Diagnosticabilité et Diagnosticabilisation de motifs temporels dans les réseaux de Petri temporels

Camille Coquand

4 Décembre 2023

Directrice : Audine Subias

Co-encadrant : Yannick Pencolé

Outline

- Réseau de Petri Temporel (RdPT)
- Motifs temporels : extension des *motifs de supervision*, événements contraints temporellement entre eux
- Vérification de propriétés de diagnostic :
  - $\rightarrow$ Diagnosticabilité
  - $\rightarrow$ Diagnosticabilisation

Diagnosticabilité et Diagnosticabilisation

Diagnostic → Identifier l'occurrence d'une faute

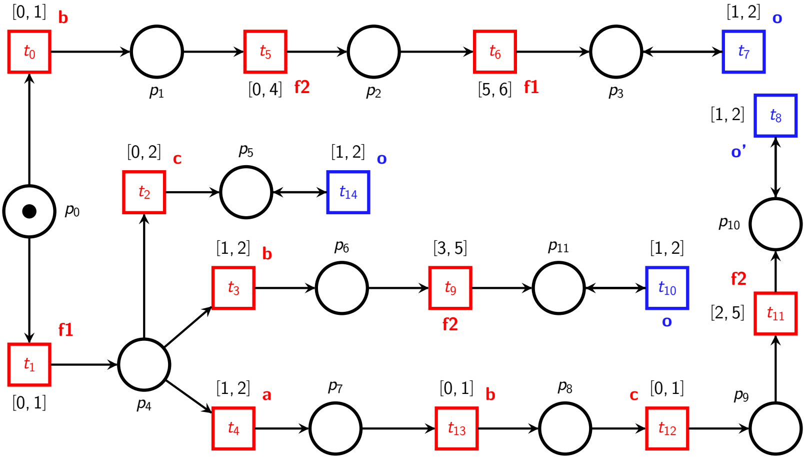
Pour une séquence d'observations :

- certain → il est certain que la faute a eu lieu
- sauf → il est certain que la faute n'a pas eu lieu
- ambigu → il n'est pas possible de conclure

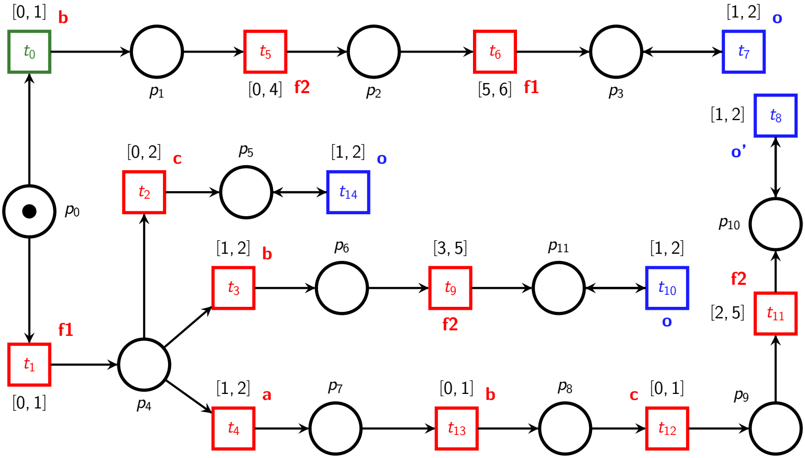
Diagnosticabilité → propriété d'un système pour une faute donnée de pouvoir, avec suffisament d'observations, différencier les trajectoires fautives des trajectoires nominales

Diagnosticabilisation → rendre un système diagnosticable pour une faute donnée en modifiant le système étudié

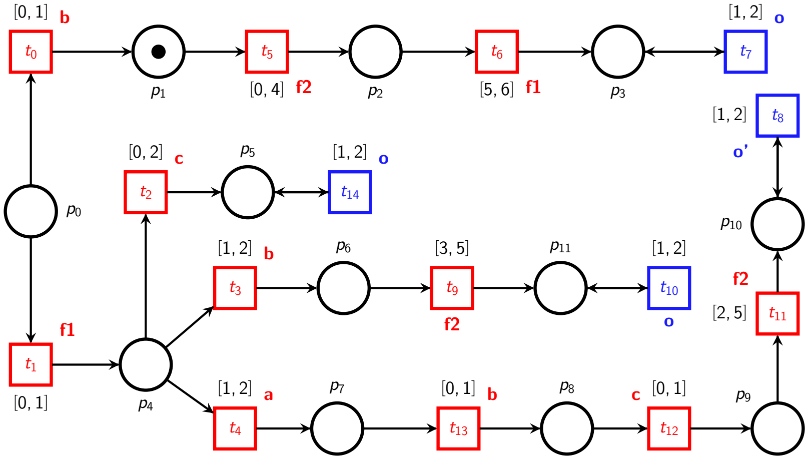
Réseau de Petri temporel



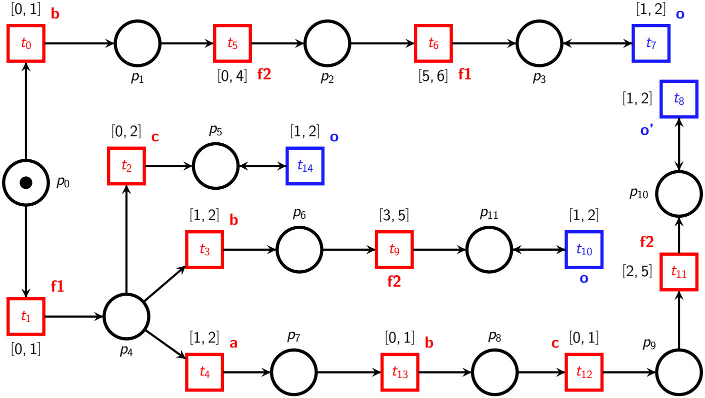
Réseau de Petri temporel



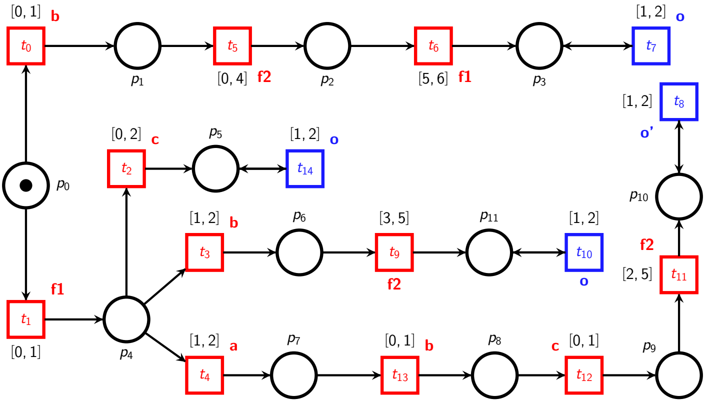
Réseau de Petri temporel



Formalisation des trajectoires d'un RdP temporel

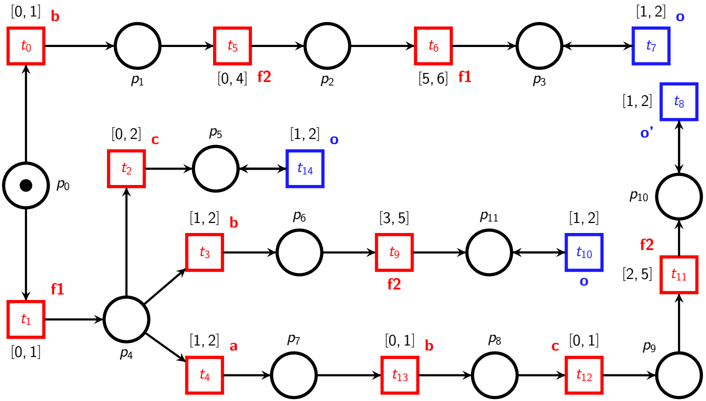


Formalisation des trajectoires d'un RdP temporel



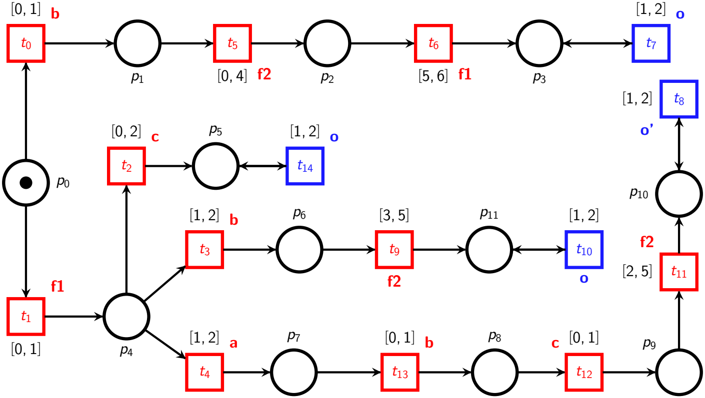
Exécution : $0t_0.2t_5.5t_6.2t_7.1t_7$

Formalisation des trajectoires d'un RdP temporel



<u>Exécution :</u> $0t_0.2t_5.5t_6.2t_7.1t_7$ $\rightarrow$ <u>Trace :</u> $0b.2f_2.5f_1.2o.1o$

Formalisation des trajectoires d'un RdP temporel



Exécution : $0t_0.2t_5.5t_6.2t_7.1t_7 \rightarrow$ Trace : $0b.2f_2.5f_1.2o.1o$
Projection observable : $9o.1o$

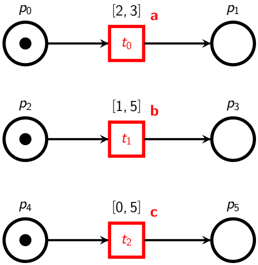
Hypothèses

A0 Intervalles statiques fermés bornés $\forall t \in T^{\Theta}$, $I_s^{\Theta}(t) = [a, b]$, où $(a, b) \in \mathbb{Q}_+ \times (\mathbb{Q}_+^* \setminus \{+\infty\})$ avec $a < b$

A1 Pas de boucle non observable, pas de marquage bloquant

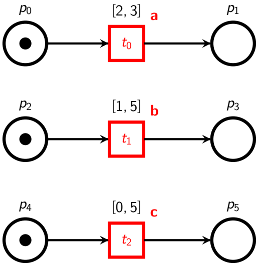
A2 Pas d'exécution *zenon*

Motif de faute temporel



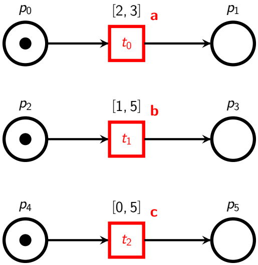
- Acyclique

Motif de faute temporel



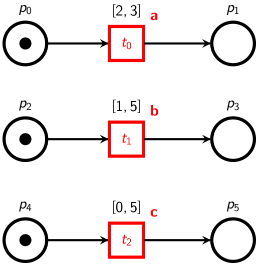
- Acyclique
- Non observable

Motif de faute temporel



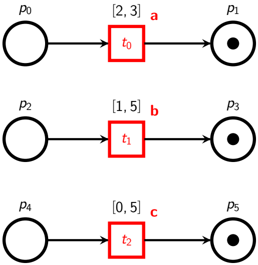
- Acyclique
- Non observable
- Intervalles fermés bornés $\rightarrow$ excution en temps borné

Motif de faute temporel



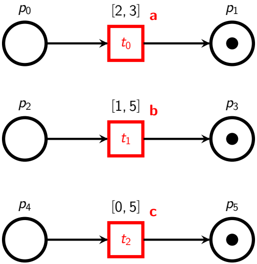
- Acyclique
- Non observable
- Intervalles fermés bornés $\rightarrow$ excution en temps borné
- Ensemble de marquages finaux

Motif de faute temporel



- Acyclique
- Non observable
- Intervalles fermés bornés $\rightarrow$ excution en temps borné
- Ensemble de marquages finaux

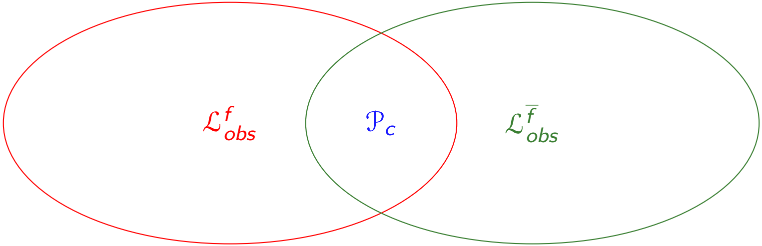
Motif de faute temporel



- Acyclique
- Non observable
- Intervalles fermés bornés $\rightarrow$ excution en temps borné
- Ensemble de marquages finaux
- Marquage initial n'est pas un marquage final

Paire critique [PCC02]

Étudier la diagnosticabilité $\rightarrow$ Étude de l'intersection des langages observables fautifs et non fautifs



Paire critique : paire d'exécutions fautive/non fautive infinies présentant la même trace observable

Système diagnosticable $\Leftrightarrow \mathcal{P}_c = \emptyset$

Diagnosticabilité : méthodes basées graphe

Atemporel :

- Recherche de cycles indéterminés (Diagnostiqueur)
- Recherche de paires critiques (Twin-plant, Verifier)

RdPT :

- Recherche de cycles indéterminés (Diagnostiqueur)
- Recherche de paires critiques (Twin-plant)

Atemporel + Motifs :

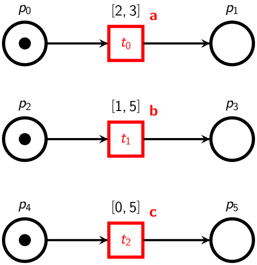
- Recherche de paires critiques (produit synchrone $\rightarrow$ Twin-plant)

RdPT + Motifs temporels :

Pas de produit synchrone

- RdPT Produit

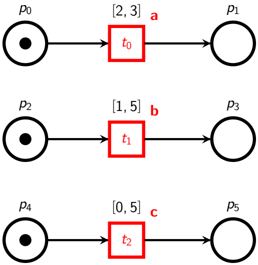
Occurrence d'un motif



Matching : le motif a t-il eu lieu ?

$\rho = 1f_1.1a.0b.0c.3f_2.2o'.1o'$

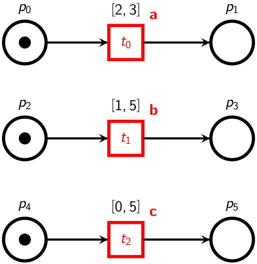
Occurrence d'un motif



Matching : le motif a t-il eu lieu ?

$\rho = 1f_1.1a.0b.0c.3f_2.2o'.1o'$

Occurrence d'un motif

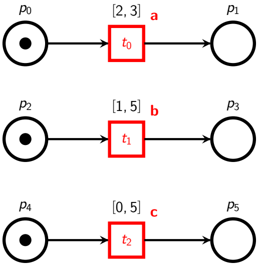


Matching : le motif a t-il eu lieu ?

$\rho = 1f_1.1a.0b.0c.3f_2.2o'.1o'$

$\rho_\Omega = 2a.0b.0c$

Occurrence d'un motif



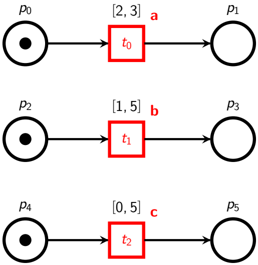
Matching : le motif a t-il eu lieu ?

$\rho = 1f_1.1a.0b.0c.3f_2.2o'.1o'$

$\rho_\Omega = 2a.0b.0c$

Temps continu

Occurrence d'un motif



Matching : le motif a t-il eu lieu ?

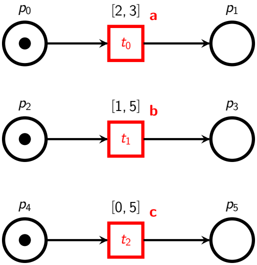
$\rho = 1f_1.1a.0b.0c.3f_2.2o'.1o'$

$\rho_\Omega = 2a.0b.0c$

Temps continu

$\rho' = 1f_1.1a.0.1b.0c.3f_2.2o'.1o'$

Occurrence d'un motif



Matching : le motif a t-il eu lieu ?

$\rho = 1f_1.1a.0b.0c.3f_2.2o'.1o'$

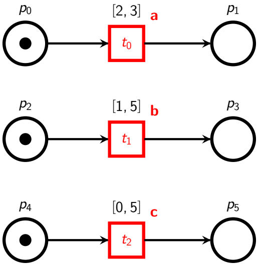
$\rho_\Omega = 2a.0b.0c$

Temps continu

$\rho' = 1f_1.1a.0.1b.0c.3f_2.2o'.1o'$

$\rho'' = 1f_1.1a.0.11b.0c.3f_2.2o'.1o'$

Occurrence d'un motif



Matching : le motif a t-il eu lieu ?

$\rho = 1f_1.1a.0b.0c.3f_2.2o'.1o'$

$\rho_\Omega = 2a.0b.0c$

Temps continu

$\rho' = 1f_1.1a.0.1b.0c.3f_2.2o'.1o'$

$\rho'' = 1f_1.1a.0.11b.0c.3f_2.2o'.1o'$

$\rho''' = 1f_1.1a.0.111b.0c.3f_2.2o'.1o'$

Graphe des Classes [BM83]

Abstraction d'un réseau de Petri temporel sous forme d'un Système de Transitions Labellisé (STL)

Avantages :

- STL $\rightarrow$ séquences de transitions du système
- Abstrait le temps de manière finie sous forme de polyèdres

Inconvénient :

- Accessibilité $\rightarrow$ le langage accepté est un sur-ensemble du langage du système

Besoin d'une autre abstraction

Contributions

3 contributions développées durant cette thèse :

1. Abstraction des préfixes du langage d'un réseau de Petri temporel $\rightarrow$ **chemin**
2. **Diagnosticabilité** de motifs temporels $\rightarrow$ **condition nécessaire et suffisante**
3. **Diagnosticabilisation** : rendre un système diagnosticable $\rightarrow$ **condition suffisante** basée sur une paramétrisation du système étudié

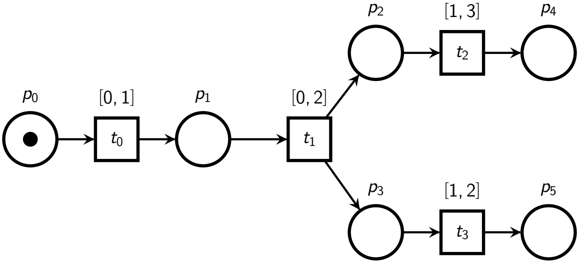
Propriétés de l'abstraction

P0 Abstraire de manière finie l'ensemble des préfixes du langage d'un RdP temporel

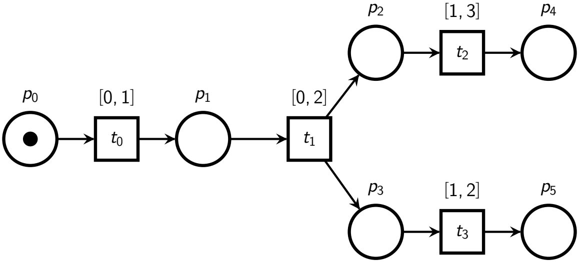
P1 Ne pas sur-approximer le langage du RdP temporel

Abstraction des trajectoires sous forme de polyèdres

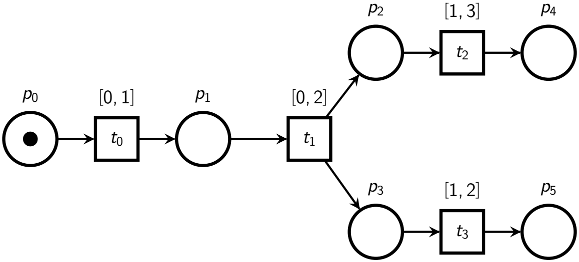
Chemin : $\pi = (\sigma, \Pi)$ paire support/enveloppe temporelle



Support étudié : $\sigma = t_0.t_1.t_2.t_3$

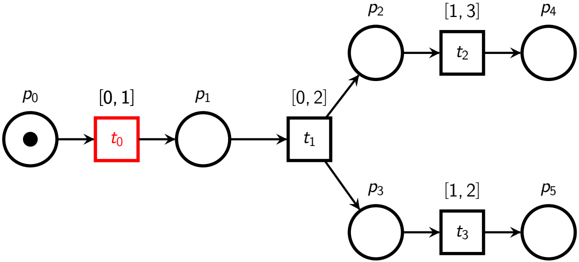


Support étudié : $\sigma = t_0.t_1.t_2.t_3 \rightarrow \{y_0, y_1, y_2, y_3\}$



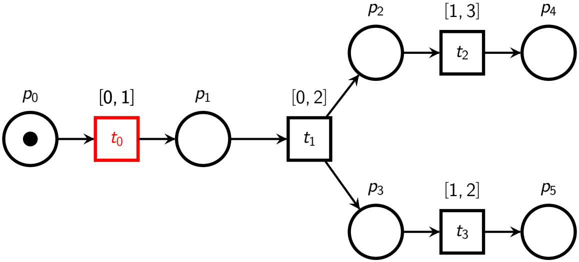
Support étudié : $\sigma = t_0.t_1.t_2.t_3 \rightarrow \{y_0, y_1, y_2, y_3\}$

$$\Pi = \left\{ \vphantom{\begin{array}{c} a \\ a \\ a \\ a \\ a \\ a \end{array}} \right.$$



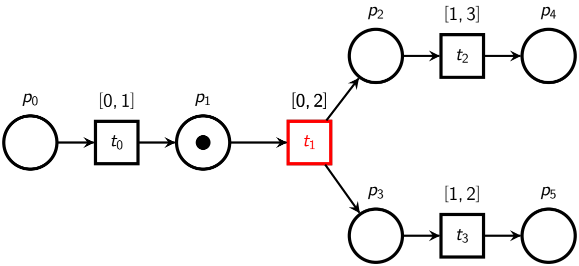
Support étudié : $\sigma = t_0.t_1.t_2.t_3 \rightarrow \{y_0, y_1, y_2, y_3\}$

$$\Pi = \begin{cases} 0 \leqslant y_0 \leqslant y_1 \leqslant y_2 \leqslant y_3 \\ \\ \\ \\ \end{cases}$$



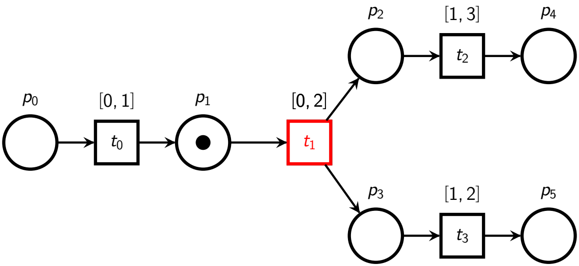
Support étudié : $\sigma = t_0.t_1.t_2.t_3 \rightarrow \{y_0, y_1, y_2, y_3\}$

$$\Pi = \begin{cases} 0 \leqslant y_0 \leqslant y_1 \leqslant y_2 \leqslant y_3 \\ 0 \leqslant y_0 \leqslant 1 \end{cases}$$



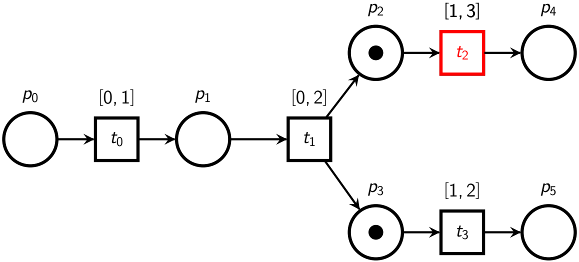
Support étudié : $\sigma = t_0.t_1.t_2.t_3 \to \{y_0, y_1, y_2, y_3\}$

$$\Pi = \begin{cases} 0 \leqslant y_0 \leqslant y_1 \leqslant y_2 \leqslant y_3 \\ \qquad\qquad 0 \leqslant y_0 \leqslant 1 \\ \\ \\ \\ \end{cases}$$



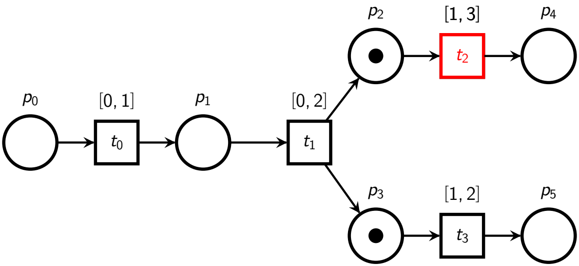
Support étudié : $\sigma = t_0.t_1.t_2.t_3 \rightarrow \{y_0, y_1, y_2, y_3\}$

$$\Pi = \begin{cases} 0 \leqslant y_0 \leqslant y_1 \leqslant y_2 \leqslant y_3 \\ 0 \leqslant y_0 \leqslant 1 \\ 0 \leqslant y_1 - y_0 \leqslant 2 \end{cases}$$



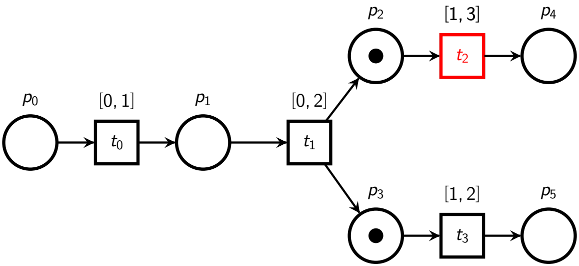
Support étudié : $\sigma = t_0.t_1.t_2.t_3 \rightarrow \{y_0, y_1, y_2, y_3\}$

$$\Pi = \begin{cases} 0 \leqslant y_0 \leqslant y_1 \leqslant y_2 \leqslant y_3 \\ 0 \leqslant y_0 \leqslant 1 \\ 0 \leqslant y_1 - y_0 \leqslant 2 \\ \\ \\ \end{cases}$$



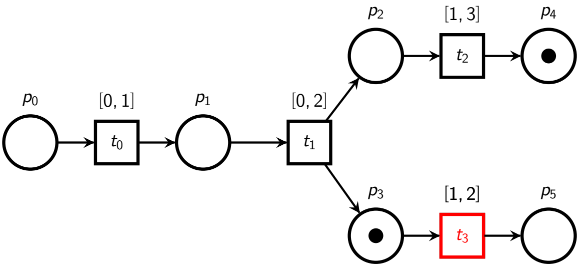
Support étudié : $\sigma = t_0.t_1.t_2.t_3 \rightarrow \{y_0, y_1, y_2, y_3\}$

$$\Pi = \begin{cases} 0 \leqslant y_0 \leqslant y_1 \leqslant y_2 \leqslant y_3 \\ 0 \leqslant y_0 \leqslant 1 \\ 0 \leqslant y_1 - y_0 \leqslant 2 \\ 1 \leqslant y_2 - y_1 \leqslant 3 \end{cases}$$



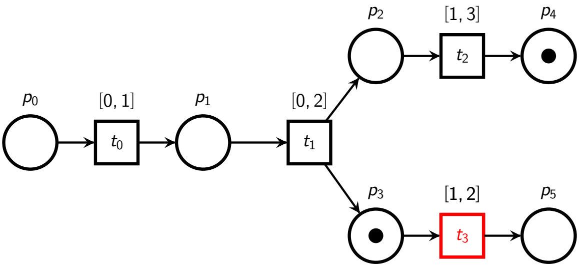
Support étudié : $\sigma = t_0.t_1.t_2.t_3 \rightarrow \{y_0, y_1, y_2, y_3\}$

$$\Pi = \begin{cases} 0 \leqslant y_0 \leqslant y_1 \leqslant y_2 \leqslant y_3 \\ 0 \leqslant y_0 \leqslant 1 \\ 0 \leqslant y_1 - y_0 \leqslant 2 \\ 1 \leqslant y_2 - y_1 \leqslant 3 \\ y_2 - y_1 \leqslant 2 \end{cases}$$



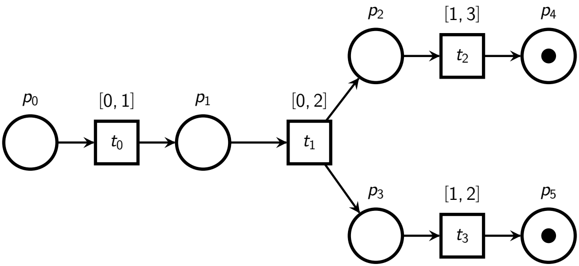
Support étudié : $\sigma = t_0.t_1.t_2.t_3 \rightarrow \{y_0, y_1, y_2, y_3\}$

$$\Pi = \begin{cases} 0 \leqslant y_0 \leqslant y_1 \leqslant y_2 \leqslant y_3 \\ 0 \leqslant y_0 \leqslant 1 \\ 0 \leqslant y_1 - y_0 \leqslant 2 \\ 1 \leqslant y_2 - y_1 \leqslant 3 \\ y_2 - y_1 \leqslant 2 \\ \end{cases}$$



Support étudié : $\sigma = t_0.t_1.t_2.t_3 \rightarrow \{y_0, y_1, y_2, y_3\}$

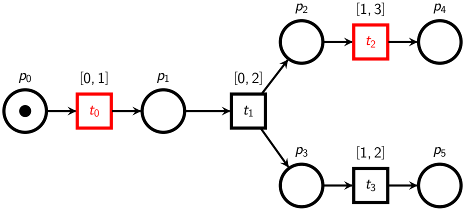
$$\Pi = \begin{cases} 0 \leqslant y_0 \leqslant y_1 \leqslant y_2 \leqslant y_3 \\ 0 \leqslant y_0 \leqslant 1 \\ 0 \leqslant y_1 - y_0 \leqslant 2 \\ 1 \leqslant y_2 - y_1 \leqslant 3 \\ y_2 - y_1 \leqslant 2 \\ 1 \leqslant y_3 - y_1 \leqslant 2 \end{cases}$$



Support étudié : $\sigma = t_0.t_1.t_2.t_3 \rightarrow \{y_0, y_1, y_2, y_3\}$

$$\Pi = \begin{cases} 0 \leqslant y_0 \leqslant y_1 \leqslant y_2 \leqslant y_3 \\ 0 \leqslant y_0 \leqslant 1 \\ 0 \leqslant y_1 - y_0 \leqslant 2 \\ 1 \leqslant y_2 - y_1 \leqslant 3 \\ y_2 - y_1 \leqslant 2 \\ 1 \leqslant y_3 - y_1 \leqslant 2 \end{cases}$$

Projection sur un sous-ensemble de transitions



$$\Pi = \begin{cases} 0 \leqslant y_0 \leqslant y_1 \leqslant y_2 \leqslant y_3 \\ 0 \leqslant y_0 \leqslant 2 \\ 0 \leqslant y_1 - y_0 \leqslant 2 \\ 1 \leqslant y_2 - y_1 \leqslant 3 \\ y_2 - y_1 \leqslant 2 \\ 1 \leqslant y_3 - y_1 \leqslant 2 \end{cases}$$

Projection sur $\{t_0, t_2\}$:

$$\rightarrow \Pi' = \begin{cases} 0 \leqslant y_0 \leqslant 1 \\ 1 \leqslant y_2 - y_0 \leqslant 4 \end{cases}$$

Conclusion

Chemins $\{\pi = (\sigma, \Pi)\}$ **:**

- Abstraction finie des exécutions s'appuyant sur un support donné
- Pour une taille de support, nombre fini de chemins
- Polyèdre $\rightarrow$ Projection $+$ Intersection

Étude de diagnosticabilité

I Synchronisation du système avec le motif

II Abstraction finie des exécutions fautives à l'aide de chemins
$\rightarrow$ **chemins fautifs**

III Identification pour un chemin fautif de ses exécutions fautives
$\rightarrow$ partition de l'enveloppe temporelle observable

IV Synthèse du Graphe d'Ambiguïtés Temporel (GAT) pour mettre en évidence l'existence de **paires critiques**

Étude de diagnosticabilité : Étapes I et II

I Synchronisation du système avec le motif

- calcul des chemins du motif $\mathcal{C}_\Omega = \{\pi_\Omega = (\sigma_\Omega, \Pi_\Omega)\}$
- identification du matching en explorant le graphe des classes du système

II Abstraction finie des exécutions fautives à l'aide de chemins $\rightarrow$ **chemins fautifs**

- support fini terminant sur une transition observable (transitions observables inévitables)
- enveloppe temporelle abstrait les exécutions fautives du support
- il existe a priori des exécutions non fautives parmi les exécutions d'un chemin fautif

Étude de diagnosticabilité : Étape **III**

III Identification pour un chemin fautif de ses exécutions fautives
$\rightarrow$ partition de l'enveloppe temporelle observable

1. pour le chemin du motif $\pi_\Omega = (\sigma_\Omega, \Pi_\Omega)$ associé au chemin fautif π_f étudié, identifier pour chaque contrainte $c \in \Pi_\Omega$ quelles exécutions de π_f satisfont c
2. pour une contrainte $c : \alpha_c \leqslant y_i - y_{i-j} \leqslant \beta_c$ concernant les transitions t_i et t_{i-j} (non observables), reporter c sur des transitions observables bien choisies (report en deux étapes) $c = c^c \oplus c^s \oplus c^a$
3. généralisation du raisonnement à l'ensemble des contraintes de Π_Ω $\rightarrow$ partition de Π_f^o

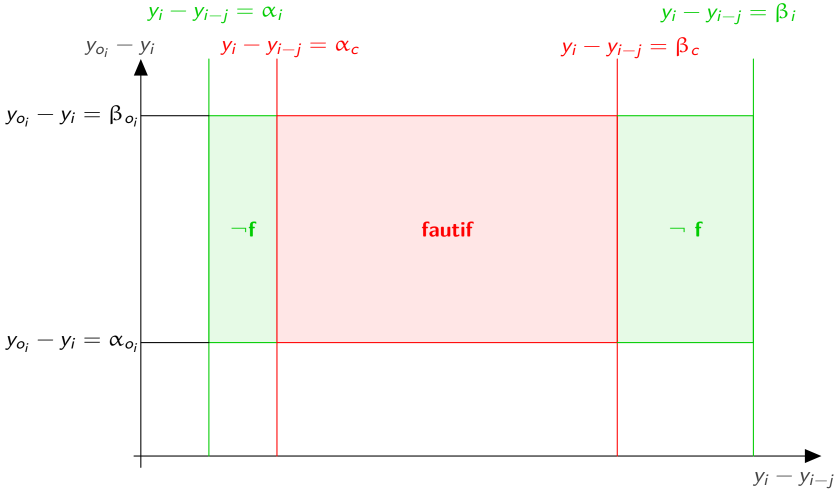
Zoom sur l'étape **III**.2

Ingrédients :

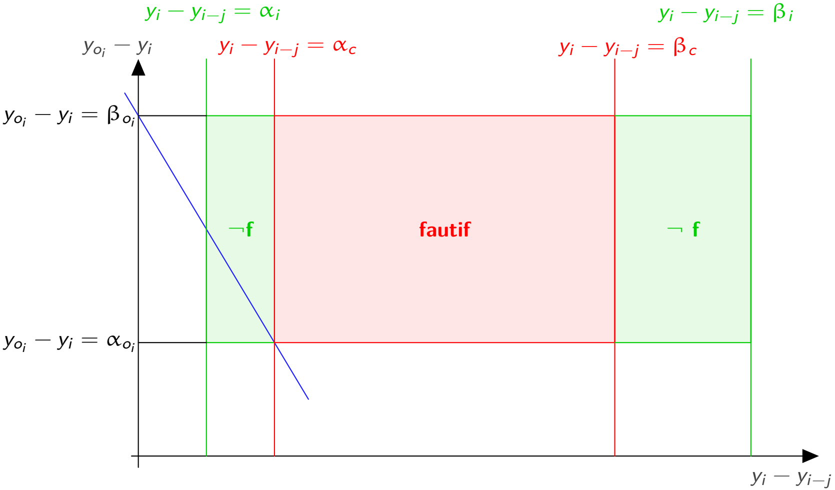
- chemin fautif $\pi_f = (\sigma_f, \Pi_f)$
- support $\sigma_f = t_0 \dots t_{o_{i-j-1}} \dots t_{i-j} \dots t_i \dots t_{o_i} \dots t_n$
- contrainte du motif c : $\alpha_c \leqslant y_i - y_{i-j} \leqslant \beta_c$
- $\Pi_f \rightarrow \alpha_i \leqslant y_i - y_{i-j} \leqslant \beta_i$, $\alpha_{o_i} \leqslant y_{o_i} - y_i \leqslant \beta_{o_i}$, $\alpha_{o_{i-j-1}} \leqslant y_{i-j} - y_{o_{i-j-1}} \leqslant \beta_{o_{i-j-1}}$

1. Report de la contrainte sur $t_{o_i} \rightarrow$ report de y_i vers y_{o_i}
2. Report de la contrainte sur $t_{o_{i-j-1}} \rightarrow$ report de y_{i-j} vers $y_{o_{i-j-1}}$

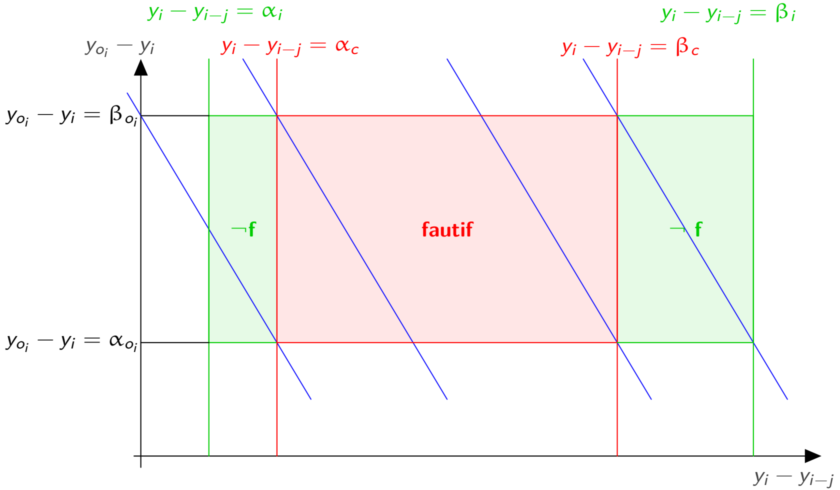
Report de y_i sur y_{o_i}



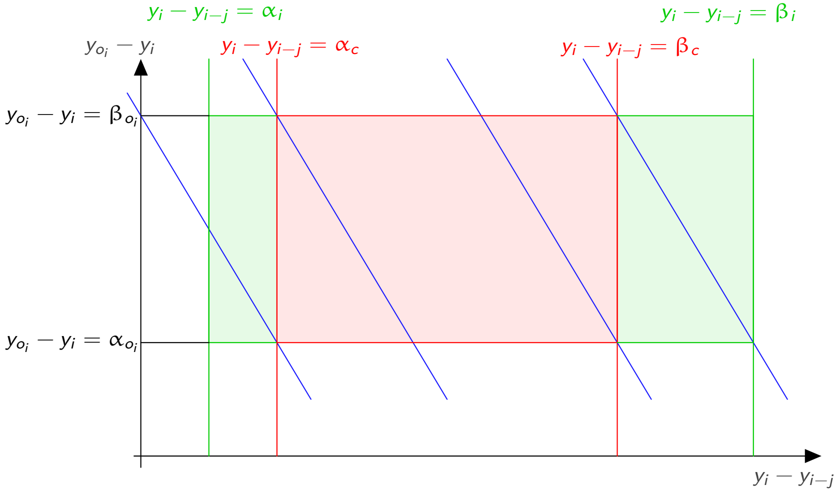
Report de y_i sur y_{o_i}



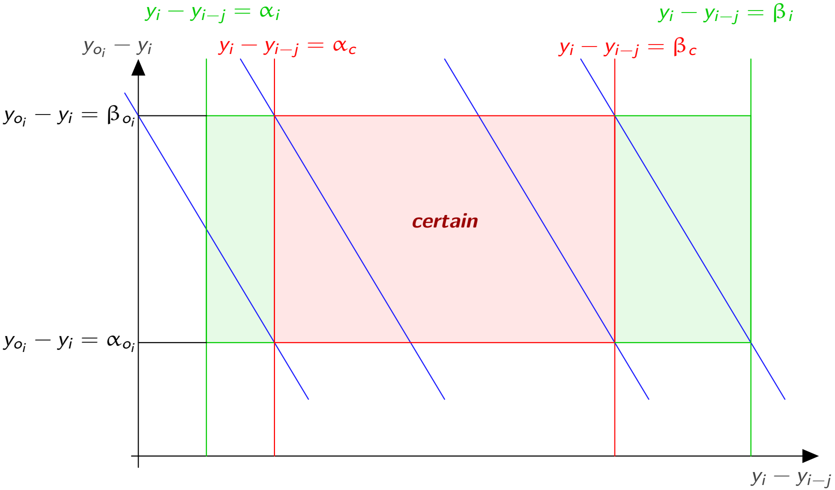
Report de y_i sur y_{o_i}



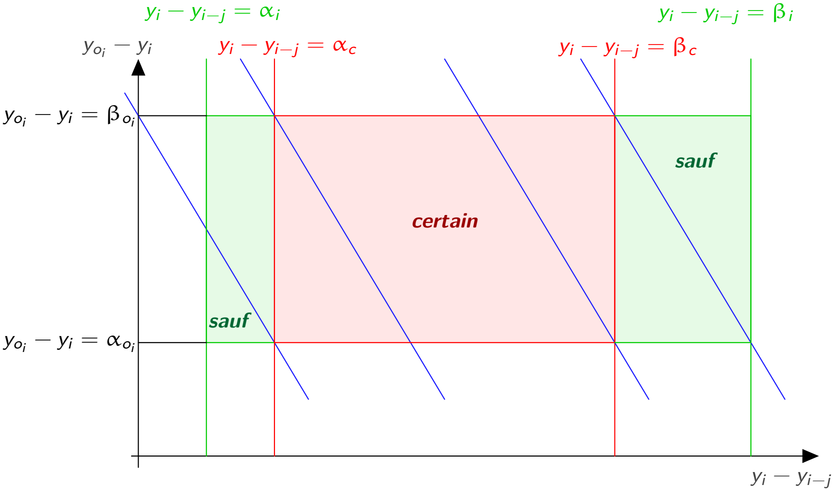
Report de y_i sur y_{o_i}



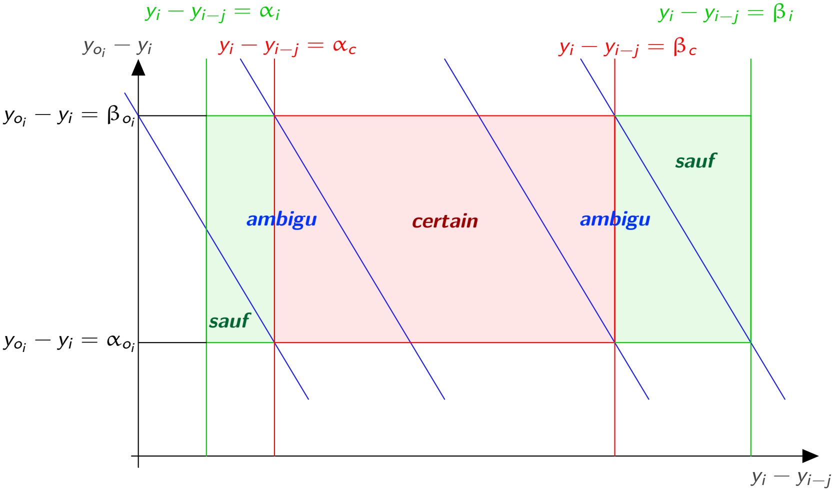
Report de y_i sur y_{o_i}



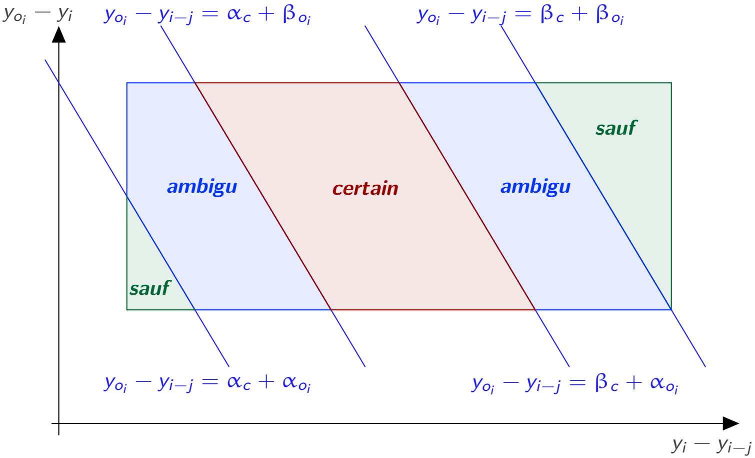
Report de y_i sur y_{o_i}



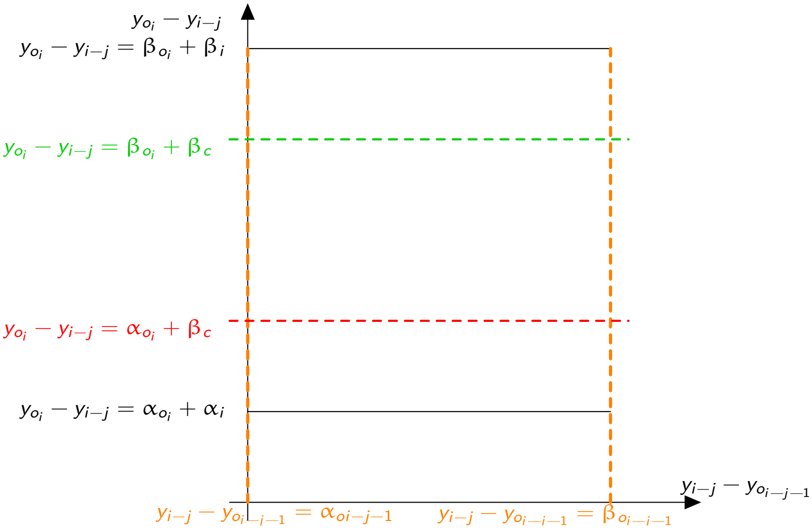
Report de y_i sur y_{o_i}



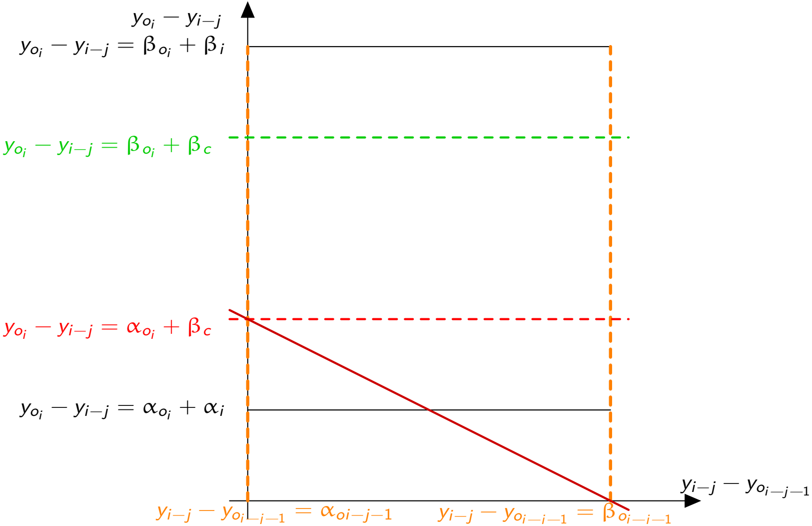
Report de y_i sur y_{o_i}



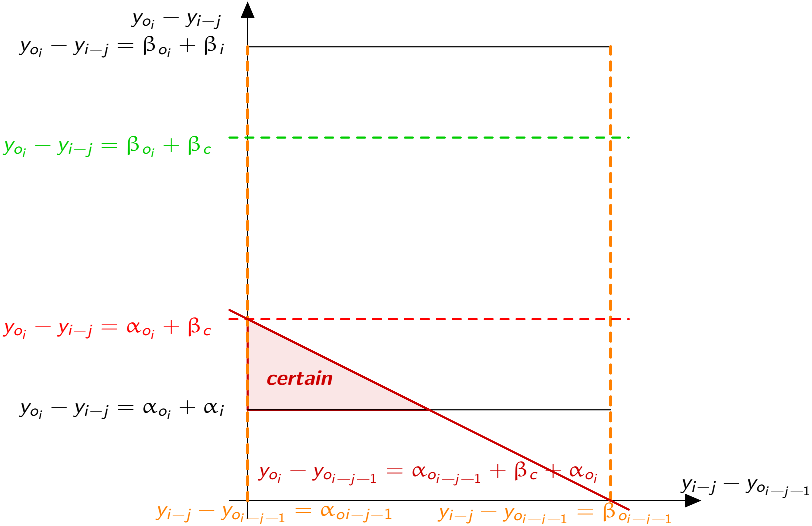
Report de y_{i-j} sur $y_{o_{i-j-1}}$



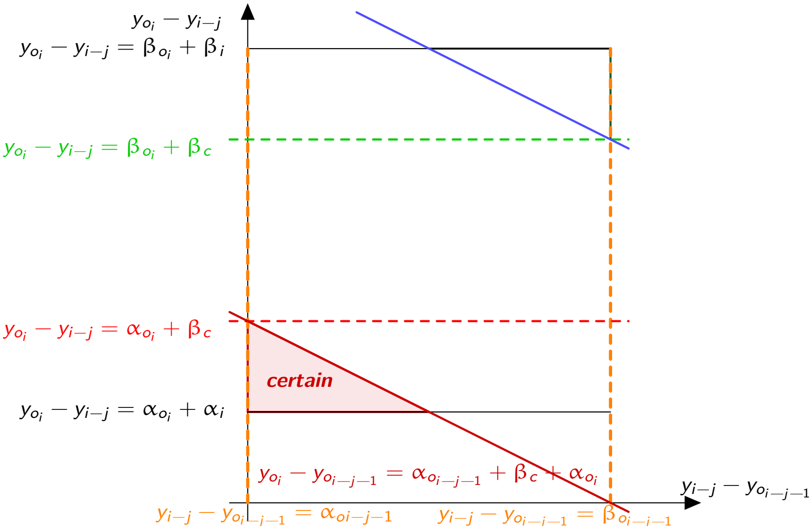
Report de y_{i-j} sur $y_{o_{i-j-1}}$



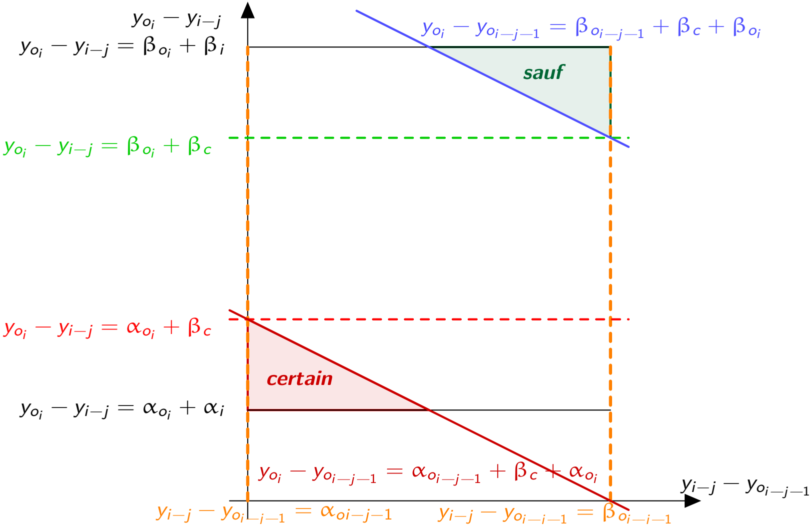
Report de y_{i-j} sur $y_{o_{i-j-1}}$



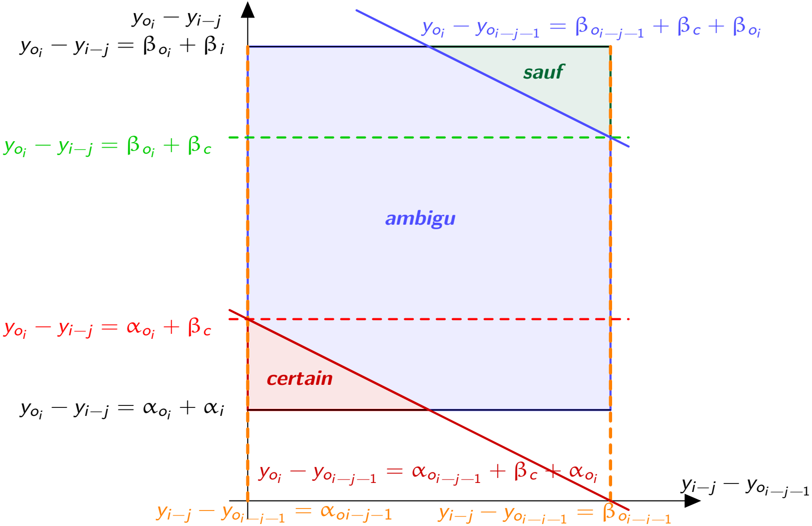
Report de y_{i-j} sur $y_{o_{i-j-1}}$



Report de y_{i-j} sur $y_{o_{i-j-1}}$



Report de y_{i-j} sur $y_{o_{i-j-1}}$



Partition de l'enveloppe temporelle : Étape **III**.3

$$c = c^c \oplus c^s \oplus c^a$$

Généralisation du raisonnement à l'ensemble des contraintes de Π_Ω :

Partition de l'enveloppe temporelle : Étape **III**.3

$$c = c^c \oplus c^s \oplus c^a$$

Généralisation du raisonnement à l'ensemble des contraintes de Π_Ω :

$$\Pi_f^c = \bigwedge_{c \in \Pi_\Omega[Sub]} (\Pi_f^o \wedge c^c)$$

Partition de l'enveloppe temporelle : Étape **III**.3

$$c = c^c \oplus c^s \oplus c^a$$

Généralisation du raisonnement à l'ensemble des contraintes de Π_Ω :

$$\Pi_f^c = \bigwedge_{c \in \Pi_\Omega[Sub]} (\Pi_f^o \wedge c^c)$$

$$\Pi_f^s = \bigvee_{c \in \Pi_\Omega[Sub]} (\Pi_f^o \wedge c^s)$$

Partition de l'enveloppe temporelle : Étape **III**.3

$$c = c^c \oplus c^s \oplus c^a$$

Généralisation du raisonnement à l'ensemble des contraintes de Π_Ω :

$$\Pi_f^c = \bigwedge_{c \in \Pi_\Omega[Sub]} (\Pi_f^o \wedge c^c)$$

$$\Pi_f^s = \bigvee_{c \in \Pi_\Omega[Sub]} (\Pi_f^o \wedge c^s)$$

$$\Pi_f^a = \Pi_f^o \wedge (\bigvee_{c \in \Pi_\Omega[Sub]} (c^a \wedge \bigwedge_{c' \neq c \in \Pi_\Omega[Sub]} \neg c'^s))$$

Partition de l'enveloppe temporelle : Étape **III**.3

- Pour un chemin fautif $\pi_f = (\sigma_f, \Pi_f)$, l'enveloppe temporelle Π_f peut être partitionnée de la manière suivante :

$$\Pi_f^o = \Pi_f^c \oplus \Pi_f^s \oplus \Pi_f^a$$

- Pour un chemin **non fautif** $\pi_{\overline{f}} = (\sigma_{\overline{f}}, \Pi_{\overline{f}})$, l'enveloppe temporelle observable se partitionne ainsi :

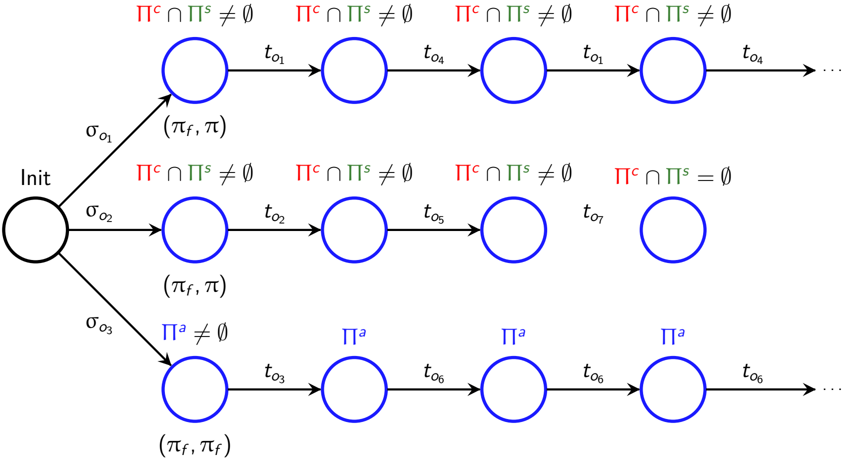
$$\Pi_{\overline{f}}^o = \Pi_{\overline{f}}^s$$

Étude de diagnosticabilité : Étape **IV**

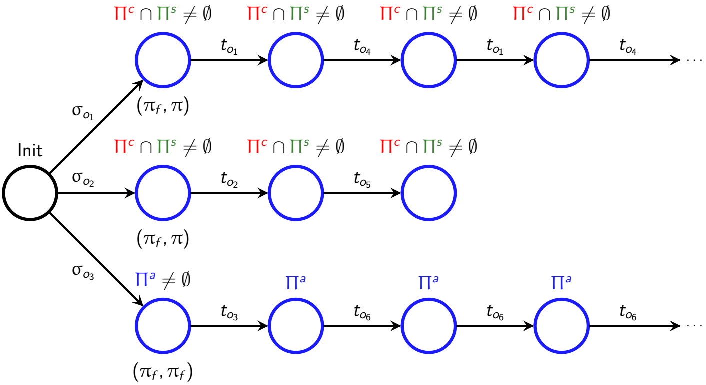
IV Synthèse du Graphe d'Ambiguïtés Temporel (GAT) pour mettre en évidence l'existence de **paires critiques**

- calcul de paires de chemins partageant la même trace observable : mêmes événements et intersection d'enveloppes temporelles observables non-vide
- condition d'arrêt pour éviter un problème infini $\rightarrow$ GAT Réduit (inspiré de la méthode du Twin-plant)

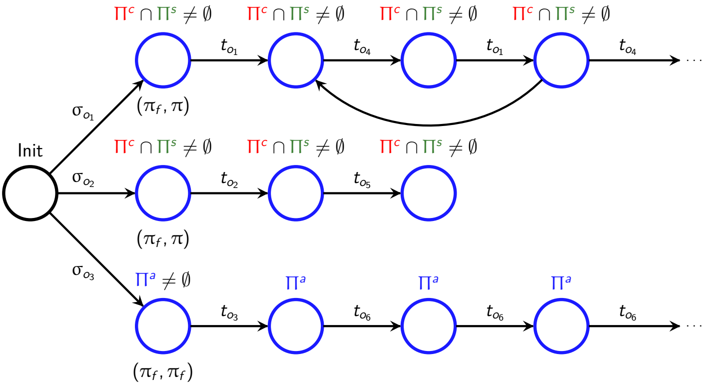
Graphe d'Ambiguïtés Temporel (GAT)



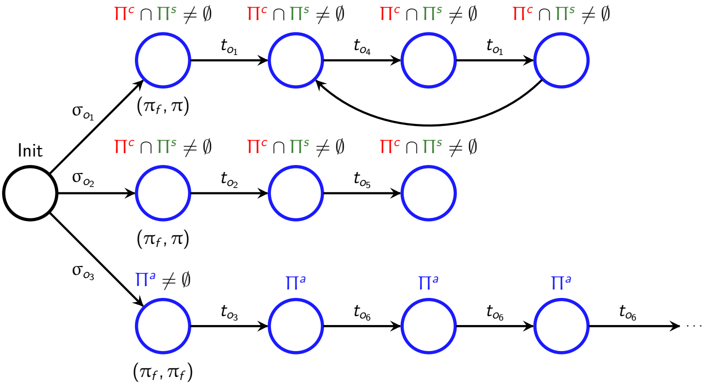
Graphe d'Ambiguïtés Temporel Réduit (GATR)



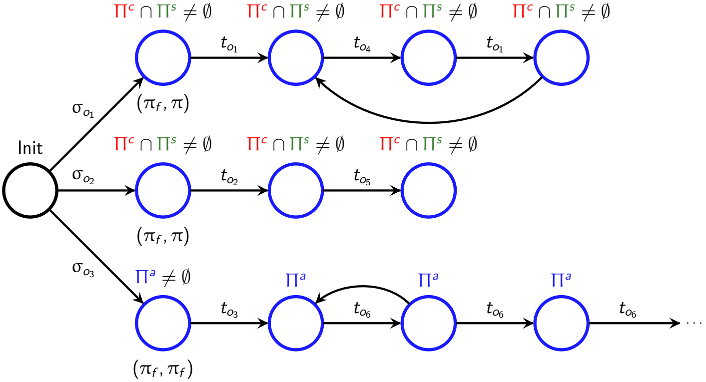
Graphe d'Ambiguïtés Temporel Réduit (GATR)



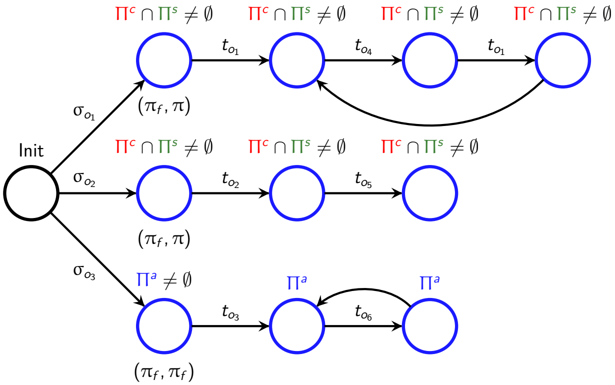
Graphe d'Ambiguïtés Temporel Réduit (GATR)



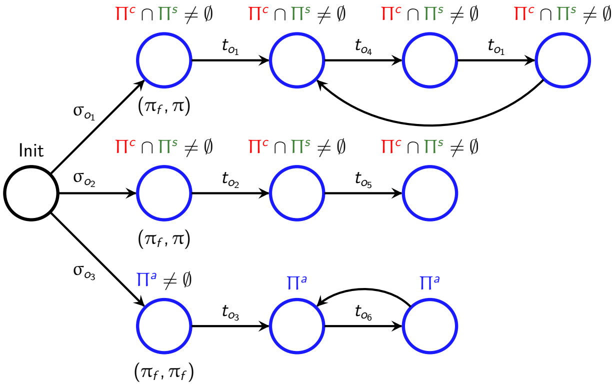
Graphe d'Ambiguïtés Temporel Réduit (GATR)



Graphe d'Ambiguïtés Temporel Réduit (GATR)



Graphe d'Ambiguïtés Temporel Réduit (GATR)



Absence de paires critiques $\Leftrightarrow$ Absence de boucles dans le GATR

Diagnosticabilisation

Que faire dans le cas où un système n'est pas diagnosticable ?

Diagnosticabilisation

Que faire dans le cas où un système n'est pas diagnosticable ?

1. ajouter ou supprimer des observations $\rightarrow$ coûteux

Diagnosticabilisation

Que faire dans le cas où un système n'est pas diagnosticable ?

1. ajouter ou supprimer des observations $\rightarrow$ coûteux
2. modifier la programmation du système en modifiant les intervalles statiques $\rightarrow$ moins coûteux

Diagnosticabilisation

Que faire dans le cas où un système n'est pas diagnosticable ?

1. ajouter ou supprimer des observations $\rightarrow$ coûteux
2. modifier la programmation du système en modifiant les intervalles statiques $\rightarrow$ moins coûteux

- paramétrisation d'un sous-ensemble des bornes des intervalles statiques du système
- pas de modification de la structure logique du système
- préservation du langage atemporel du système

Diagnosticabilisation : méthode de vérification

Pour un système Θ, un motif temporel Ω, un ensemble de paramètres Λ, existe t-il une valuation ν des paramètres de Λ pour laquelle le système engendré Θ_{Λ}^{ν} est diagnosticable pour Ω ?

I Préserver la connaissance de l'analyse de diagnosticabilité
- Préservation de la structure du graphe des classes
- Préservation des chemins fautifs et non fautifs

II Interdire les ambiguités intrinsèques aux chemins fautifs $\rightarrow$ $\Pi^{a} = \emptyset$

III Interdire les ambiguïtés du type $\Pi^{c} \cap \Pi^{s} \neq \emptyset$

Préserver la structure du graphe des classes

- Pas de connaissance a priori de l'espace d'état du système paramétré
- On souhaite étudier des solutions pour lesquelles le graphe des classes engendré G_{Λ}^{γ} est isomorphe à celui du système initial

Synthèse de contraintes qui préservent les transitions tirables depuis une classe $\rightarrow \Pi_{\Lambda}^{struct}$

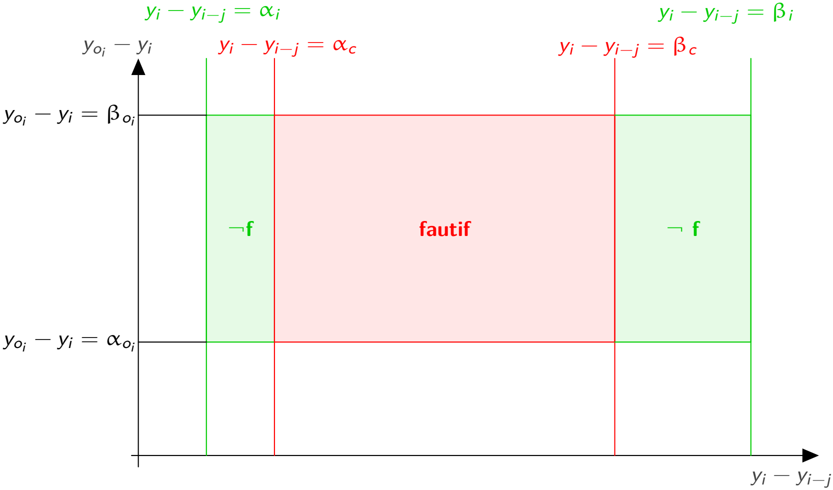
1. même langage atemporel $\rightarrow$ mêmes séquences de transitions tirables
2. mêmes supports de chemins, mais **enveloppes temporelles a priori différentes**

Préserver les chemins fautifs et non fautifs

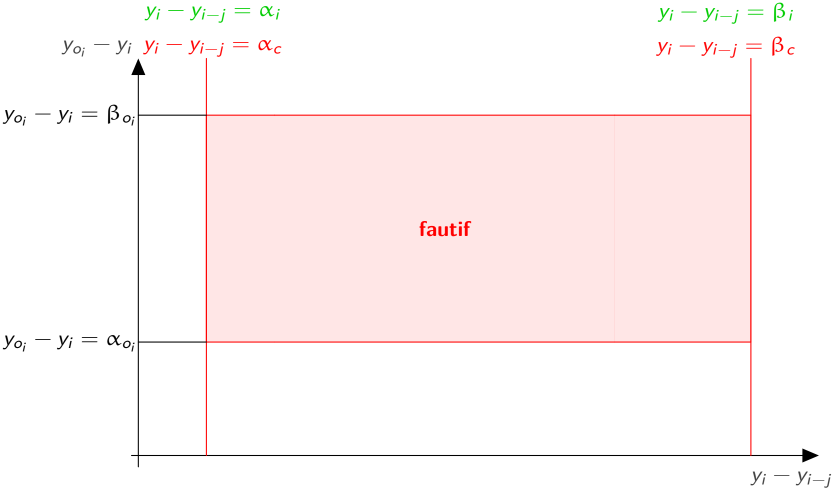
Ne pas changer le fonctionnement global (structure logique) $\rightarrow$ préserver les séquences de transitions fautives et non fautives

1. Support d'un chemin fautif dans $\Theta \Rightarrow$ support de chemin fautif dans $\Theta_\Lambda^\gamma \rightarrow \Pi_\Lambda^f$
2. Support d'un chemin non fautif dans $\Theta \Rightarrow$ support de chemin non fautif dans $\Theta_\Lambda^\gamma \rightarrow \Pi_\Lambda^{\neg f}$

Assurer l'absence d'ambiguïté dans π_f



II Assurer l'absence d'ambiguïté dans π_f



II Assurer l'absence d'ambiguïté dans π_f

Pour tout chemin fautif, et pour tout chemin du motif π_Ω associé, nous imposons :

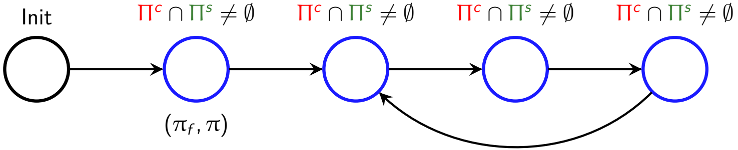
$$\forall c \in \Pi_\Omega, (\alpha_c \leqslant \alpha_i(\nu)) \wedge (\beta_i(\nu) \leqslant \beta_c)$$

L'ensemble de ces contraintes pour tous les chemins fautifs est noté

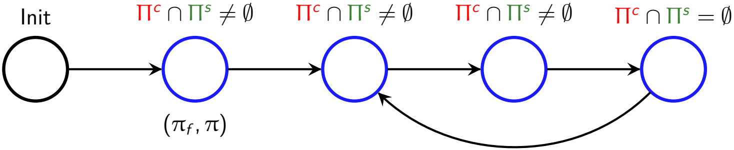
$$\Pi_\Lambda^{\neg a}$$

Conséquence : Un tel choix impose qu'une valuation des paramètres rend toute exécution d'un chemin fautif fautive.

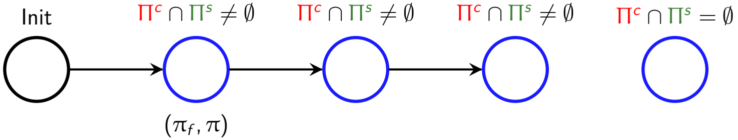
III Assurer l'absence de paires critiques



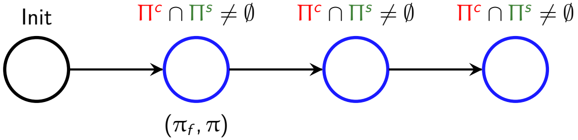
III Assurer l'absence de paires critiques



III Assurer l'absence de paires critiques

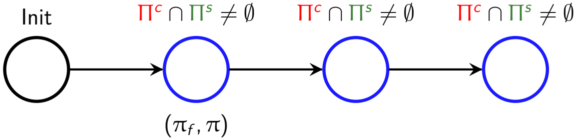


III Assurer l'absence de paires critiques



On note ces contraintes $\Pi_\Lambda^{\neg Cp}$

III Assurer l'absence de paires critiques



On note ces contraintes $\Pi_\Lambda^{\neg Cp}$

Théorème

Si une valuation ν satisfait $\Pi_\Lambda^{struct} \wedge \Pi_\Lambda^{\neg f} \wedge \Pi_\Lambda^{\neg a} \wedge \Pi_\Lambda^{\neg Cp}$, alors le système engendré par cette valuation est diagnosticable pour le motif temporel étudié.

Conclusion

Contributions :

- Extension des motifs de supervision avec du temps → **Motifs temporels**
- **Abstraction finie des préfixes du langage d'un RdPT → Chemins permet une étude de paires de trajectoires**
- **Condition nécessaire et suffisante de diagnosticabilité → GATR, méthode de vérification associée**
- **Condition suffisante pour la diagnosticabilisation → Garanties de propriétés du système engendré**

Limites :

- Motifs temporels s'exécutent en temps fini
- Complexité de la méthode
- Diagnosticabilisation → force des trajectoires à n'être que fautives

Perspectives

- Implémentation complète des deux méthodes proposées
- Étude de K-diagnosticabilité et Δ-diagnosticabilité
- Chercher les ensembles minimaux de paramètres pour pouvoir rendre un système diagnosticable
- Étendre la modélisation des motifs
- Synthèse d'un diagnostiqueur (chroniques)