universität innsbruck

SS 2024 lecture 12

Logic

Diana Gründlinger Aart Middeldorp Fabian Mitterwallner

Alexander Montag Johannes Niederhauser Daniel Rainer

Outline

Definitions

- path $s_1 \to s_2 \to \cdots$ is fair with respect to set C of CTL formulas if for all $\psi \in C$ $s_i \vDash \psi$ for infinitely many i
- A_C (E_C) denotes A (E) restricted to paths that are fair with respect to C

Lemma

$$\mathsf{E}_C[\varphi\,\mathsf{U}\,\psi] \equiv \mathsf{E}[\varphi\,\mathsf{U}\,(\psi \wedge \mathsf{E}_C\mathsf{G}\,\top)] \qquad \mathsf{E}_C\mathsf{X}\,\varphi \equiv \mathsf{EX}(\varphi \wedge \mathsf{E}_C\mathsf{G}\,\top)$$

Theorem

set of temporal connectives is adequate for CTL $\iff$

$$\text{it contains} \begin{cases} \text{at least one of } \{\mathsf{AX}, \mathsf{EX}\} \\ \text{at least one of } \{\mathsf{EG}, \mathsf{AF}, \mathsf{AU}\} \\ \mathsf{EU} \end{cases}$$

Theorem

- $\{\mathsf{X}, \mathsf{U}\}$, $\{\mathsf{X}, \mathsf{W}\}$ and $\{\mathsf{X}, \mathsf{R}\}$ are adequate sets of temporal connectives for LTL
- $\{\mathsf{U}, \mathsf{R}\}$, $\{\mathsf{U}, \mathsf{W}\}$, $\{\mathsf{U}, \mathsf{G}\}$, $\{\mathsf{F}, \mathsf{W}\}$ and $\{\mathsf{F}, \mathsf{R}\}$ are adequate sets of temporal connectives for LTL fragment consisting of negation-normal forms without X

LTL Model Checking

$\mathcal{M}, s \vDash \varphi$?

- construct labelled Büchi automaton $A_{\neg\varphi}$ for $\neg\varphi$
- combine $A_{\neg\varphi}$ and $\mathcal{M}$ into single automaton $A_{\neg\varphi} \times \mathcal{M}$
- determine whether there exists accepting path π in $A_{\neg\varphi} \times \mathcal{M}$ starting from s

Theorem

$\mathcal{M}, s \nvDash \varphi \iff$ exists accepting path in $A_{\neg\varphi} \times \mathcal{M}$ starting from state corresponding to s

Part I: Propositional Logic

algebraic normal forms, binary decision diagrams, conjunctive normal forms, DPLL, Horn formulas, natural deduction, Post's adequacy theorem, resolution, SAT, semantics, sorting networks, soundness and completeness, syntax, Tseitin's transformation

Part II: Predicate Logic

natural deduction, quantifier equivalences, resolution, semantics, Skolemization, syntax, undecidability, unification

Part III: Model Checking

adequacy, branching-time temporal logic, CTL*, fairness, linear-time temporal logic, model checking algorithms, symbolic model checking

Outline

Definition

CTL* formulas consist of

- state formulas, which are evaluated in states:

$$\varphi \ ::= \ \bot \mid \top \mid p \mid (\neg\varphi) \mid (\varphi\wedge\varphi) \mid (\varphi\vee\varphi) \mid (\varphi\rightarrow\varphi) \mid \mathsf{A}[\alpha] \mid \mathsf{E}[\alpha]$$

- path formulas, which are evaluated along paths:

$$\alpha \ ::= \ \varphi \mid (\neg\alpha) \mid (\alpha\wedge\alpha) \mid (\alpha\vee\alpha) \mid (\alpha\rightarrow\alpha) \mid (\mathsf{X}\,\alpha) \mid (\mathsf{F}\,\alpha) \mid (\mathsf{G}\,\alpha) \mid (\alpha\,\mathsf{U}\,\alpha)$$

Examples

$\mathsf{A}[\,(p\,\mathsf{U}\,r)\vee(q\,\mathsf{U}\,r)]$	$\mathsf{A}[\mathsf{X}\,p\vee\mathsf{X}\,\mathsf{X}\,p]$	$\mathsf{E}[\mathsf{G}\,\mathsf{F}\,p]$
$\mathsf{A}[\,(p\vee q)\,\mathsf{U}\,r]$	$\mathsf{A}[\mathsf{X}\,p]\vee\mathsf{A}[\mathsf{X}\,\mathsf{A}[\mathsf{X}\,p]]$	$\mathsf{E}[\mathsf{G}\,\mathsf{E}[\mathsf{F}\,p]]$

Definition

satisfaction of CTL* state formula φ in state $s\in S$ of model $\mathcal{M}=(S,\rightarrow,L)$

$$\begin{array}{lll}
\mathcal{M},s \nvDash \bot & & \\
\mathcal{M},s \vDash \top & & \\
\mathcal{M},s \vDash p & \iff & p\in L(s) \\
\mathcal{M},s \vDash \neg\varphi & \iff & \mathcal{M},s \nvDash \varphi \\
\mathcal{M},s \vDash \varphi\wedge\psi & \iff & \mathcal{M},s \vDash \varphi \text{ and } \mathcal{M},s \vDash \psi \\
\mathcal{M},s \vDash \varphi\vee\psi & \iff & \mathcal{M},s \vDash \varphi \text{ or } \mathcal{M},s \vDash \psi \\
\mathcal{M},s \vDash \varphi\rightarrow\psi & \iff & \mathcal{M},s \nvDash \varphi \text{ or } \mathcal{M},s \vDash \psi \\
\mathcal{M},s \vDash \mathsf{A}[\alpha] & \iff & \forall \text{ paths } \pi = s\rightarrow s_2\rightarrow\cdots \ \ \mathcal{M},\pi \vDash \alpha \\
\mathcal{M},s \vDash \mathsf{E}[\alpha] & \iff & \exists \text{ path } \pi = s\rightarrow s_2\rightarrow\cdots \ \ \mathcal{M},\pi \vDash \alpha
\end{array}$$

Definition

satisfaction of CTL* path formula α with respect to path $\pi = s_1 \to s_2 \to \cdots$ in $\mathcal{M} = (S, \to, L)$

$$
\begin{array}{lcl}
\mathcal{M}, \pi \vDash \varphi & \iff & \mathcal{M}, s_1 \vDash \varphi \\
\mathcal{M}, \pi \vDash \neg\alpha & \iff & \mathcal{M}, \pi \nvDash \alpha \\
\mathcal{M}, \pi \vDash \alpha \wedge \beta & \iff & \mathcal{M}, \pi \vDash \alpha \text{ and } \mathcal{M}, \pi \vDash \beta \\
\mathcal{M}, \pi \vDash \alpha \vee \beta & \iff & \mathcal{M}, \pi \vDash \alpha \text{ or } \mathcal{M}, \pi \vDash \beta \\
\mathcal{M}, \pi \vDash \alpha \to \beta & \iff & \mathcal{M}, \pi \nvDash \alpha \text{ or } \mathcal{M}, \pi \vDash \beta \\
\mathcal{M}, \pi \vDash \mathsf{X}\,\alpha & \iff & \mathcal{M}, \pi^2 \vDash \alpha \\
\mathcal{M}, \pi \vDash \mathsf{F}\,\alpha & \iff & \exists\, i \geqslant 1 \ \mathcal{M}, \pi^i \vDash \alpha \\
\mathcal{M}, \pi \vDash \mathsf{G}\,\alpha & \iff & \forall\, i \geqslant 1 \ \mathcal{M}, \pi^i \vDash \alpha \\
\mathcal{M}, \pi \vDash \alpha\,\mathsf{U}\,\beta & \iff & \exists\, i \geqslant 1 \ \mathcal{M}, \pi^i \vDash \beta \text{ and } \forall\, j < i \ \mathcal{M}, \pi^j \vDash \alpha
\end{array}
$$

Theorem

satisfaction of CTL* formulas in finite models is decidable

Definition

CTL* state (CTL, LTL) formulas φ and ψ are semantically equivalent if

$$\mathcal{M}, s \vDash \varphi \iff \mathcal{M}, s \vDash \psi$$

for all models $\mathcal{M} = (S, \to, L)$ and states $s \in S$

Remarks

- LTL formula α is equivalent to CTL* formula $\mathsf{A}[\alpha]$
- CTL is fragment of CTL* in which path formulas are "restricted" to

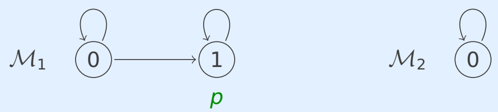
$$\alpha \ ::= \ \varphi \mid (\neg\alpha) \mid (\alpha \wedge \alpha) \mid (\alpha \vee \alpha) \mid (\alpha \to \alpha) \mid (\mathsf{X}\,\varphi) \mid (\mathsf{F}\,\varphi) \mid (\mathsf{G}\,\varphi) \mid (\varphi\,\mathsf{U}\,\varphi)$$

Lemma

AG EF p is not expressible in LTL

Proof

- suppose AG EF $p \equiv \mathsf{A}[\varphi]$ for LTL formula φ
- consider models

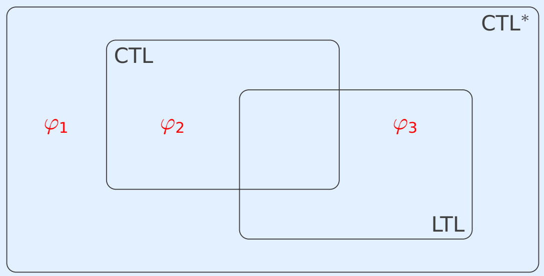


- $\mathcal{M}_1, 0 \vDash$ AG EF p
- $\mathcal{M}_1, 0 \vDash \mathsf{A}[\varphi]$
- $\mathcal{M}_2, 0 \nvDash$ AG EF p
- $\mathcal{M}_2, 0 \vDash \mathsf{A}[\varphi]$ because every path from 0 in $\mathcal{M}_2$ is also path in $\mathcal{M}_1$ ↯

Lemma

- $\mathsf{A}[\mathsf{G}\,\mathsf{F}\,p \to \mathsf{F}\,q]$ is not expressible in CTL
- $\mathsf{E}[\mathsf{G}\,\mathsf{F}\,p]$ is expressible neither in CTL nor LTL

Expressive Power



$\varphi_1 = \mathsf{E}[\mathsf{G}\,\mathsf{F}\,p]$
$\varphi_2 = $ AG EF p
$\varphi_3 = \mathsf{A}[\mathsf{G}\,\mathsf{F}\,p \to \mathsf{F}\,q]$

Outline

Particify with session ID **0992 9580**

Question

Which of the following statements are true ?

A A set of LTL connectives which contains G cannot be adequate.

B The CTL formulas $\mathsf{AG}\,\neg p \rightarrow \mathsf{EF}\,q$ and $\mathsf{EF}(p \vee q)$ are equivalent.

C The CTL formula $p \wedge \mathsf{AX}\,\mathsf{AG}\,p$ is equivalent to the LTL formula $\mathsf{G}\,p$.

D The CTL* formulas $\mathsf{E}[\mathsf{G}\,\mathsf{E}[\mathsf{F}\,p]]$ and $\mathsf{E}[\mathsf{G}\,\mathsf{F}\,p]$ are equivalent.

Outline

Remarks

- most state-of-the-art SAT solvers are based on variations of Davis – Putnam – Logemann – Loveland (DPLL) procedure (1960, 1962)
- abstract version of DPLL described in JACM paper of Nieuwenhuis, Oliveras, Tinelli (2006)

Definition (Abstract DPLL)

- states $M \parallel F$ consist of
  - list M of (possibly annotated) non-complementary literals
  - CNF F
- transition rules

$$M \parallel F \quad \Longrightarrow \quad M' \parallel F' \quad \text{or} \quad \text{fail-state} \qquad (\text{this lecture: } F = F')$$

Example

$\varphi = (\neg 1 \vee \neg 2) \wedge (2 \vee 3) \wedge (\neg 1 \vee \neg 3 \vee 4) \wedge (2 \vee \neg 3 \vee \neg 4) \wedge (1 \vee 4)$

		$\parallel$	$\neg 1 \vee \neg 2,\ 2 \vee 3,\ \neg 1 \vee \neg 3 \vee 4,\ 2 \vee \neg 3 \vee \neg 4,\ 1 \vee 4$	
$\Longrightarrow$	$\overset{d}{1}$	$\parallel$	$\neg 1 \vee \neg 2,\ 2 \vee 3,\ \neg 1 \vee \neg 3 \vee 4,\ 2 \vee \neg 3 \vee \neg 4,\ 1 \vee 4$	decide
$\Longrightarrow$	$\overset{d}{1}\ \neg 2$	$\parallel$	$\neg 1 \vee \neg 2,\ 2 \vee 3,\ \neg 1 \vee \neg 3 \vee 4,\ 2 \vee \neg 3 \vee \neg 4,\ 1 \vee 4$	unit propagate
$\Longrightarrow$	$\overset{d}{1}\ \neg 2\ 3$	$\parallel$	$\neg 1 \vee \neg 2,\ 2 \vee 3,\ \neg 1 \vee \neg 3 \vee 4,\ 2 \vee \neg 3 \vee \neg 4,\ 1 \vee 4$	unit propagate
$\Longrightarrow$	$\overset{d}{1}\ \neg 2\ 3\ 4$	$\parallel$	$\neg 1 \vee \neg 2,\ 2 \vee 3,\ \neg 1 \vee \neg 3 \vee 4,\ 2 \vee \neg 3 \vee \neg 4,\ 1 \vee 4$	unit propagate
$\Longrightarrow$	$\neg 1$	$\parallel$	$\neg 1 \vee \neg 2,\ 2 \vee 3,\ \neg 1 \vee \neg 3 \vee 4,\ 2 \vee \neg 3 \vee \neg 4,\ 1 \vee 4$	backtrack
$\Longrightarrow$	$\neg 1\ 4$	$\parallel$	$\neg 1 \vee \neg 2,\ 2 \vee 3,\ \neg 1 \vee \neg 3 \vee 4,\ 2 \vee \neg 3 \vee \neg 4,\ 1 \vee 4$	unit propagate
$\Longrightarrow$	$\neg 1\ 4\ \overset{d}{\neg 3}$	$\parallel$	$\neg 1 \vee \neg 2,\ 2 \vee 3,\ \neg 1 \vee \neg 3 \vee 4,\ 2 \vee \neg 3 \vee \neg 4,\ 1 \vee 4$	decide
$\Longrightarrow$	$\neg 1\ 4\ \overset{d}{\neg 3}\ 2$	$\parallel$	$\neg 1 \vee \neg 2,\ 2 \vee 3,\ \neg 1 \vee \neg 3 \vee 4,\ 2 \vee \neg 3 \vee \neg 4,\ 1 \vee 4$	unit propagate

Definition (Transition Rules)

- unit propagate $\qquad M \parallel F, C \vee \ell \quad \Longrightarrow \quad M\,\ell \parallel F, C \vee \ell$

  if $M \vDash \neg C$ and ℓ is undefined in M $\qquad$ unit clause

- pure literal $\qquad M \parallel F \quad \Longrightarrow \quad M\,\ell \parallel F$

  if ℓ occurs in F and ℓ^c does not occur in F and ℓ is undefined in M

- decide $\qquad M \parallel F \quad \Longrightarrow \quad M\,\overset{d}{\ell} \parallel F$

  if ℓ or ℓ^c occurs in F and ℓ is undefined in M

- fail $\qquad M \parallel F, C \quad \Longrightarrow \quad$ fail-state

  if $M \vDash \neg C$ and M contains no decision literals

- backtrack $\qquad M\,\overset{d}{\ell}\,N \parallel F, C \quad \Longrightarrow \quad M\,\ell^c \parallel F, C$

  if $M\,\overset{d}{\ell}\,N \vDash \neg C$ and N contains no decision literals

Outline

Example

$\varphi = (\neg 1 \vee 2) \wedge (\neg 3 \vee 4) \wedge (\neg 5 \vee \neg 6) \wedge (6 \vee \neg 5 \vee \neg 2)$

		$\parallel$	$\neg 1 \vee 2,\ \neg 3 \vee 4,\ \neg 5 \vee \neg 6,\ 6 \vee \neg 5 \vee \neg 2$	
$\Longrightarrow$	$\overset{d}{1}$	$\parallel$	$\neg 1 \vee 2,\ \neg 3 \vee 4,\ \neg 5 \vee \neg 6,\ 6 \vee \neg 5 \vee \neg 2$	decide
$\Longrightarrow$	$\overset{d}{1}\ 2$	$\parallel$	$\neg 1 \vee 2,\ \neg 3 \vee 4,\ \neg 5 \vee \neg 6,\ 6 \vee \neg 5 \vee \neg 2$	unit propagate
$\Longrightarrow$	$\overset{d}{1}\ 2\ \overset{d}{3}$	$\parallel$	$\neg 1 \vee 2,\ \neg 3 \vee 4,\ \neg 5 \vee \neg 6,\ 6 \vee \neg 5 \vee \neg 2$	decide
$\Longrightarrow$	$\overset{d}{1}\ 2\ \overset{d}{3}\ 4$	$\parallel$	$\neg 1 \vee 2,\ \neg 3 \vee 4,\ \neg 5 \vee \neg 6,\ 6 \vee \neg 5 \vee \neg 2$	unit propagate
$\Longrightarrow$	$\overset{d}{1}\ 2\ \overset{d}{3}\ 4\ \overset{d}{5}$	$\parallel$	$\neg 1 \vee 2,\ \neg 3 \vee 4,\ \neg 5 \vee \neg 6,\ 6 \vee \neg 5 \vee \neg 2$	decide
$\Longrightarrow$	$\overset{d}{1}\ 2\ \overset{d}{3}\ 4\ \overset{d}{5}\ \neg 6$	$\parallel$	$\neg 1 \vee 2,\ \neg 3 \vee 4,\ \neg 5 \vee \neg 6,\ 6 \vee \neg 5 \vee \neg 2$	unit propagate
$\Longrightarrow$	$\overset{d}{1}\ 2\ \neg 5$	$\parallel$	$\neg 1 \vee 2,\ \neg 3 \vee 4,\ \neg 5 \vee \neg 6,\ 6 \vee \neg 5 \vee \neg 2$	backjump

conflict is due to $\overset{d}{1}\ 2$ and $\overset{d}{5}\ \neg 6$ hence $\neg 1 \vee \neg 5$ can be inferred

Definitions

- backtract $\qquad M\,\overset{d}{\ell}\,N \parallel F, C \quad\Longrightarrow\quad M\,\ell^c \parallel F, C$

  if $M\,\overset{d}{\ell}\,N \vDash \neg C$ and N contains no decision literals

- backjump $\qquad M\,\overset{d}{\ell}\,N \parallel F, C \quad\Longrightarrow\quad M\,\ell' \parallel F, C$

  if $M\,\overset{d}{\ell}\,N \vDash \neg C$ and there exists clause $C' \vee \ell'$ such that
  - $F, C \vDash C' \vee \ell'$ $\qquad$ backjump clause
  - $M \vDash \neg C'$
  - ℓ' is undefined in M
  - ℓ' or ℓ'^c occurs in F or in $M\,\overset{d}{\ell}\,N$

Example (cont'd)

$\neg 1 \vee \neg 5$ and $\neg 2 \vee \neg 5$ are backjump clauses with respect to $\overset{d}{1}\;2\;\overset{d}{3}\;4\;\overset{d}{5}\;\neg 6 \parallel \varphi$

Definition

basic DPLL $\mathcal{B}$ consists of transition rules

- unit propagate $\qquad M \parallel F, C \vee \ell \quad\Longrightarrow\quad M\,\ell \parallel F, C \vee \ell$

  if $M \vDash \neg C$ and ℓ is undefined in M

- decide $\qquad M \parallel F \quad\Longrightarrow\quad M\,\overset{d}{\ell} \parallel F$

  if ℓ or ℓ^c occurs in F and ℓ is undefined in M

- fail $\qquad M \parallel F, C \quad\Longrightarrow\quad$ fail-state

  if $M \vDash \neg C$ and M contains no decision literals

- backjump $\qquad M\,\overset{d}{\ell}\,N \parallel F, C \quad\Longrightarrow\quad M\,\ell' \parallel F, C$

  if $M\,\overset{d}{\ell}\,N \vDash \neg C$ and there exists clause $C' \vee \ell'$ such that
  - $F, C \vDash C' \vee \ell'$ and $M \vDash \neg C'$
  - ℓ' is undefined in M and ℓ' or ℓ'^c occurs in F or in $M\,\overset{d}{\ell}\,N$

Theorem

there are no infinite derivations $\parallel F \Longrightarrow_{\mathcal{B}} S_1 \Longrightarrow_{\mathcal{B}} S_2 \Longrightarrow_{\mathcal{B}} \cdots$

Proof

- for list of distinct literals M, $|M|$ is length of M
- measure state $M_0\,\overset{d}{\ell_1}\,M_1\,\overset{d}{\ell_2}\,M_2 \ldots \overset{d}{\ell_k}\,M_k \parallel F$ where $M_0, \ldots, M_k$ contain no decision literals by tuple $(|M_0|, |M_1|, \ldots, |M_k|)$
- compare tuples lexicographically using standard order on $\mathbb{N}$
- every transition step strictly increases measure
- measure is bounded by $(n+1)$-tuple $(n, \ldots, n)$ where n is total number of atoms

Example

$\parallel \varphi = (\neg 1 \vee 2) \wedge (\neg 3 \vee 4) \wedge (\neg 5 \vee \neg 6) \wedge (6 \vee \neg 5 \vee \neg 2)$			(0)
$\Longrightarrow$	$\overset{d}{1} \parallel \varphi$	decide	$(0,0)$
$\Longrightarrow$	$\overset{d}{1}\;2 \parallel \varphi$	unit propagate	$(0,1)$
$\Longrightarrow$	$\overset{d}{1}\;2\;\overset{d}{3} \parallel \varphi$	decide	$(0,1,0)$
$\Longrightarrow$	$\overset{d}{1}\;2\;\overset{d}{3}\;4 \parallel \varphi$	unit propagate	$(0,1,1)$
$\Longrightarrow$	$\overset{d}{1}\;2\;\overset{d}{3}\;4\;\overset{d}{5} \parallel \varphi$	decide	$(0,1,1,0)$
$\Longrightarrow$	$\overset{d}{1}\;2\;\overset{d}{3}\;4\;\overset{d}{5}\;\neg 6 \parallel \varphi$	unit propagate	$(0,1,1,1)$
$\Longrightarrow$	$\overset{d}{1}\;2\;\neg 5 \parallel \varphi$	backjump	$(0,2)$

- decide $\qquad (m_0, \ldots, m_i) <_{\text{lex}} (m_0, \ldots, m_i, 0)$
- unit propagate $\qquad (m_0, \ldots, m_i) <_{\text{lex}} (m_0, \ldots, m_i + 1)$
- backjump $\qquad (m_0, \ldots, m_i) <_{\text{lex}} (m_0, \ldots, m_j + 1)$ with $j < i$

Lemma

1. if $\parallel F \Longrightarrow_{\mathcal{B}}^{*} M \parallel F'$ then
   - $F = F'$
   - M does not contain complementary literals
   - M consists of distinct literals
2. if $\parallel F \Longrightarrow_{\mathcal{B}}^{*} M_0 \, \overset{d}{\ell_1} \, M_1 \, \overset{d}{\ell_2} \, M_2 \cdots \overset{d}{\ell_k} \, M_k \parallel F$ with no decision literals in $M_0, \dots, M_k$ then $F, \ell_1, \dots, \ell_i \vDash M_i$ for all $0 \leqslant i \leqslant k$

Theorem

if $\parallel F \Longrightarrow_{\mathcal{B}} S_1 \Longrightarrow_{\mathcal{B}} \cdots \Longrightarrow_{\mathcal{B}} S_n \not\Longrightarrow_{\mathcal{B}}$ then

1. $S_n =$ fail-state if and only if F is unsatisfiable
2. $S_n = M \parallel F'$ only if F is satisfiable and $M \vDash F$

Proof

① (only if) $\parallel F \Longrightarrow_{\mathcal{B}}^{*} M \parallel F \Longrightarrow_{\text{fail}}$ fail-state

- M contains no decision literals and $M \vDash \neg C$ for some C in F
- $F \vDash C$ and $F \vDash M$ and thus $F \vDash \neg C$ and thus F is unsatisfiable

② $\parallel F \Longrightarrow_{\mathcal{B}}^{*} M \parallel F' \not\Longrightarrow_{\mathcal{B}}$

- $F = F'$ and all literals in F are defined in M, otherwise decide is applicable
- F contains no clause such that $M \vDash \neg C$, otherwise backjump or fail is applicable
- $M \vDash F$ and thus F is satisfiable

Lemma

backjump can simulate backtrack

Proof

- suppose $\parallel F \Longrightarrow_{\mathcal{B}}^{*} M \, \overset{d}{\ell} \, N \parallel F \Longrightarrow_{\text{backtrack}} M \, \ell^c \parallel F$
- $M \, \overset{d}{\ell} \, N \vDash \neg C$ for some C in F and N contains no decision literals
- write $M = M_0 \, \overset{d}{\ell_1} \, M_1 \, \overset{d}{\ell_2} \, M_2 \cdots \overset{d}{\ell_k} \, M_k$ with all decision literals displayed
- $\ell_1^c \vee \cdots \vee \ell_k^c \vee \ell^c$ is backjump clause:
  - $F, \ell_1, \dots, \ell_k, \ell \vDash \neg C \implies F, \ell_1, \dots, \ell_k, \ell$ is unsatisfiable $\implies F \vDash \ell_1^c \vee \cdots \vee \ell_k^c \vee \ell^c$
  - $M \vDash \ell_1 \wedge \cdots \wedge \ell_k$ and ℓ^c is undefined in M
- $M \, \overset{d}{\ell} \, N \parallel F \Longrightarrow_{\text{backjump}} M \, \ell^c \parallel F$

Terminology

non-chronological backtracking or conflict-driven backtracking

Question

how to find good backjump clauses ?

Answer

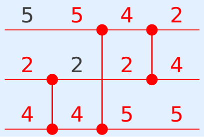
use conflict graph (lecture 13)

Outline

1. Summary of Previous Lecture
2. CTL*
3. Intermezzo
4. SAT Solving
5. **Sorting Networks**
6. Further Reading

Sorting Network

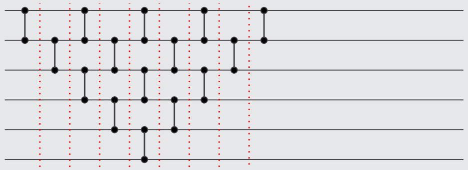
input



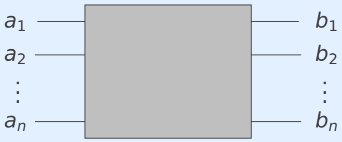
output

$4 > 2$ $4 \not> 5$ $2 \not> 4$

Example



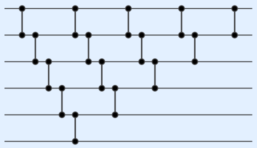
- size ($=$ number of comparators): 15
- depth: 9



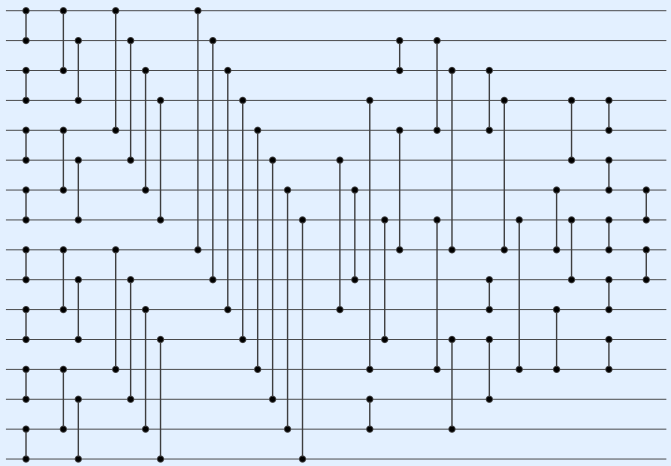
Definition

sorting network is comparator network that transforms any input sequence $a = (a_1, \dots, a_n)$ of natural numbers into sorted output sequence $b = (b_1, \dots, b_n)$:

b is permutation of a and $b_1 \leqslant \cdots \leqslant b_n$



Sorting Network ?



Questions

① how to check that comparator network is sorting network ?

② how to find optimal (with respect to size or depth) sorting networks ?

Answers

① testing all $n!$ permutations of $1, \ldots, n$ for network with n wires suffices

② very difficult problem . . .

Outline

Huth and Ryan

- Section 3.5

DPLL

- Section 2 of Solving SAT and SAT Modulo Theories: From an Abstract Davis–Putnam–Logemann–Loveland Procedure to DPLL(T)
  Robert Nieuwenhuis, Albert Oliveras, and Cesare Tinelli
  Journal of the ACM 53(6), pp. 937–977, 2006
  doi: 10.1145/1217856.1217859

Sorting Networks

- Wikipedia [accessed December 14, 2022]
- Section 5.3.4 of The Art of Computer Programming
  Donald Knuth

Important Concepts

- abstract DPLL
- basic DPLL
- backjump
- backtrack
- comparator network
- CTL*
- decide
- depth
- fail-state
- path formula
- pure literal
- size
- sorting network
- state formula
- unit propagation

homework for June 13

evaluation SS 2024