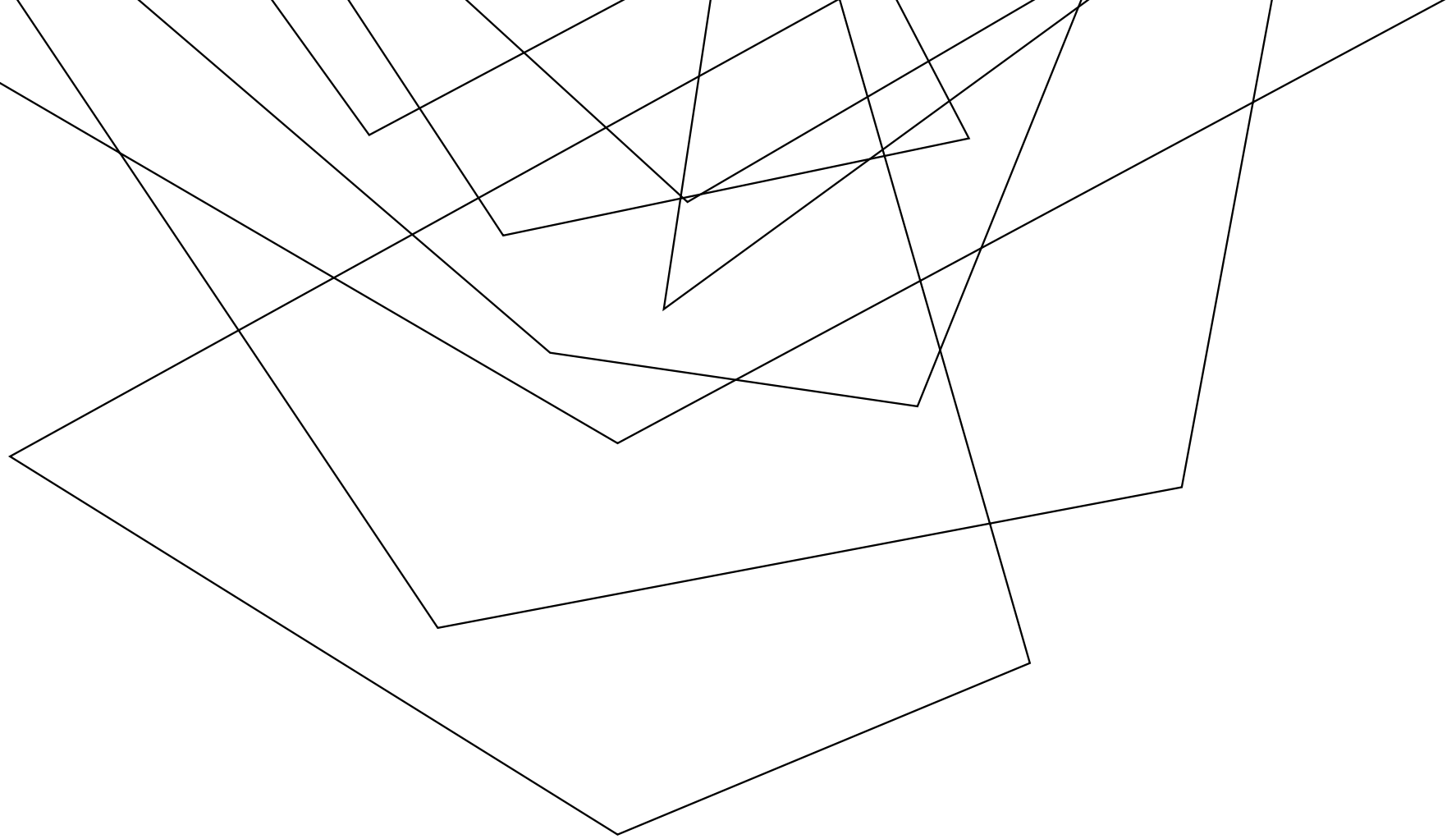
```
function DPLL(φ)
        if φ = true then
                return true
        end if
        if φ contains a false clause then
                return false
        end if
        for all unit clauses l in φ do
                φ ← UNIT-PROPAGATE(l, φ)
        end for
        for all literals l occurring pure in φ do
                φ ← PURE-LITERAL-ASSIGN(l, φ)
        end for
        l ← CHOOSE-LITERAL(φ)
        return DPLL(φ ∧ l) ∨ DPLL(φ ∧ ¬l)
end function
```

ADMINISTRIVIA AND ANNOUNCEMENTS

Quiz 3 on 11/17

Review Session on 11/10 7:00 PM

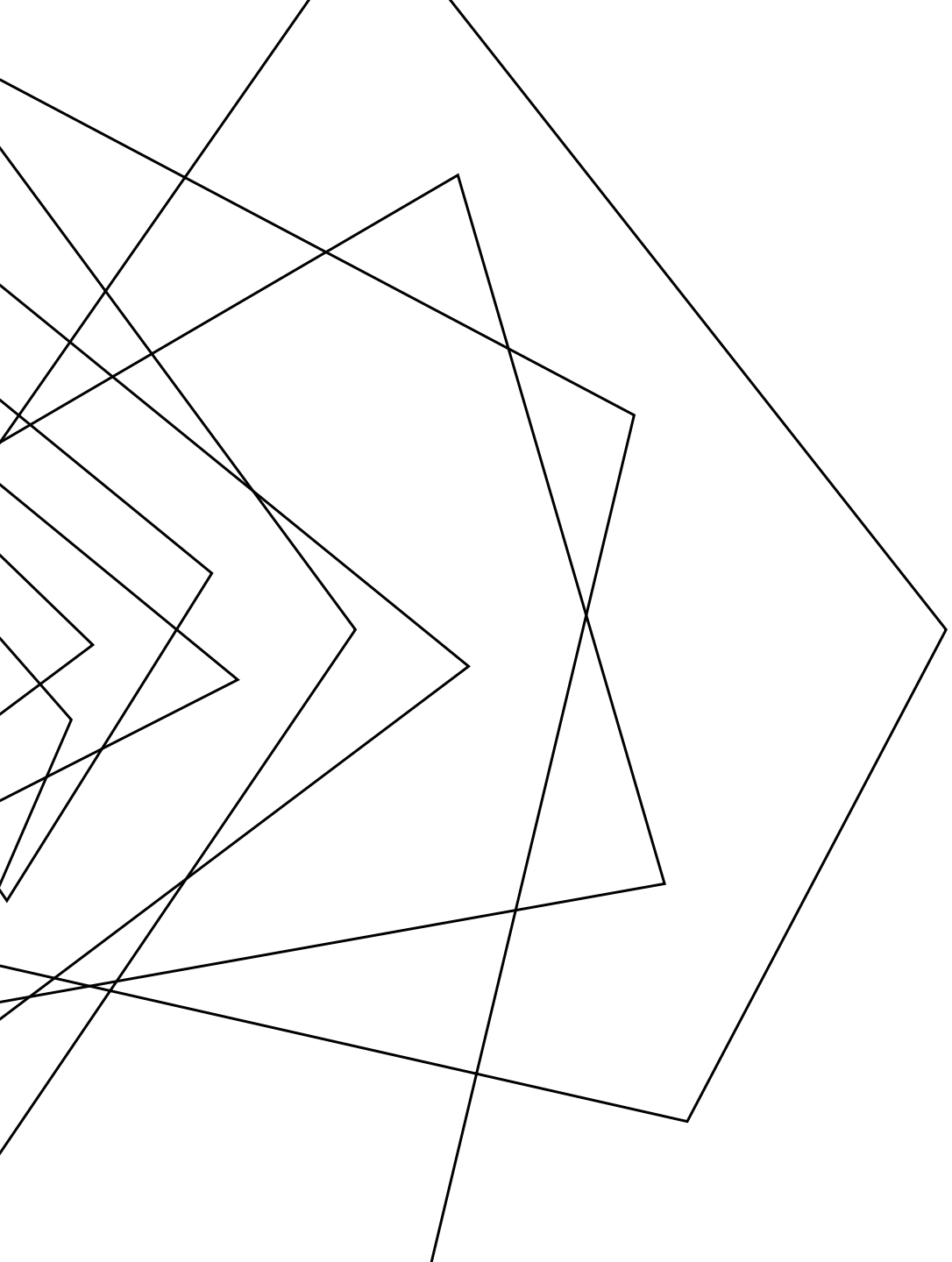
Room TBA



SMT SOLVING

EECS 677: Software Security Evaluation

Drew Davidson



WHERE WE'RE AT

TOOLS / TECHNIQUES UNDERLYING
SYMBOLIC EXECUTION

PREVIOUSLY : SATISFIABILITY

OUTLINE / OVERVIEW

THE MAGIC THAT MADE SYMBOLIC EXECUTION WORK WAS THE SOLVER

A COMPUTATIONALLY HARD PROBLEM

Famously NP-complete (the progenitor of that complexity class!)

Obvious exponential loose upper bound (brute force)

THIS LECTURE

SMT SOLVING

SATISFIABILITY BEYOND SIMPLE BOOLEAN EXPRESSIONS

Gets us (closer) to the real programs that we want to analyze

KEY PRINCIPLES

Individual theory solvers

Formulating constraints modularize a concern to a theory

THEORY SOLVERS

SMT SOLVING

SOME EXAMPLE THEORIES

Theory of linear integer arithmetic

Theory of bitvectors

Theory of arrays

Theory of strings

Theory of equality on uninterpreted (mathematical) functions

Often possible (+ convenient / necessary) to abstract away the actual behavior of a function

THEORY SIGNATURES

SMT SOLVING

The set of (non-logical) symbols and their meanings defined by that theory

Example: Theory of linear integer arithmetic:
$(0,1,+,-,\leq)$ interpreted over $\mathbb{Z}$

Once we have a set of signatures, we'll try to get our formula (i.e. path constraint) to separate concerns into theories

SEPARATING CONCERNS

SMT SOLVING

Note: we will only deal with constraints in Quantifier-Free First-Order Logic

Goal: break down the constraint system to match our core (logical) theory at the top level, with individual clauses potentially in our theory signatures

Logical symbols
– Parentheses: (,)
– Propositional connectives: ∨, ∧, ¬, →, ↔
– Variables: v1, v2, . . .
– ~~Quantifiers: ∀, ∃~~
Non-logical symbols
– Equality: =
– Functions: +, -, %, bit-wise &, f(), concat, ...
– Predicates: is_substring, ...
– Constant symbols: 0, 1.0, null`

EXAMPLE
SMT SOLVING

$$f(f(x) - f(y)) = a$$
$$\wedge$$
$$f(0) = a + 2$$
$$\wedge$$
$$x = y$$

Credit: this example due to Oliveras and Rodriguez-Carbonell, additional work by Aldrich

EXAMPLE
SMT SOLVING

Step 1: Nelson-Oppen procedure to separate theories

$f(f(x) - f(y)) = a$
$\wedge$
$f(0) = a + 2$
$\wedge$
$x = y$

$f(e_1) = a$
$\wedge$
$e_1 = f(x) - f(y)$
$\wedge$
$f(0) = a + 2$
$\wedge$
$x = y$

$f(e_1) = a$
$\wedge$
$e_1 = e_2 - e_3$
$\wedge$
$e_2 = f(x)$
$\wedge$
$e_3 = f(y)$
$\wedge$
$f(0) = a + 2$
$\wedge$
$x = y$

EXAMPLE

SMT SOLVING

Step 1: Nelson-Oppen procedure to separate theories

$f(e_1) = a$
$\wedge$
$e_1 = e_2 - e_3$
$\wedge$
$e_2 = f(x)$
$\wedge$
$e_3 = f(y)$
$\wedge$
$f(0) = a + 2$
$\wedge$
$x = y$

$f(e_1) = a$
$\wedge$
$e_1 = e_2 - e_3$
$\wedge$
$e_2 = f(x)$
$\wedge$
$e_3 = f(y)$
$\wedge$
$f(e_4) = a + 2$
$\wedge$
$e_4 = 0$
$\wedge$
$x = y$

$f(e_1) = a$
$\wedge$
$e_1 = e_2 - e_3$
$\wedge$
$e_2 = f(x)$
$\wedge$
$e_3 = f(y)$
$\wedge$
$f(e_4) = e_5$
$\wedge$
$e_4 = 0$
$\wedge$
$e_5 = a + 2$
$\wedge$
$x = y$

EXAMPLE

SMT SOLVING

$f(e_1) = a$ — Theory of EUF

$\wedge$

$e_1 = e_2 - e_3$ — Theory of integer arithmetic

$\wedge$

$e_2 = f(x)$ — Theory of EUF

$\wedge$

$e_3 = f(y)$ — Theory of EUF

$\wedge$

$f(e_4) = e_5$ — Theory of EUF

$\wedge$

$e_4 = 0$ — Theory of integer arithmetic

$\wedge$

$e_5 = a + 2$ — Theory of integer arithmetic

$\wedge$

$x = y$ — Theory of EUF AND Theory of integer arithmetic

EXAMPLE

SMT SOLVING

$f(e_1) = a$

$\wedge$

$e_1 = e_2 - e_3$

$\wedge$

$e_2 = f(x)$

$\wedge$

$e_3 = f(y)$

$\wedge$

$f(e_4) = e_5$

$\wedge$

$e_4 = 0$

$\wedge$

$e_5 = a + 2$

$\wedge$

$x = y$

$\wedge$

$f(x) = f(y)$

Some EUF Axioms

Congruence:

$x = y \Rightarrow f(x) = f(y)$

Symmetry

$x = y \Rightarrow y = x$

Transitivity:

$x = y \wedge y = z \Rightarrow x = z$

...

EXAMPLE
SMT SOLVING

$f(e_1) = a$
$\wedge$
$e_1 = e_2 - e_3$
$\wedge$
$e_2 = f(x)$
$\wedge$
$e_3 = f(y)$
$\wedge$
$f(e_4) = e_5$
$\wedge$
$e_4 = 0$
$\wedge$
$e_5 = a + 2$
$\wedge$
$x = y$
$\wedge$
$f(x) = f(y)$
$\wedge$
$e_2 = e_3$

Some EUF Axioms

Congruence:

$x = y \Rightarrow f(x) = f(y)$

Symmetry

$x = y \Rightarrow y = x$

Transitivity:

$x = y \wedge y = z \Rightarrow x = z$

...

EXAMPLE
SMT SOLVING

$f(e_1) = a$
$\wedge$
$e_1 = e_2 - e_3$
$\wedge$
$e_2 = f(x)$
$\wedge$
$e_3 = f(y)$
$\wedge$
$f(e_4) = e_5$
$\wedge$
$e_4 = 0$
$\wedge$
$e_5 = a + 2$
$\wedge$
$x = y$
$\wedge$
$f(x) = f(y)$
$\wedge$
$e_2 = e_3$

$\wedge$
$e_2 - e_3 = 0$

Some EUF Axioms

Congruence:

$x = y \Rightarrow f(x) = f(y)$

Symmetry

$x = y \Rightarrow y = x$

Transitivity:

$x = y \wedge y = z \Rightarrow x = z$

...

EXAMPLE
SMT SOLVING

$f(e_1) = a$
$\wedge$
$e_1 = e_2 - e_3$
$\wedge$
$e_2 = f(x)$
$\wedge$
$e_3 = f(y)$
$\wedge$
$f(e_4) = e_5$
$\wedge$
$e_4 = 0$
$\wedge$
$e_5 = a + 2$
$\wedge$
$x = y$
$\wedge$
$f(x) = f(y)$
$\wedge$
$e_2 = e_3$

$\wedge$
$e_2 - e_3 = 0$
$\wedge$
$e_1 = 0$

Some EUF Axioms

Congruence:

$x = y \Rightarrow f(x) = f(y)$

Symmetry

$x = y \Rightarrow y = x$

Transitivity:

$x = y \wedge y = z \Rightarrow x = z$

...

EXAMPLE

SMT SOLVING

$f(e_1) = a$

$\wedge$

$e_1 = e_2 - e_3$

$\wedge$

$e_2 = f(x)$

$\wedge$

$e_3 = f(y)$

$\wedge$

$f(e_4) = e_5$

$\wedge$

$e_4 = 0$

$\wedge$

$e_5 = a + 2$

$\wedge$

$x = y$

$\wedge$

$f(x) = f(y)$

$\wedge$

$e_2 = e_3$

$\wedge$

$e_2 - e_3 = 0$

$\wedge$

$e_1 = 0$

$\wedge$

$e_1 = e_4$

Some EUF Axioms

Congruence:

$x = y \Rightarrow f(x) = f(y)$

Symmetry

$x = y \Rightarrow y = x$

Transitivity:

$x = y \wedge y = z \Rightarrow x = z$

...

EXAMPLE
SMT SOLVING

$f(e_1) = a$
$\wedge$
$e_1 = e_2 - e_3$
$\wedge$
$e_2 = f(x)$
$\wedge$
$e_3 = f(y)$
$\wedge$
$f(e_4) = e_5$
$\wedge$
$e_4 = 0$
$\wedge$
$e_5 = a + 2$
$\wedge$
$x = y$
$\wedge$
$f(x) = f(y)$
$\wedge$
$e_2 = e_3$

$\wedge$
$e_2 - e_3 = 0$
$\wedge$
$e_1 = 0$
$\wedge$
$e_1 = e_4$
$\wedge$
$f(0) = a$

Some EUF Axioms

Congruence:

$x = y \Rightarrow f(x) = f(y)$

Symmetry

$x = y \Rightarrow y = x$

Transitivity:

$x = y \wedge y = z \Rightarrow x = z$

...

EXAMPLE
SMT SOLVING

$f(e_1) = a$
$\wedge$
$e_1 = e_2 - e_3$
$\wedge$
$e_2 = f(x)$
$\wedge$
$e_3 = f(y)$
$\wedge$
$f(e_4) = e_5$
$\wedge$
$e_4 = 0$
$\wedge$
$e_5 = a + 2$
$\wedge$
$x = y$
$\wedge$
$f(x) = f(y)$
$\wedge$
$e_2 = e_3$

$\wedge$
$e_2 - e_3 = 0$
$\wedge$
$e_1 = 0$
$\wedge$
$e_1 = e_4$
$\wedge$
$f(0) = a$
$\wedge$
$f(0) = e_5$

Some EUF Axioms

Congruence:

$x = y \Rightarrow f(x) = f(y)$

Symmetry

$x = y \Rightarrow y = x$

Transitivity:

$x = y \wedge y = z \Rightarrow x = z$

...

EXAMPLE
SMT SOLVING

$f(e_1) = a$

$\wedge$

$e_1 = e_2 - e_3$ $\wedge$ $e_2 - e_3 = 0$

$\wedge$

$e_2 = f(x)$ $\wedge$ $e_1 = 0$

$\wedge$

$e_3 = f(y)$ $\wedge$ $e_1 = e_4$

$\wedge$

$f(e_4) = e_5$ $\wedge$ $f(0) = a$

$\wedge$

$e_4 = 0$ $\wedge$ $f(0) = e_5$

$\wedge$

$e_5 = a + 2$ $\wedge$ $e_5 = a$

Arithmetic Contradiction

$\wedge$

$x = y$

$\wedge$

$f(x) = f(y)$

$\wedge$

$e_2 = e_3$

Some EUF Axioms

Congruence:

$x = y \Rightarrow f(x) = f(y)$

Symmetry

$x = y \Rightarrow y = x$

Transitivity:

$x = y \wedge y = z \Rightarrow x = z$

...

“CONVENIENT” EQUALITIES

SMT SOLVING

$f(e_1) = a$
$\wedge$
$e_1 = e_2 - e_3$
$\wedge$
$e_2 = f(x)$
$\wedge$
$e_3 = f(y)$
$\wedge$
$f(e_4) = e_5$
$\wedge$
$e_4 = 0$
$\wedge$
$e_5 = a + 2$
$\wedge$
$x = y$
$\wedge$
$f(x) = f(y)$
$\wedge$
$e_2 = e_3$

$\wedge$
$e_2 - e_3 = 0$
$\wedge$
$e_1 = 0$
$\wedge$
$e_1 = e_4$
$\wedge$
$f(0) = a$
$\wedge$
$f(0) = e_5$
$\wedge$
$e_5 = a$

The lynchpin of our success was the existence of some useful equalities. What if they aren’t in the original constraints?

Case split!

Can add logical predicates for all possible equalities…

$(e_1 = e_2 \vee e_1 \neq e_2)$
$\wedge$
$(e2 = e3 \vee e2 \neq e3)$
$\wedge$
$(e1 = e3 \vee e1 \neq e3)$
$\wedge$
…

and start making guesses

“CONVENIENT” EQUALITIES

SMT SOLVING

$x \geq 0 \wedge y = x + 1 \wedge (y > 2 \vee y < 1)$

Abstract all non-logical clauses

$p1 \wedge p2 \wedge (p3 \vee p4)$

DPLL

p1:true
p2:true
p3:false
p4: true

Linear Solver: contradiction!

Add information and start over

$p1 \wedge p2 \wedge (p3 \vee p4) \wedge (\neg p1 \vee \neg p2 \vee \neg p3)$

The lynchpin of our success was the existence of some useful equalities. What if they aren’t in the original constraints?

Case split!

Can add logical predicates for all possible equalities…

$$(e_1 = e_2 \vee e_1 \neq e_2)$$
$$\wedge$$
$$(e2 = e3 \vee e2 \neq e3)$$
$$\wedge$$
$$(e1 = e3 \vee e1 \neq e3)$$
$$\wedge$$
$$\ldots$$

and start making smart guesses

ARITHMETIC CONSTRAINTS

SMT SOLVING

We kinda danced around how the arithmetic solver works

Basic answer: Linear Algebra.

Also, something something Linear Optimization and the simplex algorithm

SOMETHING SOMETHING LINEAR OPTIMIZATION AND THE SIMPLEX ALGORITHM

SMT SOLVING

Overview [edit]

Further information: Linear programming

The simplex algorithm operates on linear programs in the canonical form

maximize $\mathbf{c^T x}$

subject to $A\mathbf{x} \leq \mathbf{b}$ and $\mathbf{x} \geq 0$

with $\mathbf{c} = (c_1, \ldots, c_n)$ the coefficients of the objective function, $(\cdot)^{\mathrm{T}}$ is the matrix transpose, and $\mathbf{x} = (x_1, \ldots, x_n)$ are the variables of the problem, A is a *p*×*n* matrix, and $\mathbf{b} = (b_1, \ldots, b_p)$. There is a straightforward process to convert any linear program into one in standard form, so using this form of linear programs results in no loss of generality.

In geometric terms, the feasible region defined by all values of $\mathbf{x}$ such that $A\mathbf{x} \leq \mathbf{b}$ and $\forall i, x_i \geq 0$ is a (possibly unbounded) convex polytope. An extreme point or vertex of this polytope is known as *basic feasible solution* (BFS).

SOMETHING SOMETHING LINEAR OPTIMIZATION AND THE SIMPLEX ALGORITHM

SMT SOLVING

Overview [edit]

Further information: Linear programming

The simplex algorithm operates on linear programs in the canonical form

maximize $\mathbf{c^T x}$

subject to $A\mathbf{x} \leq \mathbf{b}$ and $\mathbf{x} \geq 0$

with $\mathbf{c} = (c_1, \ldots, c_n)$ the coefficients of the objective function, $(\cdot)^{\mathrm{T}}$ is the matrix transpose, and $\mathbf{x} = (x_1, \ldots, x_n)$ are the variables of the problem, A is a *p*×*n* matrix, and $\mathbf{b} = (b_1, \ldots, b_p)$. There is a straightforward process to convert any linear program into one in standard form, so using this form of linear programs results in no loss of generality.

In geometric terms, the feasible region defined by all values of $\mathbf{x}$ such that $A\mathbf{x} \leq \mathbf{b}$ and $\forall i, x_i \geq 0$ is a (possibly unbounded) convex polytope. An extreme point or vertex of this polytope is known as *basic feasible solution* (BFS).

WRAP-UP
SMT SOLVERS

Hopefully I've convinced you that Solvers can be implemented

Not strictly magic, but they do employ some very clever techniques