Constructions in Rabinizer 2

Abstract. We provide the complete construction of automata and acceptance conditions of Rabinizer and show their correctness.

1 Linear Temporal Logic

This section recalls the notion of linear temporal logic (LTL). We consider a fragment with no occurrence of **U** inside any **G**:

Definition 1 (LTL Syntax). *The formulae of the* $LTL_{\setminus \mathbf{GU}}$ *-fragment of linear temporal logic are given by the following syntax for* φ*:*

$$\varphi ::= a \mid \neg a \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \mathbf{X}\varphi \mid \varphi \mathbf{U} \varphi \mid \mathbf{F}\varphi \mid \mathbf{G}\xi$$
$$\xi ::= a \mid \neg a \mid \xi \wedge \xi \mid \xi \vee \xi \mid \mathbf{X}\xi \mid \mathbf{F}\xi \mid \mathbf{G}\xi$$

over a finite fixed set Ap *of atomic propositions.*

We use the standard abbreviations $\mathbf{tt} := a \vee \neg a$, $\mathbf{ff}; = a \wedge \neg a$. We only have negations of atomic propositions, as negations can be pushed inside due to the equivalence of $\mathbf{F}\varphi$ and $\neg\mathbf{G}\neg\varphi$.

Definition 2 (LTL Semantics). *Let* $w \in (2^{Ap})^\omega$ *be a word. The* i*th letter of* w *is denoted* $w[i]$*, i.e.* $w = w[0]w[1]\cdots$*. Further, we define the* i*th suffix of* w *as* $w_i = w[i]w[i+1]\cdots$*. The semantics of a formula on* w *is then defined inductively as follows:*

$$\begin{aligned}
w &\models a &&\iff a \in w[0]\\
w &\models \neg a &&\iff a \notin w[0]\\
w &\models \varphi \wedge \psi &&\iff w \models \varphi \text{ and } w \models \psi\\
w &\models \varphi \vee \psi &&\iff w \models \varphi \text{ or } w \models \psi\\
w &\models \mathbf{X}\varphi &&\iff w_1 \models \varphi\\
w &\models \mathbf{F}\varphi &&\iff \exists\, k \in \mathbb{N} : w_k \models \varphi\\
w &\models \mathbf{G}\varphi &&\iff \forall\, k \in \mathbb{N} : w_k \models \varphi\\
w &\models \varphi \mathbf{U} \psi &&\iff \exists\, k \in \mathbb{N} : w_k \models \psi \text{ and } \forall\, 0 \le j < k : w_j \models \varphi
\end{aligned}$$

2 Preliminaries

We define a symbolic one-step unfolding $\mathfrak{Unf}$ of a formula inductively by the following rules:

$$\begin{aligned}
\mathfrak{Unf}(a) &= a\\
\mathfrak{Unf}(\neg a) &= \neg a\\
\mathfrak{Unf}(\varphi \wedge \psi) &= \mathfrak{Unf}(\varphi) \wedge \mathfrak{Unf}(\psi)\\
\mathfrak{Unf}(\varphi \vee \psi) &= \mathfrak{Unf}(\varphi) \vee \mathfrak{Unf}(\psi)\\
\mathfrak{Unf}(\mathbf{X}\varphi) &= \mathbf{X}\varphi\\
\mathfrak{Unf}(\mathbf{F}\varphi) &= \mathfrak{Unf}(\varphi) \vee \mathbf{XF}\varphi\\
\mathfrak{Unf}(\mathbf{G}\varphi) &= \mathfrak{Unf}(\varphi) \wedge \mathbf{XG}\varphi\\
\mathfrak{Unf}(\varphi\mathbf{U}\psi) &= \mathfrak{Unf}(\psi) \vee (\mathfrak{Unf}(\varphi) \wedge \mathbf{X}(\varphi\mathbf{U}\psi))
\end{aligned}$$

Further, we define the "next step" operator. This peels off one next operator wherever possible. We define

$$\begin{aligned}
\mathbf{X}^{-1}(\psi_1 \wedge \psi_2) &= \mathbf{X}^{-1}(\psi_1) \wedge \mathbf{X}^{-1}(\psi_2)\\
\mathbf{X}^{-1}(\psi_1 \vee \psi_2) &= \mathbf{X}^{-1}(\psi_1) \vee \mathbf{X}^{-1}(\psi_2)\\
\mathbf{X}^{-1}(\mathbf{X}\psi) &= \psi\\
\mathbf{X}^{-1}(\psi) &= \psi \text{ for all other types of formulae}
\end{aligned}$$

3 Algorithm

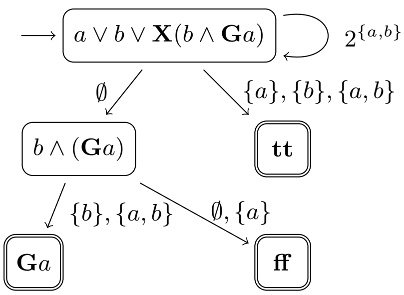
3.1 Construction of $\mathcal{B}(\xi)$

We define a finite automaton $\mathcal{B}(\xi) = (Q_\xi, i_\xi, \delta_\xi, F_\xi)$ over 2^{Ap} by

- the set of states $Q_\xi = \mathsf{B}^+(\mathrm{sf}(\xi))$, where $\mathsf{B}^+(S)$ is the set of positive Boolean functions over S and $\mathbf{tt}$ and $\mathbf{ff}$,
- the initial state $i_\xi = \xi$,
- the final states F_ξ where each atomic proposition has $\mathbf{F}$ or $\mathbf{G}$ as an ancestor in the syntactic tree (i.e. no atomic propositions are guarded by only $\mathbf{X}$'s and Boolean connectives),
- transition relation δ_ξ is defined by transitions

$$\begin{array}{ll}
\chi \xrightarrow{\nu} \mathbf{X}^{-1}(\chi[\nu]) & \text{for every } \nu \subseteq Ap \text{ and } \chi \notin F\\
i \xrightarrow{\nu} i & \text{for every } \nu \subseteq Ap
\end{array}$$

where $\chi[\nu]$ is the function χ with $\mathbf{tt}$ and $\mathbf{ff}$ plugged in for atomic propositions according to ν and $\mathbf{X}^{-1}\chi$ strips away the initial $\mathbf{X}$ (whenever there is one) from each formula in the Boolean combination χ. Note that we do not unfold inner $\mathbf{F}$- and $\mathbf{G}$-formulae. See an example for $\xi = a \vee b \vee \mathbf{X}(b \wedge \mathbf{G}a)$ below.



3.2 Construction of $\mathcal{A}(\varphi)$

The state space has two components. Beside the component keeping track of the input formula, we also keep track of the history for every recurrent formula of $\mathcal{R}$ec. The second component is then a vector of length $|\mathcal{R}\text{ec}|$ keeping the current set of states of each $\mathcal{B}(\xi)$. Formally, we define $\mathcal{A}(\varphi) = (Q, i, \delta)$ to be a deterministic finite automaton over $\Sigma = 2^{Ap}$ given by

– set of states $Q = \mathbb{C} \times \prod_{\xi \in \mathcal{R}\text{ec}} 2^{Q_\xi}$
  where $\mathbb{C} = \mathsf{B}^+(\text{sf}(\varphi) \cup \mathbf{X}\text{sf}(\varphi))$ and $\mathbf{X}S = \{\mathbf{X}s \mid s \in S\}$,
– the initial state $i = \langle \mathfrak{Unf}(\varphi), (\xi \mapsto \{i_\xi\})_{\xi \in \mathcal{R}\text{ec}} \rangle$;
– the transition function δ is defined by transitions

$$\langle \psi, (R_\xi)_{\xi \in \mathcal{R}\text{ec}} \rangle \xrightarrow{\nu} \langle \mathfrak{Unf}(\mathbf{X}^{-1}(\psi[\nu])), \big(\delta_\xi(R_\xi, \nu)\big)_{\xi \in \mathcal{R}\text{ec}} \rangle$$

3.3 Construction of generalized Rabin pairs condition $\mathcal{C}(\varphi)$

We modify the construction of [KE12] and we provide a generalized Rabin condition of the form $\bigvee_i (F_i, \bigwedge_j I_{ij})$ which can be easily degeneralized to a Rabin condition, where the conjunction is a singleton. In essence, the acceptance condition is responsible for non-deterministically guessing a set I of subformulae that hold infinitely often and then checks that (1) they indeed hold infinitely often and (2) if they hold infintiely often then also φ was satisfied in the initial state.

As for (1), whenever $\mathbf{F}\chi \in I$, we need to visit

$$\mathit{reach}_{\mathbf{F}\chi} := \{\langle \psi, (R_\xi)_{\xi \in \mathcal{R}\text{ec}(\psi)} \rangle \in Q \mid \exists q \in R_\chi \cap F_\chi : \mathbf{X}^* I \models q\}\}$$

infinitely often, where $\mathbf{X}^* S = \{\underbrace{\mathbf{X} \cdots \mathbf{X}}_{n} s \mid s \in S, n \in \mathbb{N}_0\}$. Similarly, whenever $\mathbf{G}\chi \in I$, we need to visit

$$\mathit{avoid}_{\mathbf{G}\chi} := \{\langle \psi, (R_\xi)_{\xi \in \mathcal{R}\text{ec}(\psi)} \rangle \in Q \mid \exists q \in R_\chi \cap F_\chi : \mathbf{X}^* I \not\models q\}\}$$

only finitely often. As for (2), we allow only finitely many visits of states

$$\mathit{avoid} := \{\langle \psi, (R_\xi)_{\xi \in \mathcal{R}\text{ec}(\psi)} \rangle \in Q \mid \mathbf{X}^* I \cup \bigcup_{\substack{\xi \in \mathcal{R}\text{ec}(\psi) \\ G\xi \in I}} R_\xi \not\models \psi\}\}$$

where the set I is insufficient to prove that φ holds. Altogether

$$\mathcal{C} := \bigvee_{I \subseteq \mathbf{G}_\varphi \cup \mathbf{F}_\varphi} \Big(\mathit{avoid} \cup \bigcup_{\mathbf{G}\chi \in I} \mathit{avoid}_{\mathbf{G}\chi}, \bigwedge_{\mathbf{F}\chi \in I} \mathit{reach}_{\mathbf{F}\chi} \Big)$$

4 Correctness

Given a formula φ, we have defined a Rabin automaton $\mathcal{A}(\varphi)$ and an acceptance condition $\mathcal{C} := \bigvee_{I \subseteq \mathbf{G}_\varphi \cup \mathbf{F}_\varphi} \mathcal{P}_I$. Every word $w : \mathbb{N} \to 2^{Ap}$ induces a run $\rho = \mathcal{A}(\varphi)(w) : \mathbb{N} \to Q$ starting in i and following δ. The run is thus accepting and the word is accepted if the set of states visited infinitely often $\mathrm{Inf}(\rho)$ is Muller accepting for φ. Vice versa, a run $\rho = i(\chi_1, \alpha_1)(\chi_2, \alpha_2)\cdots$ induces a word $Ap(\rho) = \alpha_1\alpha_2\cdots$. We now prove that this acceptance condition is sound and complete.

Theorem 3. *Let φ be a formula and w a word. Then w is accepted by the deterministic automaton $\mathcal{A}(\varphi)$ with the generalilzed Rabin pairs condition $\mathcal{C}(\varphi)$ if and only if $w \models \varphi$.*

The second component of the state space takes care of identifying which recurrent formulae hold infinitely often or eventually always. For a word w, let $I(w) = \{\psi \in \mathbf{G}_\varphi \mid w \models \mathbf{F}\psi\} \cup \{\psi \in \mathbf{F}_\varphi \mid w \models \mathbf{G}\psi\}$

Lemma 4 (Correctness of $\mathcal{B}(\xi)$'s). *For every word w and every $\xi \in \mathcal{R}\mathrm{ec}$,*

1. $w \models \mathbf{GF}\xi$ *iff* $\overset{\infty}{\exists}\, i \in \mathbb{N} : \exists \chi \in \mathcal{B}(\xi)(w)[i] \cap F_\xi : X^* I(w) \models \chi$,
2. $w \models \mathbf{FG}\xi$ *iff* $\overset{\infty}{\forall}\, i \in \mathbb{N} : \forall \chi \in \mathcal{B}(\xi)(w)[i] \cap F_\xi : X^* I(w) \models \chi$.

Proof. Since for every n, $w \models \mathbf{GF}\chi$ iff $I \models \mathbf{GFX}^n\chi$, and similarly $w \models \mathbf{FG}\chi$ iff $I \models \mathbf{FGX}^n\chi$, the lemma follows from the correctness of unfolding and $\mathcal{B}(\xi)$ having the initial state self-loop as the only cycle. ◻

The first component of the state space takes care of all progress or failure in finite time.

Lemma 5 (Local (finitary) correctness of $\mathcal{A}(\varphi)$). *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 $w \models \varphi$ if and only if $w_n \models \chi_n$.*

Proof. The one-step unfold produces a temporally equivalent (w.r.t. LTL satisfaction) formula. The unfold is a Boolean function over atomic propositions and elements of $\mathbf{X}\mathrm{sf}(\varphi)$. Therefore, this unfold is satisfied if and only if the next state satisfies $\mathbf{X}^{-1}(\psi)$ where ψ is the result of partial application of the Boolean function to the currently read letter of the word. We conclude by induction. ◻

Further, each occurrence of satisfaction of $\mathbf{F}$ must happen in finite time. As a consequence, a run with $\chi_i \not\equiv \mathbf{ff}$ is rejecting if and only if satisfaction of some $\mathbf{F}\psi$ is always postponed.

Proposition 6 (Completeness). *If $w \models \varphi$ then $\mathrm{Inf}(\mathcal{A}(\varphi)(w))$ is accepting w.r.t. $\mathcal{C}(\varphi)$.*

Proof. Let us show that the pair $\mathcal{P}_{I(w)}$ is satisfied.

Firstly, we show *avoid* is visited only finitely often, i.e. the first component ψ is almost always (in states of $\mathrm{Inf}(\mathcal{A}(\varphi)(w))$) entailed by $X^* I(w)$ and the current states of $\mathcal{B}(\xi)$ for each $\mathbf{G}\xi \in I(w)$. Consider some sufficiently large i (for which $I(w)$ holds) and the corresponding w_i and the current state $s_i = \langle \chi_i, (R_\xi)_{\xi \in \mathcal{R}\mathrm{ec}}$. By Lemma 5 we have $w_i \models \chi_i$. Notice that χ_i is a Boolean combination of **XF**-, **XU**- and **XG**-formulae and formulae produced by their unfolding. Whenever $\mathbf{F}\psi$ is satisfied whenever entering s_i, it is in $I(w)$ and since in ψ_i we have a *disjunction* of $\mathbf{XF}\psi$ and the rest of the unfold, the entailment of this rest is irrelevant as the disjunction is entailed directly by $\mathbf{X}^* I(w)$. Similarly, if $\psi_1 \mathbf{U} \psi_2$ holds, the unfold (again a disjunction) is entailed since eventually ψ_2 holds and we proceed by induction. Finally, if $\mathbf{G}\psi$ holds we need to show entailment of its unfolds. This is a conjunction of $\mathbf{XG}\psi$ and the unfolds of ψ and their successors. The former is entailed by $\mathbf{X}^* I(w)$, the latter are the elements of R_ξ (with **F**'s and **G**'s unfolded), which are thus entailed by R_ψ (and the unfolded **F**'s and **G**'s are entailed recursively by the same argumentation).

Secondly, $avoid_{\mathbf{G}\chi}$ is visited only finitely often for each $\mathbf{G}\chi \in I(w)$. Indeed, since $w \models \mathbf{FG}\chi$ almost all $w_i \models \chi$. Thus almost all tokens generated in $\mathcal{B}(\xi)$ end up in a final state that holds in the current position. Since there are only finitely many of those and they are elements of $\mathsf{B}^+(\mathbf{G}_\varphi \cup \mathbf{F}_\varphi)$, they are entailed by $\mathbf{X}^* I(w)$ due to Lemma 4.

Thirdly, similarly $reach_{\mathbf{F}\chi}$ is visited infinitely often for each $\mathbf{F}\chi \in I(w)$. Indeed, since $w \models \mathbf{GF}\chi$ infinitely many $w_i \models \chi$. Thus infinitely many tokens generated in $\mathcal{B}(\xi)$ end up in a final state that holds in the current position. Since there are only finitely many of those and they are elements of $\mathsf{B}^+(\mathbf{G}_\varphi \cup \mathbf{F}_\varphi)$, they are entailed by $\mathbf{X}^* I(w)$ due to Lemma 4. □

Proposition 7 (Soundness). *If* $\mathrm{Inf}(\mathcal{A}(\varphi)(w))$ *is accepting w.r.t.* $\mathcal{C}(\varphi)$ *then* $w \models \varphi$.

Proof. Let $M := \mathrm{Inf}(\mathcal{A}(\varphi)(w))$ be a accepting for pair $\mathcal{P}_I$. There is $i \in \mathbb{N}$ such that after reading i letters we come to $\mathrm{Inf}(\mathcal{A}(\varphi)(w))$ and stay there from now on and, moreover, $w_i \models \psi$ for all $\psi \in I$ by Lemma 4 and definition of $\mathcal{C}$. Denote the ith state by $\langle \psi, \mathcal{R} \rangle$. By the definition of *avoid*, we get $w_i \models \psi$. By Lemma 5, we thus get $w \models \varphi$. □

5 Optimizations

We optimize the construction as follows. Instead of keeping track of states of each $\mathcal{B}(\xi)$, only the currently relevant ones. E.g. after reading $\emptyset$ in $\mathbf{GF}a \vee (b \wedge \mathbf{GF}c)$, it is no more interesting to track if c occurs infinitely often. Formally, define $\mathcal{R}\mathrm{elRec}(\psi)(\xi)$ to be true iff ξ occurs in the Boolean combination ψ. When the first component of a state is ψ, the second component is restricted to the vector with coordinates in $\mathcal{R}\mathrm{elRec}(\psi)$. The same holds for the definition of *avoid*.

Further, since only the infinite behaviour of $\mathcal{B}(\xi)$ is important and it has acyclic structure (except for the initial states), instead of the initial state we can start in any subset of states. Therefore, we start in a subset that is most likely to occur repetitively and we thus omit unnecessary initial transient parts of $\mathcal{A}(\varphi)$.

Pseudocode

A Notation:

For LTL formula φ, $\mathrm{sf}(\varphi)$ denotes the set of all subformulae (any Boolean combination is one formula). Further, we denote by $\mathbb{T}(\varphi)$ the set of all **X**-, **F**-, **G**- and **U**-subformulae of φ. For a set S, $\mathsf{B}^+(S)$ is the set of positive Boolean functions over S. The *closure* of φ is then $\mathbb{C}(\varphi) := \{\mathbf{tt}, \mathbf{ff}\} \cup Ap \cup \{\neg a \mid a \in Ap\} \cup \mathbb{T}(\varphi) \cup \mathbf{X}\mathbb{T}(\varphi)$ where $\mathbf{X}S = \{\mathbf{X}s \mid s \in S\}$ and further $\mathbf{X}^*S = \{\underbrace{\mathbf{X}\cdots\mathbf{X}}_{n} s \mid s \in S, n \in \mathbb{N}_0\}$.

B Main algorithm:

input: $\varphi \in LTL$

1. if φ not in LTL_{-GU} then return "not in the LTL fragment"
2. compute type 2 formulae:
   $\mathbf{G}_\varphi := \{\mathbf{G}\psi \in \mathrm{sf}(\varphi)\}$
   $\mathbf{F}_\varphi := \{\mathbf{F}\psi \in \mathrm{sf}(\omega) \mid \text{for some } \omega \in \mathbf{G}_\omega\}$
   $\mathcal{R}\mathrm{ec} := \{\psi \mid \mathbf{G}\psi \in \mathbf{G}_\varphi \text{ or } \mathbf{F}\psi \in \mathbf{F}_\varphi\}$
   /*go down the tree and take every child of **G**, and further every child of **F** if you already saw **G** on this branch*/
   /* we can take progress formulae only */
3. foreach $\xi \in \mathcal{R}\mathrm{ec}$ construct $\mathcal{B}(\xi)$
4. construct $\mathcal{A}(\varphi)$
5. construct GR acceptance condition $\mathcal{C}$
6. if $\mathcal{C}$ empty then return "unsat"
7. else
8. ouput $\mathcal{A}(\varphi), \mathcal{C}$
9. perform Andreas' degeneralization and ouput its result

C Auxiliary functions:

$\mathfrak{Unf} : \mathsf{B}^+(\mathbb{C}(\varphi)) \to \mathsf{B}^+(\mathbb{C}(\varphi))$:

$$
\begin{aligned}
\mathfrak{Unf}(a) &= a \\
\mathfrak{Unf}(\neg a) &= \neg a \\
\mathfrak{Unf}(\varphi \wedge \psi) &= \mathfrak{Unf}(\varphi) \wedge \mathfrak{Unf}(\psi) \\
\mathfrak{Unf}(\varphi \vee \psi) &= \mathfrak{Unf}(\varphi) \vee \mathfrak{Unf}(\psi) \\
\mathfrak{Unf}(\mathbf{X}\varphi) &= \mathbf{X}\varphi \\
\mathfrak{Unf}(\mathbf{F}\varphi) &= \mathfrak{Unf}(\varphi) \vee \mathbf{XF}\varphi \\
\mathfrak{Unf}(\mathbf{G}\varphi) &= \mathfrak{Unf}(\varphi) \wedge \mathbf{XG}\varphi \\
\mathfrak{Unf}(\varphi\mathbf{U}\psi) &= \mathfrak{Unf}(\psi) \vee (\mathfrak{Unf}(\varphi) \wedge \mathbf{X}(\varphi\mathbf{U}\psi))
\end{aligned}
$$

$\mathbf{X}^{-1} : \mathsf{B}^{+}(\mathbb{C}(\varphi)) \to \mathsf{B}^{+}(\mathbb{C}(\varphi))$:

$$\begin{aligned}
\mathbf{X}^{-1}(\psi_1 \wedge \psi_2) &= \mathbf{X}^{-1}(\psi_1) \wedge \mathbf{X}^{-1}(\psi_2)\\
\mathbf{X}^{-1}(\psi_1 \vee \psi_2) &= \mathbf{X}^{-1}(\psi_1) \vee \mathbf{X}^{-1}(\psi_2)\\
\mathbf{X}^{-1}(\mathbf{X}\psi) &= \psi\\
\mathbf{X}^{-1}(\psi) &= \psi \text{ for all other types of formulae}
\end{aligned}$$

$(\cdot)[\nu] : \mathsf{B}^{+}(\mathbb{C}(\varphi)) \to \mathsf{B}^{+}(\mathbb{C}(\varphi))$ for $\nu \subseteq Ap$:

Consider a formula $\chi \in \mathsf{B}^{+}(\mathbb{C}(\varphi))$. For a set $S \subseteq \mathbb{C}(\varphi)$, let $\chi[S \mapsto \mathbf{tt}]$ denote the formula where $\mathbf{tt}$ is substituted for elements of S. As elements of $\mathbb{C}(\varphi)$ are considered to be atomic expressions here, the substitution is only done on the propositional level and does not go through the modality, e.g. $(a \wedge \mathbf{XG}a)[\{a\} \to \mathbf{tt}] = \mathbf{tt} \wedge \mathbf{XG}a$, which is equivalent to $\mathbf{XG}a$ in the propositional semantics. For a valuation $\nu \subseteq Ap$, we set $\chi[\nu] := \chi[\nu \cup \{\neg a \mid a \in Ap \setminus \nu\} \mapsto \mathbf{tt}]$.

$\mathcal{R}\mathrm{elRec} : \mathsf{B}^{+}(\mathbb{C}(\varphi)) \to 2^{\mathcal{R}\mathrm{ec}}$:

$\mathcal{R}\mathrm{elRec}(\psi)(\xi)$ iff ξ occurs in the Boolean combination ψ.

D Automaton $\mathcal{B}(\xi)$ construction:

input: $\xi \in \mathcal{R}\mathrm{ec}$

output: $\mathcal{B}(\xi) = (Q_\xi, i_\xi, \delta_\xi, F_\xi)$ over 2^{Ap}

1. the initial state $i_\xi := \xi$
2. $\mathit{worklist} := \{i_\xi\}$
3. while $\mathit{worklist} \neq \emptyset$
   (a) pop $q \in \mathit{worklist}$
   (b) if $q \notin F_\xi$ then foreach $\nu \subseteq Ap$
       $\mathit{new} := \mathbf{X}^{-1}(\chi[\nu])$
       add (q, ν, new) to δ_ξ
       if $\mathit{new} \notin Q$ then add new to $\mathit{worklist}$ and Q_ξ
       if (each atomic proposition has $\mathbf{F}$ or $\mathbf{G}$ as an ancestor in the syntactic tree of new) then add q to F_ξ /*i.e. no atomic propositions are guarded by only $\mathbf{X}^*$ and Boolean operators*/
4. foreach $\nu \subseteq Ap$ add (i, ν, i) to δ_ξ

E Automaton $\mathcal{A}(\varphi)$ construction:

output: $\mathcal{A}(\varphi) = (Q, i, \delta)$ over $\Sigma = 2^{Ap}$

1. for each $\mathbf{G}\xi \in \mathbf{G}_\varphi$,
   pick f_ξ to be (1) $\mathbf{tt}$ if $\mathbf{tt} \in F_\xi$ else (2) any $\psi \neq \mathbf{ff}$ if $\psi \in F_\xi$ else (3) $\mathbf{ff}$
2. for each $\mathbf{F}\xi \in \mathbf{F}_\varphi$,
   pick f_ξ to be (1) $\mathbf{ff}$ if $\mathbf{ff} \in F_\xi$ else (2) any $\psi \neq \mathbf{tt}$ if $\psi \in F_\xi$ else (3) $\mathbf{tt}$
3. for each $\xi \in \mathcal{R}\mathrm{ec}$,
   $S_\xi :=$ states on an arbitrary path from i_ξ to f_ξ including both
4. the initial state $i := \langle \mathfrak{U}\mathrm{nf}(\varphi), (S_\xi)_{\xi \in \mathcal{R}\mathrm{ec}} \rangle$
5. $\mathit{worklist} := \{i\}$
6. while $\mathit{worklist} \neq \emptyset$
   (a) pop $q := \langle \psi, (R_\xi)_{\xi \in \mathcal{R}\mathrm{elRec}(\psi)} \rangle \in \mathit{worklist}$
   (b) foreach $\nu \subseteq Ap$
      i. $\mathit{new}\psi := \mathfrak{U}\mathrm{nf}(\mathbf{X}^{-1}(\psi[\nu]))$ /*and search for the label */
      ii. foreach $\xi \in \mathcal{R}\mathrm{elRec}(\psi)$ do $\mathit{new}_\xi := \bigcup_{q \in R_\xi} \delta_\xi(q, \nu)$
      iii. $\mathit{new} := \langle \mathit{new}\psi, (\mathit{new}_\xi)_{\xi \in \mathcal{R}\mathrm{elRec}(\mathit{new}\psi)} \rangle$ /*new vector can be smaller*/
      iv. add (q, ν, new) to δ
      v. if $\mathit{new} \notin Q$ then add new to $\mathit{worklist}$ and Q

F GR acceptance condition $\mathcal{C}$ construction:

1. foreach $I \subseteq \mathbf{G}_\varphi \cup \mathbf{F}_\varphi$
   (a) $\mathit{avoid} := \{\langle \psi, (R_\xi)_{\xi \in \mathcal{R}\mathrm{elRec}(\psi)} \rangle \in Q \mid \mathbf{X}^* I \cup \bigcup_{\substack{\xi \in I: \\ \mathbf{G}\xi \in \mathrm{sf}(\varphi)}} R_\xi \not\models \psi\}\}$
   (b) foreach $\mathbf{G}\chi \in I$

$$\mathit{avoid}_{\mathbf{G}\chi} := \{\langle \psi, (R_\xi)_{\xi \in \mathcal{R}\mathrm{elRec}(\psi)} \rangle \in Q \mid \exists q \in R_\chi \cap F_\chi : \mathbf{X}^* I \not\models q\}\}$$

   /* e.g. states where the segment automaton for χ is in $\mathbf{ff}$ */
   (c) foreach $\mathbf{F}\chi \in I$

$$\mathit{reach}_{\mathbf{F}\chi} := \{\langle \psi, (R_\xi)_{\xi \in \mathcal{R}\mathrm{elRec}(\psi)} \rangle \in Q \mid \exists q \in R_\chi \cap F_\chi : \mathbf{X}^* I \models q\}\}$$

   /* e.g. states where the segment automaton for χ is in $\mathbf{tt}$ or in some element of I */
   (d) add $\left(\mathit{avoid} \cup \bigcup_{\mathbf{G}\chi \in I} \mathit{avoid}_{\mathbf{G}\chi}, \{\mathit{reach}_{\mathbf{F}\chi} \mid \mathbf{F}\chi \in I\}\right)$ to $\mathcal{C}$
2. perform redundancy removals on $\mathcal{C}$ described on the webpage of Rabinizer (version 1) and return the result
   /* such as $(\{1,2\},\{\{2\},\{3\}\})$ is redundant w.r.t. $(\{1\},\{\{2,3\}\})$ */

References

[CGK13] Krishnendu Chatterjee, Andreas Gaiser, and Jan Křetínský. Automata with generalized Rabin pairs for probabilistic model checking and LTL synthesis. In *CAV*, 2013. To appear, accessible at http://arxiv.org/abs/1304.5281.

[GKE12] Andreas Gaiser, Jan Křetínský, and Javier Esparza. Rabinizer: Small deterministic automata for LTL(F,G). In *ATVA*, pages 72–76, 2012.

[KB06] Joachim Klein and Christel Baier. Experiments with deterministic *omega*-automata for formulas of linear temporal logic. *Theor. Comput. Sci.*, 363(2):182–195, 2006.

[KE12] Jan Křetínský and Javier Esparza. Deterministic automata for the (F,G)-fragment of LTL. In *CAV*, pages 7–22, 2012.