Informatics 1

Lecture 9 Resolution (part 2)

Michael Fourman

Resolution (v 0.1)

$$\frac{X \vee G \qquad \neg G \vee Y}{X \vee Y}$$

This rule is **sound**:

if a valuation falsifies the **conclusion**
then it falsifies one of the **premises**

Constructing a refutation

If we apply this resolution rule

$$\frac{X \vee G \qquad \neg G \vee Y}{X \vee Y}$$

then given a refutation, **V** of the conclusion

$$V(X \vee Y) = \bot$$

we can produce a premise that it refutes

if $V(G) = \top$ then

V refutes $\neg G \vee Y$

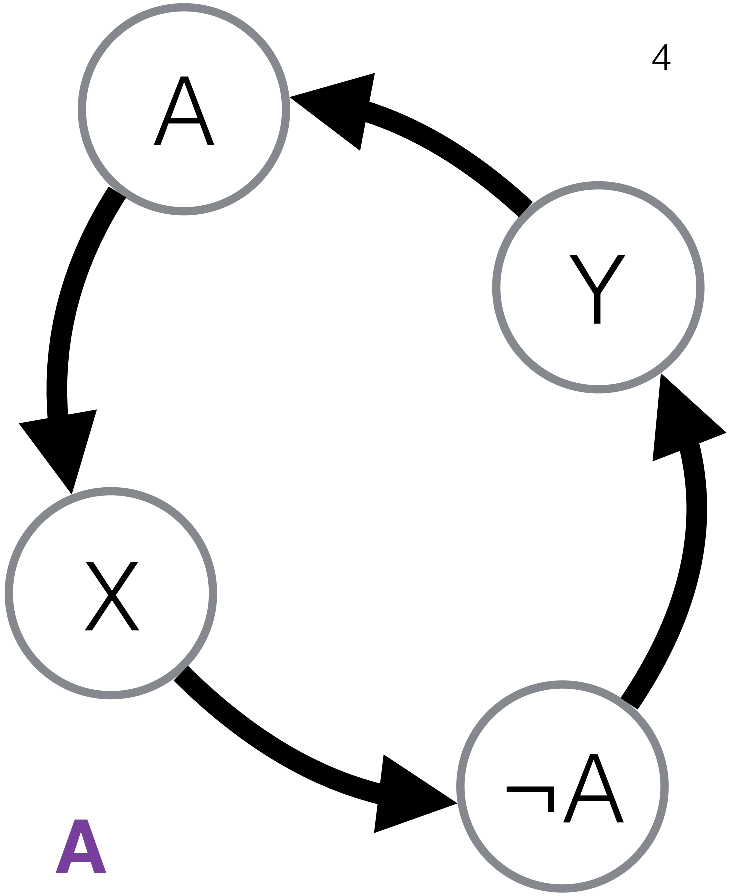
$V(\neg G \vee Y) = \bot$

if $V(G) = \bot$ then

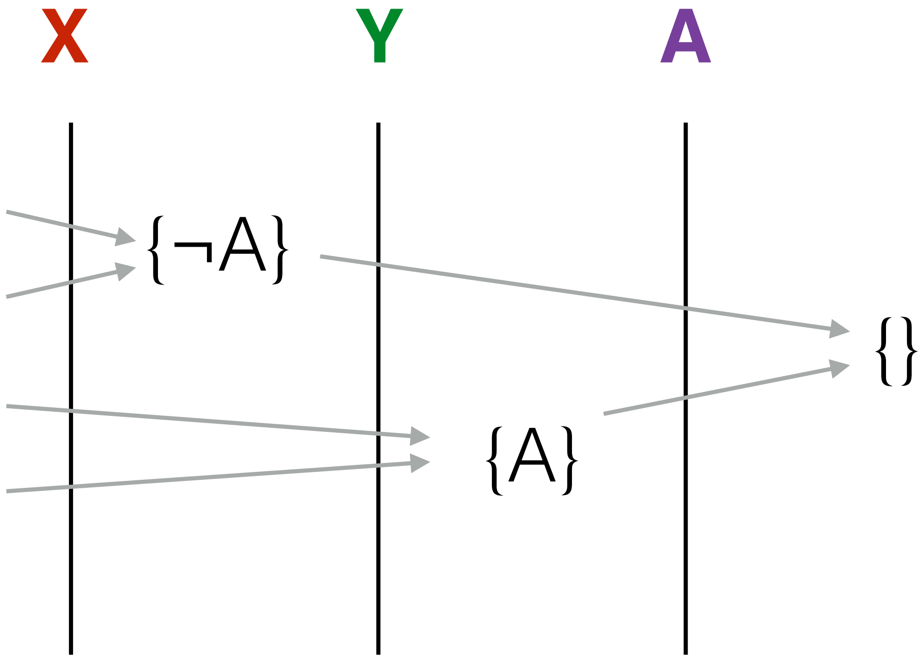
V refutes $X \vee G$

$V(X \vee G) = \bot$

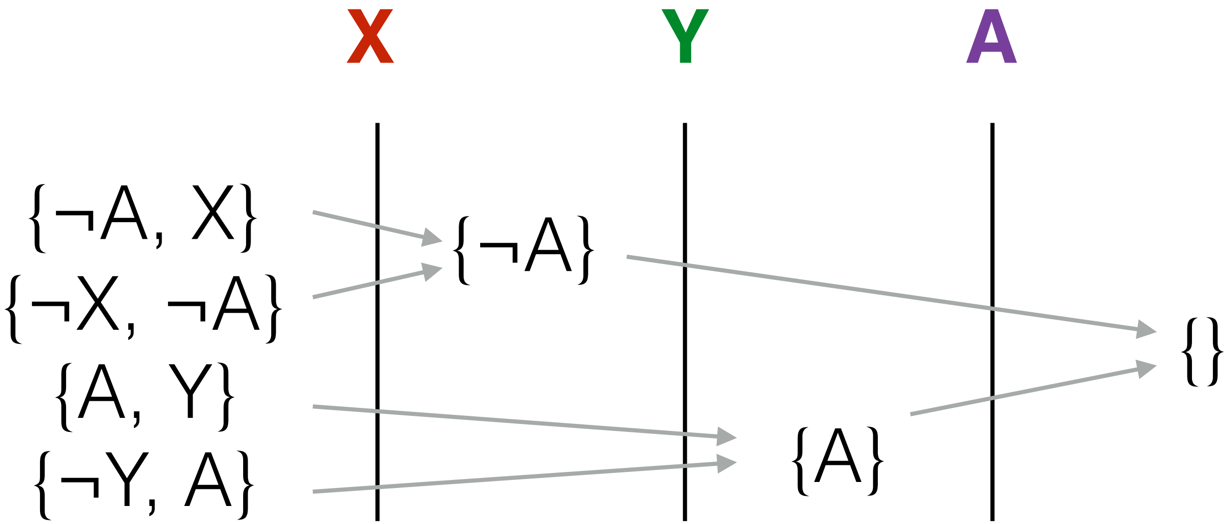
A contradictory cycle

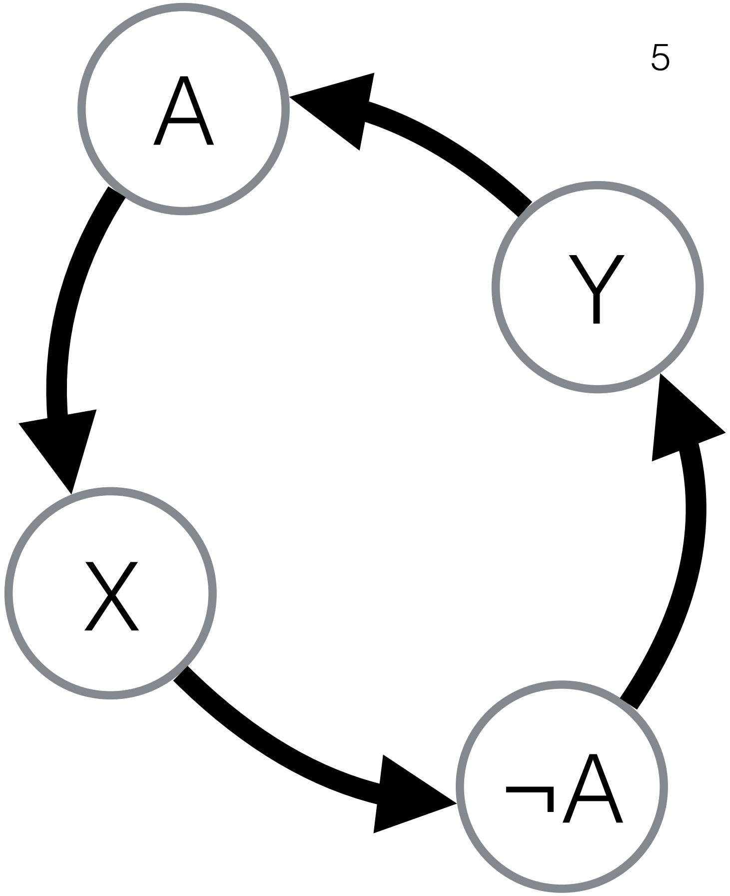


¬A∨X	{¬A, X}
¬X∨¬A	{¬X, ¬A}
A∨Y	{A, Y}
¬Y∨A	{¬Y, A}

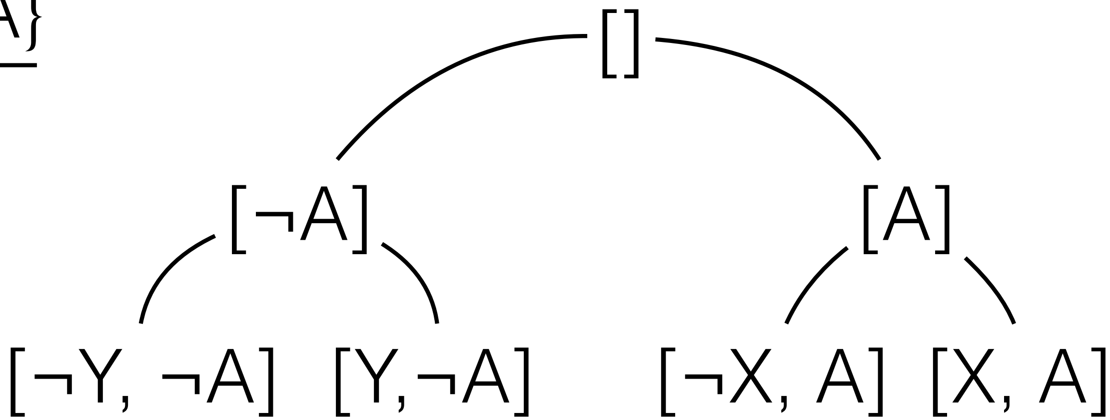


A contradictory cycle

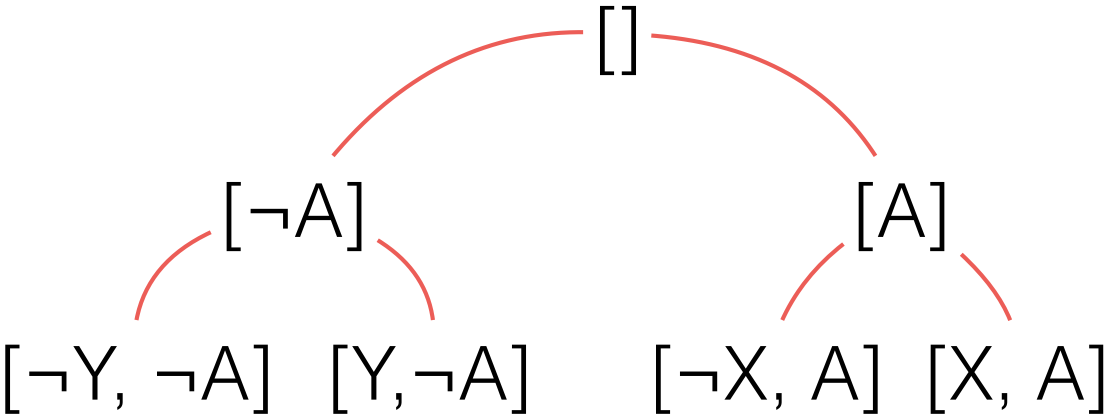


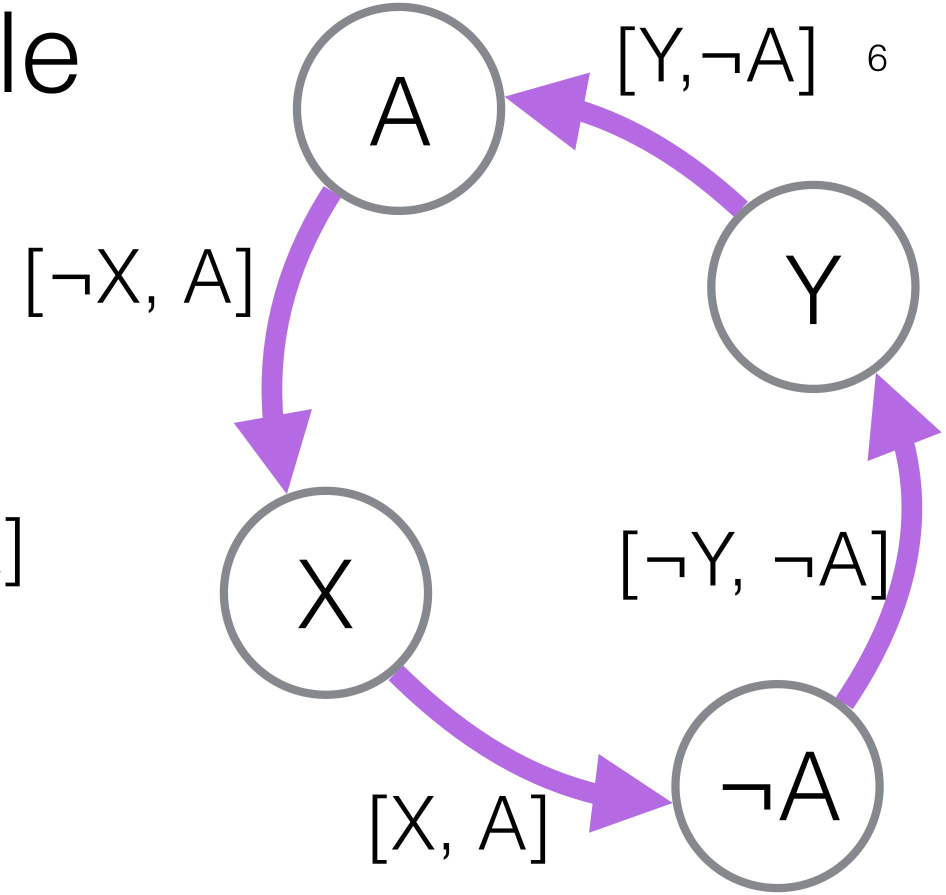


$$\dfrac{\dfrac{\{Y, A\} \qquad \{\neg Y, A\}}{\{A\}} \qquad \dfrac{\{X, \neg A\} \qquad \{\neg X, \neg A\}}{\{\neg A\}}}{\{\}}$$



A contradictory cycle





Resolution proof generates
a refutation tree.

$$\dfrac{\dfrac{\{Y, A\} \qquad \{\neg Y, A\}}{\{A\}} \qquad\qquad \dfrac{\{X, \neg A\} \qquad \{\neg X, \neg A\}}{\{\neg A\}}}{\{\}}$$

Note that we only
need to look at two
values to produce the
counterexample

Resolution on A

A

If ¬A occurs
in 3 clauses

¬A, X0
¬A, X1
¬A, X2

and A occurs
in 2 clauses

A, Y0
A, Y1

¬B, D
¬E, B

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

We have 2×3 = 6
opportunities for
resolution

Making progress

then
we can satisfy **all**
clauses in this set

then

if

If we can satisfy **all**
clauses in this set

A

¬A, X0
¬A, X1
¬A, X2
A, Y0
A, Y1

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

¬B, D
¬E, B

Making progress

A

¬A, X0
¬A, X1
¬A, X2
A, Y0
A, Y1

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

¬B, D
¬E, B

If we can satisfy **all** clauses in these two sets

then we can satisfy **all** clauses in these two sets

Making progress

A

~~¬A, X0~~
~~¬A, X1~~
~~¬A, X2~~
~~A, Y0~~
~~A, Y1~~

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

¬B, D
¬E, B

If we can satisfy **all** clauses in this set

then we can satisfy **all** clauses in this set, so we can forget them.

These are carried forward.

An example

¬A, C

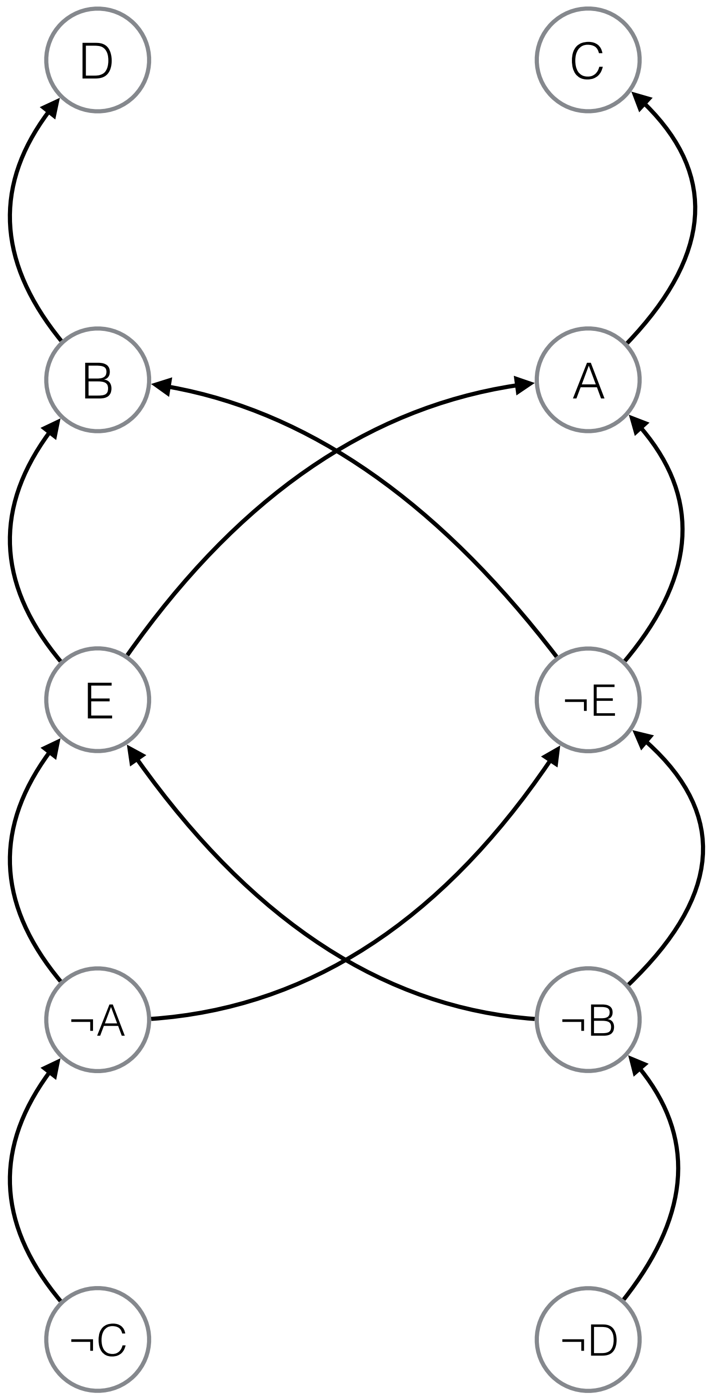
¬B, D

¬E, B

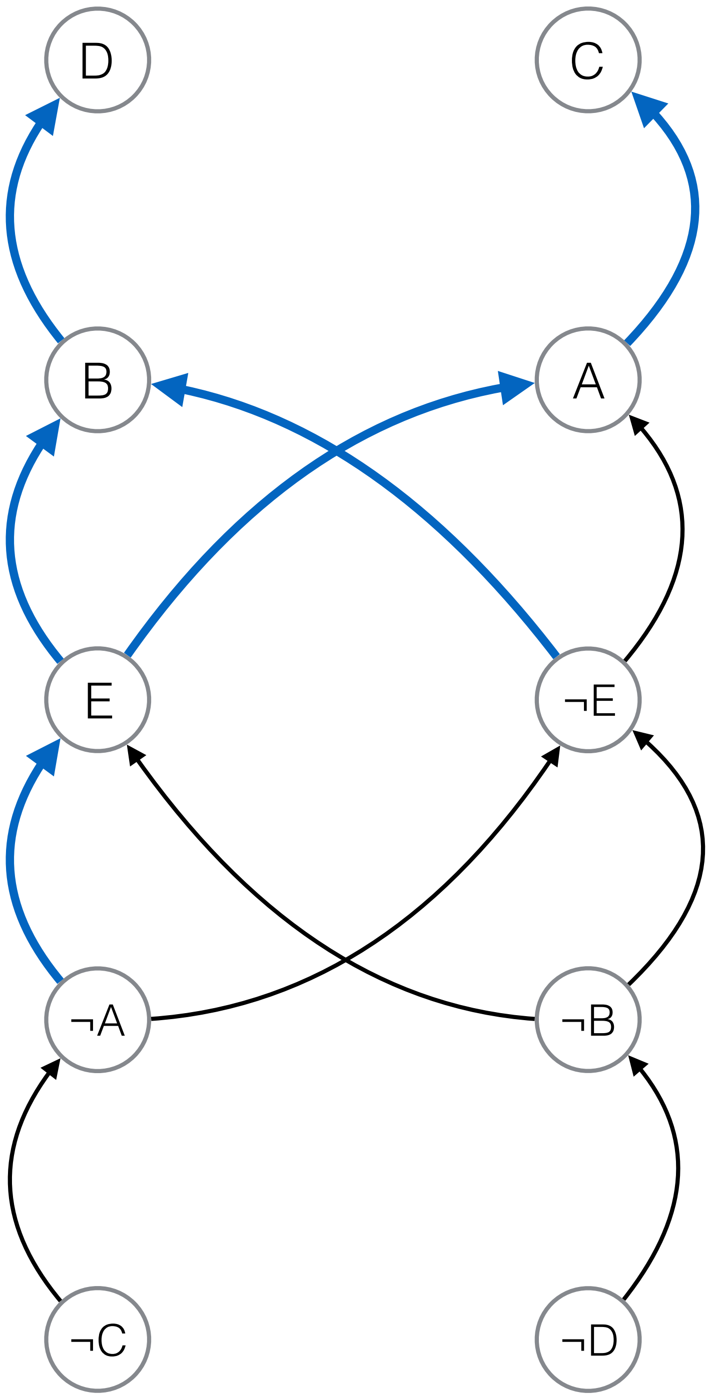
¬E, A

A, E

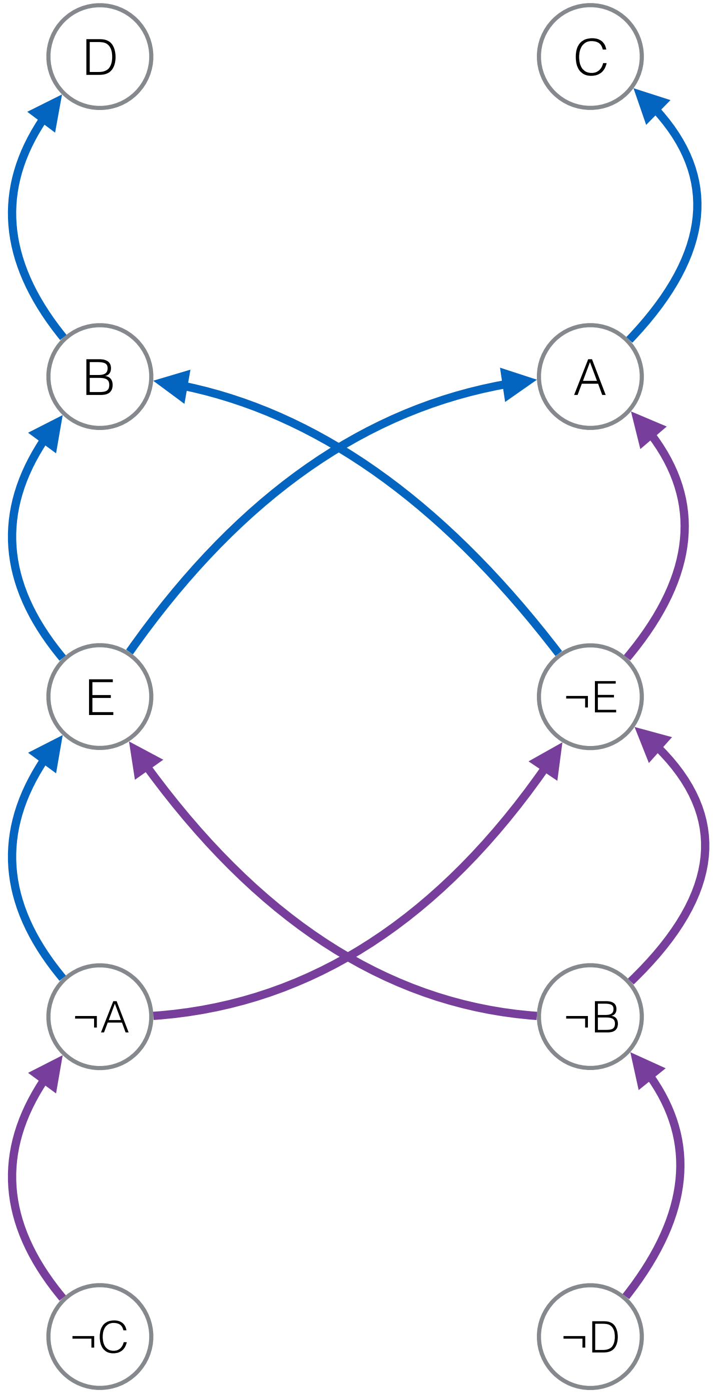
E, B



$\neg A \lor C$	$A \rightarrow C$
$\neg B \lor D$	$B \rightarrow D$
$\neg E \lor B$	$E \rightarrow B$
$\neg E \lor A$	$E \rightarrow A$
$A \lor E$	$\neg A \rightarrow E$
$E \lor B$	$\neg E \rightarrow B$



$\neg A \lor C$	$A \rightarrow C$	$\neg C \rightarrow \neg A$
$\neg B \lor D$	$B \rightarrow D$	$\neg D \rightarrow \neg B$
$\neg E \lor B$	$E \rightarrow B$	$\neg B \rightarrow \neg E$
$\neg E \lor A$	$E \rightarrow A$	$\neg A \rightarrow \neg E$
$A \lor E$	$\neg A \rightarrow E$	$\neg E \rightarrow A$
$E \lor B$	$\neg E \rightarrow B$	$\neg B \rightarrow E$



How many satisfying valuations?

$\neg A \lor C$

$\neg B \lor D$

$\neg E \lor B$

$\neg E \lor A$

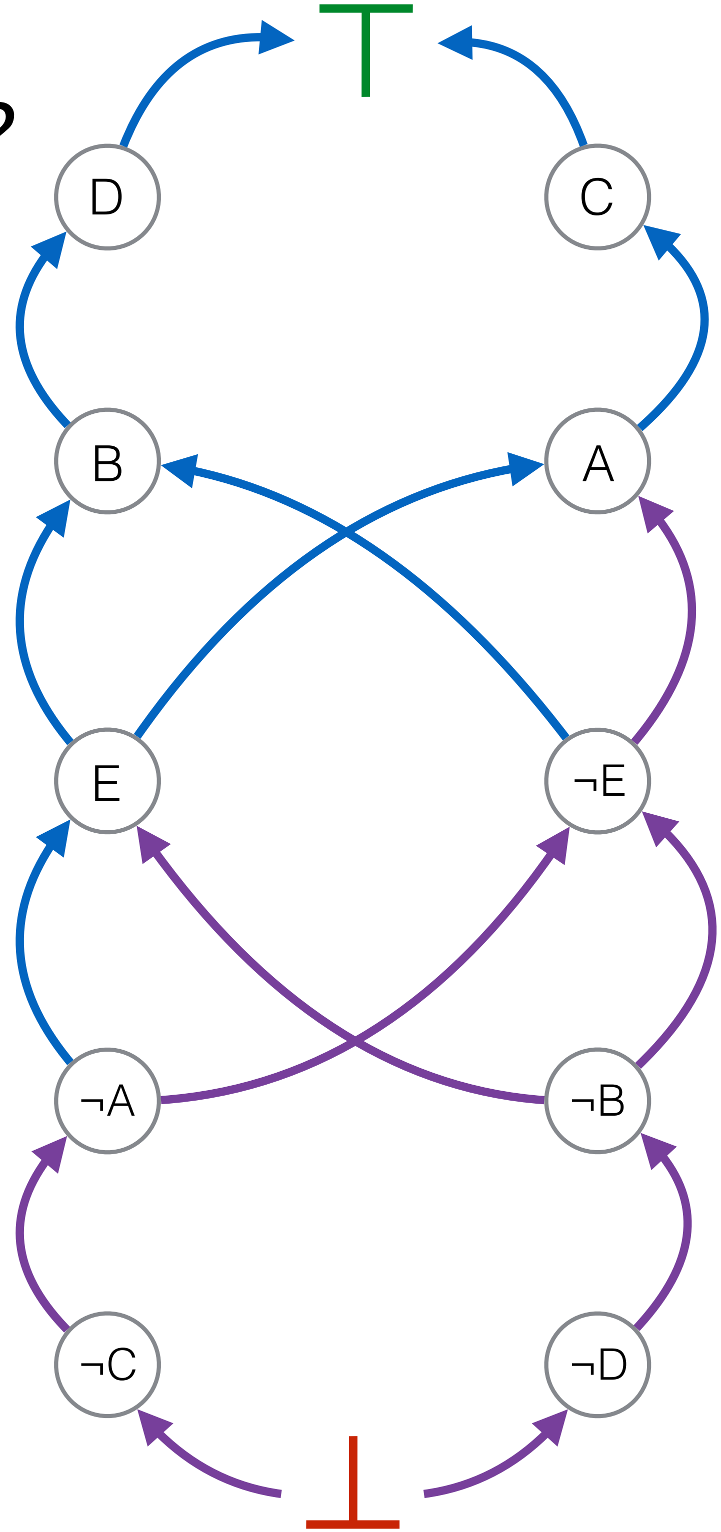
$A \lor E$

$E \lor B$

A satisfying valuation draws a line between false and true, such that

each atom is separated from its negation, and

no arrow leads from true to false.



How many satisfying valuations?

¬A ∨ C

¬B ∨ D

¬E ∨ B

¬E ∨ A

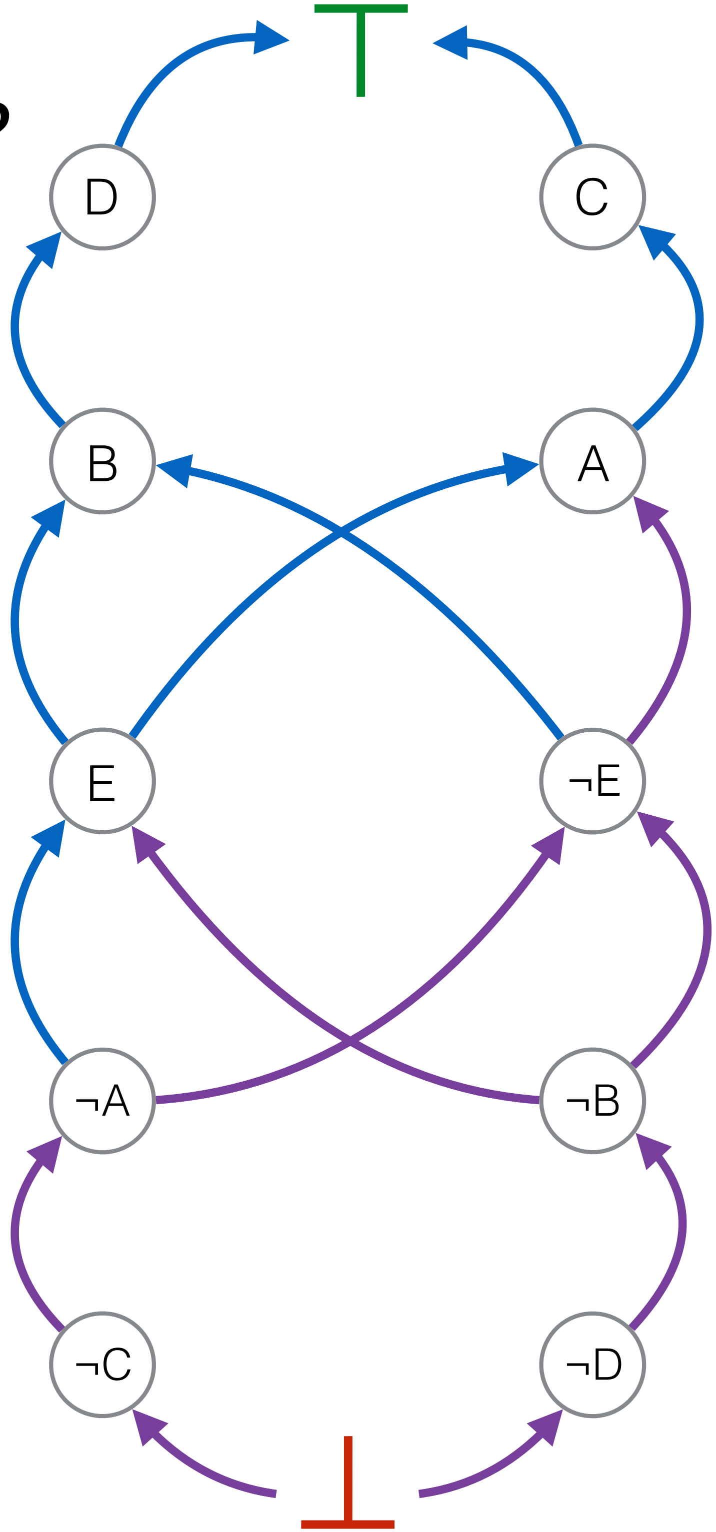
A ∨ E

E ∨ B

Unless there is a cycle including both X and ¬X, for some letter X, there is at least one satisfying valuation.

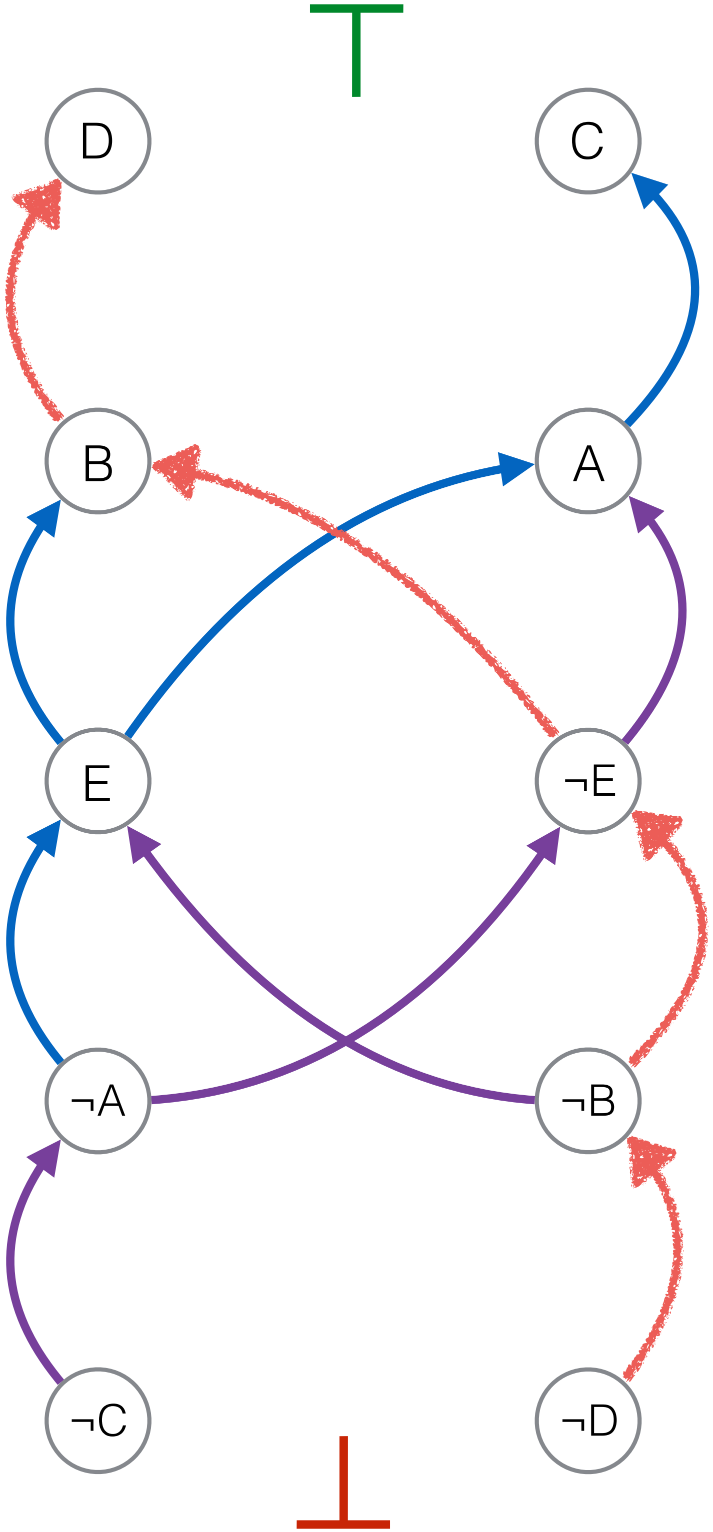
If there is a path ¬X→X then X must be true in every satisfying valuation.

If there is a path X→¬X then X must be false in every satisfying valuation.



Focus on one chain

$\neg A \vee C$	$A \rightarrow C$	$\neg C \rightarrow \neg A$
$\neg B \vee D$	$B \rightarrow D$	$\neg D \rightarrow \neg B$
$\neg E \vee B$	$E \rightarrow B$	$\neg B \rightarrow \neg E$
$\neg E \vee A$	$E \rightarrow A$	$\neg A \rightarrow \neg E$
$A \vee E$	$\neg A \rightarrow E$	$\neg E \rightarrow A$
$E \vee B$	$\neg E \rightarrow B$	$\neg B \rightarrow E$



Focus on one chain of reasoning

$\neg A \lor C$

$\mathbf{\neg B \lor D}$ $\quad B \to D \quad \neg D \to \neg B$

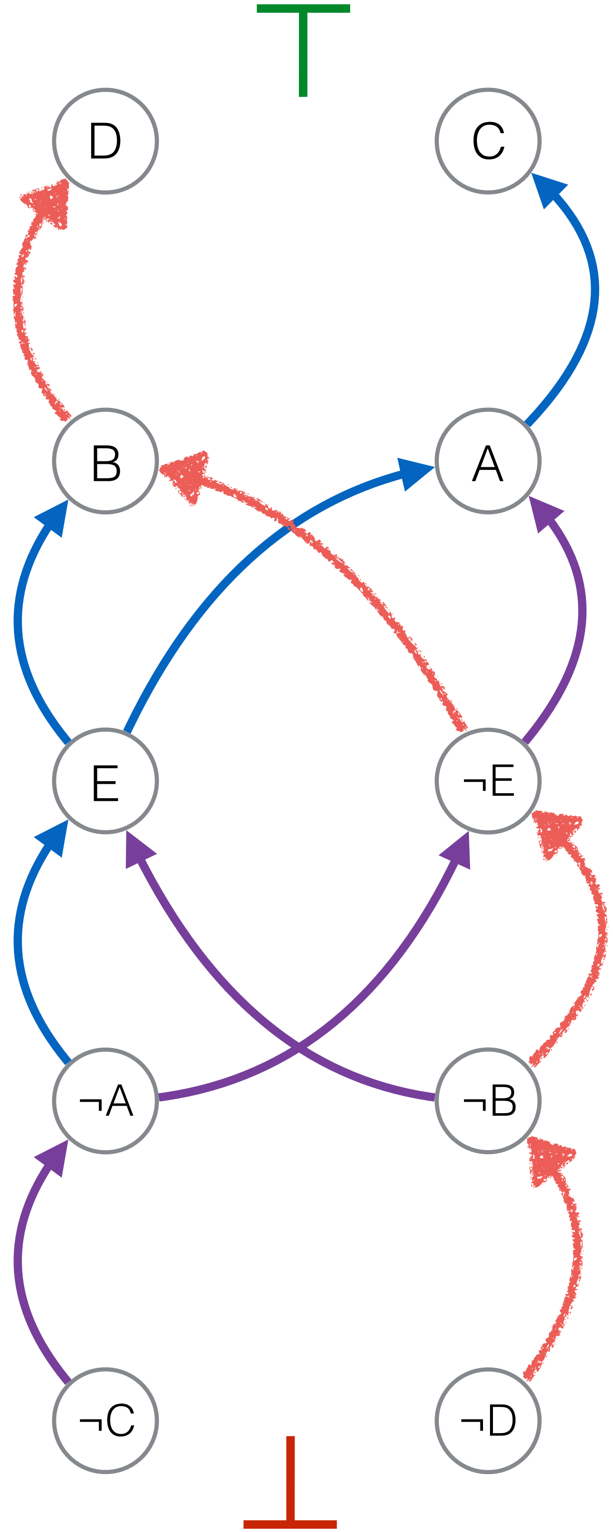
$\mathbf{\neg E \lor B}$ $\quad \neg B \to \neg E$

$\neg E \lor A$

$A \lor E$

$\mathbf{E \lor B}$ $\quad \neg E \to B$

$$\dfrac{\dfrac{\neg D \to \neg B \qquad \neg B \to \neg E}{\neg D \to \neg E} \qquad \dfrac{\neg E \to B \qquad B \to D}{\neg E \to D}}{\neg D \to D}$$



¬A, C

¬B, D

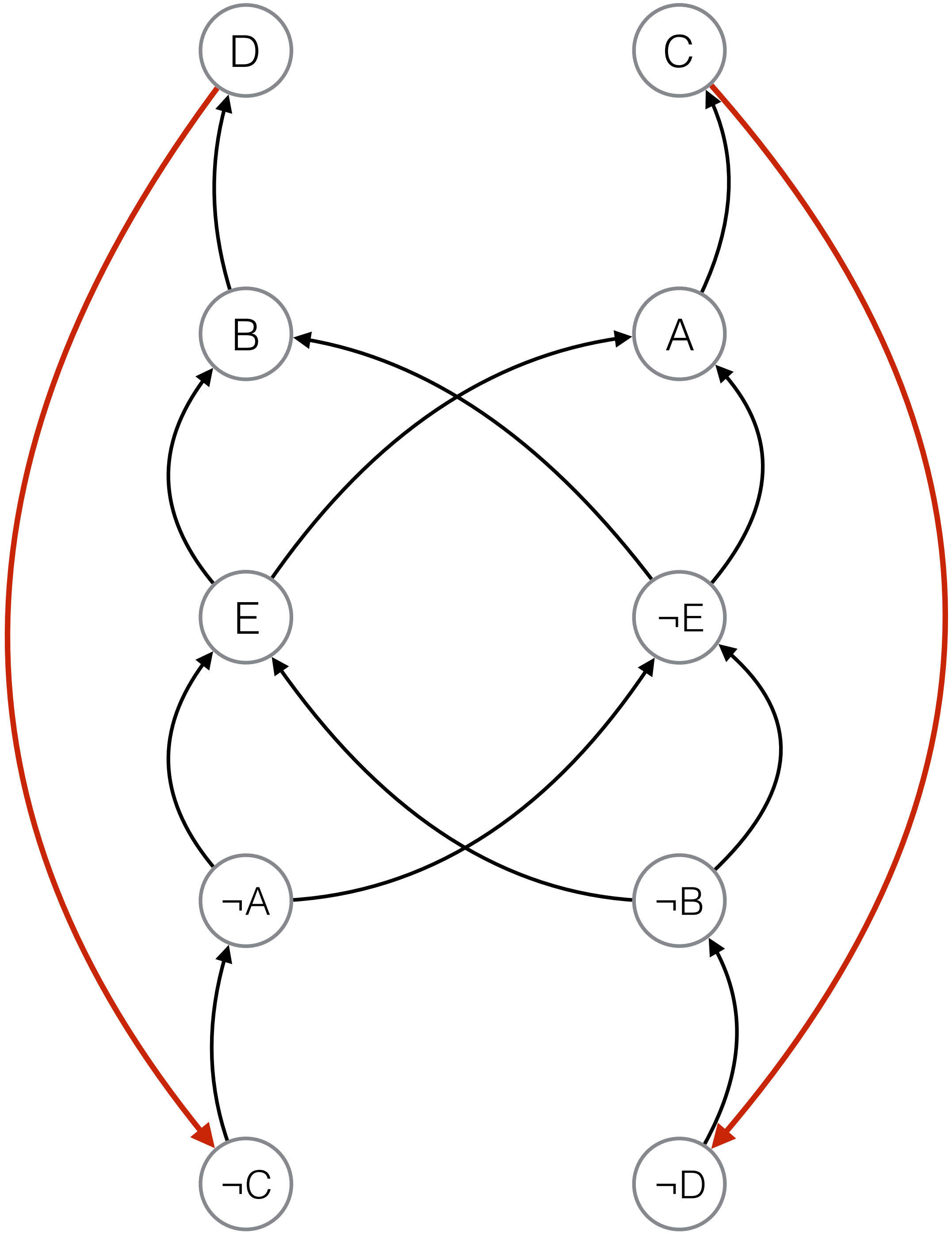
¬E, B

¬E, A

A, E

E, B

¬C, ¬D



A

¬A, C

¬B, D

¬E, B

¬E, A

A, E

E, B

¬C, ¬D

A

¬A, C	¬E, C
¬B, D	C, E
¬E, B	
¬E, A	
A, E	
E, B	
¬C, ¬D	

A	B
~~¬A, C~~	¬E, C
¬B, D	C, E
¬E, B	
~~¬E, A~~	
~~A, E~~	
E, B	
¬C, ¬D	

	A	B
~~¬A, C~~	¬E, C	¬E, D
¬B, D	C, E	E, D
¬E, B		
~~¬E, A~~		
~~A, E~~		
E, B		
¬C, ¬D		

A	B	C
~~¬A, C~~	**¬E, C**	¬E, D
~~¬B, D~~	**C, E**	E, D
~~¬E, B~~		
~~¬E, A~~		
~~A, E~~		
~~E, B~~		
¬C, ¬D		

A B C

~~¬A, C~~	**¬E, C**	¬E, D	¬E, ¬D
~~¬B, D~~	**C, E**	E, D	¬D, E
~~¬E, B~~			
~~¬E, A~~			
~~A, E~~			
~~E, B~~			
¬C, ¬D			

A	B	C	D
~~¬A, C~~	~~¬E, C~~	**¬E, D**	**¬E, ¬D**
~~¬B, D~~	~~C, E~~	**E, D**	**¬D, E**
~~¬E, B~~			
~~¬E, A~~			
~~A, E~~			
~~E, B~~			
~~¬C, ¬D~~			

A B C D

~~¬A, C~~	~~¬E, C~~	**¬E, D**	**¬E, ¬D**	¬E, E
~~¬B, D~~	~~C, E~~	**E, D**	**¬D, E**	E, E
~~¬E, B~~				¬E, ¬E
~~¬E, A~~				¬E, E
~~A, E~~				
~~E, B~~				
~~¬C, ¬D~~				

A	B	C	D	E
~~¬A, C~~	~~¬E, C~~	~~¬E, D~~	~~¬E, ¬D~~	¬E, E
~~¬B, D~~	~~C, E~~	~~E, D~~	~~¬D, E~~	**E**, E
~~¬E, B~~				**¬E**, ¬E
~~¬E, A~~				¬E, E
~~A, E~~				
~~E, B~~				
~~¬C, ¬D~~				

	A	B	C	D	E
~~¬A, C~~	~~¬E, C~~	~~¬E, D~~	~~¬E, ¬D~~	¬E, E	
~~¬B, D~~	~~C, E~~	~~E, D~~	~~¬D, E~~	**E**, E	{}
~~¬E, B~~				**¬E**, ¬E	
~~¬E, A~~				¬E, E	
~~A, E~~					
~~E, B~~					
~~¬C, ¬D~~					

Resolution

A complete proof procedure for propositional logic that works on formulas expressed in conjunctive normal form. (Robinson 1965)

Conjunctive Normal Form (CNF)

Literal: a propositional atom A or its negation ¬A
Clause: a disjunction of (a set of) literals.
CNF: a conjunction of (a set of) clauses.

Resolution

From two clauses

$$C_1 = (X \cup \{A\}),\ C_2 = (Y \cup \{\neg A\})$$

the resolution rule generates the new clause

$$(X \cup Y) = R(C_1, C_2)$$

where X and Y are sets of literals, not containing A or ¬A.

$(X \cup Y)$ is the resolvant

A is the variable resolved on

Resolution

A **resolution refutation** of a CNF $\mathcal{F}$ (a set of clauses) is a sequence C_1, C_2, …, C_m of clauses such that

$C_m = \{\}$, and

each C_i is either

a member of $\mathcal{F}$

or

the resolvant of two previous clauses in the proof:

$C_i = R(C_j, C_k)$, where $j,k < i$

Resolution

Any resolution proof can be represented as a DAG nodes are clauses in the proof.

Clauses in F are leaves: they have no incoming edges.

Every clause C_i that arises from a resolution step has two incoming edges. One from each of the clauses (C_j,C_k) that were resolved together to obtain $C_i = R(C_j,C_k)$.

Each non-leaf node C_i is labeled by the variable that was resolved away to obtain it.

When resolution ‘fails’

~~¬A, X0~~
~~¬A, X1~~
~~¬A, X2~~
~~A, Y0~~
~~A, Y1~~

~~¬B, D~~
~~¬E, B~~

A

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

B

¬E, D

If we have not produced {}, and there are no remaining opportunities for resolution, then every remaining literal is a **pure literal**.
Pure means that its negation does not occur.
We can satisfy the remaining clauses by making every literal true.

When resolution ‘fails’

~~¬A, X0~~
~~¬A, X1~~
~~¬A, X2~~
~~A, Y0~~
~~A, Y1~~

~~¬B, D~~
~~¬E, B~~

A

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

B

¬E, D

We can satisfy the remaining clauses by making every literal true.

This gives a partial valuation, which can be extended to the resolved variables in order to satisfy every clause.

Making progress

then

then
we can satisfy
all clauses in
this set

¬**A, X0**
¬**A, X1**
¬**A, X2**
A, Y0
A, Y1

¬B, D
¬E, B

A

if

If we can satisfy **all**
clauses in this set

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

Extending a partial solution

then
we can satisfy **all** clauses in this set

then

if

If we can satisfy **all** clauses in this set

A

¬A, X0
¬A, X1
¬A, X2
A, Y0
A, Y1

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

For example, if **Y0** is false then **X0,X1,X2** must all be true to make the highlighted clauses true

¬B, D
¬E, B

one way

then
we can satisfy
all clauses in
this set

then

if

If we can satisfy **all**
clauses in this set

A

¬A, X0
¬A, X1
¬A, X2
A, Y0
A, Y1

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

For example,
if **Y0** is false
then **X0,X1,X2**
must all be true
to make the
highlighted
clauses true

If **X0,X1,X2**
are all true
we can
make **A** true
to make all
these clauses true

¬B, D
¬E, B

or another

then
we can satisfy **all** clauses in this set

then

if

If we can satisfy **all** clauses in this set

A

¬A, X0
¬A, X1
¬A, X2
A, Y0
A, Y1

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

If **X0,X1,X2** are all true we can make **A** true to make all these clauses true

If one of the **Ys** is false then **X0,X1,X2** must all be true

¬B, D
¬E, B

one way or another

then
we can satisfy
all clauses in
this set

then

if

If we can satisfy **all**
clauses in this set

A

¬A, X0
¬A, X1
¬A, X2
A, Y0
A, Y1

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

If **Y0,Y1**
are both true
we can
make **A** false,
so **¬A is** true,
to make all
these clauses true

If one of the X**s**
is false
then **Y0,Y1**
must both be true

¬B, D
¬E, B

one way or another

then
we can satisfy
all clauses in
this set

then

if

If we can satisfy **all**
clauses in this set

A

¬A, **X0**
¬A, **X1**
¬A, **X2**
A, **Y0**
A, **Y1**

X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

If **Y0,Y1**
are both true
all these
clauses are true
whether we
make
A true or, **A** false,

If
the Xs and the Ys
are all true

¬B, D
¬E, B

When resolution ‘fails’

¬A, X0
¬A, X1
¬A, X2
A, Y0
A, Y1

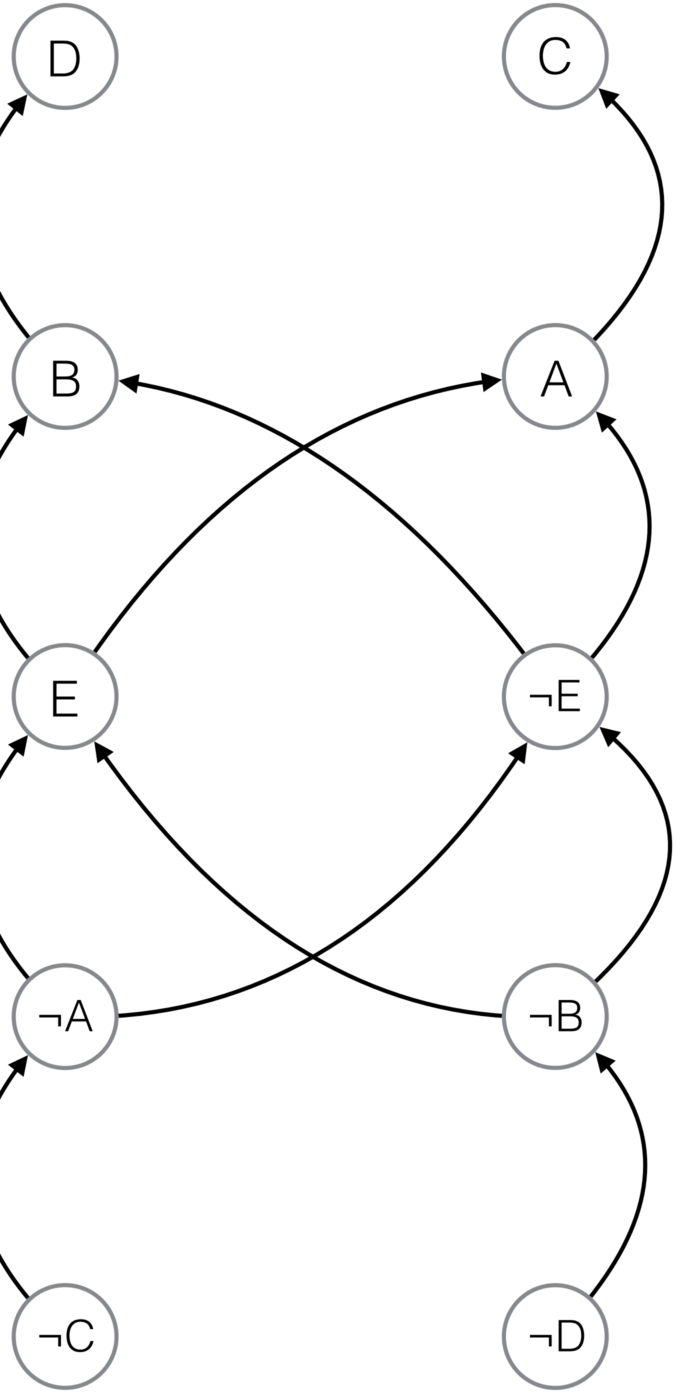
X0, Y0
X0, Y1
X1, Y0
X1, Y1
X2, Y0
X2, Y1

Then we can satisfy all clauses in these two sets

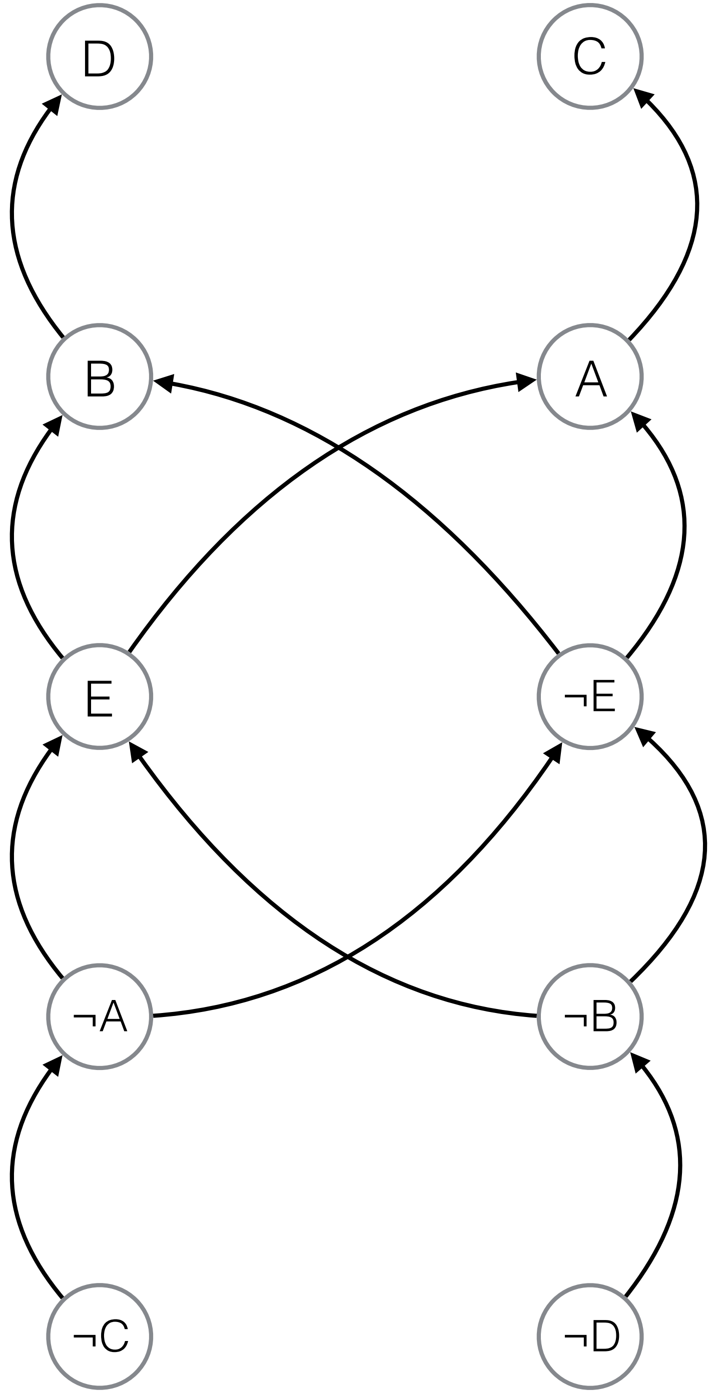
¬B, D
¬E, B

If we can satisfy all clauses in these two sets

	A	B	E
~~¬A, C~~	**¬E, C**	**¬E, D**	
~~¬B, D~~	**C, E**	**E, D**	
~~¬E, B~~			
~~¬E, A~~			
~~A, E~~			
~~E, B~~			



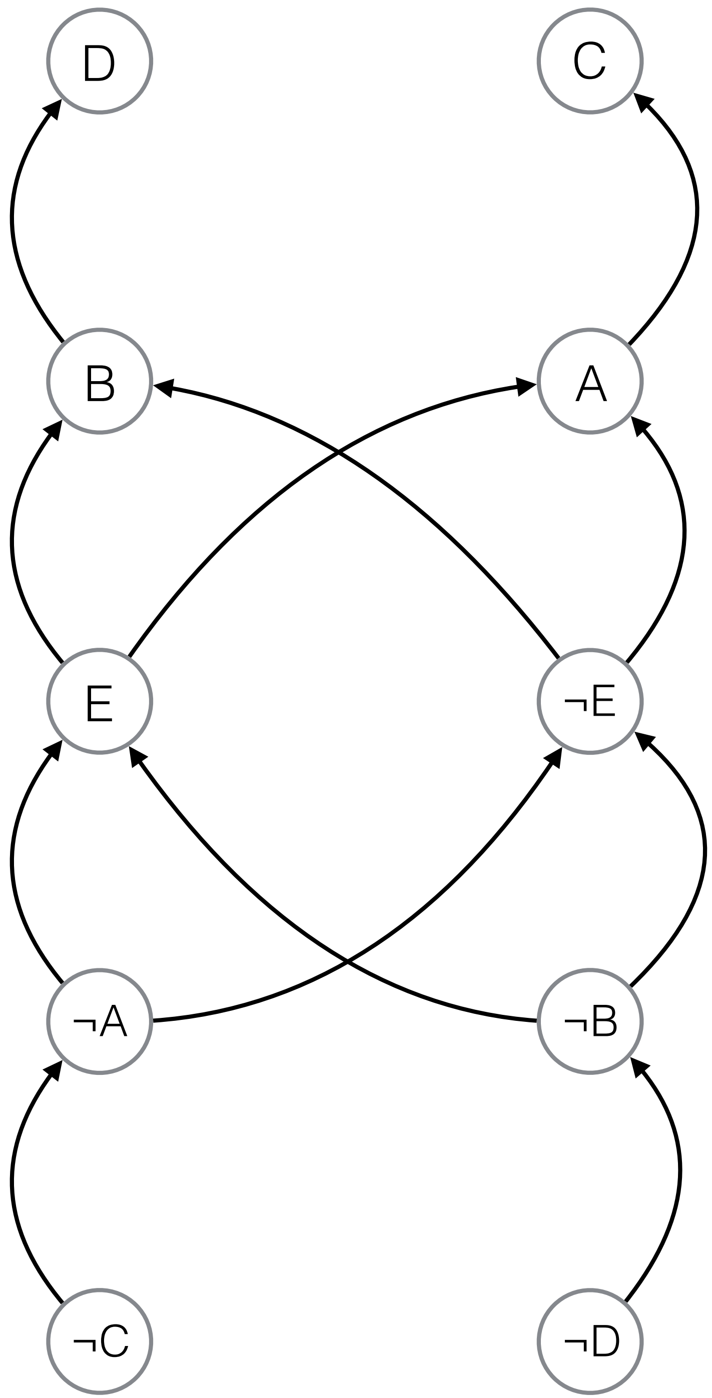
	A	B	E
~~¬A, C~~	**¬E, C**	**¬E, D**	**C**
~~¬B, D~~	**C, E**	**E, D**	**D**
~~¬E, B~~			
~~¬E, A~~			
~~A, E~~			
~~E, B~~			



A	B	E	
~~¬A, C~~	~~¬E, C~~	~~¬E, D~~	**C**
~~¬B, D~~	~~C, E~~	~~E, D~~	**D**
~~¬E, B~~			
~~¬E, A~~			
~~A, E~~			
~~E, B~~			

C and **D** are solitary literals they must be true

partial valuation [**C, D, …**]

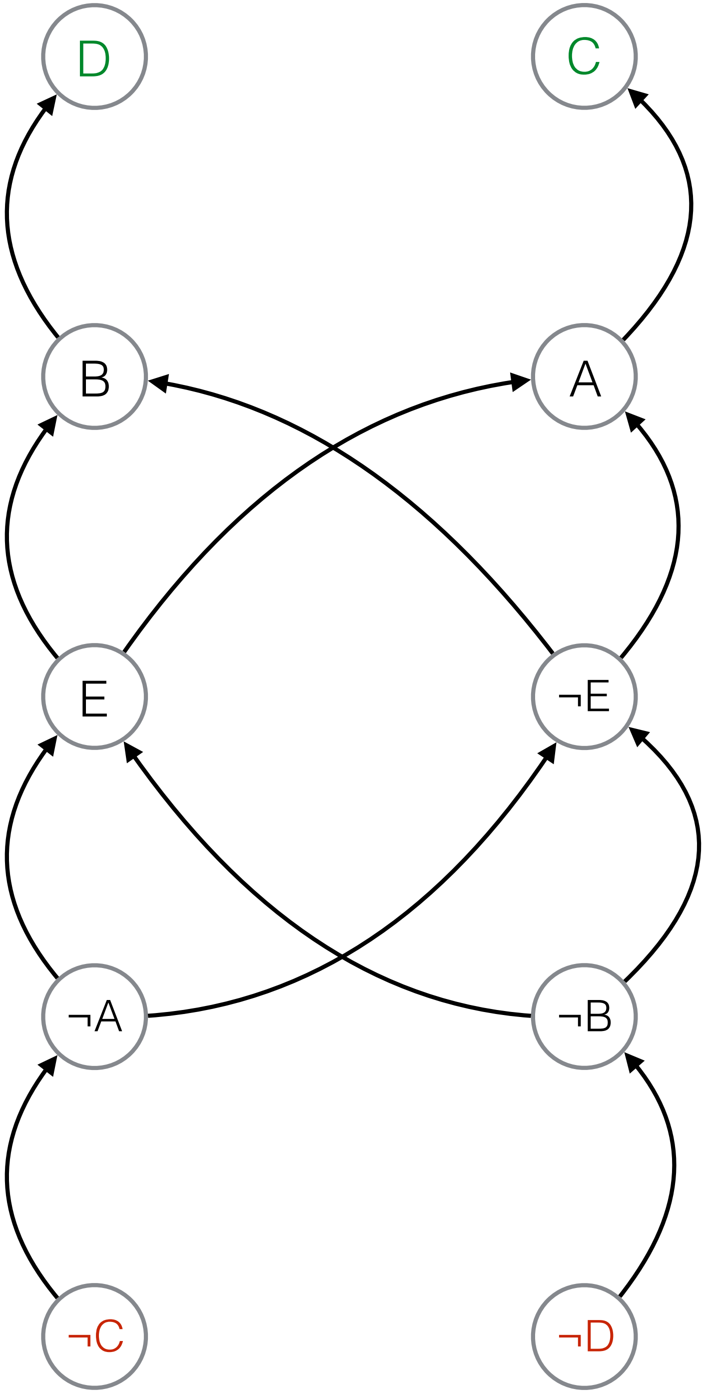


	A	B	**E**
~~¬A, C~~	**¬E, C**	**¬E, D**	**C**
~~¬B, D~~	**C, E**	**E, D**	**D**
~~¬E, B~~			
~~¬E, A~~			
~~A, E~~			
~~E, B~~			

All the **¬E** clauses
and the **E** clauses
used in resolution
are true,
so we can choose
to make **E**
true or false

partial valuation [**C, D, …**]

E can be freely chosen



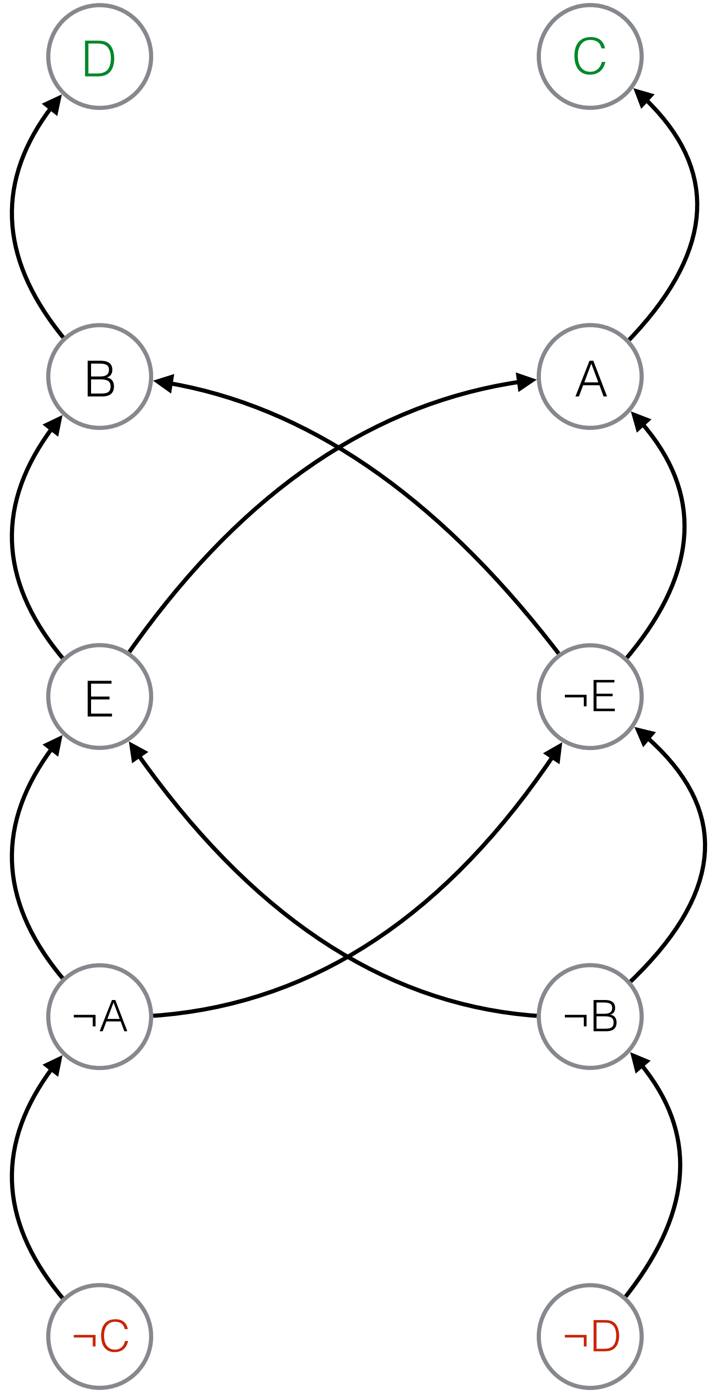
	A	**B**	E
~~¬A, C~~	¬E, C	**¬E, D**	C
¬B, D	C, E	**E, D**	D
¬E, B			
~~¬E, A~~			
~~A, E~~			
E, B			

No matter how we chose **E**
The **¬B** clause is true
since **D** is true

No matter how we chose **E**
to make both **B** clauses true
we must make **B** true

partial valuation [**C, D, …**]

E can be freely chosen



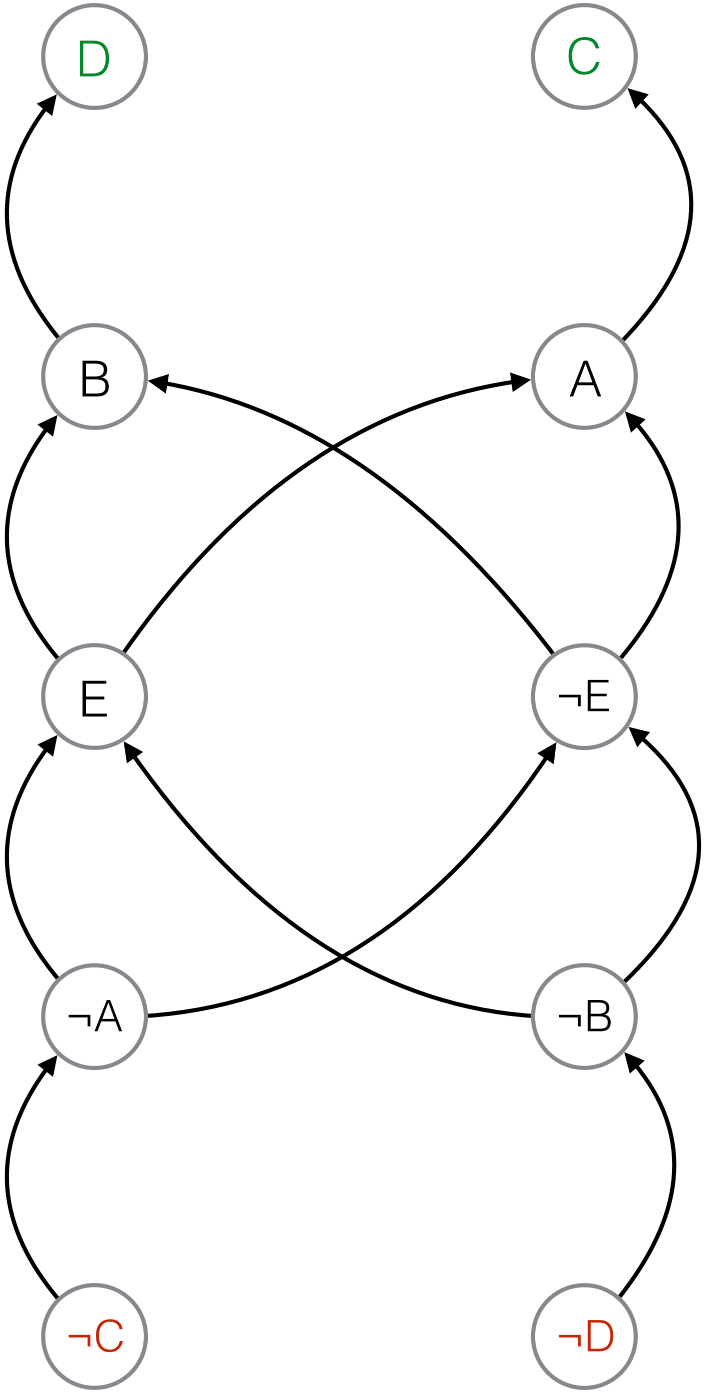
	A	B	E
~~¬A, C~~	¬E, C	**¬E, D**	C
¬B, D	C, E	**E, D**	D
¬E, B			
~~¬E, A~~			
~~A, E~~			
E, B			

No matter how we chose **E**
The **¬B** clause is true
since **D** is true

No matter how we chose **E**
to make both **B** clauses true
we must make **B** true

partial valuation [**C, D, B, …**]

E can be freely chosen



	A	B	E
¬A, C	**¬E, C**	¬E, D	C
¬B, D	**C, E**	E, D	D
¬E, B			
¬E, A			
A, E			
E, B			

No matter how we chose **E**
The **¬A** clause is true
since **C** is true

No matter how we chose **E**,
to make both **A** clauses true
we must make **A** true

partial valuation [**C, D, B, A**]

E can be freely chosen