Time-space trade-offs in proof complexity
Lecture 1

Jakob Nordström

KTH Royal Institute of Technology

17th Estonian Winter School in Computer Science
Palmse, Estonia
February 26 – March 2, 2012

The Subject Matter of This Course (Broadly Speaking)

- What is a proof?
- Which (logical) statements have efficient proofs?
- How can we find such proofs? (Can we?)
- What are good methods of reasoning about logical statements?
- What are natural notions of “efficiency” of proofs?
- How are these notions related?

So What Is a Proof?

Claim: 25957 is the product of two primes.
True or false? What kind of proof would convince us?

- "I told you so. Just factor and check it yourself!"
  Not much of a proof.

- $25957 \equiv 1 \pmod{2}$ $\quad 25957 \equiv 0 \pmod{101}$
  $25957 \equiv 1 \pmod{3}$ $\quad 25957 \equiv 1 \pmod{103}$
  $25957 \equiv 2 \pmod{5}$ $\quad \vdots$
  $\vdots$ $\quad 25957 \equiv 0 \pmod{257}$
  $25957 \equiv 19 \pmod{99}$ $\quad \vdots$
  OK, but maybe even a bit of overkill.

- "$25957 = 101 \cdot 257$; check yourself that these are primes."

Key demand: A proof should be efficiently verifiable.

So What Is a Proof?

Claim: 25957 is the product of two primes.
True or false? What kind of proof would convince us?

- "I told you so. Just factor and check it yourself!"
  Not much of a proof.

- $25957 \equiv 1 \pmod{2}$ $25957 \equiv 0 \pmod{101}$
  $25957 \equiv 1 \pmod{3}$ $25957 \equiv 1 \pmod{103}$
  $25957 \equiv 2 \pmod{5}$ $\vdots$
  $\vdots$ $25957 \equiv 0 \pmod{257}$
  $25957 \equiv 19 \pmod{99}$ $\vdots$
  OK, but maybe even a bit of overkill.

- "$25957 = 101 \cdot 257$; check yourself that these are primes."

Key demand: A proof should be efficiently verifiable.

So What Is a Proof?

Claim: 25957 is the product of two primes.
True or false? What kind of proof would convince us?

- "I told you so. Just factor and check it yourself!"
  Not much of a proof.

- $$\begin{aligned} 25957 &\equiv 1 \pmod{2} \\ 25957 &\equiv 1 \pmod{3} \\ 25957 &\equiv 2 \pmod{5} \\ &\vdots \\ 25957 &\equiv 19 \pmod{99} \end{aligned} \qquad \begin{aligned} 25957 &\equiv 0 \pmod{101} \\ 25957 &\equiv 1 \pmod{103} \\ &\vdots \\ 25957 &\equiv 0 \pmod{257} \\ &\vdots \end{aligned}$$
  OK, but maybe even a bit of overkill.

- "$25957 = 101 \cdot 257$; check yourself that these are primes."

Key demand: A proof should be efficiently verifiable.

So What Is a Proof?

Claim: 25957 is the product of two primes.
True or false? What kind of proof would convince us?

- "I told you so. Just factor and check it yourself!"
  Not much of a proof.

- $25957 \equiv 1 \pmod{2}$ $\quad 25957 \equiv 0 \pmod{101}$
  $25957 \equiv 1 \pmod{3}$ $\quad 25957 \equiv 1 \pmod{103}$
  $25957 \equiv 2 \pmod{5}$ $\quad \vdots$
  $\vdots$ $\quad 25957 \equiv 0 \pmod{257}$
  $25957 \equiv 19 \pmod{99}$ $\quad \vdots$
  OK, but maybe even a bit of overkill.

- "$25957 = 101 \cdot 257$; check yourself that these are primes."

Key demand: A proof should be efficiently verifiable.

This Course (More Concrete Second Take)

- **Proof complexity** — study of proofs in different proof systems
- This lecture: general overview
- Later lectures: cover some recent results
- Disclaimer: heavy bias towards my own work
- Mostly give intuition and sketch proofs (but skip some details)
- Goal: good preparation for reading up on details on your own

Course Organization

- Four one-hour lectures
- Slides will be put online as we go at
  www.csc.kth.se/~jakobn/teaching/ewscs12/
- More information in survey paper in course binder
  (but not the most recent results that we cover)
- See www.csc.kth.se/~jakobn/teaching/proofcplx11
  and scribe notes there for full details of proofs et cetera

Proof system

Proof system for a language L (adapted from [Cook & Reckhow '79]):

Deterministic algorithm $\mathcal{P}(x, \pi)$ that runs in time polynomial in $|x|$ and $|\pi|$ such that

- for all $x \in L$ there is a string π (a proof) such that $\mathcal{P}(x, \pi) = 1$,
- for all $x \notin L$ it holds for all strings π that $\mathcal{P}(x, \pi) = 0$.

Think of $\mathcal{P}$ as "proof checker"
Note that proof π can be very large compared to x
Only have to achieve polynomial time in $|x| + |\pi|$

Propositional proof system: proof system for the language TAUT of all valid propositional logic formulas (or tautologies)

Example Propositional Proof System

Example (Truth table)

p	q	r	$(p \wedge (q \vee r)) \leftrightarrow ((p \wedge q) \vee (p \wedge r))$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Certainly polynomial-time checkable measured in "proof" size
Why does this not make us happy?

Example Propositional Proof System

Example (Truth table)

p	q	r	$(p \land (q \lor r)) \leftrightarrow ((p \land q) \lor (p \land r))$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Certainly polynomial-time checkable measured in "proof" size
Why does this not make us happy?

Proof System Complexity

Complexity $cplx(\mathcal{P})$ of a proof system $\mathcal{P}$:

Smallest $g : \mathbb{N} \mapsto \mathbb{N}$ such that $x \in L$ if and only if there is a proof π of size $|\pi| \leq g(|x|)$ such that $\mathcal{P}(x, \pi) = 1$.

If a proof system is of polynomial complexity, it is said to be polynomially bounded or p-bounded.

Example (Truth table continued)

Truth table is a propositional proof system, but of exponential complexity!

Proof System Complexity

Complexity $cplx(\mathcal{P})$ of a proof system $\mathcal{P}$:

Smallest $g : \mathbb{N} \mapsto \mathbb{N}$ such that $x \in L$ if and only if there is a proof π of size $|\pi| \leq g(|x|)$ such that $\mathcal{P}(x, \pi) = 1$.

If a proof system is of polynomial complexity, it is said to be polynomially bounded or p-bounded.

Example (Truth table continued)

Truth table is a propositional proof system, but of exponential complexity!

Proof systems and P vs. NP

Theorem (Cook & Reckhow '79)

NP $=$ co-NP *if and only if there exists a polynomially bounded propositional proof system.*

Proof.

NP *exactly* the set of languages with p-bounded proof systems

($\Rightarrow$) $\textsc{taut} \in$ co-NP since F is *not* a tautology iff $\neg F \in \textsc{sat}$.
If NP $=$ co-NP, then $\textsc{taut} \in$ NP has a p-bounded proof system by definition.

($\Leftarrow$) Suppose there exists a p-bounded proof system. Then $\textsc{taut} \in$ NP, and since $\textsc{taut}$ is complete for co-NP it follows that NP $=$ co-NP. $\square$

Proof systems and P vs. NP

Theorem (Cook & Reckhow '79)

NP = co-NP *if and only if there exists a polynomially bounded propositional proof system.*

Proof.

NP *exactly* the set of languages with p-bounded proof systems

($\Rightarrow$) $\textsc{taut} \in$ co-NP since F is *not* a tautology iff $\neg F \in \textsc{sat}$.
If NP = co-NP, then $\textsc{taut} \in$ NP has a p-bounded proof system by definition.

($\Leftarrow$) Suppose there exists a p-bounded proof system. Then $\textsc{taut} \in$ NP, and since $\textsc{taut}$ is complete for co-NP it follows that NP = co-NP. □

Proof systems and P vs. NP

Theorem (Cook & Reckhow '79)

NP = co-NP *if and only if there exists a polynomially bounded propositional proof system.*

Proof.

NP *exactly* the set of languages with p-bounded proof systems

($\Rightarrow$) $\textsc{taut} \in$ co-NP since F is *not* a tautology iff $\neg F \in \textsc{sat}$.
If NP = co-NP, then $\textsc{taut} \in$ NP has a p-bounded proof system by definition.

($\Leftarrow$) Suppose there exists a p-bounded proof system. Then $\textsc{taut} \in$ NP, and since $\textsc{taut}$ is complete for co-NP it follows that NP = co-NP. □

Proof systems and P vs. NP

Theorem (Cook & Reckhow '79)

NP $=$ co-NP *if and only if there exists a polynomially bounded propositional proof system.*

Proof.

NP *exactly* the set of languages with p-bounded proof systems

($\Rightarrow$) $\textsc{taut} \in$ co-NP since F is *not* a tautology iff $\neg F \in \textsc{sat}$.
If NP $=$ co-NP, then $\textsc{taut} \in$ NP has a p-bounded proof system by definition.

($\Leftarrow$) Suppose there exists a p-bounded proof system. Then $\textsc{taut} \in$ NP, and since $\textsc{taut}$ is complete for co-NP it follows that NP $=$ co-NP. □

Polynomial Simulation

The convential wisdom is that NP $\neq$ co-NP
Seems that proof of this is lightyears away
(Would imply P $\neq$ NP as a corollary)

Reason 1 for proof complexity: approach this distant goal by studying successively stronger proof systems and relating their strengths

Definition (p-simulation)

$\mathcal{P}_1$ polynomially simulates, or p-simulates, $\mathcal{P}_2$ if there exists a polynomial-time computable function f such that for all $F \in \textsc{taut}$ it holds that $\mathcal{P}_2(F, \pi) = 1$ iff $\mathcal{P}_1(F, f(\pi)) = 1$.

Polynomial Simulation

The convential wisdom is that NP $\neq$ co-NP
Seems that proof of this is lightyears away
(Would imply P $\neq$ NP as a corollary)

Reason 1 for proof complexity: approach this distant goal by studying successively stronger proof systems and relating their strengths

Definition (p-simulation)

$\mathcal{P}_1$ polynomially simulates, or p-simulates, $\mathcal{P}_2$ if there exists a polynomial-time computable function f such that for all $F \in \text{TAUT}$ it holds that $\mathcal{P}_2(F, \pi) = 1$ iff $\mathcal{P}_1(F, f(\pi)) = 1$.

Polynomial Equivalence

Definition (p-equivalence)

Two propositional proof systems $\mathcal{P}_1$ and $\mathcal{P}_2$ are polynomially equivalent, or p-equivalent, if each proof system p-simulates the other.

If $\mathcal{P}_1$ p-simulates $\mathcal{P}_2$ but $\mathcal{P}_2$ does not p-simulate $\mathcal{P}_1$, then $\mathcal{P}_1$ is strictly stronger than $\mathcal{P}_2$

Lots of results relating strength of different propositional proof systems

But not focus of this course (though might touch briefly on one example)

A Fundamental Theoretical Problem...

The constructive version of the question:

Problem

Given a propositional logic formula F, can we decide efficiently whether it is true no matter how we assign values to its variables?

TAUT: Fundamental problem in theoretical computer science ever since Stephen Cook's NP-completeness paper in 1971

(And significance realized much earlier — cf. Gödel's letter in 1956)

These days recognized as one of the main challenges for all of mathematics — one of the million dollar "Millennium Problems"

A Fundamental Theoretical Problem. . .

The constructive version of the question:

Problem

Given a propositional logic formula F, can we decide efficiently whether it is true no matter how we assign values to its variables?

TAUT: Fundamental problem in theoretical computer science ever since Stephen Cook's NP-completeness paper in 1971

(And significance realized much earlier — cf. Gödel's letter in 1956)

These days recognized as one of the main challenges for all of mathematics — one of the million dollar "Millennium Problems"

. . . with Huge Practical Implications

- All known algorithms run in exponential time in worst case
- But enormous progress on applied computer programs last 10-15 years
- These so-called SAT solvers are routinely deployed to solve large-scale real-world problems with millions of variables
- Used in e.g. hardware verification, software testing, software package management, artificial intelligence, cryptography, bioinformatics, . . .
- But we also know small example formulas with only hundreds of variables that trip up even state-of-the-art SAT solvers

Automated Theorem Proving or SAT Solving

Reason 2 for proof complexity: understand proof systems used for solving formulas occurring in "real-world applications"

- Study proof systems used by SAT solvers
- Model actual methods of reasoning used by SAT solvers as "refinements" (subsystems) of these systems
- Prove upper and lower bounds in these systems
- Try to explain or predict theoretically what happens in practice

This course:

- Focus on proof systems used for SAT solving (resolution & polynomial calculus; won't get to cutting planes)
- Pure proof complexity results; no "low-level modelling"

Automated Theorem Proving or SAT Solving

Reason 2 for proof complexity: understand proof systems used for solving formulas occurring in "real-world applications"

- Study proof systems used by SAT solvers
- Model actual methods of reasoning used by SAT solvers as "refinements" (subsystems) of these systems
- Prove upper and lower bounds in these systems
- Try to explain or predict theoretically what happens in practice

This course:

- Focus on proof systems used for SAT solving (resolution & polynomial calculus; won't get to cutting planes)
- Pure proof complexity results; no "low-level modelling"

Potential and Limitations of Mathematical Reasoning

Reason 3 for proof complexity: understand how deep / hard various mathematical truths are

- Look at logic encoding of various mathematical truths (e.g. combinatorial principles)
- Determine how strong proof systems are needed to provide efficient proofs
- Tells us how powerful mathematical tools are needed for establishing such statements

Fascinating area, but this course will not go into this at all

Transforming Tautologies to Unsatisfiable CNFs

Any propositional logic formula F can be converted to formula F' in conjunctive normal form (CNF) such that

- F' only linearly larger than F
- F' unsatisfiable iff F tautology

Idea [Tseitin '68]:

- Introduce new variable x_G for each subformula $G \doteq H_1 \circ H_2$ in F, $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$
- Translate G to set of disjunctive clauses $Cl(G)$ which enforces that truth value of x_G is computed correctly given x_{H_1} and x_{H_2}

Transforming Tautologies to Unsatisfiable CNFs

Any propositional logic formula F can be converted to formula F' in conjunctive normal form (CNF) such that

- F' only linearly larger than F
- F' unsatisfiable iff F tautology

Idea [Tseitin '68]:

- Introduce new variable x_G for each subformula $G \doteq H_1 \circ H_2$ in F, $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$
- Translate G to set of disjunctive clauses $Cl(G)$ which enforces that truth value of x_G is computed correctly given x_{H_1} and x_{H_2}

Sketch of Transformation

Two examples for $\lor$ and $\rightarrow$ ($\land$ and $\leftrightarrow$ are analogous):

$$G \equiv H_1 \lor H_2 : \qquad Cl(G) := \begin{aligned}&\left(\neg x_G \lor x_{H_1} \lor x_{H_2}\right)\\ \land\ &\left(x_G \lor \neg x_{H_1}\right)\\ \land\ &\left(x_G \lor \neg x_{H_2}\right)\end{aligned}$$

$$G \equiv H_1 \rightarrow H_2 : \qquad Cl(G) := \begin{aligned}&\left(\neg x_G \lor \neg x_{H_1} \lor x_{H_2}\right)\\ \land\ &\left(x_G \lor x_{H_1}\right)\\ \land\ &\left(x_G \lor \neg x_{H_2}\right)\end{aligned}$$

- Finally, add clause $\neg x_F$

Proof Systems for Refuting Unsatisfiable CNFs

- Easy to verify that constructed CNF formula F' is unsatisfiable iff F is a tautology
- So any sound and complete proof system which produces refutations of formulas in conjunctive normal form can be used as a propositional proof system
- From now on and for the rest of this course, we will discuss only such proof systems

Some Notation and Terminology

- Literal a: variable x or its negation $\overline{x}$ (rather than $\neg x$)
- Let $\overline{\overline{x}} = x$
- Clause $C = a_1 \lor \ldots \lor a_k$: set of literals
  At most k literals: k-clause
- CNF formula $F = C_1 \land \ldots \land C_m$: set of clauses
  k-CNF formula: CNF formula consisting of k-clauses
- $Vars(\cdot)$: set of variables in clause or formula
  $Lit(\cdot)$: set of literals in clause or formula
- $F \vDash D$: semantical implication, $\alpha(F)$ true $\Rightarrow \alpha(D)$ true for all truth value assignments α
- $[n] = \{1, 2, \ldots, n\}$

Sequential Proof Systems

A proof system $\mathcal{P}$ is sequential if a proof π in $\mathcal{P}$ is a

- sequence of lines $\pi = \{L_1, \ldots, L_\tau\}$
- of some prescribed syntactic form (depending on the proof system in question)
- where each line is derived from previous lines by one of a finite set of allowed inference rules

(This will become clearer when we get some examples)

A proof of an unsatisfiable CNF formula refutes the formula
Use terms "proof" and "refutation" interchangeably

Complexity Measures (High-level Intuition)

View a proof as

- non-deterministic Turing machine computation,
- special read-only input tape from which the clauses of F (the axioms) can be downloaded
- working memory where all derivation steps are made

Interested in measuring

- size/length of proofs
- space of proofs

Size of a proof $\approx$ time of the computation

Space $\approx$ memory consumption (how much to remember simultaneously)

Length and Space (Generic Definitions)

Definition (Length)

Length $L(\pi)$ of refutation $\pi = \#$ derivation steps
($\approx \#$ lines counted with repetitions)

Length of refuting F in $\mathcal{P}$
$L_{\mathcal{P}}(F \vdash \bot)$ = minimal length of any refutation

Definition (Space)

Space $Sp(\pi)$ of refutation $\pi =$ "size" of largest configuration in π

Space of refuting F in $\mathcal{P}$
$Sp_{\mathcal{P}}(F \vdash \bot)$ = minimal space of any refutation

These definitions to be made more precise for specific proof systems

Resolution

Resolution proof system usually attributed to [Blake '37]
Used in connection with SAT solving in 1960s [DP60,DLL62,Rob65]

Lines in refutation are disjunctive clauses

Just one inference rule, the resolution rule:

$$\frac{B \vee x \quad C \vee \overline{x}}{B \vee C}$$

$B \vee C$ is the resolvent of $B \vee x$ and $C \vee \overline{x}$

Observation

If F is a satisfiable CNF formula and D is derived from clauses $C_1, C_2 \in F$ by the resolution rule, then $F \wedge D$ is satisfiable.

Prove F unsatisfiable by deriving the unsatisfiable empty clause $\perp$ (the clause with no literals) from F by resolution

Resolution Sound and Complete

Resolution is sound and implicationally complete.

Sound If there is a resolution derivation $\pi : F \vdash A$ then $F \vDash A$

Complete If $F \vDash A$ then there is a resolution derivation $\pi : F \vdash A'$ for some $A' \subseteq A$.

In particular:

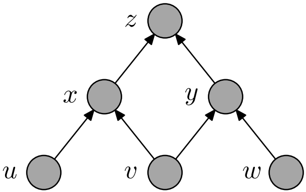
$$F \text{ is unsatisfiable} \Leftrightarrow \exists \text{ resolution refutation of } F$$

Resolution as a Sequential Proof System

- Goal: Refute given CNF formula (i.e., prove it is unsatisfiable)
- Proof system operates with disjunctive clauses
- Proof/refutation is "presented on blackboard"
- Derivation steps:
  - Write down clauses of CNF formula being refuted (axiom clauses)
  - Infer new clauses by resolution rule
  - Erase clauses that are not currently needed (to save space on blackboard)
- Refutation ends when empty clause $\perp$ is derived

Example CNF Formula

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

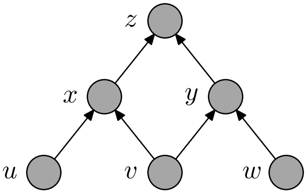


Defined in terms of directed acyclic graph (DAG):

- source vertices true
- truth propagates upwards
- but sink vertex is false

Example CNF Formula

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

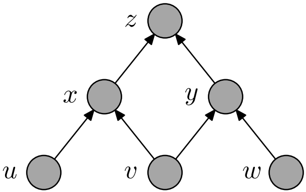


Defined in terms of directed acyclic graph (DAG):

- source vertices true
- truth propagates upwards
- but sink vertex is false

Example CNF Formula

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

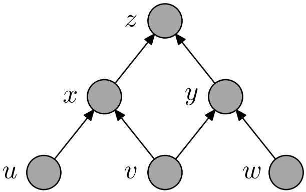


Defined in terms of directed acyclic graph (DAG):

- source vertices true
- truth propagates upwards
- but sink vertex is false

Example CNF Formula

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$



Defined in terms of directed acyclic graph (DAG):

- source vertices true
- truth propagates upwards
- but sink vertex is false

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	0
max # lines on board	0
max # literals on board	0

Can download axioms, erase used clauses or infer new clauses by resolution rule

$$\frac{B \lor x \quad C \lor \overline{x}}{B \lor C}$$

(but only from clauses on board!)

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	1
max # lines on board	1
max # literals on board	1

u

Download axiom 1: u

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	2
max # lines on board	2
max # literals on board	2

u
v

Download axiom 1: u
Download axiom 2: v

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	3
max # lines on board	3
max # literals on board	5

u
v
$\overline{u} \lor \overline{v} \lor x$

Download axiom 1: u
Download axiom 2: v
Download axiom 4: $\overline{u} \lor \overline{v} \lor x$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	3
max # lines on board	3
max # literals on board	5

u
v
$\overline{u} \lor \overline{v} \lor x$

Download axiom 1: u
Download axiom 2: v
Download axiom 4: $\overline{u} \lor \overline{v} \lor x$
Infer $\overline{v} \lor x$ from
u and $\overline{u} \lor \overline{v} \lor x$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	4
max # lines on board	4
max # literals on board	7

u
v
$\overline{u} \lor \overline{v} \lor x$
$\overline{v} \lor x$

Download axiom 1: u
Download axiom 2: v
Download axiom 4: $\overline{u} \lor \overline{v} \lor x$
Infer $\overline{v} \lor x$ from
u and $\overline{u} \lor \overline{v} \lor x$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	4
max # lines on board	4
max # literals on board	7

u
v
$\overline{u} \lor \overline{v} \lor x$
$\overline{v} \lor x$

Download axiom 2: v
Download axiom 4: $\overline{u} \lor \overline{v} \lor x$
Infer $\overline{v} \lor x$ from
u and $\overline{u} \lor \overline{v} \lor x$
Erase the clause $\overline{u} \lor \overline{v} \lor x$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \vee \overline{v} \vee x$
5. $\overline{v} \vee \overline{w} \vee y$
6. $\overline{x} \vee \overline{y} \vee z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	4
max # lines on board	4
max # literals on board	7

u
v
$\overline{v} \vee x$

Download axiom 2: v
Download axiom 4: $\overline{u} \vee \overline{v} \vee x$
Infer $\overline{v} \vee x$ from
u and $\overline{u} \vee \overline{v} \vee x$
Erase the clause $\overline{u} \vee \overline{v} \vee x$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	4
max # lines on board	4
max # literals on board	7

u
v
$\overline{v} \lor x$

Download axiom 4: $\overline{u} \lor \overline{v} \lor x$
Infer $\overline{v} \lor x$ from
u and $\overline{u} \lor \overline{v} \lor x$
Erase the clause $\overline{u} \lor \overline{v} \lor x$
Erase the clause u

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	4
max # lines on board	4
max # literals on board	7

v

$\overline{v} \lor x$

Download axiom 4: $\overline{u} \lor \overline{v} \lor x$
Infer $\overline{v} \lor x$ from
u and $\overline{u} \lor \overline{v} \lor x$
Erase the clause $\overline{u} \lor \overline{v} \lor x$
Erase the clause u

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	4
max # lines on board	4
max # literals on board	7

v
$\overline{v} \lor x$

u and $\overline{u} \lor \overline{v} \lor x$
Erase the clause $\overline{u} \lor \overline{v} \lor x$
Erase the clause u
Infer x from
v and $\overline{v} \lor x$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	5
max # lines on board	4
max # literals on board	7

v
$\overline{v} \lor x$
x

u and $\overline{u} \lor \overline{v} \lor x$
Erase the clause $\overline{u} \lor \overline{v} \lor x$
Erase the clause u
Infer x from
v and $\overline{v} \lor x$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	5
max # lines on board	4
max # literals on board	7

Board:

v

$\overline{v} \lor x$

x

Erase the clause $\overline{u} \lor \overline{v} \lor x$
Erase the clause u
Infer x from
v and $\overline{v} \lor x$
Erase the clause $\overline{v} \lor x$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	5
max # lines on board	4
max # literals on board	7

Board:

v

x

Erase the clause $\overline{u} \lor \overline{v} \lor x$
Erase the clause u
Infer x from
  v and $\overline{v} \lor x$
Erase the clause $\overline{v} \lor x$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	5
max # lines on board	4
max # literals on board	7

Board:

v

x

Erase the clause u

Infer x from

v and $\overline{v} \lor x$

Erase the clause $\overline{v} \lor x$

Erase the clause v

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	5
max # lines on board	4
max # literals on board	7

x

Erase the clause u
Infer x from
v and $\overline{v} \lor x$
Erase the clause $\overline{v} \lor x$
Erase the clause v

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	6
max # lines on board	4
max # literals on board	7

x

$\overline{x} \lor \overline{y} \lor z$

Infer x from

v and $\overline{v} \lor x$

Erase the clause $\overline{v} \lor x$

Erase the clause v

Download axiom 6: $\overline{x} \lor \overline{y} \lor z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \vee \overline{v} \vee x$
5. $\overline{v} \vee \overline{w} \vee y$
6. $\overline{x} \vee \overline{y} \vee z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	6
max # lines on board	4
max # literals on board	7

x

$\overline{x} \vee \overline{y} \vee z$

Erase the clause $\overline{v} \vee x$
Erase the clause v
Download axiom 6: $\overline{x} \vee \overline{y} \vee z$
Infer $\overline{y} \vee z$ from
x and $\overline{x} \vee \overline{y} \vee z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	7
max # lines on board	4
max # literals on board	7

x

$\overline{x} \lor \overline{y} \lor z$

$\overline{y} \lor z$

Erase the clause $\overline{v} \lor x$

Erase the clause v

Download axiom 6: $\overline{x} \lor \overline{y} \lor z$

Infer $\overline{y} \lor z$ from
x and $\overline{x} \lor \overline{y} \lor z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	7
max # lines on board	4
max # literals on board	7

Blackboard:

x

$\overline{x} \lor \overline{y} \lor z$

$\overline{y} \lor z$

Erase the clause v

Download axiom 6: $\overline{x} \lor \overline{y} \lor z$

Infer $\overline{y} \lor z$ from

x and $\overline{x} \lor \overline{y} \lor z$

Erase the clause $\overline{x} \lor \overline{y} \lor z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \vee \overline{v} \vee x$
5. $\overline{v} \vee \overline{w} \vee y$
6. $\overline{x} \vee \overline{y} \vee z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	7
max # lines on board	4
max # literals on board	7

x
$\overline{y} \vee z$

Erase the clause v
Download axiom 6: $\overline{x} \vee \overline{y} \vee z$
Infer $\overline{y} \vee z$ from
x and $\overline{x} \vee \overline{y} \vee z$
Erase the clause $\overline{x} \vee \overline{y} \vee z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	7
max # lines on board	4
max # literals on board	7

x

$\overline{y} \lor z$

Download axiom 6: $\overline{x} \lor \overline{y} \lor z$
Infer $\overline{y} \lor z$ from
x and $\overline{x} \lor \overline{y} \lor z$
Erase the clause $\overline{x} \lor \overline{y} \lor z$
Erase the clause x

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	7
max # lines on board	4
max # literals on board	7

$\overline{y} \lor z$

Download axiom 6: $\overline{x} \lor \overline{y} \lor z$
Infer $\overline{y} \lor z$ from
x and $\overline{x} \lor \overline{y} \lor z$
Erase the clause $\overline{x} \lor \overline{y} \lor z$
Erase the clause x

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	8
max # lines on board	4
max # literals on board	7

$\overline{y} \lor z$
$\overline{v} \lor \overline{w} \lor y$

Infer $\overline{y} \lor z$ from
x and $\overline{x} \lor \overline{y} \lor z$
Erase the clause $\overline{x} \lor \overline{y} \lor z$
Erase the clause x
Download axiom 5: $\overline{v} \lor \overline{w} \lor y$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	8
max # lines on board	4
max # literals on board	7

$\overline{y} \lor z$
$\overline{v} \lor \overline{w} \lor y$

Erase the clause $\overline{x} \lor \overline{y} \lor z$
Erase the clause x
Download axiom 5: $\overline{v} \lor \overline{w} \lor y$
Infer $\overline{v} \lor \overline{w} \lor z$ from
$\overline{y} \lor z$ and $\overline{v} \lor \overline{w} \lor y$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	9
max # lines on board	4
max # literals on board	8

$\overline{y} \lor z$
$\overline{v} \lor \overline{w} \lor y$
$\overline{v} \lor \overline{w} \lor z$

Erase the clause $\overline{x} \lor \overline{y} \lor z$
Erase the clause x
Download axiom 5: $\overline{v} \lor \overline{w} \lor y$
Infer $\overline{v} \lor \overline{w} \lor z$ from
$\overline{y} \lor z$ and $\overline{v} \lor \overline{w} \lor y$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	9
max # lines on board	4
max # literals on board	8

$\overline{y} \lor z$
$\overline{v} \lor \overline{w} \lor y$
$\overline{v} \lor \overline{w} \lor z$

Erase the clause x
Download axiom 5: $\overline{v} \lor \overline{w} \lor y$
Infer $\overline{v} \lor \overline{w} \lor z$ from
$\overline{y} \lor z$ and $\overline{v} \lor \overline{w} \lor y$
Erase the clause $\overline{v} \lor \overline{w} \lor y$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	9
max # lines on board	4
max # literals on board	8

$\overline{y} \lor z$
$\overline{v} \lor \overline{w} \lor z$

Erase the clause x
Download axiom 5: $\overline{v} \lor \overline{w} \lor y$
Infer $\overline{v} \lor \overline{w} \lor z$ from
$\overline{y} \lor z$ and $\overline{v} \lor \overline{w} \lor y$
Erase the clause $\overline{v} \lor \overline{w} \lor y$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \vee \overline{v} \vee x$
5. $\overline{v} \vee \overline{w} \vee y$
6. $\overline{x} \vee \overline{y} \vee z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	9
max # lines on board	4
max # literals on board	8

$\overline{y} \vee z$
$\overline{v} \vee \overline{w} \vee z$

Download axiom 5: $\overline{v} \vee \overline{w} \vee y$
Infer $\overline{v} \vee \overline{w} \vee z$ from
$\overline{y} \vee z$ and $\overline{v} \vee \overline{w} \vee y$
Erase the clause $\overline{v} \vee \overline{w} \vee y$
Erase the clause $\overline{y} \vee z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	9
max # lines on board	4
max # literals on board	8

$\overline{v} \lor \overline{w} \lor z$

Download axiom 5: $\overline{v} \lor \overline{w} \lor y$
Infer $\overline{v} \lor \overline{w} \lor z$ from
$\overline{y} \lor z$ and $\overline{v} \lor \overline{w} \lor y$
Erase the clause $\overline{v} \lor \overline{w} \lor y$
Erase the clause $\overline{y} \lor z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	10
max # lines on board	4
max # literals on board	8

$\overline{v} \lor \overline{w} \lor z$

v

Infer $\overline{v} \lor \overline{w} \lor z$ from
$\overline{y} \lor z$ and $\overline{v} \lor \overline{w} \lor y$
Erase the clause $\overline{v} \lor \overline{w} \lor y$
Erase the clause $\overline{y} \lor z$
Download axiom 2: v

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	11
max # lines on board	4
max # literals on board	8

$\overline{v} \lor \overline{w} \lor z$

v

w

$\overline{y} \lor z$ and $\overline{v} \lor \overline{w} \lor y$

Erase the clause $\overline{v} \lor \overline{w} \lor y$

Erase the clause $\overline{y} \lor z$

Download axiom 2: v

Download axiom 3: w

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	12
max # lines on board	4
max # literals on board	8

$\overline{v} \lor \overline{w} \lor z$
v
w
$\overline{z}$

Erase the clause $\overline{v} \lor \overline{w} \lor y$
Erase the clause $\overline{y} \lor z$
Download axiom 2: v
Download axiom 3: w
Download axiom 7: $\overline{z}$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	12
max # lines on board	4
max # literals on board	8

$\overline{v} \lor \overline{w} \lor z$
v
w
$\overline{z}$

Download axiom 2: v
Download axiom 3: w
Download axiom 7: $\overline{z}$
Infer $\overline{w} \lor z$ from
v and $\overline{v} \lor \overline{w} \lor z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	13
max # lines on board	5
max # literals on board	8

$\overline{v} \lor \overline{w} \lor z$
v
w
$\overline{z}$
$\overline{w} \lor z$

Download axiom 2: v
Download axiom 3: w
Download axiom 7: $\overline{z}$
Infer $\overline{w} \lor z$ from
v and $\overline{v} \lor \overline{w} \lor z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	13
max # lines on board	5
max # literals on board	8

$\overline{v} \lor \overline{w} \lor z$
v
w
$\overline{z}$
$\overline{w} \lor z$

Download axiom 3: w
Download axiom 7: $\overline{z}$
Infer $\overline{w} \lor z$ from
v and $\overline{v} \lor \overline{w} \lor z$
Erase the clause v

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	13
max # lines on board	5
max # literals on board	8

$\overline{v} \lor \overline{w} \lor z$
w
$\overline{z}$
$\overline{w} \lor z$

Download axiom 3: w
Download axiom 7: $\overline{z}$
Infer $\overline{w} \lor z$ from
v and $\overline{v} \lor \overline{w} \lor z$
Erase the clause v

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	13
max # lines on board	5
max # literals on board	8

$\overline{v} \lor \overline{w} \lor z$
w
$\overline{z}$
$\overline{w} \lor z$

Download axiom 7: $\overline{z}$
Infer $\overline{w} \lor z$ from
v and $\overline{v} \lor \overline{w} \lor z$
Erase the clause v
Erase the clause $\overline{v} \lor \overline{w} \lor z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \vee \overline{v} \vee x$
5. $\overline{v} \vee \overline{w} \vee y$
6. $\overline{x} \vee \overline{y} \vee z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	13
max # lines on board	5
max # literals on board	8

w
$\overline{z}$
$\overline{w} \vee z$

Download axiom 7: $\overline{z}$
Infer $\overline{w} \vee z$ from
v and $\overline{v} \vee \overline{w} \vee z$
Erase the clause v
Erase the clause $\overline{v} \vee \overline{w} \vee z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	13
max # lines on board	5
max # literals on board	8

w

$\overline{z}$

$\overline{w} \lor z$

v and $\overline{v} \lor \overline{w} \lor z$

Erase the clause v

Erase the clause $\overline{v} \lor \overline{w} \lor z$

Infer z from

w and $\overline{w} \lor z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	14
max # lines on board	5
max # literals on board	8

w
$\overline{z}$
$\overline{w} \lor z$
z

v and $\overline{v} \lor \overline{w} \lor z$
Erase the clause v
Erase the clause $\overline{v} \lor \overline{w} \lor z$
Infer z from
w and $\overline{w} \lor z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	14
max # lines on board	5
max # literals on board	8

w
$\overline{z}$
$\overline{w} \lor z$
z

Erase the clause v
Erase the clause $\overline{v} \lor \overline{w} \lor z$
Infer z from
w and $\overline{w} \lor z$
Erase the clause w

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	14
max # lines on board	5
max # literals on board	8

$\overline{z}$
$\overline{w} \lor z$
z

Erase the clause v
Erase the clause $\overline{v} \lor \overline{w} \lor z$
Infer z from
w and $\overline{w} \lor z$
Erase the clause w

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	14
max # lines on board	5
max # literals on board	8

$\overline{z}$

$\overline{w} \lor z$

z

Erase the clause $\overline{v} \lor \overline{w} \lor z$

Infer z from

w and $\overline{w} \lor z$

Erase the clause w

Erase the clause $\overline{w} \lor z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	14
max # lines on board	5
max # literals on board	8

$\overline{z}$

z

Erase the clause $\overline{v} \lor \overline{w} \lor z$

Infer z from

w and $\overline{w} \lor z$

Erase the clause w

Erase the clause $\overline{w} \lor z$

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	14
max # lines on board	5
max # literals on board	8

$\overline{z}$

z

w and $\overline{w} \lor z$
Erase the clause w
Erase the clause $\overline{w} \lor z$
Infer $\bot$ from
$\overline{z}$ and z

Example Resolution Refutation

1. u
2. v
3. w
4. $\overline{u} \lor \overline{v} \lor x$
5. $\overline{v} \lor \overline{w} \lor y$
6. $\overline{x} \lor \overline{y} \lor z$
7. $\overline{z}$

Blackboard bookkeeping	
total # clauses on board	15
max # lines on board	5
max # literals on board	8

$\overline{z}$

z

$\bot$

w and $\overline{w} \lor z$
Erase the clause w
Erase the clause $\overline{w} \lor z$
Infer $\bot$ from
$\overline{z}$ and z

Complexity Measures of Interest: Length and Space

- Length: Lower bound on time for proof search algorithm
- Space: Lower bound on memory for proof search algorithm

Length

\# clauses written on blackboard counted with repetitions

Space

Somewhat less straightforward — several ways of measuring

$$x$$
$$\overline{y} \lor z$$
$$\overline{v} \lor \overline{w} \lor y$$

Clause space: 3

Total space: 6

Complexity Measures of Interest: Length and Space

- Length: Lower bound on time for proof search algorithm
- Space: Lower bound on memory for proof search algorithm

Length
$\#$ clauses written on blackboard counted with repetitions

Space
Somewhat less straightforward — several ways of measuring

x
$\overline{y} \vee z$
$\overline{v} \vee \overline{w} \vee y$

Clause space: 3

Total space: 6

Complexity Measures of Interest: Length and Space

- Length: Lower bound on time for proof search algorithm
- Space: Lower bound on memory for proof search algorithm

Length

$\#$ clauses written on blackboard counted with repetitions

Space

Somewhat less straightforward — several ways of measuring

x
$\overline{y} \lor z$
$\overline{v} \lor \overline{w} \lor y$

Clause space: 3

Total space: 6

Complexity Measures of Interest: Length and Space

- Length: Lower bound on time for proof search algorithm
- Space: Lower bound on memory for proof search algorithm

Length

\# clauses written on blackboard counted with repetitions

Space

Somewhat less straightforward — several ways of measuring

1. x
2. $\overline{y} \lor z$
3. $\overline{v} \lor \overline{w} \lor y$

Clause space: 3

Total space: 6

Complexity Measures of Interest: Length and Space

- Length: Lower bound on time for proof search algorithm
- Space: Lower bound on memory for proof search algorithm

Length

clauses written on blackboard counted with repetitions

Space

Somewhat less straightforward — several ways of measuring

$$x^{1}$$
$$\overline{y}^{2} \lor z^{3}$$
$$\overline{v}^{4} \lor \overline{w}^{5} \lor y^{6}$$

Clause space: 3

Total space: 6

Complexity Measures of Interest: Length and Space

- Length: Lower bound on time for proof search algorithm
- Space: Lower bound on memory for proof search algorithm

Length

clauses written on blackboard counted with repetitions
(in our example resolution refutation 15)

Space

Somewhat less straightforward — several ways of measuring

$$x$$
$$\overline{y} \lor z$$
$$\overline{v} \lor \overline{w} \lor y$$

Clause space: 3
(in our refutation 5)
Total space: 6
(in our refutation 8)

Cutting Planes: Informal Description

- Geometric proof system introduced by [Cook, Coullard & Turán '87]
- Translate clauses to linear inequalities for real variables in $[0,1]$
- For instance, $x \vee y \vee \overline{z}$ gets translated to $x + y + (1 - z) \geq 1$, i.e., $x + y - z \geq 0$
- Manipulate linear inequalities to derive contradiction $0 \geq 1$

Cutting Planes: Inference Rules

Lines in cutting planes (CP) refutation: linear inequalities with integer coefficients

Derivation rules:

Variable axioms $\dfrac{}{x \geq 0}$ and $\dfrac{}{-x \geq -1}$ for all variables x

Addition $$\frac{\sum a_i x_i \geq A \qquad \sum b_i x_i \geq B}{\sum (a_i + b_i) x_i \geq A + B}$$

Multiplication $\dfrac{\sum a_i x_i \geq A}{\sum c a_i x_i \geq cA}$ for a positive integer c

Division $\dfrac{\sum c a_i x_i \geq A}{\sum a_i x_i \geq \lceil A/c \rceil}$ for a positive integer c

CP-refutation: derivation of inequality $0 \geq 1$

Cutting Planes Measures

Length
derivation steps

Size
symbols needed to represent proof (coefficients can be huge)

Line space
Linear inequalities in any configuration
(Analogue of clause space)

Total space
Total # variables in configuration counted with repetitions
+ log of coefficients

Polynomial Calculus

- Algrebraic system introduced by [Clegg, Edmonds & Impagliazzo '96] under the name of "Gröbner proof system"
- Clauses are interpreted as multilinear polynomial equations
- Here, natural to flip convention and think of 0 as true and 1 as false
- For instance, clause $x \lor y \lor \overline{z}$ gets translated to $xy(1 - z) = 0$ or $xy - xyz = 0$
- Derive contradiction by showing that there is no common root for the polynomial equations corresponding to all the clauses

Polynomial Calculus: Inference Rules

Lines in polynomial calculus (PC) refutation: multivariate polynomial equations $p = 0$, where $p \in \mathbb{F}[x, y, z, \ldots]$ for some fixed (finite) field $\mathbb{F}$

Customary to omit "$= 0$" and only write p

The derivation rules are as follows, where $\alpha, \beta \in \mathbb{F}$, $p, q \in \mathbb{F}[x, y, z, \ldots]$, and x is any variable:

Boolean axioms $\dfrac{}{x^2 - x}$ for all variables x (forcing 0/1-solutions)

Linear combination $\dfrac{p \qquad q}{\alpha p + \beta q}$

Multiplication $\dfrac{p}{xp}$

PC-refutation: derivation of constant 1 (i.e., $1 = 0$)

(Note that multilinearity follows w.l.o.g. from $x^2 = x$ for all variables x)

Polynomial Calculus: Alternate View

Can also (equivalently) consider PC-derivation to be calculation in ideal generated by polynomials corresponding to clauses

Then a refutation concludes by proving that 1 is in this ideal, i.e., that the ideal is everything

Clearly: 1 is in ideal $\Rightarrow$ there is no common root

Less obvious: no common root $\Rightarrow$ 1 has to be in ideal
(requires some algebra)

Polynomial Calculus Measures

Length
derivation steps
(≈ # polynomial equations counted with repetitions)

Size
Total # monomials in the refutation counted with repetitions

(Monomial) space
Maximal # monomials in any configuration counted with repetitions
(Again an analogue of clause space)

Total space
Total # variables in any configuration counted with repetitions

Main Focus of Course

Look at resolution and polynomial calculus

- Relatively weak proof systems, so there is chance to understand them
- Also, because of this they can be (and are) used for SAT solving (as opposed to stronger systems)

Want to understand these systems and prove upper and lower bounds on

- size/length
- space
- size/length-space trade-offs

Interesting questions in their own right

Also hope that better understanding can say something about potential and limitations of SAT solving

State of the Art

- Resolution: much known
- Polynomial calculus: some known; some recent developments (we will cover results from as of yet unpublished papers)
- Cutting planes: still very poorly understood (and we won't have time to discuss it much)

Lots of good open questions for all three systems