Fairness and promptness in initialized systems

Youssouf Oualhadj[1] Léo Tible[1] Daniele Varacca[1]

September 12, 2022

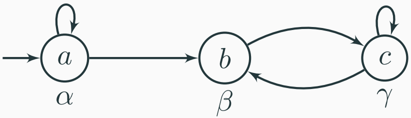
[1]Laboratoire d'Algorithmique, Complexité et Logique, UPEC

Model Checking problem

Model (of a system) $\models$ Specification (good behaviors)

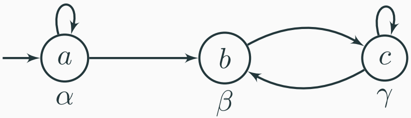
Model and specification

Kripke structure



Model and specification

Kripke structure

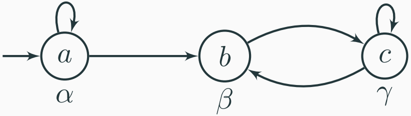


Linear Temporal Logic
LTL is the set formulas ϕ defined using the following grammar:

$$\phi ::= \alpha \mid \neg\phi \mid \phi \vee \phi \mid \mathsf{X}\ \phi \mid \phi\ \mathsf{U}\ \phi\ .$$

Model and specification

Kripke structure



A specification

$$\phi = \mathsf{F}\beta \text{ (Reachability)}$$

$$\phi = \mathsf{G}\neg\gamma \text{ (Safety)}$$

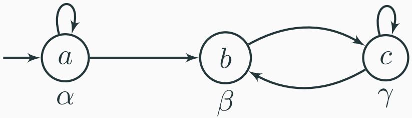
$$\phi = \mathsf{GF}\beta = \mathsf{F}^{\infty}\beta \text{ (Büchi)}$$

Problem

Model checking
Given an LTS $\mathcal{L}$ and an LTL formula ϕ, the *universal model checking problem* consists in checking whether $\forall \rho \in \mathsf{Runs}(\mathcal{L}),\ \rho \models \phi$.

Then, we write $\mathcal{L} \models \phi$.

Example

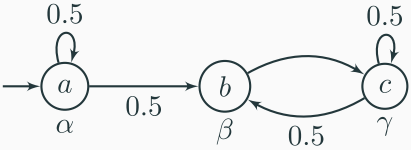


Here, $\mathcal{L} \models \alpha$ and $\mathcal{L} \not\models \mathsf{F}^{\infty}\gamma$

Fairness

Intuition
If something *can* happen infinity often, it *should* happen infinity often.

Fairness : Markov chain



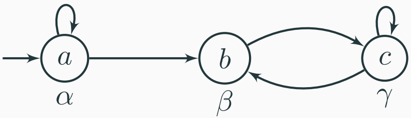
Probability mesure : $\mathbb{P}_{\mathcal{L}}$

Problem

Fair model checking
Given an LTS $\mathcal{L}$ and an LTL formula ϕ, the *fair model checking problem* consists in checking whether $\mathbb{P}_{\mathcal{L}}(\{\rho \in \mathsf{Runs}(\mathcal{L}) \mid \rho \models \phi\}) = 1$.

Then we write $\mathcal{L} \models_{\mathsf{AS}} \phi$.

Example



Here, $\mathcal{L} \models_{\mathsf{AS}} \mathsf{F}^{\infty}\gamma$

Complexity of Model Checking problems

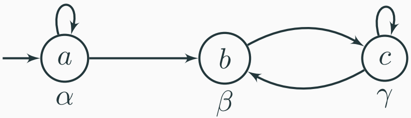
Model Checking of LTL [SC85]
Both the universal and fair model checking problem for LTL are PSPACE complete.

Goal
To look at fragment of LTL.

Müller Formulas

Müller formulas $\mathsf{L}(\mathsf{F}^\infty)$
The set of Müller formulas is a fragment of LTL where the only temporal operator used is F^∞.

Example

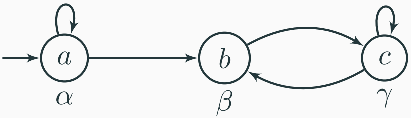


$$\mathsf{F}^\infty\alpha,\ \mathsf{F}^\infty(\mathsf{F}^\infty\beta \vee \gamma), \ldots$$

Müller Formulas

Müller formulas $\mathsf{L}(\mathsf{F}^\infty)$
The set of Müller formulas is a fragment of LTL where the only temporal operator used is F^∞.

Example



$$\mathsf{F}^\infty\alpha,\ \mathsf{F}^\infty(\mathsf{F}^\infty\beta \vee \gamma), \ldots$$

Müller formulas [SVV07]

- The universal model checking of $\mathsf{L}(\mathsf{F}^\infty)$ is coNP complete.
- The fair model checking of $\mathsf{L}(\mathsf{F}^\infty)$ is linear in $|\mathcal{L}|$ and $|\phi|$.

Proof ideas

Nice properties
The satisfaction of $\phi \in \mathsf{L}(\mathsf{F}^\infty)$ only depends on the set of states visited infinitly often ($\Rightarrow$ prefix independence).

The set of states visited infinitly often is a Strongly Connected Set (SCS).

Proof ideas

Nice properties
The satisfaction of $\phi \in \mathsf{L}(\mathsf{F}^\infty)$ only depends on the set of states visited infinitly often ($\Rightarrow$ prefix independence).

The set of states visited infinitly often is a Strongly Connected Set (SCS).

Universal model checking
Idea : guess a "faulty" SCS, and check it is faulty : coNP.

Proof ideas

Nice properties
The satisfaction of $\phi \in \mathsf{L}(\mathsf{F}^\infty)$ only depends on the set of states visited infinitly often ($\Rightarrow$ prefix independence).

The set of states visited infinitly often is a Strongly Connected Set (SCS).

Universal model checking
Idea : guess a "faulty" SCS, and check it is faulty : coNP.

Fair model checking
Key : almost surely a run ends in a Bottom Strongly Connected Componnent (BSCC).

Idea : check each BSCC by structural induction over the formula.

Prompt Linear Temporal Logic
pLTL is LTL with the added operator $\mathsf{F_P}$.

$\mathsf{F_P}$: given k, $(\rho, k) \models \mathsf{F_P}\phi$ iff $\exists i \leq k, (\rho[i..], k) \models \phi$ ("finally with bounded horizon").

Prompt LTL

Prompt Linear Temporal Logic
$pLTL$ is LTL with the added operator $\mathsf{F_P}$.

$\mathsf{F_P}$: given k, $(\rho, k) \models \mathsf{F_P}\phi$ iff $\exists i \leq k, (\rho[i..], k) \models \phi$ ("finally with bounded horizon").

Model checking
Universal : $\exists k, \forall \rho \in \mathsf{Runs}(\mathcal{L}),\ (\rho, k) \models \phi$

Fair : $\exists k, \mathbb{P}_{\mathcal{L}}(\{\rho \in \mathsf{Runs}(\mathcal{L}) \mid (\rho, k) \models \phi\}) = 1$

Prompt LTL

Prompt Linear Temporal Logic
$pLTL$ is LTL with the added operator $\mathsf{F_P}$.

$\mathsf{F_P}$: given k, $(\rho, k) \models \mathsf{F_P}\phi$ iff $\exists i \leq k, (\rho[i..], k) \models \phi$ ("finally with bounded horizon").

Model checking
Universal : $\exists k, \forall \rho \in \mathsf{Runs}(\mathcal{L}),\ (\rho, k) \models \phi$

Fair : $\exists k, \mathbb{P}_{\mathcal{L}}(\{\rho \in \mathsf{Runs}(\mathcal{L}) \mid (\rho, k) \models \phi\}) = 1$

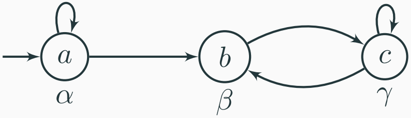
Model Checking of pLTL [KPV09]
Both the universal and fair model checking problem for pLTL are PSPACE complete.

Prompt Müller

Prompt Müller fragment $\mathsf{L}(\mathsf{F}^{\infty}_{\mathsf{P}})$
The set of prompt Müller formulas is a fragment of pLTL where the only temporal operator is $\mathsf{F}^{\infty}_{\mathsf{P}}$.

$\mathsf{F}^{\infty}_{\mathsf{P}}\phi$: There is a bound k such that ϕ is true somewhere in each "window" of size k.

Example



$$\phi = \mathsf{F}^{\infty}_{\mathsf{P}}\gamma$$

$$(aaabcbc^{\omega}, 5) \models \phi$$

$$(aaaaabcbc^{\omega}, 5) \not\models \phi$$

Our results of Model Checking

New Theorem

- The universal model checking of $\mathsf{L}(\mathsf{F}^{\infty}_{\mathsf{P}})$ is coNP complete.

Our results of Model Checking

New Theorem

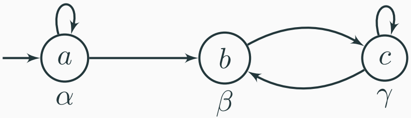
- The universal model checking of $\mathsf{L}(\mathsf{F}^{\infty}_{\mathsf{P}})$ is coNP complete.
- The fair model checking of $\mathsf{L}(\mathsf{F}^{\infty}_{\mathsf{P}})$ is coNP complete.

Idea : the issue is that the fragment is not prefix independent.

Prefix independence

Prefix independent prompt Müller fragment $\mathsf{F}(\mathsf{L}^{+}(\mathsf{F}^{\infty}_{\mathsf{P}}))$
A formula ϕ is in $\mathsf{F}(\mathsf{L}^{+}(\mathsf{F}^{\infty}_{\mathsf{P}}))$ iff there is $\psi \in \mathsf{L}^{+}(\mathsf{F}^{\infty}_{\mathsf{P}})$ such that $\phi = \mathsf{F}\psi$.

Example



$$\psi = \mathsf{F}^{\infty}_{\mathsf{P}}\gamma$$

$\mathcal{L} \not\models \psi$ $\qquad$ $\mathcal{L} \not\models \mathsf{F}\psi$

$\mathcal{L} \not\models_{\mathsf{AS}} \psi$ $\qquad$ $\mathcal{L} \models_{\mathsf{AS}} \mathsf{F}\psi$

Our results of Model Checking

New Theorem

- The universal model checking of $\mathsf{F}(\mathsf{L}^{+}(\mathsf{F}_{\mathsf{P}}^{\infty}))$ is coNP complete.

Our results of Model Checking

New Theorem

- The universal model checking of $\mathsf{F}(\mathsf{L}^+(\mathsf{F}_\mathsf{P}^\infty))$ is coNP complete.
- The fair model checking of $\mathsf{F}(\mathsf{L}^+(\mathsf{F}_\mathsf{P}^\infty))$ is linear in $|\phi|$ and quadratic in $|\mathsf{S}|$.

Conclusion

Model check.	LTL	$\mathsf{L}(\mathsf{F}^\infty)$	pLTL	$\mathsf{L}(\mathsf{F}^\infty_\mathsf{P})$	$\mathsf{F}(\mathsf{L}^+(\mathsf{F}^\infty_\mathsf{P}))$
Universal	PSPACE-c	coNP-c	PSPACE-c	coNP-c	coNP-c
Fair	PSPACE-c	Linear	PSPACE-c	coNP-c	Linear

Conclusion

Model check.	LTL	$L(F^\infty)$	pLTL	$L(F_P^\infty)$	$F(L^+(F_P^\infty))$
Universal	PSPACE-c	coNP-c	PSPACE-c	coNP-c	coNP-c
Fair	PSPACE-c	Linear	PSPACE-c	coNP-c	Linear

Synthesis	LTL	$L(F^\infty)$	pLTL	$L(F_P^\infty)$	$F(L^+(F_P^\infty))$
Universal	2EXP-c	PSPACE-c	2EXP-c	?	?
Fair	2EXP-c	NP-c	2EXP-c	?	?

Conclusion

Model check.	LTL	$L(F^\infty)$	pLTL	$L(F_P^\infty)$	$F(L^+(F_P^\infty))$
Universal	PSPACE-c	coNP-c	PSPACE-c	coNP-c	coNP-c
Fair	PSPACE-c	Linear	PSPACE-c	coNP-c	Linear

Synthesis	LTL	$L(F^\infty)$	pLTL	$L(F_P^\infty)$	$F(L^+(F_P^\infty))$
Universal	2EXP-c	PSPACE-c	2EXP-c	?	?
Fair	2EXP-c	NP-c	2EXP-c	?	?

Thanks !

References

[KPV09] Orna Kupferman, Nir Piterman, and Moshe Vardi. From liveness to promptness. *Formal Methods in System Design*, 34, 04 2009.

[PR89] A. Pnueli and Roni Rosner. On the synthesis of a reactive module. *Automata Languages and Programming*, 372:179–190, 01 1989.

[SC85] A. Sistla and Edmund Clarke. The complexity of propositional linear temporal logics. *J. ACM*, 32:733–749, 07 1985.

[SVV07] Matthias Schmalz, Hagen Völzer, and Daniele Varacca. Model checking almost all paths can be less expensive than checking all paths. volume 4855, pages 532–543, 12 2007.

[VV12] Hagen Völzer and Daniele Varacca. Defining fairness in reactive and concurrent systems. *J. ACM*, 59(3):13:1–13:37, 2012.