Coercition temporelle de réseaux de Petri

Didier Lime Claude Martinez Olivier H. Roux

IRCCyN / École Centrale de Nantes

16 novembre 2011

Introduction

- Objectifs:
  - Restreindre les comportements temporels d'un modèle réseaux de Petri pour assurer le respect d'une spécification,
  - Préserver la structure du modèle,
  - Obtenir un algorithme.
- Travaux apparentés:
  - Supervision des systèmes de production (modèles temporisés),
  - Synthèse de paramètres temporels.

Plan

Réseaux de Petri temporels (TPN)

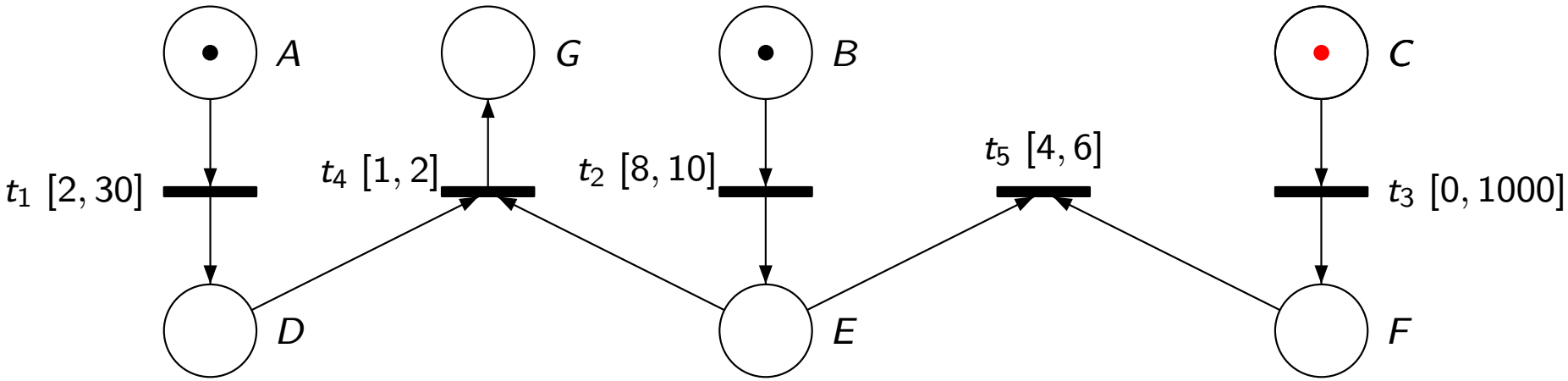


Figure: Un réseau de Petri temporel $\mathcal{N}$

$\{A, B, C\}$
$\nu(t_1) = 0$
$\nu(t_2) = 0$
$\nu(t_3) = 0$

Réseaux de Petri temporels (TPN)

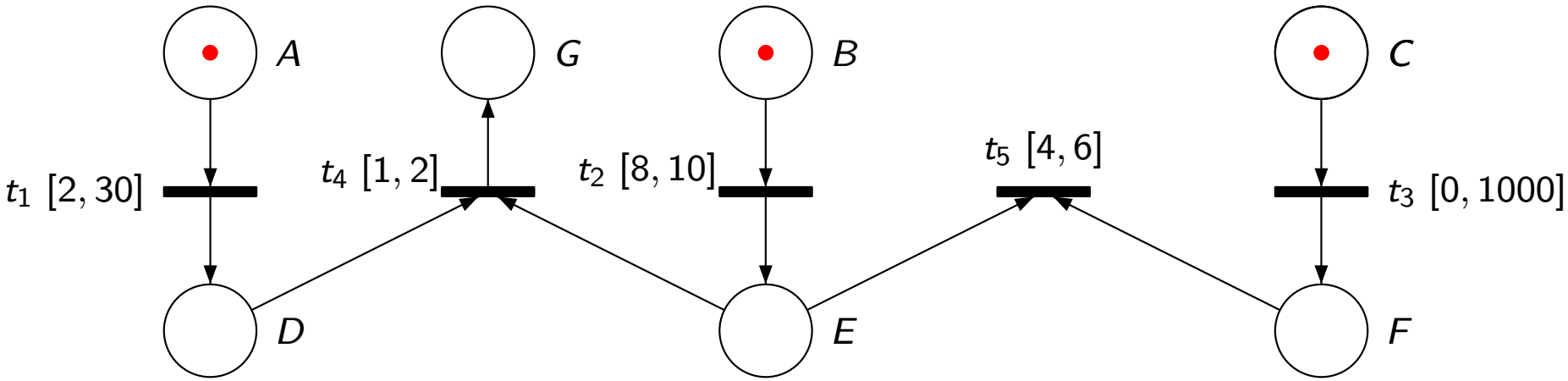


Figure: Un réseau de Petri temporel $\mathcal{N}$

$$\begin{array}{l}\{A, B, C\} \\ \nu(t_1) = 0 \\ \nu(t_2) = 0 \\ \nu(t_3) = 0\end{array} \quad \xrightarrow{\epsilon(8,3)} \quad \begin{array}{l}\{A, B, C\} \\ \nu(t_1) = 8, 3 \\ \nu(t_2) = 8, 3 \\ \nu(t_3) = 8, 3\end{array}$$

Réseaux de Petri temporels (TPN)

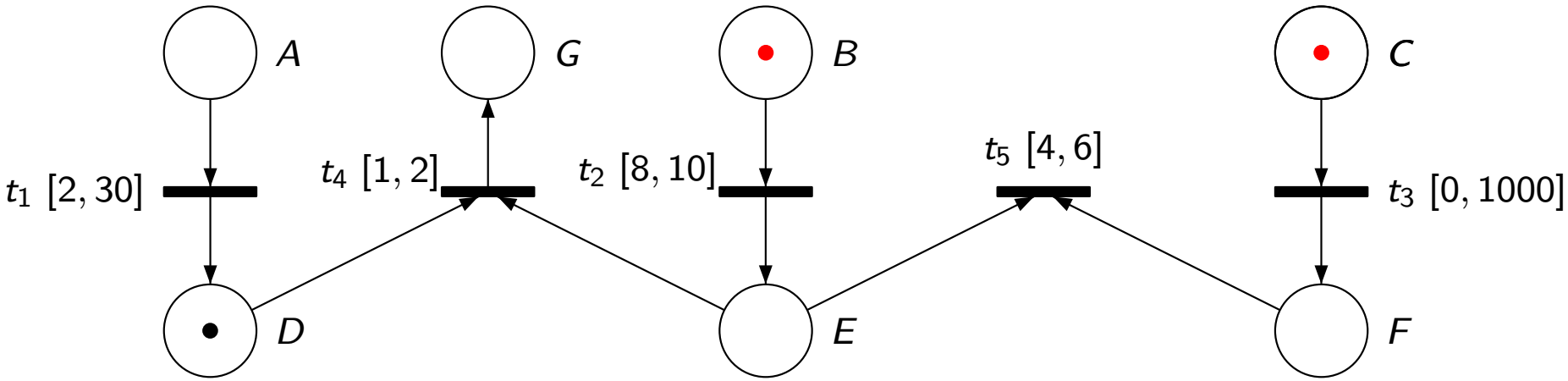


Figure: Un réseau de Petri temporel $\mathcal{N}$

$$\{A, B, C\} \quad \nu(t_1) = 0 \quad \nu(t_2) = 0 \quad \nu(t_3) = 0 \xrightarrow{\epsilon(8,3)} \{A, B, C\} \quad \nu(t_1) = 8, 3 \quad \nu(t_2) = 8, 3 \quad \nu(t_3) = 8, 3 \xrightarrow{t_1} \{B, C, D\} \quad \nu(t_2) = 8, 3 \quad \nu(t_3) = 8, 3$$

Réseaux de Petri temporels (TPN)

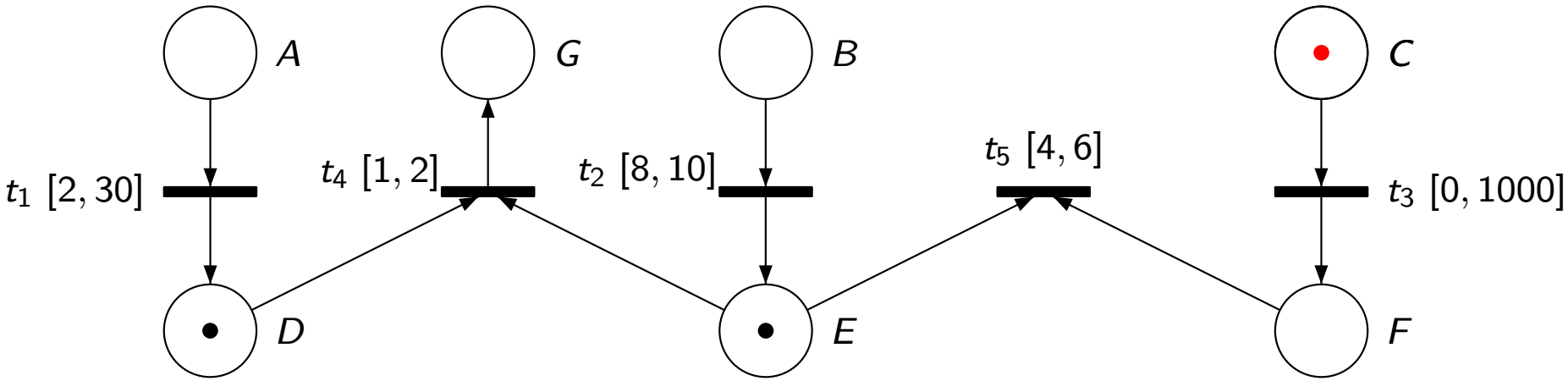


Figure: Un réseau de Petri temporel $\mathcal{N}$

$$\begin{array}{l}\{A,B,C\}\\ \nu(t_1)=0\\ \nu(t_2)=0\\ \nu(t_3)=0\end{array} \xrightarrow{\epsilon(8,3)} \begin{array}{l}\{A,B,C\}\\ \nu(t_1)=8,3\\ \nu(t_2)=8,3\\ \nu(t_3)=8,3\end{array} \xrightarrow{t_1} \begin{array}{l}\{B,C,D\}\\ \nu(t_2)=8,3\\ \nu(t_3)=8,3\end{array} \xrightarrow{t_2} \begin{array}{l}\{C,D,E\}\\ \nu(t_3)=8,3\\ \nu(t_4)=0\end{array}$$

Réseaux de Petri temporels (TPN)

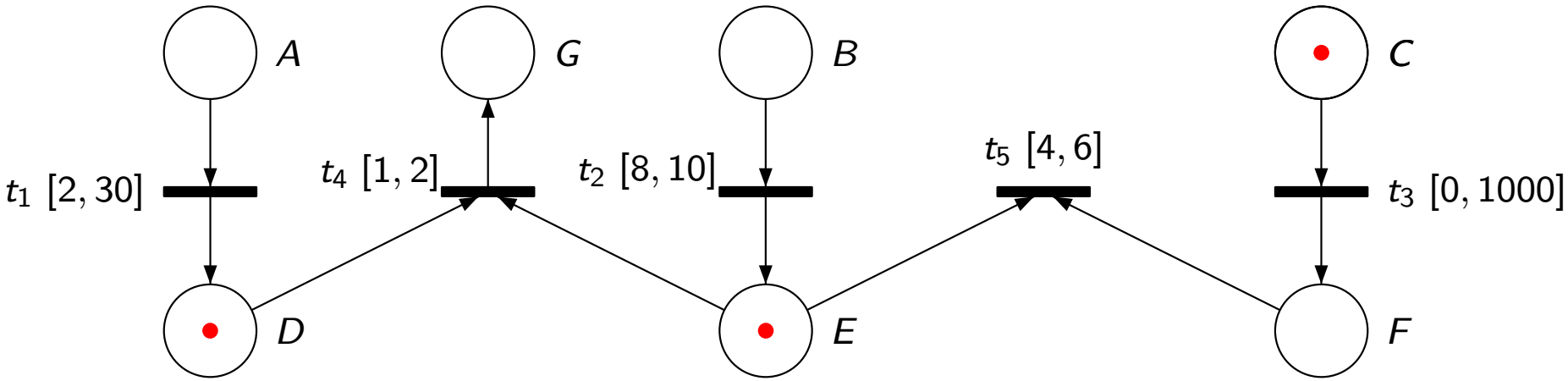


Figure: Un réseau de Petri temporel $\mathcal{N}$

$$\begin{array}{l}\{A,B,C\}\\ \nu(t_1)=0\\ \nu(t_2)=0\\ \nu(t_3)=0\end{array} \xrightarrow{\epsilon(8,3)} \begin{array}{l}\{A,B,C\}\\ \nu(t_1)=8,3\\ \nu(t_2)=8,3\\ \nu(t_3)=8,3\end{array} \xrightarrow{t_1} \begin{array}{l}\{B,C,D\}\\ \nu(t_2)=8,3\\ \nu(t_3)=8,3\end{array} \xrightarrow{t_2} \begin{array}{l}\{C,D,E\}\\ \nu(t_3)=8,3\\ \nu(t_4)=0\end{array} \xrightarrow{\epsilon(2)} \begin{array}{l}\{C,D,E\}\\ \nu(t_3)=10,3\\ \nu(t_4)=2\end{array}$$

Réseaux de Petri temporels (TPN)

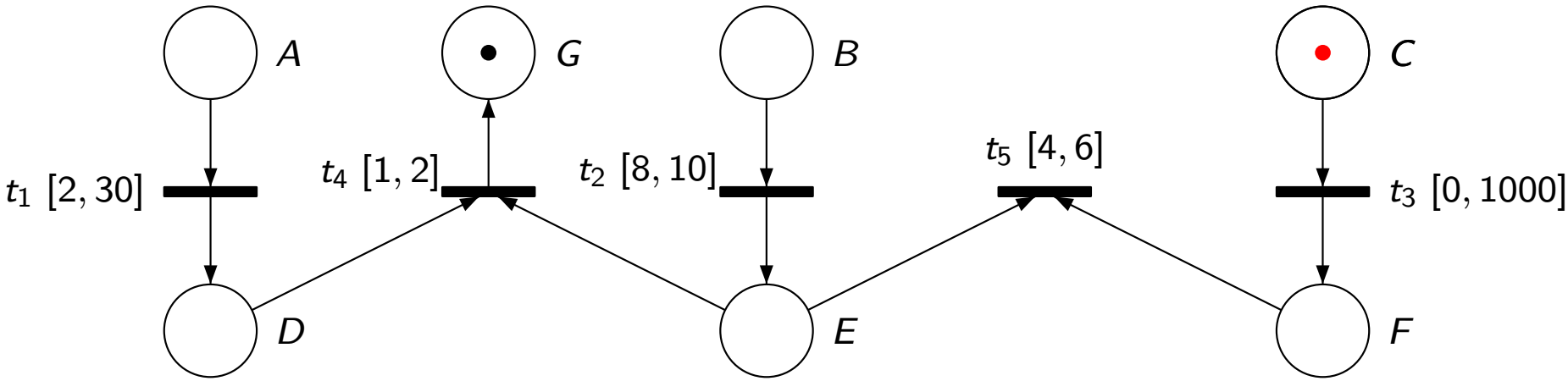


Figure: Un réseau de Petri temporel $\mathcal{N}$

$$\begin{array}{l}\{A,B,C\}\\ \nu(t_1)=0\\ \nu(t_2)=0\\ \nu(t_3)=0\end{array} \xrightarrow{\epsilon(8,3)} \begin{array}{l}\{A,B,C\}\\ \nu(t_1)=8,3\\ \nu(t_2)=8,3\\ \nu(t_3)=8,3\end{array} \xrightarrow{t_1} \begin{array}{l}\{B,C,D\}\\ \nu(t_2)=8,3\\ \nu(t_3)=8,3\end{array} \xrightarrow{t_2} \begin{array}{l}\{C,D,E\}\\ \nu(t_3)=8,3\\ \nu(t_4)=0\end{array} \xrightarrow{\epsilon(2)} \begin{array}{l}\{C,D,E\}\\ \nu(t_3)=10,3\\ \nu(t_4)=2\end{array}$$

$$\xrightarrow{t_4} \begin{array}{l}\{C,G\}\\ \nu(t_3)=10,3\end{array}$$

Réseaux de Petri temporels (TPN)

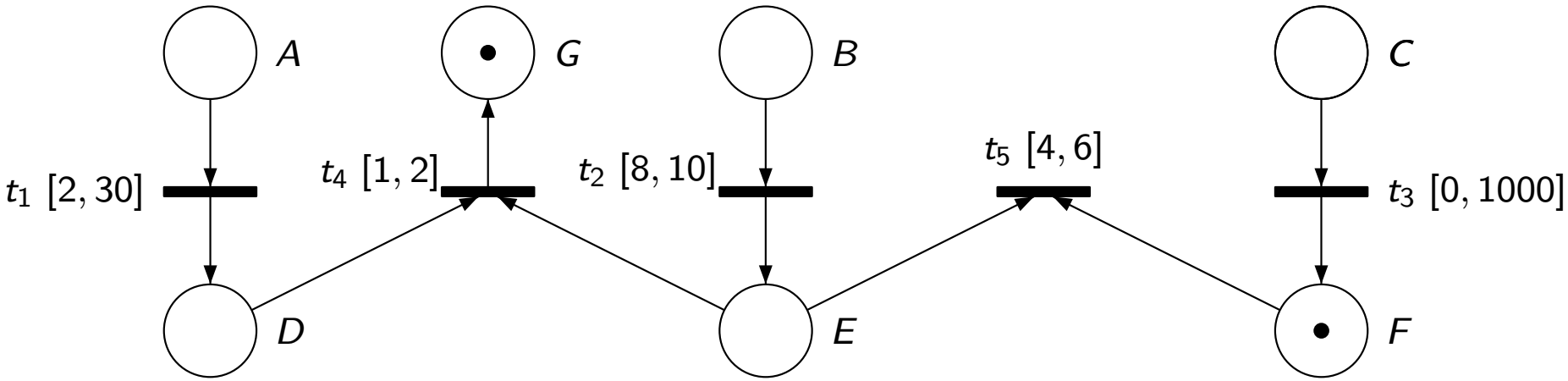


Figure: Un réseau de Petri temporel $\mathcal{N}$

$$\begin{array}{l}\{A,B,C\}\\ \nu(t_1)=0\\ \nu(t_2)=0\\ \nu(t_3)=0\end{array} \xrightarrow{\epsilon(8,3)} \begin{array}{l}\{A,B,C\}\\ \nu(t_1)=8,3\\ \nu(t_2)=8,3\\ \nu(t_3)=8,3\end{array} \xrightarrow{t_1} \begin{array}{l}\{B,C,D\}\\ \nu(t_2)=8,3\\ \nu(t_3)=8,3\end{array} \xrightarrow{t_2} \begin{array}{l}\{C,D,E\}\\ \nu(t_3)=8,3\\ \nu(t_4)=0\end{array} \xrightarrow{\epsilon(2)} \begin{array}{l}\{C,D,E\}\\ \nu(t_3)=10,3\\ \nu(t_4)=2\end{array}$$

$$\xrightarrow{t_4} \begin{array}{l}\{C,G\}\\ \nu(t_3)=10,3\end{array} \xrightarrow{t_3} \{F,G\}$$

Plan

Problème de la coercition temporelle de réseaux de Petri

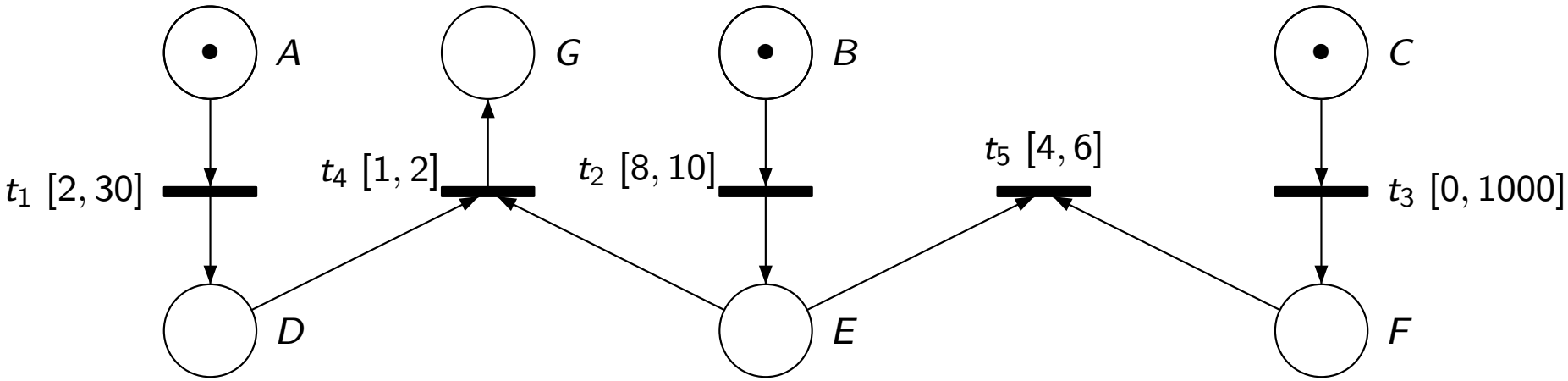


Figure: Un réseau de Petri temporel $\mathcal{N}$

Problème de la coercition temporelle de réseaux de Petri

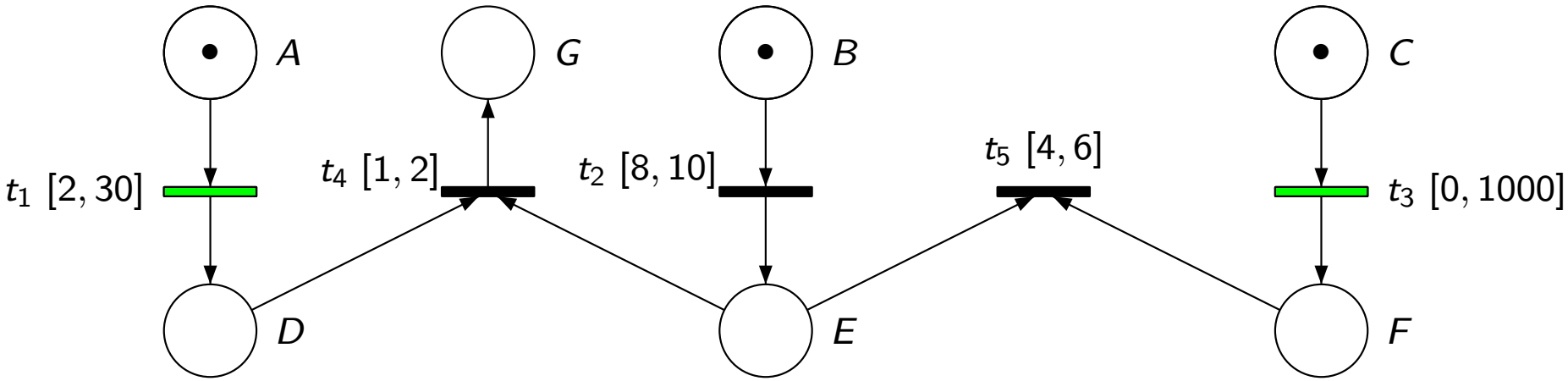


Figure: Un réseau de Petri temporel $\mathcal{N}$

Problème de la coercition temporelle de réseaux de Petri

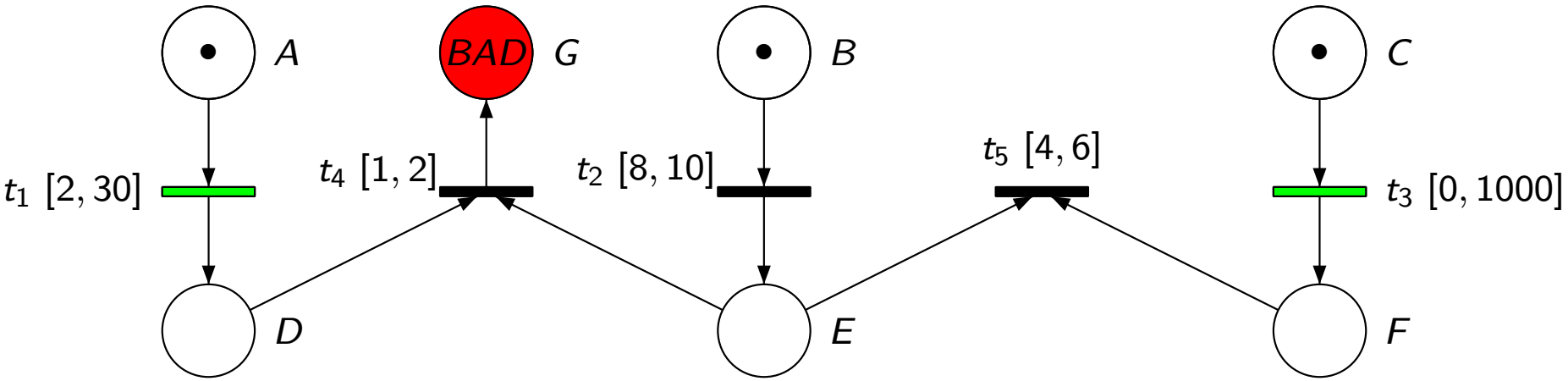


Figure: Un réseau de Petri temporel $\mathcal{N}$

Une coercition

Definition (Coercition d'un TPN)

Etant donnés

- un réseau de Petri temporel $\mathcal{N} = (P, T, {}^\bullet(.), (.)^\bullet, M_0, (\alpha, \beta))$,
- un ensemble de transitions $T_r \subseteq T$,

Une coercition

Definition (Coercition d'un TPN)

Etant donnés

- un réseau de Petri temporel $\mathcal{N} = (P, T, {}^\bullet(.), (.)^\bullet, M_0, (\alpha, \beta))$,
- un ensemble de transitions $T_r \subseteq T$,

une coercition c de $\mathcal{N}$ sur T_r est un réseau de Petri $[\![(\mathcal{N}, T_r)]\!]_c = (P, T, {}^\bullet(.), (.)^\bullet, M_0, (\alpha^c, \beta^c))$ tel que :

$$\forall t \in T,\ tq\ t \notin T_r \begin{cases} \alpha^c(t) = \alpha(t) \\ \beta^c(t) = \beta(t) \end{cases} \quad \text{et } \forall t \in T_r, \begin{cases} \alpha^c(t) \geq \alpha(t) \\ \beta^c(t) \leq \beta(t) \\ \alpha^c(t) \leq \beta^c(t) \end{cases}$$

Ensemble de coercition

Definition (Ensemble de coercition relatif à une propriété)

Etant donnés

- un réseau de Petri temporel $\mathcal{N}$,
- un sous ensemble de ses transitions T_r,
- et une propriété φ,

l'ensemble de coercition de $\mathcal{N}$ sur T_r relatif à φ est l'ensemble $\chi(\mathcal{N}, T_r, \varphi)$ tel que :

$$\forall c \in \chi(\mathcal{N}, T_r, \varphi),\ [\![(\mathcal{N}, T_r)]\!]_c \models \varphi$$

SCP_φ

Definition (Existence et synthèse des coercitions d'un TPN : SCP_φ)

Etant donnés un réseau $\mathcal{N}$, un sous ensemble de ses transitions T_r, et une propriété φ,

- **Existence** : Existe-t-il une coercition c, telle que $[\![(\mathcal{N}, T_r)]\!]_c \models \varphi$?
- **Synthèse** : Déterminer le plus grand ensemble $\chi(\mathcal{N}, T_r, \varphi)$.

L'existence d'une coercition se ramène à $\chi(\mathcal{N}, T_r, \varphi) \neq \emptyset$.
Nous regroupons ces 2 problèmes sous le nom SCP_φ.

Décidabilité

Theorem (SCP_{CTL} est décidable pour les TPN bornés)

Etant donnés un réseau de Petri temporel borné $\mathcal{N}$, un ensemble de transitions T_r, et une propriété CTL quelconque φ,

- *le problème de l'existence d'une coercition est* **décidable**
- *et il existe un algorithme de synthèse de l'ensemble de coercition $\chi(\mathcal{N}, T_r, \varphi)$.*

Plan

Espace d'état coercible

Definition (Classe d'états coercible)

Une *classe d'états coercible* C d'un tpn $\mathcal{N} = (P, T, {}^\bullet(.), (.)^\bullet, M_0, (\alpha, \beta))$ coercible sur l'ensemble $T_r \subseteq T$, est une paire (M, D) telle que

- M est un marquage du réseau
- D est un domaine de tirs représenté par un polyèdre convexe sur
  - α_i et β_i $\forall t_i \in T_r$
  - θ_j pour les dates de tir des transitions sensibilisées t_j

Espace d'état coercible

Definition (Classe d'états coercible)

Une *classe d'états coercible* C d'un tpn $\mathcal{N} = (P, T, {}^\bullet(.), (.)^\bullet, M_0, (\alpha, \beta))$ coercible sur l'ensemble $T_r \subseteq T$, est une paire (M, D) telle que

- M est un marquage du réseau
- D est un domaine de tirs représenté par un polyèdre convexe sur
  - α_i et β_i $\forall t_i \in T_r$
  - θ_j pour les dates de tir des transitions sensibilisées t_j

Definition (Condition (ou domaine) d'accessibilité d'une classe)

- $\Delta(D)$, la projection D sur les variables α_i et β_i $(\forall t_i \in T_r)$
- $D_{|\alpha,\beta}$ est le plus petit polyèdre convexe contenant tous (et seulement) les points entiers de $\Delta(D)$.

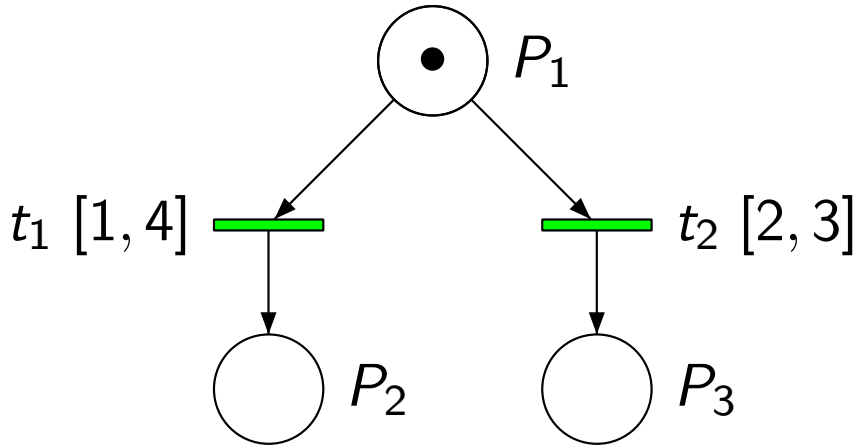
Graphe des classes d'états coercibles

Definition (Successeur d'une classe d'états coercible C=(M,D))

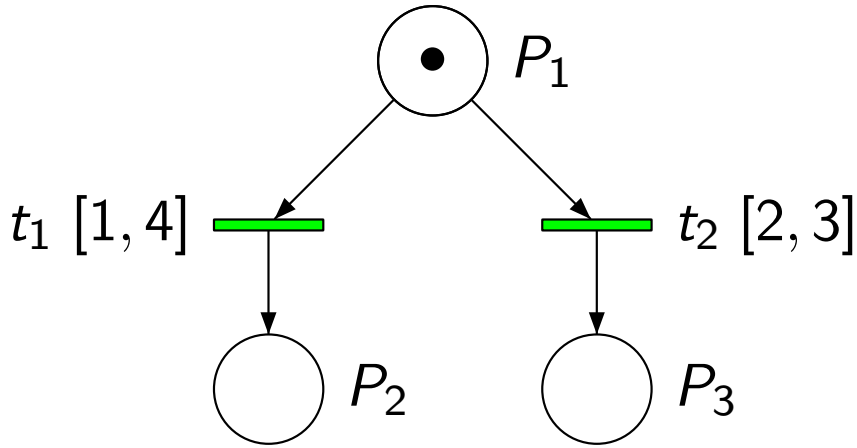
Le successeur de C par le tir de t_f : $C = (M, D) \xrightarrow{t_f} C' = (M', D')$, est calculé de la manière suivante :

- $M' = M -^{\bullet} t_f + t_f^{\bullet}$
- $D' = next(D, t_f)$ est calculé à partir de D par les étapes suivantes :
  1. intersection avec les contraintes de tirabilité : $\forall j,\ \theta_f \leq \theta_j$
  2. substitution des variables $\forall t_j \neq t_i : \theta_j = \theta_f + \theta'_j$,
  3. élimination des variables θ relatives aux transitions désensibilisées par le tir de t_f (par exemple par la méthode de Fourier-Motzkin),
  4. projection sur $\mathbb{N}$ des variables α et β.
  5. ajout des inéquations relatives aux transitions nouvellement sensibilisées

Graphe des classes d'états coercibles



Graphe des classes d'états coercibles

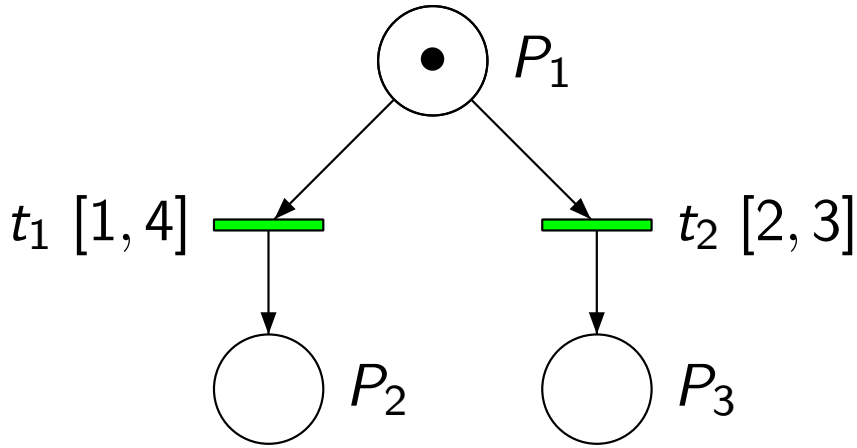


$\mathbf{C_0} = (\mathbf{M_0}, \mathbf{D_0})$:

$M_0 = \{P_1\}$

$$D_0 = \begin{cases} 1 \leq \alpha_1 \leq \beta_1 \leq 4 \\ 2 \leq \alpha_2 \leq \beta_2 \leq 3 \\ \alpha_1 \leq \theta_1 \leq \beta_1, \\ \alpha_2 \leq \theta_2 \leq \beta_2, \end{cases}$$

Graphe des classes d'états coercibles



$\mathbf{C_0} = (\mathbf{M_0}, \mathbf{D_0})$:

$M_0 = \{P_1\}$

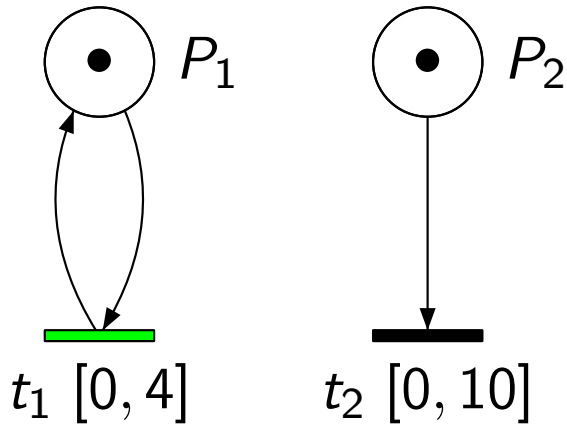
$$D_0 = \begin{cases} 1 \leq \alpha_1 \leq \beta_1 \leq 4 \\ 2 \leq \alpha_2 \leq \beta_2 \leq 3 \\ \alpha_1 \leq \theta_1 \leq \beta_1, \\ \alpha_2 \leq \theta_2 \leq \beta_2, \end{cases} \xrightarrow{t_1}$$

$\mathbf{C_1} = (\mathbf{M_1}, \mathbf{D_1})$:

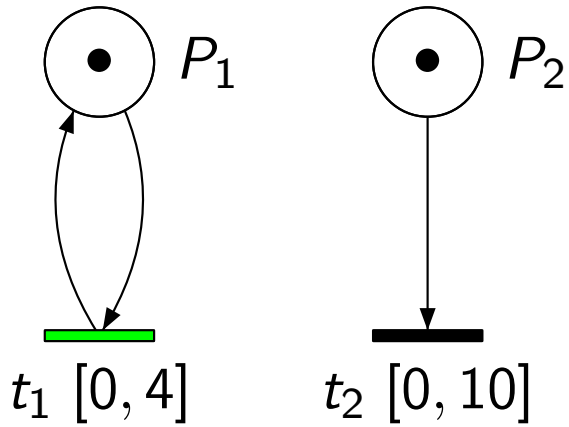
$M_1 = (P_2)$

$$D_1 = \begin{cases} 1 \leq \alpha_1 \leq \beta_1 \leq 4 \\ 2 \leq \alpha_2 \leq \beta_2 \leq 3 \\ \beta_2 \geq \alpha_1 \end{cases}$$

Terminaison de l'algorithme

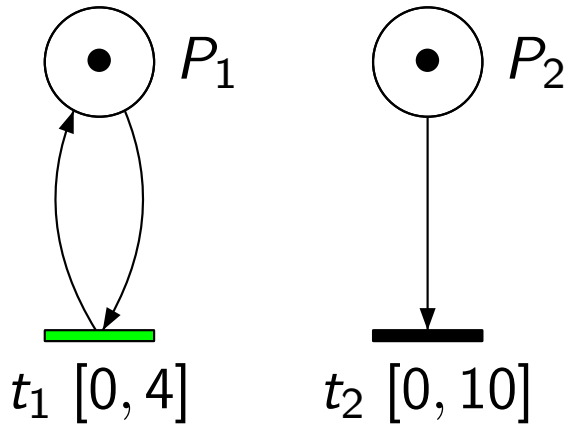


Terminaison de l'algorithme



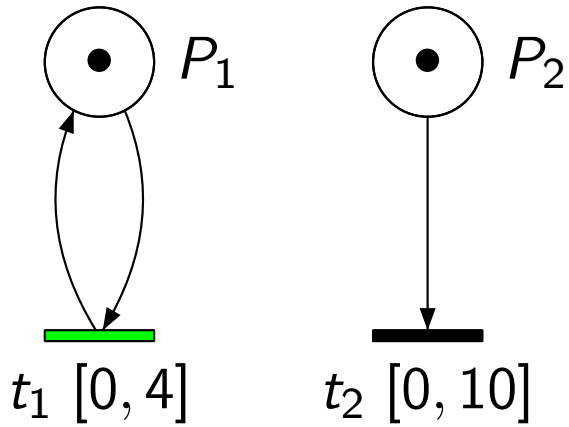
- **Problème** : Le tir répété n fois de t_1 conduit à l'équation $0 \leq \theta_2 \leq -n * \alpha_1 + 10$ et donc à $n * \alpha_1 \leq 10$

Terminaison de l'algorithme



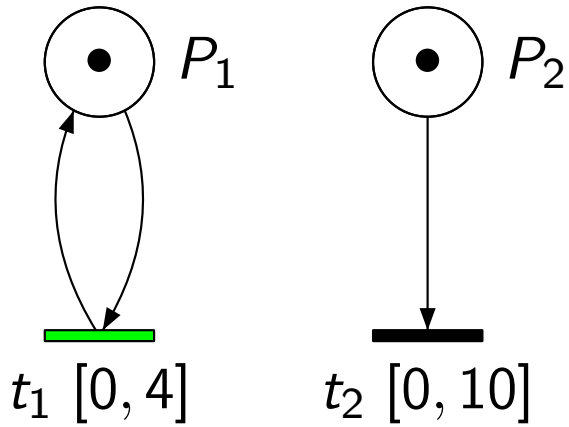
- **Problème** : Le tir répété n fois de t_1 conduit à l'équation $0 \leq \theta_2 \leq -n * \alpha_1 + 10$ et donc à $n * \alpha_1 \leq 10$
  - Les variables α_1 et β_1 sont entières donc $\alpha_1 \leq \lfloor 10/n \rfloor$ et donc pour $n > 10$ nous obtenons $\alpha_1 = 0$.

Terminaison de l'algorithme



- **Problème** : Le tir répété n fois de t_1 conduit à l'équation $0 \leq \theta_2 \leq -n * \alpha_1 + 10$ et donc à $n * \alpha_1 \leq 10$
  - Les variables α_1 et β_1 sont entières donc $\alpha_1 \leq \lfloor 10/n \rfloor$ et donc pour $n > 10$ nous obtenons $\alpha_1 = 0$.
- **Problème** : Le tir répété n fois de t_1 conduit à l'équation $\theta_2 \geq -n * \beta_1 + 10$

Terminaison de l'algorithme



- **Problème** : Le tir répété n fois de t_1 conduit à l'équation $0 \leq \theta_2 \leq -n * \alpha_1 + 10$ et donc à $n * \alpha_1 \leq 10$
  - Les variables α_1 et β_1 sont entières donc $\alpha_1 \leq \lfloor 10/n \rfloor$ et donc pour $n > 10$ nous obtenons $\alpha_1 = 0$.
- **Problème** : Le tir répété n fois de t_1 conduit à l'équation $\theta_2 \geq -n * \beta_1 + 10$
  - Le cas Zenon est traité en distinguant $\beta_1 = 0$ de $\beta_1 \geq 1$

Plan

Ensemble de coercition relatif à une propriété

Fragment de CTL : formules non imbriquées et sans l'opérateur X

Ensemble de coercition relatif à une propriété

Fragment de CTL : formules non imbriquées et sans l'opérateur X

Deux algorithmes récursifs dédiés dont la condition de terminaison est l'inclusion :

- Algorithme EU: $\phi = \exists \varphi \mathcal{U} \psi$
- Algorithme AU: $\phi = \forall \varphi \mathcal{U} \psi$

Ensemble de coercition relatif à une propriété

Algorithme EU: pour les formules de la forme $\phi = \exists \varphi \mathcal{U} \psi$. Soit la classe $C = (M, D)$, nous calculons :

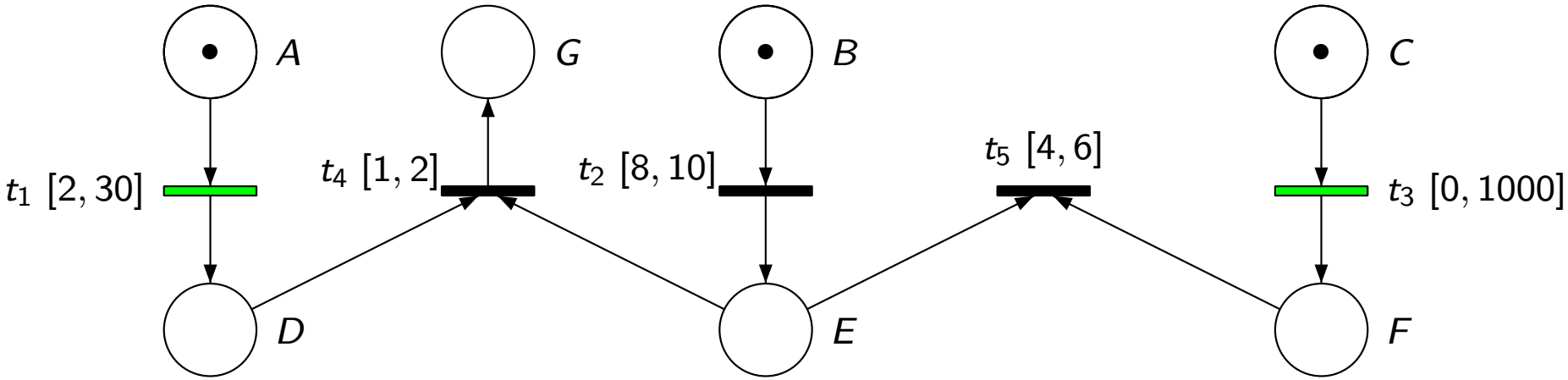
$$F_{\exists \varphi \mathcal{U} \psi}(C) = D_{|\alpha,\beta} \wedge \Bigg(M \models \psi \vee \Bigg(M \models \varphi \wedge M \not\models \psi \wedge \Bigg(\bigvee_{\substack{t \in firable(C) \\ C' = succ(C,t)}} F_{\exists \varphi \mathcal{U} \psi}(C') \Bigg) \Bigg) \Bigg)$$

Ensemble de coercition relatif à une propriété

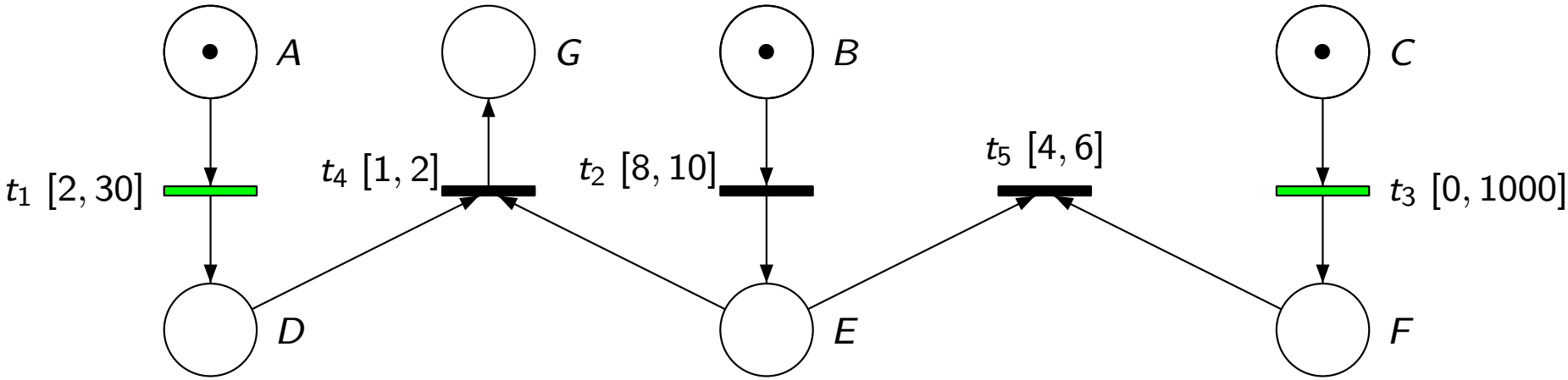
Algorithme AU: Pour les formules de la forme $\phi = \forall \varphi \mathcal{U} \psi$. Soit la classe $C = (M, D)$, nous calculons :

$$F_{\forall \varphi \mathcal{U} \psi}(C) = D_{|\alpha,\beta} \wedge \left(M \models \psi \vee \left(M \models \varphi \wedge M \not\models \psi \wedge \left(\bigwedge_{\substack{t \in firable(C) \\ C'=(M',D')=succ(C,t)}} \left(F_{\forall \varphi \mathcal{U} \psi}(C') \vee \neg D'_{|\alpha,\beta} \right) \right) \right) \right)$$

Exemples

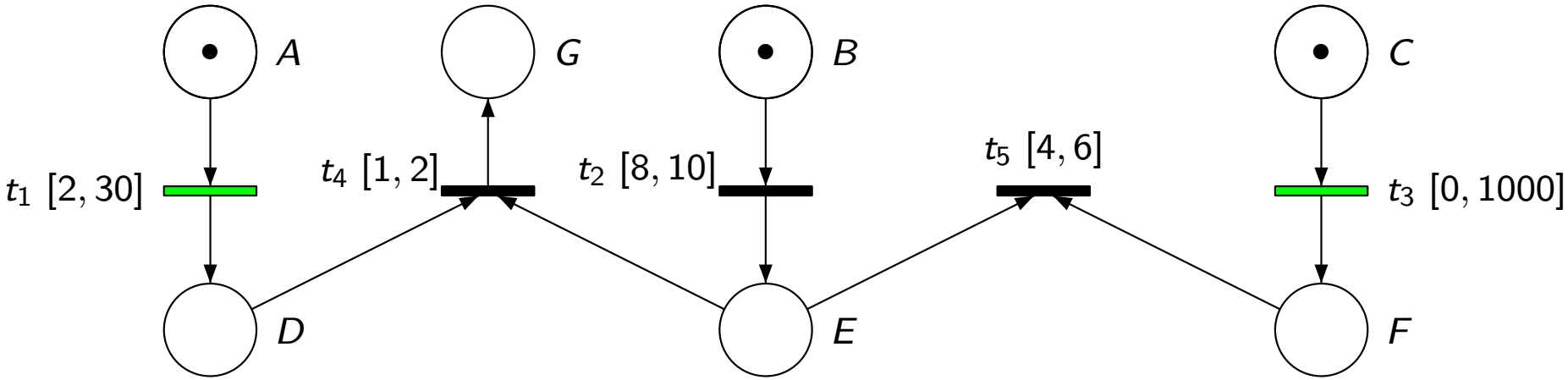


Exemples



Considérons le réseau $\mathcal{N}$ avec $T_r = \{t_1, t_3\}$ et la formule $\forall\Box(M(G) = 0)$

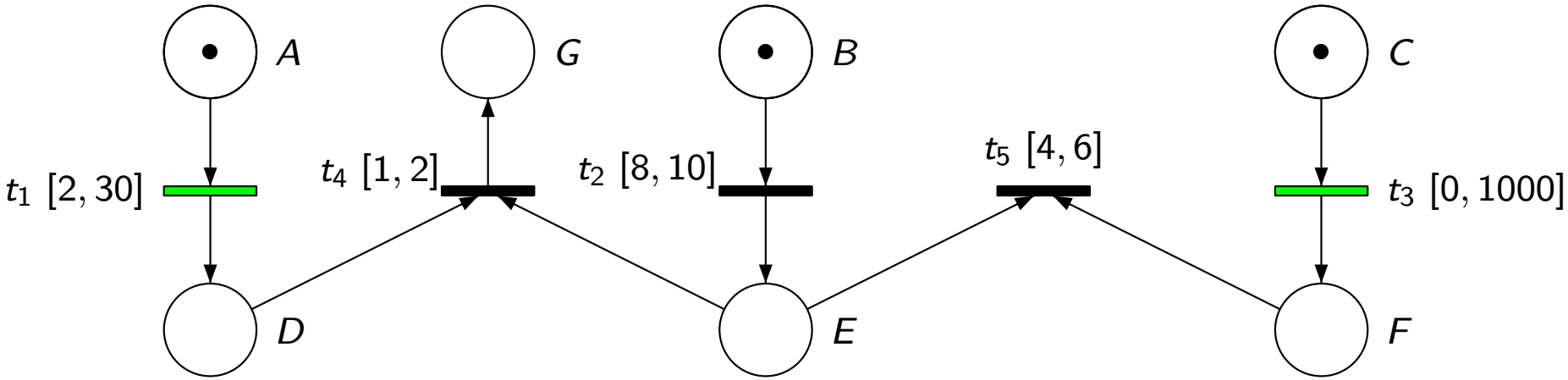
Exemples



Considérons le réseau $\mathcal{N}$ avec $T_r = \{t_1, t_3\}$ et la formule $\forall\Box(M(G) = 0)$

Le résultat est : $\{\alpha_3 \leq \beta_3 \leq \alpha_1 - 6\} \wedge \{\alpha_1 \geq 16\} \wedge \{\alpha_1 \leq \beta_1 \leq 30\}$

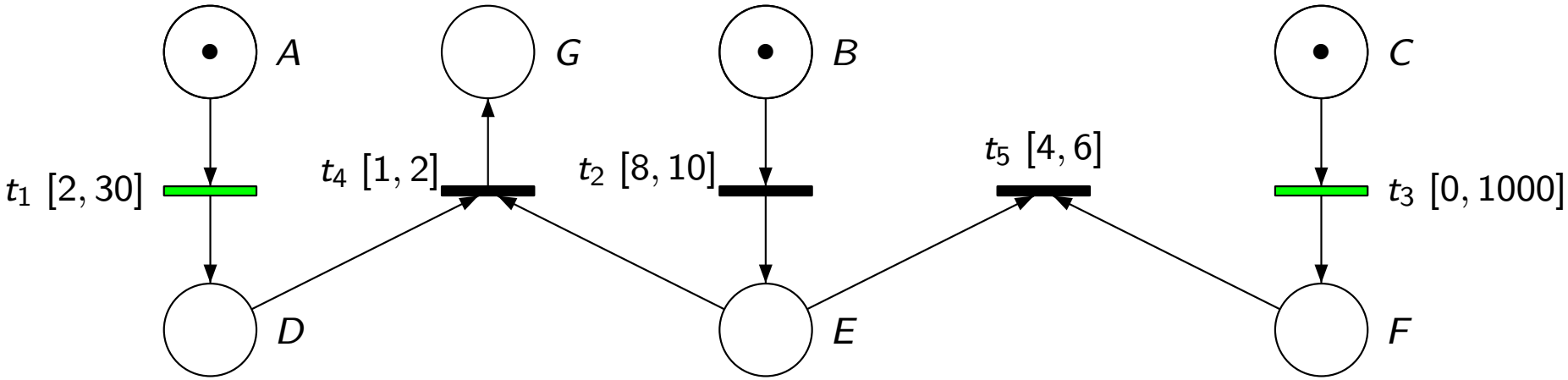
Exemples



Considérons le réseau $\mathcal{N}$ avec $T_r = \{t_1, t_3\}$ et la formule $\forall\Box(M(G) = 0)$
Le résultat est : $\{\alpha_3 \leq \beta_3 \leq \alpha_1 - 6\} \wedge \{\alpha_1 \geq 16\} \wedge \{\alpha_1 \leq \beta_1 \leq 30\}$

et avec la formule $\forall\Diamond(M(G) > 0)$

Exemples



Considérons le réseau $\mathcal{N}$ avec $T_r = \{t_1, t_3\}$ et la formule $\forall\Box(M(G) = 0)$
Le résultat est : $\{\alpha_3 \leq \beta_3 \leq \alpha_1 - 6\} \wedge \{\alpha_1 \geq 16\} \wedge \{\alpha_1 \leq \beta_1 \leq 30\}$

et avec la formule $\forall\Diamond(M(G) > 0)$
Le résultat est : $(\{\beta_1 \leq \alpha_3 + 1\} \vee \{\beta_1 \leq 9\}) \wedge \{2 \leq \alpha_1 \leq \beta_1 \leq 30\}$

Conclusion

Conclusion :

- SCP_{φ} est décidable. (Une sous classe décidable du model-checking paramétrique)

Conclusion

Conclusion :

- SCP_{φ} est décidable. (Une sous classe décidable du model-checking paramétrique)
- un algorithme symbolique

Conclusion

Conclusion :

- SCP_{φ} est décidable. (Une sous classe décidable du model-checking paramétrique)
- un algorithme symbolique
- une implémentation dans l'outil **Roméo**

Perpectives :

Conclusion

Conclusion :

- SCP_{φ} est décidable. (Une sous classe décidable du model-checking paramétrique)
- un algorithme symbolique
- une implémentation dans l'outil **Roméo**

Perpectives :

- Ajouter ∞ en borne max des transitions

Conclusion

Conclusion :

- SCP_{φ} est décidable. (Une sous classe décidable du model-checking paramétrique)
- un algorithme symbolique
- une implémentation dans l'outil **Roméo**

Perpectives :

- Ajouter ∞ en borne max des transitions
- Améliorer l'efficacité de l'implémentation