From LTL to Deterministic ω-automata

Vaishnavi Sundararajan
Chennai Mathematical Institute

Formal Methods Update Meeting
IIT Delhi
July 27 & 28, 2013

Outline

Why deterministic automata?

- Model-checking needs only nondeterminstic Büchi automata (NBAs) for emptiness checking
- Deterministic automata needed for important problems like
  - Synthesis of reactive modules for LTL specifications
  - Model-checking Markov decision processes
- NBA to deterministic Rabin automaton (DRA)

What [1] does

- Considers the (**F**, **G**)-fragment of LTL for direct translation to DRAs
- Constructs deterministic Muller automaton for input formula φ
- States are formulas, not atoms (maximal consistent set of subformulas)
- Efficiently transforms this to a standard DRA

($\mathbf{F}$, $\mathbf{G}$)-fragment of LTL: Syntax

$$\varphi, \psi \in \Phi ::= a \mid \neg a \mid \varphi \wedge \psi \mid \varphi \vee \psi \mid \mathbf{F}\varphi \mid \mathbf{G}\varphi$$

where $a \in Ap$, Ap a finite fixed set of atomic propositions.

- Standard abbreviations: $\mathbf{tt} := a \vee \neg a, \mathbf{ff} := a \wedge \neg a$
- Push negations inside to atomic propositions, $\mathbf{F}a = \neg\mathbf{G}\neg a$
- *No* $\mathbf{X}$ *or* $\mathbf{U}$ *allowed in formulas!*

(**F**, **G**)-fragment of LTL: Semantics

Word $w = w[0]w[1]\cdots \in (2^{Ap})^\omega$
i^{th} suffix of w: $w_i = w[i]w[i+1]\cdots$

$$w \models a \iff a \in w[0]$$

$$w \models \neg a \iff a \notin w[0]$$

$$w \models \varphi \wedge \psi \iff w \models \varphi \text{ and } w \models \psi$$

$$w \models \varphi \vee \psi \iff w \models \varphi \text{ or } w \models \psi$$

$$w \models \mathbf{F}\varphi \iff \exists k \geq 0 \ w_k \models \varphi$$

$$w \models \mathbf{G}\varphi \iff \forall k \geq 0 \ w_k \models \varphi$$

Symbolic one-step unfolding $\mathfrak{U}$

$$\mathfrak{U}(a) = a$$
$$\mathfrak{U}(\neg a) = \neg a$$
$$\mathfrak{U}(\varphi \wedge \psi) = \mathfrak{U}(\varphi) \wedge \mathfrak{U}(\psi)$$
$$\mathfrak{U}(\varphi \vee \psi) = \mathfrak{U}(\varphi) \vee \mathfrak{U}(\psi)$$
$$\mathfrak{U}(\mathbf{F}\varphi) = \mathfrak{U}(\varphi) \vee \mathbf{XF}\varphi$$
$$\mathfrak{U}(\mathbf{G}\varphi) = \mathfrak{U}(\varphi) \wedge \mathbf{XG}\varphi$$

Example 1

$$\begin{aligned}
\mathfrak{U}(\mathbf{F}(\mathbf{G}a \vee \mathbf{G}b) &= \mathfrak{U}(\mathbf{G}a \vee \mathbf{G}b) \vee \mathbf{XF}(\mathbf{G}a \vee \mathbf{G}b) \\
&= \mathfrak{U}(\mathbf{G}a) \vee \mathfrak{U}(\mathbf{G}b) \vee \mathbf{XF}(\mathbf{G}a \vee \mathbf{G}b) \\
&= (a \wedge \mathbf{XG}a) \vee (b \wedge \mathbf{XG}b) \vee \mathbf{XF}(\mathbf{G}a \vee \mathbf{G}b)
\end{aligned}$$

Notation

For φ, an arbitrary but fixed formula

- $\mathbb{F}, \mathbb{G}$: Sets of all subformulae of φ of form $\mathbf{F}\psi, \mathbf{G}\psi$ respectively
- $\mathbb{T} := \mathbb{F} \cup \mathbb{G}$: Set of all temporal formulae
- $\mathbf{X}\Psi := \{\mathbf{X}\psi \mid \psi \in \Psi\}$ for a set of formulae Ψ
- $\mathbb{C}(\varphi) := Ap \cup \{\neg a \mid a \in Ap\} \cup \mathbf{X}\mathbb{T}$ is the *closure* of φ
- states(φ) is the set $2^{2^{\mathbb{C}(\varphi)}}$
- ψ, χ: Element of states(φ), positive Boolean formula over $\mathbb{C}(\varphi)$
- α, β: One-step history of the word read

More notation

- For $\psi \in \text{states}(\varphi)$ and $\alpha \subseteq Ap$, $\text{red}(\psi, \alpha)$, called the α-reduct of ψ, is the formula got by:
  - Replacing all $a \in \alpha$ not occurring inside a modal context in ψ by **tt**.
  - Replacing all $a \in Ap \setminus \alpha$ not inside a modal context in ψ by **ff**
- $\text{red}(\psi, \alpha)$ is a positive boolean combination of formulas of the form $\mathbf{X}\psi'$ where $\psi' \in \mathbb{T}$.
- Since $\mathbf{X}$ distributes over $\wedge$ and $\vee$, $\text{red}(\psi, \alpha)$ is equivalent to $\mathbf{X}\chi$ where χ is a positive Boolean formula over $\mathbb{T}$.

Deterministic Automaton

For a formula φ, we define $\mathcal{A}(\varphi) = (Q, i, \delta)$ to be a deterministic finite automaton over $\Sigma = 2^{Ap}$, where

- Set of states $Q = \{i\} \cup \big(\text{states}(\varphi) \times 2^{Ap}\big)$
- Initial state i
- Transition function δ can be partitioned into the two following sets
  - $\{(i, \alpha, \langle \mathfrak{U}(\varphi), \alpha\rangle)\}$
  - $\{\big(\langle\psi, \alpha\rangle, \beta, \big\langle \mathfrak{U}(\mathbf{X}^{-1}\text{red}(\psi, \alpha)), \beta\big\rangle\big) \mid \langle\psi, \alpha\rangle \in Q, \beta \in \Sigma\}$

where $\mathbf{X}^{-1}\psi$ removes $\mathbf{X}$s from ψ.
Intuitively, a state (ψ, α) corresponds to the situation where α is being read and ψ needs to be satisfied.

Example: $\varphi = \mathbf{F}(\mathbf{G}a \vee \mathbf{G}b)$

$$\mathfrak{U}(\varphi) = (a \wedge \mathbf{XG}a) \vee (b \wedge \mathbf{XG}b) \vee (X\mathbf{F}\varphi)$$

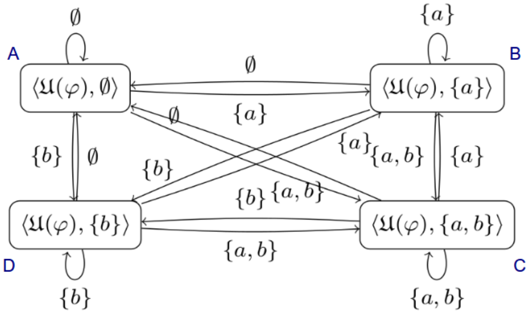

Figure: Automaton $\mathcal{A}_\varphi$ for $\mathbf{F}(\mathbf{G}a \vee \mathbf{G}b)$

Example: $\varphi = \mathbf{F}(\mathbf{G}a \vee \mathbf{G}b)$

- Words $\mathcal{A}_\varphi$ should accept: $ababab(a)^\omega$, $ababa(b)^\omega$, a^ω etc
- Words $\mathcal{A}_\varphi$ should reject: $(ab)^\omega$, $(aba)^\omega$ etc
- Both a and b false in state A: A cannot be in a Muller accepting set.
- $\{B, C, D\}$ not a Muller accepting set: neither $\mathbf{G}a$ nor $\mathbf{G}b$ is eventually made true.
- $\{B\}, \{C\}$ and $\{D\}$ are Muller accepting sets for runs $(a)^\omega, (\{a, b\})^\omega$ and $(b)^\omega$ respectively
- $\{B, C\}$ and $\{C, D\}$ are Muller accepting sets for runs $(a\{a, b\})^\omega$ and $(b\{a, b\})^\omega$ respectively

Muller accepting sets for $\mathcal{A}_\varphi = \{\{B\}, \{C\}, \{D\}, \{B, C\}, \{C, D\}\}$

Corresponding Rabin pairs for $\mathcal{A}_\varphi = (\{B, C\}, \{A, D\}), (\{C, D\}, \{A, B\})$

Muller acceptance condition

A set $M \subseteq Q$ is *Muller accepting* for φ if there is a set $I \subseteq \mathbb{T}$ such that the following are satisfied:

1. C_1: For each $(\chi, \alpha) \in M$, we have $\mathbf{X}I \models_\alpha \chi$,
2. C_2: For each $\mathbf{F}\psi \in I$, there is $(\chi, \alpha) \in M$ with $I \models_\alpha \psi$,
3. C_3: For each $\mathbf{G}\psi \in I$ and for each $(\chi, \alpha) \in M$, $I \models_\alpha \psi$,

where $I \models_\alpha \chi$ is shorthand for saying that $I \implies \mathrm{red}(\chi, \alpha)$ is (an instance of) a propositional tautology.

- M is Muller accepting for φ if it is Muller accepting for some I.
- Acceptance condition for φ: Set of all Muller accepting sets $\{M_1, M_2, \cdots\}$.

Example: $\varphi = \mathbf{F}(\mathbf{G}a \vee \mathbf{G}b)$

$$\mathbb{T} = \{\{\mathbf{G}a\}, \{\mathbf{G}b\}, \varphi\} \qquad I = \{\mathbf{G}a\} \subseteq \mathbb{T}$$

$$\chi = \mathfrak{U}(\varphi) = (a \wedge \mathbf{XG}a) \vee (b \wedge \mathbf{XG}b) \vee \mathbf{XF}\varphi$$

Condition	Required	Possible choices for M
C_1	$\models_{PL} XGa \implies \mathrm{red}(\chi, \alpha)$	$\{B\}, \{C\}, \{B, C\}$
C_2	No **F** conditions in I	$\{B\}, \{C\}, \{B, C\}$
C_3	$\models_{PL} Ga \implies \mathrm{red}(a, \alpha)$	$\{B\}, \{C\}, \{B, C\}$

Each of $\{B\}, \{C\}$ and $\{B, C\}$ is Muller accepting for $I = \{\mathbf{G}a\}$.
Doing this for each $I \subseteq \mathbb{T}$, we get

Acceptance condition for $\varphi : \{\{B\}, \{C\}, \{D\}, \{B, C\}, \{C, D\}\}$

Correctness

Theorem 1

Let φ be a formula and w a word. Then w is accepted by the deterministic automaton $\mathcal{A}(\varphi)$ with the Muller condition $\mathcal{M}(\varphi)$ iff $w \models \varphi$.

Proposition 1.1 (Finitary correctness)

Let w be a word and $\mathcal{A}(\varphi)(w) = i(\chi_0, \alpha_0)(\chi_1, \alpha_1)\cdots$ the corresponding run. Then, for all $n \in \mathbb{N}$, we have $w \models \varphi$ iff $w_n \models \chi_n$.

Proposition 1.2 (Completeness)

If $w \models \phi$ then $Inf(\mathcal{A}(\phi)(w))$ is a Muller accepting set.

$M := \mathrm{Inf}(\mathcal{A}(\phi)(w))$ is Muller accepting for

$$I := \{\psi \in \mathbb{F} \mid w \models \mathbf{G}\psi\} \cup \{\psi \in \mathbb{G} \mid w \models \mathbf{F}\psi\}$$

Soundness

Proposition 1.3

Let ρ be a run. If Inf(ρ) is Muller accepting for I, then

- $Ap(\rho) \models \mathbf{G}\psi$ *for each* $\psi \in I \cap \mathbb{F}$ *and*
- $Ap(\rho) \models \mathbf{F}\psi$ *for each* $\psi \in I \cap \mathbb{G}$

Proposition 1.4 (Soundness)

If Inf($\mathcal{A}(\phi)(w)$) is a Muller accepting set then $w \models \phi$.

Generalized Rabin automaton

A generalized Rabin automaton is a deterministic ω-automaton $=(Q, i, \delta)$ together with a generalized Rabin condition $\mathcal{GR} \in \mathcal{B}^{+}(2^{Q} \times 2^{Q})$. A run ρ of $\mathcal{A}$ is accepting if $\mathrm{Inf}(\rho) \models \mathcal{GR}$.

For a formula φ, the generalized Rabin condition $\mathcal{GR}(\varphi)$ is

$$\bigvee_{I \subseteq \mathbb{T}} \left(\left(\{(\chi, \alpha) \mid I \not\models_{\alpha} \chi \wedge \bigwedge_{\mathbf{G}\psi \in I} \psi\}, Q \right) \wedge \bigwedge_{\mathbf{F}\omega \in I} (\emptyset, \{(\chi, \alpha) \mid I \models_{\alpha} \omega\}) \right)$$

Proposition 1.5

Let φ be a formula and w a word. Then w is accepted by the deterministic automaton $\mathcal{A}(\varphi)$ with the generalized Rabin condition $\mathcal{GR}(\varphi)$ iff $w \models \varphi$.

Can efficiently obtain a set of Rabin pairs for φ from $\mathcal{GR}(\varphi)$.

Summary

- Considers only reachable state space
- In state (χ, α), α only records letters from χ
- Smaller automata than ltl2dstar for most fairness conditions
- More optimizations in the Rabinizer tool [2]
  - Redundant states removed
  - Merges conjunctions of "compatible" Rabin pairs
  - One-step history considers equivalence classes of letters
  - No special initial state without any other use

Bibliography

- Jan Kretínský and Javier Esparza:
  Deterministic Automata for the (F, G)-Fragment of LTL
  CAV (2012) 7–22.

- Andreas Gaiser, Jan Kretínský and Javier Esparza:
  Rabinizer: Small Deterministic Automata for LTL(F, G)
  ATVA (2012) 72–76.

Thank you!