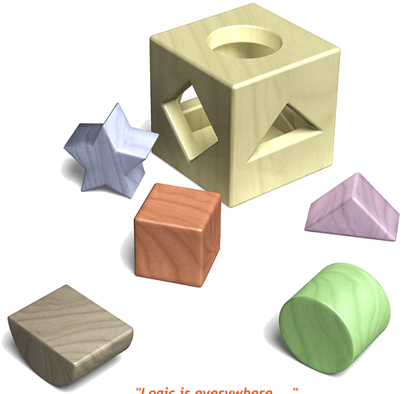
SAT Solving – Conflict Analysis

Steffen Hölldobler and Norbert Manthey
International Center for Computational Logic
Technische Universität Dresden
Germany

- Conflict Analysis
- Implication Graphs
- Unique Implication Points



"Logic is everywhere ..."

Conflict Analysis – Warm Up

- Given a formiula F, and interpretation J and a clause C, and the current decision level is n.
  - (we always prefer termination and unit propagation over search)

Conflict Analysis – Warm Up

- **Given a formiula F, and interpretation J and a clause C, and the current decision level is n.**
  - **(we always prefer termination and unit propagation over search)**

- **When is C a conflict clause?**

Conflict Analysis – Warm Up

- **Given a formiula F, and interpretation J and a clause C, and the current decision level is n.**
  - **(we always prefer termination and unit propagation over search)**

- **When is C a conflict clause?**

- **Which 2 properties do we like to have from learned clauses?**

Conflict Analysis – Warm Up

- **Given a formiula *F*, and interpretation *J* and a clause *C*, and the current decision level is *n*.**
  - **(we always prefer termination and unit propagation over search)**
- **When is *C* a conflict clause?**
- **Which 2 properties do we like to have from learned clauses?**
- **How many literals in *C* have at least the decision level *n*?**

Conflict Analysis – Advanced Implementation

Example of the sophisticated linear resolution derivation algorithm

Conflict Analysis – Revisited

- **Let F be a formula in CNF, L a literal, and J a partial interpretation.**
- **$C \in F$ is called conflict clause under J iff $C|_J = [\,]$.**

Conflict Analysis – Revisited

- Let F be a formula in CNF, L a literal, and J a partial interpretation.
- $C \in F$ is called conflict clause under J iff $C|_J = [\,]$.
- A clause C is relevant for L (or is a reason for L) in $F :: J$
  iff $C \in F$ and there exist I' and I such that $J = I', L, I$ and $C|_I = [L]$.
- relevantL($L, F :: J$) $= \{C \in F \mid C$ is relevant for L in $F :: J\}$.

Conflict Analysis – Revisited

- Let F be a formula in CNF, L a literal, and J a partial interpretation.
- $C \in F$ is called conflict clause under J iff $C|_J = [\,]$.
- A clause C is relevant for L (or is a reason for L) in $F :: J$
  iff $C \in F$ and there exist I' and I such that $J = I', L, I$ and $C|_I = [L]$.
- relevantL($L, F :: J$) $= \{C \in F \mid C$ is relevant for L in $F :: J\}$.
- A clause C is relevant (or is a reason clause) in $F :: J$
  iff there exists an L such that C is relevant for L in $F :: J$.
- relevant($F :: J$) $= \{C \in F \mid C$ is relevant in $F :: J\}$.

An Old Example Revisited

- Let $F = \langle [1, 2], [2, \bar{3}], [\bar{2}, \bar{3}, 4], [\bar{1}, 3], [\bar{4}] \rangle$
- We may obtain

$$
\begin{array}{llll}
F :: () & \leadsto_{\mathit{UNIT}} & F :: (\bar{4}) & (F|_{(\bar{4})} = \langle [1, 2], [2, \bar{3}], [\bar{2}, \bar{3}], [\bar{1}, 3] \rangle) \\
& \leadsto_{\mathit{DECIDE}} & F :: (\bar{4}, \dot{1}) & (F|_{(\bar{4},1)} = \langle [2, \bar{3}], [\bar{2}, \bar{3}], [3] \rangle) \\
& \leadsto_{\mathit{UNIT}} & F :: (\bar{4}, \dot{1}3) & (F|_{(\bar{4},1,3)} = \langle [2], [\bar{2}] \rangle) \\
& \leadsto_{\mathit{UNIT}} & F :: (\bar{4}, \dot{1}, 3, 2) & (F|_{(\bar{4},1,3,2)} = \langle [\,] \rangle)
\end{array}
$$

An Old Example Revisited

- Let $F = \langle [1, 2], [2, \overline{3}], [\overline{2}, \overline{3}, 4], [\overline{1}, 3], [\overline{4}] \rangle$
- We may obtain

$$
\begin{array}{llll}
F :: () & \leadsto_{UNIT} & F :: (\overline{4}) & (F|_{(\overline{4})} = \langle [1, 2], [2, \overline{3}], [\overline{2}, \overline{3}], [\overline{1}, 3] \rangle) \\
& \leadsto_{DECIDE} & F :: (\overline{4}, \dot{1}) & (F|_{(\overline{4},1)} = \langle [2, \overline{3}], [\overline{2}, \overline{3}], [3] \rangle) \\
& \leadsto_{UNIT} & F :: (\overline{4}, \dot{1}3) & (F|_{(\overline{4},1,3)} = \langle [2], [\overline{2}] \rangle) \\
& \leadsto_{UNIT} & F :: (\overline{4}, \dot{1}, 3, 2) & (F|_{(\overline{4},1,3,2)} = \langle [\,] \rangle)
\end{array}
$$

- We find

$$
\text{relevantL}(\overline{4}, F :: (\overline{4}, \dot{1}, 3, 2)) \quad = \quad \{[\overline{4}]\}
$$

An Old Example Revisited

- Let $F = \langle [1, 2], [2, \overline{3}], [\overline{2}, \overline{3}, 4], [\overline{1}, 3], [\overline{4}] \rangle$
- We may obtain

$$
\begin{array}{llll}
F :: () & \leadsto_{UNIT} & F :: (\overline{4}) & (F|_{(\overline{4})} = \langle [1, 2], [2, \overline{3}], [\overline{2}, \overline{3}], [\overline{1}, 3] \rangle) \\
& \leadsto_{DECIDE} & F :: (\overline{4}, \dot{1}) & (F|_{(\overline{4}, 1)} = \langle [2, \overline{3}], [\overline{2}, \overline{3}], [3] \rangle) \\
& \leadsto_{UNIT} & F :: (\overline{4}, \dot{1}3) & (F|_{(\overline{4}, 1, 3)} = \langle [2], [\overline{2}] \rangle) \\
& \leadsto_{UNIT} & F :: (\overline{4}, \dot{1}, 3, 2) & (F|_{(\overline{4}, 1, 3, 2)} = \langle [\,] \rangle)
\end{array}
$$

- We find

$$
\begin{array}{lll}
\text{relevantL}(\overline{4}, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \{[\overline{4}]\} \\
\text{relevantL}(1, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \emptyset
\end{array}
$$

An Old Example Revisited

- Let $F = \langle [1,2], [2,\overline{3}], [\overline{2},\overline{3},4], [\overline{1},3], [\overline{4}] \rangle$
- We may obtain

$$\begin{array}{llll}
F :: () & \leadsto_{UNIT} & F :: (\overline{4}) & (F|_{(\overline{4})} = \langle [1,2], [2,\overline{3}], [\overline{2},\overline{3}], [\overline{1},3] \rangle) \\
 & \leadsto_{DECIDE} & F :: (\overline{4}, \dot{1}) & (F|_{(\overline{4},1)} = \langle [2,\overline{3}], [\overline{2},\overline{3}], [3] \rangle) \\
 & \leadsto_{UNIT} & F :: (\overline{4}, \dot{1}3) & (F|_{(\overline{4},1,3)} = \langle [2], [\overline{2}] \rangle) \\
 & \leadsto_{UNIT} & F :: (\overline{4}, \dot{1}, 3, 2) & (F|_{(\overline{4},1,3,2)} = \langle [\,] \rangle)
\end{array}$$

- We find

$$\begin{array}{lll}
\mathrm{relevantL}(\overline{4}, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \{[\overline{4}]\} \\
\mathrm{relevantL}(1, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \emptyset \\
\mathrm{relevantL}(3, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \{[\overline{1}, 3]\}
\end{array}$$

An Old Example Revisited

- Let $F = \langle [1,2], [2,\overline{3}], [\overline{2},\overline{3},4], [\overline{1},3], [\overline{4}] \rangle$
- We may obtain

$$\begin{array}{llll} F :: () & \leadsto_{UNIT} & F :: (\overline{4}) & (F|_{(\overline{4})} = \langle [1,2], [2,\overline{3}], [\overline{2},\overline{3}], [\overline{1},3] \rangle) \\ & \leadsto_{DECIDE} & F :: (\overline{4}, \dot{1}) & (F|_{(\overline{4},1)} = \langle [2,\overline{3}], [\overline{2},\overline{3}], [3] \rangle) \\ & \leadsto_{UNIT} & F :: (\overline{4}, \dot{1}3) & (F|_{(\overline{4},1,3)} = \langle [2], [\overline{2}] \rangle) \\ & \leadsto_{UNIT} & F :: (\overline{4}, \dot{1}, 3, 2) & (F|_{(\overline{4},1,3,2)} = \langle [\,] \rangle) \end{array}$$

- We find

$$\begin{array}{lcl} \text{relevantL}(\overline{4}, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \{[\overline{4}]\} \\ \text{relevantL}(1, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \emptyset \\ \text{relevantL}(3, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \{[\overline{1}, 3]\} \\ \text{relevantL}(2, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \{[2, \overline{3}]\} \end{array}$$

An Old Example Revisited

- **Let** $F = \langle [1,2], [2,\overline{3}], [\overline{2},\overline{3},4], [\overline{1},3], [\overline{4}] \rangle$
- **We may obtain**

$$
\begin{array}{llll}
F :: () & \leadsto_{UNIT} & F :: (\overline{4}) & (F|_{(\overline{4})} = \langle [1,2], [2,\overline{3}], [\overline{2},\overline{3}], [\overline{1},3] \rangle) \\
 & \leadsto_{DECIDE} & F :: (\overline{4}, \dot{1}) & (F|_{(\overline{4},1)} = \langle [2,\overline{3}], [\overline{2},\overline{3}], [3] \rangle) \\
 & \leadsto_{UNIT} & F :: (\overline{4}, \dot{1}3) & (F|_{(\overline{4},1,3)} = \langle [2], [\overline{2}] \rangle) \\
 & \leadsto_{UNIT} & F :: (\overline{4}, \dot{1}, 3, 2) & (F|_{(\overline{4},1,3,2)} = \langle [\,] \rangle)
\end{array}
$$

- **We find**

$$
\begin{array}{lll}
\text{relevantL}(\overline{4}, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \{[\overline{4}]\} \\
\text{relevantL}(1, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \emptyset \\
\text{relevantL}(3, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \{[\overline{1}, 3]\} \\
\text{relevantL}(2, F :: (\overline{4}, \dot{1}, 3, 2)) & = & \{[2, \overline{3}]\} \\
\text{relevant}(F :: (\overline{4}, \dot{1}, 3, 2)) & = & \{[\overline{4}], [\overline{1}, 3], [2, \overline{3}]\}
\end{array}
$$

Another Example

- Consider $F = \langle[\overline{1}, \overline{3}, 4], [\overline{2}, \overline{3}, 4]\rangle$.
- We may obtain

$$
\begin{array}{llll}
F :: () & \leadsto_{DECIDE} & F :: (\dot{1}) & (F|_{(1)} = \langle[\overline{3}, 4], [\overline{2}, \overline{3}, 4]\rangle) \\
 & \leadsto_{DECIDE} & F :: (\dot{1}, \dot{2}) & (F|_{(1,2)} = \langle[\overline{3}, 4], [\overline{3}, 4]\rangle) \\
 & \leadsto_{DECIDE} & F :: (\dot{1}, \dot{2}, \dot{3}) & (F|_{(1,2,3)} = \langle[4], [4]\rangle) \\
 & \leadsto_{UNIT} & F :: (\dot{1}, \dot{2}, \dot{3}, 4) & (F|_{(1,2,3,4)} = \langle\rangle) \\
 & \leadsto_{SAT} & \text{SAT} &
\end{array}
$$

- We find

$$
\text{relevantL}(4, F :: (\dot{1}, \dot{2}, \dot{3}, 4)) = \{[\overline{1}, \overline{3}, 4], [\overline{2}, \overline{3}, 4]\}
$$

Multiple Conflict Clauses

- In the sequel
  - we will apply UNIT whenever possible,
  - we will prefer literals wrt their absolut value,
  - we prefer positive over negative literals.
- Consider $F = \langle [\overline{1}, 2], [\overline{1}, 3], [\overline{1}, 4], [\overline{2}, \overline{4}], [\overline{3}, \overline{4}] \rangle$.
- We obtain

$$\begin{array}{llll} F :: () & \leadsto_{DECIDE} & F :: (\dot{1}) & (F|_{(1)} = \langle [2], [3], [4], [\overline{2}, \overline{4}], [\overline{3}, \overline{4}] \rangle) \\ & \leadsto_{UNIT} & F :: (\dot{1}, 2) & (F|_{(1,2)} = \langle [3], [4], [\overline{4}], [\overline{3}, \overline{4}] \rangle) \\ & \leadsto_{UNIT} & F :: (\dot{1}, 2, 3) & (F|_{(1,2,3)} = \langle [4], [\overline{4}], [\overline{4}] \rangle) \\ & \leadsto_{UNIT} & F :: (\dot{1}, 2, 3, 4) & (F|_{(1,2,3,4)} = \langle [], [] \rangle) \end{array}$$

- There are two conflict clauses, viz. $[\overline{2}, \overline{4}]$ and $[\overline{3}, \overline{4}]$.

Multiple Conflicts

- **Recall** We will apply UNIT whenever possible.
- **Observation** Conflicts can only arise
  - after an application of UNIT to some $F :: J$ and
  - if $F|_J$ contains two unit clauses $[A]$ and $[\overline{A}]$.

Multiple Conflicts

- **Recall** We will apply UNIT whenever possible.
- **Observation** Conflicts can only arise
  - after an application of UNIT to some $F :: J$ and
  - if $F|_J$ contains two unit clauses $[A]$ and $[\overline{A}]$.
- Consider $F = \langle[\overline{1}, \overline{2}, 3], [\overline{2}, \overline{3}]\rangle$.
  - We obtain

$$\begin{array}{lllr} F :: () & \leadsto_{DECIDE} & F :: (\dot{1}) & (F|_{(1)} = \langle[\overline{2}, 3], [\overline{2}, \overline{3}]\rangle) \\ & \leadsto_{DECIDE} & F :: (\dot{1}, \dot{2}) & (F|_{(1,2)} = \langle[3], [\overline{3}]\rangle) \end{array}$$

Multiple Conflicts

- **Recall** We will apply UNIT whenever possible.
- **Observation** Conflicts can only arise
  - after an application of UNIT to some $F :: J$ and
  - if $F|_J$ contains two unit clauses $[A]$ and $[\overline{A}]$.
- Consider $F = \langle [\overline{1}, \overline{2}, 3], [\overline{2}, \overline{3}] \rangle$.
  - We obtain

$$\begin{array}{lll} F :: () & \leadsto_{DECIDE} \; F :: (\dot{1}) & (F|_{(1)} = \langle [\overline{2}, 3], [\overline{2}, \overline{3}] \rangle) \\ & \leadsto_{DECIDE} \; F :: (\dot{1}, \dot{2}) & (F|_{(1,2)} = \langle [3], [\overline{3}] \rangle) \end{array}$$

  - Two different conflicts are obtained
    - by applying unit wrt $[3]$ or $[\overline{3}]$
    - yielding the conflict clauses $[\overline{2}, \overline{3}]$ or $[\overline{1}, \overline{2}, 3]$, respectively.

Multiple Conflicts

- **Recall** We will apply UNIT whenever possible.
- **Observation** Conflicts can only arise
  - after an application of UNIT to some $F :: J$ and
  - if $F|_J$ contains two unit clauses $[A]$ and $[\overline{A}]$.
- Consider $F = \langle[\overline{1}, \overline{2}, 3], [\overline{2}, \overline{3}]\rangle$.
  - We obtain

$$\begin{array}{llll} F :: () & \leadsto_{DECIDE} & F :: (\dot{1}) & (F|_{(1)} = \langle[\overline{2}, 3], [\overline{2}, \overline{3}]\rangle) \\ & \leadsto_{DECIDE} & F :: (\dot{1}, \dot{2}) & (F|_{(1,2)} = \langle[3], [\overline{3}]\rangle) \end{array}$$

  - Two different conflicts are obtained
    - by applying unit wrt $[3]$ or $[\overline{3}]$
    - yielding the conflict clauses $[\overline{2}, \overline{3}]$ or $[\overline{1}, \overline{2}, 3]$, respectively.
- We assume that one particular conflict is selected.

Multiple Conflicts

- **Recall** We will apply UNIT whenever possible.
- **Observation** Conflicts can only arise
  - after an application of UNIT to some $F :: J$ and
  - if $F|_J$ contains two unit clauses $[A]$ and $[\overline{A}]$.
- Consider $F = \langle [\overline{1}, \overline{2}, 3], [\overline{2}, \overline{3}] \rangle$.
  - We obtain

$$\begin{array}{llll} F :: () & \leadsto_{DECIDE} & F :: (\dot{1}) & (F|_{(1)} = \langle [\overline{2}, 3], [\overline{2}, \overline{3}] \rangle) \\ & \leadsto_{DECIDE} & F :: (\dot{1}, \dot{2}) & (F|_{(1,2)} = \langle [3], [\overline{3}] \rangle) \end{array}$$

  - Two different conflicts are obtained
    - by applying unit wrt $[3]$ or $[\overline{3}]$
    - yielding the conflict clauses $[\overline{2}, \overline{3}]$ or $[\overline{1}, \overline{2}, 3]$, respectively.
- We assume that one particular conflict is selected.
- As we will see later, this selection is not very important.

Implication Graphs

- In the sequel we assume that $|\text{relevantL}(L, F :: J)| \leq 1$ for all L.
  - If $|\text{relevantL}(L, F :: J)| > 1$ for some L, then
    - all relevant clauses wrt L except one are deleted and
    - we write $\text{relevantL}(L, F :: J) = C$ instead of $\text{relevantL}(L, F :: J) = \{C\}$.
  - Different selections lead to different implication graphs.

Implication Graphs

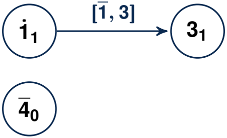
- In the sequel we assume that $|\text{relevantL}(L, F :: J)| \leq 1$ for all L.
  - If $|\text{relevantL}(L, F :: J)| > 1$ for some L, then
    - all relevant clauses wrt L except one are deleted and
    - we write $\text{relevantL}(L, F :: J) = C$ instead of $\text{relevantL}(L, F :: J) = \{C\}$.
  - Different selections lead to different implication graphs.
- Let F be a formula in CNF and J a partial interpretation.
- The **explanation** of L' is the set $\{\overline{L} \mid L \in \text{relevantL}(L', F :: J) \setminus \{L'\}\}$
- An **implication graph for $F :: J$** is a graph $(\mathcal{V}, \mathcal{E})$, where
  - $\mathcal{V} = J$,
  - $\mathcal{E} = \{(L, L') \in \mathcal{V} \times \mathcal{V} \mid$ there exists $C = \text{relevantL}(L', F :: J)$ and $\overline{L} \in (C \setminus \{L'\}\}$,
  - each $(L, L') \in \mathcal{E}$ is directed from L to L'.

Implication Graphs

- In the sequel we assume that $|\text{relevantL}(L, F :: J)| \leq 1$ for all L.
  - If $|\text{relevantL}(L, F :: J)| > 1$ for some L, then
    - all relevant clauses wrt L except one are deleted and
    - we write $\text{relevantL}(L, F :: J) = C$ instead of $\text{relevantL}(L, F :: J) = \{C\}$.
  - Different selections lead to different implication graphs.
- Let F be a formula in CNF and J a partial interpretation.
- The explanation of L' is the set $\{\overline{L} \mid L \in \text{relevantL}(L', F :: J) \setminus \{L'\}\}$
- An implication graph for $F :: J$ is a graph $(\mathcal{V}, \mathcal{E})$, where
  - $\mathcal{V} = J$,
  - $\mathcal{E} = \{(L, L') \in \mathcal{V} \times \mathcal{V} \mid$ there exists $C = \text{relevantL}(L', F :: J)$ and $\overline{L} \in (C \setminus \{L'\}\}$,
  - each $(L, L') \in \mathcal{E}$ is directed from L to L'.
- It is sometimes convenient
  - to label a vertex by its decision level in J (depicted as subscript)
  - to label an edge by the clause who caused it (the reason).

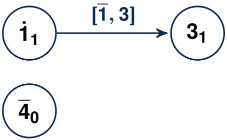
Implication Graphs for an Old Example

- Let $F = \langle [1, 2], [2, \overline{3}], [\overline{2}, \overline{3}, 4], [\overline{1}, 3], [\overline{4}] \rangle$
  - Consider $J_1 = (\overline{4}, \dot{1}, 3)$
    - relevant$(L, F :: J_1) = \{[\overline{4}], [\overline{1}, 3]\}$

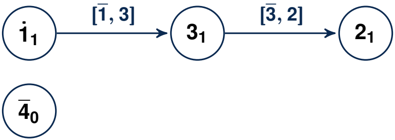


Implication Graphs for an Old Example

- Let $F = \langle [1,2], [2,\overline{3}], [\overline{2},\overline{3},4], [\overline{1},3], [\overline{4}] \rangle$
  - Consider $J_1 = (\overline{4}, \dot{1}, 3)$
    - relevant$(L, F :: J_1) = \{[\overline{4}], [\overline{1}, 3]\}$

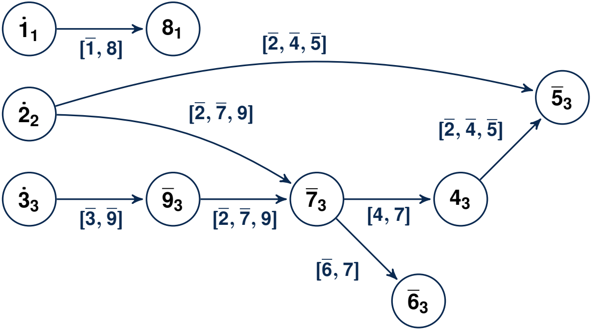


  - Consider $J_2 = (\overline{4}, \dot{1}, 3, 2)$
    - relevant$(L, F :: J_2) = \{[\overline{4}], [\overline{1}, 3], [\overline{3}, 2]\}$



Implication Graphs – Another Example

- Let $F = \langle[\overline{1}, 8], [\overline{2}, \overline{4}, \overline{5}], [\overline{3}, \overline{9}], [\overline{2}, \overline{7}, 9], [4, 7], [\overline{6}, 7]\rangle$
- Let $J = (\overline{5}, \overline{6}, 4, \overline{7}, \overline{9}, \dot{3}, \dot{2}, 8, \dot{1})$.

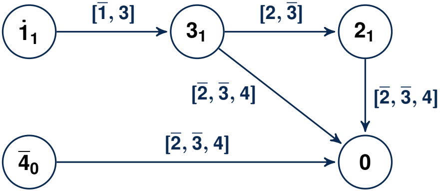


Conflict Graphs

- Consider $F :: J$ and let $(\mathcal{V}, \mathcal{E})$ be the implication graph for $F :: J$.
- Let $C \in F$ and suppose $C|_J = [\,]$.
- $(\mathcal{V}', \mathcal{E}')$ is the conflict graph for $F :: J$ and C if
  - $\mathcal{V}' = \mathcal{V} \cup \{0\}$ and
  - $\mathcal{E}' = \mathcal{E} \cup \{(L, 0) \mid L \in \mathcal{V}$ and $\overline{L} \in C\}$.
- 0 is called conflict node.
- Conflict graphs are acyclic.

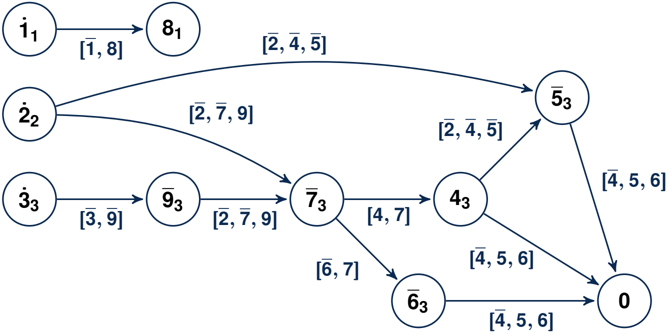
The Conflict Graph of an Old Example

- Let $F = \langle[1, 2], [2, \overline{3}], [\overline{2}, \overline{3}, 4], [\overline{1}, 3], [\overline{4}]\rangle$.
- Consider $J = ({\color{orange}2}, 3, \dot{1}, \overline{4})$.
- **Note** $[\overline{2}, \overline{3}, 4]|_J = [\,]$.



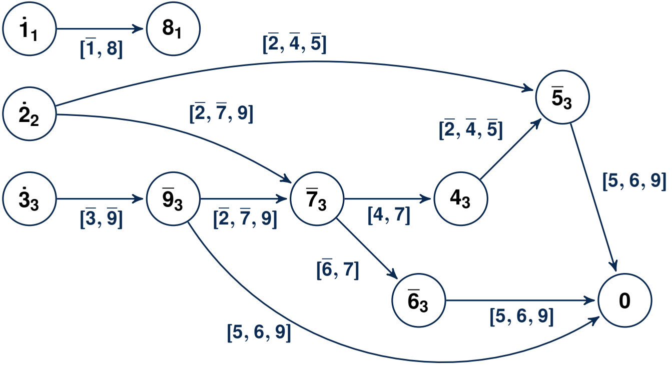
Conflict Graphs – Another Example (1)

- Let $F = \langle [\overline{1}, 8], [\overline{2}, \overline{4}, \overline{5}], [\overline{3}, \overline{9}], [\overline{2}, \overline{7}, 9], [4, 7], [\overline{6}, 7], [\overline{4}, 5, 6], [5, 6, 9] \rangle$.
- Let $J = (\overline{5}, \overline{6}, 4, \overline{7}, \overline{9}, \dot{3}, \dot{2}, 8, \dot{1})$.
- **Note** $[\overline{4}, 5, 6]|_J = [\,]$.



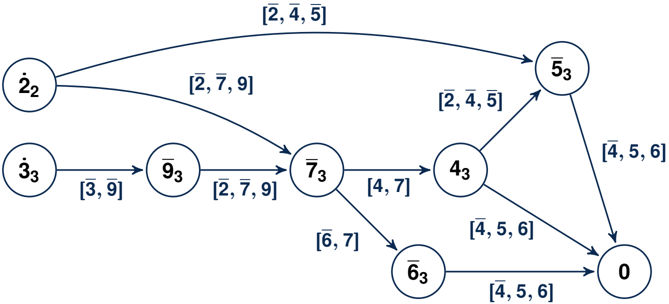
Conflict Graphs – Another Example (2)

- **Let** $F = \langle[\overline{1}, 8], [\overline{2}, \overline{4}, \overline{5}], [\overline{3}, \overline{9}], [\overline{2}, \overline{7}, 9], [4, 7], [\overline{6}, 7], [\overline{4}, 5, 6], [5, 6, 9]\rangle$.
- **Let** $J = (\overline{5}, \overline{6}, 4, \overline{7}, \overline{9}, \dot{3}, \dot{2}, 8, \dot{1})$.
- **Note** There is another conflict clause $[5, 6, 9]|_J = [\,]$.



Reduced Conflict Graph

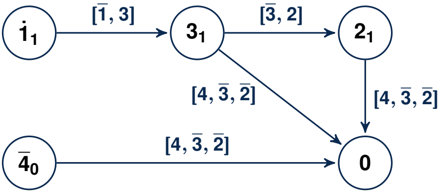
- Let $(\mathcal{V}, \mathcal{E})$ be a conflict graph.
- The reduced conflict graph of $(\mathcal{V}, \mathcal{E})$ is the subgraph of $(\mathcal{V}, \mathcal{E})$ containing all vertices which are connected to 0 and all edges between these vertces.



- In the sequel, we consider only reduced conflict graphs.

Paths

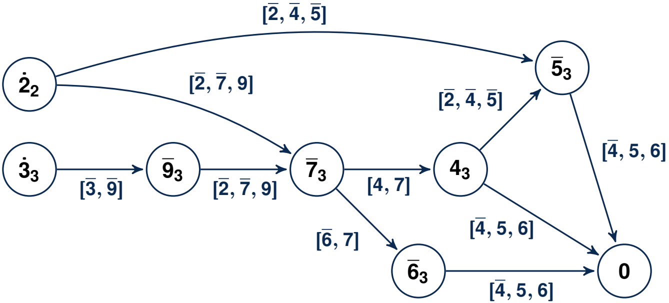
- Let $(\mathcal{V}, \mathcal{E})$ be a reduced conflict graph.
- Let $\mathcal{V}_S = \{L \in \mathcal{V} \mid L$ is a decision literal or its decision level is $0\}$.
- Let $paths(\mathcal{V}, \mathcal{E})$ be the set of all directed paths from nodes in $\mathcal{V}_S$ to 0 in $(\mathcal{V}, \mathcal{E})$.
- Consider



  - $\mathcal{V}_S = \{1, \overline{4}\}$.
  - $paths(\mathcal{V}, \mathcal{E}) = \{(1, 3, 2, 0), (1, 3, 0), (\overline{4}, 0)\}$.

Paths – Another Example

- **Consider**



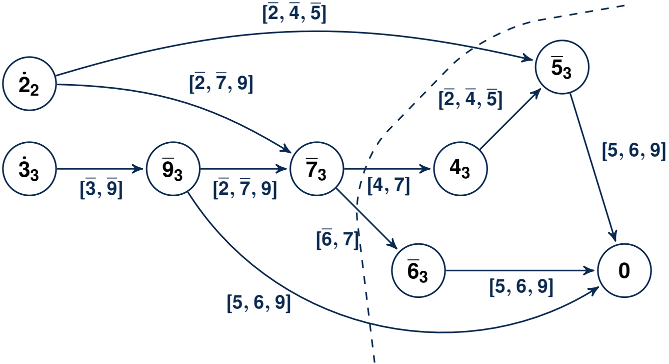
  - $\mathcal{V}_S = \{2, 3\}$
  - $paths(\mathcal{V}, \mathcal{E}) = \{ (2, \overline{5}, 0), (2, \overline{7}, 4, \overline{5}, 0), (2, \overline{7}, 4, 0), (2, \overline{7}, \overline{6}, 0),$
    $(3, \overline{9}, \overline{7}, 4, \overline{5}, 0), (3, \overline{9}, \overline{7}, 4, 0), (3, \overline{9}, \overline{7}, \overline{6}, 0)\}$

Cuts

- Let $(\mathcal{V}, \mathcal{E})$ be a reduced conflict graph.
- A cut $(\mathcal{V}_R, \mathcal{V}_C)$ through $(\mathcal{V}, \mathcal{E})$ is a partition of $\mathcal{V}$ into $\mathcal{V}_R$ and $\mathcal{V}_C$ such that
  - $\mathcal{V}_R \cap \mathcal{V}_C = \emptyset$,
  - $\mathcal{V}_R \cup \mathcal{V}_C = \mathcal{V}$,
  - $\mathcal{V}_S \subseteq \mathcal{V}_R$,
  - $\{0\} \subseteq \mathcal{V}_C$, and
  - each $p \in paths(\mathcal{V}, \mathcal{E})$ is partitioned into two subpaths p_R and p_C such that
    - p_R and p_C have no vertex in common,
    - p is obtained by adding an edge from the end of p_R to the start of p_C,
    - $\mathcal{V}_R$ contains all vertices occurring in p_R, and
    - $\mathcal{V}_C$ contains all vertices occurring in p_C.

Cuts – Example

- **Consider**



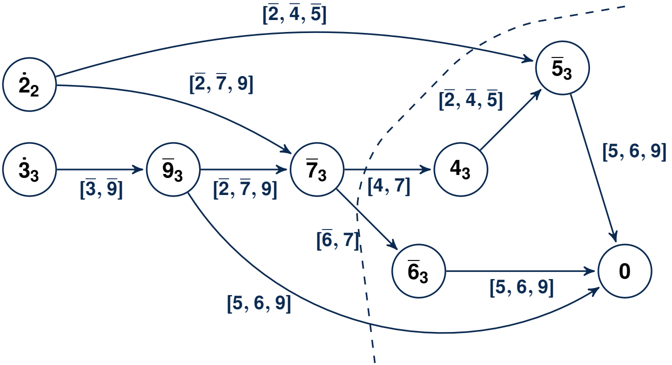
$$\mathcal{V}_R = (\{2, 3, \overline{7}, \overline{9}\}, \mathcal{V}_C = \{4, \overline{5}, \overline{6}, 0\})$$

Cut Clauses

- Let $(\mathcal{V}_R, \mathcal{V}_C)$ be a cut through a reduced conflict graph $(\mathcal{V}, \mathcal{E})$.
- Let $(\mathcal{V}_R, \mathcal{E}_R)$ be the subgraph of $(\mathcal{V}, \mathcal{E})$
  which consists only of the vertices $\mathcal{V}_R$ and edges between elements of $\mathcal{V}_R$.
- Let $\mathcal{V}'_R$ be the subset of $\mathcal{V}_R$ containing all vertices,
  where an outgoing edge was cut by $(\mathcal{V}_R, \mathcal{V}_C)$.
- The cut clause C_R determined by $(\mathcal{V}_R, \mathcal{V}_C)$ is the clause $\overline{\mathcal{V}'_R}$.

Cut Clauses – Example

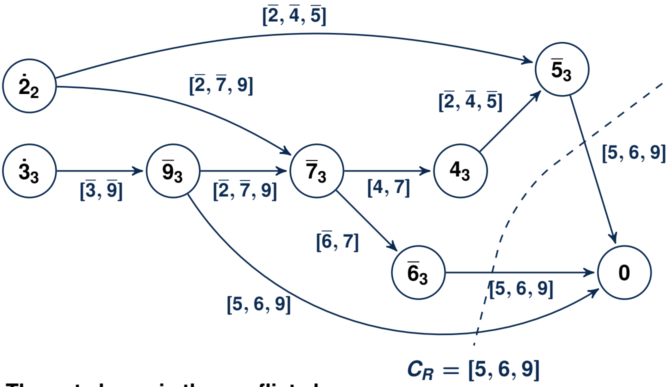
- **Consider**



$$C_R = [\overline{2}, 7, 9]$$

Initial Cuts

- Let $(\mathcal{V}, \mathcal{E})$ be a reduced conflict graph.
- Let $(\mathcal{V}_R, \mathcal{V}_C)$ be a cut through $(\mathcal{V}, \mathcal{E})$.
- $(\mathcal{V}_R, \mathcal{V}_C)$ is an initial cut if $\mathcal{V}_C = \{0\}$.

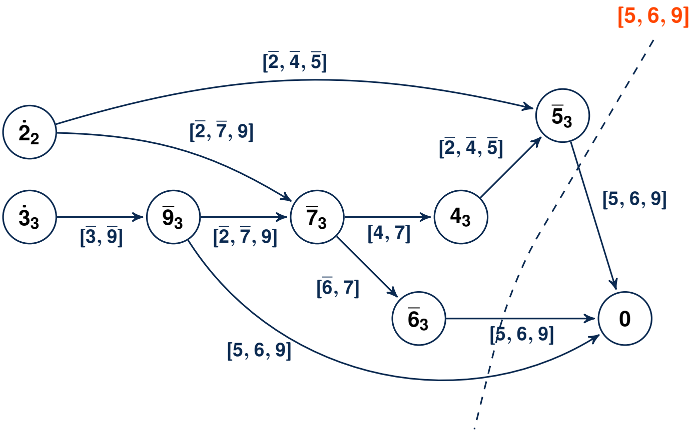


- **Note** The cut clause is the conflict clause.

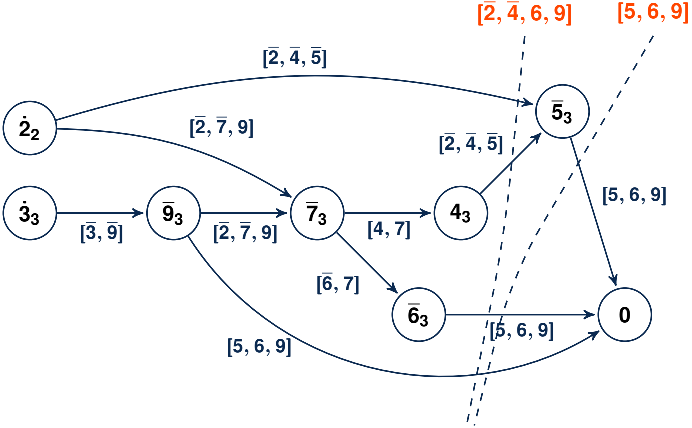
Shifted Cuts

- Let $(\mathcal{V}, \mathcal{E})$ be a reduced conflict graph.
- Recall $\mathcal{V}_S = \{L \in \mathcal{V} \mid L$ is a decision literal or its decision level is $0\}$.
- Let $(\mathcal{V}_R, \mathcal{V}_C)$ be a cut through $(\mathcal{V}, \mathcal{E})$.
- Let $(\mathcal{V}_R, \mathcal{E}_R)$ be the subgraph of $(\mathcal{V}, \mathcal{E})$ which consists only of the vertices $\mathcal{V}_R$ and edges between elements of $\mathcal{V}_R$.
- Let $L \in \mathcal{V}_R \setminus \mathcal{V}_S$ such that the outdegree of L in $(\mathcal{V}_R, \mathcal{E}_R)$ is 0.
- $(\mathcal{V}_R \setminus \{L\}, \mathcal{V}_C \cup \{L\})$ is the cut obtained from $(\mathcal{V}_R, \mathcal{V}_C)$ by shifting L.

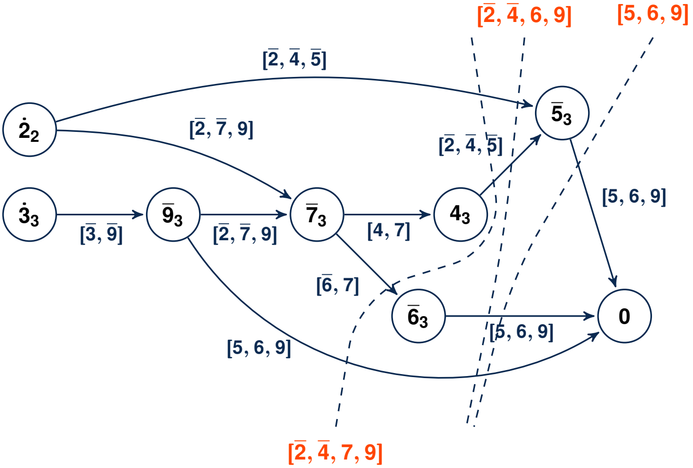
Shifted Cuts – Examples



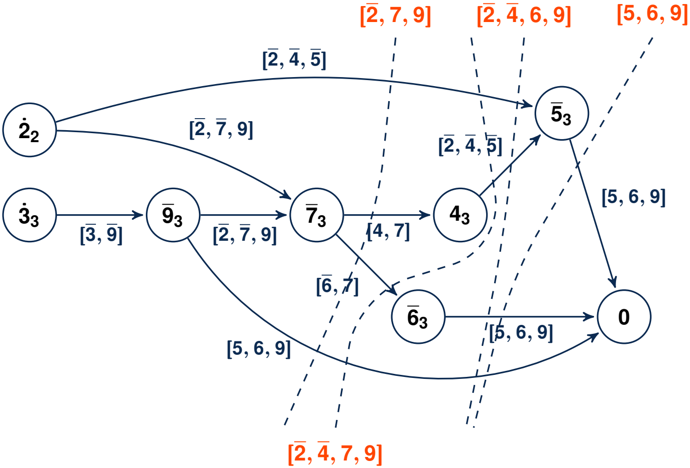
Shifted Cuts – Examples



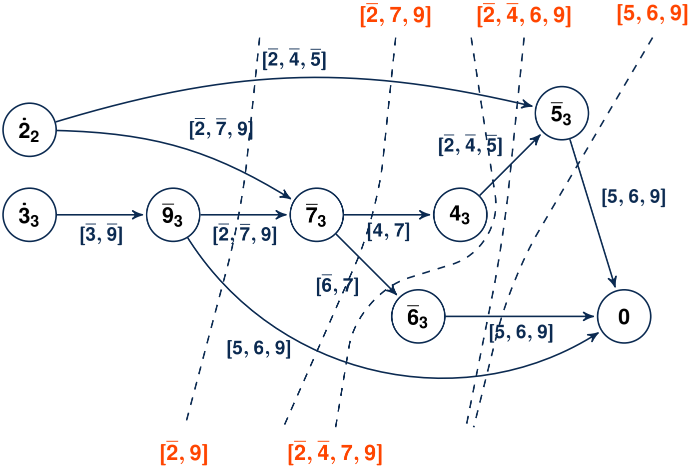
Shifted Cuts – Examples



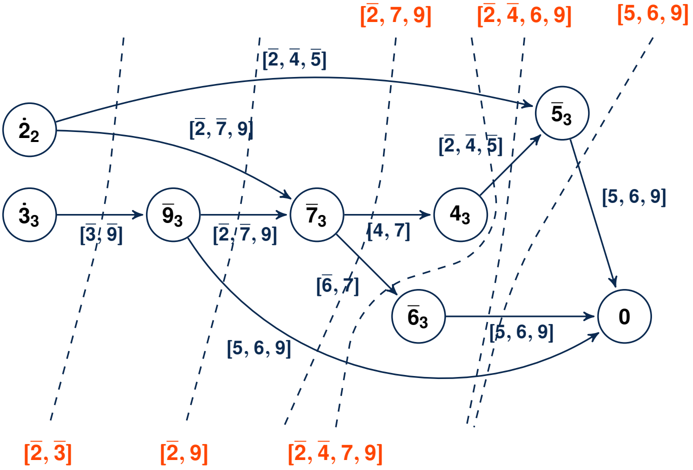
Shifted Cuts – Examples



Shifted Cuts – Examples



Shifted Cuts – Examples



Corresponding Linear Resolution Derivation

- We obtain

1	$[\overline{3}, \overline{9}]$	relevant clause
2	$[\overline{2}, \overline{7}, 9]$	relevant clause
3	$[4, 7]$	relevant clause
4	$[\overline{6}, 7]$	relevant clause
5	$[\overline{2}, \overline{4}, \overline{5}]$	relevant clause
6	$[5, 6, 9]$	conflict clause
7	$[\overline{2}, \overline{4}, 6, 9]$	res(6,5)
8	$[\overline{2}, \overline{4}, 7, 9]$	res(7,4)
9	$[\overline{2}, 7, 9]$	res(8,3)
10	$[\overline{2}, 9]$	res(9,2)
11	$[\overline{2}, \overline{3}]$	res(10,1)

Corresponding Linear Resolution Derivation

- **We obtain**

1	$[\overline{3}, \overline{9}]$	relevant clause
2	$[\overline{2}, \overline{7}, 9]$	relevant clause
3	$[4, 7]$	relevant clause
4	$[\overline{6}, 7]$	relevant clause
5	$[\overline{2}, \overline{4}, \overline{5}]$	relevant clause
6	$[5, 6, 9]$	conflict clause
7	$[\overline{2}, \overline{4}, 6, 9]$	res(6,5)
8	$[\overline{2}, \overline{4}, 7, 9]$	res(7,4)
9	$[\overline{2}, 7, 9]$	res(8,3)
10	$[\overline{2}, 9]$	res(9,2)
11	$[\overline{2}, \overline{3}]$	res(10,1)

- **Observation** **Cut-clauses corresponding to a cut, which was generated by a sequence of shifts from the initial cut, can also be generated by linear resolution derivations from the conflict clause using relevant clauses.**

Unique Implication Points

- Let $(\mathcal{V}, \mathcal{E})$ be a (reduced) conflict graph.
- Let $L^* \in \mathcal{V}$ be the decision literal with the highest decision level in $\mathcal{V}$.
  - L^* is unique.
- Let $paths^*(\mathcal{V}, \mathcal{E})$ be the set of all directed paths from L^* to 0 in $(\mathcal{V}, \mathcal{E})$.
  - $paths^*(\mathcal{V}, \mathcal{E}) \subseteq paths(\mathcal{V}, \mathcal{E})$.

Unique Implication Points

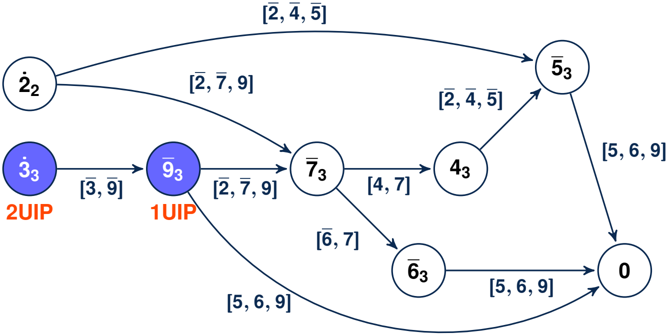
- Let $(\mathcal{V}, \mathcal{E})$ be a (reduced) conflict graph.
- Let L^* $\in \mathcal{V}$ be the decision literal with the highest decision level in $\mathcal{V}$.
  - L^* is unique.
- Let $paths^*(\mathcal{V}, \mathcal{E})$ be the set of all directed paths from L^* to 0 in $(\mathcal{V}, \mathcal{E})$.
  - $paths^*(\mathcal{V}, \mathcal{E}) \subseteq paths(\mathcal{V}, \mathcal{E})$.
- $L \in \mathcal{V}$ is a unique implication point (UIP)
  if it occurs in all paths in $paths^*(\mathcal{V}, \mathcal{E})$.
  - L^* is always a UIP.
  - The decision level of an UIP is equal to the decision level of L^*.

Unique Implication Points

- Let $(\mathcal{V}, \mathcal{E})$ be a (reduced) conflict graph.
- Let L^* $\in \mathcal{V}$ be the decision literal with the highest decision level in $\mathcal{V}$.
  - L^* is unique.
- Let $paths^*(\mathcal{V}, \mathcal{E})$ be the set of all directed paths from L^* to 0 in $(\mathcal{V}, \mathcal{E})$.
  - $paths^*(\mathcal{V}, \mathcal{E}) \subseteq paths(\mathcal{V}, \mathcal{E})$.
- $L \in \mathcal{V}$ is a unique implication point (UIP)
  if it occurs in all paths in $paths^*(\mathcal{V}, \mathcal{E})$.
  - L^* is always a UIP.
  - The decision level of an UIP is equal to the decision level of L^*.
- The UIPs of a reduced conflict graph can be ordered:
  - The 1UIP is the UIP which is closest to 0.
  - The nUIP is the UIP which is n-closest to 0.

Unique implication Points – Example

- **Consider**



UIP Clauses

- **Let $(\mathcal{V}, \mathcal{E})$ be a (reduced) conflict graph.**
- **Let $(\mathcal{V}_R, \mathcal{V}_C)$ be a cut through $(\mathcal{V}, \mathcal{E})$.**
- **Let C be the cut clause determined by $(\mathcal{V}_R, \mathcal{V}_C)$.**

UIP Clauses

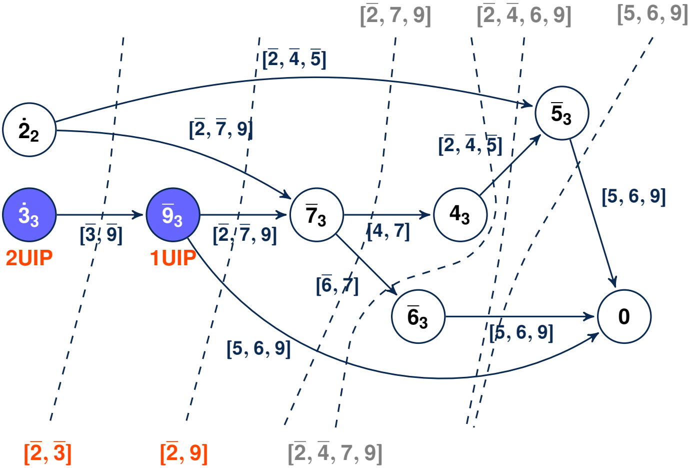
- Let $(\mathcal{V}, \mathcal{E})$ be a (reduced) conflict graph.
- Let $(\mathcal{V}_R, \mathcal{V}_C)$ be a cut through $(\mathcal{V}, \mathcal{E})$.
- Let C be the cut clause determined by $(\mathcal{V}_R, \mathcal{V}_C)$.
- C is a UIP clause if
  - C contains a literal L such that $\overline{L}$ is a UIP in $(\mathcal{V}, \mathcal{E})$ and
  - there is no literal in C whose level is identical to the level of L.

UIP Clauses

- Let $(\mathcal{V}, \mathcal{E})$ be a (reduced) conflict graph.
- Let $(\mathcal{V}_R, \mathcal{V}_C)$ be a cut through $(\mathcal{V}, \mathcal{E})$.
- Let C be the cut clause determined by $(\mathcal{V}_R, \mathcal{V}_C)$.
- C is a UIP clause if
  - C contains a literal L such that $\overline{L}$ is a UIP in $(\mathcal{V}, \mathcal{E})$ and
  - there is no literal in C whose level is identical to the level of L.
- C is a nUIP clause if
  - if C is a UIP clause and
  - if $L \in C$ such that $\overline{L}$ is a UIP in $(\mathcal{V}, \mathcal{E})$ then $\overline{L}$ is the nUIP in $(\mathcal{V}, \mathcal{E})$.

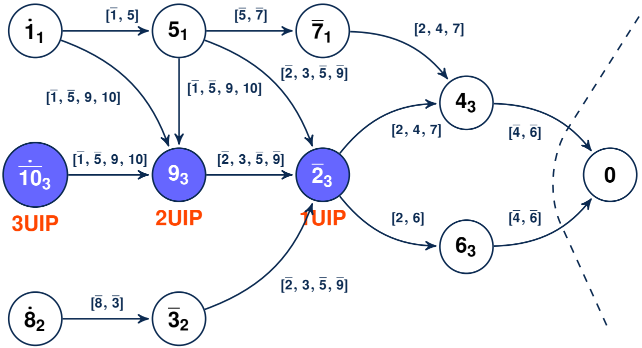
UIP Clauses – Examples

- **Consider**



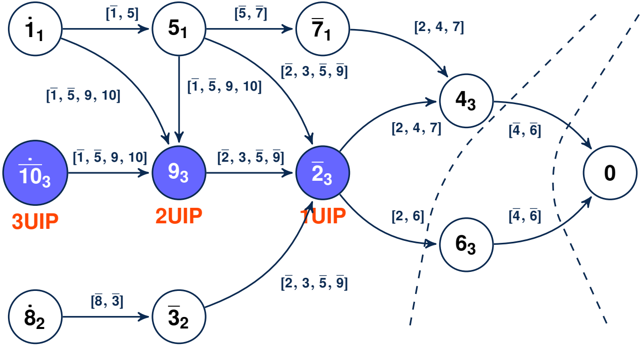
Yet Another Example

- Let $F = \langle [2, 6], [\bar{8}, \bar{3}], [\bar{4}, \bar{6}], [\bar{1}, \bar{5}, 9, 10], [2, 4, 7], [\bar{1}, 5], [\bar{2}, 3, \bar{5}, \bar{9}], [\bar{5}, \bar{7}] \rangle$.
- Let $J = (6, 4, \bar{2}, 9, \dot{\overline{10}}, \bar{3}, \dot{8}, \bar{7}, 5, \dot{\bar{1}})$.



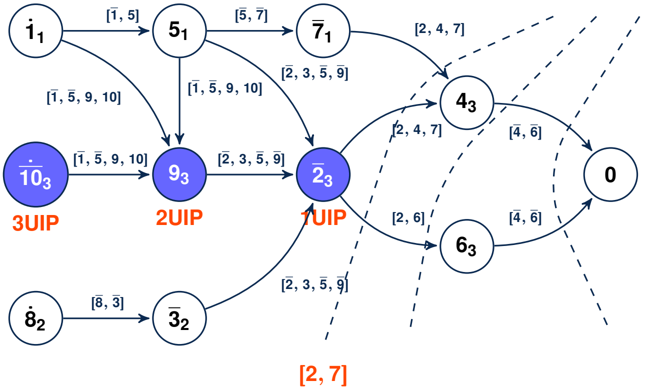
Yet Another Example

- Let $F = \langle [2, 6], [\overline{8}, \overline{3}], [\overline{4}, \overline{6}], [\overline{1}, \overline{5}, 9, 10], [2, 4, 7], [\overline{1}, 5], [\overline{2}, 3, \overline{5}, \overline{9}], [\overline{5}, \overline{7}] \rangle$.
- Let $J = (6, 4, \overline{2}, 9, \dot{\overline{10}}, \overline{3}, \dot{8}, \overline{7}, 5, \dot{1})$.



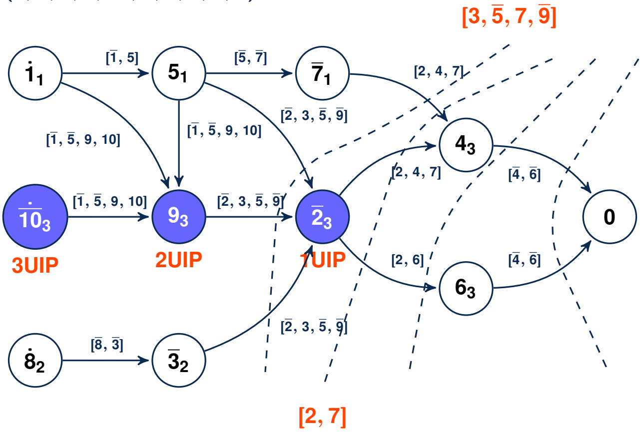
Yet Another Example

- Let $F = \langle [2,6], [\overline{8},\overline{3}], [\overline{4},\overline{6}], [\overline{1},\overline{5},9,10], [2,4,7], [\overline{1},5], [\overline{2},3,\overline{5},\overline{9}], [\overline{5},\overline{7}] \rangle$.
- Let $J = (6, 4, \overline{2}, 9, \dot{\overline{10}}, \overline{3}, \dot{8}, \overline{7}, 5, \dot{1})$.



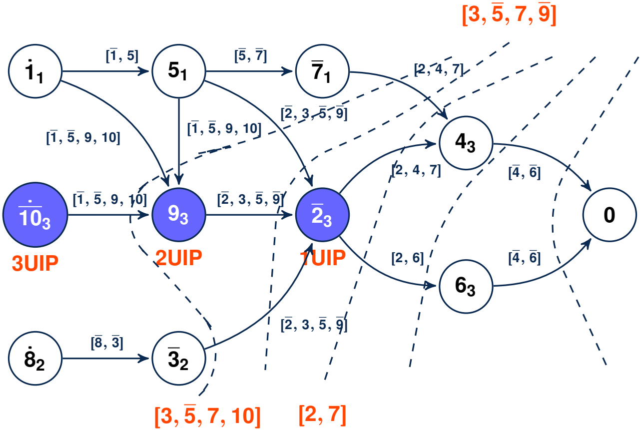
Yet Another Example

- Let $F = \langle[2, 6], [\overline{8}, \overline{3}], [\overline{4}, \overline{6}], [\overline{1}, \overline{5}, 9, 10], [2, 4, 7], [\overline{1}, 5], [\overline{2}, 3, \overline{5}, \overline{9}], [\overline{5}, \overline{7}]\rangle$.
- Let $J = (6, 4, \overline{2}, 9, \dot{\overline{10}}, \overline{3}, \dot{8}, \overline{7}, 5, \dot{\overline{1}})$.



Yet Another Example

- Let $F = \langle [2, 6], [\overline{8}, \overline{3}], [\overline{4}, \overline{6}], [\overline{1}, \overline{5}, 9, 10], [2, 4, 7], [\overline{1}, 5], [\overline{2}, 3, \overline{5}, \overline{9}], [\overline{5}, \overline{7}] \rangle$.
- Let $J = (6, 4, \overline{2}, 9, \dot{\overline{10}}, \overline{3}, \dot{8}, \overline{7}, 5, \dot{1})$.



UIP Clauses – Remarks

- UIP clauses are not unique.
  - In the previous example, $[2, 7]$ and $[2, \overline{5}]$ are 1UIP clauses.

UIP Clauses – Remarks

- **UIP clauses are not unique.**
  - **In the previous example, $[2, 7]$ and $[2, \overline{5}]$ are 1UIP clauses.**
- **Let**
  - ***C* be a UIP clause,**
  - ***m* the level of the literal whose complement is UIP, and**
  - ***n* the next highest level assigned to some literal in *C*.**

  An application of CDBL to $F :: J$ using *C* removes all literals from *J* whose level is higher than *n*.

UIP Clauses – Remarks

- **UIP clauses are not unique.**
  - **In the previous example, $[2, 7]$ and $[2, \overline{5}]$ are 1UIP clauses.**
- **Let**
  - **C be a UIP clause,**
  - **m the level of the literal whose complement is UIP, and**
  - **n the next highest level assigned to some literal in C.**

  An application of CDBL to $F :: J$ using C removes all literals from J whose level is higher than n.
- **Most modern SAT-solvers learn 1UIP clauses.**
- **If several 1UIP clauses exist, then often the shortest one is prefered.**

Conflict Graphs vs Resolution Derivations

- **Consider the following linear resolution derivation:**

1	$[\overline{1}, 5]$	relevant clause
2	$[\overline{5}, \overline{7}]$	relevant clause
3	$[\overline{8}, \overline{3}]$	relevant clause
4	$[\overline{1}, \overline{5}, 9, 10]$	relevant clause
5	$[\overline{2}, 3, \overline{5}, \overline{9}]$	relevant clause
6	$[2, 4, 7]$	relevant clause
7	$[2, 6]$	relevant clause
8	$[\overline{4}, \overline{6}]$	conflict clause
9	$[2, \overline{4}]$	res(8,7)
10	$[2, 7]$	res(9,6)
11	$[2, \overline{5}]$	res(10,2)
12	$[\overline{1}, 2]$	res(12,1)

- **Note $[\overline{1}, 2]$ is not a cut-clause! (for the given graph)**
- **Pipatsriawat, Darwiche: On Modern Clause-Learning Satisfiability Solvers. Journal Automated Reasoning 44, 277-301:2010.**

- **How can we construct a graph, such that $[\overline{1}, 2]$ is a cut clause?**

How to make $[\overline{1}, 2]$ a cut clause

- Let $F = \langle[2, 6], [\overline{8}, \overline{3}], [\overline{4}, \overline{6}], [\overline{1}, \overline{5}, 9, 10], [2, 4, 7], [\overline{1}, 5], [\overline{2}, 3, \overline{5}, \overline{9}], [\overline{5}, \overline{7}]\rangle$.
- Let $J = ?$.