# Logic and Mechanized Reasoning
## Basic SAT Techniques

**Marijn J.H. Heule**

Carnegie Mellon University

Tseitin Transformation

Unit Propagation and Resolution

Pure Literals and Autarkies

Tseitin Transformation

Unit Propagation and Resolution

Pure Literals and Autarkies

# Tseitin: Introduction

Recall: converting a propositional formula $A$ into CNF can result in an exponential blowup. How to avoid that?

# Tseitin: Introduction

Recall: converting a propositional formula $A$ into CNF can result in an exponential blowup. How to avoid that?

Idea: focus on converting $A$ into an equisatisfiable CNF formula $A'$ (instead of logical equivalence)

# Tseitin: Introduction

Recall: converting a propositional formula $A$ into CNF can result in an exponential blowup. How to avoid that?

Idea: focus on converting $A$ into an equisatisfiable CNF formula $A'$ (instead of logical equivalence)

How: add definitions and replace parts of the formula (can be seen as the reverse of substitution)

# Tseitin: Small Example

Consider the formula $\Gamma = p \vee (q \wedge r)$

# Tseitin: Small Example

Consider the formula $\Gamma = p \vee (q \wedge r)$

We can add the definition $d \leftrightarrow (q \wedge r)$

# Tseitin: Small Example

Consider the formula $\Gamma = p \vee (q \wedge r)$

We can add the definition $d \leftrightarrow (q \wedge r)$

Replacing $(q \wedge r)$ by $d$ results in CNF $p \vee d$

# Tseitin: Small Example

Consider the formula $\Gamma = p \vee (q \wedge r)$

We can add the definition $d \leftrightarrow (q \wedge r)$

Replacing $(q \wedge r)$ by $d$ results in CNF $p \vee d$

The clauses representing the definition are:

$$(\neg d \vee q) \wedge (\neg d \vee r) \wedge (d \vee \neg q \vee \neg r)$$

An equisatisfiable formula of $\Gamma$ in CNF is:

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

Satisfying the resulting formula satisfies $\Gamma$ on original variables

# Tseitin: A Linear-Size Transformation

Why is the Tseitin transformation interesting?

- Each connective can be replaced by a new definition
- At most a linear number of definitions
- Definitions can be easily converted into clauses
- Easily obtain a satisfying assignment for original formula
- Resulting in an efficient transformation into CNF

# Tseitin: Implementation and Optimizations

Implementation:

- Convert the formula first to NNF
- Generate the definitions from left to right

# Tseitin: Implementation and Optimizations

Implementation:

- Convert the formula first to NNF
- Generate the definitions from left to right

Optimizations:

- Reuse definitions when possible
- Avoid definitions by interpreting an NNF formula as a CNF formula: e.g. $p \vee (q \wedge \neg r) \vee \neg s$
- Mostly one direction of definition is required

# Tseitin: Definitions into Clauses

It is easy to turn a definition $d \leftrightarrow \mathrm{DEF}(p_1, \ldots, p_n)$ into clauses

Example

| def | $\Gamma_d$ | $\Gamma_{\neg d}$ |
|---|---|---|
| $\mathrm{AND}(p_1, \ldots, p_n)$ | $(d \vee \neg p_1 \vee \cdots \vee \neg p_n)$ | $(\neg d \vee p_1), \ldots, (\neg d \vee p_n)$ |
| $\mathrm{OR}(p_1, \ldots, p_n)$ | $(d \vee \neg p_1), \ldots, (d \vee \neg p_n)$ | $(\neg d \vee p_1 \vee \cdots \vee p_n)$ |
| $\mathrm{ITE}(c, t, f)$ | $(d \vee \neg c \vee \neg t), (d \vee c \vee \neg f)$ | $(\neg d \vee \neg c \vee t), (\neg d \vee c \vee f)$ |

## Tseitin: Larger Example without Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$

# Tseitin: Larger Example without Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$
- $d_1 \leftrightarrow d_0 \wedge \neg r$

# Tseitin: Larger Example without Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$
- $d_1 \leftrightarrow d_0 \wedge \neg r$
- $d_2 \leftrightarrow \neg p \vee \neg q$

# Tseitin: Larger Example without Optimization

Consider the formula $\Gamma = \neg(p \land q \leftrightarrow r) \land (s \rightarrow (p \land t))$

Convert into NNF:

$$\big((p \land q \land \neg r) \lor (r \land (\neg p \lor \neg q))\big) \land (\neg s \lor (p \land t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \land q$
- $d_1 \leftrightarrow d_0 \land \neg r$
- $d_2 \leftrightarrow \neg p \lor \neg q$
- $d_3 \leftrightarrow r \land d_2$

# Tseitin: Larger Example without Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$
- $d_1 \leftrightarrow d_0 \wedge \neg r$
- $d_2 \leftrightarrow \neg p \vee \neg q$
- $d_3 \leftrightarrow r \wedge d_2$
- $d_4 \leftrightarrow d_1 \vee d_3$

# Tseitin: Larger Example without Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$
- $d_1 \leftrightarrow d_0 \wedge \neg r$
- $d_2 \leftrightarrow \neg p \vee \neg q$
- $d_3 \leftrightarrow r \wedge d_2$
- $d_4 \leftrightarrow d_1 \vee d_3$
- $d_5 \leftrightarrow p \wedge t$

# Tseitin: Larger Example without Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$
- $d_1 \leftrightarrow d_0 \wedge \neg r$
- $d_2 \leftrightarrow \neg p \vee \neg q$
- $d_3 \leftrightarrow r \wedge d_2$
- $d_4 \leftrightarrow d_1 \vee d_3$
- $d_5 \leftrightarrow p \wedge t$
- $d_6 \leftrightarrow \neg s \vee d_5$

# Tseitin: Larger Example without Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$
- $d_1 \leftrightarrow d_0 \wedge \neg r$
- $d_2 \leftrightarrow \neg p \vee \neg q$
- $d_3 \leftrightarrow r \wedge d_2$
- $d_4 \leftrightarrow d_1 \vee d_3$
- $d_5 \leftrightarrow p \wedge t$
- $d_6 \leftrightarrow \neg s \vee d_5$
- $d_7 \leftrightarrow d_4 \wedge d_6$

# Tseitin: Larger Example with Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF and interpret as CNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$

# Tseitin: Larger Example with Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF and interpret as CNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$
- $d_1 \leftrightarrow d_0 \wedge \neg r$

# Tseitin: Larger Example with Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \to (p \wedge t))$

Convert into NNF and interpret as CNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$
- $d_1 \leftrightarrow d_0 \wedge \neg r$
- $d_2 \leftrightarrow \neg p \vee \neg q$

# Tseitin: Larger Example with Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF and interpret as CNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$
- $d_1 \leftrightarrow d_0 \wedge \neg r$
- $d_2 \leftrightarrow \neg p \vee \neg q$
- $d_3 \leftrightarrow r \wedge d_2$

# Tseitin: Larger Example with Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF and interpret as CNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$
- $d_1 \leftrightarrow d_0 \wedge \neg r$
- $d_2 \leftrightarrow \neg p \vee \neg q$
- $d_3 \leftrightarrow r \wedge d_2$
- $d_4 \leftrightarrow p \wedge t$

# Tseitin: Larger Example with Optimization

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF and interpret as CNF:

$$\big((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))\big) \wedge (\neg s \vee (p \wedge t))$$

Which results in the following definitions:

- $d_0 \leftrightarrow p \wedge q$
- $d_1 \leftrightarrow d_0 \wedge \neg r$
- $d_2 \leftrightarrow \neg p \vee \neg q$
- $d_3 \leftrightarrow r \wedge d_2$
- $d_4 \leftrightarrow p \wedge t$

Final result: $(d_1 \vee d_3) \wedge (\neg s \vee d_4)$ plus definition clauses

# Tseitin: Plaisted-Greenbaum Encoding

In most cases only one direction of the definition is required.

Example

Recall the formula $\Gamma = p \vee (q \wedge r)$

The Tseitin transformation resulted in the CNF:

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

Which clause is redundant (not required for equisatisfiability)?

# Tseitin: Plaisted-Greenbaum Encoding

In most cases only one direction of the definition is required.

Example

Recall the formula $\Gamma = p \vee (q \wedge r)$

The Tseitin transformation resulted in the CNF:

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

Which clause is redundant (not required for equisatisfiability)?

Removing $(d \vee \neg q \vee \neg r)$ reduces $d \leftrightarrow q \wedge r$ to $d \rightarrow q \wedge r$

# Tseitin: Plaisted-Greenbaum Encoding

In most cases only one direction of the definition is required.

Example

Recall the formula $\Gamma = p \lor (q \land r)$

The Tseitin transformation resulted in the CNF:

$$(p \lor d) \land (\neg d \lor q) \land (\neg d \lor r) \land (d \lor \neg q \lor \neg r)$$

Which clause is redundant (not required for equisatisfiability)?

Removing $(d \lor \neg q \lor \neg r)$ reduces $d \leftrightarrow q \land r$ to $d \rightarrow q \land r$

When starting with NNF, we only need $d \rightarrow \mathrm{DEF}$

# Tseitin: Bringing it all Together

Consider the formula $\Gamma = \neg(p \wedge q \leftrightarrow r) \wedge (s \rightarrow (p \wedge t))$

Convert into NNF and interpret as CNF:

$$((p \wedge q \wedge \neg r) \vee (r \wedge (\neg p \vee \neg q))) \wedge (\neg s \vee (p \wedge t))$$

The Tseitin transformation results in the following clauses:

$$(d_3 \vee d_1) \wedge (d_4 \vee \neg s) \wedge (\neg d_0 \vee p) \wedge (\neg d_0 \vee q) \wedge (\neg p \vee \neg q \vee d_0) \wedge$$
$$(\neg d_1 \vee d_0) \wedge (\neg d_1 \vee \neg r) \wedge (\neg d_0 \vee r \vee d_1) \wedge (\neg d_2 \vee \neg p \vee \neg q) \wedge$$
$$(p \vee d_2) \wedge (q \vee d_2) \wedge (\neg d_3 \vee r) \wedge (\neg d_3 \vee d_2) \wedge$$
$$(\neg r \vee \neg d_2 \vee d_3) \wedge (\neg d_4 \vee p) \wedge (\neg d_4 \vee t) \wedge (\neg p \vee \neg t \vee d_4)$$

Plaisted-Greenbaum removed the colored ones ($d_i \leftarrow \mathrm{DEF}$).

Tseitin Transformation

Unit Propagation and Resolution

Pure Literals and Autarkies

# Unit Propagation: Introduction

Unit propagation (UP) is the most important SAT solving simplification technique:

- A clause is unit if it has only one literal
- The only way to satisfy it is assigning the literal to $\top$
- Removing falsified literals can produce unit clauses
- Satisfying unit clauses until fixpoint can be expensive

# Unit Propagation: Partial Assignments

Evaluation of clauses and formulas can be generalized to partial assignments:

- Only some variables are assigned to $\top$, $\bot$
- For a clause $C$, $[\![C]\!]_\tau$ removes literals falsified by $\tau$ from $C$
  - $[\![C]\!]_\tau = \top$ if $\tau$ satisfies a literal in $C$
- For a formula $\Gamma$, $[\![\Gamma]\!]_\tau$ replaces all clauses $C \in \Gamma$ by $[\![C]\!]_\tau$
  - Clauses satisfied by $\tau$ are removed from $[\![\Gamma]\!]_\tau$

Partial assignments are very important in SAT solving

# Unit Propagation: Extending the Assignment

Unit propagation makes unit clauses true until fixpoint

Given an assignment $\tau$ and a formula $\Gamma$, unit propagation extends $\tau$ by assigning all unit clauses in $[\![\Gamma]\!]_\tau$ to $\top$.

# Unit Propagation: Extending the Assignment

Unit propagation makes unit clauses true until fixpoint

Given an assignment $\tau$ and a formula $\Gamma$, unit propagation extends $\tau$ by assigning all unit clauses in $[\![\Gamma]\!]_\tau$ to $\top$.

Two possible fixpoints (termination)

1. $[\![\Gamma]\!]_\tau$ contains a falsified clause ($\bot$)
2. $[\![\Gamma]\!]_\tau$ contains no more unit clauses

# Unit Propagation: Extending the Assignment

Unit propagation makes unit clauses true until fixpoint

Given an assignment $\tau$ and a formula $\Gamma$, unit propagation extends $\tau$ by assigning all unit clauses in $[\![\Gamma]\!]_\tau$ to $\top$.

Two possible fixpoints (termination)

1. $[\![\Gamma]\!]_\tau$ contains a falsified clause ($\bot$)
2. $[\![\Gamma]\!]_\tau$ contains no more unit clauses

Unit propagation can consume 90% of solver runtime

- Data-structures are optimized for unit propagation
- Unit propagation is hard to parallelize

# Unit Propagation: Example

$$\Gamma_{\text{unit}} := (\neg p_1 \vee \neg p_3 \vee p_4) \wedge (\neg p_1 \vee \neg p_2 \vee p_3) \wedge$$
$$(\neg p_1 \vee p_2) \wedge (p_1 \vee p_3 \vee p_6) \wedge (\neg p_1 \vee p_4 \vee \neg p_5) \wedge$$
$$(p_1 \vee \neg p_6) \wedge (p_4 \vee p_5 \vee p_6) \wedge (p_5 \vee \neg p_6)$$

# Unit Propagation: Example

$$\Gamma_{\text{unit}} := (\neg p_1 \vee \neg p_3 \vee p_4) \wedge (\neg p_1 \vee \neg p_2 \vee p_3) \wedge$$
$$(\neg p_1 \vee p_2) \wedge (p_1 \vee p_3 \vee p_6) \wedge (\neg p_1 \vee p_4 \vee \neg p_5) \wedge$$
$$(p_1 \vee \neg p_6) \wedge (p_4 \vee p_5 \vee p_6) \wedge (p_5 \vee \neg p_6)$$

$\tau = \{p_1 = \top\}$

# Unit Propagation: Example

$$\Gamma_{\text{unit}} := (\neg p_1 \vee \neg p_3 \vee p_4) \wedge (\neg p_1 \vee \neg p_2 \vee p_3) \wedge (\neg p_1 \vee p_2) \wedge (p_1 \vee p_3 \vee p_6) \wedge (\neg p_1 \vee p_4 \vee \neg p_5) \wedge (p_1 \vee \neg p_6) \wedge (p_4 \vee p_5 \vee p_6) \wedge (p_5 \vee \neg p_6)$$

$\tau = \{p_1 = \top,\ p_2 = \top\}$

# Unit Propagation: Example

$$\Gamma_{\text{unit}} := (\neg p_1 \vee \neg p_3 \vee p_4) \wedge (\neg p_1 \vee \neg p_2 \vee p_3) \wedge$$
$$(\neg p_1 \vee p_2) \wedge (p_1 \vee p_3 \vee p_6) \wedge (\neg p_1 \vee p_4 \vee \neg p_5) \wedge$$
$$(p_1 \vee \neg p_6) \wedge (p_4 \vee p_5 \vee p_6) \wedge (p_5 \vee \neg p_6)$$

$\tau = \{p_1 = \top,\ p_2 = \top,\ p_3 = \top\}$

# Unit Propagation: Example

$$\Gamma_{\text{unit}} := (\neg p_1 \vee \neg p_3 \vee p_4) \wedge (\neg p_1 \vee \neg p_2 \vee p_3) \wedge$$
$$(\neg p_1 \vee p_2) \wedge (p_1 \vee p_3 \vee p_6) \wedge (\neg p_1 \vee p_4 \vee \neg p_5) \wedge$$
$$(p_1 \vee \neg p_6) \wedge (p_4 \vee p_5 \vee p_6) \wedge (p_5 \vee \neg p_6)$$

$\tau = \{p_1 = \top,\ p_2 = \top,\ p_3 = \top,\ p_4 = \top\}$

# Unit Propagation: Proposition

### Proposition

*Unit propagation does not change the number of satisfying assignments*

True or false?

# Unit Propagation: Proposition

Proposition

*Unit propagation does not change the number of satisfying assignments*

True or false?

Proof.

True. Let formula $\Gamma$ have a unit clause $p$. All satisfying assignments of $\Gamma$ must assign $p$ to $\top$. Hence there cannot be a satisfying assignment with $p$ assigned to $\bot$. $\square$

# Unit Propagation: Resolution

The resolution rule allows for a formula containing the clauses $C \vee p$ and $\neg p \vee D$ to be extended by the clause $C \vee D$

$$\frac{C \vee p \qquad \neg p \vee D}{C \vee D}$$

# Unit Propagation: Resolution

The resolution rule allows for a formula containing the clauses $C \vee p$ and $\neg p \vee D$ to be extended by the clause $C \vee D$

$$\frac{C \vee p \qquad \neg p \vee D}{C \vee D}$$

Resolution proofs:

- A *resolution proof* is a sequence $C_1, \ldots, C_m$ of clauses.
- Every clause is either contained in the formula or derived from two earlier clauses via the *resolution rule*.
- $C_m$ is the *empty clause* (containing no literals): $\bot$.
- There exists a resolution proof for every unsatisfiable formula.

# Unit Propagation: Resolution Proofs

Example

$\Gamma := (\neg p \lor \neg q \lor r) \land (\neg r) \land (p \lor \neg q) \land (\neg s \lor q) \land (s)$

*Resolution proof*: $(\neg p \lor \neg q \lor r)$, $(\neg r)$, $(\neg p \lor \neg q)$, $(p \lor \neg q)$, $(\neg q)$, $(\neg s \lor q)$, $(\neg s)$, $(s)$, $\bot$

$$\dfrac{\dfrac{\neg s \lor q \qquad \dfrac{\dfrac{\neg p \lor \neg q \lor r \qquad \neg r}{\neg p \lor \neg q} \qquad p \lor \neg q}{\neg q}}{\neg s} \qquad s}{\bot}$$

# Unit Propagation: Relation to Resolution

Let $\Gamma$ be a formula. A clause $C$ is implied by $\Gamma$ via unit propagation (UP) if UP on $\Gamma \wedge \neg C$ results in a conflict.

Example

$$\begin{aligned}\Gamma \ := \ & (p \vee q \vee \neg r) \wedge (\neg p \vee \neg q \vee r) \wedge (q \vee r \vee \neg s) \ \wedge \\ & (\neg q \vee \neg r \vee s) \wedge (p \vee r \vee s) \wedge (\neg p \vee \neg r \vee \neg s)\end{aligned}$$

# Unit Propagation: Relation to Resolution

Let $\Gamma$ be a formula. A clause $C$ is implied by $\Gamma$ via unit propagation (UP) if UP on $\Gamma \wedge \neg C$ results in a conflict.

Example

$$\begin{aligned}\Gamma := \ & (p \vee q \vee \neg r) \wedge (\neg p \vee \neg q \vee r) \wedge (q \vee r \vee \neg s) \ \wedge \\ & (\neg q \vee \neg r \vee s) \wedge (p \vee r \vee s) \wedge (\neg p \vee \neg r \vee \neg s)\end{aligned}$$

| clause | $(p \vee q)$ |
| --- | --- |
| units | $\neg p \wedge \neg q$ |

# Unit Propagation: Relation to Resolution

Let $\Gamma$ be a formula. A clause $C$ is implied by $\Gamma$ via unit propagation (UP) if UP on $\Gamma \land \neg C$ results in a conflict.

Example

$$\begin{aligned}\Gamma \; := \; & (p \lor q \lor \neg r) \land (\neg p \lor \neg q \lor r) \land (q \lor r \lor \neg s) \; \land \\ & (\neg q \lor \neg r \lor s) \land (p \lor r \lor s) \land (\neg p \lor \neg r \lor \neg s)\end{aligned}$$

| clause | $(p \lor q)$ | $(p \lor q \lor \neg r)$ |
|---|---|---|
| units | $\neg p \land \neg q$ | $\neg r$ |

# Unit Propagation: Relation to Resolution

Let $\Gamma$ be a formula. A clause $C$ is implied by $\Gamma$ via unit propagation (UP) if UP on $\Gamma \wedge \neg C$ results in a conflict.

Example

$$\begin{aligned}\Gamma \;:=\; & (p \vee q \vee \neg r) \wedge (\neg p \vee \neg q \vee r) \wedge (q \vee r \vee \neg s) \;\wedge \\ & (\neg q \vee \neg r \vee s) \wedge (p \vee r \vee s) \wedge (\neg p \vee \neg r \vee \neg s)\end{aligned}$$

| clause | $(p \vee q)$ | $(p \vee q \vee \neg r)$ | $(q \vee r \vee \neg s)$ |
|---|---|---|---|
| units | $\neg p \wedge \neg q$ | $\neg r$ | $\neg s$ |

# Unit Propagation: Relation to Resolution

Let $\Gamma$ be a formula. A clause $C$ is implied by $\Gamma$ via unit propagation (UP) if UP on $\Gamma \wedge \neg C$ results in a conflict.

Example

$$\begin{aligned} \Gamma \ := \ & (p \vee q \vee \neg r) \wedge (\neg p \vee \neg q \vee r) \wedge (q \vee r \vee \neg s) \ \wedge \\ & (\neg q \vee \neg r \vee s) \wedge (p \vee r \vee s) \wedge (\neg p \vee \neg r \vee \neg s) \end{aligned}$$

| clause | $(p \vee q)$ | $(p \vee q \vee \neg r)$ | $(q \vee r \vee \neg s)$ | $(p \vee r \vee s)$ |
|---|---|---|---|---|
| units | $\neg p \wedge \neg q$ | $\neg r$ | $\neg s$ | $\perp$ |

# Unit Propagation: Relation to Resolution

Let $\Gamma$ be a formula. A clause $C$ is implied by $\Gamma$ via unit propagation (UP) if UP on $\Gamma \wedge \neg C$ results in a conflict.

## Example

$$\begin{aligned}\Gamma := \; & (p \vee q \vee \neg r) \wedge (\neg p \vee \neg q \vee r) \wedge (q \vee r \vee \neg s) \;\wedge \\ & (\neg q \vee \neg r \vee s) \wedge (p \vee r \vee s) \wedge (\neg p \vee \neg r \vee \neg s)\end{aligned}$$

| clause | $(p \vee q)$ | $(p \vee q \vee \neg r)$ | $(q \vee r \vee \neg s)$ | $(p \vee r \vee s)$ |
|---|---|---|---|---|
| units | $\neg p \wedge \neg q$ | $\neg r$ | $\neg s$ | $\perp$ |

$$\dfrac{\dfrac{(p \vee r \vee s) \qquad (q \vee r \vee \neg s)}{(p \vee q \vee r)} \qquad (p \vee q \vee \neg r)}{(p \vee q)}$$

Tseitin Transformation

Unit Propagation and Resolution

## Pure Literals and Autarkies

# Autarkies: Pure Literal Rule

A literal $l$ is pure in a CNF formula $\Gamma$ if the literal $\neg l$ does not occur in $\Gamma$.

# Autarkies: Pure Literal Rule

A literal $l$ is pure in a CNF formula $\Gamma$ if the literal $\neg l$ does not occur in $\Gamma$.

The pure literal rule simplifies a formula by making pure literals true.

# Autarkies: Pure Literal Rule

A literal $l$ is pure in a CNF formula $\Gamma$ if the literal $\neg l$ does not occur in $\Gamma$.

The pure literal rule simplifies a formula by making pure literals true.

Example

Consider the formula $\Gamma = (p \vee \neg q) \wedge (q \vee \neg r) \wedge (\neg q \vee r)$.
The literal $p$ is pure in $\Gamma$.
Let $\tau(p) = \top$. The pure literal rule will reduce $\Gamma$ to $[\![\Gamma]\!]_\tau$.
In other words, it will remove the first clause.

# Autarkies: Proposition

Proposition

*Assigning a pure literal to* $\top$ *does not change the number of satisfying assignments*

True or false?

# Autarkies: Proposition

**Proposition**

*Assigning a pure literal to $\top$ does not change the number of satisfying assignments*

True or false?

**Proof.**

False. A counterexample:
$\Gamma = (p \vee \neg q) \wedge (q \vee \neg r) \wedge (\neg q \vee r)$ has three satisfying assignments, while $[\![\Gamma]\!]_\tau$ with $\tau(p) = \top$ has only two. $\square$

# Autarkies: Definition

An autarky is a partial assignment that satisfies all clauses that are “touched” by the assignment:

- a pure literal is an autarky
- a satisfying assignment is an autarky
- “interesting” autarkies are between pure literals and satisfying assignments
- removing clauses that are satisfied by an autarky results in an equisatisfiable formula
- observe that for an autarky $\tau$ it holds that $[\![\Gamma]\!]_\tau \subseteq \Gamma$

# Autarkies: Example

$$\Gamma_{\text{unit}} := (\neg p_1 \vee \neg p_3 \vee p_4) \wedge (\neg p_1 \vee \neg p_2 \vee p_3) \wedge$$
$$(\neg p_1 \vee p_2) \wedge (p_1 \vee p_3 \vee p_6) \wedge (\neg p_1 \vee p_4 \vee \neg p_5) \wedge$$
$$(p_1 \vee \neg p_6) \wedge (p_4 \vee p_5 \vee p_6) \wedge (p_5 \vee \neg p_6)$$

# Autarkies: Example

$$\Gamma_{\text{unit}} := (\neg p_1 \vee \neg p_3 \vee p_4) \wedge (\neg p_1 \vee \neg p_2 \vee p_3) \wedge$$
$$(\neg p_1 \vee p_2) \wedge (p_1 \vee p_3 \vee p_6) \wedge (\neg p_1 \vee p_4 \vee \neg p_5) \wedge$$
$$(p_1 \vee \neg p_6) \wedge (p_4 \vee p_5 \vee p_6) \wedge (p_5 \vee \neg p_6)$$

$\tau = \{p_1 = \top\}$

# Autarkies: Example

$$\Gamma_{\text{unit}} := (\neg p_1 \vee \neg p_3 \vee p_4) \wedge (\neg p_1 \vee \neg p_2 \vee p_3) \wedge$$
$$(\neg p_1 \vee p_2) \wedge (p_1 \vee p_3 \vee p_6) \wedge (\neg p_1 \vee p_4 \vee \neg p_5) \wedge$$
$$(p_1 \vee \neg p_6) \wedge (p_4 \vee p_5 \vee p_6) \wedge (p_5 \vee \neg p_6)$$

$\tau = \{p_1 = \top,\ p_2 = \top\}$

# Autarkies: Example

$$\begin{gathered}\Gamma_{\text{unit}} := (\neg p_1 \vee \neg p_3 \vee p_4) \wedge (\neg p_1 \vee \neg p_2 \vee p_3) \wedge \\ (\neg p_1 \vee p_2) \wedge (p_1 \vee p_3 \vee p_6) \wedge (\neg p_1 \vee p_4 \vee \neg p_5) \wedge \\ (p_1 \vee \neg p_6) \wedge (p_4 \vee p_5 \vee p_6) \wedge (p_5 \vee \neg p_6)\end{gathered}$$

$\tau = \{p_1 = \top,\ p_2 = \top,\ p_3 = \top\}$

# Autarkies: Example

$$\Gamma_{\text{unit}} := (\neg p_1 \lor \neg p_3 \lor p_4) \land (\neg p_1 \lor \neg p_2 \lor p_3) \land \\ (\neg p_1 \lor p_2) \land (p_1 \lor p_3 \lor p_6) \land (\neg p_1 \lor p_4 \lor \neg p_5) \land \\ (p_1 \lor \neg p_6) \land (p_4 \lor p_5 \lor p_6) \land (p_5 \lor \neg p_6)$$

$$\tau = \{p_1 = \top,\ p_2 = \top,\ p_3 = \top,\ p_4 = \top\}$$

# Autarkies: Example

$$\Gamma_{\text{unit}} := (\neg p_1 \vee \neg p_3 \vee p_4) \wedge (\neg p_1 \vee \neg p_2 \vee p_3) \wedge$$
$$(\neg p_1 \vee p_2) \wedge (p_1 \vee p_3 \vee p_6) \wedge (\neg p_1 \vee p_4 \vee \neg p_5) \wedge$$
$$(p_1 \vee \neg p_6) \wedge (p_4 \vee p_5 \vee p_6) \wedge (p_5 \vee \neg p_6)$$

$$\tau = \{p_1 = \top,\ p_2 = \top,\ p_3 = \top,\ p_4 = \top\}$$

**The extended $\tau$ is an autarky for $\Gamma_{\text{unit}}$**

# Autarkies: Theorem

Theorem (Monien and Speckenmeyer, 1985)

*Let $\tau$ be an autarky for formula $\Gamma$. Then $\Gamma$ and $[\![\Gamma]\!]_\tau$ are equisatisfiable.*

Proof.

If $\Gamma$ is satisfiable, then since $[\![\Gamma]\!]_\tau \subseteq \Gamma$, we know that $[\![\Gamma]\!]_\tau$ is satisfiable as well.

Conversely, suppose $[\![\Gamma]\!]_\tau$ is satisfiable and let $\tau_1$ be an assignment that satisfies $[\![\Gamma]\!]_\tau$. We can assume that $\tau_1$ only assigns values to the variables of $[\![\Gamma]\!]_\tau$, which are distinct from the variables of $\tau$. Then the assignment $\tau_2$ which is the union of $\tau$ and $\tau_1$ satisfies $\Gamma$. $\square$