Diagnostic et Diagnosticabilité

Formation sur les Systèmes à Événements Discrets (SED)

2e édition
Mars 2025
Nantes

Société d'Automatique,
de Génie Industriel & de Productique

Introduction

Diagnostic de systèmes à événements discrets: rappels

- Soit G le modèle d'un SED
  - G: automates classiques, réseaux de Petri,….
    - G produit des événements observables (Σ_o) et non-observables (Σ_u)

$$\Sigma = \Sigma_o \cup \Sigma_u$$

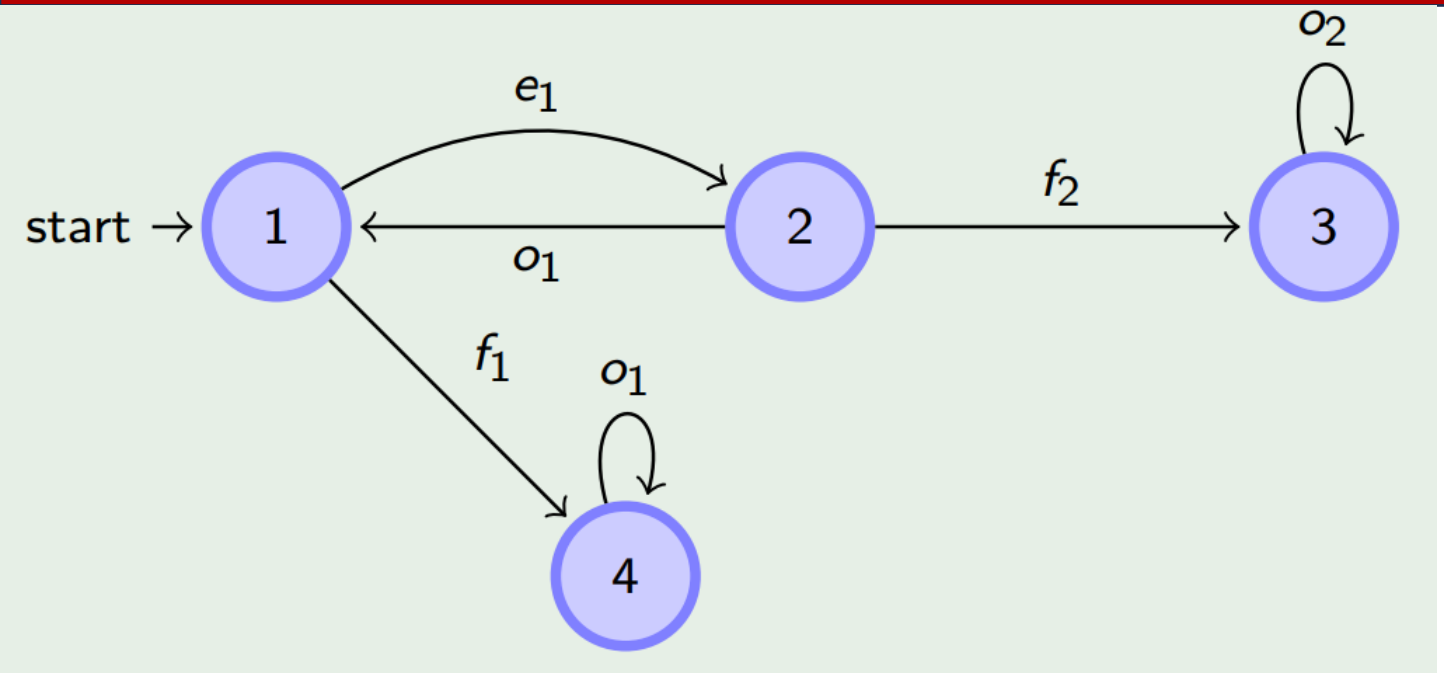
- G peut subir des défaillances: événements non-observables

$$f_1, f_2, \ldots \in \Sigma_f \subseteq \Sigma_u$$

- $L(G)$ langage des séquences de G, $L_o(G)$ langage observable de G
- Soit $\sigma \in L_o(G)$, une séquence d'observations de G

- Diagnostic: quels sont les événements de Σ_f qui ont pu avoir lieu dans G si on observe σ sur le système ?

Diagnostic de systèmes à événements discrets: exemple



- $\sigma_1 = o1$ Diagnostic: f1 (ambigu), peut être normal
- $\sigma_2 = o1\ o1$ Diagnostic: f1 (ambigu), peut être normal
- $\sigma_3 = o1\ o1\ o2$ Diagnostic: f2 (certain)

(diagnostic sans fermeture silencieuse)

- o1, o2 observables
- f1, f2 défaillances
- e1 non observable
- Langage L(G):

  (e1o1)* + e1(o1e1)* +
  (e1o1)*f2o2*+
  (e1o1)*f1o1*
- Langage Lo(G):
  - o1* + o1*o2*

Du comportement séquentiel au comportement temporel

Supposons un système de type client/serveur web et les deux scénarios suivants:

1. Envoi de la requête **r** à la date **d**, réception de **p** à la date **d + 1ms**
2. Envoi de la requête **r** à la date **d**, réception de **p** à la date **d + 1 an**

Y'en a-t-il un scénario plus normal que l'autre ?

- Du point de vue séquentiel, on observe **r** suivi de **p**
- Du point de vue temporel, on observe:
  1. Scénario 1: **(r,d) (p,d+1)**
  2. Scénario 2: **(r,d) (p,d+31536000000)**

Seul le temps (information quantitative) discrimine les deux scénarios.
Le temps est une dimension observable (horloge)
Observations: flux d'événements datés: (a,1), (b,3), (a,4)…

Comment utiliser la mesure du temps pour raffiner un diagnostic ?

Diagnostic de systèmes à événements discrets temporels

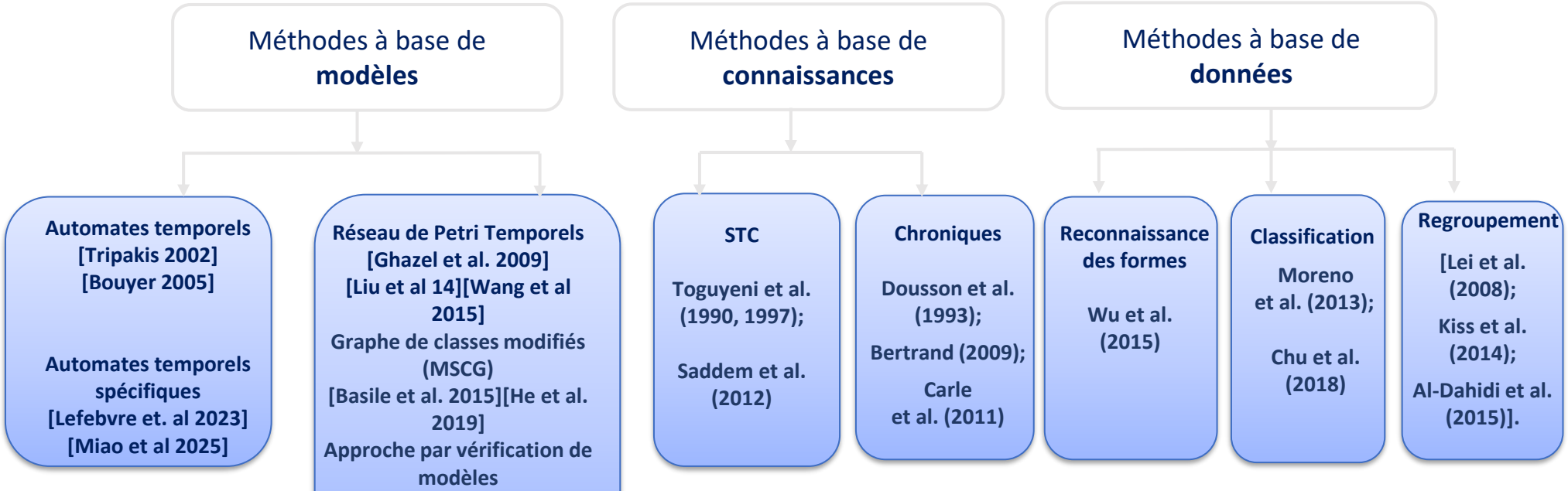
- L(G) devient un langage temporel:
  - La séquence (e1,3) (o1, 5) (e1, 8) (f2, 10) (o2, 24) est un mot de L(G)
  - Représentation en dates relatives :

    (e1,3), (o1,2), (e1,3)(f2,2)(o2,14) ou encore 3 e1. 2 o1. 3 e1. 2 f2. 14 o2

- Le langage observable Lo(G) devient un langage temporel
  - Séquence observée: $\sigma =$ (o1,5)(o,24)
  - En dates relatives: (o1,5)(o2,19) ou encore 5 o1. 19 o2.

- Diagnostic: quels sont les événements de Σ_f qui ont pu avoir lieu dans G si on observe σ sur le système ?

Méthodes de diagnostic des SED temporels et temporisés



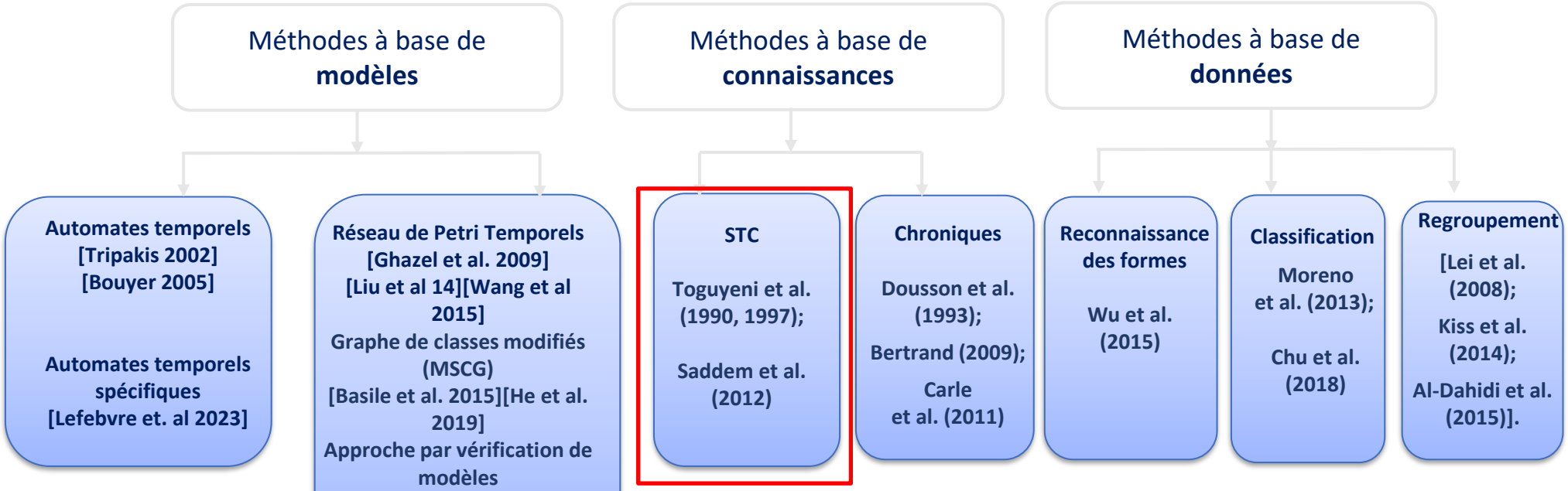
- La validation de la cohérence et la complétude des défauts à diagnostiquer.
- Problème d'explosion combinatoire lors de l'implémentation sur des systèmes réels

- Connaissance des symptômes des fautes qu'elles modélisent
- L'acquisition des connaissances expertes et de leur mise à jour

- Ne nécessite pas une modélisation explicite du système
- Apprendre des expériences
- Amélioration des performances
- Une étape de prétraitement des données
- Une grande quantité de données

Approche par Signature Temporelle Causale (STC)

Méthodes de diagnostic des SED temporels et temporisés



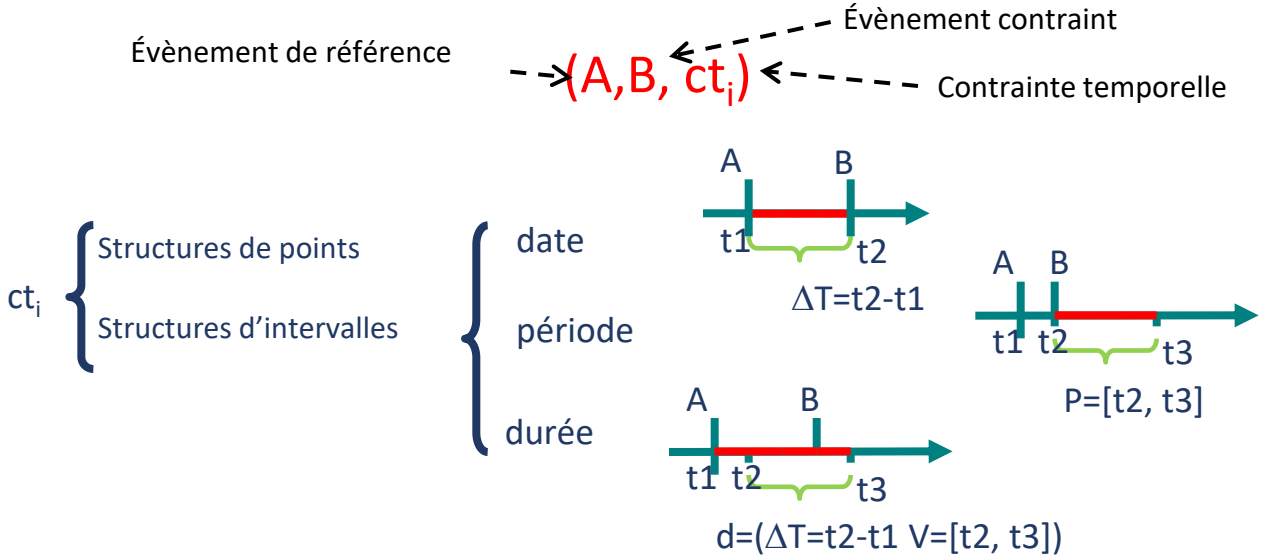
- La validation de la cohérence et la complétude des défauts à diagnostiquer.
- Problème d'explosion combinatoire lors de l'implémentation sur des systèmes réels

- Connaissance des symptômes des fautes qu'elles modélisent
- L'acquisition des connaissances expertes et de leur mise à jour

- Ne nécessite pas une modélisation explicite du système
- Apprendre des expériences
- Amélioration des performances
- Une étape de prétraitement des données
- Une grande quantité de données

Signatures Temporelles Causales

Une STC est un sous-ensemble d'événements observables partiellement ordonnés qui caractérise un comportement d'un système. Ces événements sont contraints par un ensemble de contraintes temporelles portant sur leurs occurrences.

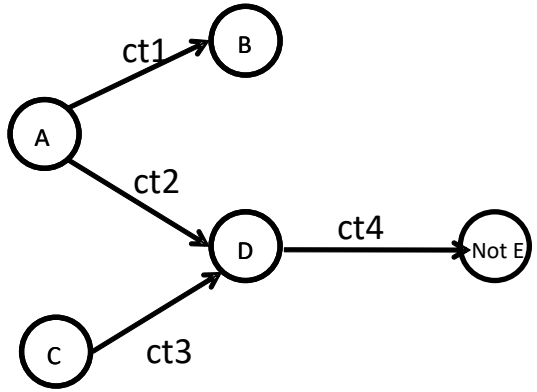


Signatures Temporelles Causales

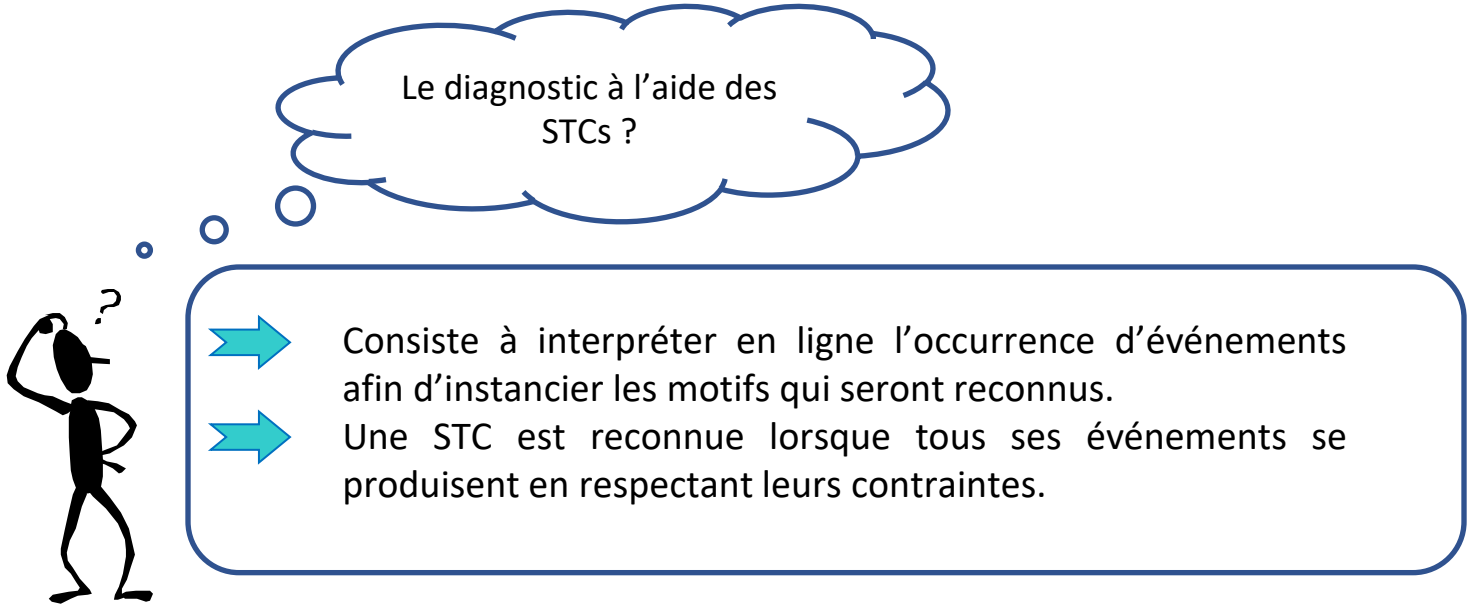
Syntaxe

$$\underbrace{(In, A, nct) * (In, C, nct) * (A, B, ct_1) * (A, D, ct_2) * (C, D, ct_3) * (D, not\ E, ct_4)}_{\text{Condition}} \rightarrow \underbrace{G}_{\text{Conséquence}}$$

Sémantique



Signatures Temporelles Causales



Diagnostic à base de STC

Base de STC

STC1 : (In, S1, [0,+∞[)*(S1, S2, [1,3])*(S2, S3,[1,2]) => F1
STC2 : (In, S2, [0,+∞[)*(S2, S1, [1,3]) => F2
STC3 : (In, S3, [0,+∞[)*(S3, S1, [1,2]) => F3

Observation

(S1, 0) (S2, 2) (S2, 3) (S3, 4) (S1, 5)

Diagnostic à base de STC

Basé sur le concept de monde : un ensemble d'hypothèses cohérentes d'affectations d'événements reçus par la tâche de diagnostic à une et une seule instance de STC.

- *Evènement attendu* { Evtatt, Tmin, Tmax }
- *L* : ensemble de STC instanciées { ensemble EvtPasse, ensemble Evènement attendu}
- *Monde* {L, ensemble de Causes, deadline (Monde) = Min(Tmax de l'ensemble STC instanciées) }

Diagnostic à base de STC

1. Hypothèse du monde Fermé : si un evt n'est pas explicitement reçu par la tâche de diagnostic, il est supposé être inexistant.

Diagnostic à base de STC

1. Hypothèse du monde Fermé

2: Une seule affectation par symptôme : Si un symptôme survient, il est affecté à une et une seule STC dans un monde.

Diagnostic à base de STC

1. Hypothèse du monde Fermé

2: Une seule affectation par symptôme

3: Duplication des mondes : Si plusieurs STC sont candidates pour l'affectation d'un symptôme, dupliquer le monde en cours en autant de mondes que d'hypothèses d'affectation cohérente.

Diagnostic à base de STC

1. Hypothèse du monde Fermé

2: Une seule affectation par symptôme

3: Duplication des mondes

4: Si un événement n'est affectable à aucune des STC du monde considéré ou si un événement attendu n'est pas survenu à MAX_TOi alors le monde considéré est supprimé

Diagnostic à base de STC

1. Hypothèse du monde Fermé

2: Une seule affectation par symptôme

3: Duplication des mondes

4: Suppression du monde fondé sur un ensemble d'hypothèses incohérentes

5: Fusion des deux mondes

Soient deux mondes Wi et Wj. Si Causesi = Causesj ; Li = Lj ; NEi = NEj ; MIN_TOi = MIN_TOj ; MAX_TOi = MAX_Toj , alors garder W i et supprimer Wj.

Diagnostic à base de STC

1. Hypothèse du monde Fermé

2: Une seule affectation par symptôme

3: Duplication des mondes

4: Suppression du monde fondé sur un ensemble d'hypothèses incohérentes

5: Fusion de deux mondes

6: Utilisation des causes inférées : Si un seul monde subsiste à une date et l'ensemble de ses causes n'est pas vide, les causes identifiées correspondent au diagnostic. Si Li est vide, alors réinitialiser tous les attributs du monde.

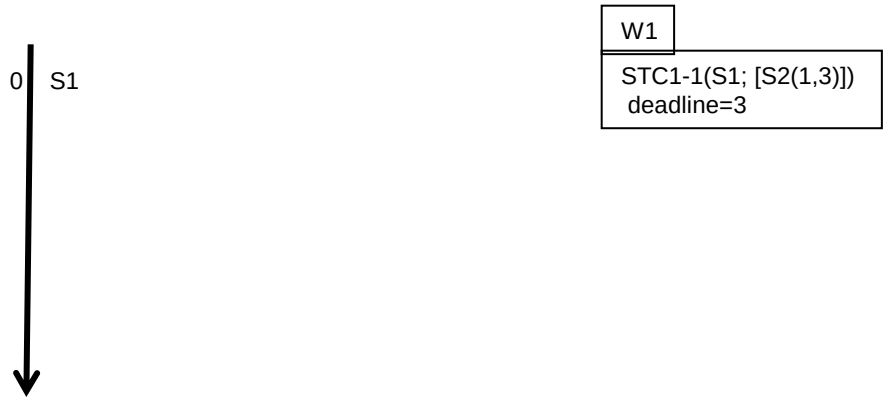
Diagnostic à base de STC

Base de STC

STC1 : (In, S1, [0,+∞[)*(S1, S2, [1,3])*(S2, S3,[1,2]) => F1
STC2 : (In, S2, [0,+∞[)*(S2, S1, [1,3]) => F2
STC3 : (In, S3, [0,+∞[)*(S3, S1, [1,2]) => F3

Observation

(S1, 0) (S2, 2) (S2, 3) (S3, 4) (S1, 5)



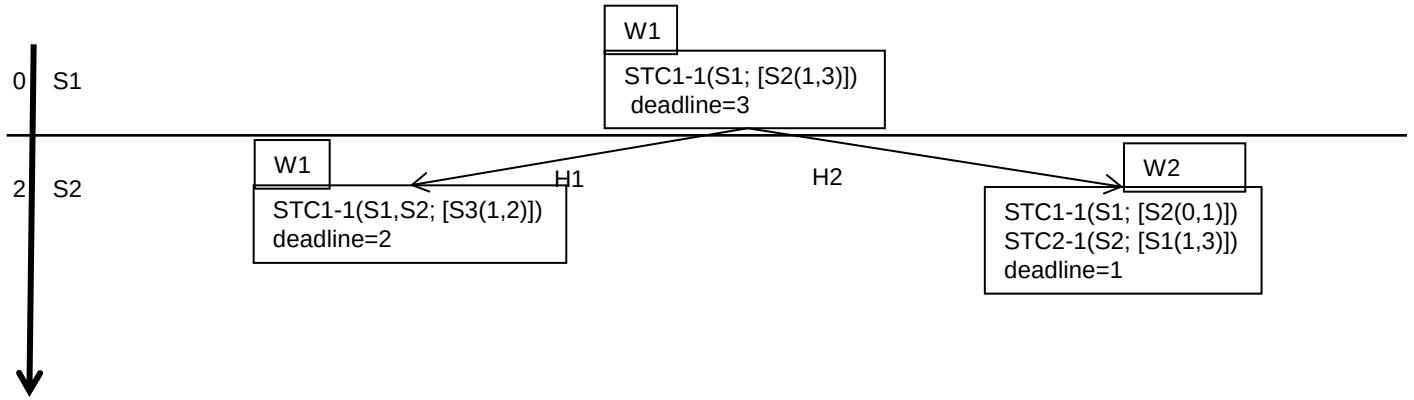
Diagnostic à base de STC

Base de STC

STC1 : (In, S1, [0,+∞[)*(S1, S2, [1,3])*(S2, S3,[1,2]) => F1
STC2 : (In, S2, [0,+∞[)*(S2, S1, [1,3]) => F2
STC3 : (In, S3, [0,+∞[)*(S3, S1, [1,2]) => F3

Observation

(S1, 0) (S2, 2) (S2, 3) (S3, 4) (S1, 5)



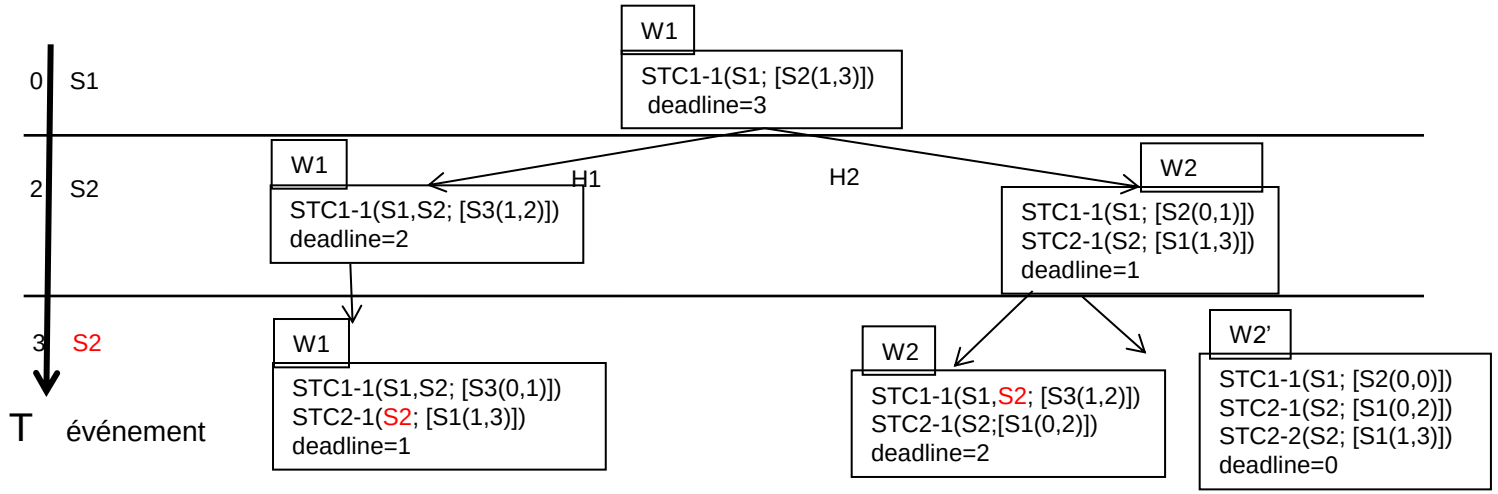
Diagnostic à base de STC

Base de STC

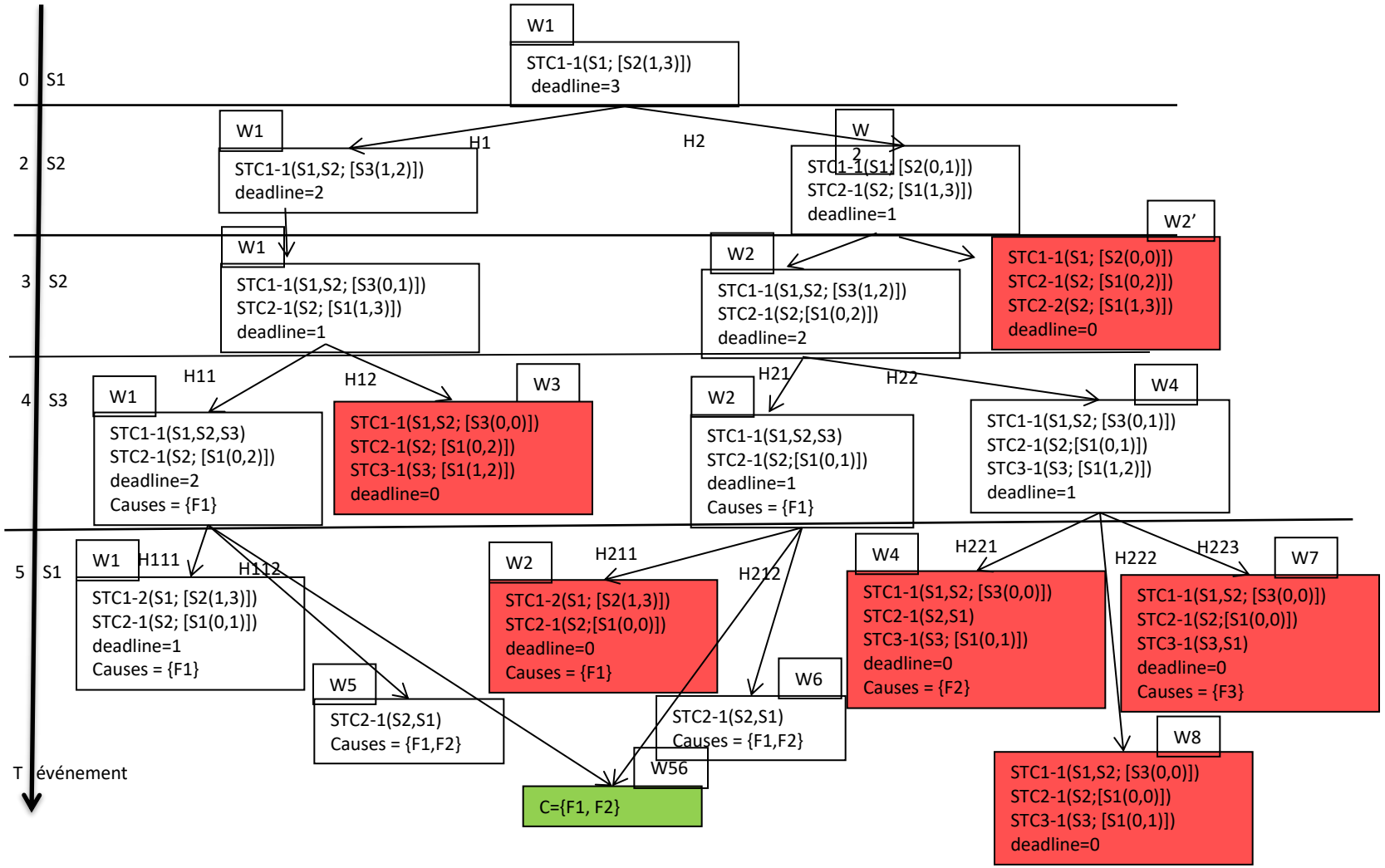
STC1 : (In, S1, [0,+∞[)*(S1, S2, [1,3])*(S2, S3,[1,2]) => F1
STC2 : (In, S2, [0,+∞[)*(S2, S1, [1,3]) => F2
STC3 : (In, S3, [0,+∞[)*(S3, S1, [1,2]) => F3

Observation

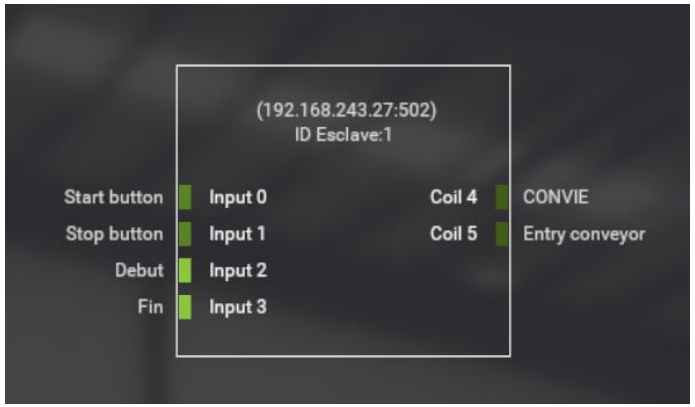
(S1, 0) (S2, 2) (S2, 3) (S3, 4) (S1, 5)



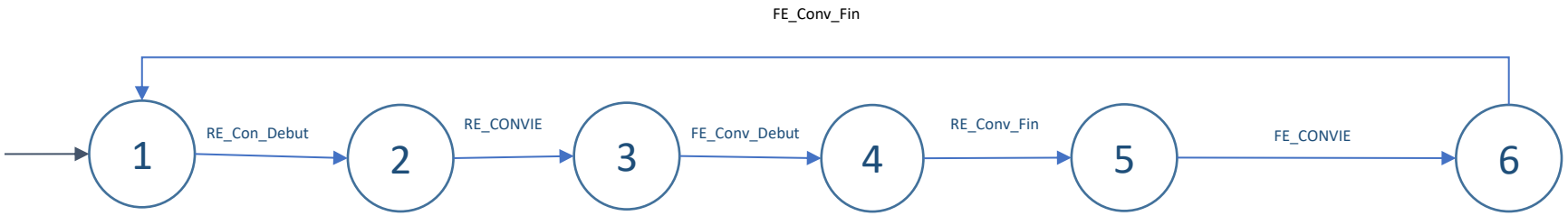
Diagnostic à base de STC



Système du convoyeur

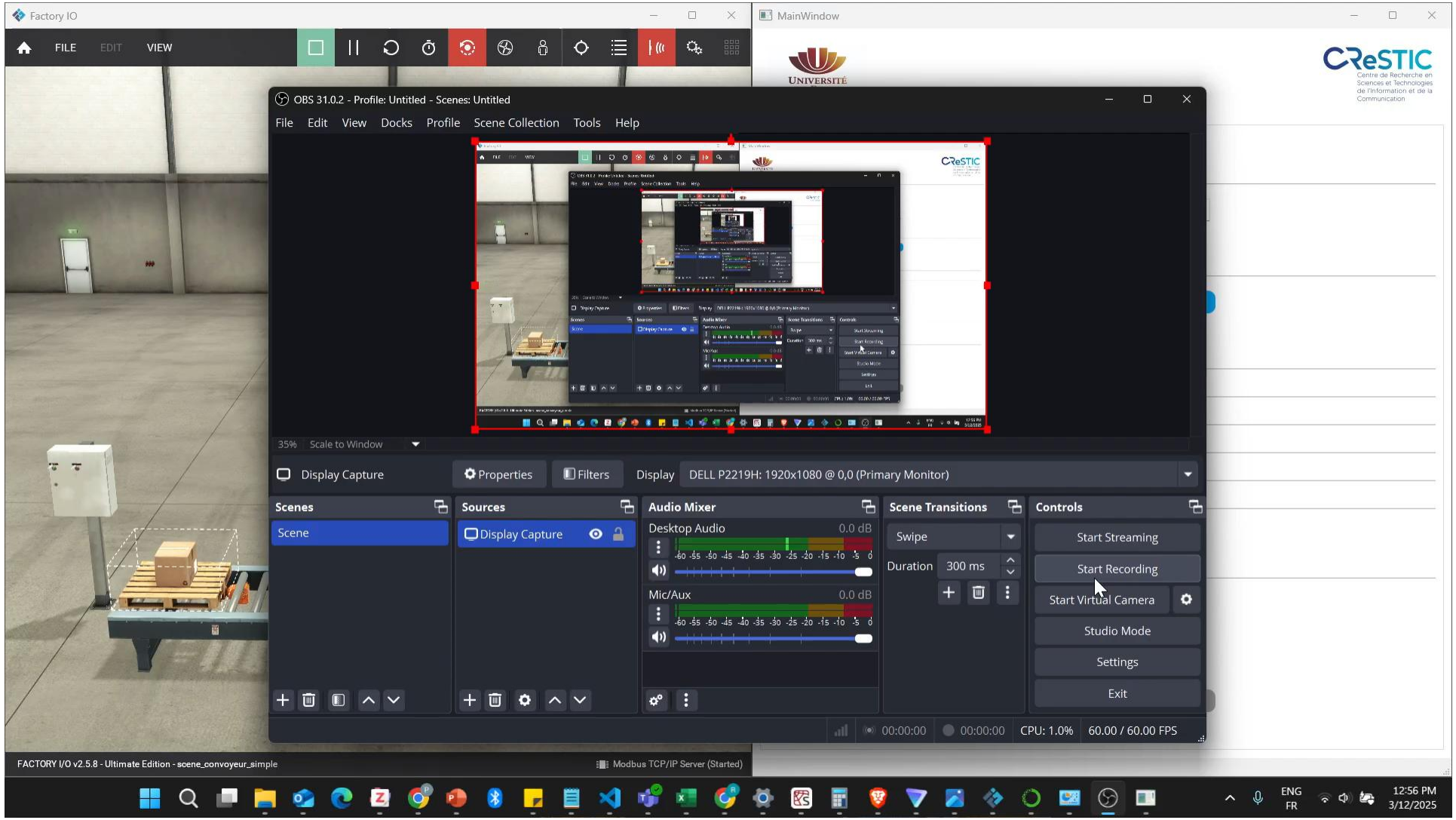


Système du convoyeur

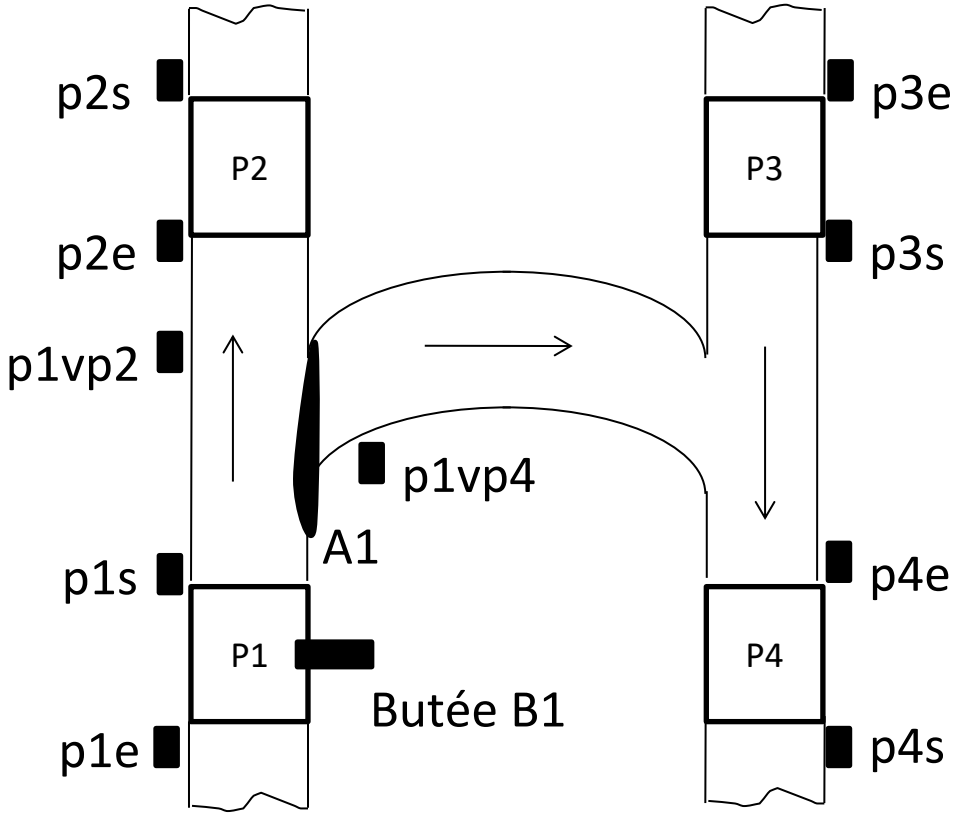


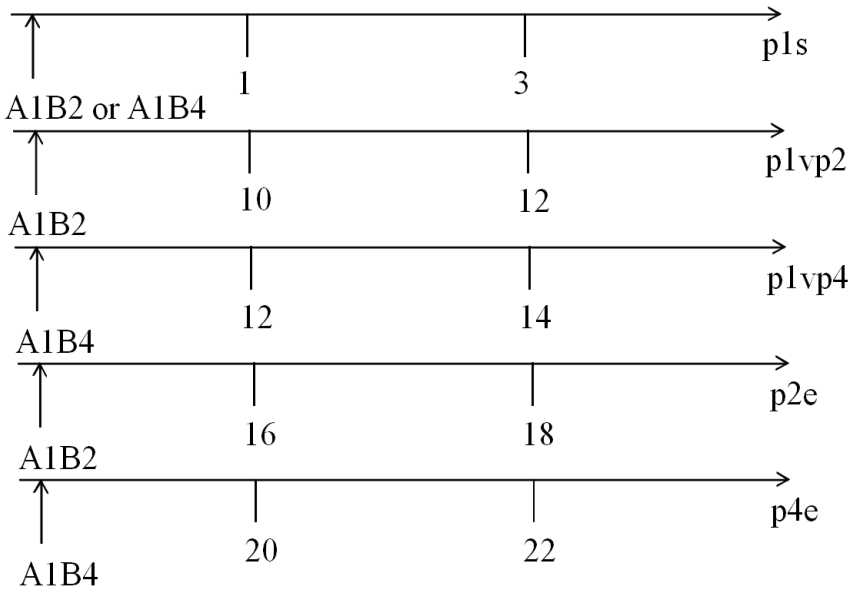
- (In, RE_Debut, [-1, -1]) * (RE_Debut, FE_Debut, [2.3 , 2.6]) * (FE_ Debut, RE_Fin, [6.7, 6.92]) * (RE_ Fin, FE_Fin, [1.7, 2.1]) => Normal
- (In, RE_Debut, [-1, -1]) * (RE_Debut, S_FE_Debut, [2.65 , 2.7]) => collage0Convoyeur
- (In, RE_Debut, [-1, -1]) * (RE_Debut, S_FE_Debut, [2.65 , 2.65]* (RE_ Debut, RE_Fin, [6.7, 6.92]) (RE_ Fin, FE_Fin, [1.7, 2.1]) =>collage1Debut
- (In, RE_Debut, [-1, -1]) * (RE_Debut, FE_Debut, [2.3 , 2.6]) * (FE_ Debut, S_RE_Fin, [**7**, **7**]) => collage0Fin

Diagnostic du Système du convoyeur



Application sur un cas d'étude : convoyeur à bande





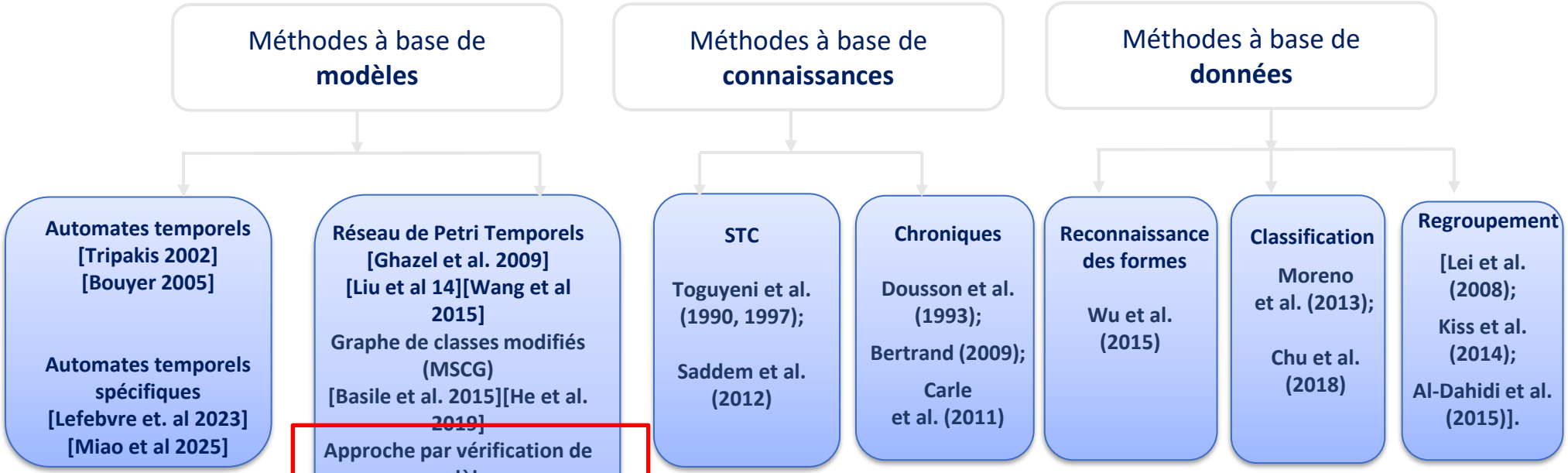
Application sur un cas d'étude : convoyeur à bande

- **STC1**: (A1B2, S1p1s, [3,3]) * (A1B2, p1vp2, [10,12]) * (A1B2, p2e, [16,18]) →(collage, p1s, 0)
- **STC2:** (A1B4, S1p1s, [3,3]) * (A1B4, p1vp4, [12,14]) * (AB1, p4e, [20,22]) → (collage, p1s, 0)
- **STC3:** (A1B2, p1s, [1,3]) * (A1B2, S1p1vp2, [12,12])* (A1B2, S1p2e, [18,18]) → (blocage, A1, P1vP4)
- **STC4**: (A1B4, p1s, [1,3]) * (A1B4, S1p1vp4, [14,14]) * (A1B4, S1p4e, [22, 22]) → (blocage, A1, P1vP2)
- **STC5**: (A1B2, S1p1s, [3,3]) * (A1B2, S1p1vp2, [12,12])) * (A1B2, S1p2e, [18,18]) → (blocage, B1, sortie)
- **STC6**: (A1B4, S1p1s, [3,3]) * (A1B4, S1p1vp4, [14,14])) * (A1B4, S1p4e, [22,22]) → (blocage, B1, sortie)

Diagnostic et diagnosticabilité de motifs dans les réseaux de Petri temporels

Une approche par vérification automatique de modèles

Méthodes de diagnostic des SED temporels et temporisés



- La validation de la cohérence et la complétude des défauts à diagnostiquer.
- Problème d'explosion combinatoire lors de l'implémentation sur des systèmes réels

- Connaissance des symptômes des fautes qu'elles modélisent
- L'acquisition des connaissances expertes et de leur mise à jour

- Ne nécessite pas une modélisation explicite du système
- Apprendre des expériences
- Amélioration des performances
- Une étape de prétraitement des données
- Une grande quantité de données

Motivations générales

- Comment résoudre des problèmes de diagnostic sur des systèmes à événements discrets temporels à l'aide d'outils de vérification de modèles (*model-checking*) ?

- Type de problèmes
  - **Diagnostic** : déterminer si un phénomène spécifique (un défaut, une défaillance) a eu lieu dans le système à partir des observations du système

  - **Diagnosticabilité** : déterminer si un phénomène spécifique pourra toujours être diagnostiqué avec certitude et en temps fini

- Et autres (prédiction, prédictibilité…)

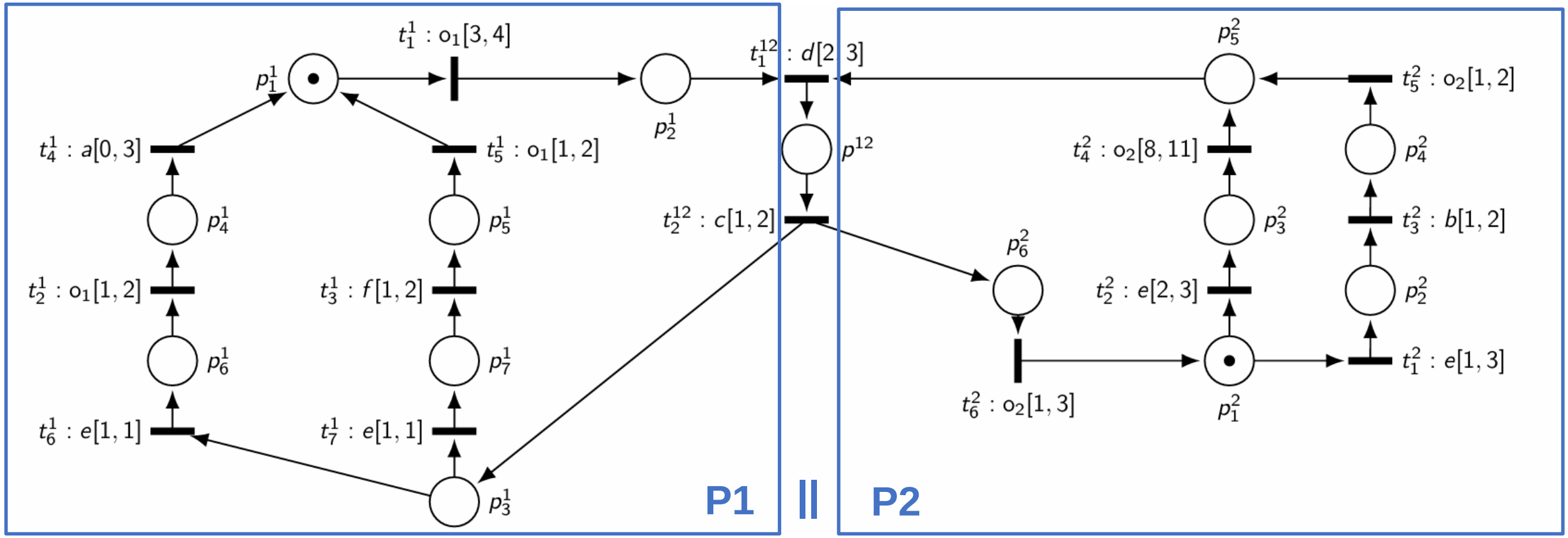
Motivations générales (2)

- Comment résoudre des problèmes de diagnostic sur des systèmes à événements discrets temporels à l'aide d'outils de vérification de modèles (*model-checking*) ?

- Type de systèmes temporels:
  - Réseaux de Petri temporels
  - On supposera qu'ils sont saufs, pas de multi-sensibilisations, intervalles du type [a,b], pas de cycle zénon, pas de systèmes bloquants, pas de cycles non-observables.
  - Vérification automatique basée sur l'exploitation des SCG (Strong Class Graph)

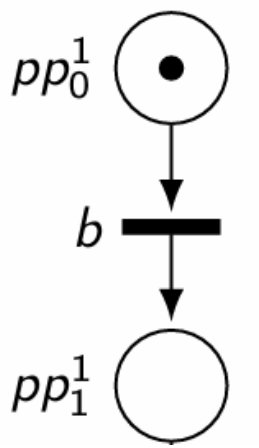
- Type de phénomènes
  - Événement de défaut
  - Motifs de défauts (séquences, ordres partiels)
  - Motifs temporels de défauts

Exemple: deux processus communicants P1||P2



- Processus P1 et P2 communiquent avec les transitions synchronisées t_1^{12} et t_2^{12} et la ressource partagée p^{12}
- Seuls les événements o_1 et o_2 sont observables
- P1 n'émet que des o_1 et P2 que des o_2

Modèles de motifs de défauts par réseau de Petri

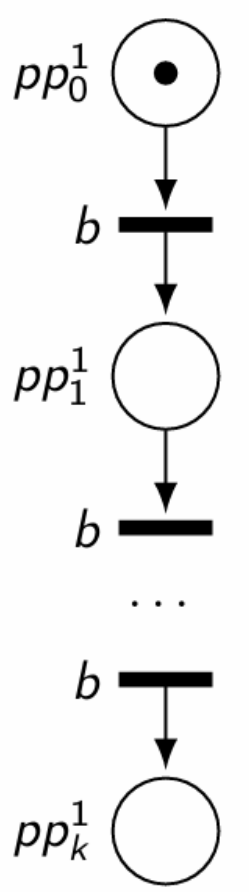


Événement simple $\Omega_1^b(1)$

Une occurrence de b (sur P2) est considérée comme un défaut

Marquages accepteurs

$$Q_{\Omega_1^b(1)} = \{ M : M(pp_1^1) = 1 \}$$



Séquence d'événements $\Omega_1^b(k)$

k occurrences de b sur P2 sont considérées comme un défaut

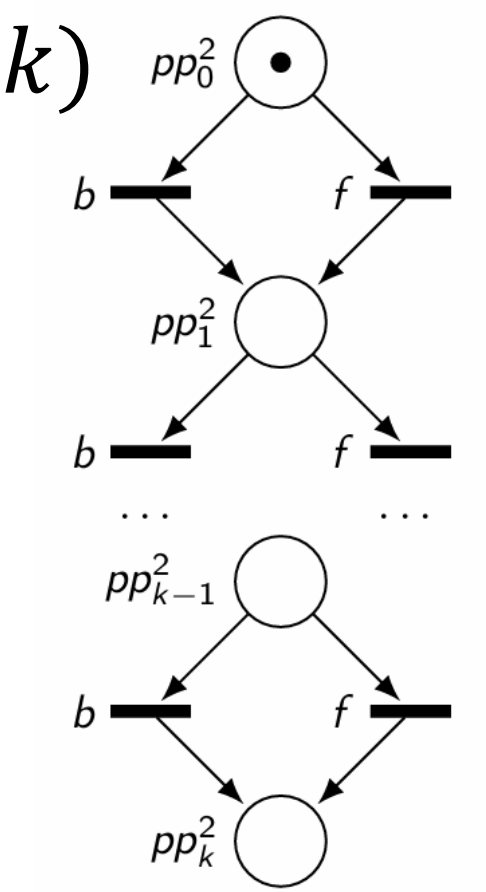
k-1 occurrences de b n'est pas un défaut

Marquages accepteurs

$$Q_{\Omega_1^b(k)} = \{ M : M(pp_k^1) = 1 \}$$

Motifs plus expressifs

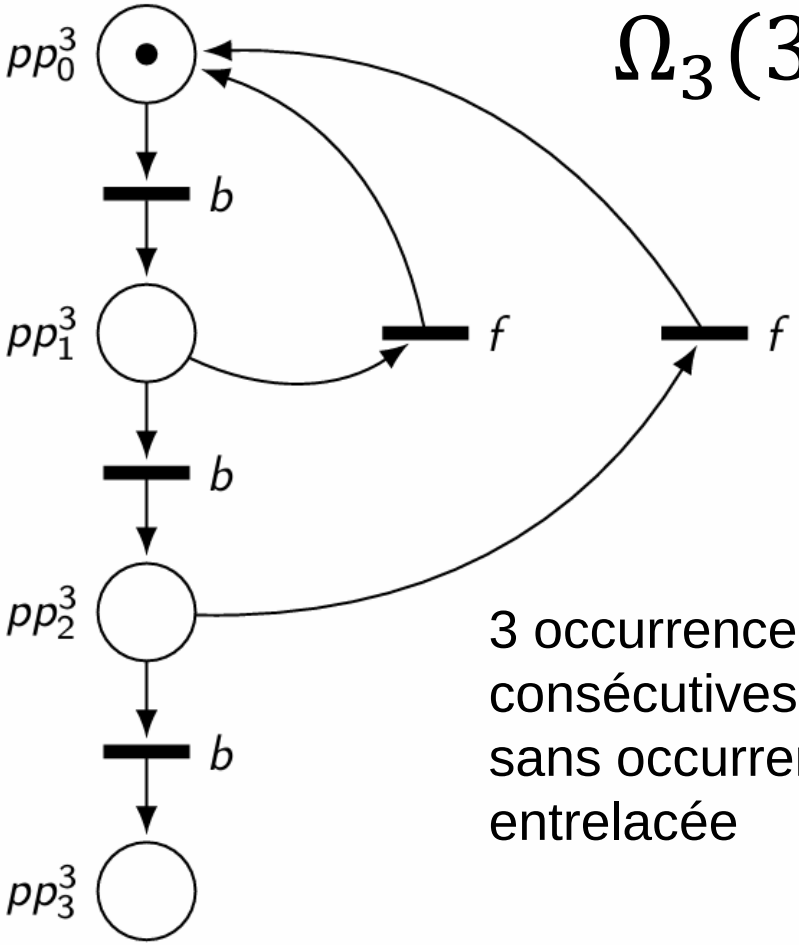
$\Omega_2(k)$



Alternatives
k occurrences de b (sur P2) ou f (sur P1) peu importe le nombre de b et le nombre de f

$$Q_{\Omega_2(k)} = \{ M : M(pp_k^2) = 1 \}$$

$\Omega_3(3)$



3 occurrences consécutives de b sans occurrence de f entrelacée

$$Q_{\Omega_3(3)} = \{ M : M(pp_3^3) = 1 \}$$

Diagnosticabilité temporelle

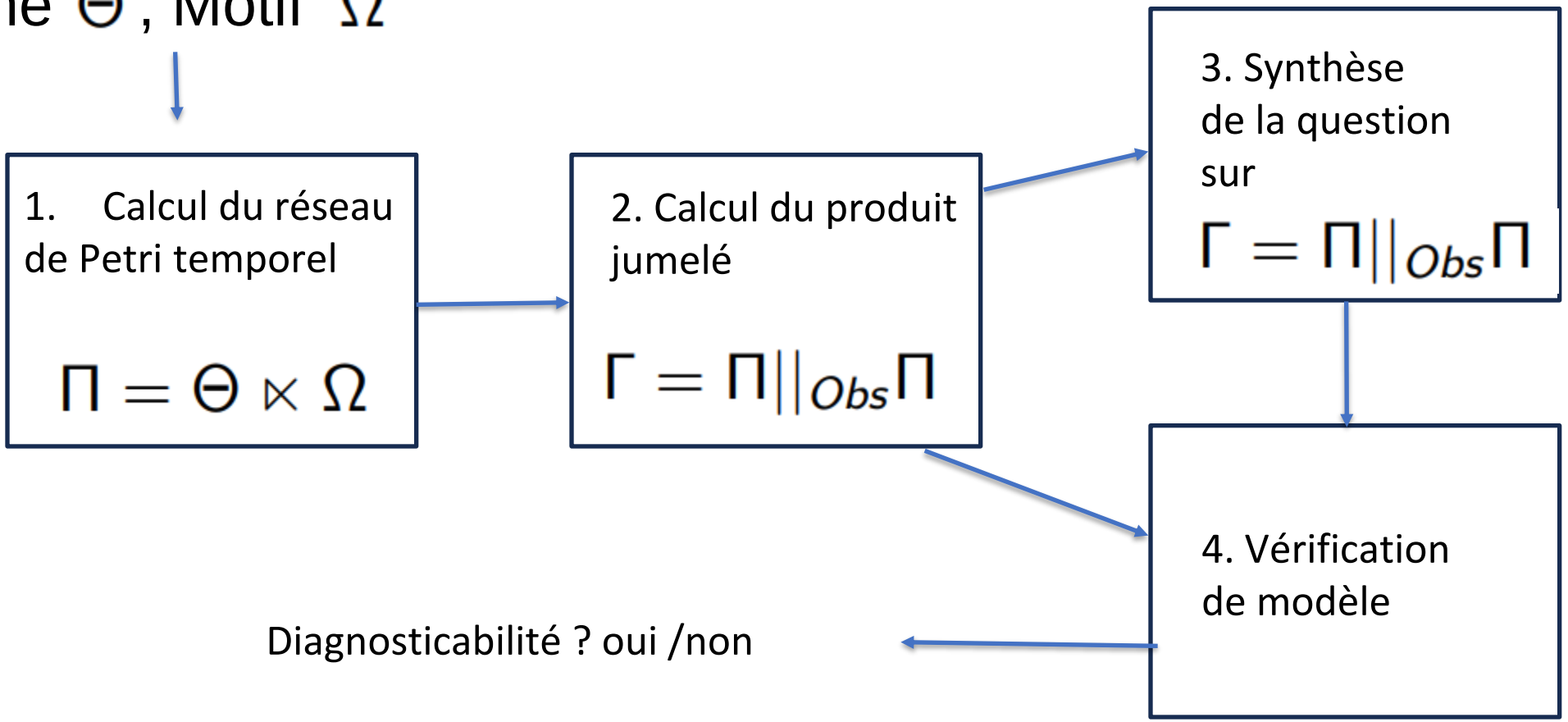
- Un système est Ω-diagnosticable si

$$(\exists \tau \in \mathbb{R}^+), \forall \rho_1, \rho_2 \in \mathcal{L}(\Theta) : \rho_1 = \rho_1' \rho_1'', time(\rho_1'') \geq \tau,$$
$$\rho_1' \ni \Omega \wedge \mathscr{P}_{\Sigma_\Theta \to \Sigma_\Theta^o}(\rho_1) = \mathscr{P}_{\Sigma_\Theta \to \Sigma_\Theta^o}(\rho_2) \implies \rho_2 \ni \Omega.$$

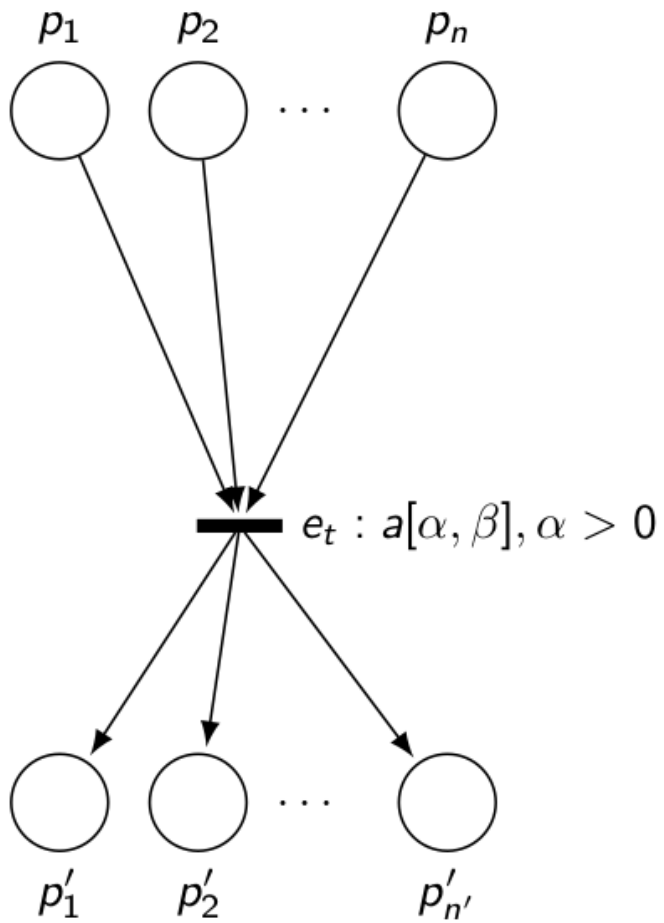
- **Intuition:** il existe toujours une durée finie τ après l'occurrence du motif ($\rho_1' \ni \Omega$) après laquelle on a assez d'observations ($\mathscr{P}_{\Sigma_\Theta \to \Sigma_\Theta^o}(\rho_1)$) pour être certain que le motif a eu lieu ($\rho_2 \ni \Omega$)
- Extension au motif de la diagnosticabilité temporelle de faute simple [Tripakis 2002]

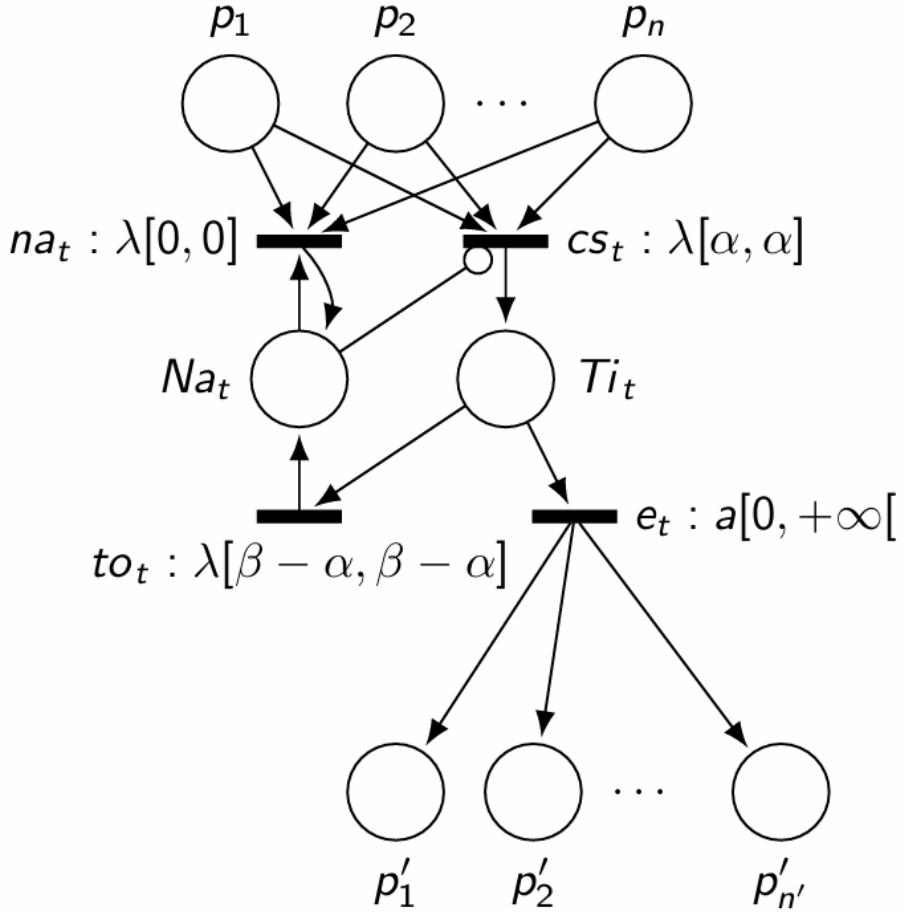
Vérification de la diagnosticabilité temporelle: méthode

Système Θ , Motif Ω



Décomposition temporelle d'une transition t





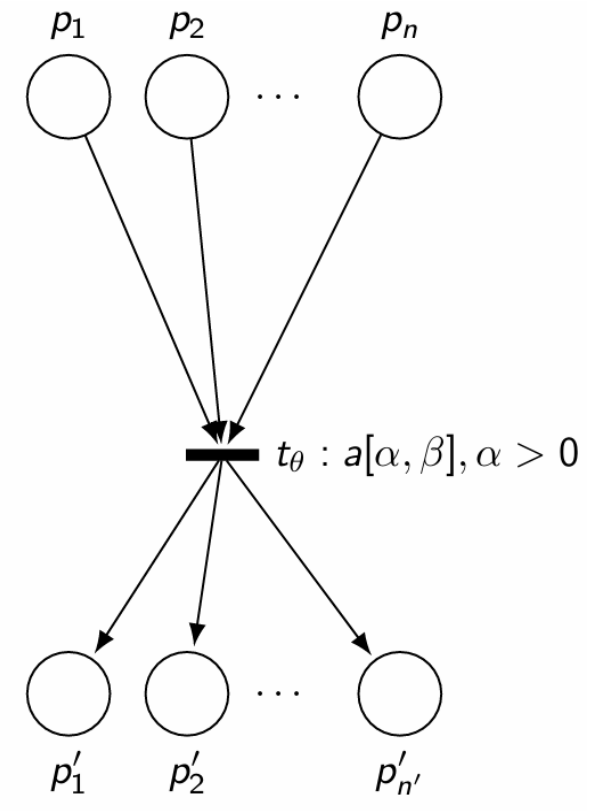
cs : *clock starting*
(démarrage de l'horloge relative à la sensibilisation de t)

to: *timeout*
(date d'expiration de t relative à la sensibilisation de t)

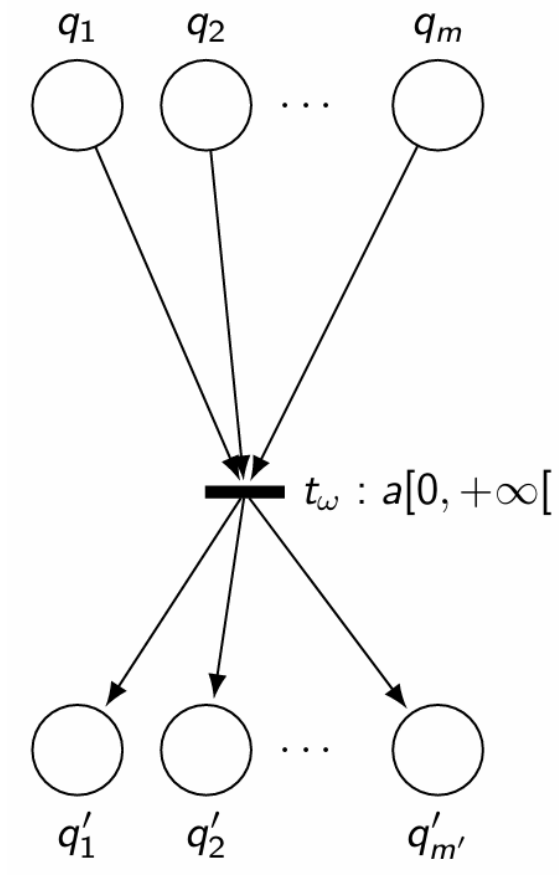
na: *non admissible*
(exprime des tirs de t illégaux)

Produit système/motif

- Comment synchroniser les deux transitions suivantes ?

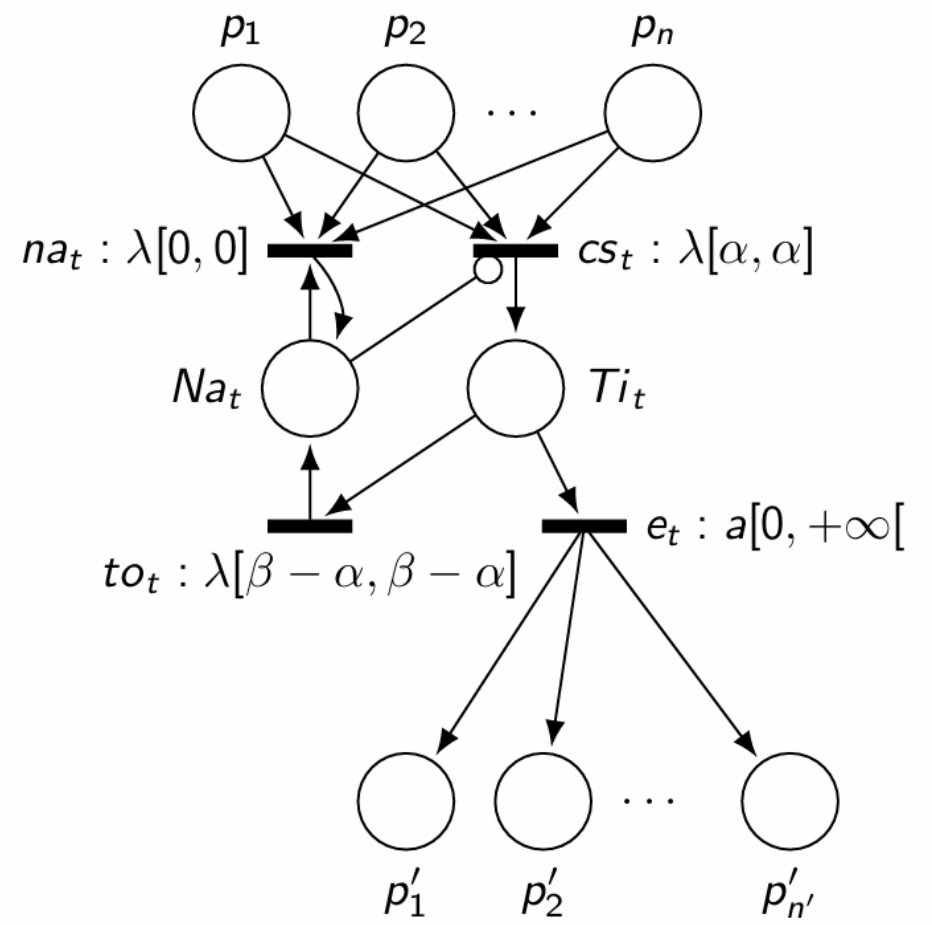


Transition issue du système

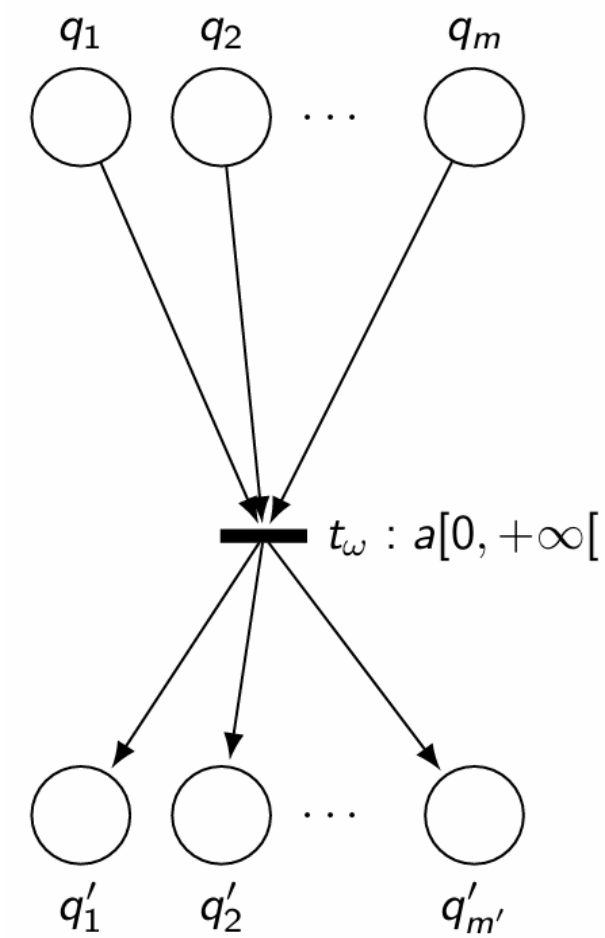


Transition issue du motif

Produit système/motif: étape 1

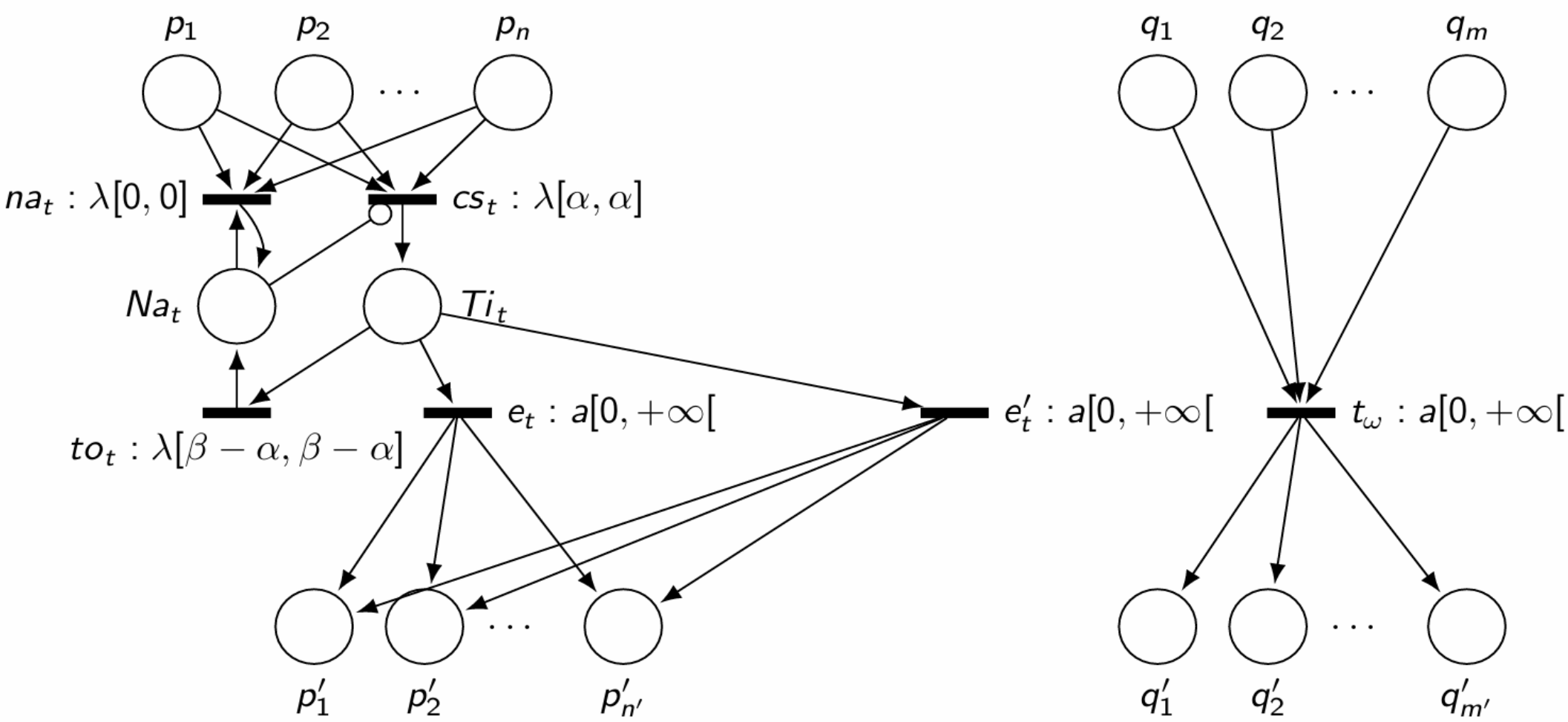


Décomposition temporelle de la transition du système



Transition issue du motif

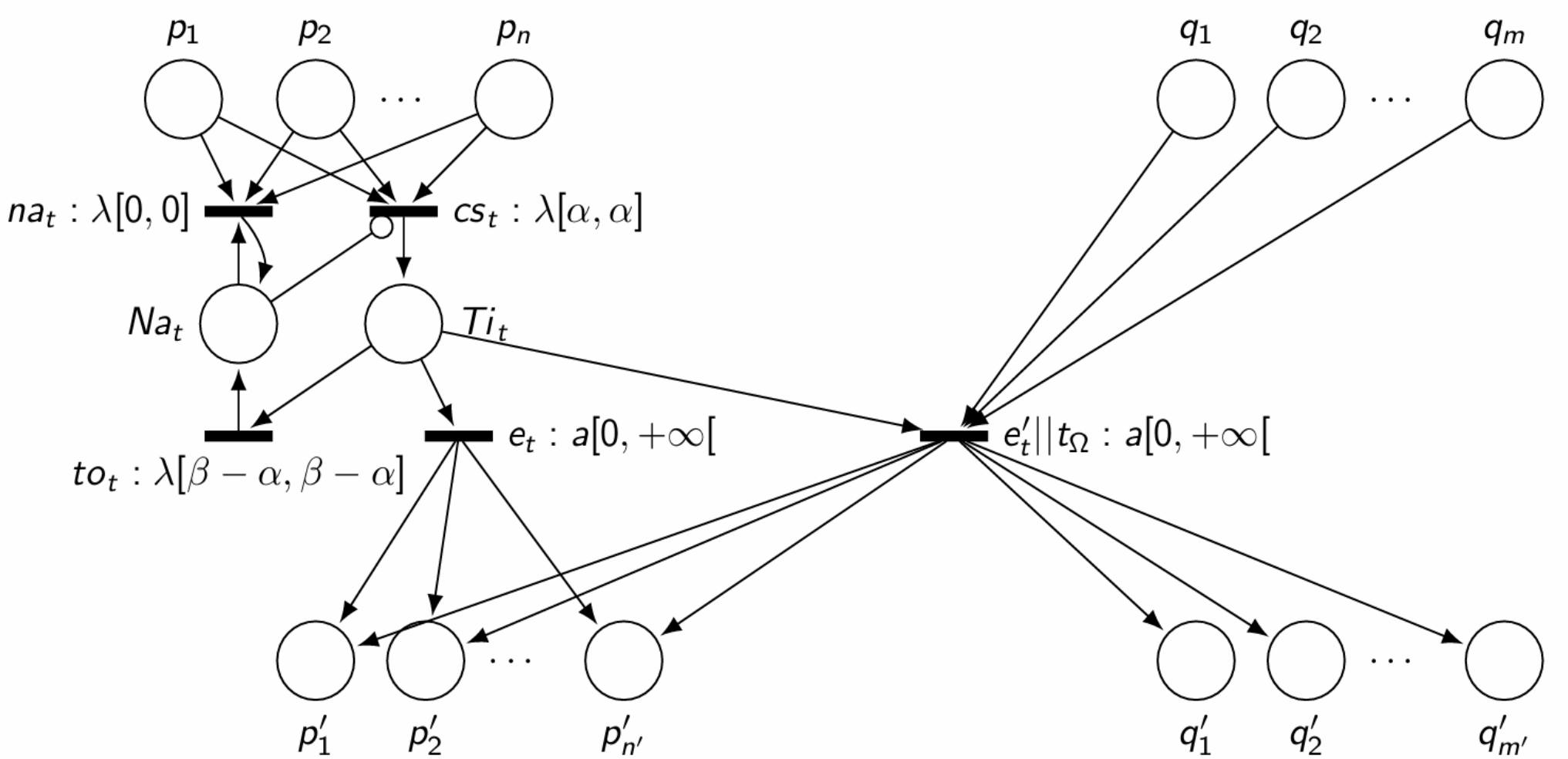
Produit système/motif: étape 2 duplication



Décomposition temporelle de la transition du système

Transition issue du motif

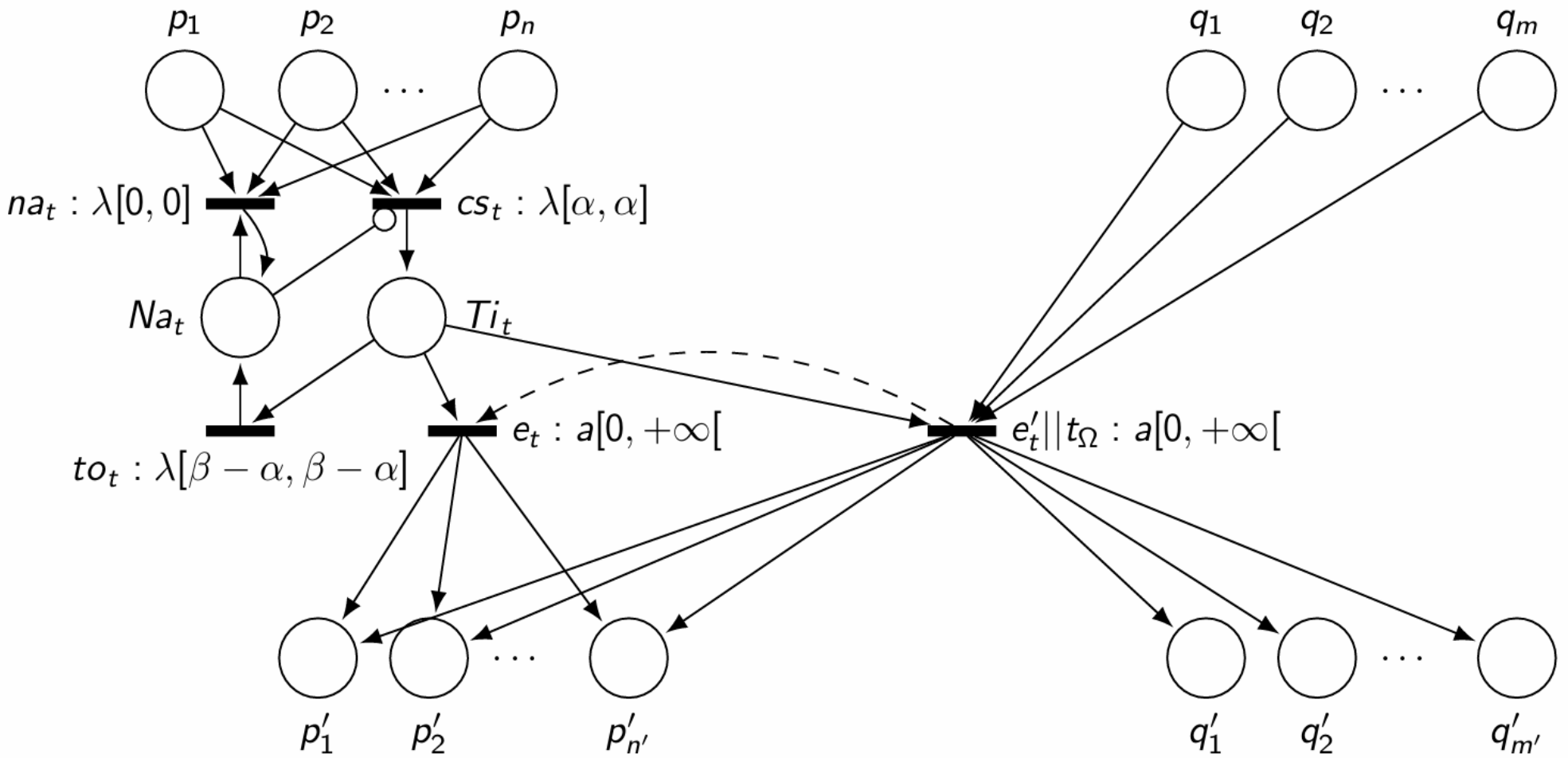
Produit système/motif: étape 3 fusion



Décomposition temporelle de la transition du système

Transition issue du motif

Produit système/motif: étape 4 priorité



Décomposition temporelle de la transition du système

Transition issue du motif

Propriété du produit système/motif

- Non-intrusif (observateur)
- Reconnaissance du motif au plus tôt
- Produit asymétrique:
  - Le comportement du système est synchronisé avec le motif, … ou pas
  - Le motif est toujours synchronisé
- Le langage admissible du produit = ensemble des séquences du système où le motif a eu lieu

$$\mathcal{L}(\Theta \ltimes \Omega, Q^{match}_{\Theta \ltimes \Omega}) = \{\rho \in \mathcal{L}(\Theta) | \rho \ni \Omega\}$$

Synthèse du produit jumelé

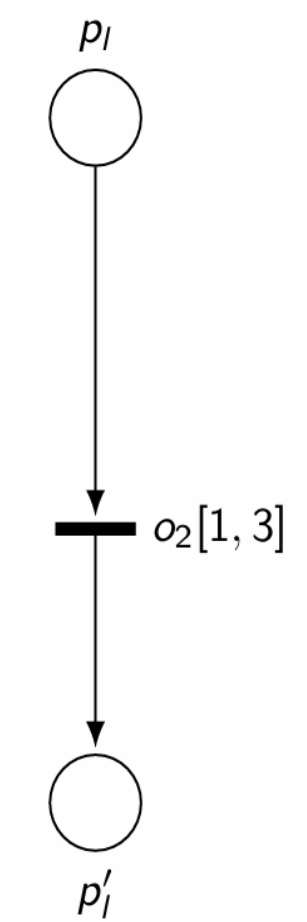
Objectifs du produit jumelé Γ

1. Une séquence du produit jumelé représente un couple de séquences ρ_l ρ_r de $\Theta \ltimes \Omega$ qui produit la même séquence observable
2. Une paire critique est une séquence infinie de Γ telle que:
   1. Le motif se produit dans ρ_l
   2. Le motif ne se produit pas dans ρ_r
3. Diagnosticabilité = Pas de paire critique dans Γ

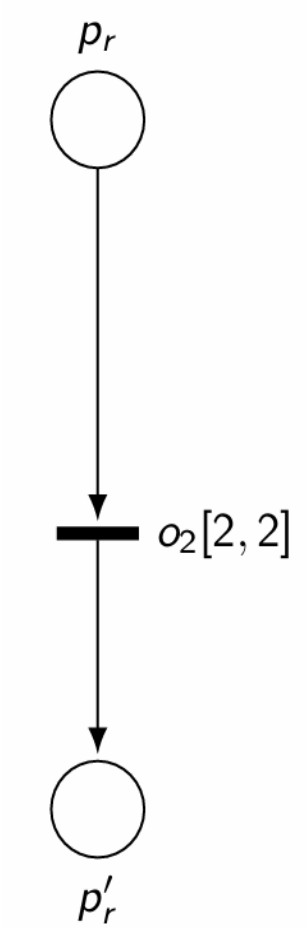
Calcul de Γ par fusion de transitions de deux copies de $\Theta \ltimes \Omega$

- Fusion des transitions observables seulement $\Gamma = \Pi_l ||_{obs} \Pi_r$
- Les transitions non observables sont seulement dupliquées

Fusion de transition: exemple

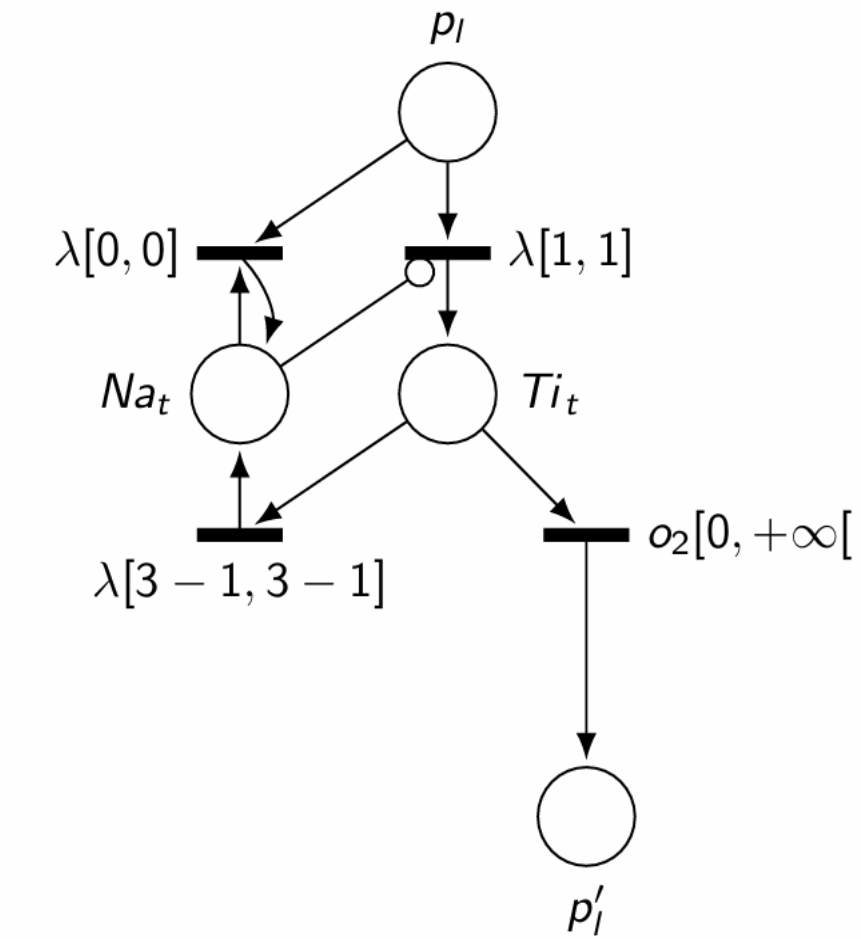


Transition de $\Theta \ltimes \Omega$ (gauche)

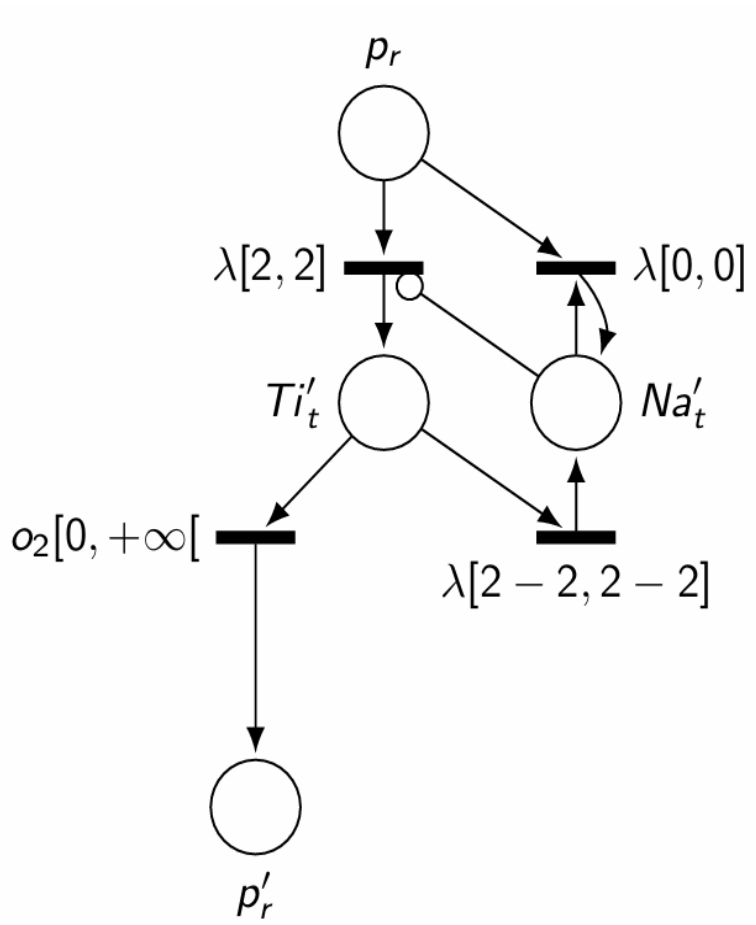


Transition de $\Theta \ltimes \Omega$ (droite)

Fusion de transition: décompositions temporelles

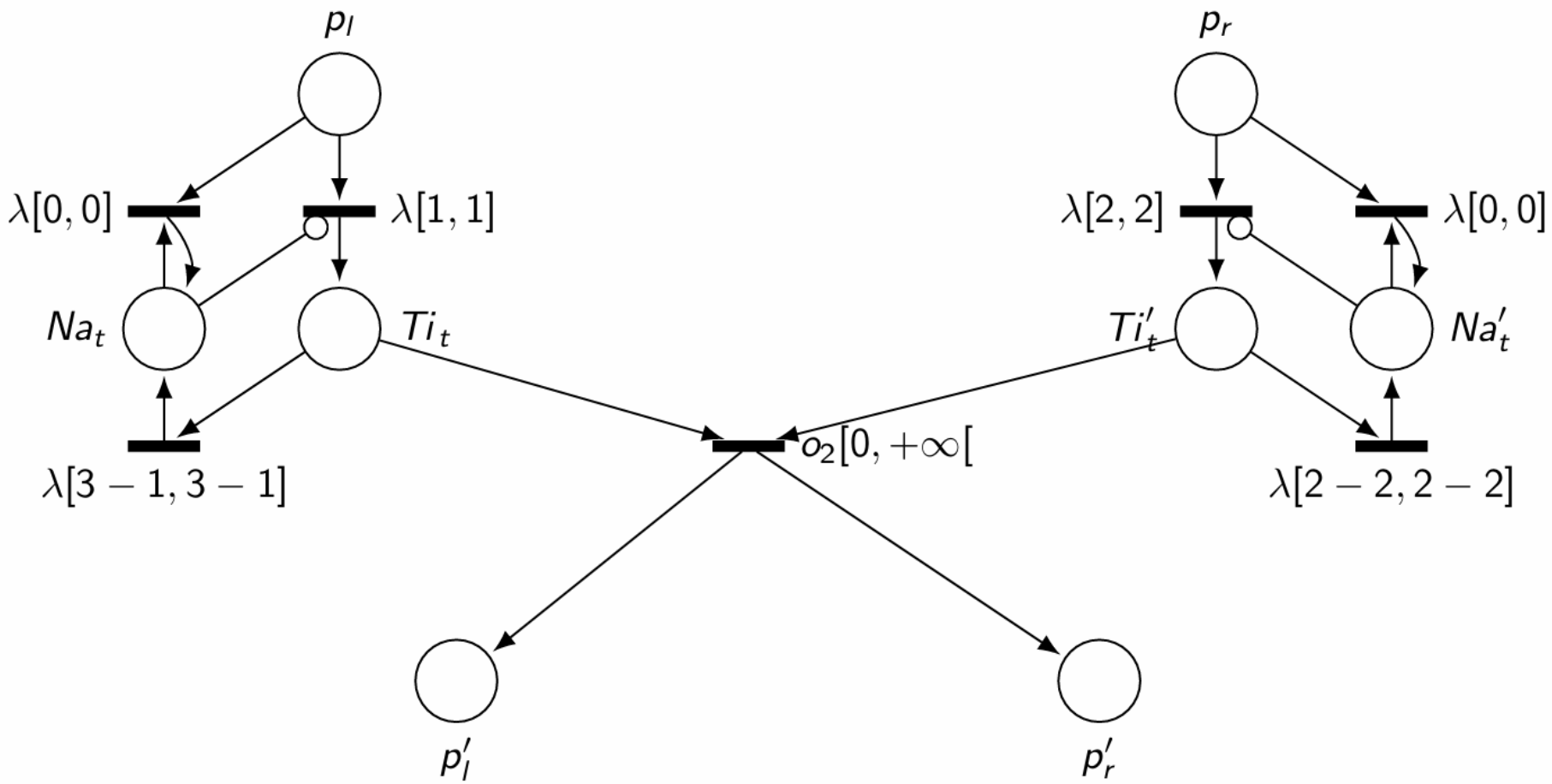


Transition de $\Theta \ltimes \Omega$ (gauche)



Transition de $\Theta \ltimes \Omega$ (droite)

Fusion de transition: synchronisation temporelle



Partie du produit jumelé $\Gamma = \Pi_l ||_{obs} \Pi_r$

Synthèse de la question en SE-LTL

- Logique (State/Event Linear Temporal Logic)

$$\varphi ::= r \mid \neg\varphi \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \bigcirc \varphi \mid \Box\varphi \mid \Diamond\varphi \mid \varphi \mathsf{U} \varphi$$

$$r ::= e \mid e \bigtriangleup e$$

$$e ::= p \mid a \mid c \mid e \bigtriangledown e$$

- p: place, a: transition, c: valeur, $\bigtriangleup \in \{=, <, >, \leq, \geq\}$ $\bigtriangledown \in \{+, -, *, /\}$
- Opérateurs: $\bigcirc$ (next) $\Box$ (always) $\Diamond$ (eventually) U (until)

SE-LTL: vérification sur les marquages

- Exemple: propriété sur un ensemble de marquages

$$\text{Markings}(N, Q) \equiv (p_1 = 1 \wedge p_2 = 1)$$

N: réseau de Petri temporel

Q: ensemble des marquages M de N tel que M(p1)=1 et M(p2)=1.

- Exemple: toute exécution de N mène par une continuation finie à un marquage de Q

$$N \models \Box\Diamond\text{Markings}(N, Q)$$

Vérifier qu'un marquage du produit est admissible

- Marquage admissible = pas de jetons dans les places Na

$$\mathrm{Adm}(\Pi_i) \equiv \bigwedge_{Na\Theta \in \Pi_i} Na\Theta = 0.$$

- Marquage admissible de $\Gamma = \Pi_l ||_{obs} \Pi_r$

$$\mathrm{Adm}(\Gamma) \equiv \mathrm{Adm}(\Pi_l) \wedge \mathrm{Adm}(\Pi_r).$$

Vérifier les occurrences du motif

- Pour que le motif ait lieu dans une exécution de Π_i
  1. Elle doit être admissible
  2. Elle doit atteindre un marquage incluant un marquage accepteur du motif: Q_Ω

$$\text{Match}(\Pi_i) \equiv \text{Adm}(\Pi_i) \wedge \text{Markings}(\Omega, Q_\Omega).$$

- Pour que le motif n'ait pas lieu dans une exécution de Π_i
  1. Elle doit être admissible
  2. Elle doit atteindre un marquage n'incluant pas un marquage accepteur du motif: Q_Ω

$$\text{NoMatch}(\Pi_i) \equiv \text{Adm}(\Pi_i) \wedge \neg\text{Markings}(\Omega, Q_\Omega).$$

- Il y a ambiguïté dans une exécution du produit jumelé si:

$$\text{Ambiguous}(\Gamma) = \text{Match}(\Pi_l) \wedge \text{NoMatch}(\Pi_r).$$

Vérifier la diagnosticabilité du motif

- Système diagnosticable = pas de paire critique

1. Il existe toujours une continuation finie de tout exécution ambiguë qui ne l'est plus

$$\Box(\text{Ambiguous}(\Gamma) \Rightarrow \Diamond(\text{Adm}(\Gamma) \Rightarrow \text{Match}(\Pi_r)))$$

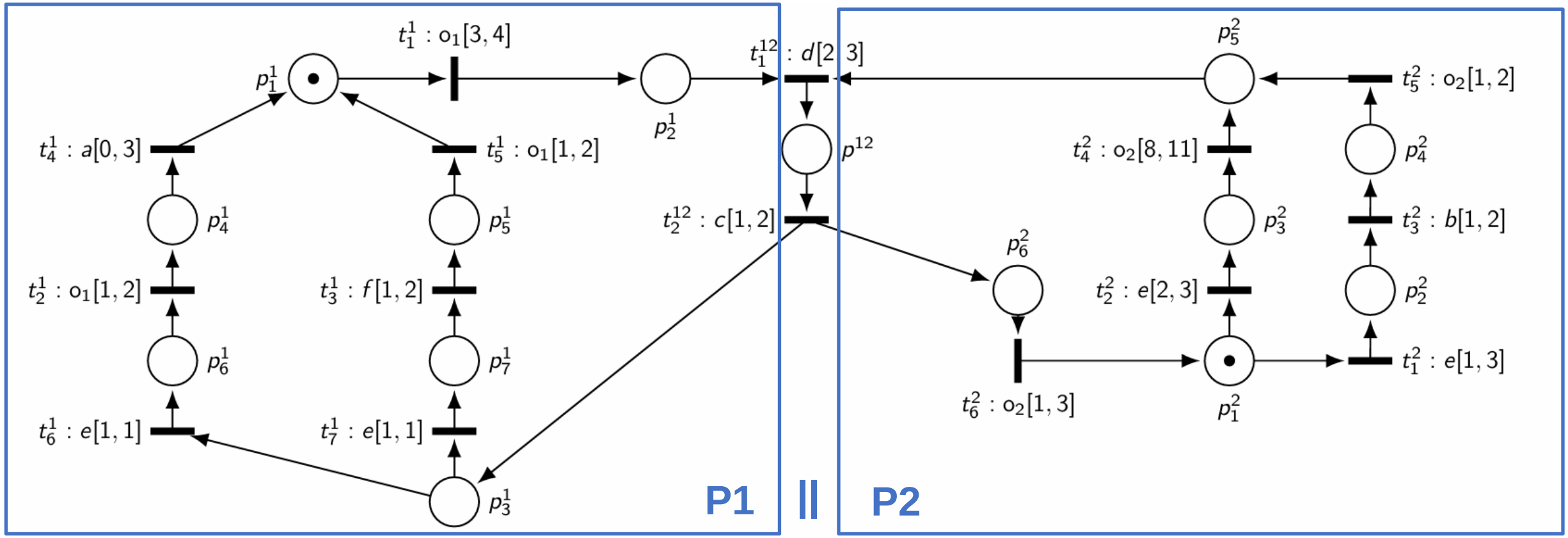
2. Ou une continuation finie bloquante

$$\Box(\text{Ambiguous}(\Gamma) \Rightarrow \Diamond(\text{dead}))$$

D'où la question de diagnosticabilité finale:

$$\varphi = \text{Diagnosable}(\Theta, \Omega) = \Box(\text{Ambiguous}(\Gamma) \Rightarrow \Diamond(\text{Adm}(\Gamma) \Rightarrow (\text{Match}(\Pi_r) \vee \text{dead}))$$

Exemple: deux processus communicants P1||P2



- Processus P1 et P2 communiquent avec les transitions synchronisées t_1^{12} et t_2^{12} et la ressource partagée p^{12}
- Seuls les événements o_1 et o_2 sont observables
- P1 n'émet que des o_1 et P2 que des o_2

Quelques exemples de résultats

Case	Pattern	System	Result
1	$\Omega_1^b(1)$: one event b	Θ	OK
2	$\Omega_1^b(2)$: two events b	Θ	OK
3	$\Omega_1^b(10)$: ten events b	Θ	OK
4	$\Omega_1^f(1)$: one event f	Θ	KO
5	$\Omega_2(1)$: one event f or one event b	Θ	KO
6	$\Omega_2(1)$: one event f or one event b	$\Theta : a[0,3] \rightarrow a[2,3]$	OK
7	$\Omega_3(4)$: 4 consecutive events b without an f	$\Theta : a[0,3] \rightarrow a[2,3]$	OK
8	$\Omega_3(4)$: 4 consecutive events b without an f	Θ	KO

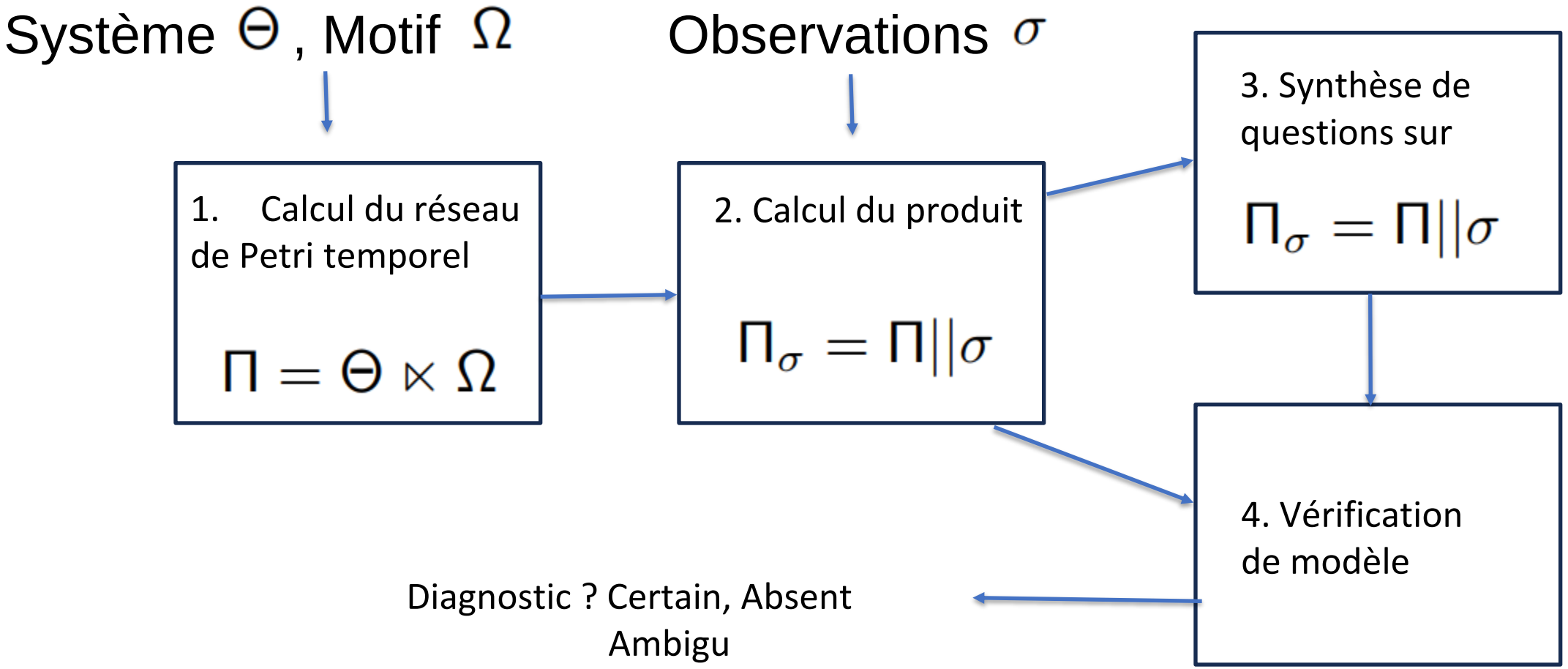
Diagnostic: quelques exemples

- Après l'étude de diagnosticabilité, passons maintenant au diagnostic
- Supposons que l'on observe:

$$\sigma = 3o_1 2o_2 8o_2 2o_1 1\lambda$$

- Le diagnostic est:
  - Un événement b a certainement eu lieu: $\Omega_1^b(1) - faulty$
  - L'événement b a pu avoir lieu deux fois, mais pas sûr: $\Omega_1^b(2) - ambiguous$
  - Deux occurrences parmi {b,f} ont eu lieu: $\Omega_2(2) - faulty$
  - En aucun cas, on a eu l'occurrence de 3 b consécutifs sans l'occurrence d'un f: $\Omega_3(3) - safe$

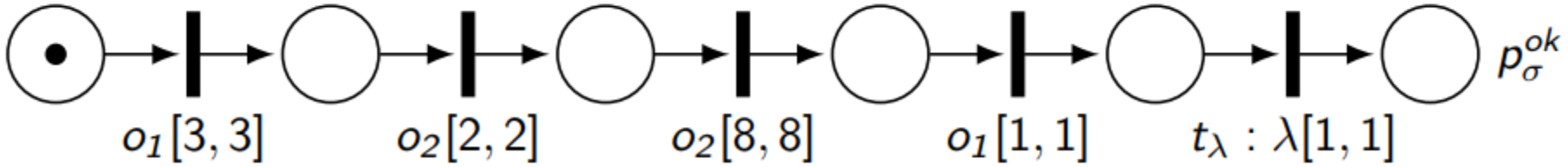
Principe de la méthode de diagnostic



Représentation des observations datées

$\sigma = 3o_1 2o_2 8o_2 1o_1 1\lambda.$

$\xi_\sigma =$



λ n' est pas un événement observable, c'est un point temporel. La durée effective de l'observation est de 15 unités de temps même si le dernier événement produit est la date 14.

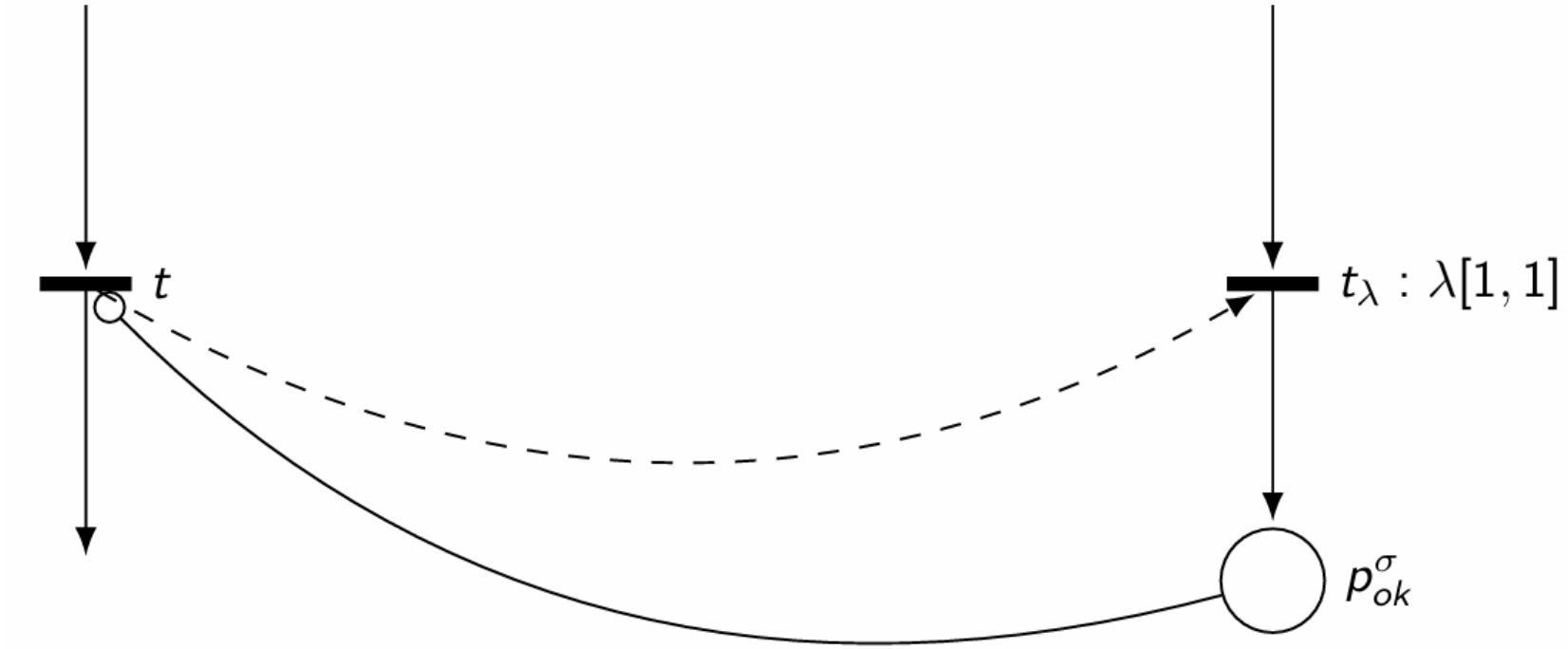
Composition des observations et du système

- Pour connaître le diagnostic d'un motif Ω sur le système Θ ayant produit les observations σ
- On va composer (synchroniser) ξ_σ avec $\Theta \ltimes \Omega$ -> $\Pi_\sigma^{\Theta,\Omega}$
- Objectif de la composition $\Pi_\sigma^{\Theta,\Omega}$

  Extraire de $\Theta \ltimes \Omega$ les exécutions dont la séquence observable associée est σ avec la même durée

  1. Désactivation de toutes transitions observables dans $\Theta \ltimes \Omega$ qui ne peut pas émettre un événement observable de σ , (ajout de places non admissible Na)
  2. Synchroniser les transitions de ξ_σ avec les transitions observables de $\Theta \ltimes \Omega$ émettant les mêmes événements.
  3. Ajout de priorités et arcs inhibiteurs dans la composition pour bloquer le temps sur la durée de σ

Comment bloquer le temps?



- Au temps 1 de sensibilisation de t_λ
  - Seule t_λ peut être tirée (priorité)
  - plus aucun tir de transitions t dans le système n'est autorisé après t_λ (inhibition)

Résumé de la construction de $\prod_{\sigma}^{\Theta,\Omega}$

1. On considère un système Θ et un motif Ω et la séquence observée σ
2. Calcul du produit asymétrique $\Theta \ltimes \Omega$
3. Calcul de $\prod_{\sigma}^{\Theta,\Omega}$ par fusion de transition avec ξ_{σ} (idem produit jumelé)

Propriété de $\prod_{\sigma}^{\Theta,\Omega}$: toute exécution de Θ produisant σ et de même durée que σ est associée à une exécution de $\prod_{\sigma}^{\Theta,\Omega}$ atteignant le marquage

$M(p_{\sigma}^{ok}) = 1$ (et réciproquement)

Vérifier la cohérence avec les observations

- Une exécution du système Θ est cohérente avec les observations ssi il existe une exécution de $\Pi_\sigma^{\Theta,\Omega}$ telle que
  1. Elle est admissible
  2. Elle atteint un marquage tel que $M(p_\sigma^{ok}) = 1$

$$\text{Consistent}(\Pi_\sigma^{\Theta,\Omega}) \equiv$$
$$\text{Adm}(\Pi_\sigma^{\Theta,\Omega}) \wedge \text{Markings}(\Omega, \{p_\sigma^{ok}\})$$

Questions de diagnostic

- Question 1: le motif a-t-il eu lieu dans toutes les exécutions possibles du système cohérentes avec les observations ?

$$\mathsf{Faulty}(\Theta, \Omega, \sigma) \equiv$$

$$\Box(\mathsf{Consistent}(\Pi_\sigma^{\Theta,\Omega}) \Rightarrow \mathsf{Match}(\Pi_\sigma^{\Theta,\Omega}))$$

- Question 2: le motif n'a-t-il jamais eu lieu dans toutes les exécutions possibles du système cohérentes avec les observations ?

$$\mathsf{Safe}(\Theta, \Omega, \sigma) \equiv$$

$$\Box(\mathsf{Consistent}(\Pi_\sigma^{\Theta,\Omega}) \Rightarrow \mathsf{NoMatch}(\Pi_\sigma^{\Theta,\Omega}))$$

Résumé de la méthode de diagnostic

- Problème 1, vérifier:

$$\Pi_{\sigma}^{\Theta,\Omega} \models \mathrm{Faulty}(\Theta, \Omega, \sigma)$$

  - Si vrai, le motif a certainement eu lieu. FIN.
  - Si faux, résoudre le problème 2
- Problème 2, vérifier:

$$\Pi_{\sigma}^{\Theta,\Omega} \models \mathrm{Safe}(\Theta, \Omega, \sigma)$$

  - Si vrai, le motif n'a certainement pas eu lieu. FIN
  - Si faux, le motif a pu avoir lieu. FIN

Quelques résultats

	pl $\Pi_\sigma^{\Theta,\Omega}$	tr $\Pi_\sigma^{\Theta,\Omega}$	arcs $\Pi_\sigma^{\Theta,\Omega}$	prio $\Pi_\sigma^{\Theta,\Omega}$	st SSG	tr SSG	Faulty	Safe	time
$\Omega_1^b(1)$	44	56	217	57	273	456	T	F	45ms
$\Omega_1^b(2)$	45	57	222	60	269	451	F	F	49ms
$\Omega_2(2)$	47	62	243	69	273	456	T	F	47ms
$\Omega_3(3)$	48	63	248	72	270	452	F	T	49ms

Logiciel Diades en collaboration avec le model-checker TINA

https://gitlab.laas.fr/ypencole/diades

Conclusions

Conclusions

- Diagnostic temporel
  - Etude de l'impact du temps et de son observation pour une meilleure discrimination
- Présentation de deux types de méthodes:
  - Méthode à base de connaissance: Signature Temporelle Causale
  - Méthode à base de modèles: Approche par Model-Checking [SCG]
- D'autres travaux existent:
  - Souvent basé sur des abstractions du temps:
    - approche MSCG [Basile et al.] basé sur une modification du SCG pour mieux caractériser les exécutions en cohérence avec les observations
    - Graphe de fautes temporels
    - Abstraction en temps séquentiel de certains types d'automates temporels

Références bibliographiques

Références bibliographiques
Méthodes à base de modèles

Automates temporels:

- Tripakis (2002). *Fault Diagnosis for Timed Automata.* Formal Techniques in Real-Time and Fault-Tolerant Systems.
- Bouyer, Chevalier, D'Souza (2005). *Fault diagnosis using timed automata* ». In : International Conference on Foundations of Software Science and Computation Structures.
- Lefebvre, Li et Liang (2023). *Diagnosis of timed patterns for discrete event systems by means of state isolation*. Automatica
- Miao,Lai,Komenda,Lahaye (2025). *Decentralized Fault Diagnosis for Constant-Time Automata.* IEEE Control Systems Letters

Réseaux de Petri Temporels:

- Ghazel, Toguyéni, Yim (2009). *State observer for DES under partial observation with time Petri nets*. In : Discrete Event Dynamic Systems.
- Liu, Ghazel, Toguyéni (2014). *Diagnosis of labeled time Petri nets using time interval splitting*. In : IFAC Proceedings Volumes.
- Wang, Mahulea, Silva (2015). *Diagnosis of Time Petri Nets Using Fault Diagnosis Graph*. IEEE Transactions on Automatic Control.
- Basile, Cabasino, Seatzu (2015). *State Estimation and Fault Diagnosis of Labeled Time Petri Net Systems With Unobservable Transitions*. IEEE Transactions on Automatic Control.
- Basile, Cabasino, Seatzu (2017). *Diagnosability Analysis of Labeled Time Petri Net Systems*. In : IEEE Transactions on Automatic Control
- Lubat, Dal Zilio, Le Botlan, Pencolé, Subias (2020). *A New Product Construction for the Diagnosability of Patterns in Time Petri Net*. 59th IEEE Conference on Decision and Control.
- Pencolé, Subias (2021). *Diagnosability of event patterns in safe labeled time Petri nets : A model-checking approach* ». IEEE Transactions on Automation Science and Engineering.
- Coquand, Subias et Pencolé (2022). *Observable Simple Temporal Network synthesis for the diagnosis of time patterns in time Petri nets* ». 11th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes.

Approches algébriques (max,+)

- Le Corronc, Pencolé, Sahuguède, Paya (2021). *Failure detection and localization for timed event graphs in (max,+)-algebra*. Discrete Event Dynamic Systems.
- Velasquez, Pencolé, Le Corronc (2024). *Analysis and control of timed event graphs in (max,+) algebra for the active localization of time failures.* Discrete Event Dynamic Systems

Remerciements et crédits

Équipe pédagogique

Auteur.rice.s : Ramla Saddem, Yannick Pencolé

Intervenant.e.s : Ramla Saddem, Yannick Pencolé

Crédits