Vérification de système par composition de spécification CTL

Une Démarche incrémentale

Cécile Braunstein

UPMC/LIP6/ASIM

Plan

- Introduction au Model checking de propriétés CTL
- La conception incrémentale
- Composition de spécifi cations CTL
- Planning prévisionnel

Méthode de validation d'architecture

- Simulation
  - Problème de taux de couverture et cher en temps
- Vérifi cation déductive (Theorem prover)
  - Pas complètement automatisable
- Model checking
  - Complètement automatisable mais explosion combinatoire

Model checking

Étant donnés un système et une spécifi cation,

Est-ce que ce système satisfait la spécifi cation ?

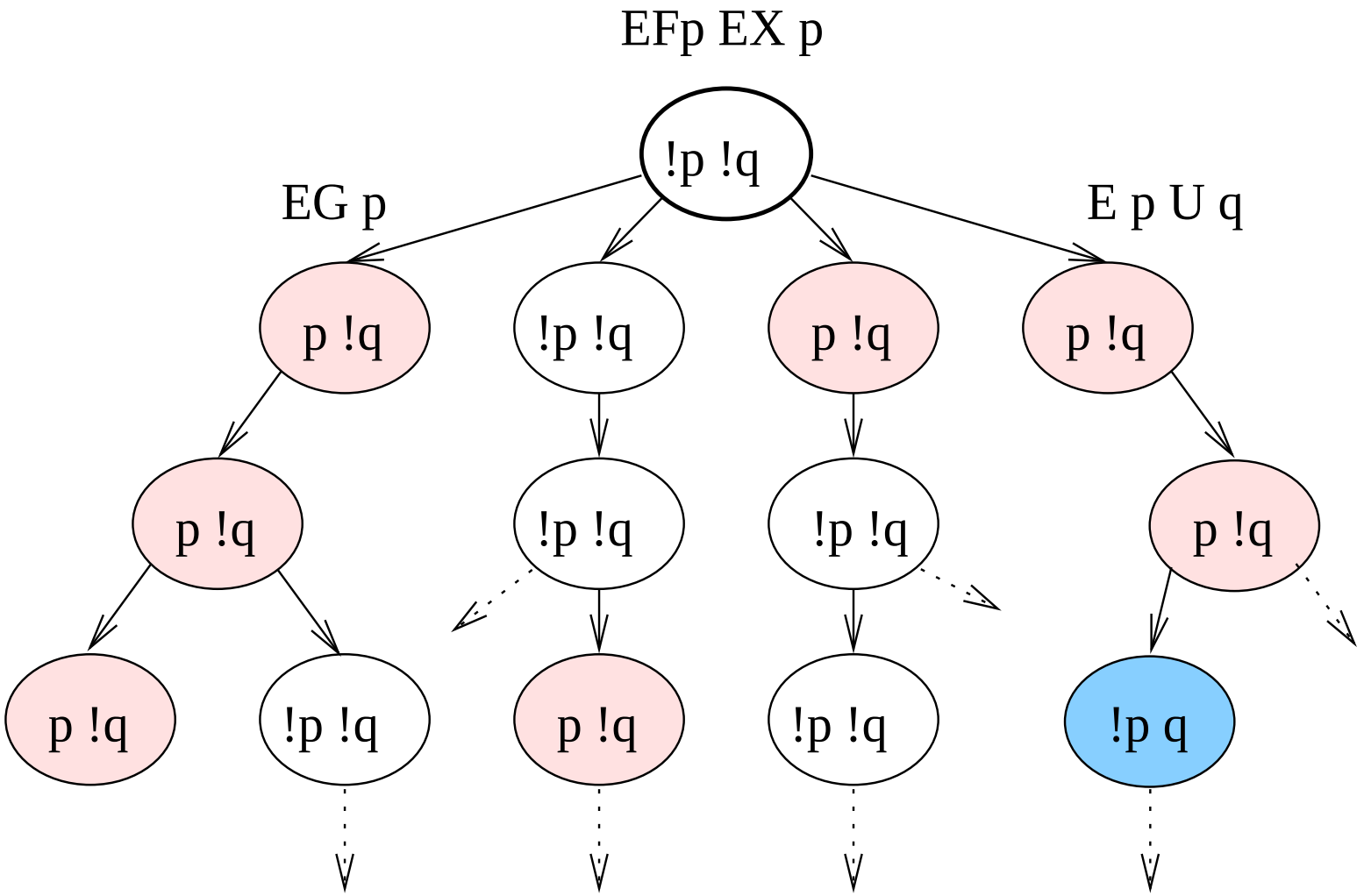
Restriction du problème :

- Le système : Une machine d'états fi nis (structure de Kripke) M = $\langle S, s_0, AP, \mathcal{L}, R \rangle$
- La spécifi cation : Un ensemble de formules CTL Φ.
- La question : $\forall \varphi \in \Phi, M, s_0 \models \varphi$?

CTL

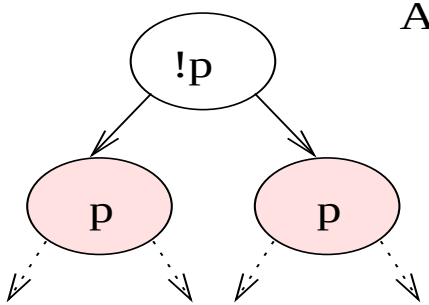
- Logique temporelle arborescente
  - Notion d'ordre d'événements dans le futur
  - Notion de possible et d'inévitable
- Défi nie sur l'arbre d'exécution obtenu par déroulement de la structure de Kripke
- Opérateurs
  - logique propositionnelle : $\neg, \vee, \wedge, \Rightarrow$...
  - opérateurs temporels : X, F, U, G
  - quantifi cateurs de chemin : E, A
- Tout opérateur temporel est quantifi é

CTL

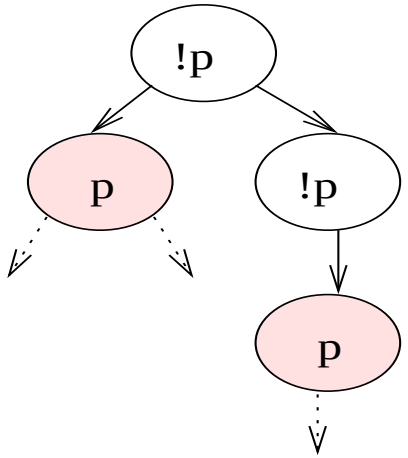


CTL

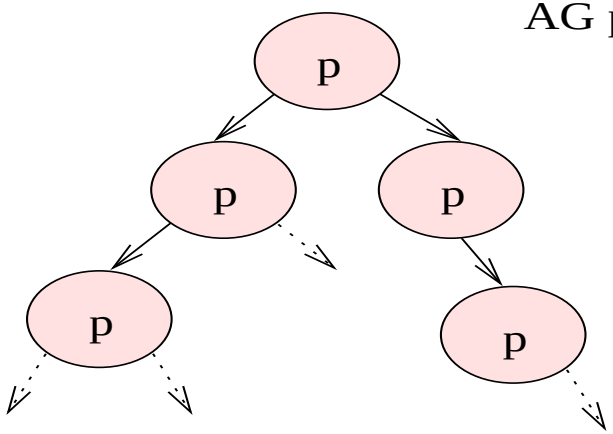
AX p



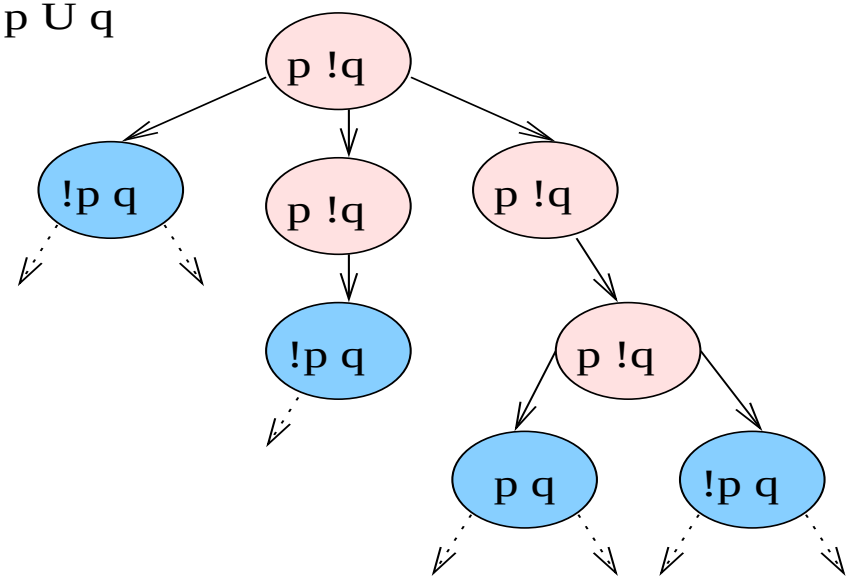
AF p



AG p



A p U q



Le problème de l'explosion combinatoire

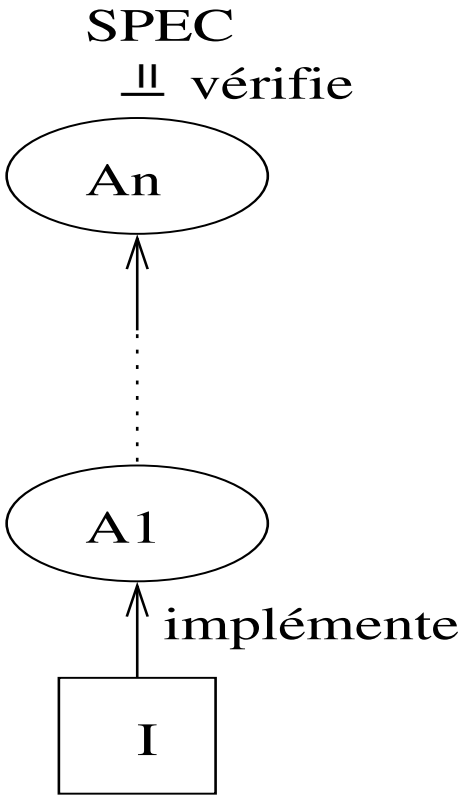
- Model checking explicite : 10^6 états
- Symbolic model checking (BDD) : 10^{20} états (Burch/Clarke/McMillan 90)
- Une architecture : > 10^6 registres 1 bit
  - Explosion combinatoire : Systèmes trop grands et trop complexes
  - L'évaluation d'une formule CTL sur une structure de Kripke : complexité linéaire sur la taille du modèle et de la formule

Des Solutions

- Représentation de l'espace d'état dans une structure compacte : BDD, DDD . . .
- Sur les modèles :
  - Enlever de l'information
  - Prendre en compte la structure du système en le composant

Abstraction

Vérification Bottom-Up



- Abstraction des données
- Pré-ordre sur les niveaux d'abstraction (relation de simulation)

$$I \preceq A1 \preceq \ldots \preceq An \models SPEC$$

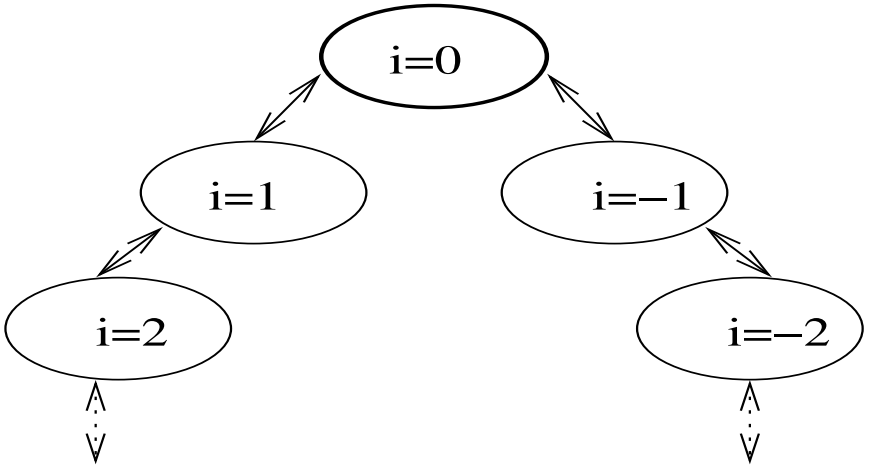
  - Préservation des propriétés ACTL (Grumberg et Long 94)

$$SPEC \models \varphi_{ACTL} \Rightarrow I \models \varphi_{ACTL}$$

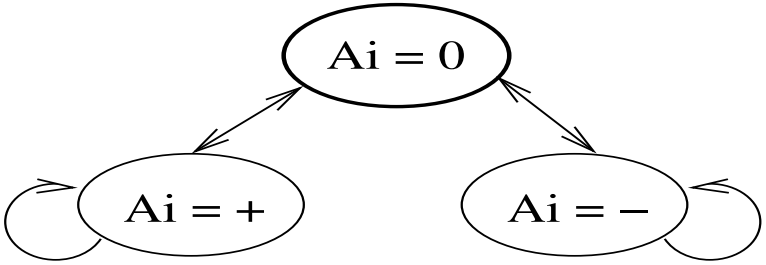
  - Le model checking est effectué sur le modèle réduit

Abstraction - Exemple

Le modèle concret

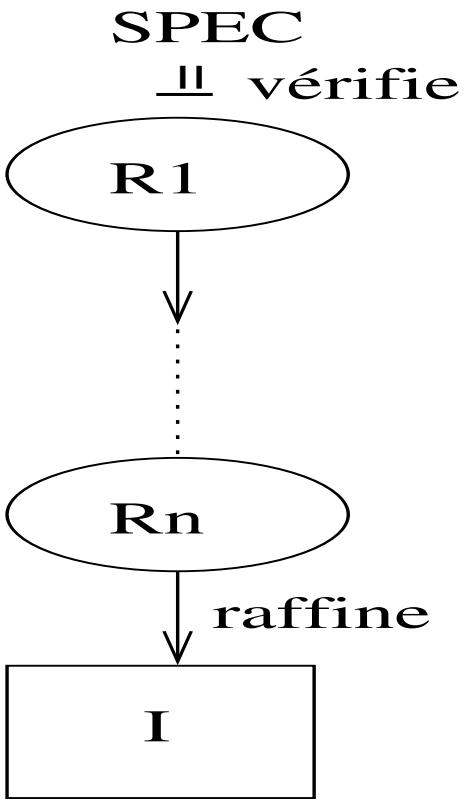


Une Abstraction



Raffinement

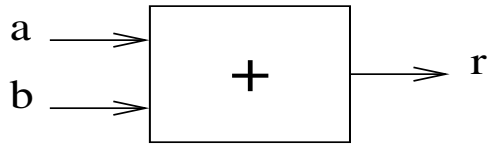
Top-Down



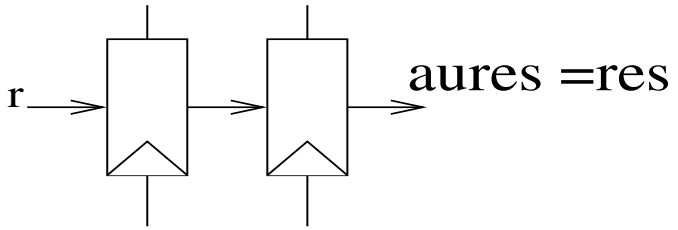
- Rendre des valeurs déterministes
- Spécialiser du comportement
- Associer à chaque objet de R_j une interprétation dans R_{j-1}.
  $I \preceq R_n \preceq \ldots \preceq R_1 \models SPEC$
  - Aucun nouveau comportement ajouté
  - Préservations des propriétés ACTL

Raffinement - Exemple

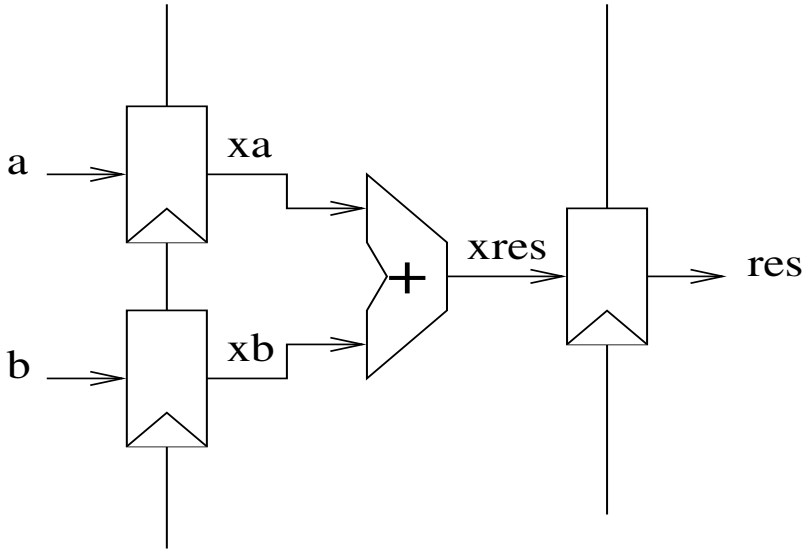
Abstraction



Rafinement map



Modèle concret

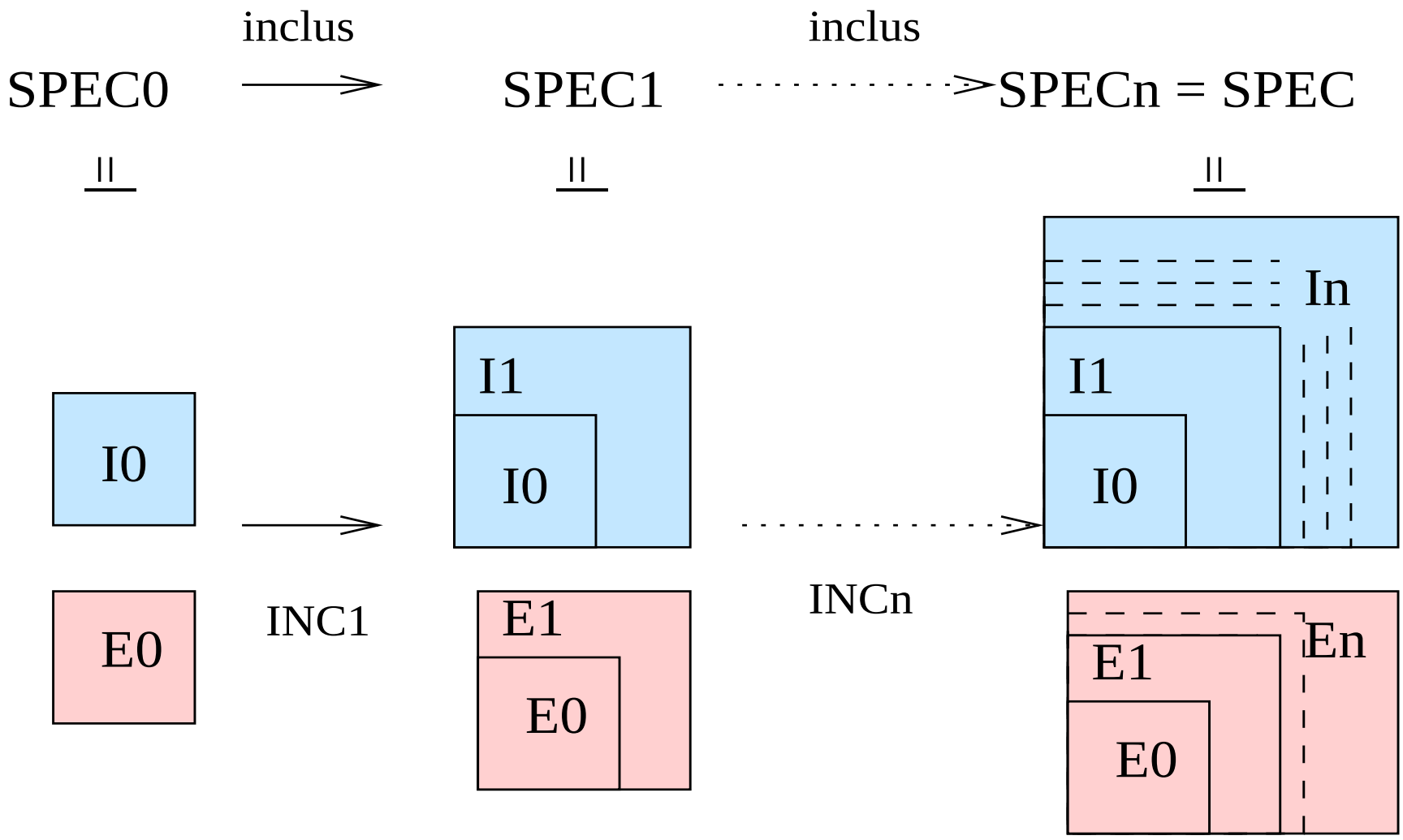


Feature integration (Plath/Ryan 2001)

- Projet d'intégration de services pour les téléphones (FireWorks)
- Ajout de nouveaux comportements à un système de base (téléphone)
- Vérifi ent à posteriori si les ajouts sont compatibles ou non

(Téléphone	(Téléphone
+ Transfert d'appel si occupé)	+ Rappel automatique)
+ Boîte vocale si occupé	+ Transfert d'appel
Mauvaise interaction	Bonne interaction

La conception incrémentale



La conception incrémentale

- **Additions successives de nouveaux comportements**
- $M_i \rightarrow M_{i+1}$, au moins 1 nouvel événement sur l'interface $\equiv$ 1 nouveau signal
- Une valeur muette et une valeur active sont attribuées à chaque nouveau signal
- M_{i+1} englobe le comportement de M_i (relation de simulation)
- $K(M_{i+1})$ inclut $K(M_i)$ avec tous les états qui étaient dans $K(M_i)$ étiquetés par la valeur muette.

Résultats

- Formalisation d'une méthode de conception
- Théorème sur la transformation de formules CTL [AVOCS 04] [STTT 05]
- Théorèmes sur la transformation et la préservation de propriétés CTL dans le cas d'une architecture pipeline
- Application à la plate-forme VCI-PI
  - Construction incrémentale de 6 wrappers
  - Application des théorèmes
  - Pour 2 maîtres et 1 esclave : temps de vérifi cation trop grand (>24h)

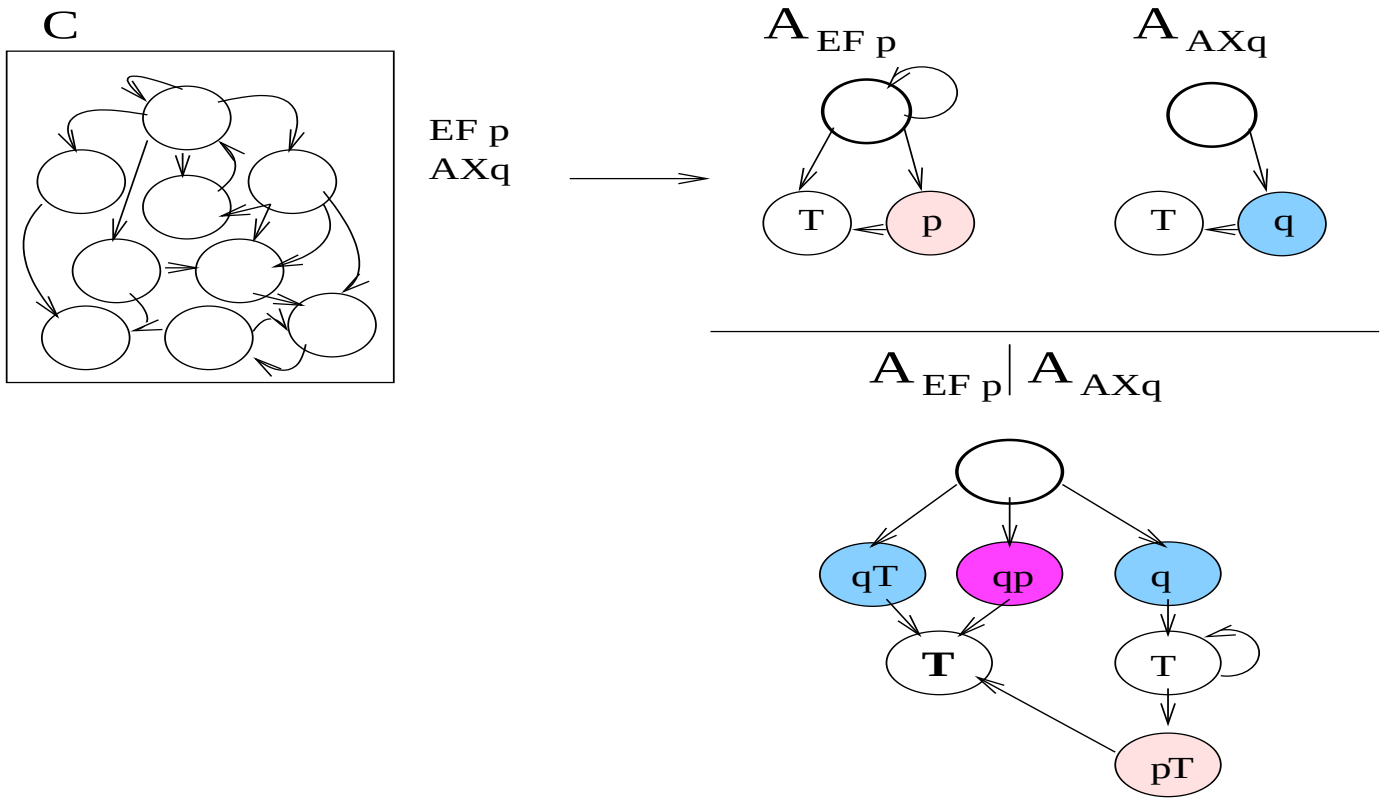
Composition de spécification

But : Vérifi er une propriété globale (sur plusieurs composants)

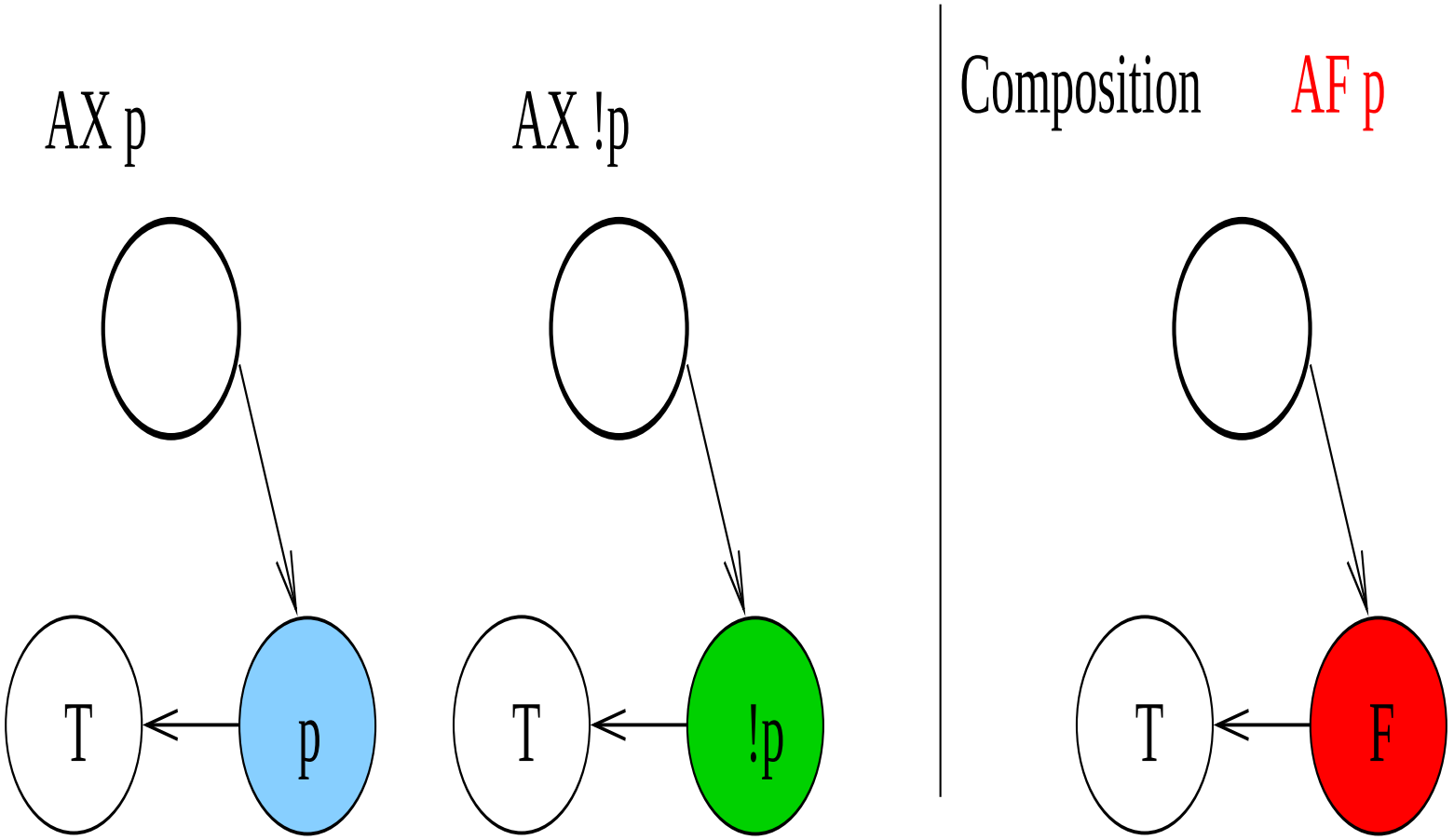
- Abstraction conservative par composant :
  - Construction d'un modèle plus simple (A) vérifi ant un sous-ensemble des propriétés du composant (M)
- Processus itératif :
  - Si $A \models \varphi \Rightarrow M \models \varphi$
  - Sinon Raffi ner = ajouter des contraintes à partir du contre-exemple

Abstraction d'un composant

- Le composant (C) vérifi e au moins φ_1 et φ_2
- A_{φ_1,φ_2} : un modèle vérifi ant $\varphi_1 \wedge \varphi_2$



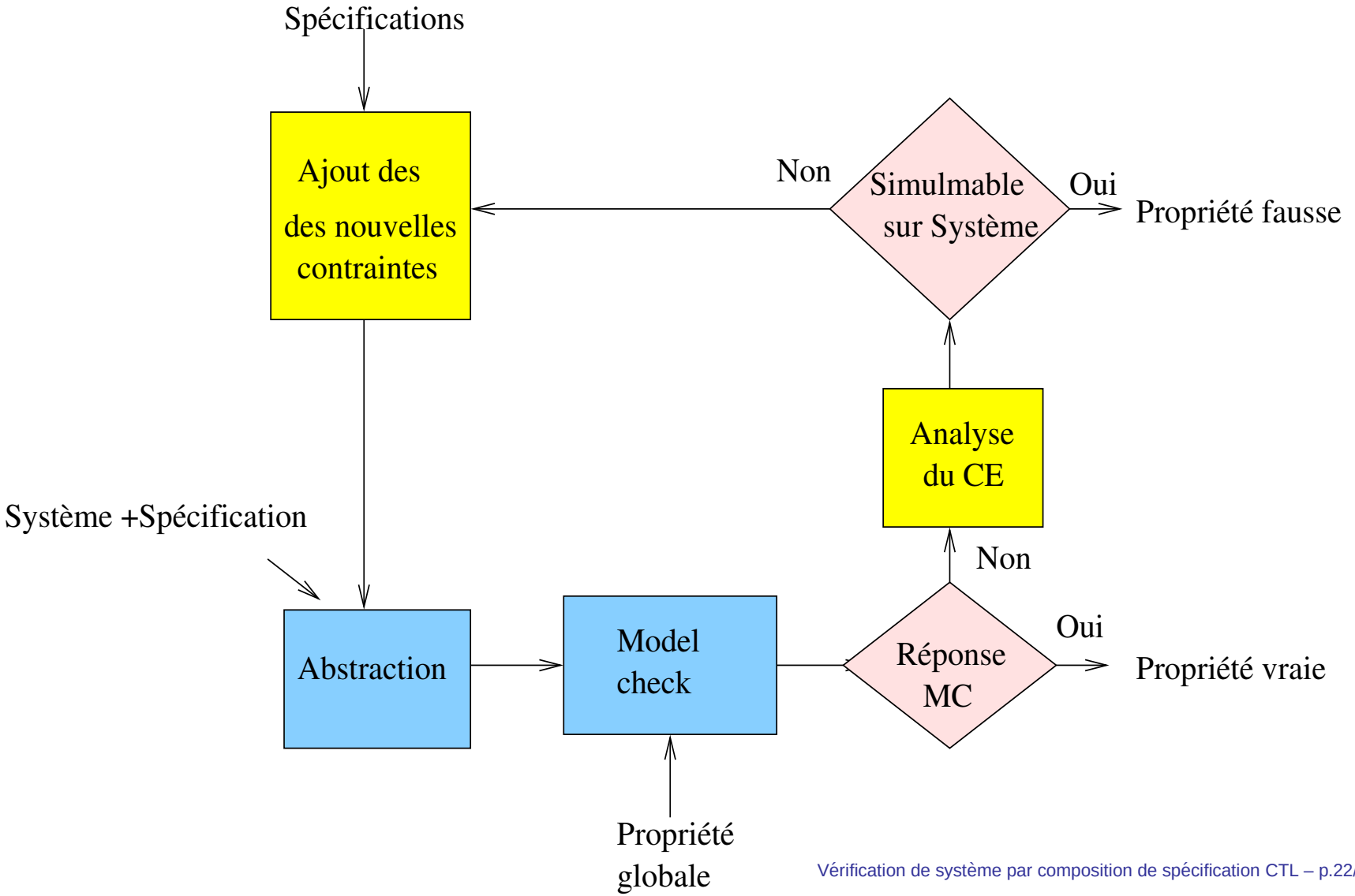
Composition d'abstraction



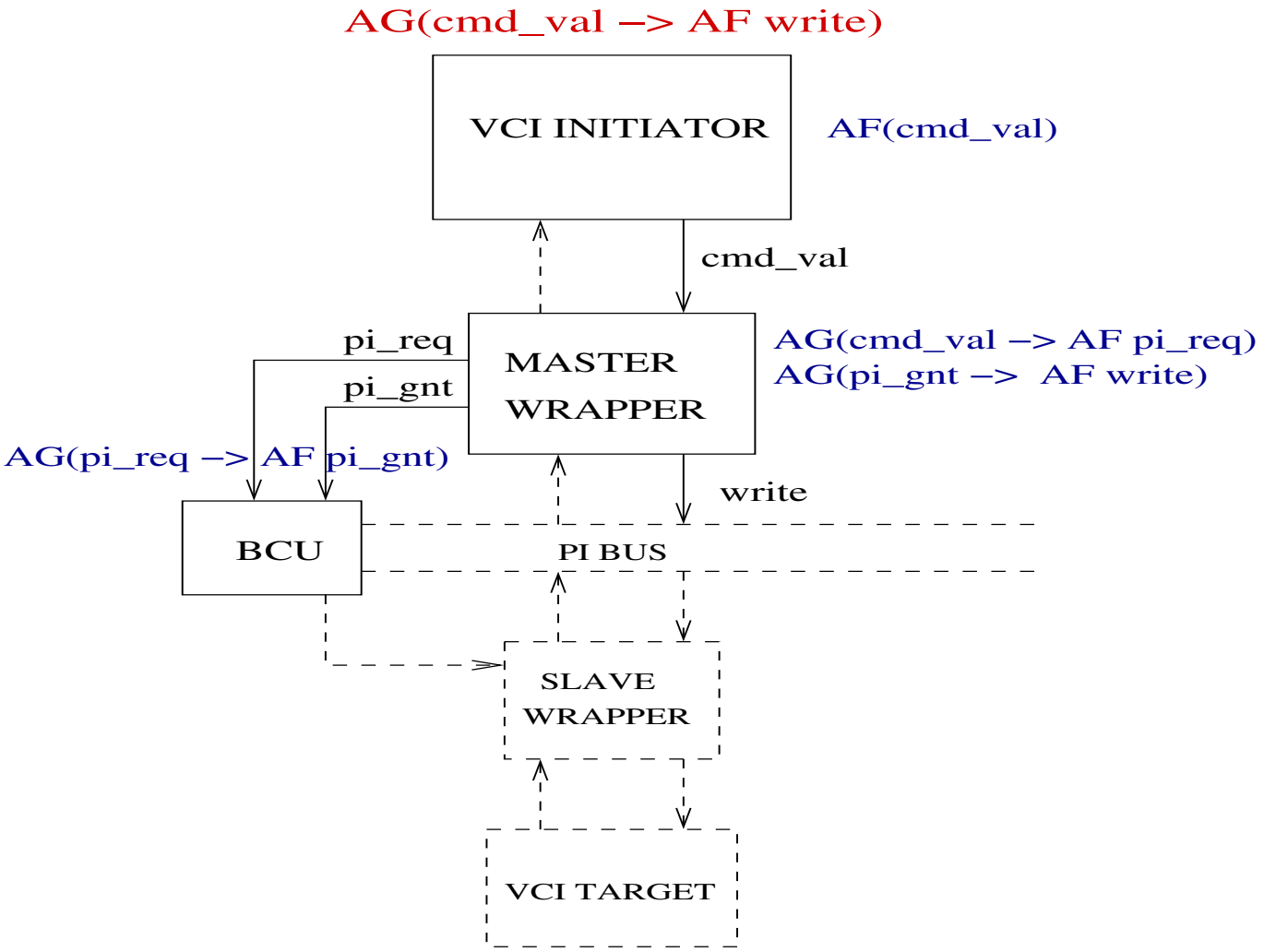
Composition d'abstraction

- Soit $C_1|C_2$, $A = A_{\varphi_{C1}}|A_{\varphi_{C2}}$ et ϕ une propriété globale à vérifi er
  1. Si $A \models \phi \Rightarrow C_1|C_2 \models \phi$
  2. Si il existe un chemin σ vers un état faux alors on analyse le contre-exemple
     - ➢ Si σ peut s'exécuter sur $C_1|C_2$ alors $C_1|C_2 \not\models \phi$
     - ➢ Si σ ne peut pas s'exécuter sur $C_1|C_2$ alors retirer σ e.g. ajouter une contrainte qui invalide ce chemin

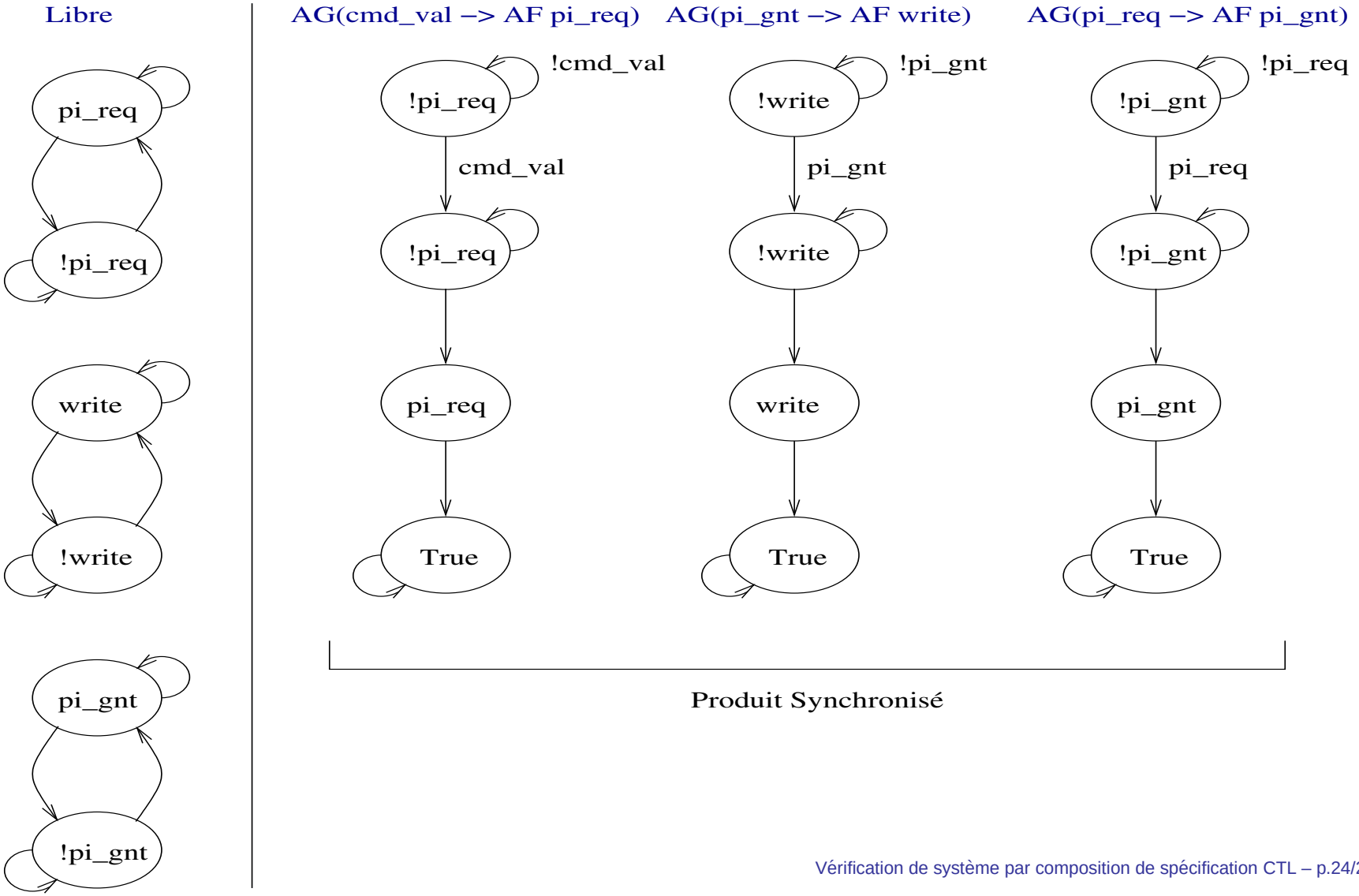
Boucle de raffinement du software model checking



Étude de cas


AG(cmd_val –> AF write)
VCI INITIATOR
AF(cmd_val)
cmd_val
pi_req
pi_gnt
MASTER
WRAPPER
AG(cmd_val –> AF pi_req)
AG(pi_gnt –> AF write)
AG(pi_req –> AF pi_gnt)
write
BCU
PI BUS
SLAVE
WRAPPER
VCI TARGET

Ajout des contraintes



Résultats

	Plate forme	Complète (1M/1S)	Abstraite (1M/1S)
	Nombre de variables Bdd	315	199
	Profondeur du FSM	542	9
	# états accessibles	1.082×10^{6}	1.14×10^{14}
sans reorder	analyse d' accessibilité en secondes	647.85 s	1 s
sans reorder	Vérification propriété en secondes	145.86 s	0.01 s
sift	analyse d' accessibilité en secondes	19.53 s	1 s
sift	Vérification propriété en secondes	7.94s	0.01 s

Travail à venir

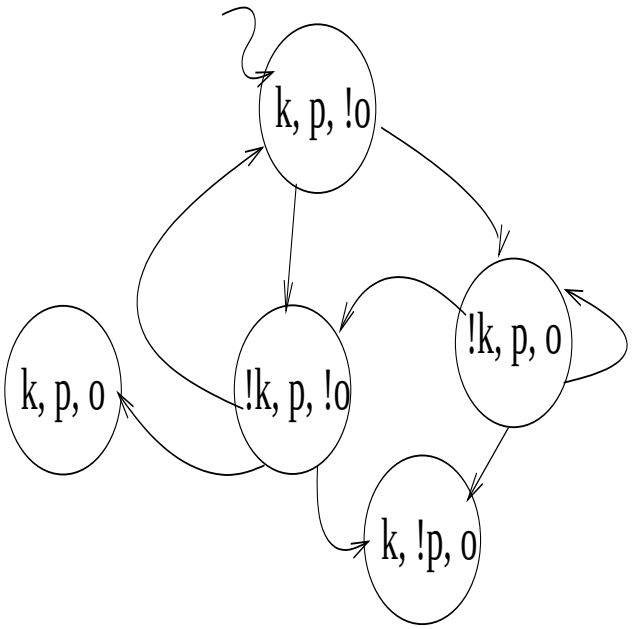
- Formaliser l'abstraction et la composition
- Prouver que la composition d'abstraction est conservative
- Défi nir + Implémenter la construction de l'abstraction (Stage M2)
- Intégrer la boucle de raffi nement
- Développer un prototype intégré dans VIS

Planning

Decembre Janvier	Formalisation de l'abstraction +composition	
Février	Preuve conservativité	Séminaire MeFoSyLoMA
Mars	Boucle de raffinement	FMCAD 06
Avril Mai Juin	Réalisation d'un prototype	Rédaction
Juillet Aout Septembre		
Octobre	Soutenance	

Stage M2

Structure de Kripke



M = $\langle S, I, AP, \mathcal{L}, R \rangle$

- S ensemble d'états.
- $s_0 \subseteq S$ ensemble d'états initiaux.
- AP proposition atomique.
- $\mathcal{L} = \{l_0, \ldots, l_{|AP|-1}\}$ vecteur de $|\text{AP}|$ fonctions.
- $R \subseteq S \times S$ relation de transitions.