Fundamentals of Programming Languages IV

LTL Model Checking

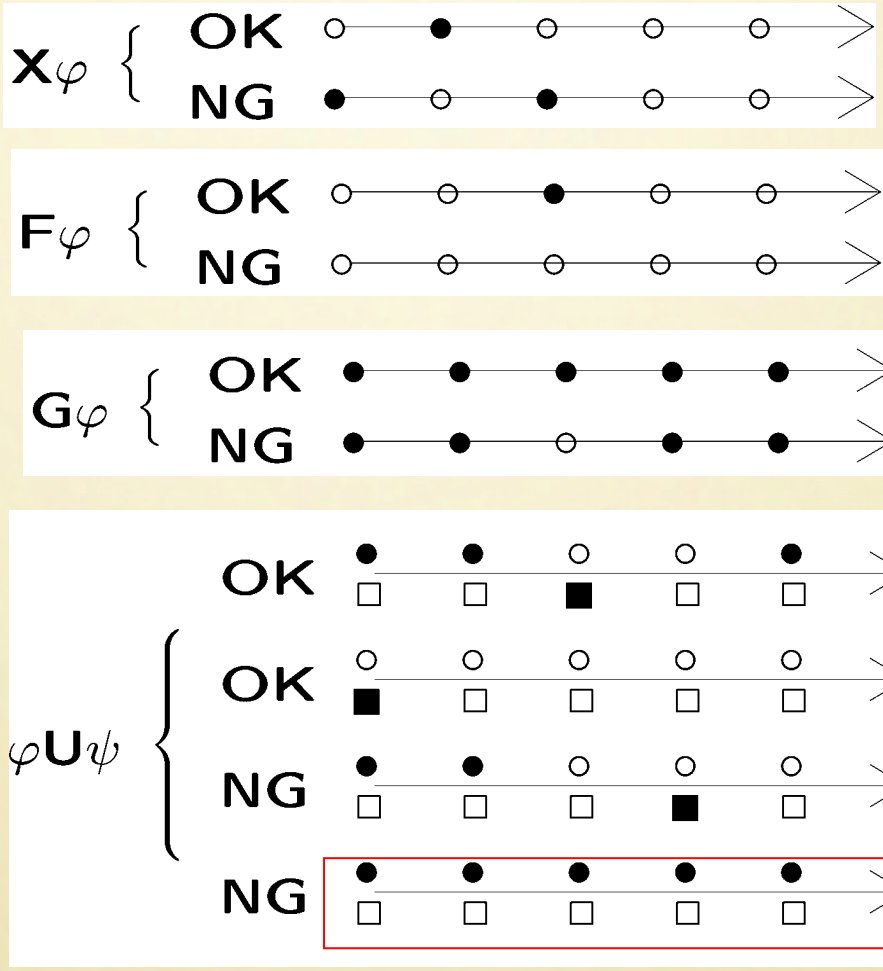
Guoqiang Li

School of Software, Shanghai Jiao Tong University

Reviews

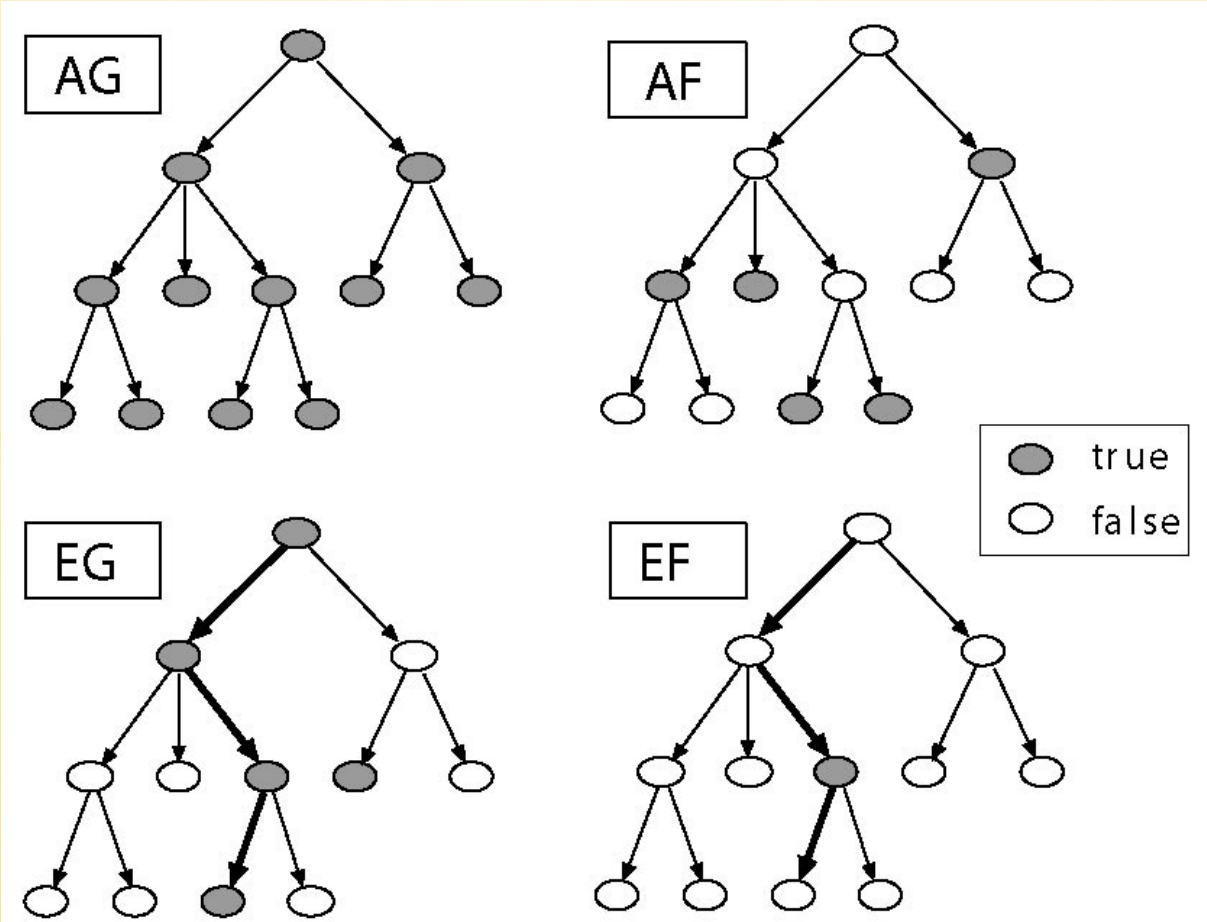
Temporal Operators

- Next
- Finally
- Globally
- Until



● : φ, ■ : ψ, ○ : $\neg\varphi$, □ : $\neg\psi$

Path Operators, *A*, *E*



- *AG*: safety, bad things will never happen.
- *AF*: liveness, good things will eventually happen.

LTL Model Checking

Complicity of LTL Model Checking

Tableau method: $O((|S| + |R|) \times 2^{O(|\varphi|)})$

Complicity of LTL Model Checking

Tableau method: $O((|S| + |R|) \times 2^{O(|\varphi|)})$

At least NP-hard: consider Hamilton path of G

- $M, q_0 \models E(F\, p_1 \wedge \ldots \wedge F\, p_n \wedge G(P_1 \rightarrow X\, G\, \neg p_1) \wedge \ldots \wedge G(P_n \rightarrow X\, G\, \neg p_n))$
- $M = (\{q_0, q_f\} \cup V(G), \{q_0\}, \{(q_0, v), (v, q_f), (q_f, q_f) \mid v \in V(G)\} \cup E(G), L)$
- $L(v_i) = \{p_i\}$

Scenario of LTL Model Checking

- $A\,\varphi$ is a LTL, then the only state sub-formulas in φ are atomic propositions.
- $M, s \models A\,\varphi \Longleftrightarrow M, s \models \neg E\,\neg\varphi$
- $M, s \models F\,\varphi \Longleftrightarrow M, s \models true\,U\,\varphi$
- $M, s \models G\,\varphi \Longleftrightarrow M, s \models \neg F\,\neg\varphi$
- It is sufficient to only consider the temporal operators X, U with $\neg, \vee$ wrapped by E.

Scenario of LTL Model Checking

- $A\,\varphi$ is a LTL, then the only state sub-formulas in φ are atomic propositions.
- $M, s \models A\,\varphi \iff M, s \models \neg E\,\neg\varphi$
- $M, s \models F\,\varphi \iff M, s \models true\ U\ \varphi$
- $M, s \models G\,\varphi \iff M, s \models \neg F\,\neg\varphi$
- It is sufficient to only consider the temporal operators X, U with $\neg, \vee$ wrapped by E.
- Construct closure $cl(\varphi)$ of φ, which is the set of formulae related to the truth value of φ.
- Construct graph of atoms (transition graph on truth table of $cl(\varphi)$)
- $M, s \models E\,\varphi$ is equivalent to existence of an eventuality sequence, which is detected as a SCC.

Closure of LTL formula

- The smallest set of formulae containing φ, where
  - $\neg\phi \in cl(\varphi)$ iff $\phi \in cl(\varphi)$;
  - if $\psi \vee \phi \in cl(\varphi)$, then $\psi, \phi \in cl(\varphi)$;
  - if $X\,\psi \in cl(\varphi)$, then $\psi \in cl(\varphi)$;
  - if $\neg X\,\psi \in cl(\varphi)$, then $X\,\neg\psi \in cl(\varphi)$;
  - if $\psi\,U\,\phi \in cl(\varphi)$, then $\psi, \phi, X(\psi\,U\,\phi) \in cl(\varphi)$.
- To keep finite (linear to $|\varphi|$), $\neg\neg$ is eliminated.
- By construction, at most one X would be added.
- Size of $cl(f)$ is linear in the size of f.
- e.g. $cl(\delta\,U\,\psi) =$

Closure of LTL formula

- The smallest set of formulae containing φ, where
  - $\neg\phi \in cl(\varphi)$ iff $\phi \in cl(\varphi)$;
  - if $\psi \vee \phi \in cl(\varphi)$, then $\psi, \phi \in cl(\varphi)$;
  - if $X\,\psi \in cl(\varphi)$, then $\psi \in cl(\varphi)$;
  - if $\neg X\,\psi \in cl(\varphi)$, then $X\,\neg\psi \in cl(\varphi)$;
  - if $\psi\,U\,\phi \in cl(\varphi)$, then $\psi, \phi, X(\psi\,U\,\phi) \in cl(\varphi)$.
- To keep finite (linear to $|\varphi|$), $\neg\neg$ is eliminated.
- By construction, at most one X would be added.
- Size of $cl(f)$ is linear in the size of f.
- e.g. $cl(\delta\,U\,\psi) =$
  - $\{\delta, \neg\delta, \psi, \neg\psi,$
  - $\delta\,U\,\psi, \neg(\delta\,U\,\psi), X(\delta\,U\,\psi), \neg X(\delta\,U\,\psi),$
  - $X\neg(\delta\,U\,\psi)\}$

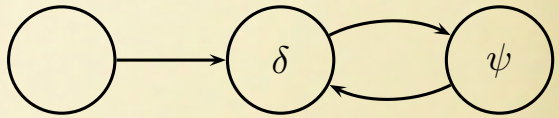
Atoms (wrt. φ)

- Atom (s, K) with $s \in S$ and $K \subseteq cl(\varphi) \cup AP$, where
  - for each $p \in AP$, $p \in K$, iff $p \in L(s)$;
  - for every $\delta \in cl(\varphi)$, $\delta \in K$, iff $\neg\delta \notin K$;
  - for every $\delta \vee \psi \in cl(\varphi)$, $\delta \vee \psi \in K$ iff $\delta \in K$ or $\psi \in K$;
  - for every $\neg X\,\delta \in cl(\varphi)$, $\neg X\,\delta \in K$ iff $X\,(\neg\delta) \in K$;
  - for every $\delta\,U\,\psi \in cl(\varphi)$, $\delta\,U\,\psi \in K$ iff $\psi \in K$ or $\delta, X(\delta\,U\,\psi) \in K$.
- Intuitively, K is the maximum consistent truth valuation at s.

Atoms (wrt. φ)

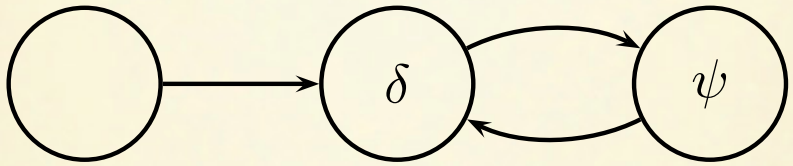
- Atom (s, K) with $s \in S$ and $K \subseteq cl(\varphi) \cup AP$, where
  - for each $p \in AP$, $p \in K$, iff $p \in L(s)$;
  - for every $\delta \in cl(\varphi)$, $\delta \in K$, iff $\neg\delta \notin K$;
  - for every $\delta \vee \psi \in cl(\varphi)$, $\delta \vee \psi \in K$ iff $\delta \in K$ or $\psi \in K$;
  - for every $\neg X\,\delta \in cl(\varphi)$, $\neg X\,\delta \in K$ iff $X\,(\neg\delta) \in K$;
  - for every $\delta\,U\,\psi \in cl(\varphi)$, $\delta\,U\,\psi \in K$ iff $\psi \in K$ or $\delta, X(\delta\,U\,\psi) \in K$.
- Intuitively, K is the maximum consistent truth valuation at s.

e.g.



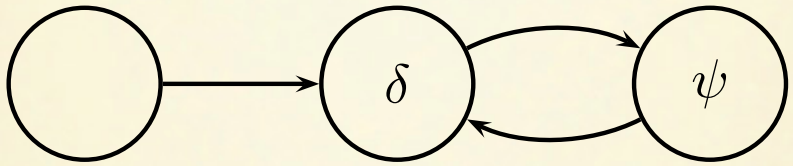
$cl(\delta\,U\,\psi) =$
$\{\delta, \psi, \delta\,U\,\psi, X(\delta\,U\,\psi),$
$\neg\delta, \neg\psi, \neg(\delta\,U\,\psi),$
$\neg X(\delta\,U\,\psi), X\neg(\delta\,U\,\psi)\}$

Example of Atoms



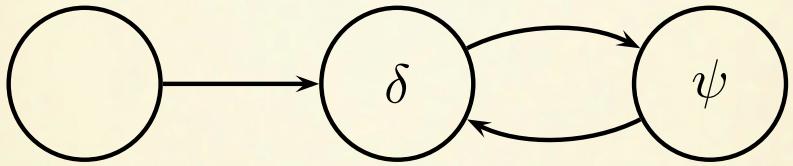
- Tableau:
  - [T,T,T,T], [T,T,T,F], [T,T,F,T], [T,T,F,F]
  - [T,F,T,T], [T,F,T,F], [T,F,F,T], [T,F,F,F]
  - [F,T,T,T], [F,T,T,F], [F,T,F,T], [F,T,F,F]
  - [F,F,T,T], [F,F,T,F], [F,F,F,T], [F,F,F,F]

Example of Atoms



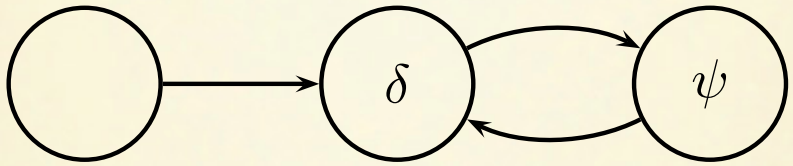
- Tableau:
  - [T,T,T,T], [T,T,T,F], [T,T,F,T], [T,T,F,F]
  - [T,F,T,T], [T,F,T,F], [T,F,F,T], [T,F,F,F]
  - [F,T,T,T], [F,T,T,F], [F,T,F,T], [F,T,F,F]
  - [F,F,T,T], [F,F,T,F], [F,F,F,T], [F,F,F,F]
- $s_0 : (L(s_0) = \neg\delta \wedge \neg\psi)$
  - [F,F,T,T], [F,F,T,F], [F,F,F,T], [F,F,F,F]

Example of Atoms



- Tableau:
  - [T,T,T,T], [T,T,T,F], [T,T,F,T], [T,T,F,F]
  - [T,F,T,T], [T,F,T,F], [T,F,F,T], [T,F,F,F]
  - [F,T,T,T], [F,T,T,F], [F,T,F,T], [F,T,F,F]
  - [F,F,T,T], [F,F,T,F], [F,F,F,T], [F,F,F,F]
- $s_0 : (L(s_0) = \neg\delta \wedge \neg\psi)$
  - [F,F,T,T], [F,F,T,F], [F,F,F,T], [F,F,F,F]
- $s_1 : (L(s_1) = \delta \wedge \neg\psi)$
  - [T,F,T,T], [T,F,T,F], [T,F,F,T], [T,F,F,F]

Example of Atoms

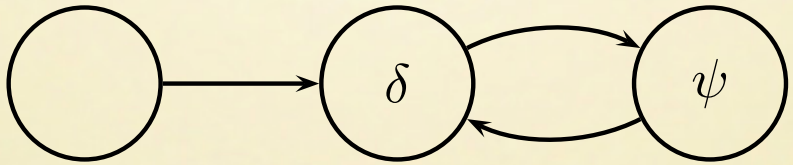


- Tableau:
  - [T,T,T,T], [T,T,T,F], [T,T,F,T], [T,T,F,F]
  - [T,F,T,T], [T,F,T,F], [T,F,F,T], [T,F,F,F]
  - [F,T,T,T], [F,T,T,F], [F,T,F,T], [F,T,F,F]
  - [F,F,T,T], [F,F,T,F], [F,F,F,T], [F,F,F,F]
- $s_0 : (L(s_0) = \neg\delta \wedge \neg\psi)$
  - [F,F,T,T], [F,F,T,F], [F,F,F,T], [F,F,F,F]
- $s_1 : (L(s_1) = \delta \wedge \neg\psi)$
  - [T,F,T,T], [T,F,T,F], [T,F,F,T], [T,F,F,F]
- $s_2 : (L(s_2) = \neg\delta \wedge \psi)$
  - [F,T,T,T], [F,T,T,F], [F,T,F,T], [F,T,F,F]

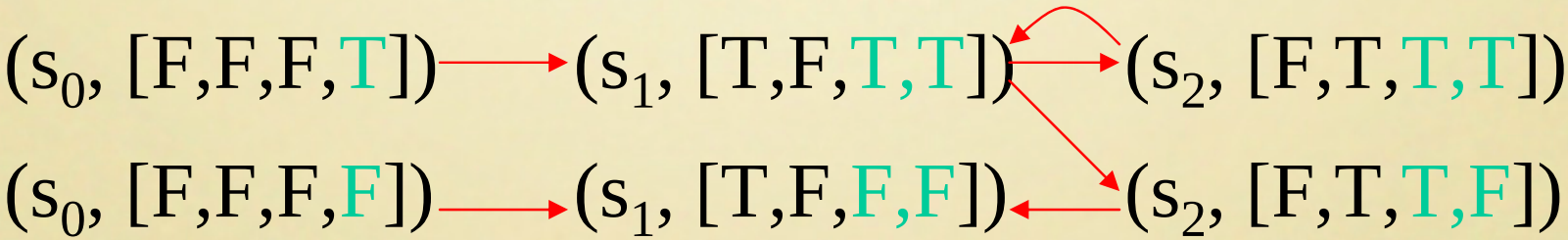
Graph of Atoms

- For Kripke structure $M = (S, S_0, R, L)$, formula φ, define a graph of atoms where nodes are atoms and edged are:

$$\{((s,K),(s',K')) \mid (s,s') \in R \wedge \forall\, (X\,\delta) \in cl(\varphi), X\,\delta \in K \iff \delta \in K'\}$$

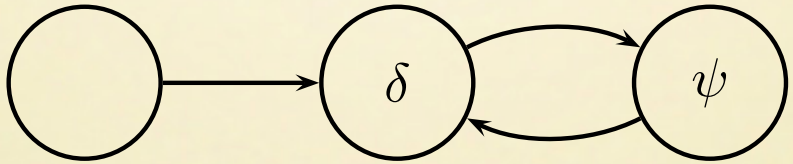


$$\delta, \psi, \delta\ U\ \psi, X(\delta\ U\ \psi)$$

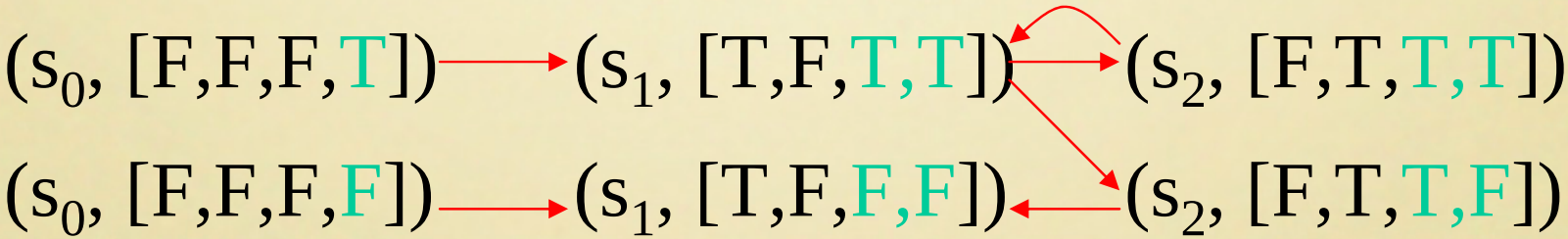


Eventuality Sequence

- An eventuality sequence is an infinite path π in a graph of atoms, satisfying:
  - If $\delta \, U \, \psi \in K$ for an atom (s, K) on π, then there exists an atom (s', K') on π after (s, K) with $\psi \in K'$.

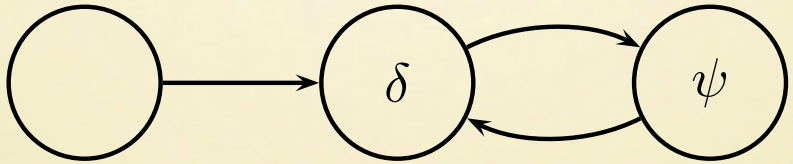


$\delta, \psi, \delta \, U \, \psi, X(\delta \, U \, \psi)$

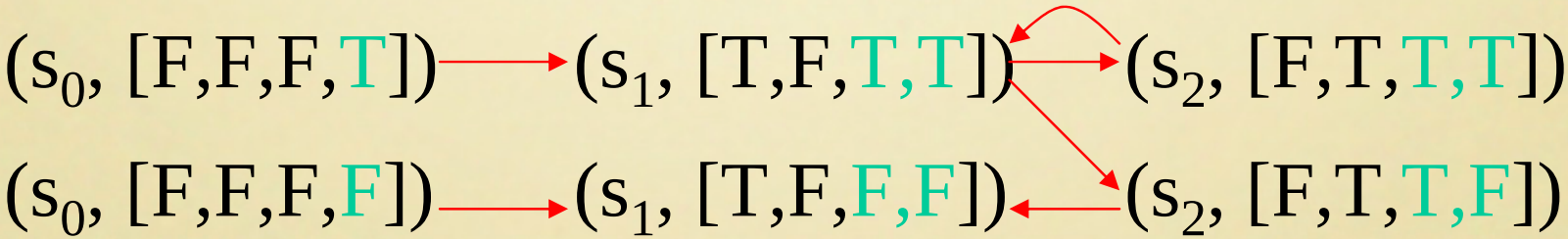


Eventuality Sequence

- An eventuality sequence is an infinite path π in a graph of atoms, satisfying:
  - If $\delta \, U \, \psi \in K$ for an atom (s, K) on π, then there exists an atom (s', K') on π after (s, K) with $\psi \in K'$.
  - Don't care on δ between (s, K) and (s', K'), why?



$\delta, \psi, \delta \, U \, \psi, X(\delta \, U \, \psi)$



Key Lemma

Lemma:
$M, s \models E\,\varphi$ iff there exists an eventuality sequence starting from an atom (s, K) with $\varphi \in K$.

Proof Sketch ($\Longrightarrow$)

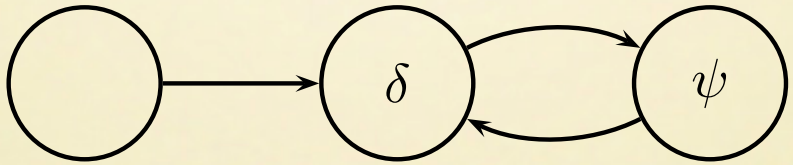
- $M, s_0 \models E\,\varphi$, if there exists an eventuality sequence $\pi = (s_0, K_0), (s_1, K_1), (s_2, K_2) \ldots$ with $\varphi \in K_0$.
- let $\pi^i = (s_i, K_i), (s_{i+1}, K_{i+1}), (s_{i+2}, K_{i+2}) \ldots$, we will prove "$\pi^i \models \delta \Longleftrightarrow \delta \in K_i$, *for each* $\delta \in cl(\varphi)$" by induction on the structure of formula.
  - Case $\delta = X\,\gamma$: By construction of a graph of atoms, $((s_i, K_i), (s_{i+1}, K_{i+1}))$ implies $X\,\gamma \in K_i \Longleftrightarrow \gamma \in K_{i+1}$. Thus, $X\,\gamma \in K_i \Longleftrightarrow \gamma \in K_{i+1} \Longleftrightarrow \pi^{i+1} \models \gamma \Longleftrightarrow \pi^i \models X\,\gamma$.
  - Case $\delta = \gamma\,U\,\psi$:
    - By definition of π, there exists (first) $j \geq i$ with $\psi \in K_j$.
    - Then $\delta \in K_j$ (by definition of atom), and $\pi^j \models \psi$ (by induction hypothesis); thus $\pi^j \models \delta$.
    - Note that $\psi \notin K_i \wedge \ldots \wedge \psi \notin K_{j-1}$; then $\gamma, X\,\delta \in K_i$ $\Longleftrightarrow \gamma \in K_i \wedge \delta \in K_{i+1} \Longleftrightarrow \gamma \in K_i \wedge \ldots \wedge \gamma \in K_{j-i} \Longleftrightarrow \pi^i \models \gamma \wedge \ldots \pi^{j-1} \models \gamma \Longleftrightarrow \pi^i \models \delta$.

Proof Sketch ($\Longleftarrow$)

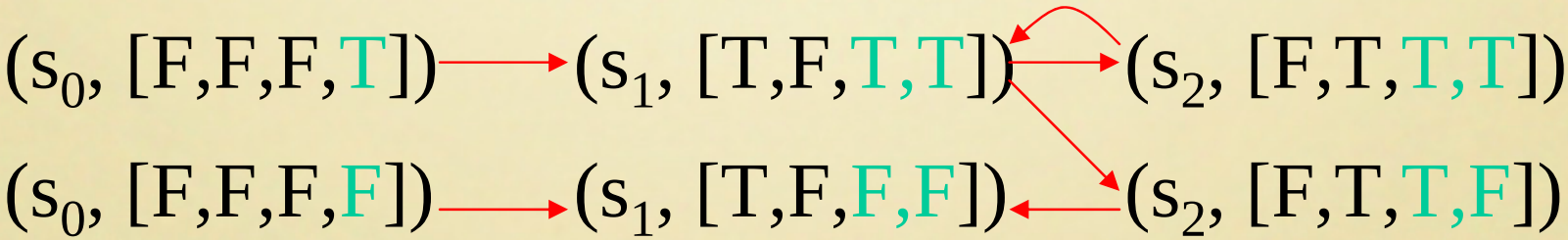
- $M, s_0 \models E\,\varphi$ only if there exists an eventuality sequence starting from an atom (s, K) with $\varphi \in K$.
- Let $\pi = s_0, s_1, s_2, \ldots$, s.t. $M, \pi \models \varphi$. Then, $(s_0, K_0), (s_1, K_1), (s_2, K_2), \ldots$ is an eventuality sequence where $K_i = \{\delta \mid \delta \in cl(\varphi) \wedge M, \pi^i \models \delta\}$ for $\pi^i = s_i, s_{i+1}, s_{i+2}, \ldots$.

Self-fulfilling SCC in Graph of Atoms

A non-trivial SCC C in a graph of atoms is self-fulfilling iff, for every atom (s, K) in C with $\delta\, U\, \psi \in K$, there exists an atom (s', K') in C such that $\psi \in K'$.
(i.e., there is an eventuality sequence that covers SCC C).

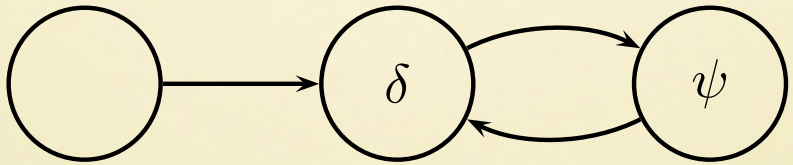


$\delta, \psi, \delta\, U\, \psi, X(\delta\, U\, \psi)$



Self-fulfilling SCC in Graph of Atoms

A non-trivial SCC C in a graph of atoms is self-fulfilling iff, for every atom (s, K) in C with $\delta\, U\, \psi \in K$, there exists an atom (s', K') in C such that $\psi \in K'$.
(i.e., there is an eventuality sequence that covers SCC C).



Lemma:
There exists an eventuality sequence starting at an atom (s, K) iff there exists a path from (s, K) to a self-fulfilling SCC.

Proof

$\Rightarrow$: Assume that there is an eventuality sequence starting at (s, K). Consider the set C' of all atoms that appear infinitely often in this sequence. The set C' is a subset of a (maximal) strongly connected component C of G. Consider a subformula $\delta U \varphi$, and an atom $(s, K) \in C$ such that $\delta U \varphi \in K$. Because C is strongly connected, there is a finite path in C from (s, K) into C'. If φ appears on the path, we are done! Otherwise, since C' comes from an eventuality sequence, and φ is in some atom of C'.
$\Leftarrow$: Trivial.

LTL Model Checking

$M, s \models E\,\phi$ iff there exists atom $A = (s, K)$ such that $\phi \in K$ and there exists a path from A to a self-fulfilling strongly connected component.

Summary of Algorithm

- Construct a graph of atoms for a formula φ, and compute self-fulfilling SCCs.
- Finding an eventuality sequence to self-fulfilling SCC by depth-first search.
- Atoms may multiplicand at most the exponential of the size of closure, (which is linear to $|\varphi|$).
- Complexity: $O((|S| + |R|) \times 2^{O(|\varphi|)})$

On-the-Fly Model Checking

Büchi Automata

A Büchi automaton is a tuple $A = (\Sigma, S, \delta, S_0, F)$ where

- Σ is an alphabet,
- S is a set of states,
- $\delta : S \times \Sigma \to S$ (deterministic) or $\delta : S \times \Sigma \to 2^S$ (nondeterministic) is a transition function,
- $S_0 \subseteq S$ is a set of initial states (a singleton for deterministic automata), and
- $F \subseteq S$ is a set of accepting states.

Infinite Runs

A word w is accepted by an automaton $A = (\Sigma, S, \delta, S_0, F)$ if there is a labeling

$$\rho : \mathbb{N} \to S$$

of the word by states such that

- $\rho(0) \in S_0$,
- $\forall i \geq 0, \rho(i+1) \in \delta(\rho(i), w(i))$,
- $inf(\rho) \cap F \neq \emptyset$.

Generalized Büchi Automata

The acceptance condition of a generalized Büchi automaton is a set of sets of states $\mathcal{F} \subseteq 2^S$, and the requirement is that some state of each of the sets $F_i \in \mathcal{F}$ appears infinitely often.

More formally, a generalized Büchi $A = (\Sigma, S, \delta, S_0, \mathcal{F})$ accepts a word w if there is a labeling ρ of w by states of A that satisfies the same first two conditions as given for Büchi automata, the third being replaced by:

- For each $F_i \in \mathcal{F}$, $inf(\rho) \cap F_i \neq \emptyset$.

Encoding Generalized Büchi automata

Given a generalized Büchi automaton $A = (\Sigma, S, \delta, S_0, \mathcal{F})$, where $\mathcal{F} = \{F_1, \ldots, F_k\}$, the Büchi automaton $A' = (\Sigma, S', \delta', S'_0, F')$ defined as follows accepts the same language as A.

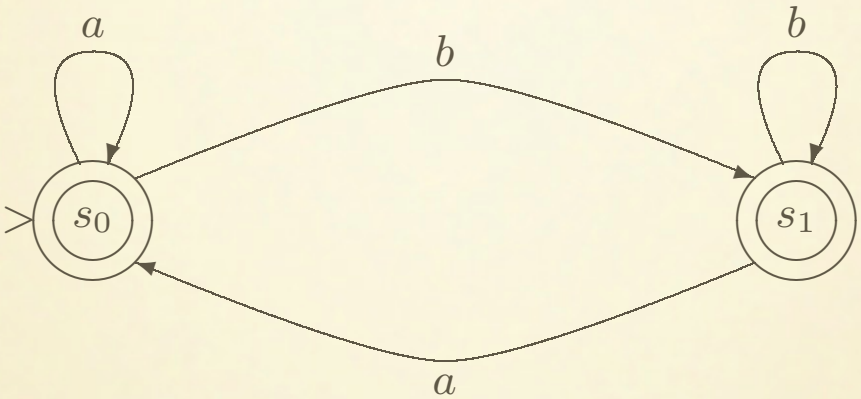
- $S' = S \times \{1, \ldots, k\}$.
- $S'_0 = S_0 \times \{1\}$.
- δ' is defined by $(t, i) \in \delta'((s, j), a)$ if

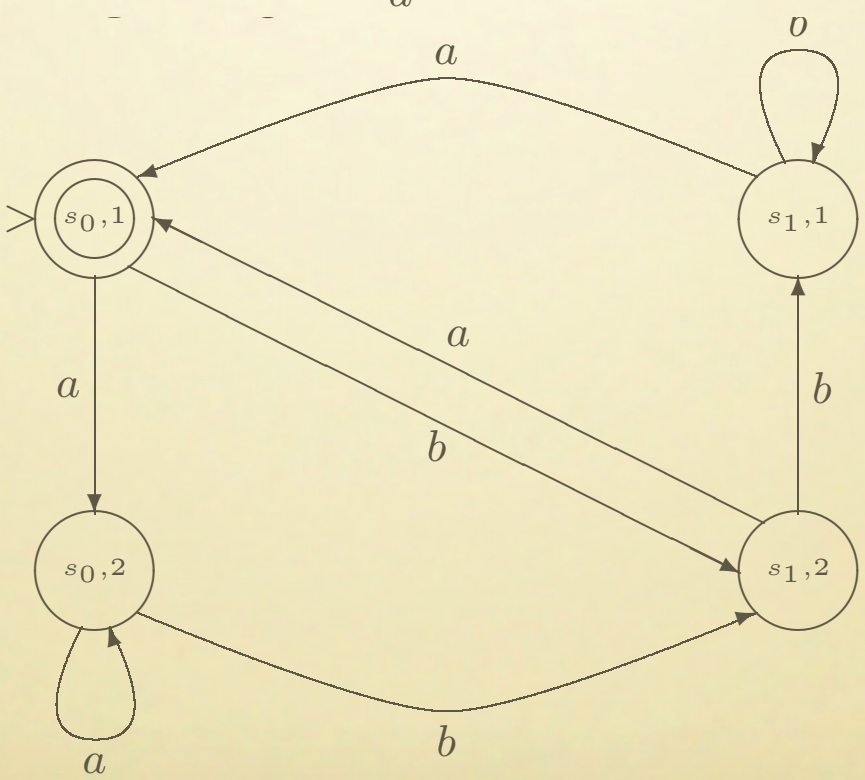
$$t \in \delta(s, a) \wedge \begin{cases} i = j & \text{if } s \notin F_j \\ i = (j \mod k) + 1 & \text{otherwise} \end{cases}$$

.

- $F' = F_1 \times \{1\}$.

An Example





From Temporal Logic to Automata

Problem Statement

Given an LTL formula φ built from a set of atomic propositions AP, construct an automaton on infinite words over the alphabet 2^{AP} that accepts exactly the infinite sequences satisfying φ.

A Dialect of LTL Logic

- true, false, p, and $\neg p$, for all $p \in AP$;
- $\varphi_1 \wedge \varphi_2$ and $\varphi_1 \vee \varphi_2$, where φ_1 and φ_2 are LTL formulas;
- $X\varphi_1$, $\varphi_1 U \varphi_2$, and $\varphi_1 R \varphi_2$, where φ_1 and φ_2 are LTL formulas.

A Dialect of LTL Logic

- $true$, $false$, p, and $\neg p$, for all $p \in AP$;
- $\varphi_1 \wedge \varphi_2$ and $\varphi_1 \vee \varphi_2$, where φ_1 and φ_2 are LTL formulas;
- $X\varphi_1$, $\varphi_1 U \varphi_2$, and $\varphi_1 R \varphi_2$, where φ_1 and φ_2 are LTL formulas.

$\varphi_1 R \varphi_2$: it requires φ_2 always be true, a requirement that is released as soon as φ_1 becomes true.

The Way to Handle Negation

$$\sigma \not\models \varphi_1 U \varphi_2 \Leftrightarrow \sigma \models (\neg\varphi_1) R (\neg\varphi_2)$$

$$\sigma \not\models X\varphi \Leftrightarrow \sigma \models X\neg\varphi$$

Closure of a Formula

$\varphi \in cl(\varphi)$

$\varphi_1 \wedge \varphi_2 \in cl(\varphi) \Rightarrow \varphi_1, \varphi_2 \in cl(\varphi)$

$\varphi_1 \vee \varphi_2 \in cl(\varphi) \Rightarrow \varphi_1, \varphi_2 \in cl(\varphi)$

$X\varphi_1 \in cl(\varphi) \Rightarrow \varphi_1 \in cl(\varphi)$

$\varphi_1 U \varphi_2 \in cl(\varphi) \Rightarrow \varphi_1, \varphi_2 \in cl(\varphi)$

$\varphi_1 R \varphi_2 \in cl(\varphi) \Rightarrow \varphi_1, \varphi_2 \in cl(\varphi)$

Closure of a Formula

$$\varphi \in cl(\varphi)$$
$$\varphi_1 \wedge \varphi_2 \in cl(\varphi) \Rightarrow \varphi_1, \varphi_2 \in cl(\varphi)$$
$$\varphi_1 \vee \varphi_2 \in cl(\varphi) \Rightarrow \varphi_1, \varphi_2 \in cl(\varphi)$$
$$X\varphi_1 \in cl(\varphi) \Rightarrow \varphi_1 \in cl(\varphi)$$
$$\varphi_1 U \varphi_2 \in cl(\varphi) \Rightarrow \varphi_1, \varphi_2 \in cl(\varphi)$$
$$\varphi_1 R \varphi_2 \in cl(\varphi) \Rightarrow \varphi_1, \varphi_2 \in cl(\varphi)$$

Example

$$cl(F\neg p) = cl(true U \neg p) = \{F\neg p, \neg p, true\}$$

Hintikka Structure

A valid closure labeling $\tau : \mathbb{N} \to 2^{cl(\varphi)}$ of a sequence $\sigma : \mathbb{N} \to 2^{AP}$ has to satisfy.

If a formula $\varphi_1 \in cl(\varphi)$ labels a position i, then the sequence $\sigma^i \models \varphi_1$.

Rules for Labeling Sequences

1. $\mathit{false} \notin \tau(i)$;
2. for $p \in AP$, if $p \in \tau(i)$ then $p \in \sigma(i)$, and if $\neg p \in \tau(i)$ then $p \notin \sigma(i)$;
3. if $\varphi_1 \wedge \varphi_2 \in \tau(i)$ then $\varphi_1 \in \tau(i)$ and $\varphi_2 \in \tau(i)$;
4. if $\varphi_1 \vee \varphi_2 \in \tau(i)$ then $\varphi_1 \in \tau(i)$ or $\varphi_2 \in \tau(i)$;
5. if $X\varphi_1 \in \tau(i)$ then $\varphi_1 \in \tau(i+1)$;
6. if $\varphi_1 U \varphi_2 \in \tau(i)$ then either $\varphi_2 \in \tau(i)$, or $\varphi_1 \in \tau(i)$ and $\varphi_1 U \varphi_2 \in \tau(i+1)$;
7. if $\varphi_1 R \varphi_2 \in \tau(i)$ then $\varphi_2 \in \tau(i)$, and either $\varphi_1 \in \tau(i)$ or $\varphi_1 R \varphi_2 \in \tau(i+1)$;
8. if $\varphi_1 U \varphi_2 \in \tau(i)$ then there exists a $j > i$ such that $\varphi_2 \in \tau(j)$.

Key Lemmas

Lemma.
Consider a formula φ defined over a set of propositions AP, a sequence $\sigma : \mathbb{N} \to 2^{AP}$, and a closure labeling $\tau : \mathbb{N} \to 2^{cl(\varphi)}$ satisfying rules 1-8. For every formula $\varphi' \in cl(\varphi)$ and $i \geq 0$, one has that if $\varphi' \in \tau(i)$ then $\sigma^i \models \varphi'$.

Key Lemmas

Lemma.
Consider a formula φ defined over a set of propositions AP, a sequence $\sigma : \mathbb{N} \to 2^{AP}$, and a closure labeling $\tau : \mathbb{N} \to 2^{cl(\varphi)}$ satisfying rules 1-8. For every formula $\varphi' \in cl(\varphi)$ and $i \geq 0$, one has that if $\varphi' \in \tau(i)$ then $\sigma^i \models \varphi'$.

Lemma.
Consider a formula φ defined over a set of propositions AP and a sequence $\sigma : \mathbb{N} \to 2^{AP}$. If $\sigma \models \varphi$, there exists a closure labeling $\tau : \mathbb{N} \to 2^{cl(\varphi)}$ satisfying rules 1-8 and such that $\varphi \in \tau(0)$.

Correctness

Theorem

Consider a formula φ defined over a set of propositions AP and a sequence $\sigma : \mathbb{N} \to 2^{AP}$. One then has that $\sigma \models \varphi$, iff there is a closure labeling $\tau : \mathbb{N} \to 2^{cl(\varphi)}$ satisfying rules 1-8 and such that $\varphi \in \tau(0)$.

Defining the Automaton

Encoding to Büchi Automata Σ, S

Given a formula φ, a generalized Büchi automaton accepting exactly the sequences $\sigma : \mathbb{N} \to 2^{AP}$ satisfying φ can be defined as follows. The automaton is $A_\varphi = (\Sigma, S, \delta, S_0, \mathcal{F})$ where,

- $\Sigma = 2^{AP}$,
- $S \subseteq 2^{cl(\varphi)}$, and for each $s \in S$
  - $\mathit{false} \notin s$;
  - if $\varphi_1 \wedge \varphi_2 \in s$, then $\varphi_1 \in s$ and $\varphi_2 \in s$.
  - if $\varphi_1 \vee \varphi_2 \in s$, then $\varphi_1 \in s$ or $\varphi_2 \in s$.

Encoding to Büchi Automata δ, S_0

Given a formula φ, a generalized Büchi automaton accepting exactly the sequences $\sigma : \mathbb{N} \to 2^{AP}$ satisfying φ can be defined as follows. The automaton is $A_\varphi = (\Sigma, S, \delta, S_0, \mathcal{F})$ where,

- $t \in \delta(s, a)$ iff,
  - For all $p \in AP$, if $p \in s$ then $p \in a$.
  - For all $p \in AP$, if $\neg p \in s$ then $p \notin a$.
  - If $X\varphi \in s$, then $\varphi \in t$.
  - If $\varphi_1 U \varphi_2 \in s$ then either $\varphi_2 \in s$, or $\varphi_1 \in s$ and $\varphi_1 U \varphi_2 \in t$.
  - If $\varphi_1 R \varphi_2 \in s$ then $\varphi_2 \in s$ and either $\varphi_1 \in s$, or $\varphi_1 R \varphi_2 \in t$.

Encoding to Büchi Automata δ, S_0

Given a formula φ, a generalized Büchi automaton accepting exactly the sequences $\sigma : \mathbb{N} \to 2^{AP}$ satisfying φ can be defined as follows. The automaton is $A_\varphi = (\Sigma, S, \delta, S_0, \mathcal{F})$ where,

- $t \in \delta(s, a)$ iff,
  - For all $p \in AP$, if $p \in s$ then $p \in a$.
  - For all $p \in AP$, if $\neg p \in s$ then $p \notin a$.
  - If $X\varphi \in s$, then $\varphi \in t$.
  - If $\varphi_1 U \varphi_2 \in s$ then either $\varphi_2 \in s$, or $\varphi_1 \in s$ and $\varphi_1 U \varphi_2 \in t$.
  - If $\varphi_1 R \varphi_2 \in s$ then $\varphi_2 \in s$ and either $\varphi_1 \in s$, or $\varphi_1 R \varphi_2 \in t$.
- $S_0 = \{s \in S \mid \varphi \in s\}$.

Encoding to Büchi Automata $\mathcal{F}$

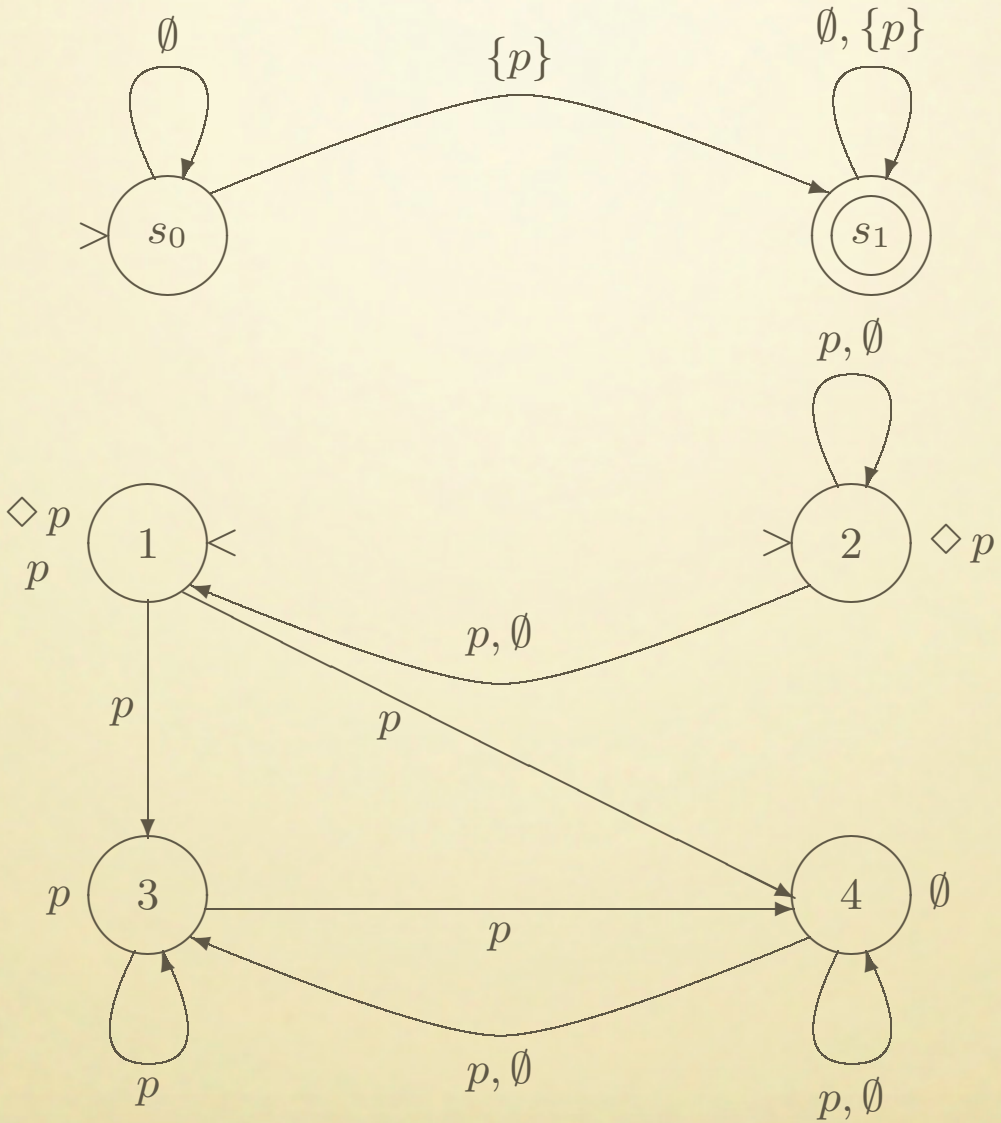
Given a formula φ, a generalized Büchi automaton accepting exactly the sequences $\sigma : \mathbb{N} \to 2^{AP}$ satisfying φ can be defined as follows. The automaton is $A_\varphi = (\Sigma, S, \delta, S_0, \mathcal{F})$ where,

- If the eventualities appearing in $cl(\varphi)$ are $e_1(\varphi_1), \ldots, e_m(\varphi_m)$, $\mathcal{F} = \{\Phi_1, \Phi_2 \ldots \Phi_m\}$, where

$$\Phi_i = \{s \in S \mid e_i(\varphi_i), \varphi_i \in s \vee e_i(\varphi_i) \notin s\}$$

for every eventuality formula $e(\varphi') = \varphi U \varphi'$

An Example *F p*



Optimizations

Omitting Redundant Transitions

Building the Automaton by Need

Identifying Equivalent States

Simplifying the Formula.

Early Detection of Inconsistencies.

Moving Propositions from States to Transitions.

Reports

Rep5. Bounded model checking for LTL (0/3) (Maximal 3 students).