IA159 Formal Verification Methods

LTL→BA via Alternating 1-Weak BA

Jan Strejček

Department of Computer Science
Faculty of Informatics
Masaryk University

Focus and sources

Focus

- alternating 1-weak Büchi automata (A1W)
- translation LTL→A1W
- translation A1W→BA

Source

- M. Y. Vardi: *An Automata-Theoretic Approach to Linear Temporal Logic*, LNCS 1043, Springer, 1995.

LTL→BA translations in general

- applications in automata-based LTL model checking, vacuity checking (checks trivial validity of a specification formula), . . .
- two LTL→BA translations
  - LTL → generalized Büchi automata → BA
  - LTL → alternating 1-weak Büchi automata → BA
- the latter translation is more popular
  - size-reducing optimizations of alternating 1-weak BA
  - produces smaller BA (in some cases)

Alternating Büchi automata

Positive boolean formulae

Positive boolean formulae over set Q ($\mathcal{B}^+(Q)$) are defined as

$$\varphi \ ::= \ \top \ | \ \bot \ | \ q \ | \ \varphi_1 \wedge \varphi_2 \ | \ \varphi_1 \vee \varphi_2$$

where $\top$ stands for true, $\bot$ stands for false, and q ranges over Q.

$S \subseteq Q$ is a model of φ $\iff$ the valuation assigning true just to elements of S satisfies φ

S is a minimal model of φ (written $S \models \varphi$) $\iff$ S is a model of φ and no proper subset of S is a model of φ

Examples of positive boolean formulae

formulae of $\mathcal{B}^+(\{p, q, r\})$	(minimal) models
$\bot$	no model
$\top$	$\emptyset, \{p\}, \{q\}, \{r\}, \{p, q\}, \ldots$
$p \wedge q$	$\{p, q\}, \{p, q, r\}$
$p \vee (q \wedge r)$	$\{p\}, \{p, q\}, \{p, r\}, \{q, r\}, \{p, q, r\}$
$p \wedge (q \vee r)$	$\{p, q\}, \{p, r\}, \{p, q, r\}$

Examples of positive boolean formulae

formulae of $\mathcal{B}^+(\{p, q, r\})$	(minimal) models
$\bot$	no model
$\top$	$\emptyset$, $\{p\}$, $\{q\}$, $\{r\}$, $\{p, q\}$, ...
$p \wedge q$	$\{p, q\}$, $\{p, q, r\}$
$p \vee (q \wedge r)$	$\{p\}$, $\{p, q\}$, $\{p, r\}$, $\{q, r\}$, $\{p, q, r\}$
$p \wedge (q \vee r)$	$\{p, q\}$, $\{p, r\}$, $\{p, q, r\}$

minimal models = clauses in disjunctive normal form

$$\varphi \equiv \bigvee_{S \models \varphi} \left(\bigwedge_{p \in S} p \right)$$

Alternating Büchi automata

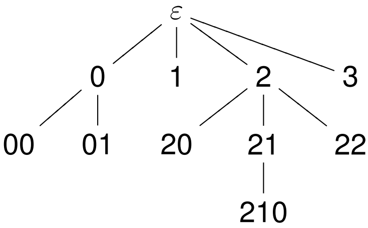
An alternating Büchi automaton is a tuple $\mathcal{A} = (\Sigma, Q, \delta, q_0, F)$, where

- Σ is a finite alphabet,
- Q is a finite set of states,
- $\delta : Q \times \Sigma \to \mathcal{B}^+(Q)$ is a transition function,
- $q_0 \in Q$ is an initial state,
- $F \subseteq Q$ is a set of accepting states.

Trees

A tree is a set $T \subseteq \mathbb{N}_0^*$ such that if $xc \in T$, where $x \in \mathbb{N}_0^*$ and $c \in \mathbb{N}_0$, then also

- $x \in T$ and
- $xc' \in T$ for all $0 \leq c' < c$.

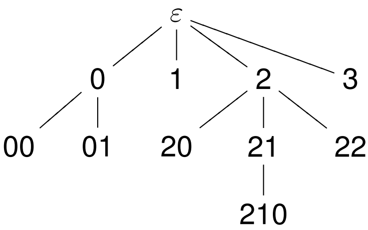


$$T = \{\ \varepsilon, 0, 1, 2, 3, 00, 01, 20, 21, 22, 210\ \}$$

Trees

A tree is a set $T \subseteq \mathbb{N}_0^*$ such that if $xc \in T$, where $x \in \mathbb{N}_0^*$ and $c \in \mathbb{N}_0$, then also

- $x \in T$ and
- $xc' \in T$ for all $0 \leq c' < c$.



$$T = \{ \varepsilon, 0, 1, 2, 3, 00, 01, 20, 21, 22, 210 \}$$

A Q-labeled tree is a pair (T, r) of a tree T and a labeling function $r : T \to Q$.

Alternating Büchi automata: a run

A run of an alternating BA $\mathcal{A} = (\Sigma, Q, \delta, q_0, F)$ on word $w = w(0)w(1)\ldots \in \Sigma^\omega$ is a Q-labeled tree (T, r) such that

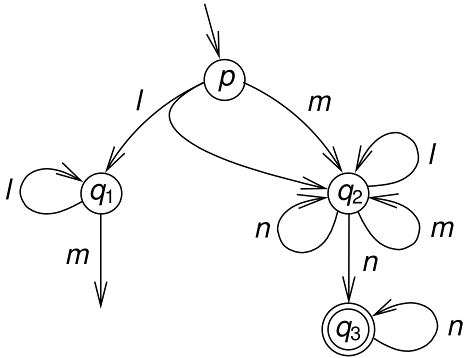
- $r(\varepsilon) = q_0$ and
- for each $x \in T$: $\{r(xc) \mid c \in \mathbb{N}_0, xc \in T\} \models \delta(r(x), w(|x|))$.

A run (T, r) is accepting iff for each infinite path π in T it holds that $\mathit{Inf}(\pi) \cap F \neq \emptyset$, where $\mathit{Inf}(\pi)$ is the set of all labels (i.e. states) appearing on π infinitely often.

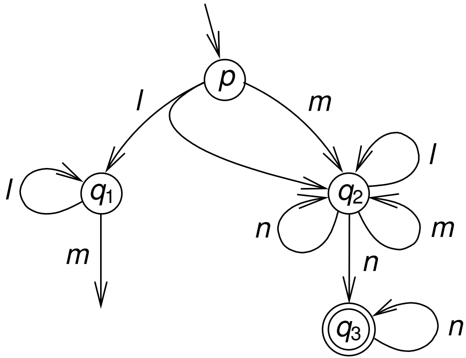
An automaton $\mathcal{A}$ accepts a word w iff there is an accepting run of $\mathcal{A}$ on w. We set

$$L(\mathcal{A}) = \{w \in \Sigma^\omega \mid \mathcal{A} \text{ accepts } w\}.$$

Example of an alternating Büchi automaton



Example of an alternating Büchi automaton



Accepts the language $l^* m(l + m + n)^* n^\omega$.

Alternating 1-weak Büchi automata (A1W)

Let $\mathcal{A} = (\Sigma, Q, \delta, q_0, F)$ be an alternating BA. For each $p \in Q$ we define the set of all successors of p as

$$Succ(p) = \{q \mid \exists l \in \Sigma, S \subseteq Q : S \cup \{q\} \models \delta(p, l)\}.$$

Automaton $\mathcal{A}$ is 1-weak (or linear or very weak) if there exists a partial order $\leq$ on Q such that for all $p, q \in Q$ it holds:

$$q \in Succ(p) \implies q \leq p$$

alternating 1-weak Büchi automaton = A1W automaton

Notes

- standard Büchi automata are alternating Büchi automata where each $\delta(p, l)$ is $\perp$ or a disjunction of states
- A1W automata have the same expressive power as LTL

LTL→A1W

LTL→A1W

Input: an LTL formula φ and an alphabet $\Sigma = 2^{AP'}$ for some finite $AP' \subseteq AP$

Output: A1W automaton $\mathcal{A} = (\Sigma, Q, \delta, q_\varphi, F)$ accepting $L^\Sigma(\varphi)$

LTL→A1W

Input: an LTL formula φ and an alphabet $\Sigma = 2^{AP'}$ for some finite $AP' \subseteq AP$

Output: A1W automaton $\mathcal{A} = (\Sigma, Q, \delta, q_\varphi, F)$ accepting $L^\Sigma(\varphi)$

- $Q = \{q_\psi, q_{\neg\psi} \mid \psi \text{ is a subformula of } \varphi\}$

LTL→A1W

Input: an LTL formula φ and an alphabet $\Sigma = 2^{AP'}$ for some finite $AP' \subseteq AP$

Output: A1W automaton $\mathcal{A} = (\Sigma, Q, \delta, q_\varphi, F)$ accepting $L^\Sigma(\varphi)$

- $Q = \{q_\psi, q_{\neg\psi} \mid \psi \text{ is a subformula of } \varphi\}$
- δ is defined as follows (where $\overline{\alpha} \in \mathcal{B}^+(Q)$ satisfies $\overline{\alpha} \equiv \neg\alpha$)

$$
\begin{array}{lcl}
\delta(q_\top, I) & = & \top \\
\delta(q_a, I) & = & \top \text{ if } a \in I, \ \bot \text{ otherwise} \\
\delta(q_{\neg\psi}, I) & = & \overline{\delta(q_\psi, I)} \\
\delta(q_{\psi\wedge\rho}, I) & = & \delta(q_\psi, I) \wedge \delta(q_\rho, I) \\
\delta(q_{\mathsf{X}\psi}, I) & = & q_\psi \\
\delta(q_{\psi\mathsf{U}\rho}, I) & = & \delta(q_\rho, I) \vee (\delta(q_\psi, I) \wedge q_{\psi\mathsf{U}\rho})
\end{array}
\qquad
\begin{array}{lcl}
\overline{\top} & = & \bot \\
\overline{\bot} & = & \top \\
\overline{q_{\neg\psi}} & = & q_\psi \\
\overline{q_\psi} & = & q_{\neg\psi} \\
\overline{\beta \wedge \gamma} & = & \overline{\beta} \vee \overline{\gamma} \\
\overline{\beta \vee \gamma} & = & \overline{\beta} \wedge \overline{\gamma}
\end{array}
$$

Input: an LTL formula φ and an alphabet $\Sigma = 2^{AP'}$ for some finite $AP' \subseteq AP$

Output: A1W automaton $\mathcal{A} = (\Sigma, Q, \delta, q_\varphi, F)$ accepting $L^\Sigma(\varphi)$

- $Q = \{q_\psi, q_{\neg\psi} \mid \psi \text{ is a subformula of } \varphi\}$
- δ is defined as follows (where $\overline{\alpha} \in \mathcal{B}^+(Q)$ satisfies $\overline{\alpha} \equiv \neg\alpha$)

$$
\begin{aligned}
\delta(q_\top, I) &= \top \\
\delta(q_a, I) &= \top \text{ if } a \in I,\ \bot \text{ otherwise} \\
\delta(q_{\neg\psi}, I) &= \overline{\delta(q_\psi, I)} \\
\delta(q_{\psi\wedge\rho}, I) &= \delta(q_\psi, I) \wedge \delta(q_\rho, I) \\
\delta(q_{\mathsf{X}\psi}, I) &= q_\psi \\
\delta(q_{\psi\mathsf{U}\rho}, I) &= \delta(q_\rho, I) \vee (\delta(q_\psi, I) \wedge q_{\psi\mathsf{U}\rho})
\end{aligned}
\qquad
\begin{aligned}
\overline{\top} &= \bot \\
\overline{\bot} &= \top \\
\overline{q_{\neg\psi}} &= q_\psi \\
\overline{q_\psi} &= q_{\neg\psi} \\
\overline{\beta \wedge \gamma} &= \overline{\beta} \vee \overline{\gamma} \\
\overline{\beta \vee \gamma} &= \overline{\beta} \wedge \overline{\gamma}
\end{aligned}
$$

- $F = \{q_{\neg(\psi\mathsf{U}\rho)} \mid \psi\,\mathsf{U}\,\rho \text{ is a subformula of } \varphi\}$

LTL→A1W

Theorem

Given an LTL formula φ and an alphabet Σ, one can construct an A1W automaton $\mathcal{A}$ accepting $L^{\Sigma}(\varphi)$ and such that the number of states of $\mathcal{A}$ is linear in the length of φ.

LTL→BA via alternating 1-weak BA

A1W→BA

A1W→BA

Input: an alternating BA $\mathcal{A} = (\Sigma, Q, \delta, q_0, F)$
Output: a BA $\mathcal{A}' = (\Sigma, Q', \delta', q'_0, F')$ accepting $L(\mathcal{A})$

A1W→BA

Input: an alternating BA $\mathcal{A} = (\Sigma, Q, \delta, q_0, F)$
Output: a BA $\mathcal{A}' = (\Sigma, Q', \delta', q_0', F')$ accepting $L(\mathcal{A})$

Intuitively, $\mathcal{A}'$ guesses labeling of each level of the computation tree of $\mathcal{A}$. Moreover, $\mathcal{A}'$ has to divide the set of states into two sets: states labeling paths with recent occurrence of an accepting states and the other states.

A1W→BA

Input: an alternating BA $\mathcal{A} = (\Sigma, Q, \delta, q_0, F)$
Output: a BA $\mathcal{A}' = (\Sigma, Q', \delta', q'_0, F')$ accepting $L(\mathcal{A})$

- $Q' = 2^Q \times 2^Q$

A1W→BA

Input: an alternating BA $\mathcal{A} = (\Sigma, Q, \delta, q_0, F)$
Output: a BA $\mathcal{A}' = (\Sigma, Q', \delta', q_0', F')$ accepting $L(\mathcal{A})$

- $Q' = 2^Q \times 2^Q$
- $q_0' = (\{q_0\}, \emptyset)$

A1W→BA

Input: an alternating BA $\mathcal{A} = (\Sigma, Q, \delta, q_0, F)$
Output: a BA $\mathcal{A}' = (\Sigma, Q', \delta', q_0', F')$ accepting $L(\mathcal{A})$

- $Q' = 2^Q \times 2^Q$
- $q_0' = (\{q_0\}, \emptyset)$
- $\delta'((U, V), l)$ is defined as:
  - if $U \neq \emptyset$ then

$$\begin{aligned}\delta'((U,V),l) = \{(U',V') \mid\ & \exists X, Y \subseteq Q \text{ such that} \\ & X \models \textstyle\bigwedge_{q \in U} \delta(q,l) \text{ and} \\ & Y \models \textstyle\bigwedge_{q \in V} \delta(q,l) \text{ and} \\ & U' = X \smallsetminus F \text{ and } V' = Y \cup (X \cap F)\}\end{aligned}$$

  - if $U = \emptyset$ then

$$\begin{aligned}\delta'((\emptyset,V),l) = \{(U',V') \mid\ & \exists Y \subseteq Q \text{ such that} \\ & Y \models \textstyle\bigwedge_{q \in V} \delta(q,l) \text{ and} \\ & U' = Y \smallsetminus F \text{ and } V' = Y \cap F)\}\end{aligned}$$

A1W→BA

Input: an alternating BA $\mathcal{A} = (\Sigma, Q, \delta, q_0, F)$
Output: a BA $\mathcal{A}' = (\Sigma, Q', \delta', q_0', F')$ accepting $L(\mathcal{A})$

- $Q' = 2^Q \times 2^Q$
- $q_0' = (\{q_0\}, \emptyset)$
- $\delta'((U, V), l)$ is defined as:
  - if $U \neq \emptyset$ then

$$\delta'((U,V),l) = \{(U',V') \mid \exists X, Y \subseteq Q \text{ such that } X \models \bigwedge_{q \in U} \delta(q,l) \text{ and } Y \models \bigwedge_{q \in V} \delta(q,l) \text{ and } U' = X \smallsetminus F \text{ and } V' = Y \cup (X \cap F)\}$$

  - if $U = \emptyset$ then

$$\delta'((\emptyset,V),l) = \{(U',V') \mid \exists Y \subseteq Q \text{ such that } Y \models \bigwedge_{q \in V} \delta(q,l) \text{ and } U' = Y \smallsetminus F \text{ and } V' = Y \cap F)\}$$

- $F' = \{\emptyset\} \times 2^Q$

LTL→A1W

Theorem

Given an alternating BA $\mathcal{A} = (\Sigma, Q, \delta, q_0, F)$, one can construct a BA $\mathcal{A}'$ accepting $L(\mathcal{A})$ and such that the number of states of $\mathcal{A}'$ is $2^{\mathcal{O}(|Q|)}$.

Corollary

Given an LTL formula φ and an alphabet Σ, one can construct a BA $\mathcal{A}'$ accepting $L^{\Sigma}(\varphi)$ and such that the number of states of $\mathcal{A}'$ is $2^{\mathcal{O}(|\varphi|)}$.

Coming next week

Partial order reduction

- When can a state/transition be safely removed from a Kripke structure?
- What is a stuttering principle?
- Can we effectively compute the reduction?