Combinational Equivalence Checking

Normal Forms and Tseitin Encoding

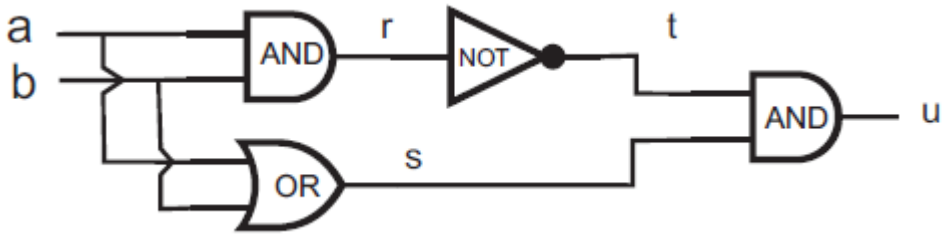
Bettina Könighofer

IAIK – Graz University of Technology
bettina.koenighofer@iaik.tugraz.at

Motivation

- Basic Concepts
  - Normal Forms
  - Tseitin Encoding
- Understand Relation between
  - Satisfiability
  - Validity
  - Entailment
  - Equivalence
- Application
  - Equivalence Checking

Combinational Equivalence Checking



?

Equivalence of Circuits

- Circuit Optimization Tools
  - Big market
  - Lot's of money

- But: Mistakes!

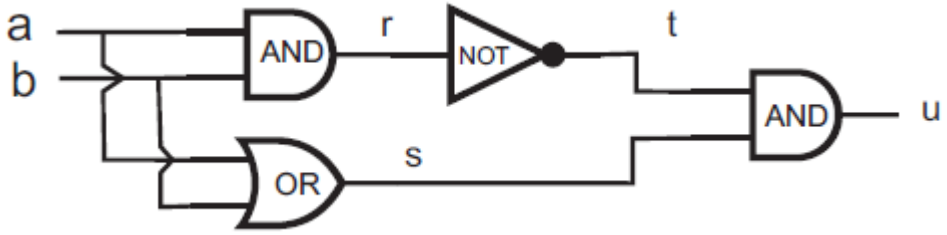
- Check Equivalence
  - Modeling with propositional formulas
  - Reduction to Formula Equivalence

Equivalence of Formulas

- Truth Table: $\phi \vDash \psi$ and $\psi \vDash \phi$?
  - Exponentially large
  - → Not practicable

- Better way?
  - Reduction to SAT

Circuit Equivalence



$$u = t \wedge s$$
$$= \neg r \wedge (a \vee b)$$
$$= \neg(a \wedge b) \wedge (a \vee b)$$

$$q = a \oplus b$$
$$= (a \wedge \neg b) \vee (\neg a \wedge b)$$

Circuits are **equivalent** $\Leftrightarrow$ $u \oplus q$ is **unsatisfiable**.

SAT Solver takes CNF as input → Convert to CNF using **Tseitin** Encoding

Duality: Validity – Satisfiability

- ϕ is valid $\Leftrightarrow$ $\neg\phi$ is not satisfiable.
  - Example:
    $\phi = (x \vee \neg x)$ is valid.
    (Truth Table: All rows T.)

    $\neg\phi = \neg(x \vee \neg x) \equiv \neg x \wedge x$ is not satisfiable.
    (Truth Table: All rows F.)

- ϕ is satisfiable $\Leftrightarrow$ $\neg\phi$ is not valid.

➔ Only **one** decision procedure

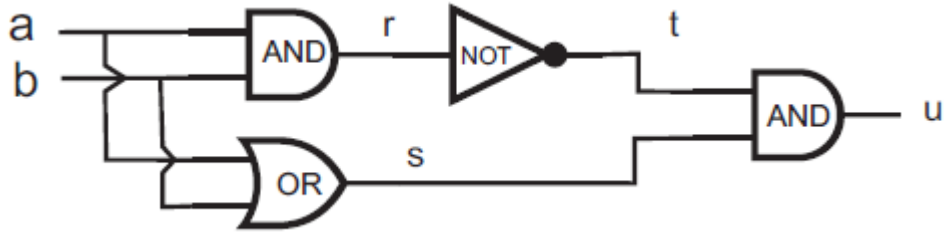
Reductions

Solve / using	ϕ satisfiable?	ϕ valid?	$\phi \vdash \psi$?	$\phi \equiv \psi$?
Satisfiability	✓	$\neg\phi$ not satisfiable?	$\phi \wedge \neg\psi$ not satisfiable?	$\phi \oplus \psi$ not satisfiable?
Validity	$\neg\phi$ not valid?	✓	$\phi \rightarrow \psi$ valid?	$\phi \leftrightarrow \psi$ valid?
Entailment	$\top \nvdash \neg\phi$?	$\top \vdash \phi$?	✓	$\phi \vdash \psi$ and $\psi \vdash \phi$?
Equivalence	$\phi \not\equiv \bot$?	$\phi \equiv \top$?	$\phi \rightarrow \psi \equiv \top$?	✓

SAT Problem

- Given formula, decide:
  - SAT
  - UNSAT
- NP-complete
  - $\mathrm{P} \neq \mathrm{NP} \Rightarrow$ worst-case exponential
- Automated Tools: **"SAT Solver"**
  - Surprisingly high scalability
  - **Require formula to be in special form**
    - Conjunctive Normal Form (CNF)

Circuit Equivalence



$$u = t \wedge s$$
$$= \neg r \wedge (a \vee b)$$
$$= \neg(a \wedge b) \wedge (a \vee b)$$

$$q = a \oplus b$$
$$= (a \wedge \neg b) \vee (\neg a \wedge b)$$

Circuits are **equivalent** $\Leftrightarrow$ $u \oplus q$ is **unsatisfiable**.

SAT Solver takes CNF as input → Convert to CNF using **Tseitin** Encoding

Terminology

- **Definitions:**
  - **Literal:** propositional symbol or its negation
    - Example: $p,\ \neg q$
  - **Clause:** disjunction of literals
    - Example: $(p \vee \neg q \vee r)$
  - **Cube:** conjunction of literals
    - Example: $(\neg x \wedge y \wedge \neg z)$

Disjunctive Normal Form (DNF)

- Disjunction of cubes:

$$(a_1 \wedge a_2 \wedge \cdots \wedge a_n) \vee (b_1 \wedge \cdots \wedge b_m) \vee \cdots$$

where each a_i, b_j is a literal.

DNF from Truth Table

Example:

p	q	r	$(r \vee q) \rightarrow (p \wedge \neg q)$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

$\neg p \wedge \neg q \wedge \neg r$

V

$p \wedge \neg q \wedge \neg r$

V

$p \wedge \neg q \wedge r$

DNF: $(\neg p \wedge \neg q \wedge \neg r)$ V $(p \wedge \neg q \wedge \neg r)$ V $(p \wedge \neg q \wedge r)$

Conjunctive Normal Form (CNF)

- Conjunction of clauses:

$$(a_1 \vee a_2 \vee \cdots \vee a_n) \wedge (b_1 \vee \cdots \vee b_m) \wedge \cdots$$

where each a_i, b_j is a literal.

CNF from Truth Table

Example:

p	q	r	$(p \vee \neg q) \rightarrow r$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

$p \vee q \vee r$

$\wedge$

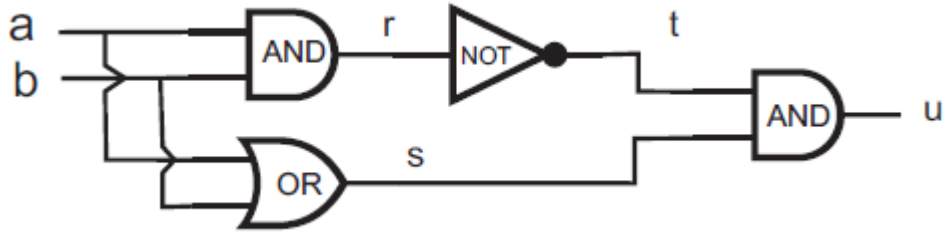
$\neg p \vee q \vee r$

$\wedge$

$\neg p \vee \neg q \vee r$

CNF: $(p \vee q \vee r) \wedge (\neg p \vee q \vee r) \wedge (\neg p \vee \neg q \vee r)$

Circuit Equivalence



$$u = t \wedge s$$
$$= \neg r \wedge (a \vee b)$$
$$= \neg(a \wedge b) \wedge (a \vee b)$$

$$q = a \oplus b$$
$$= (a \wedge \neg b) \vee (\neg a \wedge b)$$

Circuits are **equivalent** $\Leftrightarrow$ $u \oplus q$ is **unsatisfiable**.

SAT Solver takes CNF as input → Convert to CNF using **Tseitin** Encoding

Ways of Obtaining a CNF

- Via Truth Table
  - Exponential size
- Via Replacement Rules, DeMorgan, Distributivity
  - Exponential size
- **Tseitin Encoding**
  - Linear Blowup ☺
  - **Produces equisatisfiable Formula**

Equisatisfiability

- **Definition Equisatisfiability**

  ϕ and ψ are *equisatisfiable*

  either both satisfiable, or both unsatisfiable.

Tseitin Encoding

- **Step 1:**
  - **A**ssign new variables to all nodes in the parse tree / to each sub-formula
- Step 2:
  - Add new clauses for each new variable
  - Tseitin Rewrite Rules

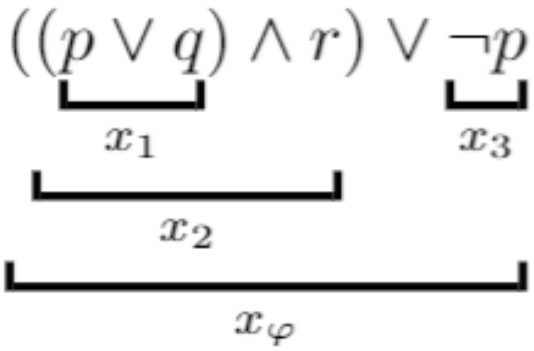
$$\chi \leftrightarrow (\varphi \vee \psi) \quad \Leftrightarrow \quad (\neg\varphi \vee \chi) \wedge (\neg\psi \vee \chi) \wedge (\neg\chi \vee \varphi \vee \psi)$$

$$\chi \leftrightarrow (\varphi \wedge \psi) \quad \Leftrightarrow \quad (\neg\chi \vee \varphi) \wedge (\neg\chi \vee \psi) \wedge (\neg\varphi \vee \neg\psi \vee \chi)$$

$$\chi \leftrightarrow \neg\varphi \quad \Leftrightarrow \quad (\neg\chi \vee \neg\varphi) \wedge (\varphi \vee \chi)$$

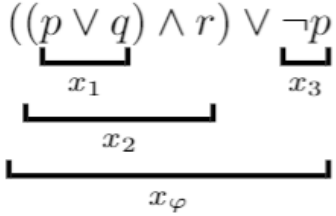
Tseitin Encoding

- **Step 1:**
  - **A**ssign new variables to all nodes in the parse tree / to each sub-formula



Tseitin Encoding

- **Step 2:**
  - Add new clauses for each new variable



$$\chi \leftrightarrow (\varphi \vee \psi) \quad \Leftrightarrow \quad (\neg\varphi \vee \chi) \wedge (\neg\psi \vee \chi) \wedge (\neg\chi \vee \varphi \vee \psi)$$
$$\chi \leftrightarrow (\varphi \wedge \psi) \quad \Leftrightarrow \quad (\neg\chi \vee \varphi) \wedge (\neg\chi \vee \psi) \wedge (\neg\varphi \vee \neg\psi \vee \chi)$$
$$\chi \leftrightarrow \neg\varphi \quad \Leftrightarrow \quad (\neg\chi \vee \neg\varphi) \wedge (\varphi \vee \chi)$$

$$\begin{aligned} CNF(\varphi) = \quad & (\neg p \vee x_1) \wedge (\neg q \vee x_1) \wedge (\neg x_1 \vee p \vee q) \\ & \wedge (\neg x_2 \vee x_1) \wedge (\neg x_2 \wedge r) \wedge (\neg x_1 \vee \neg r \vee x_2) \\ & \wedge (\neg x_3 \vee \neg p) \wedge (p \vee x_3) \\ & \wedge (\neg x_2 \vee x_\varphi) \wedge (\neg x_3 \vee x_\varphi) \wedge (\neg x_\varphi \vee x_2 \vee x_3) \\ & \wedge x_\varphi \end{aligned}$$

Rewriting to CNF

- $r \leftrightarrow (p \wedge q)$

- $(r \rightarrow p) \wedge (r \rightarrow q) \wedge (p \wedge q \rightarrow r)$

$a \rightarrow b \equiv \neg a \vee b$

- $(\neg r \vee p) \wedge (\neg r \vee q) \wedge (\neg p \vee \neg q \vee r)$

Rewriting to CNF

- $r \leftrightarrow (p \vee q)$

- $(p \rightarrow r) \wedge (q \rightarrow r) \wedge (r \rightarrow p \vee q)$

- $(\neg p \vee r) \wedge (\neg q \vee r) \wedge (\neg r \vee p \vee q)$

Rewriting to CNF

- $r \leftrightarrow \neg p$

- $(r \rightarrow \neg p) \wedge (\neg p \rightarrow r)$

- $(\neg r \vee \neg p) \wedge (p \vee r)$

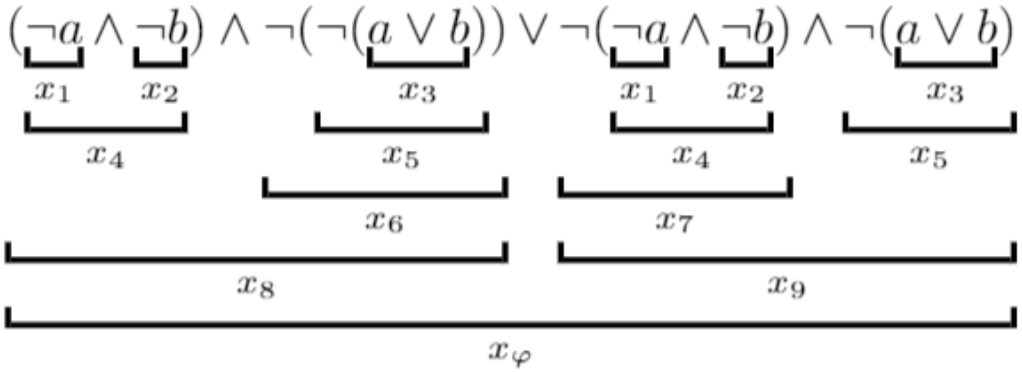
Example $\varphi_1 = \neg a \wedge \neg b$ and $\varphi_2 = \neg(a \vee b)$

- Step 1: Build the XOR

$$\varphi = \varphi_1 \oplus \varphi_2 = (\neg a \wedge \neg b) \oplus \neg(a \vee b)$$
$$= (\neg a \wedge \neg b) \wedge \neg(\neg(a \vee b)) \ \vee \ \neg(\neg a \wedge \neg b) \wedge \neg(a \vee b)$$

Example $\varphi_1 = \neg a \wedge \neg b$ and $\varphi_2 = \neg(a \vee b)$

- Step 2: Transform to CNF using Tseitin

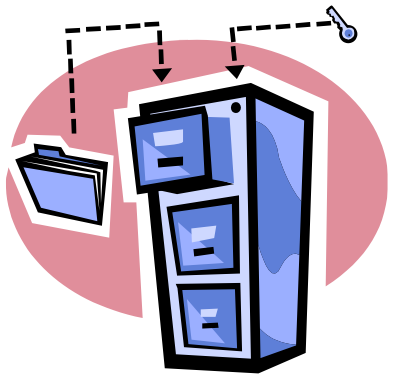


$$CNF(\varphi) = x_\varphi \wedge$$
$$(\neg x_8 \vee x_\varphi) \wedge (\neg x_9 \vee x_\varphi) \wedge (\neg x_\varphi \vee x_9 \vee x_8) \wedge$$
$$(\neg x_9 \vee x_7) \wedge (\neg x_9 \vee x_5) \wedge (\neg x_7 \vee \neg x_5 \vee x_9) \wedge$$
$$(\neg x_8 \vee x_4) \wedge (\neg x_8 \vee x_6) \wedge (\neg x_4 \vee \neg x_6 \vee x_8) \wedge$$
$$(\neg x_7 \vee \neg x_4) \wedge (x_7 \vee x_4) \wedge$$
$$(\neg x_6 \vee \neg x_5) \wedge (x_6 \vee x_5) \wedge$$
$$(\neg x_5 \vee \neg x_3) \wedge (x_5 \vee x_3) \wedge$$
$$(\neg x_4 \vee x_1) \wedge (\neg x_4 \vee x_3) \wedge (\neg x_1 \vee \neg x_3 \vee x_4) \wedge$$
$$(\neg a \vee x_3) \wedge (\neg b \vee x_3) \wedge (\neg x_3 \vee a \vee b) \wedge$$
$$(\neg x_2 \vee \neg b) \wedge (x_2 \vee b) \wedge$$
$$(\neg x_1 \vee \neg a) \wedge (x_1 \vee a)$$

Example $\varphi_1 = \neg a \wedge \neg b$ and $\varphi_2 = \neg(a \vee b)$

- Step 3: Give CNF(φ) to SAT solver

φ_1 and φ_2 are equivalent $\Leftrightarrow$ CNF(φ) is unsatisfiable.



Summary

- Relation between…
  - Satisfiability, validity, entailment, equivalence.
  - Circuit Equivalence → Satisfiability
- Normal Forms
  - Disjunctive Normal Form (DNF)
  - Conjunctive Normal Form (CNF)
  - From Truth Tables
- Equisatisfiability
- Tseitin's Encoding
- DIMACS Format

→ Learning Targets