Analyse de sûreté de systèmes complexes

Quentin Peyras

EPITA

24 avril 2023

Model-checking de LTL

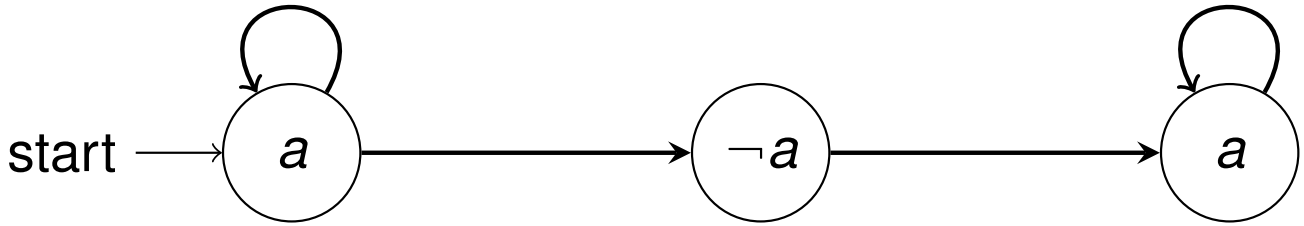
Problème du model-checking

Entrée : $\mathcal{S}$ une structure de Kripke et ϕ une formule LTL
Problème : Est-ce que $\pi \models \phi$ pour toute trace π de $\mathcal{S}$?

Un exemple problématique

Contre-exemple

On veut vérifier si $\mathbf{F}\,\mathbf{G}a$ est vérifiée par la structure suivante.



Conclusion

- On ne peut pas raisonner directement sur la structure de Kripke,
- On doit pouvoir représenter un ensemble de trace LTL sous une forme de système de transition.

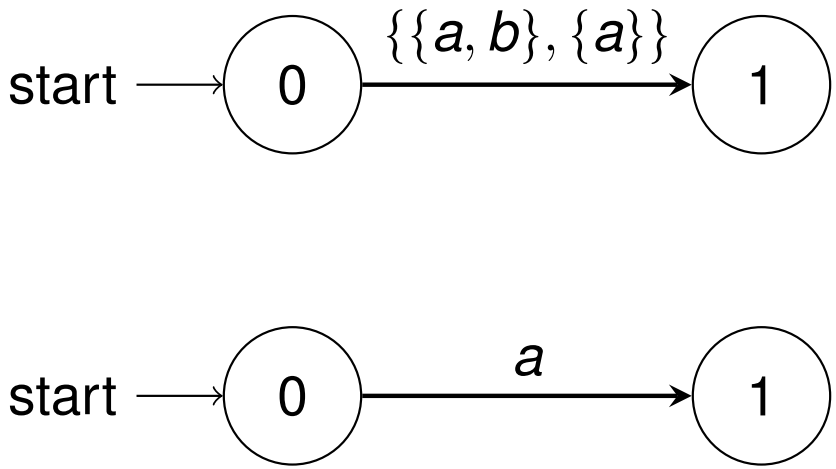
Automate de Büchi

Définition

Un **automate de Büchi** est un tuple $(\Sigma, Q, I, F, \delta, \rho)$ où :

- Σ est un alphabet (souvent, $\Sigma = 2^{\mathbf{AP}}$),
- Q est un ensemble d'état,
- I est l'ensemble des états initiaux,
- F est l'ensemble des états acceptants,
- $\delta \subseteq Q \times Q$ est l'ensemble des transitions,
- $\rho : \delta \to 2^{\Sigma}$ est une fonction qui associe à chaque transition l'ensemble des variables qui sont vérifiées par cette transition.

Automate de Büchi



Automate de Büchi

Définition

Un **automate de Büchi** est un tuple $(\Sigma, Q, I, F, \delta, \rho)$ où :

- Σ est un alphabet (souvent, $\Sigma = 2^{\mathbf{AP}}$),
- Q est un ensemble d'état,
- I est l'ensemble des états initiaux,
- F est l'ensemble des états acceptants,
- $\delta \subseteq Q \times Q$ est l'ensemble des transitions,
- $\rho : \delta \to 2^{\Sigma}$ est une fonction qui associe à chaque transition l'ensemble des variables qui sont vérifiées par cette transition.

Chemin acceptant

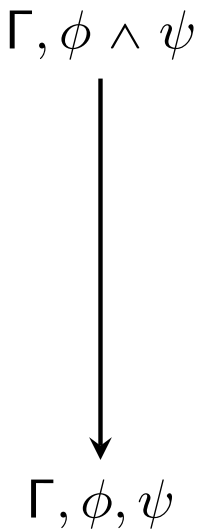
Un chemin acceptant d'un automate de Büchi est une suite **infinie** d'états reliés par des transitions et passant infiniment souvent par un état acceptant.

Forme normale négative

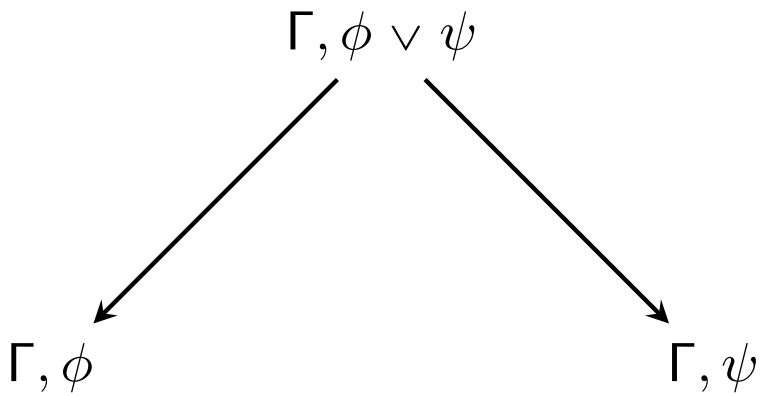
Règles de transformations

- $\neg\neg p \rightarrow p$,
- $\neg(\phi_1 \vee \phi_2) \rightarrow \neg\phi_1 \wedge \neg\phi_2$,
- $\neg(\phi_1 \wedge \phi_2) \rightarrow \neg\phi_1 \vee \neg\phi_2$,
- $\neg \mathbf{X}\,\phi \rightarrow \mathbf{X}\,\neg\phi$,
- $\neg\mathbf{G}\phi \rightarrow \mathbf{F}\,\neg\phi$
- $\neg\,\mathbf{F}\,\phi \rightarrow \mathbf{G}\neg\phi$

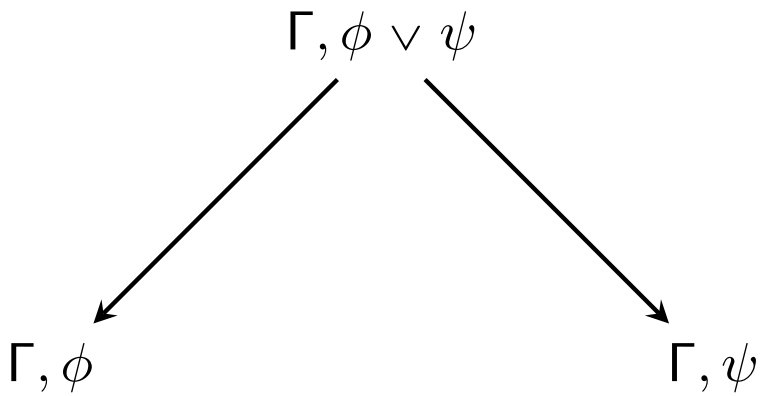
Tableau



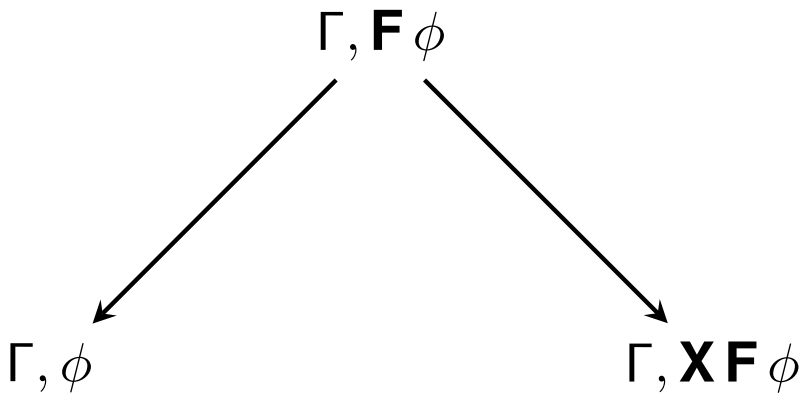
Tableau



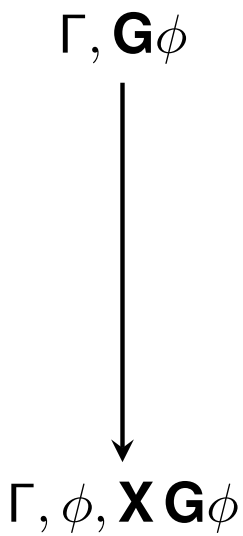
Tableau



Tableau



Tableau



Exemple

$$\mathbf{F}\,\mathbf{G}a \wedge \mathbf{G}b$$

Exemple

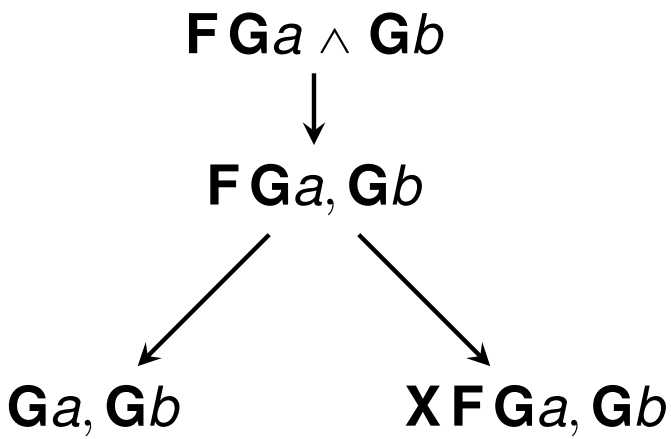
$$\mathbf{F}\,\mathbf{G}a \wedge \mathbf{G}b$$

$$\downarrow$$

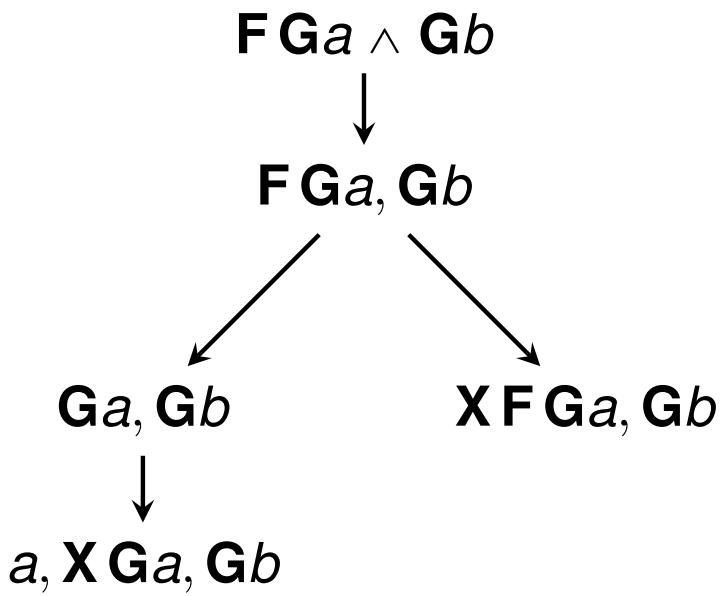
$$\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$$

$$\mathbf{X}\,\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$$

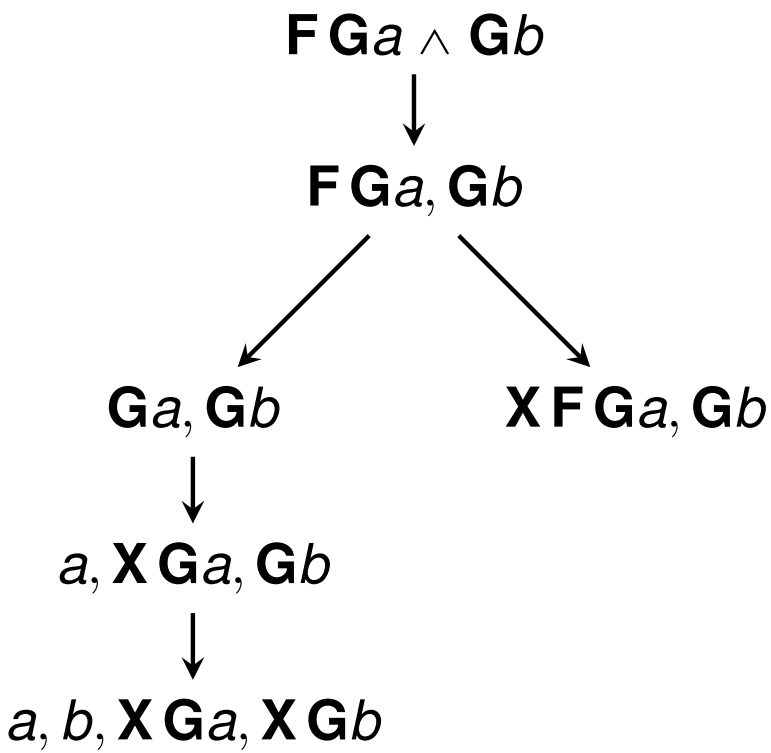
Exemple


$\mathbf{F}\,\mathbf{G}a \wedge \mathbf{G}b$
$\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$
$\mathbf{G}a, \mathbf{G}b$
$\mathbf{X}\,\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$

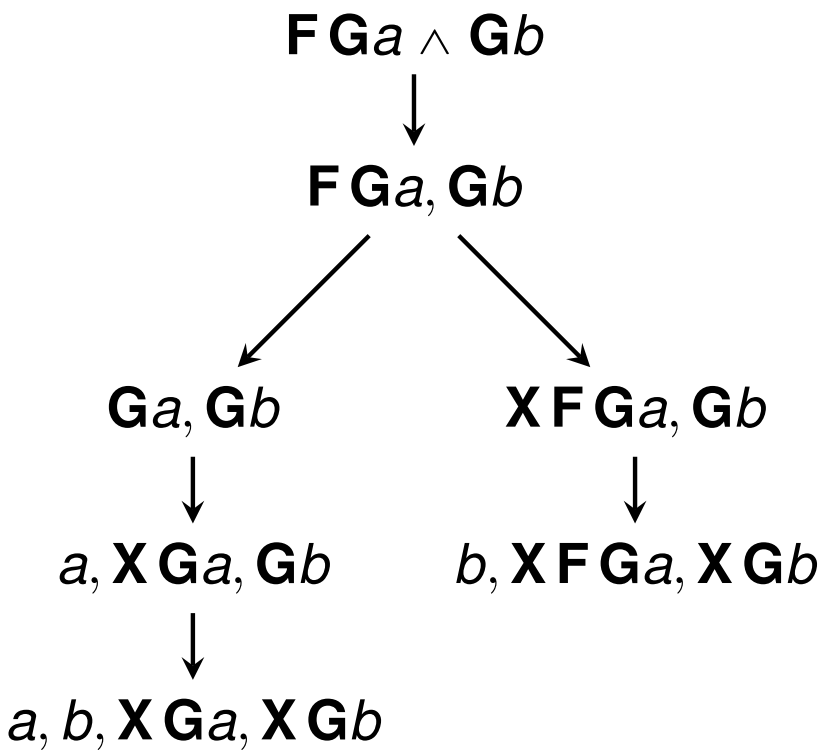
Exemple


$\mathbf{F}\,\mathbf{G}a \wedge \mathbf{G}b$
$\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$
$\mathbf{G}a, \mathbf{G}b$
$\mathbf{X}\,\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$
$a, \mathbf{X}\,\mathbf{G}a, \mathbf{G}b$

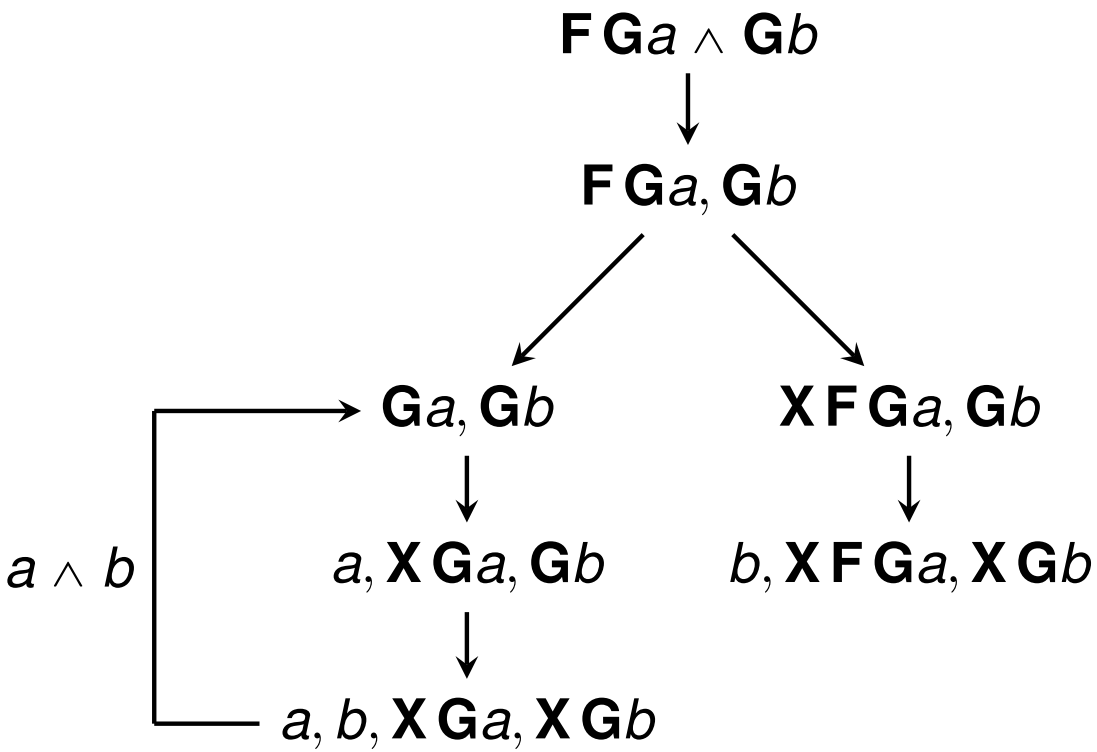
Exemple


$\mathbf{F}\,\mathbf{G}a \wedge \mathbf{G}b$
$\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$
$\mathbf{G}a, \mathbf{G}b$
$\mathbf{X}\,\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$
$a, \mathbf{X}\,\mathbf{G}a, \mathbf{G}b$
$a, b, \mathbf{X}\,\mathbf{G}a, \mathbf{X}\,\mathbf{G}b$

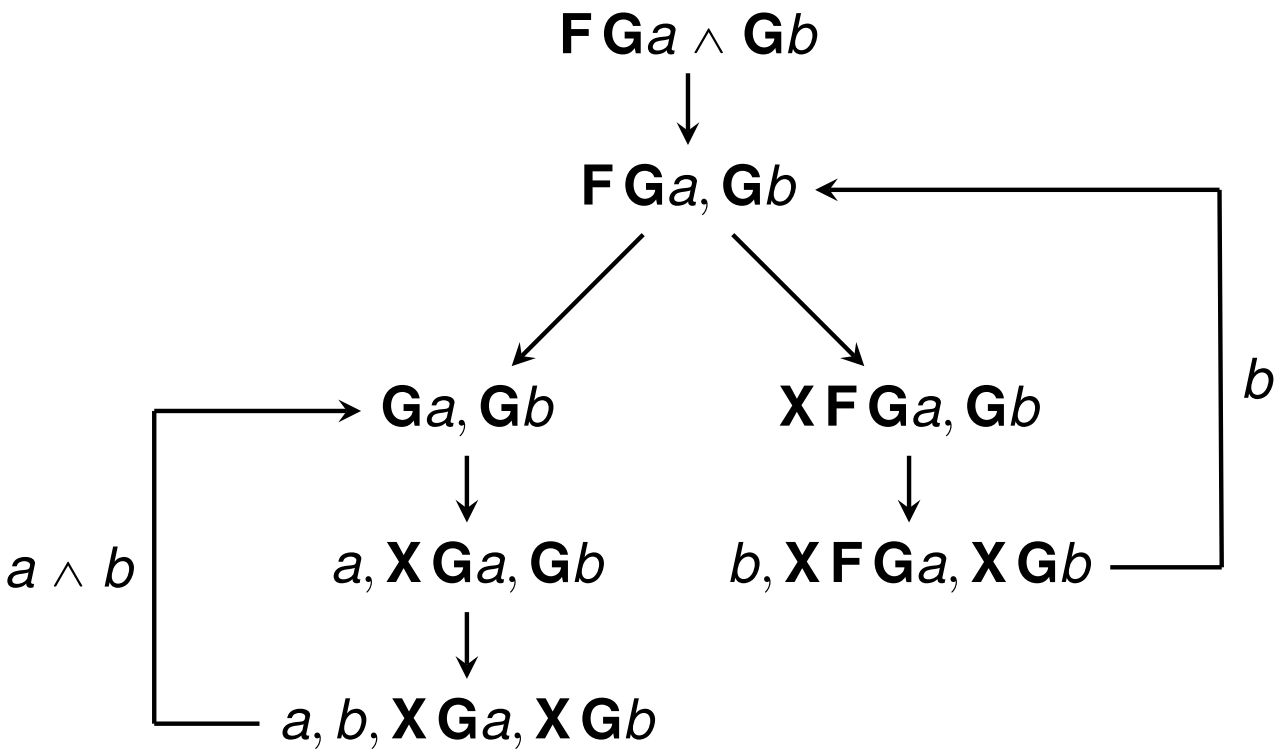
Exemple


$\mathbf{F}\,\mathbf{G}a \wedge \mathbf{G}b$
$\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$
$\mathbf{G}a, \mathbf{G}b$
$\mathbf{X}\,\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$
$a, \mathbf{X}\,\mathbf{G}a, \mathbf{G}b$
$b, \mathbf{X}\,\mathbf{F}\,\mathbf{G}a, \mathbf{X}\,\mathbf{G}b$
$a, b, \mathbf{X}\,\mathbf{G}a, \mathbf{X}\,\mathbf{G}b$

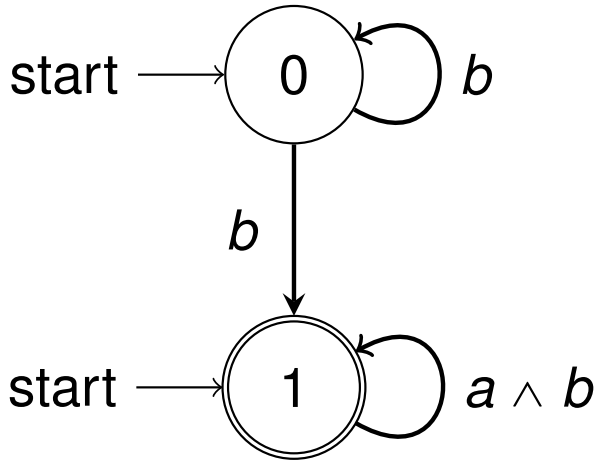
Exemple


$\mathbf{F}\,\mathbf{G}a \wedge \mathbf{G}b$
$\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$
$\mathbf{G}a, \mathbf{G}b$
$\mathbf{X}\,\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$
$a \wedge b$
$a, \mathbf{X}\,\mathbf{G}a, \mathbf{G}b$
$b, \mathbf{X}\,\mathbf{F}\,\mathbf{G}a, \mathbf{X}\,\mathbf{G}b$
$a, b, \mathbf{X}\,\mathbf{G}a, \mathbf{X}\,\mathbf{G}b$

Exemple


$\mathbf{F}\,\mathbf{G}a \wedge \mathbf{G}b$
$\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$
$\mathbf{G}a, \mathbf{G}b$
$\mathbf{X}\,\mathbf{F}\,\mathbf{G}a, \mathbf{G}b$
$a \wedge b$
$a, \mathbf{X}\,\mathbf{G}a, \mathbf{G}b$
$b, \mathbf{X}\,\mathbf{F}\,\mathbf{G}a, \mathbf{X}\,\mathbf{G}b$
b
$a, b, \mathbf{X}\,\mathbf{G}a, \mathbf{X}\,\mathbf{G}b$

Exemple



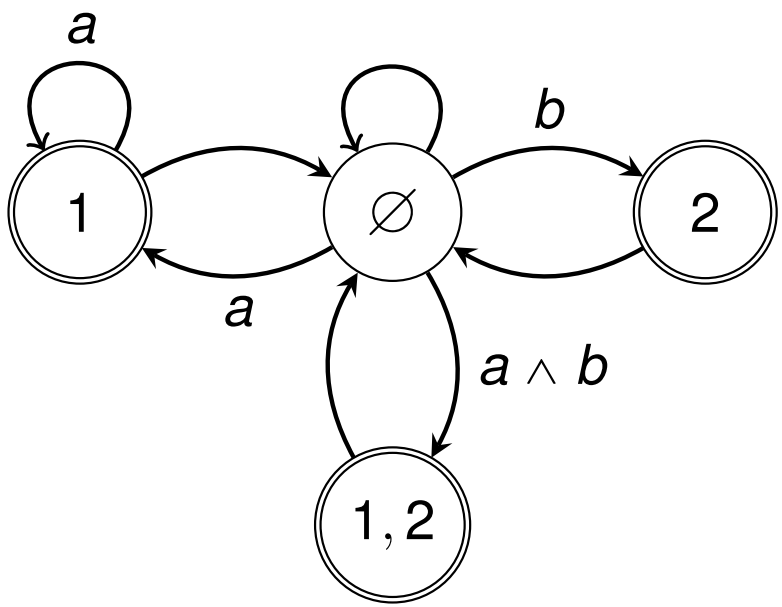
Spot

spot.lre.epita.fr/app/

Un autre exemple

Une autre formule

G F $a \wedge$ **G F** b



Condition généralisée

On doit passer infiniment souvent par des "1" **et** par des "2".

Automate de Büchi généralisé

Définition

Un **automate de Büchi** est un tuple $(\Sigma, Q, I, \mathcal{F}, \delta, \rho)$ où :

- Σ est un alphabet (souvent, $\Sigma = 2^{\mathbf{AP}}$),
- Q est un ensemble d'état,
- I est l'ensemble des états initiaux,
- $\mathcal{F}$ est une liste d'ensemble d'états acceptants,
- $\delta \subseteq Q \times Q$ est l'ensemble des transitions,
- $\rho : \delta \to 2^{\Sigma}$ est une fonction qui associe à chaque transition l'ensemble des variables qui sont vérifiées par cette transition.

Chemin acceptant

Un chemin acceptant d'un automate de Büchi est une suite **infinie** d'états reliés par des transitions et passant infiniment souvent par chacun des ensembles de $\mathcal{F}$.

Produit synchrone

But

Composer l'automate représentant les traces du système avec celui représentant la propriété à vérifier. Pour obtenir $A_S \times A_{\neg\phi}$ tel que :

$$L(A_S \times A_{\neg\phi}) = L(A_S) \cap L(A_{\neg\phi})$$

Langage vide

$L(A_S) \cap L(A_{\neg\phi})$ est l'ensemble des traces de S qui ne satisfont pas ϕ. Donc $S \models \phi$ ssi $L(A_S \times A_{\neg\phi}) = \varnothing$.

Produit synchrone

Définition

- $Q = Q_1 \times Q_2$,
- $I = I_1 \times I_2$ est l'ensemble des états initiaux,
- $\mathcal{F} = \{f \times Q_2 | f \in \mathcal{F}_1\} \cup \{Q_1 \times f | f \in \mathcal{F}_2\}$,
- $\delta = \{((q_1, q_2), (s_1, s_2)) | (q_1, s_1) \in \delta_1 \wedge (q_2, s_2) \in \delta_2\}$ est l'ensemble des transitions,
- pour tout élément de δ, $\rho((q_1, q_2), (s_1, s_2)) = \rho_1(q_1, s_1) \cap \rho_2(q_2, s_2)$

Test du lanagage vide automate de buchi non généralisé

Problème

Soit A un automate de büchi, est-ce que A accepte au moins une exécution ?

Solution

Est-ce qu'il existe $q \in F$ tel que :

- q est accessible depuis au moins un état initial,
- q est accessible en au moins 1 étape depuis lui-même

Donc on se réduit à de l'accessibilité dans un graphe.

Test du lanagage vide "généralisé"

Problème

Soit A un automate de büchi généralisé, est-ce que A accepte au moins une exécution ?

Solution

Est-ce qu'il existe $(q_1, \ldots, q_n) \in F_1, \ldots F_n$ tel que :

- q_1 est accessible depuis au moins un état initial,
- pour tout i, q_{i+1} est accessible depuis q_i,
- q_1 est accessible depuis q_n.

Donc on se réduit encore et toujours à de l'accessibilité dans un graphe.

Tableau