Accélération abstraite pour l'amélioration de la précision en Analyse des Relations Linéaires

Soutenance de thèse de Laure Gonnord

VERIMAG/UJF
Grenoble, France

Jury :

Yves Ledru
François Irigoin
Philippe Schnoebelen
Thomas Reps
Bertrand Jeannet
Nicolas Halbwachs

Contexte (1)

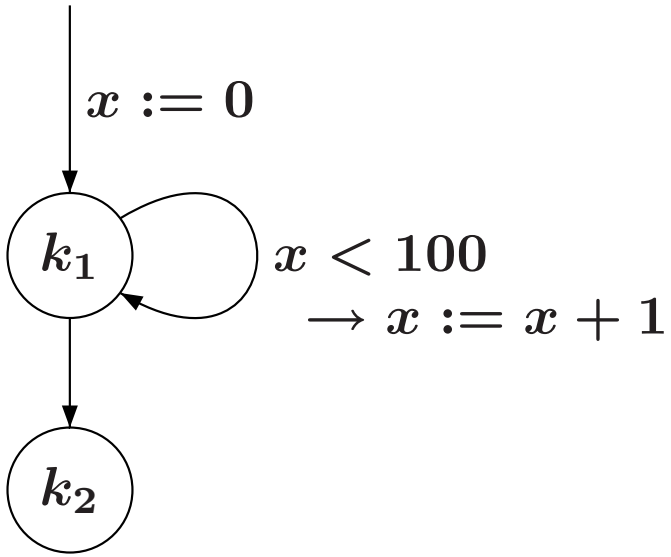
Vérification de propriétés de programmes **à nombre d'états infini.**

- Propriétés de **sûreté**.
- Ensemble infini d'états : propriétés en général indécidables.
  - Décision interactive.
  - Classes restreintes + abstractions.
  - Classe indécidable ou trop coûteuse : approximations, ici vérification « conservative » (surapproximations).

Contexte (2)

Programme numérique et **Graphe de Flot de Contrôle** (GFC) :

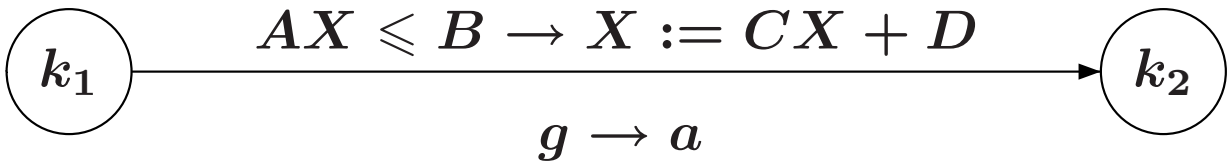
```
x := 0
while  x<100 do
    x := x+1
end
```



Invariant en k_2 : $0 \leqslant x \leqslant 100$.

Contexte (3)

Vérification de propriétés **numériques** sur des GFC avec conditions et actions **affines** :

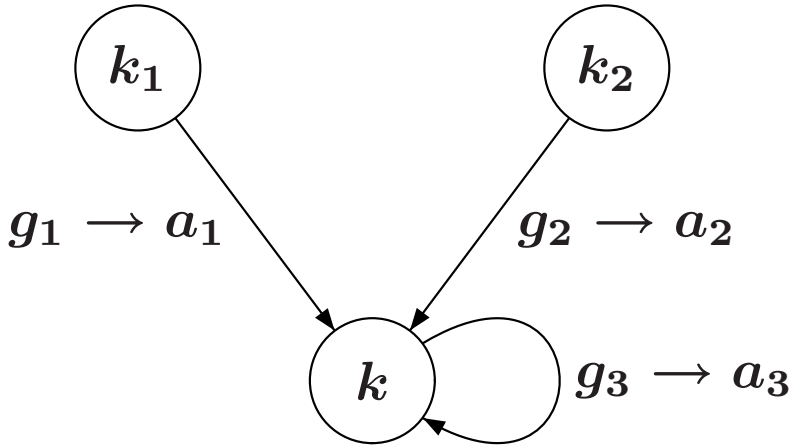


- A matrice, B vecteur.
- C matrice, D vecteur.
- Exemple : $x \leqslant 3 \wedge y + z \leqslant 8 \rightarrow z := 2x + 10; y := -5.2$

Propriétés numériques : inéquations linéaires.

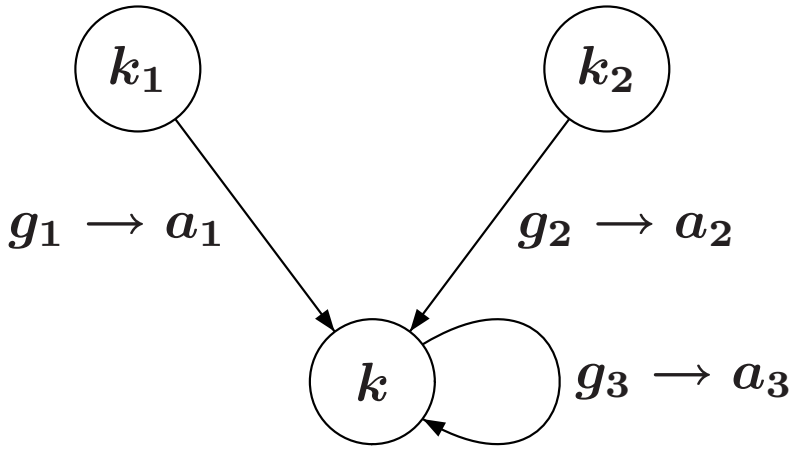
Contexte (4)

$\mathcal{A}_k$ désigne l'ensemble des valuations au point de contrôle k :



Contexte (4)

$\mathcal{A}_k$ désigne l'ensemble des valuations au point de contrôle k :

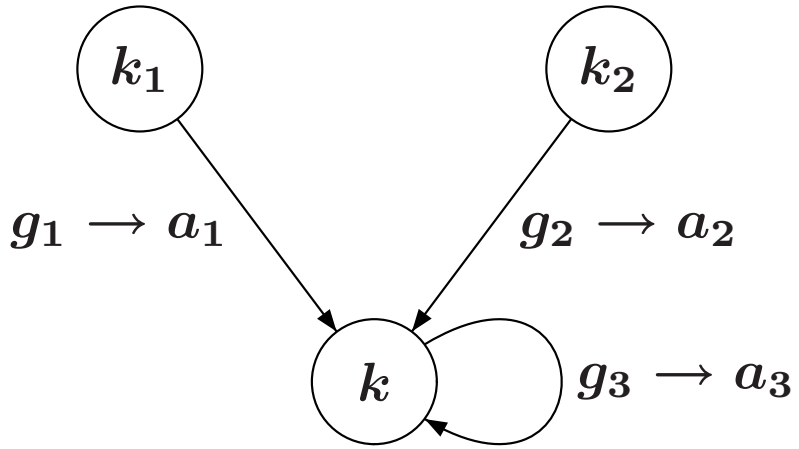


$$\mathcal{A}_k = a_1(\mathcal{A}_{k_1} \cap g_1) \cup a_2(\mathcal{A}_{k_2} \cap g_2) \cup a_3(\mathcal{A}_k \cap g_3)$$

▶ Sémantique « collectrice » : système d'équations, plus petit **point fixe**. Résolution itérative.

Contexte (4)

$\mathcal{A}_k$ désigne l'ensemble des valuations au point de contrôle k :



$$\mathcal{A}_k = a_1(\mathcal{A}_{k_1} \cap g_1) \cup a_2(\mathcal{A}_{k_2} \cap g_2) \cup a_3(\mathcal{A}_k \cap g_3)$$

- Sémantique « collectrice » : système d'équations, plus petit **point fixe**. Résolution itérative.
- Problèmes rencontrés :
  - Représentation des ensembles de valuations (éventuellement infinis).
  - Convergence des analyses en un nombre fini d'itérations.

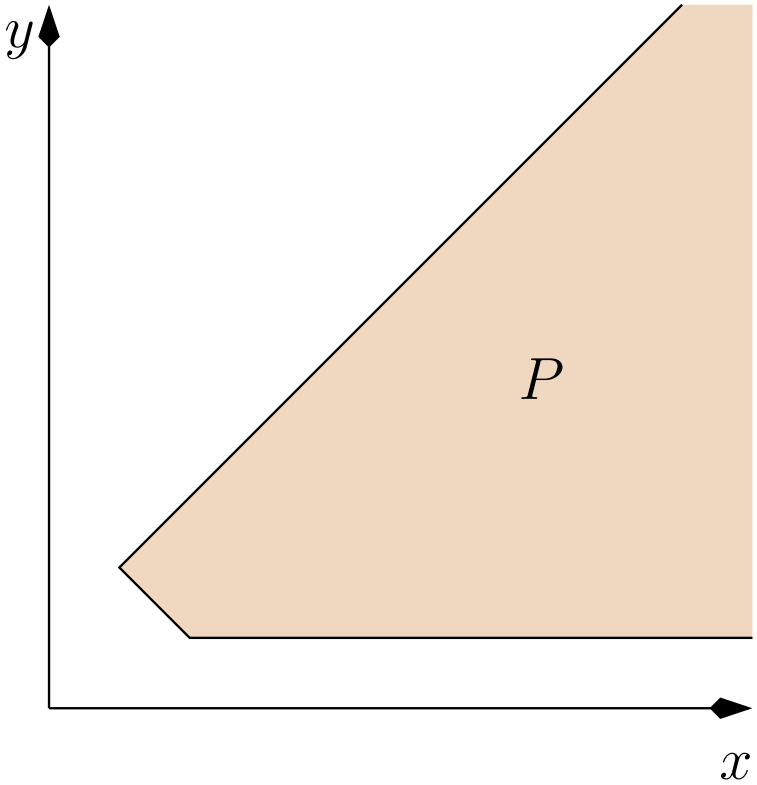
Contexte (5)

Méthode choisie : Analyse des relations linéaires [Cousot&Halbwachs 78] :

- Restriction à des propriétés qui sont des inégalités linéaires.
- Convergence par un **opérateur d'élargissement** (extrapolation).

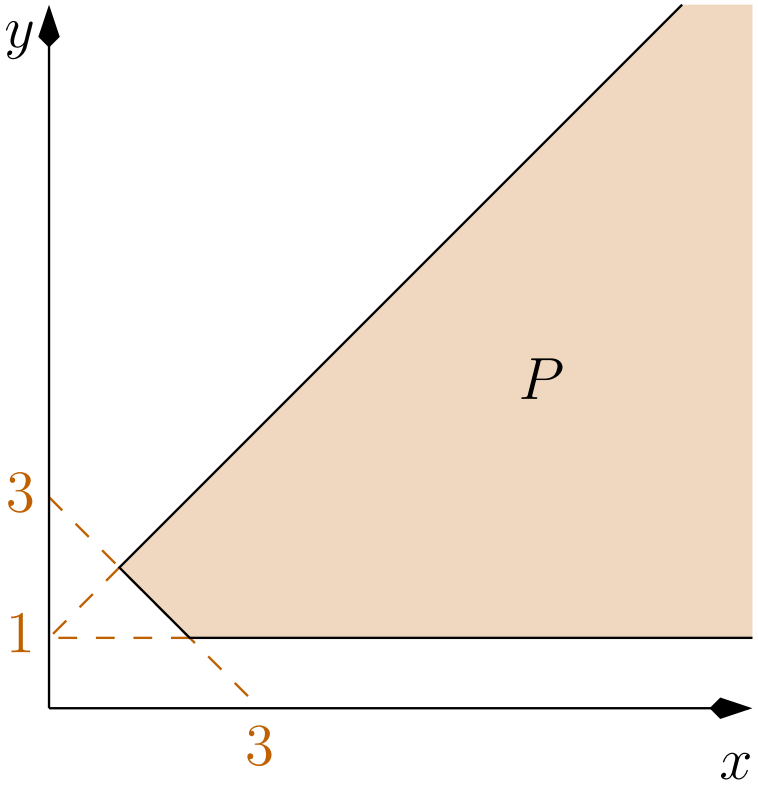
Le treillis des polyèdres convexes (1)

Double représentation



Le treillis des polyèdres convexes (1)

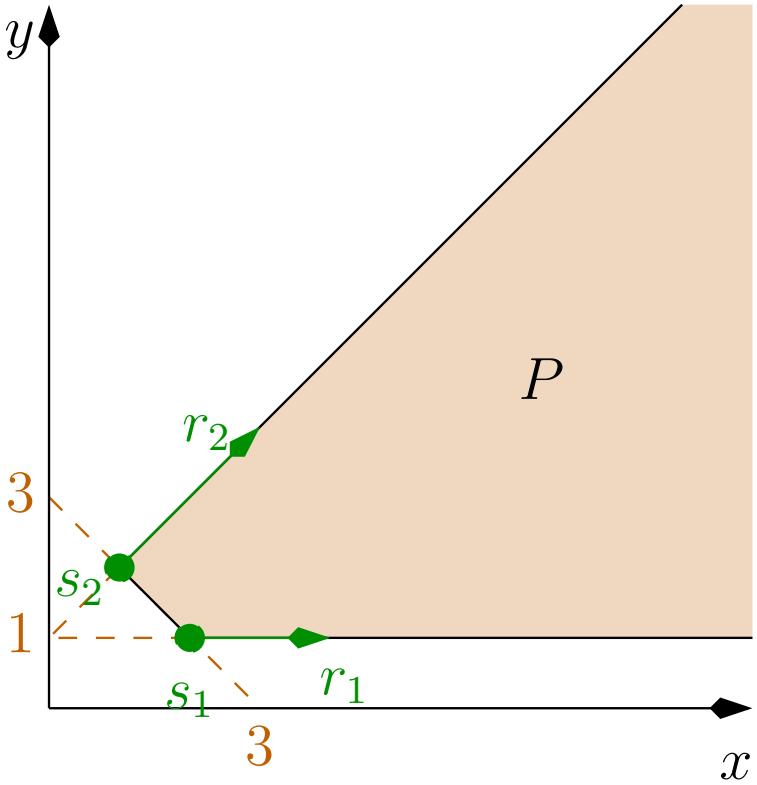
Double représentation



$$
\begin{aligned}
P &= \{(x,y) \mid \\
&\quad 1 \leqslant y \leqslant x+1 \wedge x+y \geqslant 3\} \\
&= cons\{AX \leqslant b\}
\end{aligned}
$$

Le treillis des polyèdres convexes (1)

Double représentation

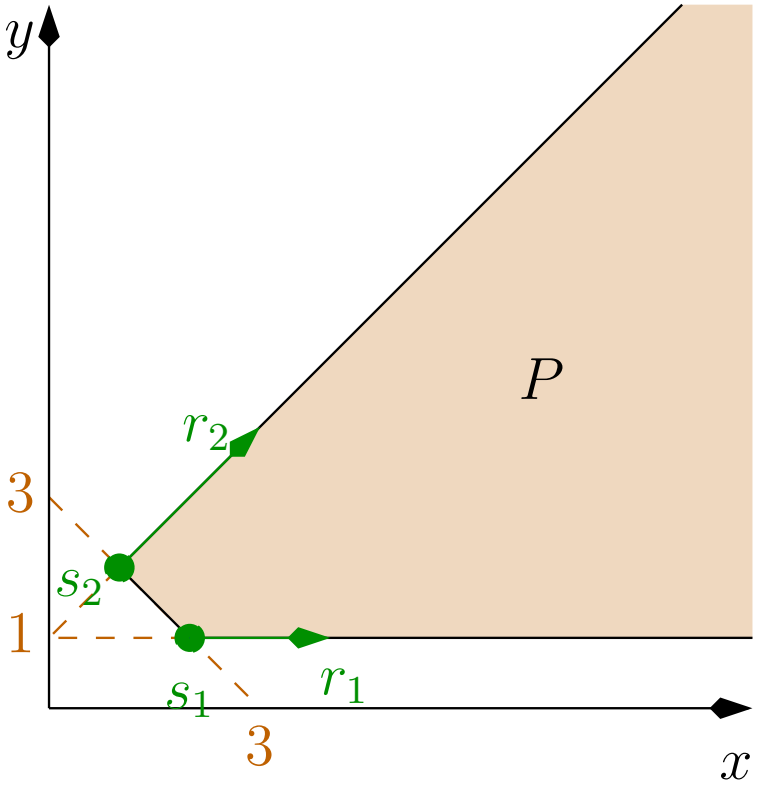


$$P = \{(x,y) \mid 1 \leqslant y \leqslant x+1 \wedge x+y \geqslant 3\} = cons\{AX \leqslant b\}$$

$$P = \{\lambda s_1 + (1-\lambda)s_2 + \mu_1 r_1 + \mu_2 r_2 \mid \lambda \in [0,1],\ \mu_1, \mu_2 \geqslant 0\} = gen(S,R)\}$$

Le treillis des polyèdres convexes (1)

Double représentation



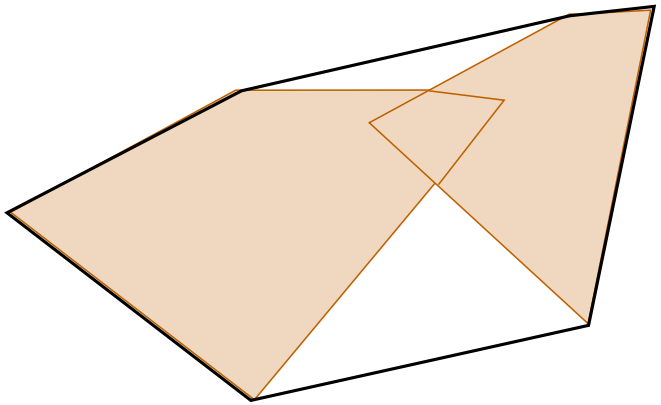
$$P = \{(x,y) \mid 1 \leqslant y \leqslant x+1 \wedge x+y \geqslant 3\} = cons\{AX \leqslant b\}$$

$$P = \{\lambda s_1 + (1-\lambda)s_2 + \mu_1 r_1 + \mu_2 r_2 \mid \lambda \in [0,1],\ \mu_1, \mu_2 \geqslant 0\} = gen(S,R)\}$$

- Deux représentations **finies** et complémentaires.
- Algorithmique disponible.

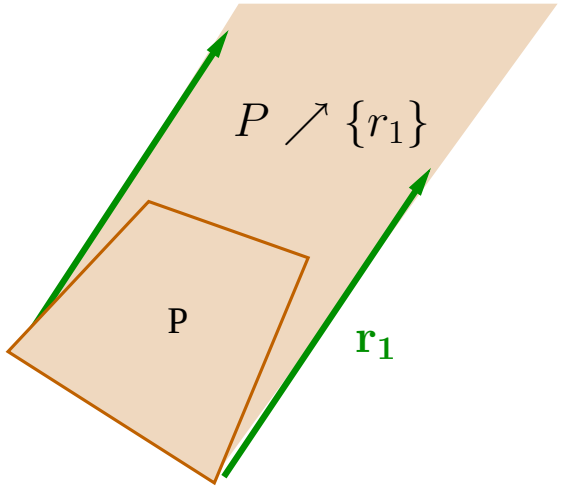
Le treillis des polyèdres convexes (3)

- Intersection, test du vide.
- Transformation affine : $a(P) = \{CX + D \mid X \in P\}$.
- Union convexe (perte de précision) :



Le treillis des polyèdres convexes (3)

- Intersection, test du vide.
- Transformation affine : $a(P) = \{CX + D \mid X \in P\}$.
- Union convexe (perte de précision) :
- Ajout de rayons $P \nearrow R = \{X + \sum_{r_j \in R} \mu_j r_j \mid X \in P, \mu_j \in \mathbb{Q}^+\}$

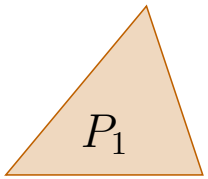


Le treillis des polyèdres convexes (2)

Élargissement : $P_1 \nabla P_2$: extrapolation de la limite.

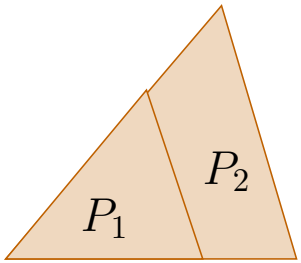
Le treillis des polyèdres convexes (2)

Élargissement : $P_1 \nabla P_2$: extrapolation de la limite.



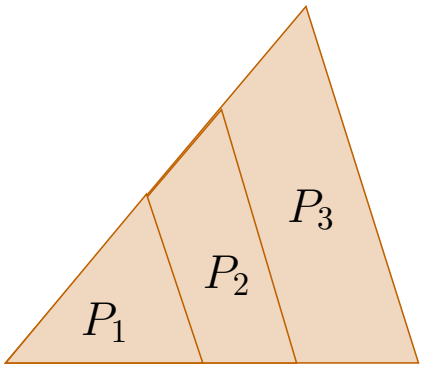
Le treillis des polyèdres convexes (2)

Élargissement : $P_1 \nabla P_2$: extrapolation de la limite.



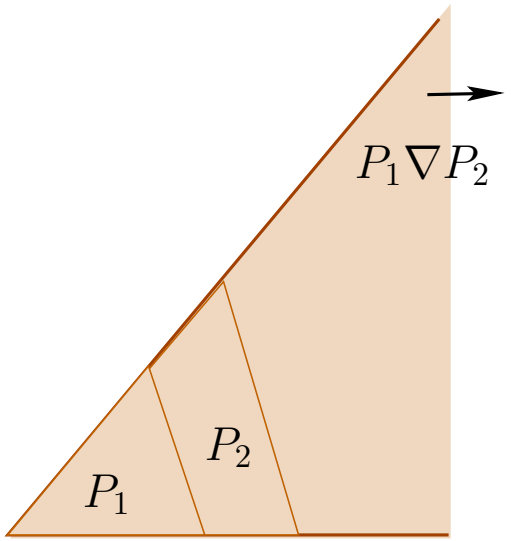
Le treillis des polyèdres convexes (2)

Élargissement : $P_1 \nabla P_2$: extrapolation de la limite.

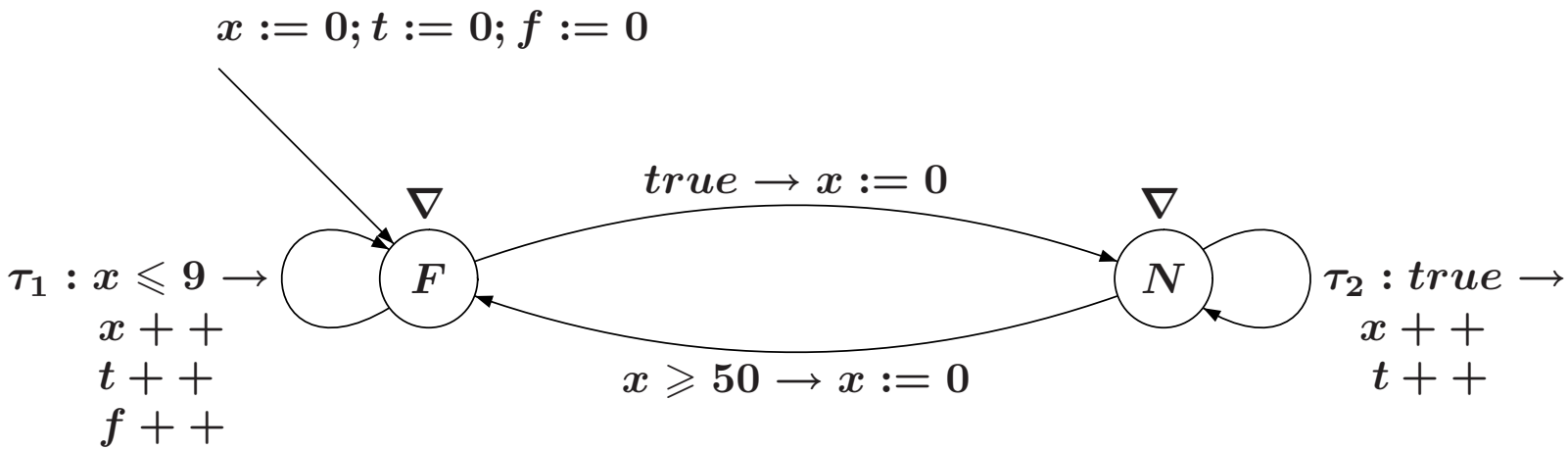


Le treillis des polyèdres convexes (2)

Élargissement : $P_1 \nabla P_2$: extrapolation de la limite.

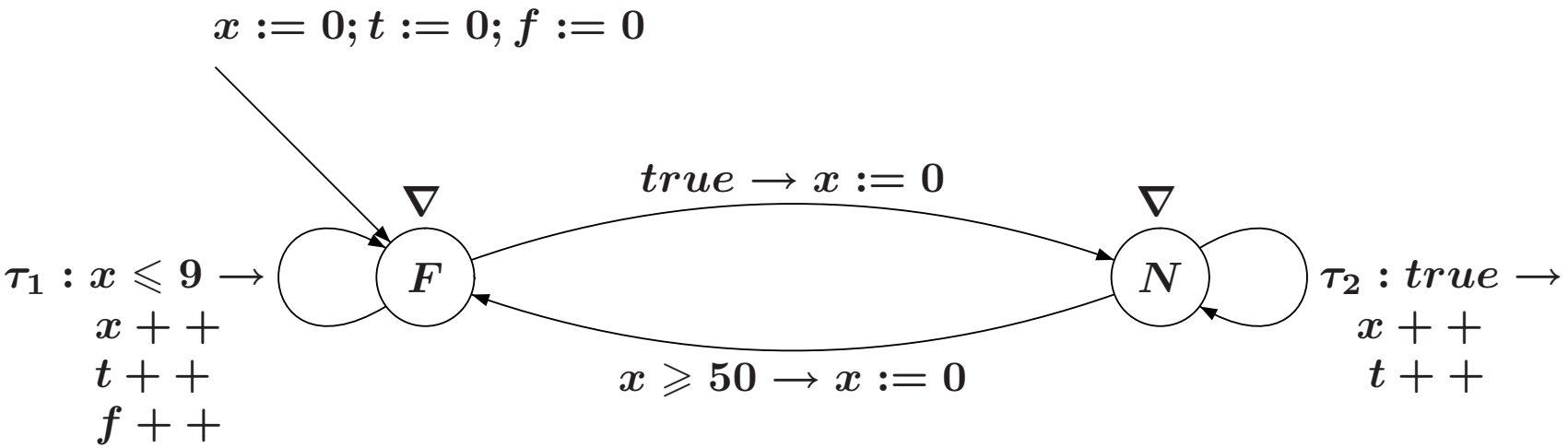


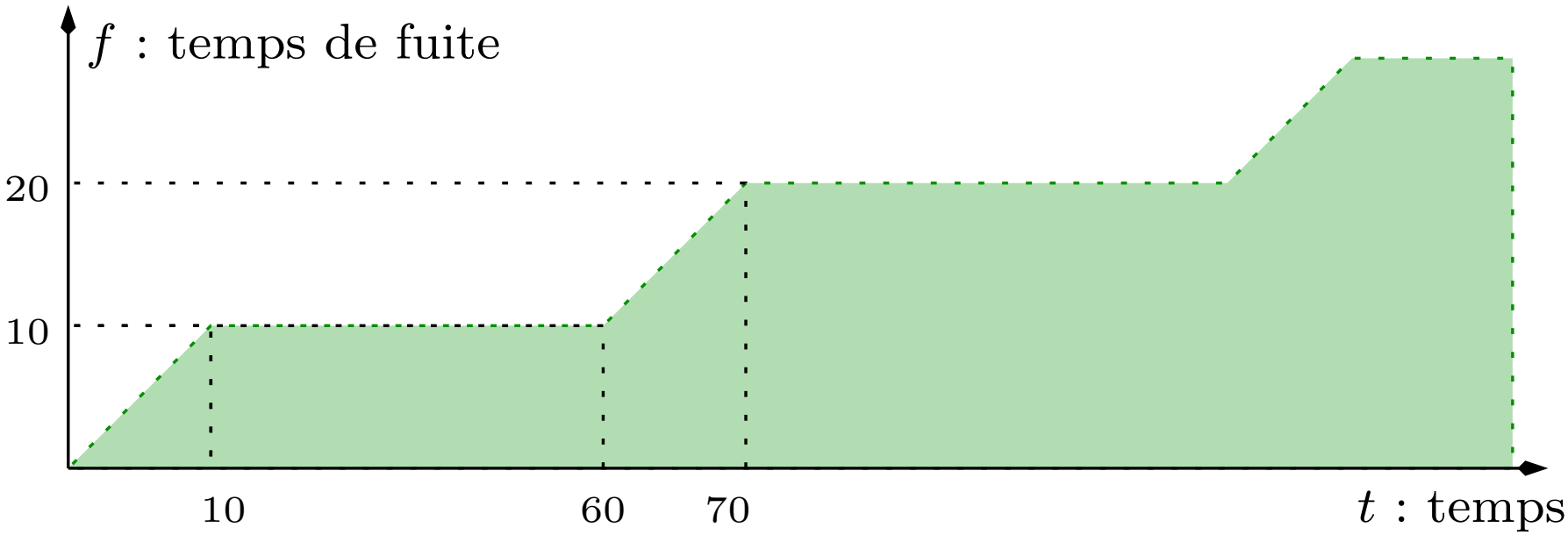
L'exemple de la chaudière - 1



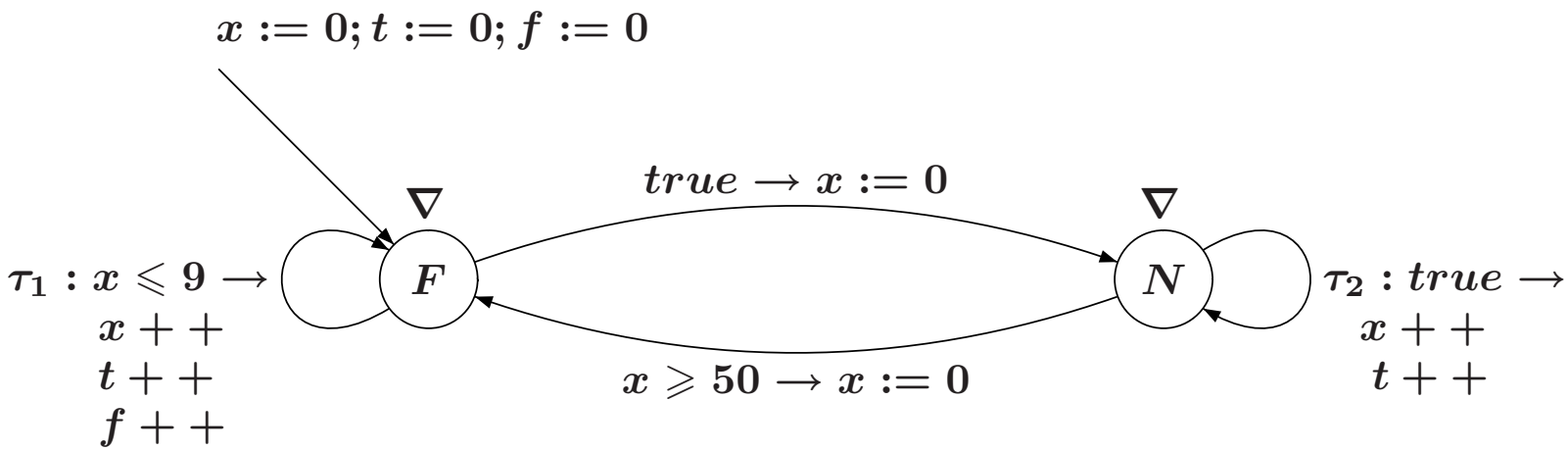
- f le temps de fuite global.
- t le temps global.
- x variable locale.

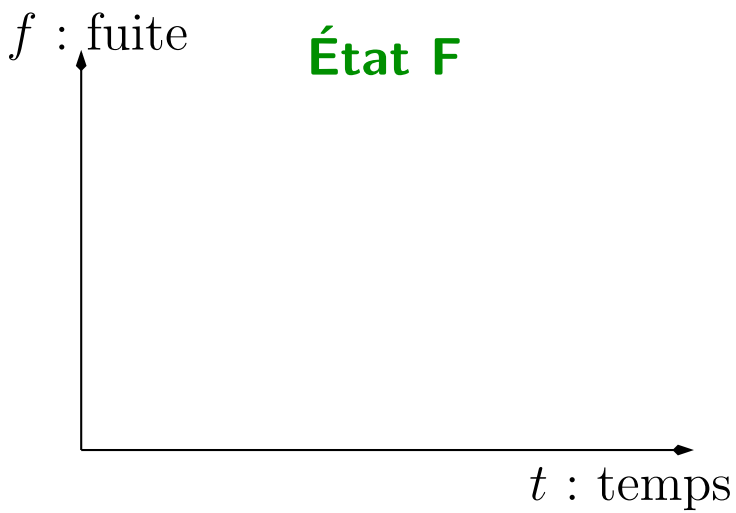
La chaudière 2 - Comportement réel

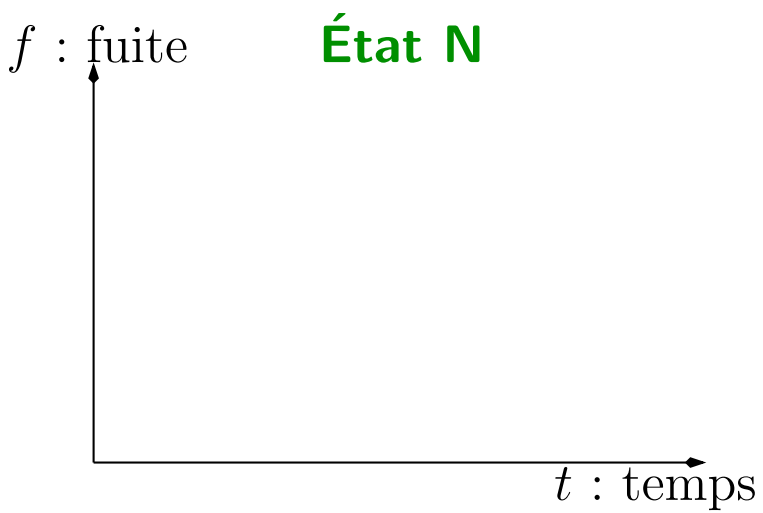




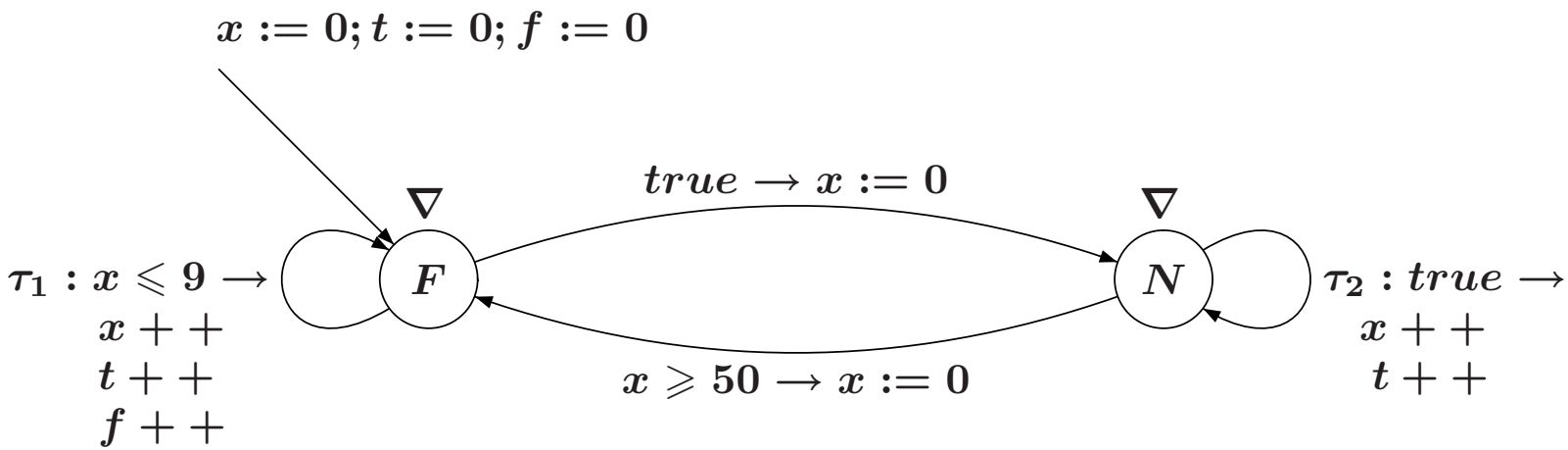
La chaudière 3 - Analyse des Relations Linéaires simple

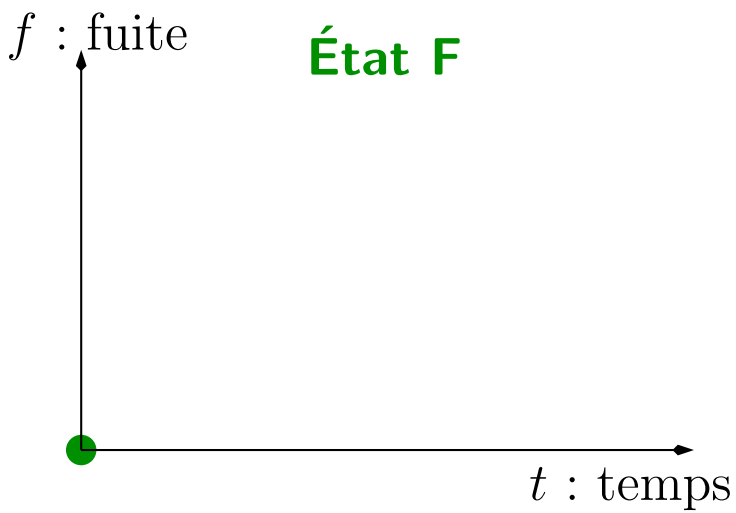


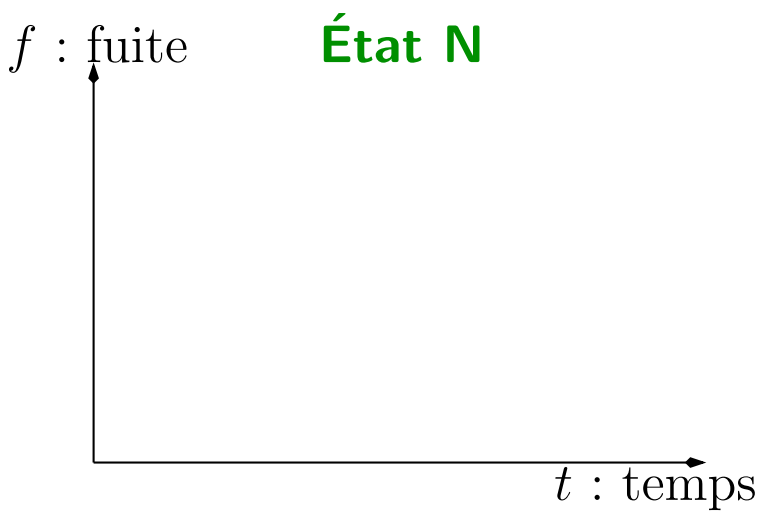




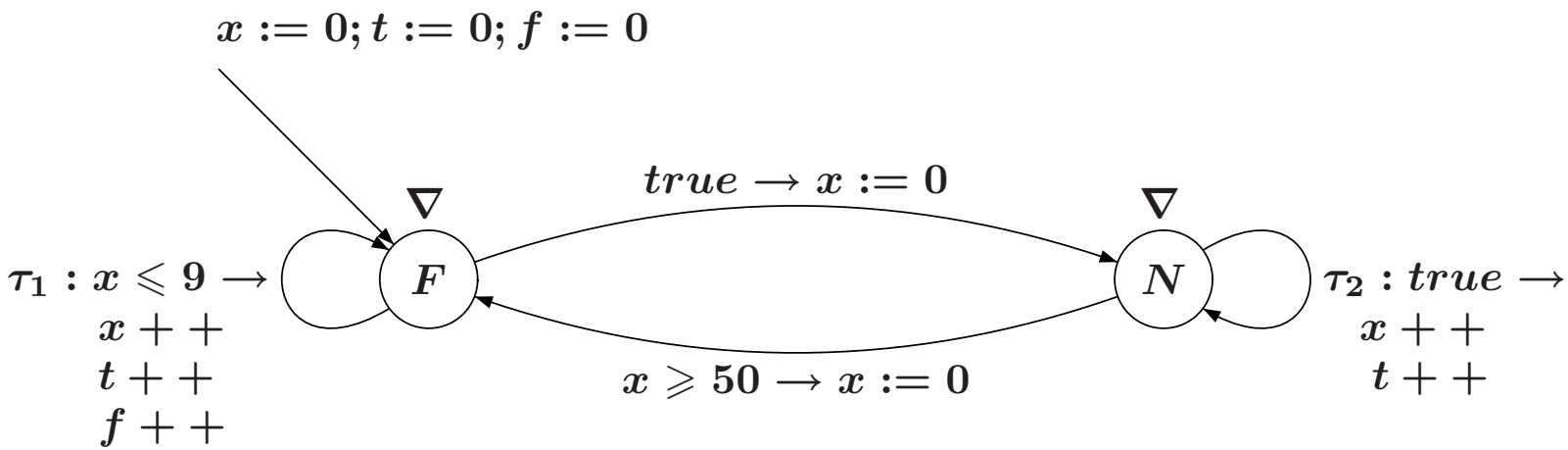
La chaudière 3 - Analyse des Relations Linéaires simple

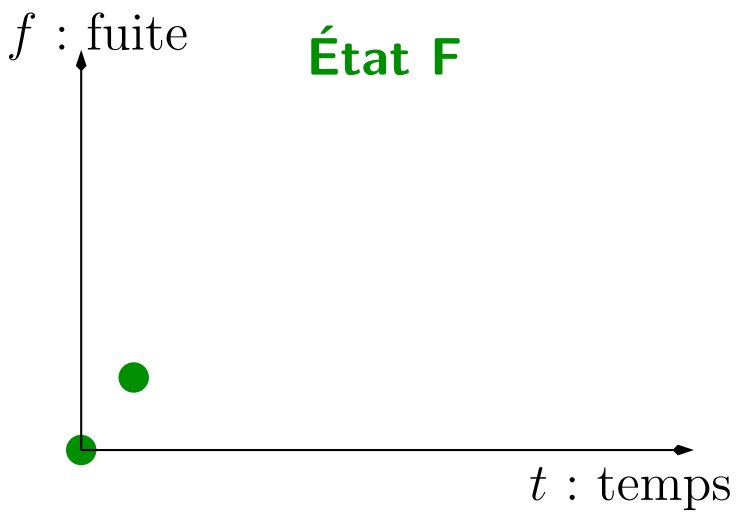


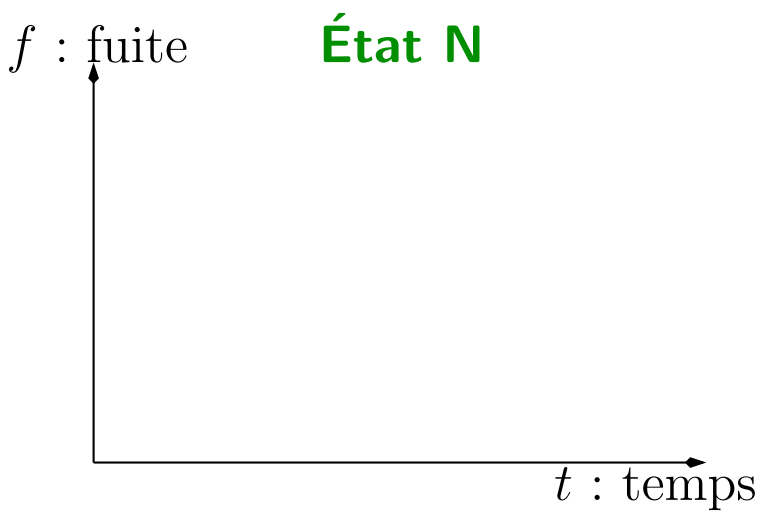




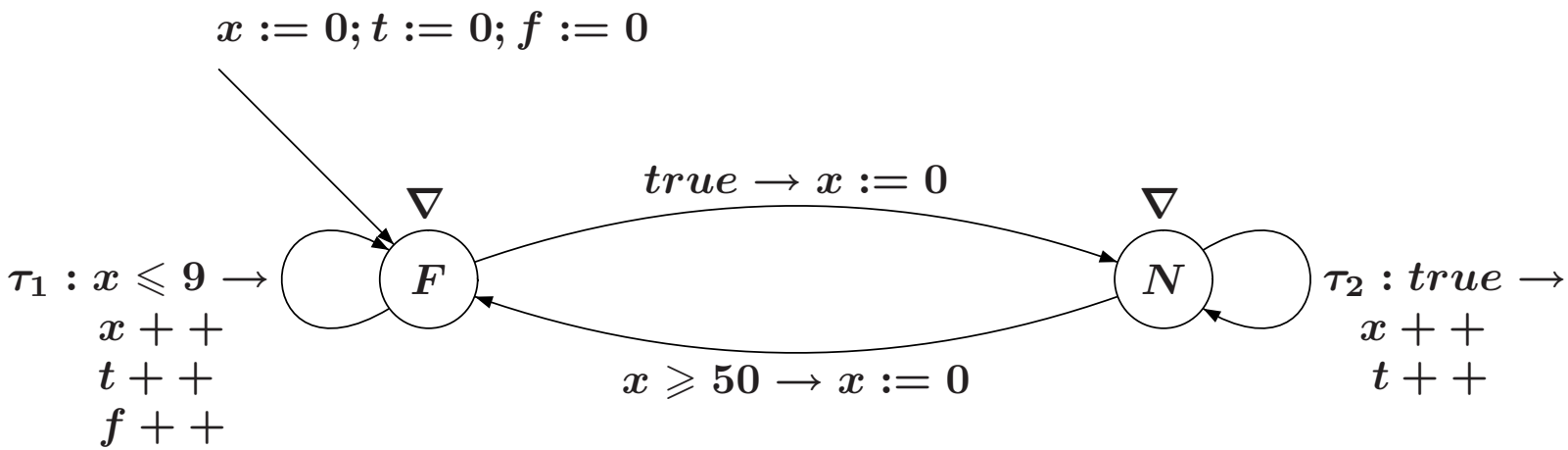
La chaudière 3 - Analyse des Relations Linéaires simple

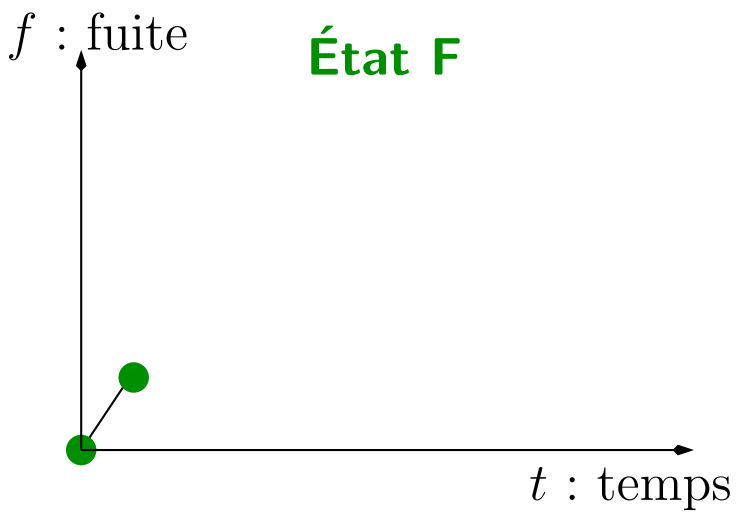


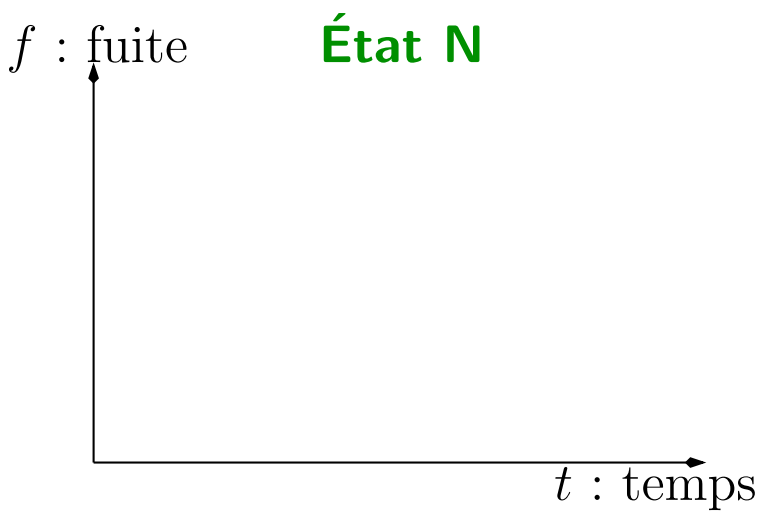




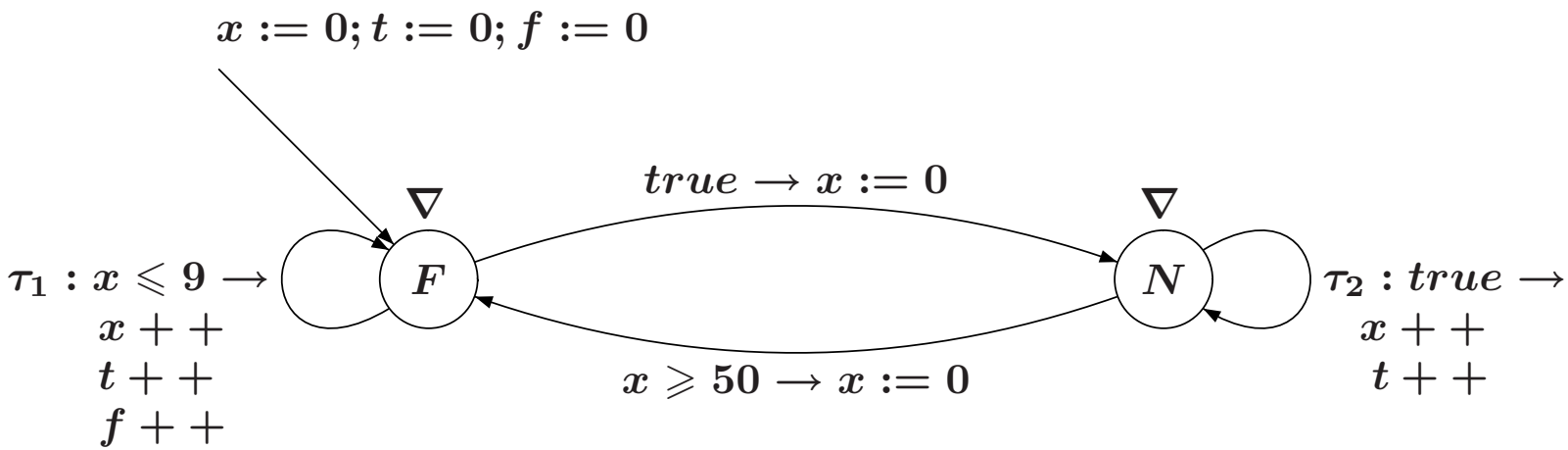
La chaudière 3 - Analyse des Relations Linéaires simple

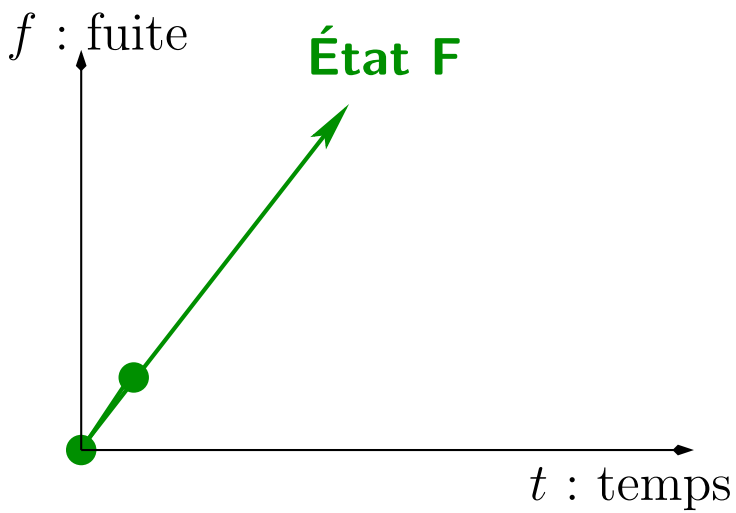


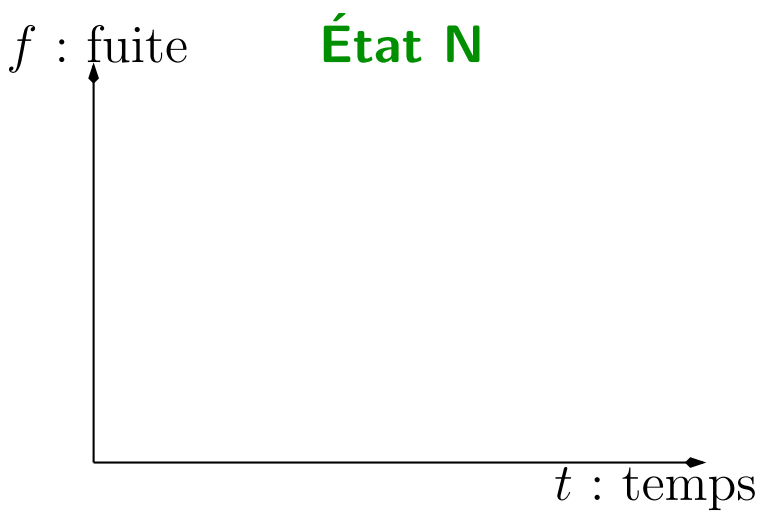




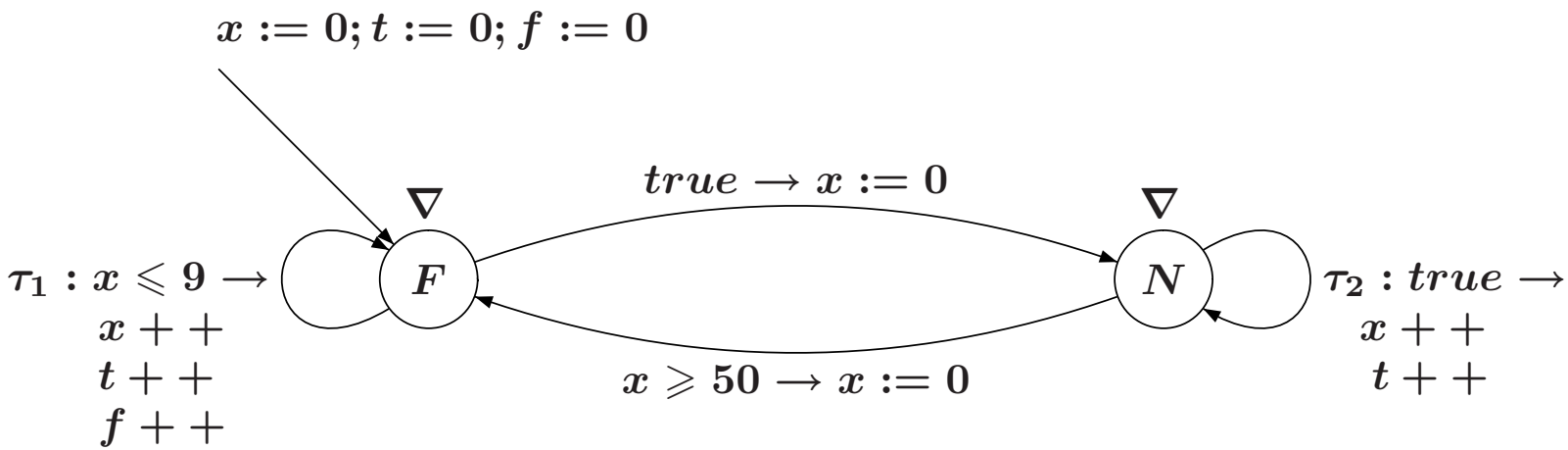
La chaudière 3 - Analyse des Relations Linéaires simple

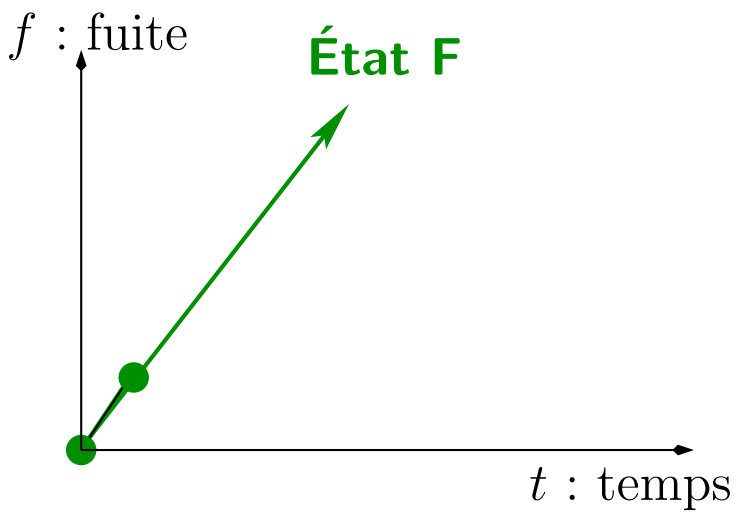


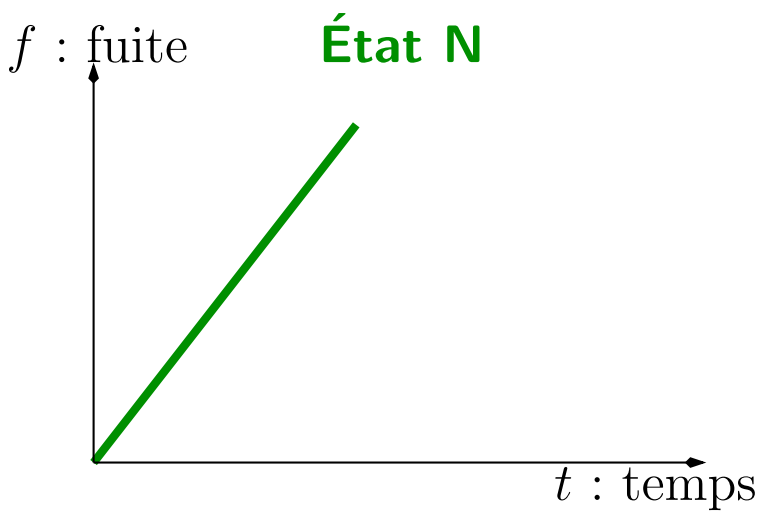




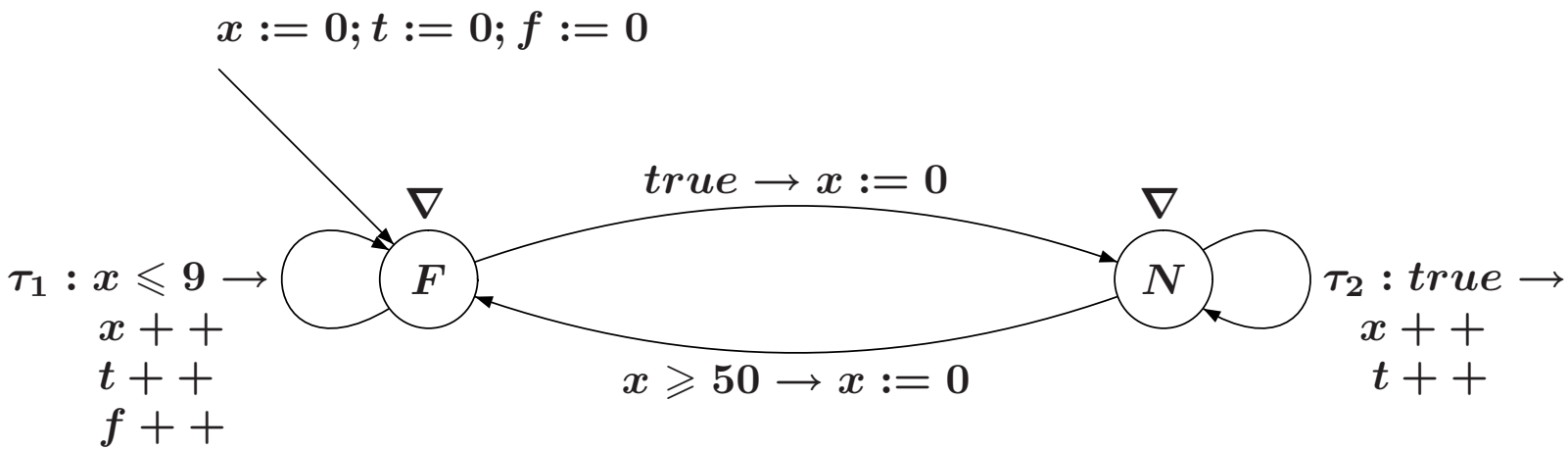
La chaudière 3 - Analyse des Relations Linéaires simple

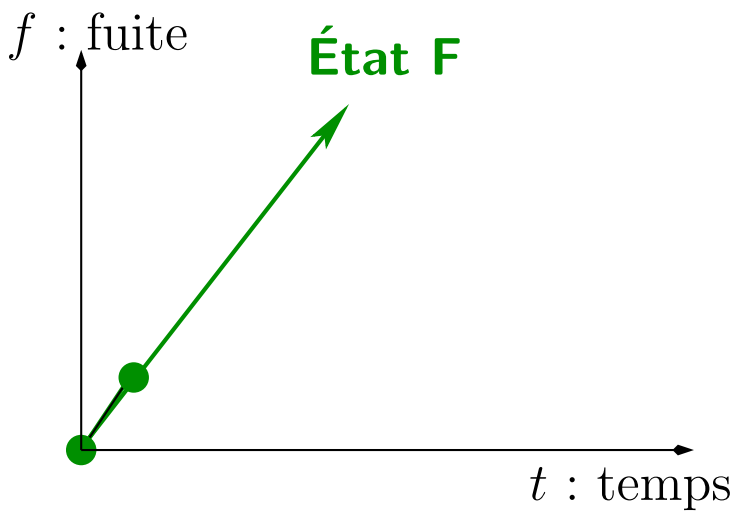


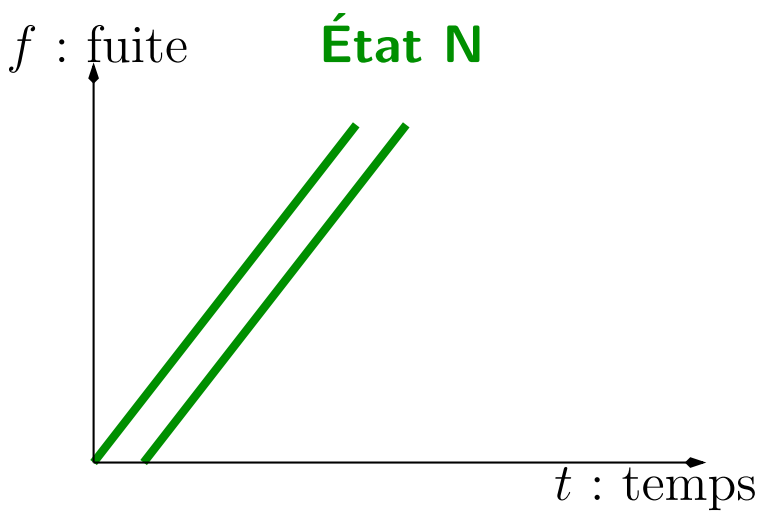




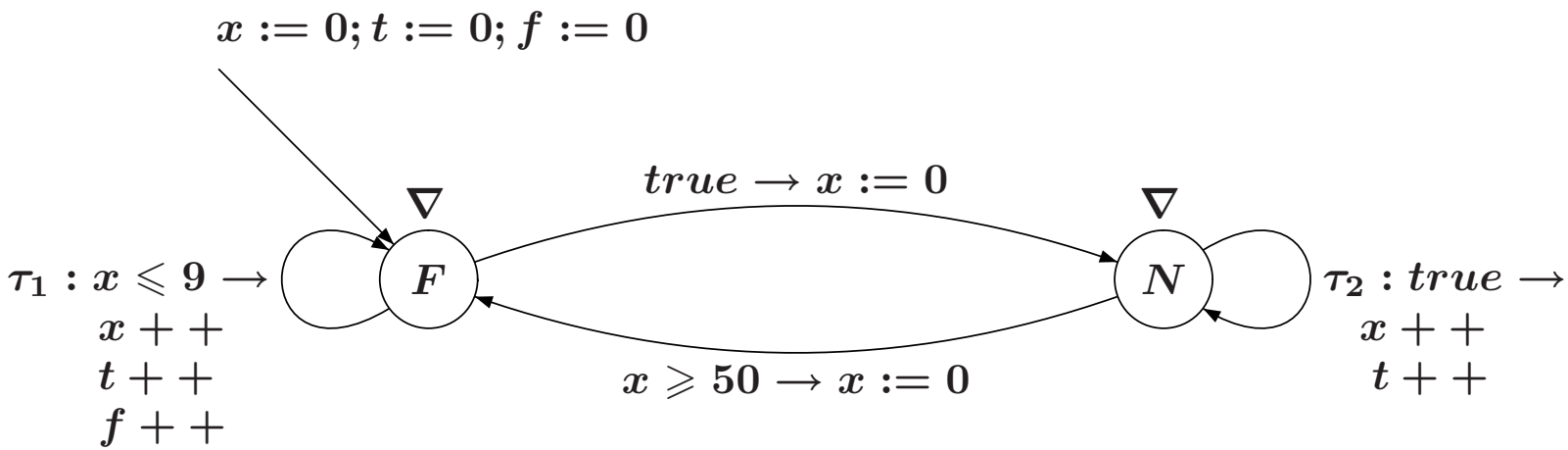
La chaudière 3 - Analyse des Relations Linéaires simple

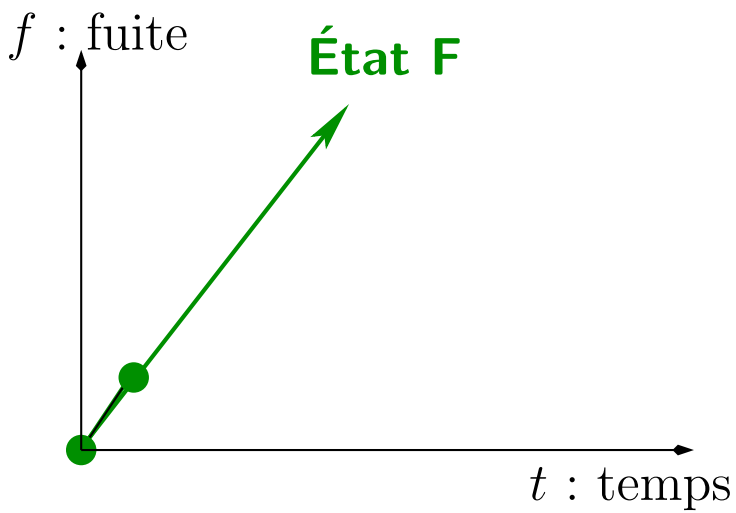


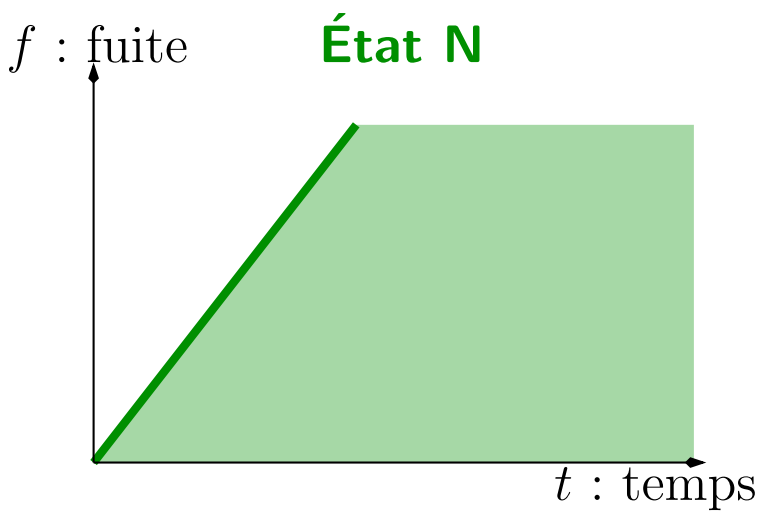




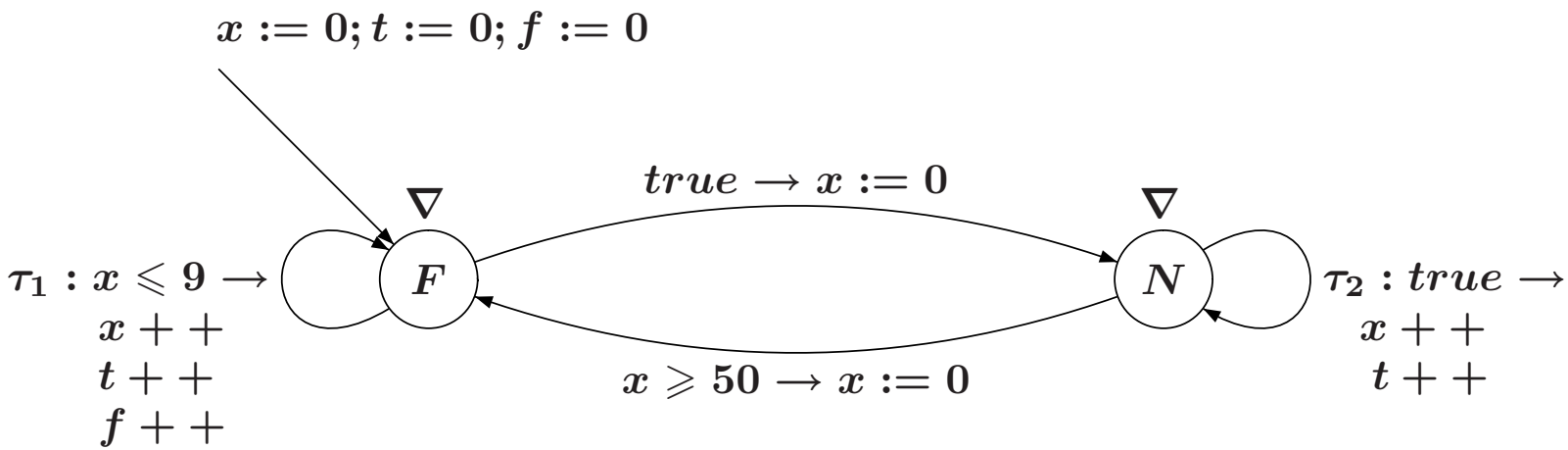
La chaudière 3 - Analyse des Relations Linéaires simple

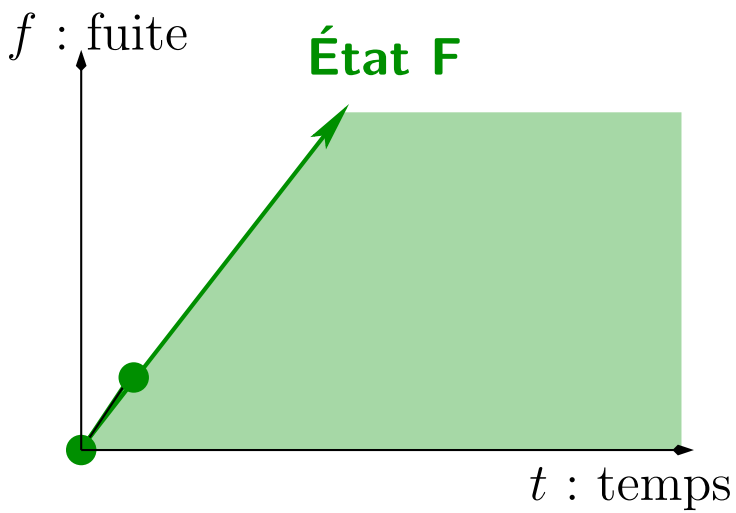


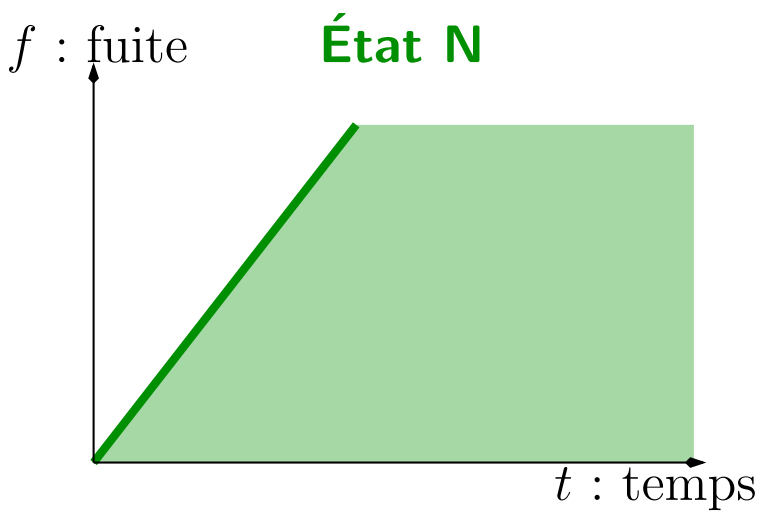




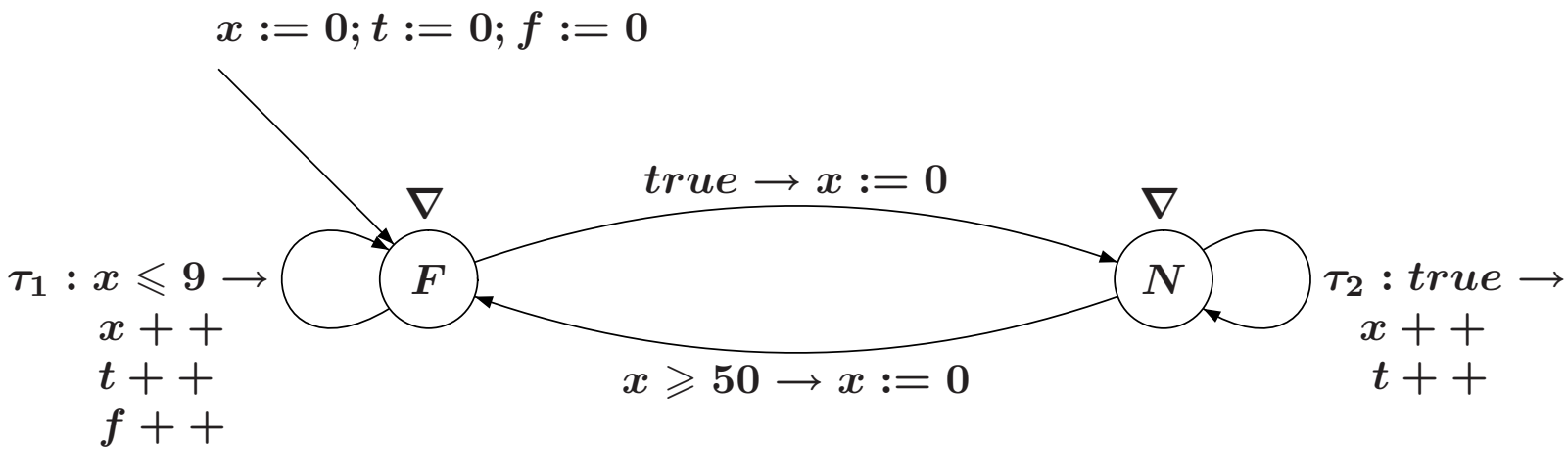
La chaudière 3 - Analyse des Relations Linéaires simple

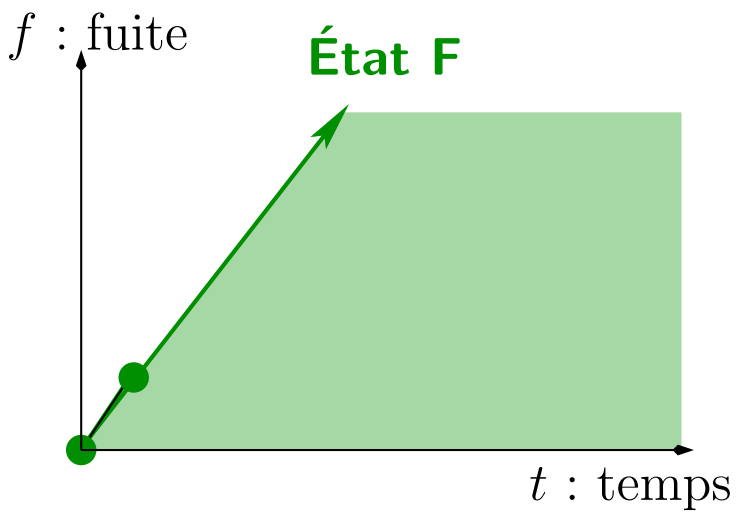


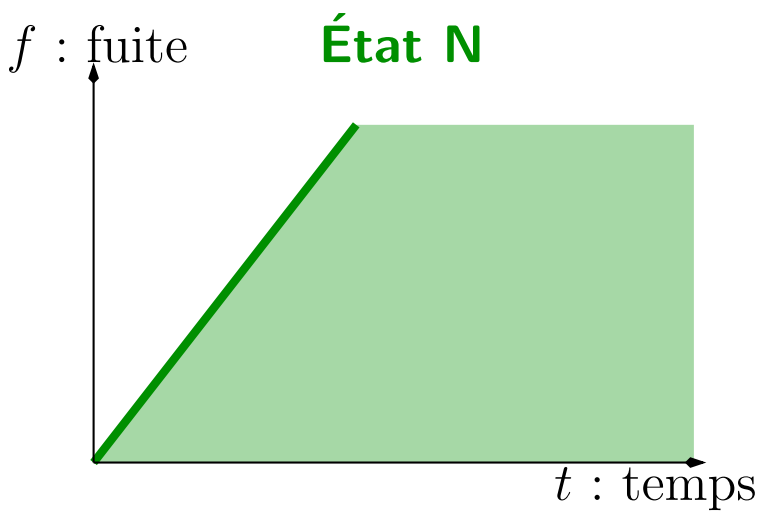




La chaudière 3 - Analyse des Relations Linéaires simple

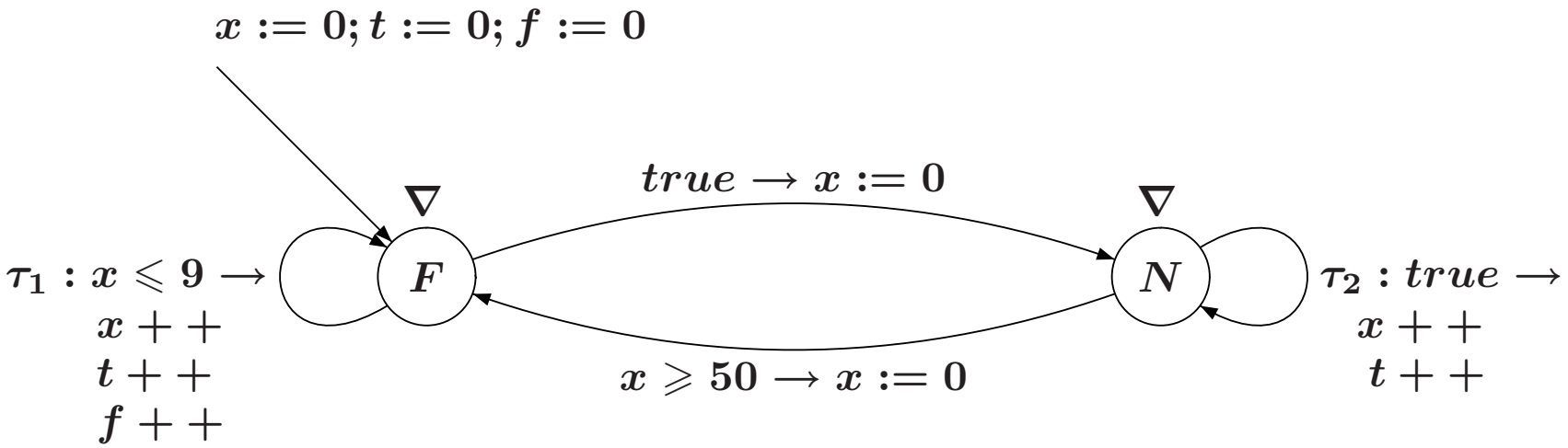


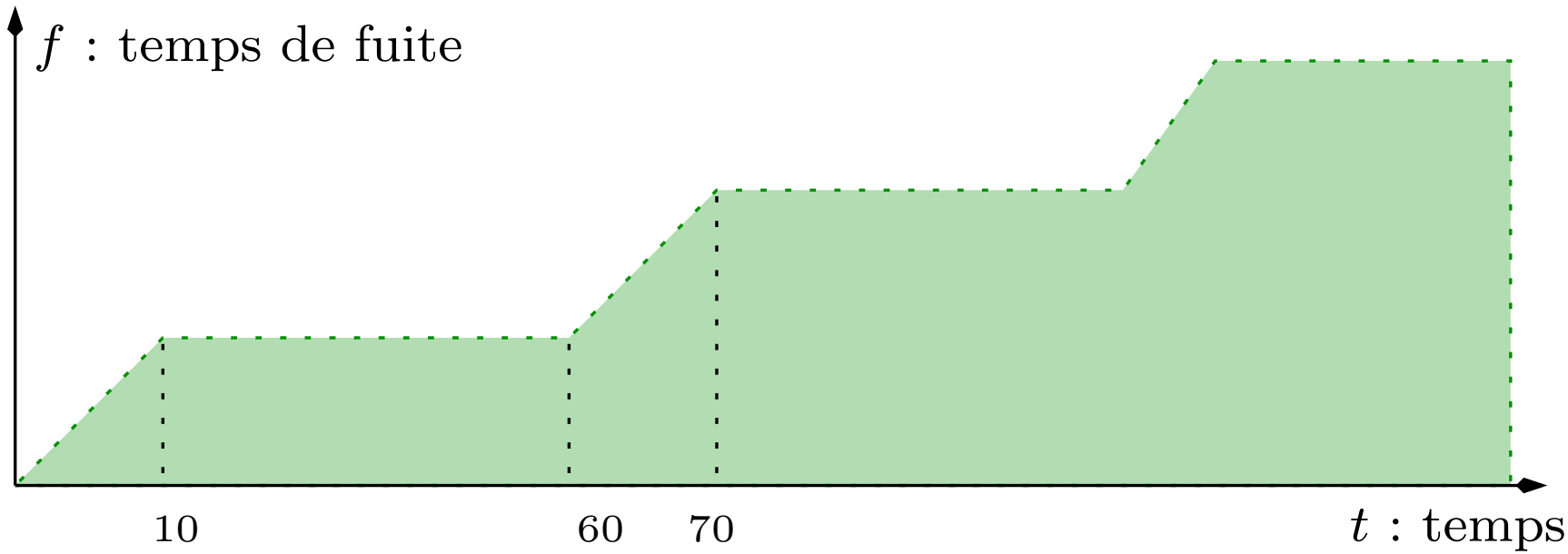




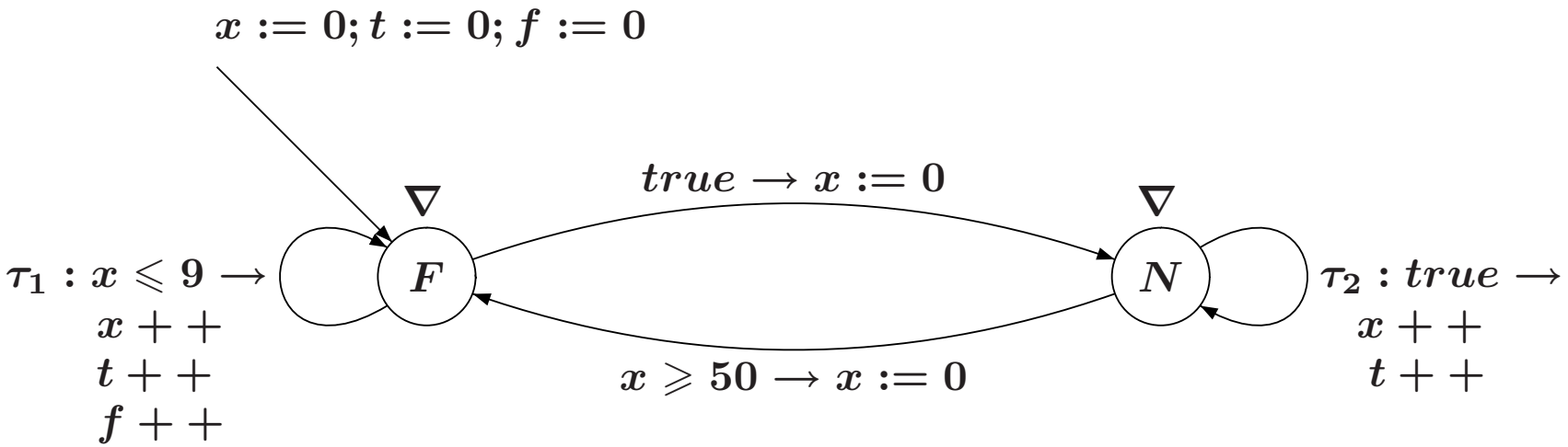
▶ Convergence, approximation supérieure, mais manque de précision.

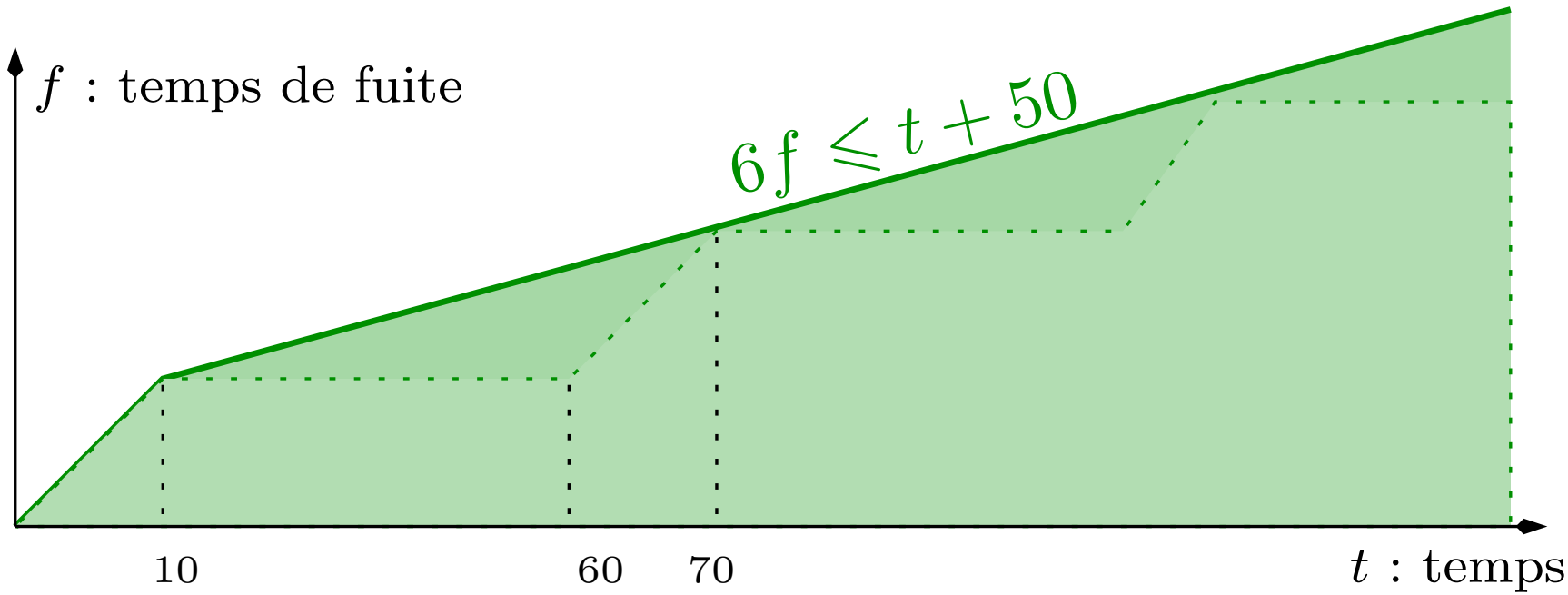
La chaudière - Invariant voulu





La chaudière - Invariant voulu





Amélioration de la précision

Sources d'approximation : union convexe et élargissement.

Amélioration de la précision

Sources d'approximation : union convexe et élargissement.

- Quelques méthodes existantes :
  - **Retarder** l'application de l'opérateur d'élargissement.
    Inconvénient : le coût.
  - **Améliorer** l'élargissement [Bagnara&Hill&Zafanella].
    Inconvénient : pas de résultat absolu sur la précision globale (et perte de performance).
  - **Alterner** élargissement et rétrécissement [Gopan&Reps : CAV 2006]. Peut être combinée avec notre méthode.

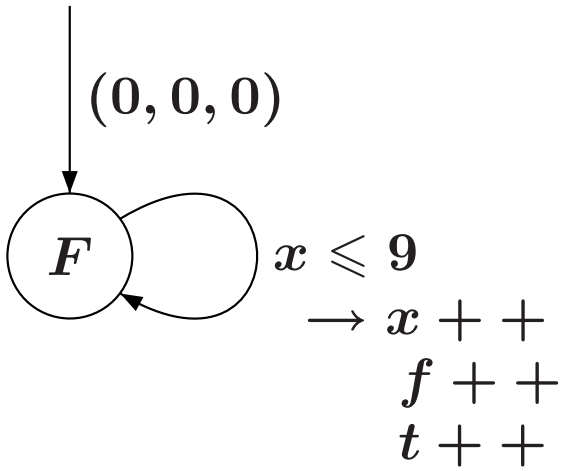
Amélioration de la précision

Sources d'approximation : union convexe et élargissement.

- Quelques méthodes existantes :
  - **Retarder** l'application de l'opérateur d'élargissement.
    Inconvénient : le coût.
  - **Améliorer** l'élargissement [Bagnara&Hill&Zafanella].
    Inconvénient : pas de résultat absolu sur la précision globale (et perte de performance).
  - **Alterner** élargissement et rétrécissement [Gopan&Reps : CAV 2006]. Peut être combinée avec notre méthode.
- Seule la première donne l'invariant cherché (retard = 60 itérations).

Exemple de la chaudière (3)

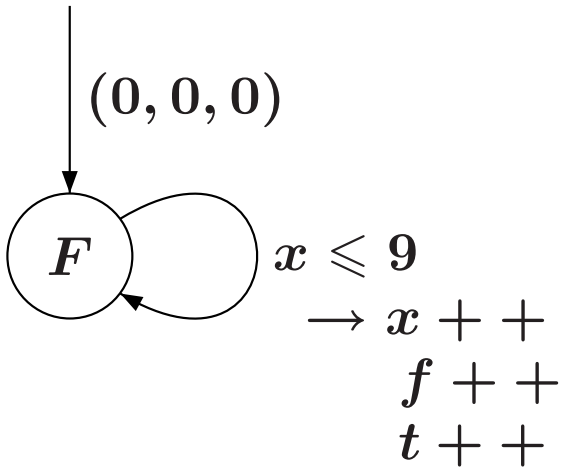
La boucle est « accélérable » :



- Effet **exact** : $\exists i \in \mathbb{N}, x = f = t = i, 0 \leqslant i \leqslant 10$

Exemple de la chaudière (3)

La boucle est « accélérable » :



- Effet **exact** : $\exists i \in \mathbb{N}, x = f = t = i, 0 \leqslant i \leqslant 10$
- **Méthodes d'accélération.**

Les méthodes d'accélération

Accélération

[Boigelot&Wolper, Common&Jurski, Finkel&Sutre&Leroux&Bardin&Schnoebelen]

- On calcule l'**effet exact** des boucles sur des ensembles d'**entiers**.
- Codage sous forme d'automates représentant des formules de Presburger ($< \mathbb{N}, \leqslant, +, \exists >$).

Les méthodes d'accélération

Accélération
[Boigelot&Wolper, Common&Jurski, Finkel&Sutre&Leroux&Bardin&Schnoebelen]

- On calcule l'**effet exact** des boucles sur des ensembles d'**entiers**.
- Codage sous forme d'automates représentant des formules de Presburger $(< \mathbb{N}, \leqslant, +, \exists >)$.

▶ Inconvénients : classes restreintes de programmes, haute complexité.

Accélération et Analyse des Relations Linéaires

Vers une accélération abstraite

- [Su/Wagner] **Résolution exacte du système abstrait** sur le treillis des intervalles. Pas d'élargissement.
- PIPS [Irigoin et al.] **Surapproximation** de la fermeture transitive de la *relation* de transition.

Accélération et Analyse des Relations Linéaires

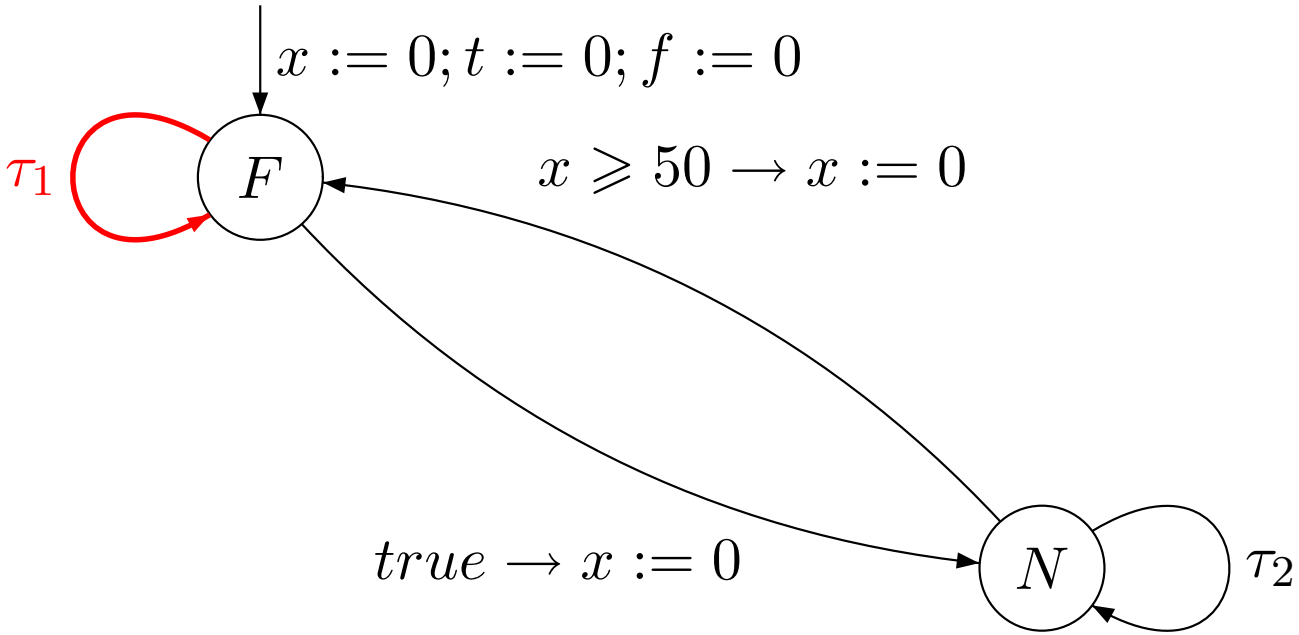
Vers une accélération abstraite

- [Su/Wagner] **Résolution exacte du système abstrait** sur le treillis des intervalles. Pas d'élargissement.
- PIPS [Irigoin et al.] **Surapproximation** de la fermeture transitive de la *relation* de transition.

Objectif de cette thèse : accélération « abstraite » **à faible coût** pour les polyèdres convexes, combinée avec l'élargissement.

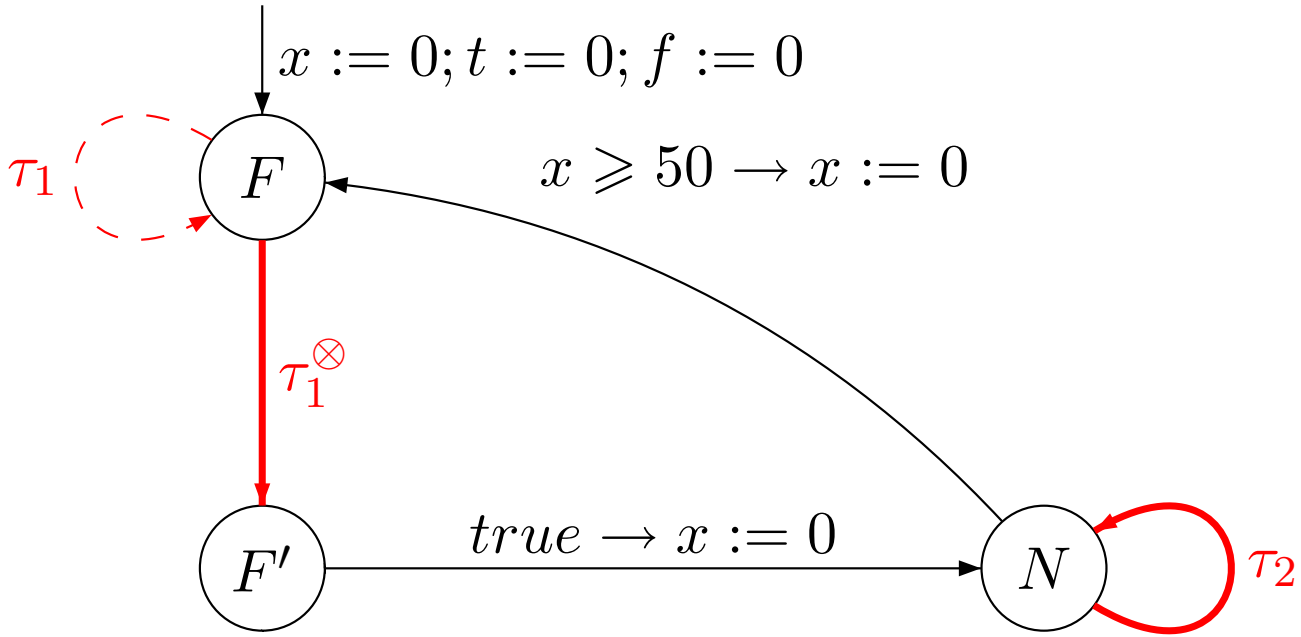
Accélération et Analyse des Relations Linéaires

On veut remplacer les boucles $(\tau_i : g_i \rightarrow a_i)$



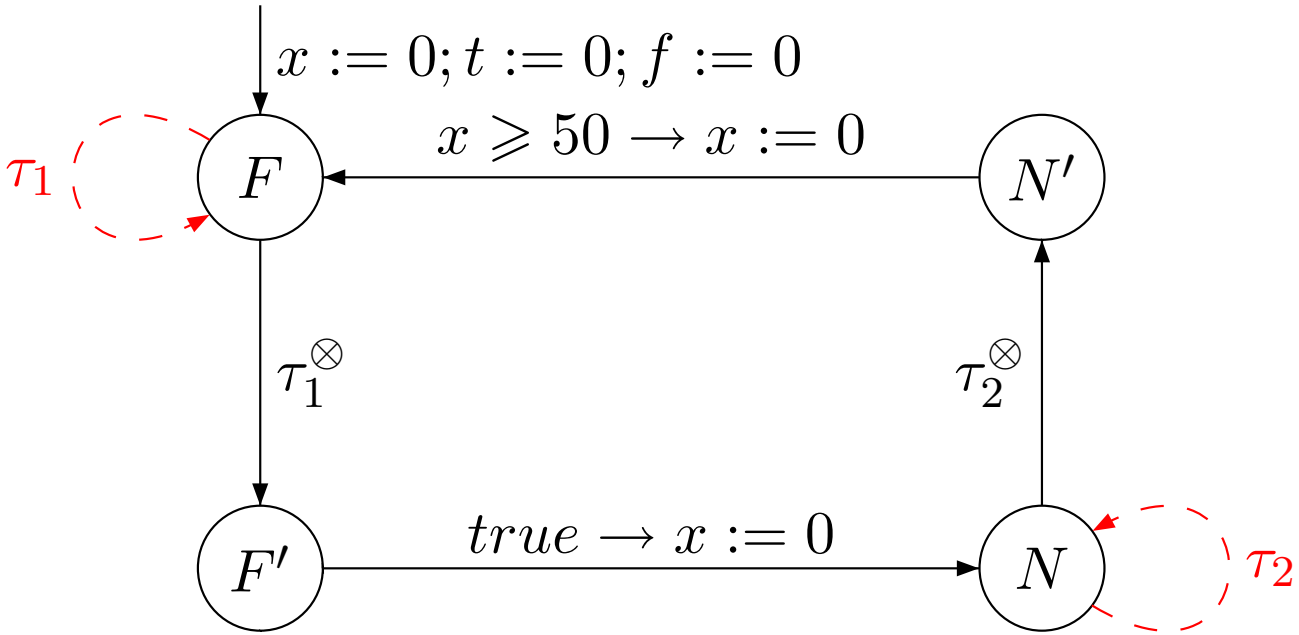
Accélération et Analyse des Relations Linéaires

On veut remplacer les boucles $(\tau_i : g_i \rightarrow a_i)$



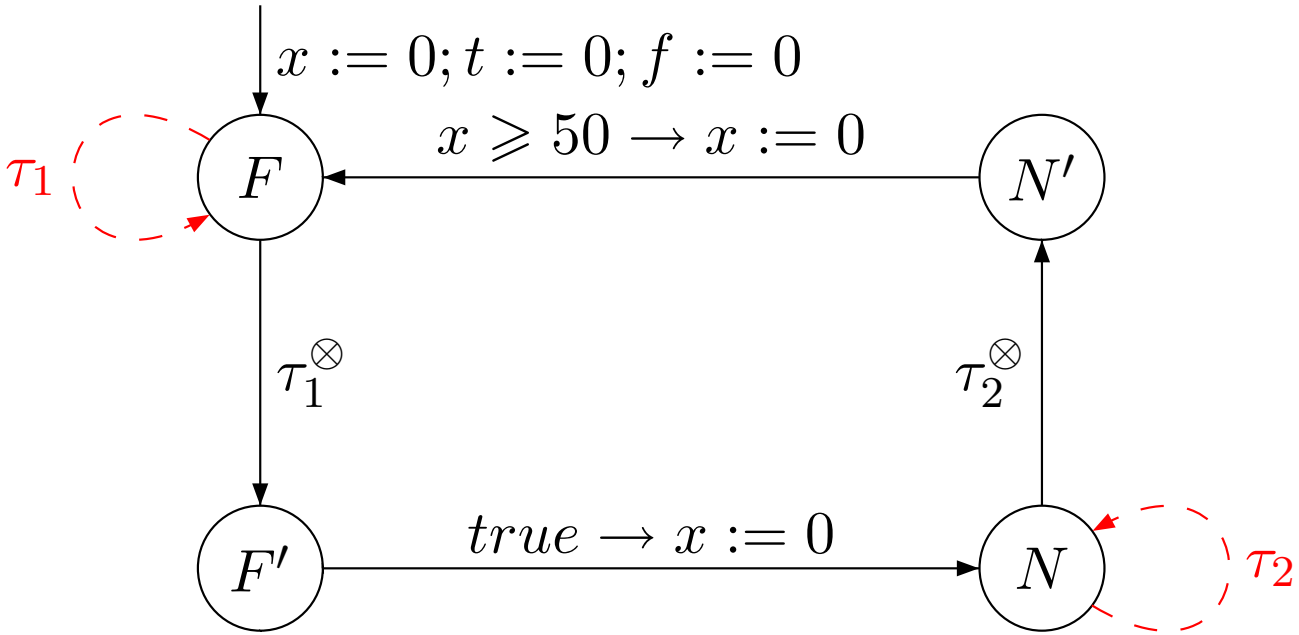
Accélération et Analyse des Relations Linéaires

On veut remplacer les boucles $(\tau_i : g_i \rightarrow a_i)$



Accélération et Analyse des Relations Linéaires

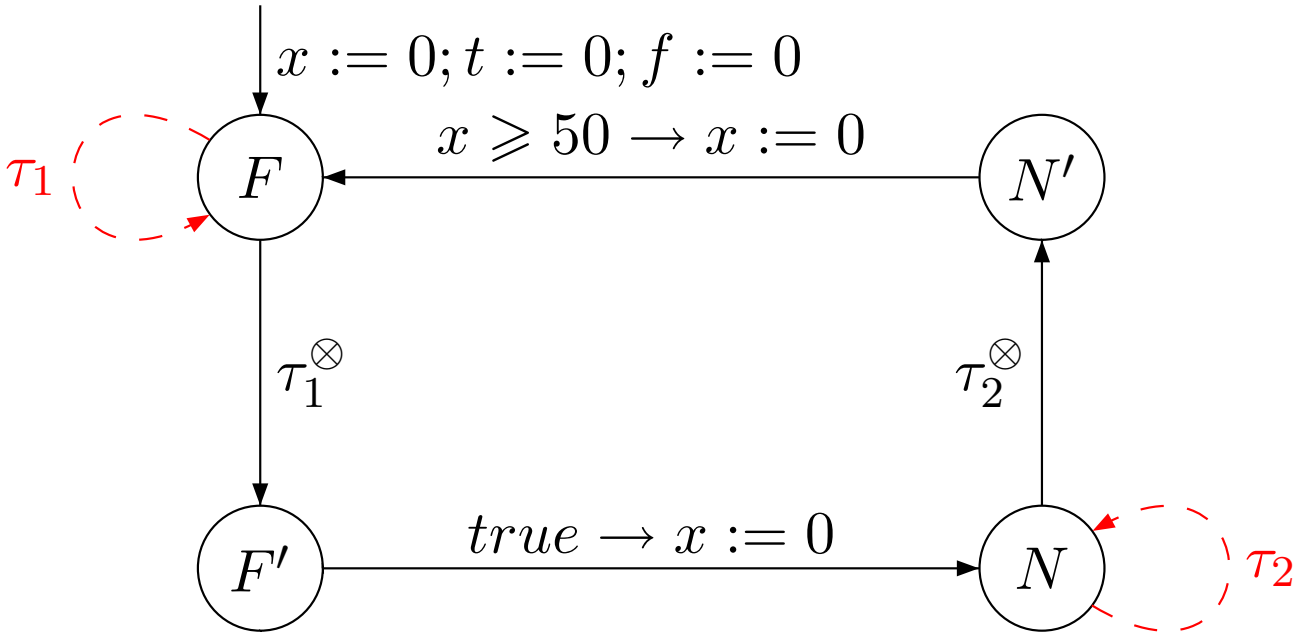
On veut remplacer les boucles $(\tau_i : g_i \rightarrow a_i)$



▶ $\tau_i^{\otimes}$ résume l'effet d'une application de τ_i un **nombre quelconque de fois**

Accélération et Analyse des Relations Linéaires

On veut remplacer les boucles $(\tau_i : g_i \rightarrow a_i)$

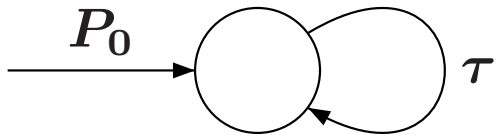


- $\tau_i^{\otimes}$ résume l'effet d'une application de τ_i un **nombre quelconque de fois**
- Boucle englobante : **accélérée** ou **élargie**.

Boucles simples

On veut **caractériser** $\tau^*(P_0) = \bigcup_{i \in \mathbb{N}} \tau^i(P_0)$, avec :

$\tau(X) =$ si $AX \leqslant B$ alors $CX + D$ sinon X

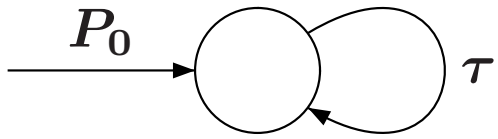


- résultats d'accélération si $\exists p, C^{2p} = C^p$

Boucles simples

On veut **caractériser** $\tau^*(P_0) = \bigcup_{i \in \mathbb{N}} \tau^i(P_0)$, avec :

$\tau(X) =$ si $AX \leqslant B$ alors $CX + D$ sinon X



- résultats d'accélération si $\exists p, C^{2p} = C^p$

Translations : $C = \mathrm{Id}$

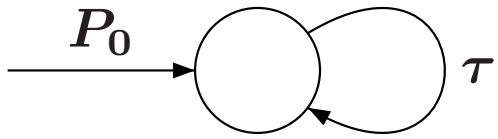
$$\tau^*(P_0) = \{X \mid \exists i \in \mathbb{N}, \exists X_0 \in P_0, AX_0 \leqslant B, A(X - D) \leqslant B, X = X_0 + iD\} \cup P_0$$

Ensemble non convexe + arithmétique

Boucles simples

On veut **caractériser** $\tau^*(P_0) = \bigcup_{i \in \mathbb{N}} \tau^i(P_0)$, avec :

$\tau(X) =$ si $AX \leqslant B$ alors $CX + D$ sinon X



- résultats d'accélération si $\exists p, C^{2p} = C^p$

Translations : $C = \mathrm{Id}$

$$\tau^*(P_0) = \{X \mid \exists i \in \mathbb{N}, \exists X_0 \in P_0, AX_0 \leqslant B, A(X - D) \leqslant B, X = X_0 + iD\} \cup P_0$$

Ensemble non convexe + arithmétique

Accélération dense (vs discrète)

$$\tau^{\otimes}(P_0) = \{X \mid \exists i \in \mathbb{Q}^+, \exists X_0 \in P_0, AX_0 \leqslant B, A(X - D) \leqslant B, X = X_0 + iD\} \sqcup P_0$$

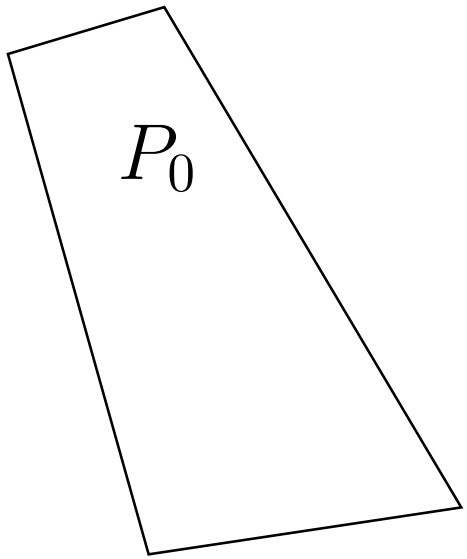
Polyèdre convexe

Translation simple, unique boucle (2)

$$\tau(X) = \text{ si } AX \leqslant B \text{ alors } X + D \text{ sinon } X$$

Algorithme : ajout de rayon !

$$\tau^{\otimes}(P_0) = ((P_0 \cap (AX \leqslant B)) \nearrow \{D\}) \cap (A(X - D) \leqslant B) \sqcup P_0$$

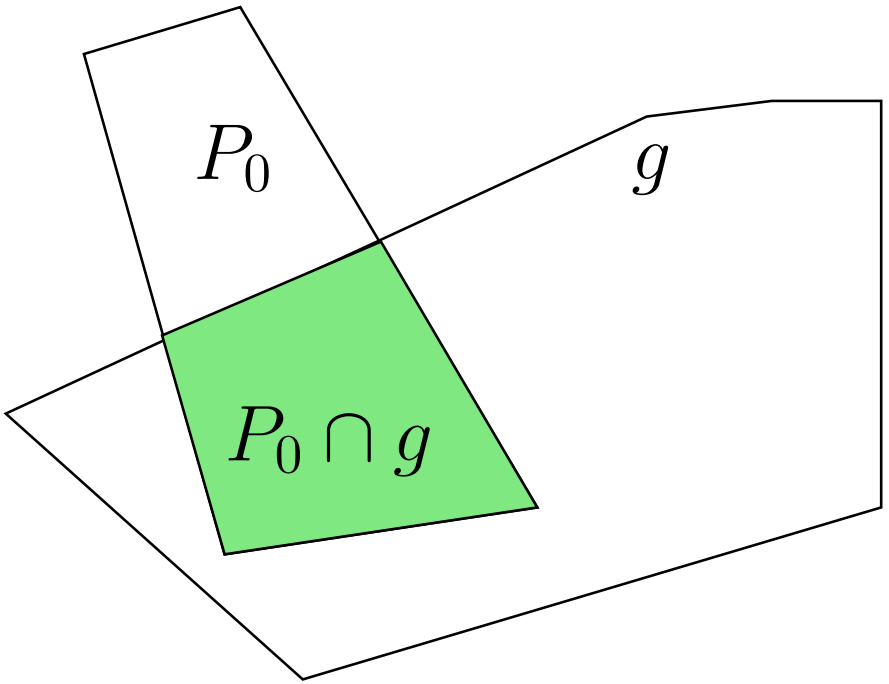


Translation simple, unique boucle (2)

$$\tau(X) = \text{ si } AX \leqslant B \text{ alors } X + D \text{ sinon } X$$

Algorithme : ajout de rayon !

$$\tau^{\otimes}(P_0) = ((P_0 \cap (AX \leqslant B)) \nearrow \{D\}) \cap (A(X - D) \leqslant B) \sqcup P_0$$

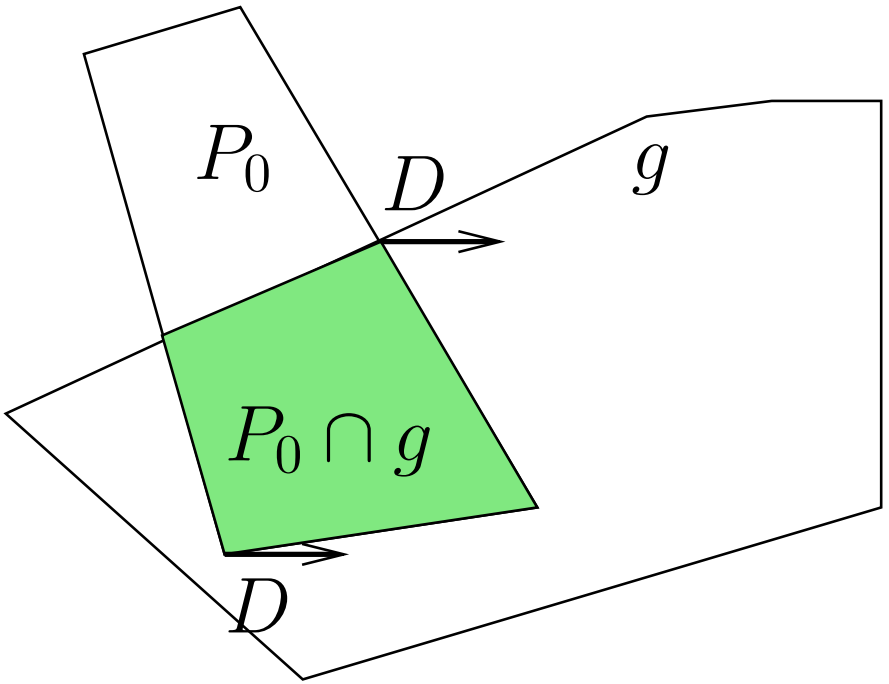


Translation simple, unique boucle (2)

$$\tau(X) = \text{ si } AX \leqslant B \text{ alors } X + D \text{ sinon } X$$

Algorithme : ajout de rayon !

$$\tau^{\otimes}(P_0) = ((P_0 \cap (AX \leqslant B)) \nearrow \{D\}) \cap (A(X - D) \leqslant B) \sqcup P_0$$

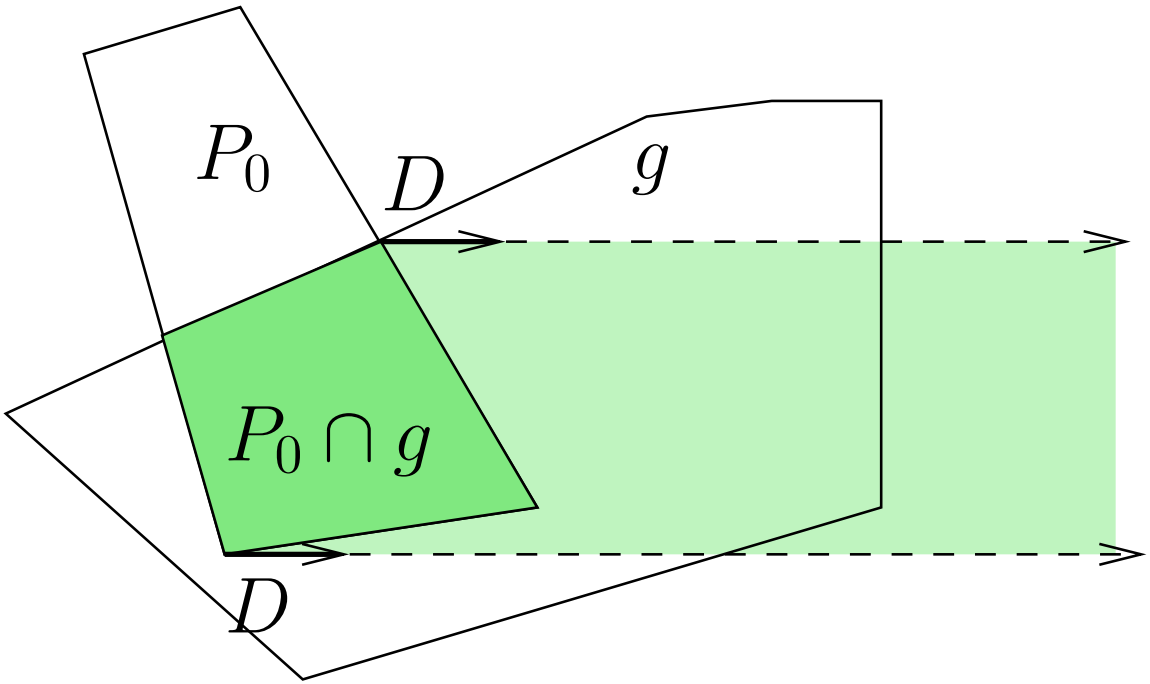


Translation simple, unique boucle (2)

$$\tau(X) = \text{ si } AX \leqslant B \text{ alors } X + D \text{ sinon } X$$

Algorithme : ajout de rayon !

$$\tau^{\otimes}(P_0) = ((P_0 \cap (AX \leqslant B)) \nearrow \{D\}) \cap (A(X - D) \leqslant B) \sqcup P_0$$

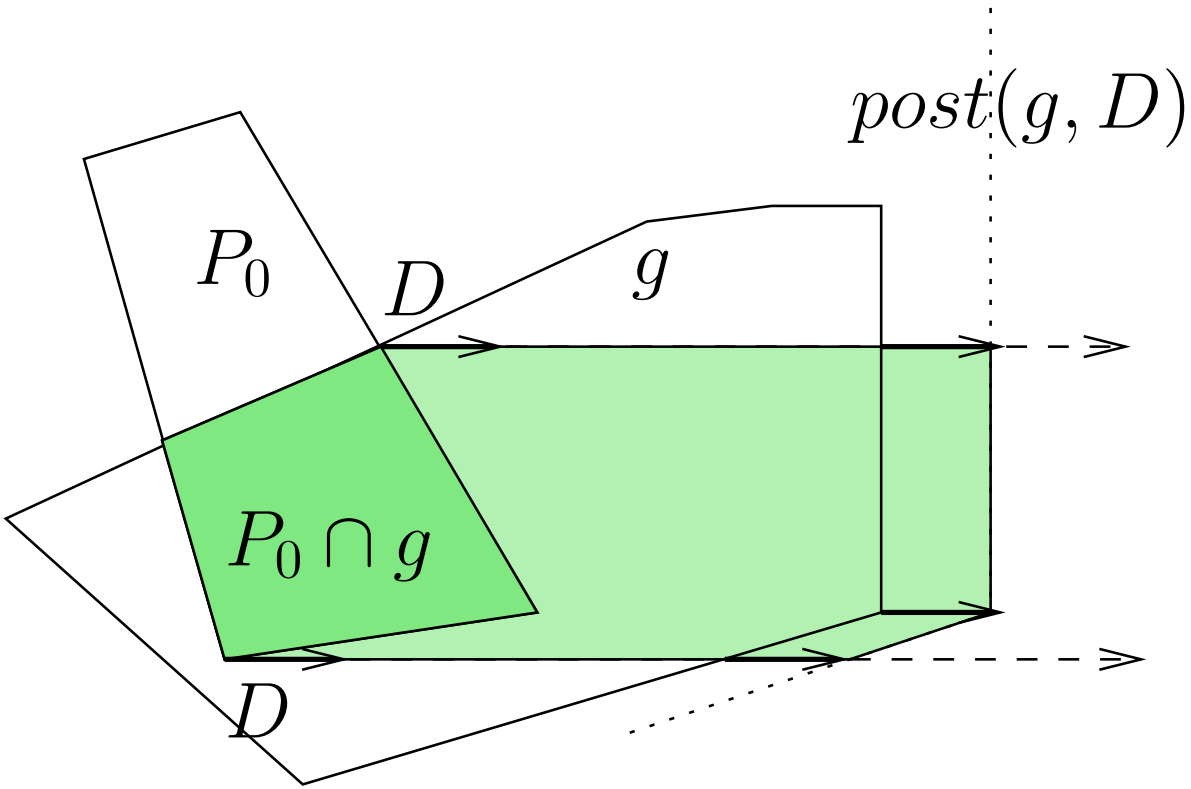


Translation simple, unique boucle (2)

$$\tau(X) = \text{ si } AX \leqslant B \text{ alors } X + D \text{ sinon } X$$

Algorithme : ajout de rayon !

$$\tau^{\otimes}(P_0) = ((P_0 \cap (AX \leqslant B)) \nearrow \{D\}) \cap (A(X - D) \leqslant B) \sqcup P_0$$

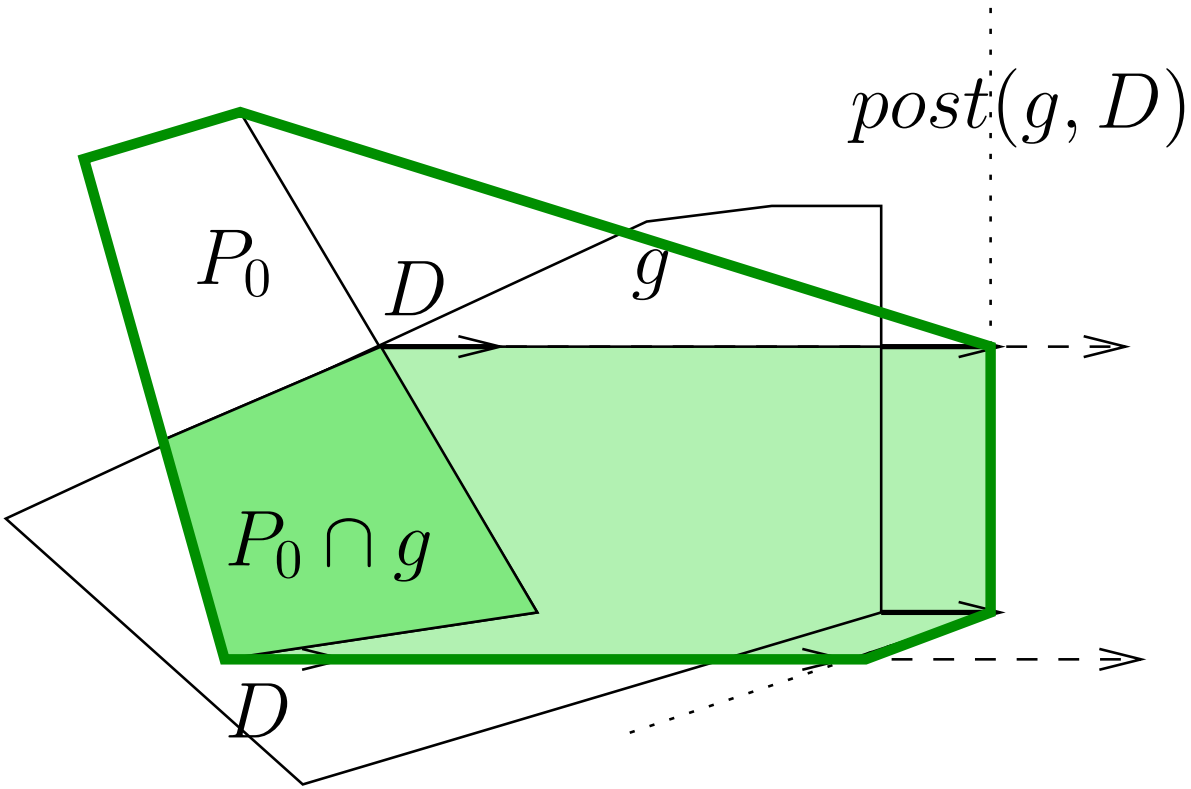


Translation simple, unique boucle (2)

$$\tau(X) = \text{ si } AX \leqslant B \text{ alors } X + D \text{ sinon } X$$

Algorithme : ajout de rayon !

$$\tau^{\otimes}(P_0) = ((P_0 \cap (AX \leqslant B)) \nearrow \{D\}) \cap (A(X - D) \leqslant B) \sqcup P_0$$



Unique boucle, remarques

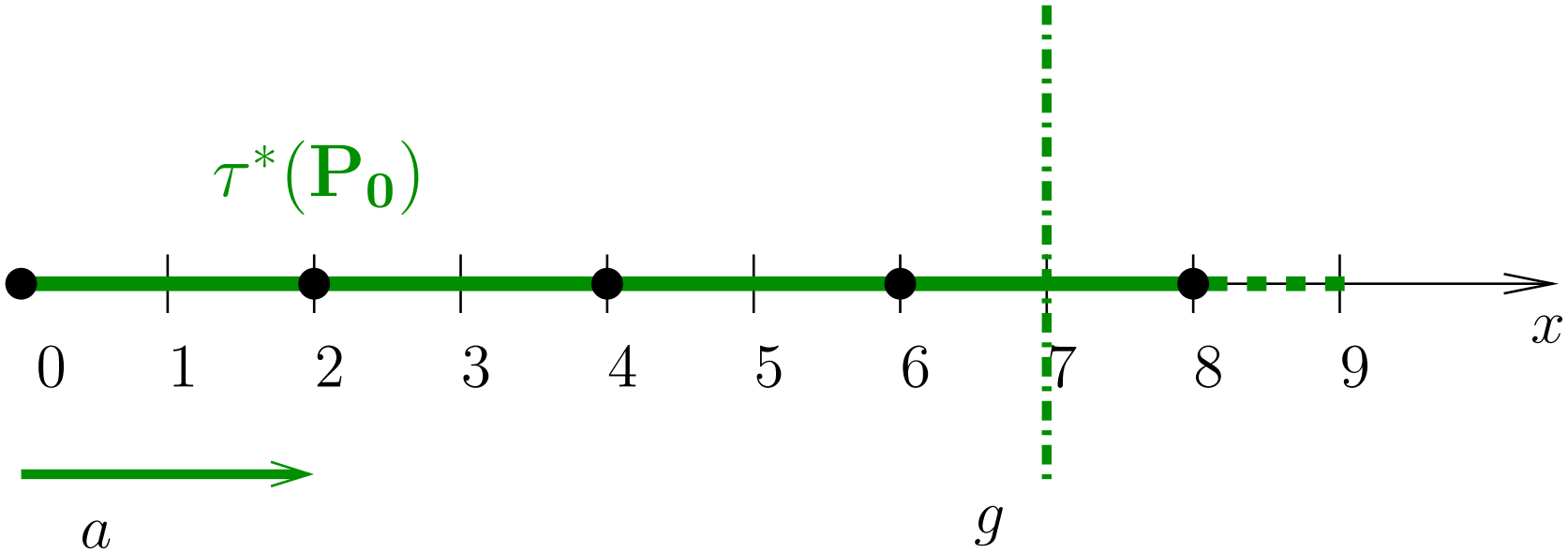
Que perd-on ?

$$\tau = \begin{cases} g = (x \leqslant 7) \\ a = (x := x + 2) \end{cases}$$

$$P_0 = \{x = 0\}$$

$$\tau^*(P_0) = \{0, 2, 4, 6, 8\}$$

$$\begin{aligned} \tau^{\otimes}(P_0) &= P_0 \nearrow (1) \cap (x - 2 \leqslant 7) \\ &= \{0 \leqslant x \leqslant 9\} \end{aligned}$$



Une classe de transition traitée

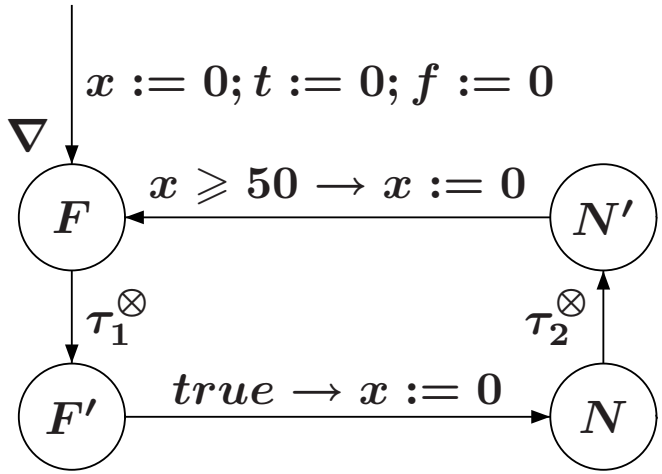
$$\tau(X) = \text{si } AX \leqslant B \text{ alors } CX + D \text{ sinon } X$$

Proposition Si il existe p tel que $C^{2p} = C^p$, alors on sait calculer une sur-approximation convexe de $\tau^*(P_0)$.

Remarques

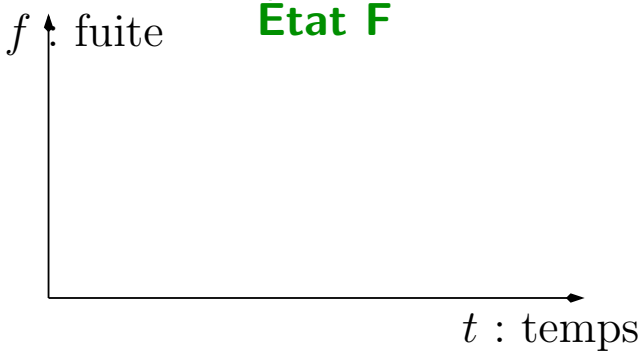
- Changement de base, puis ajout de rayon.
- En pratique : $p \leqslant 3$.

Ex. : application à la chaudière

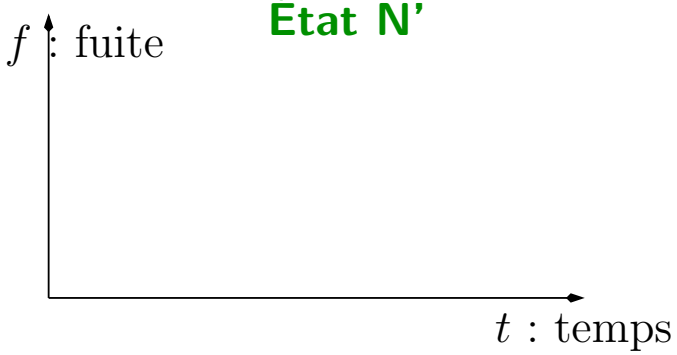


- $\tau_1^{\otimes}$ = "ajoute le rayon $(1, 1, 1)$ tant que $x \leqslant 10$"
- $\tau_2^{\otimes}$ = "ajoute le rayon $(1, 0, 1)$"

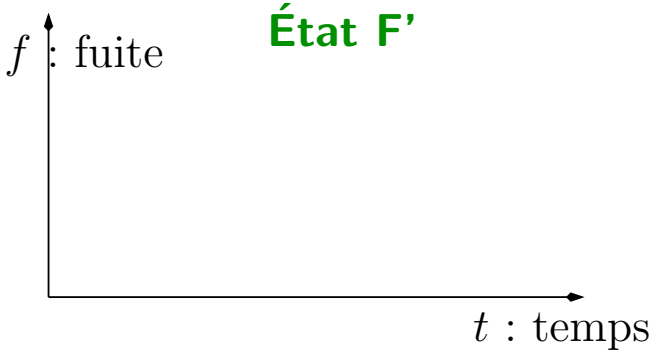
État F



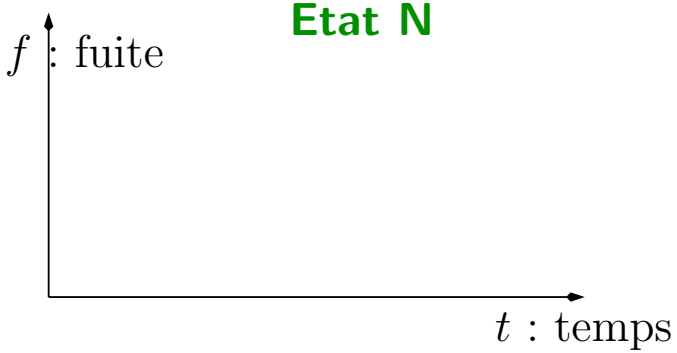
État N'



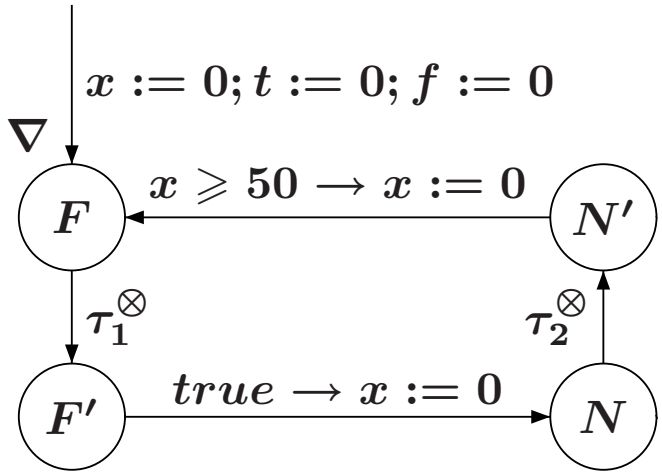
État F'



État N

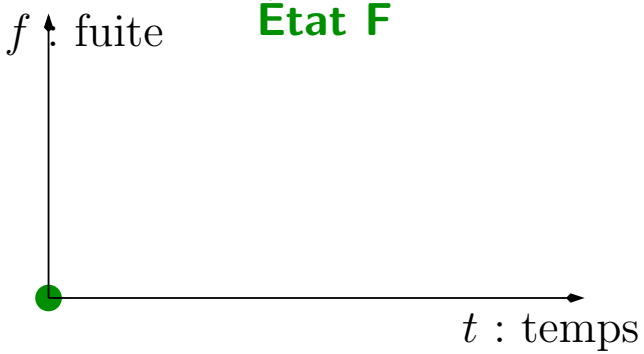


Ex. : application à la chaudière

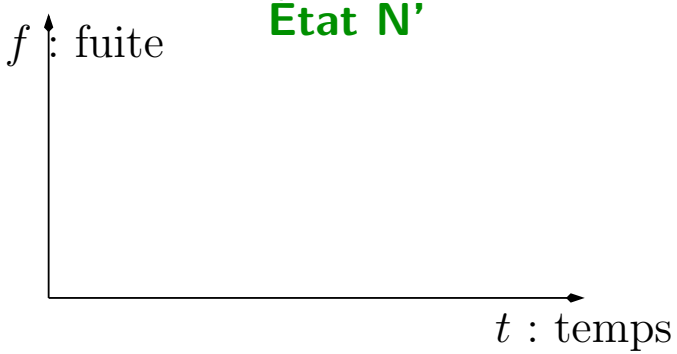


- $\tau_1^{\otimes}$ = "ajoute le rayon $(1, 1, 1)$ tant que $x \leqslant 10$"
- $\tau_2^{\otimes}$ = "ajoute le rayon $(1, 0, 1)$"

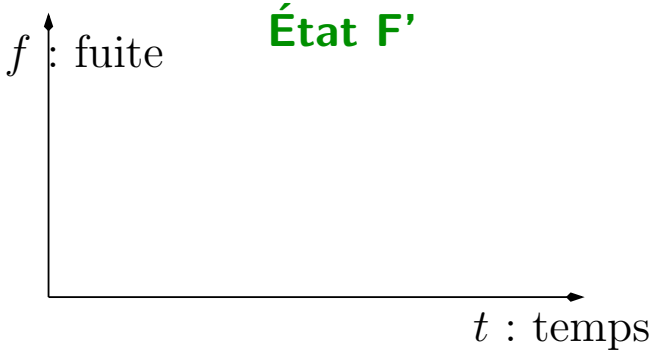
État F



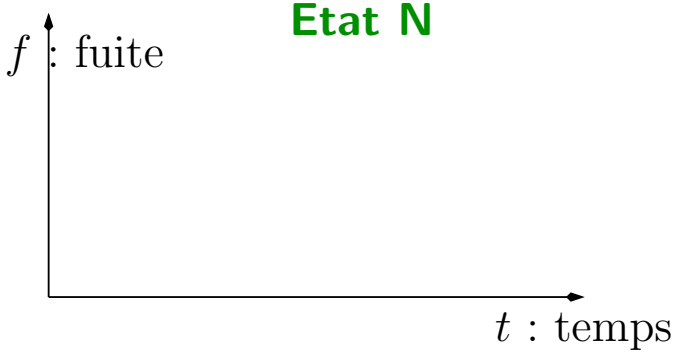
État N'



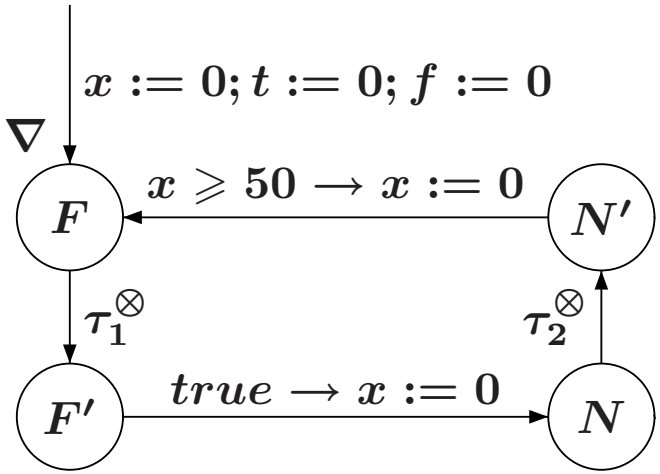
État F'



État N

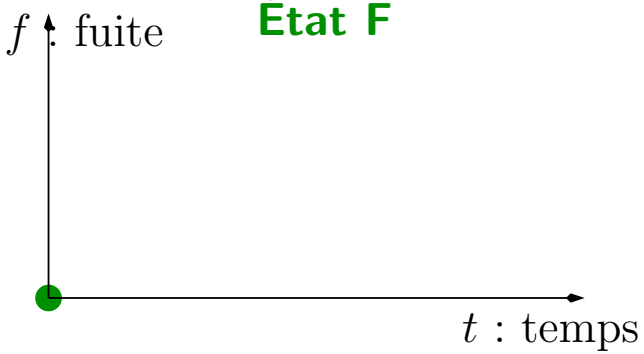


Ex. : application à la chaudière

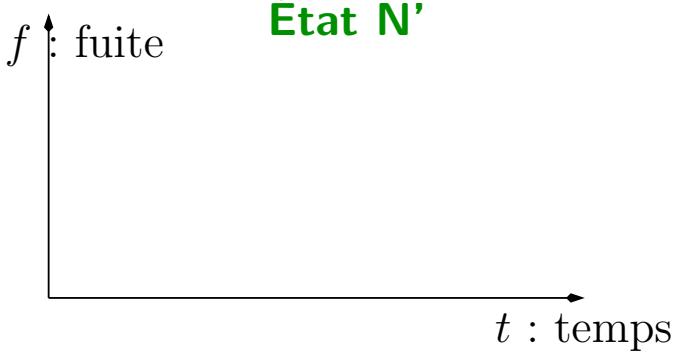


- $\tau_1^{\otimes}$ = "ajoute le rayon $(1, 1, 1)$ tant que $x \leqslant 10$"
- $\tau_2^{\otimes}$ = "ajoute le rayon $(1, 0, 1)$"

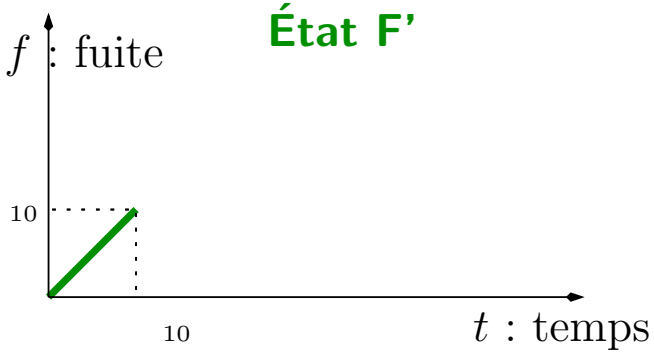
État F



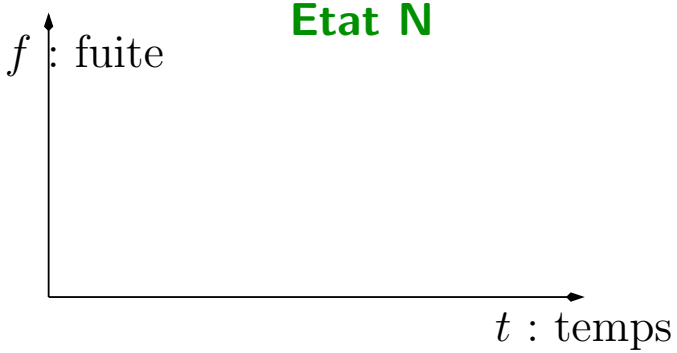
État N'



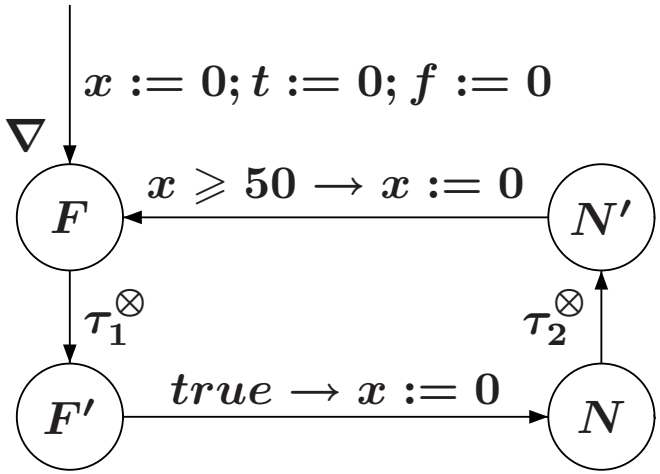
État F'



État N

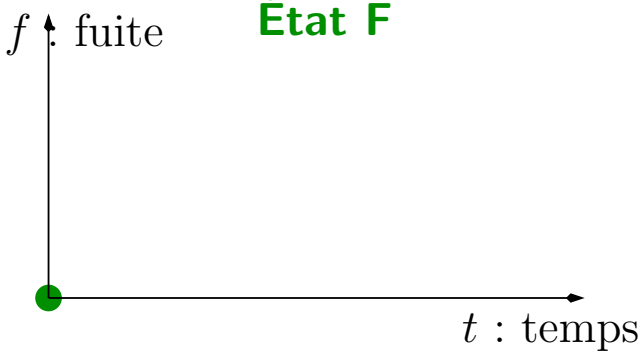


Ex. : application à la chaudière

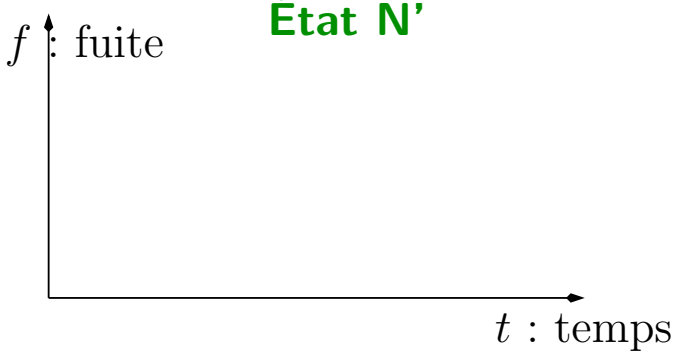


- $\tau_1^{\otimes}$ = "ajoute le rayon $(1, 1, 1)$ tant que $x \leqslant 10$"
- $\tau_2^{\otimes}$ = "ajoute le rayon $(1, 0, 1)$"

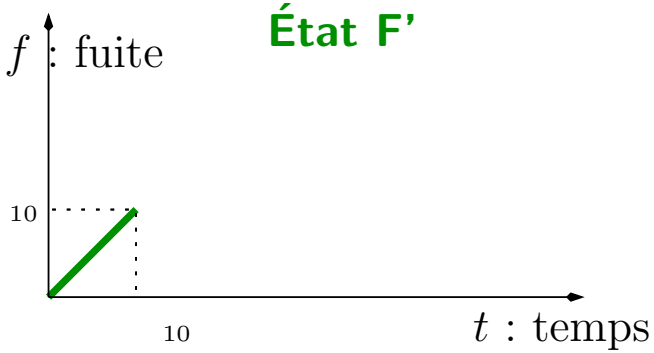
État F



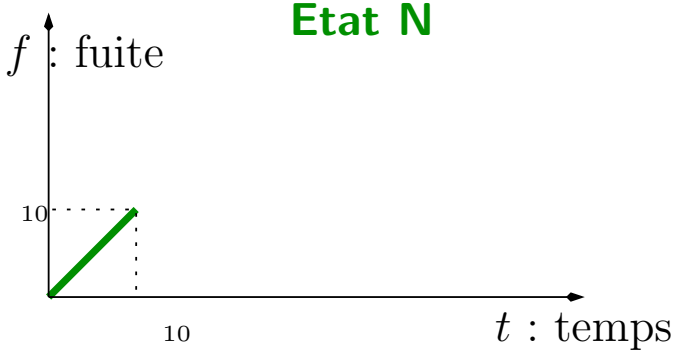
État N'



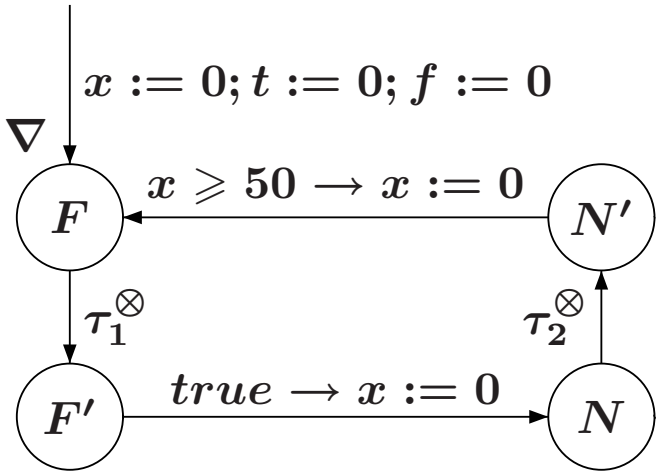
État F'



État N

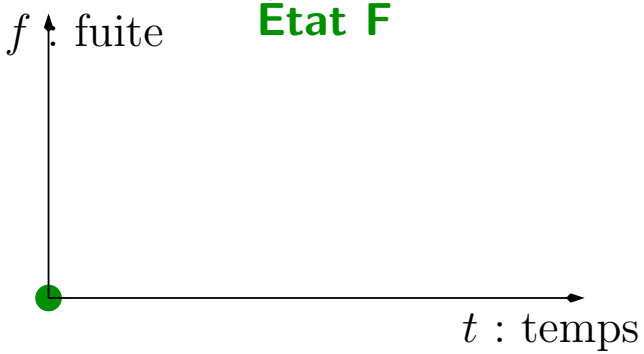


Ex. : application à la chaudière

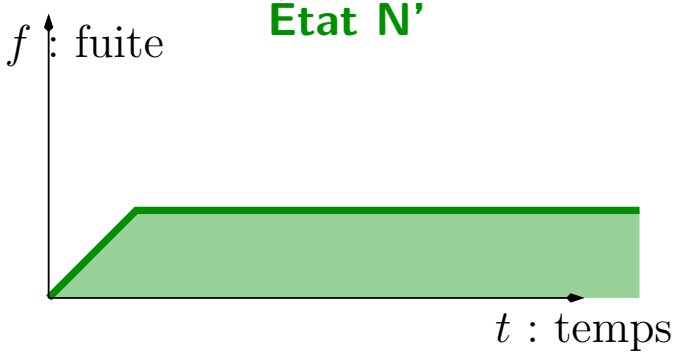


- $\tau_1^{\otimes}$ = "ajoute le rayon $(1, 1, 1)$ tant que $x \leqslant 10$"
- $\tau_2^{\otimes}$ = "ajoute le rayon $(1, 0, 1)$"

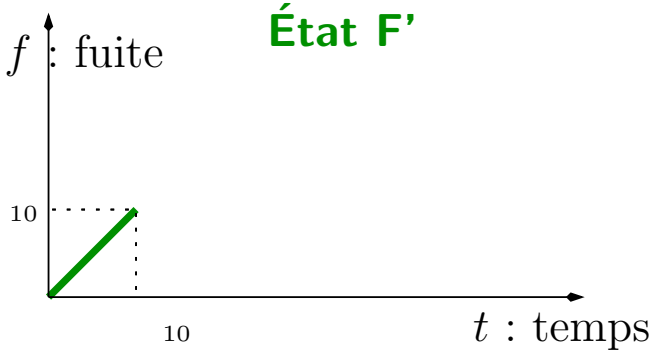
État F



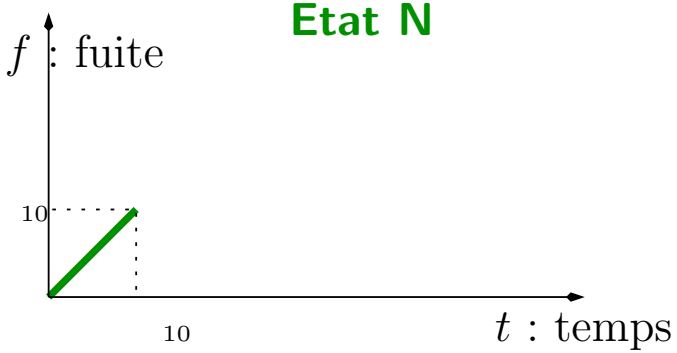
État N'



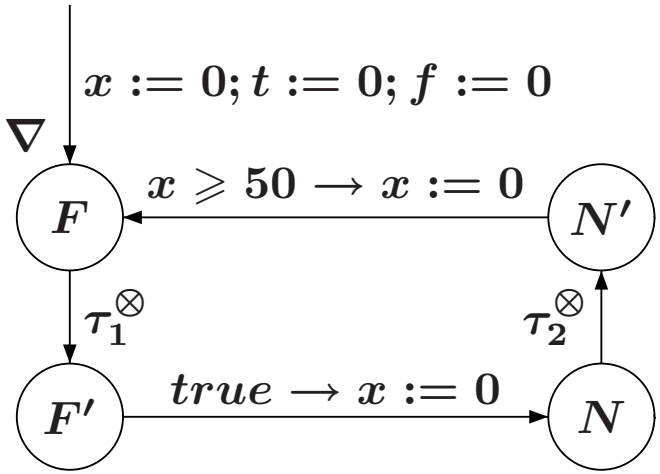
État F'



État N

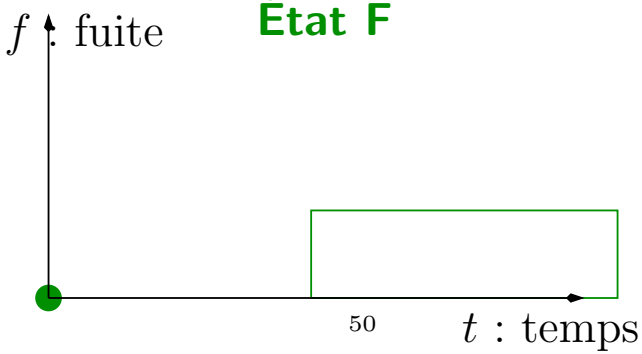


Ex. : application à la chaudière

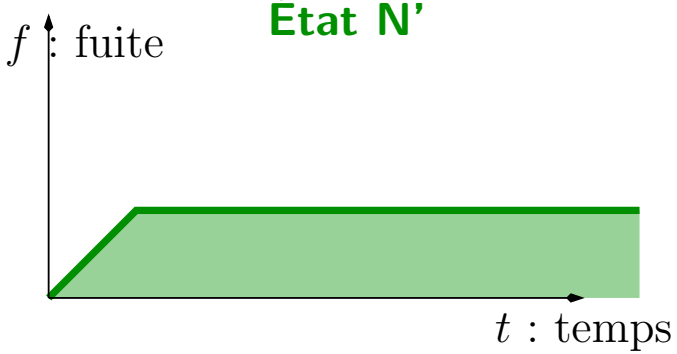


- $\tau_1^{\otimes}$ = "ajoute le rayon $(1, 1, 1)$ tant que $x \leqslant 10$"
- $\tau_2^{\otimes}$ = "ajoute le rayon $(1, 0, 1)$"

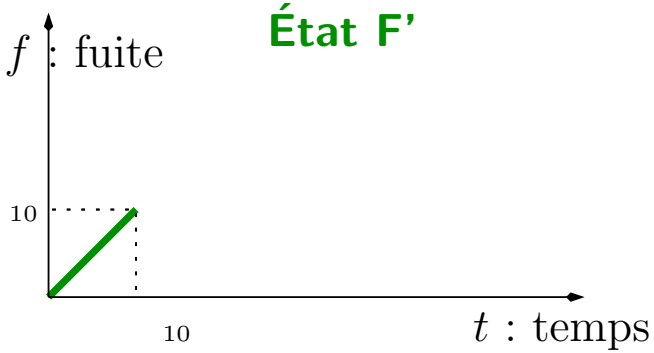
État F



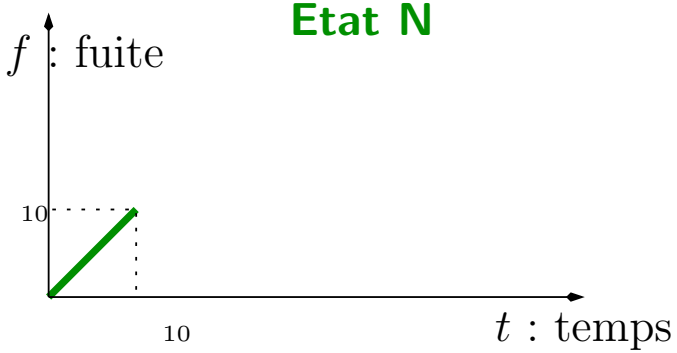
État N'



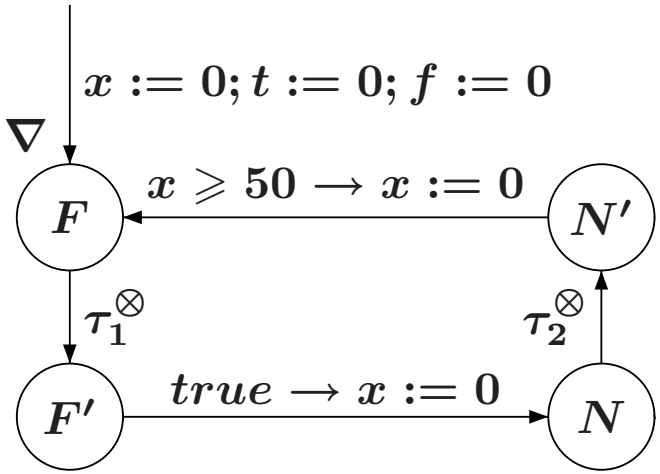
État F'



État N

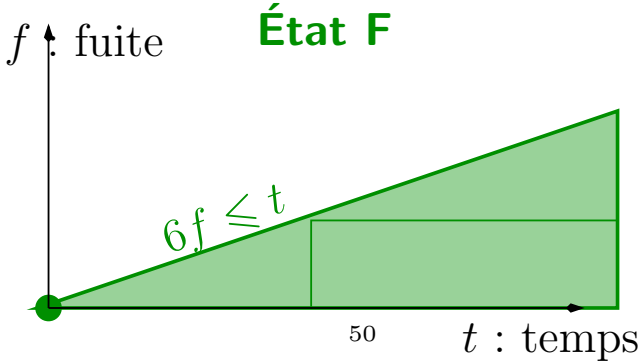


Ex. : application à la chaudière

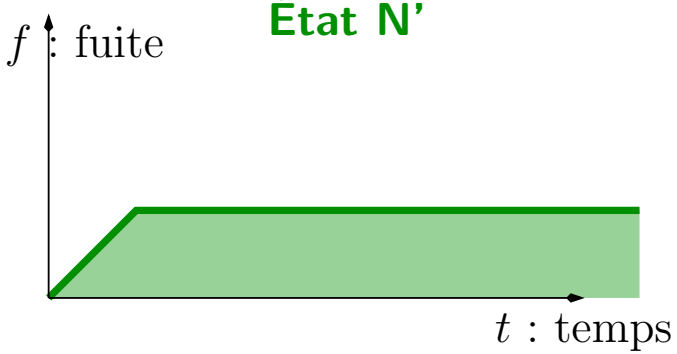


- $\tau_1^{\otimes}$ = "ajoute le rayon $(1, 1, 1)$ tant que $x \leqslant 10$"
- $\tau_2^{\otimes}$ = "ajoute le rayon $(1, 0, 1)$"

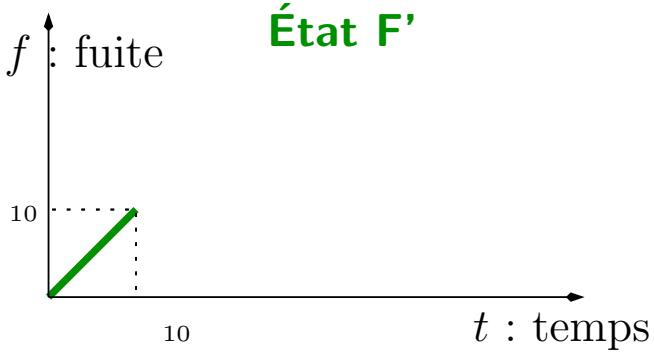
État F



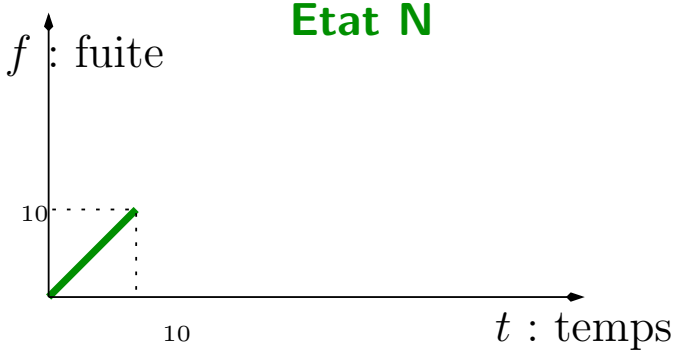
État N'



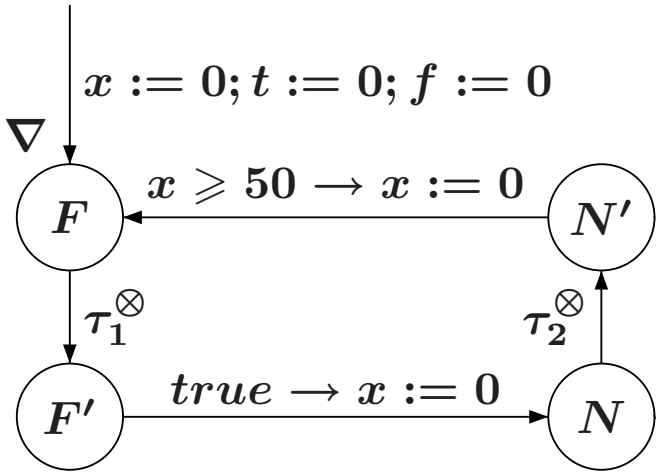
État F'



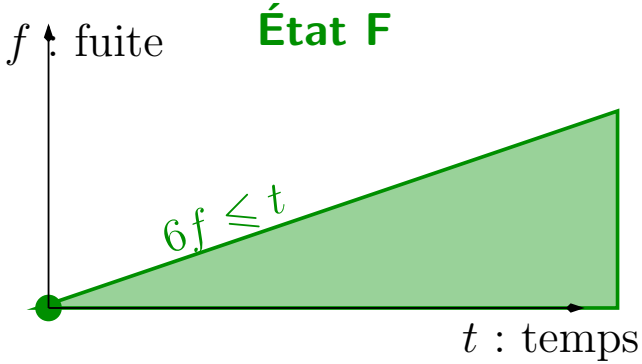
État N

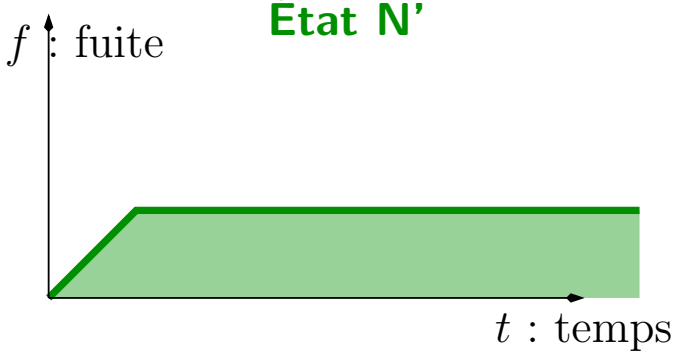


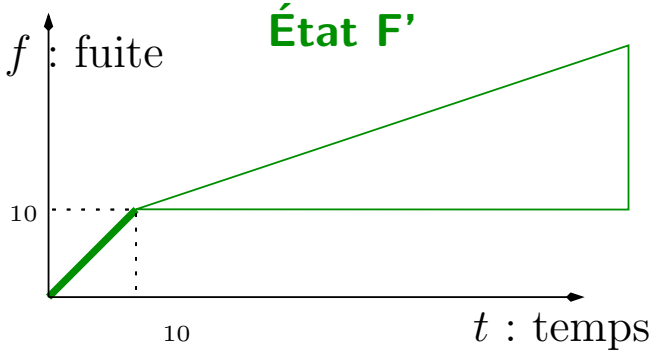
Ex. : application à la chaudière

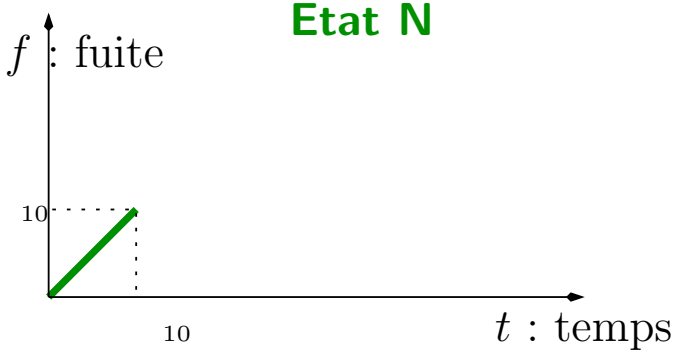


- $\tau_1^{\otimes}$ = "ajoute le rayon $(1, 1, 1)$ tant que $x \leqslant 10$"
- $\tau_2^{\otimes}$ = "ajoute le rayon $(1, 0, 1)$"

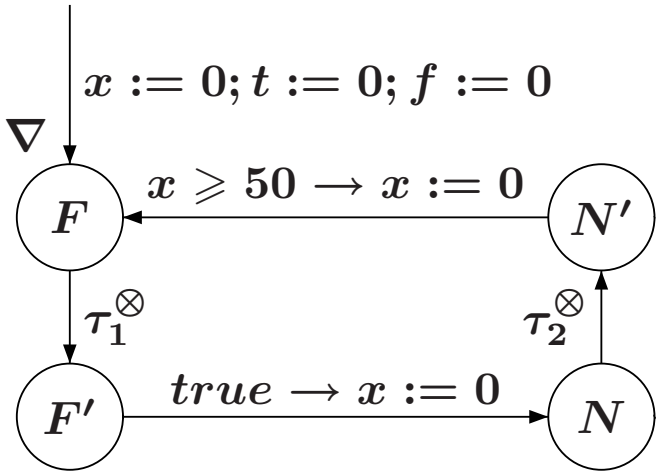






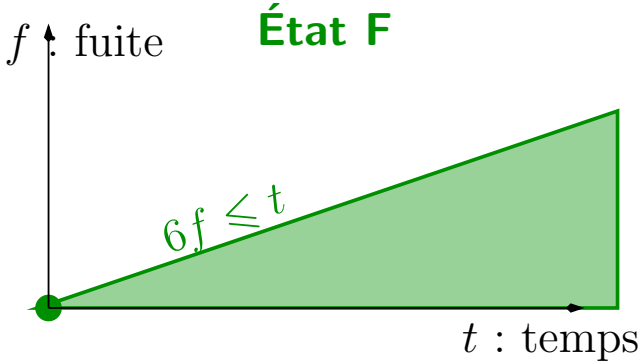


Ex. : application à la chaudière

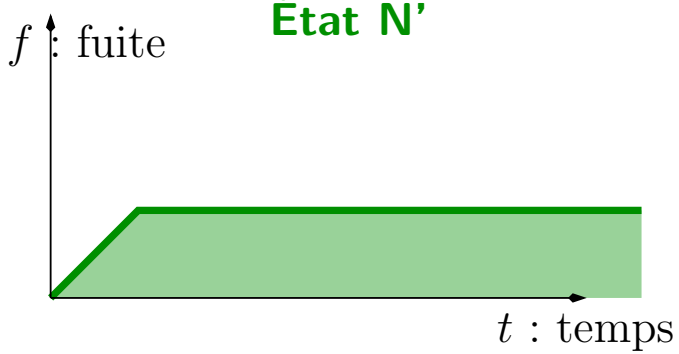


- $\tau_1^{\otimes}$ = "ajoute le rayon $(1, 1, 1)$ tant que $x \leqslant 10$"
- $\tau_2^{\otimes}$ = "ajoute le rayon $(1, 0, 1)$"

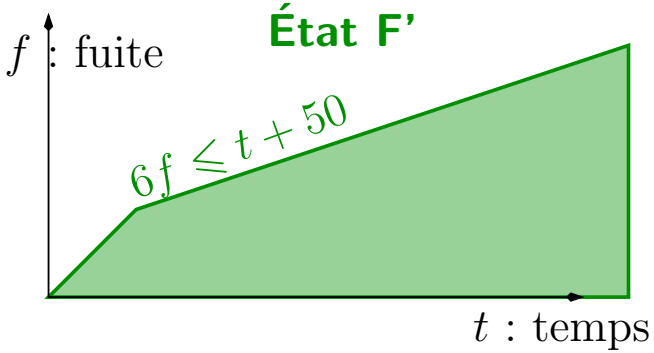
État F



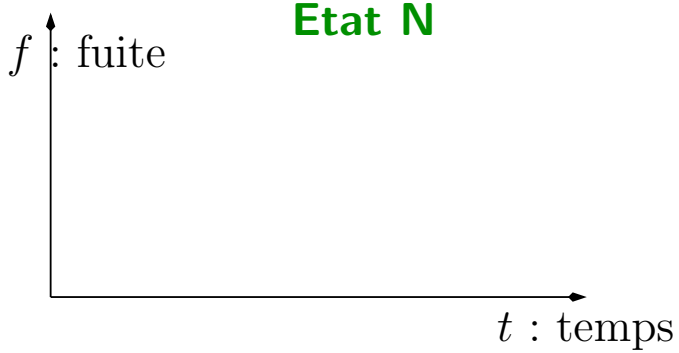
État N'



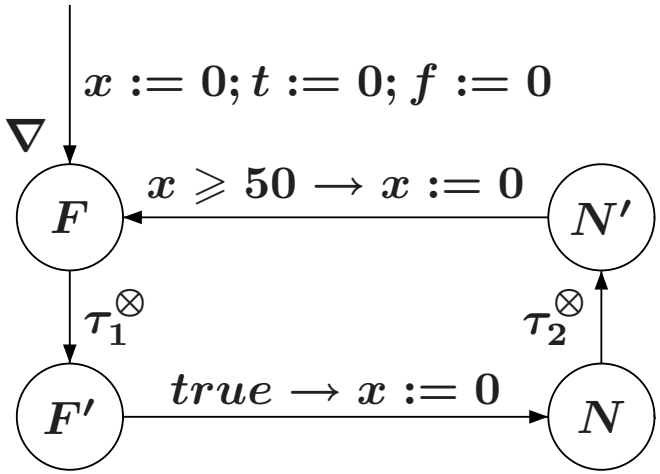
État F'



État N

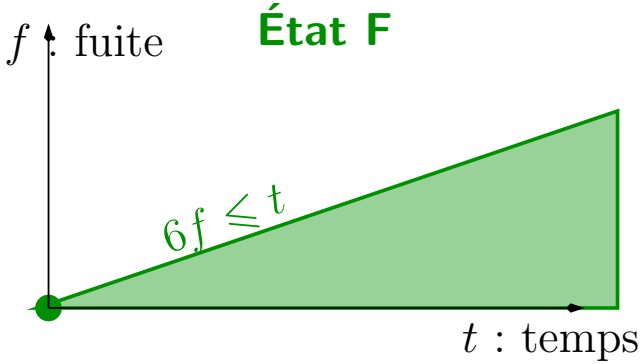


Ex. : application à la chaudière

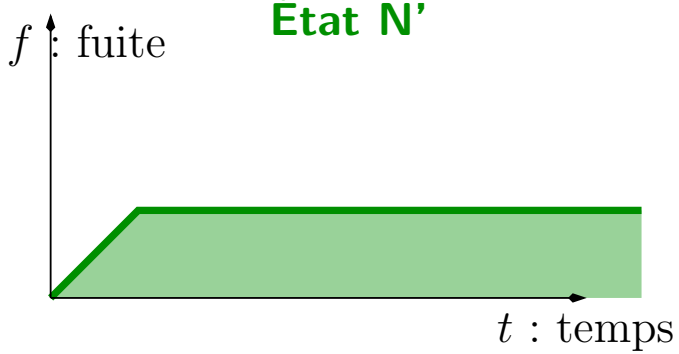


- $\tau_1^{\otimes}$ = "ajoute le rayon $(1, 1, 1)$ tant que $x \leqslant 10$"
- $\tau_2^{\otimes}$ = "ajoute le rayon $(1, 0, 1)$"

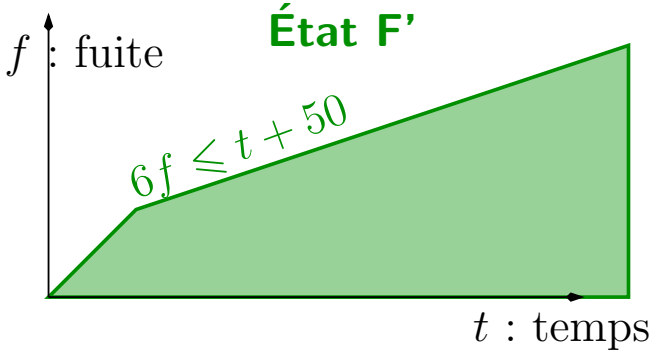
État F



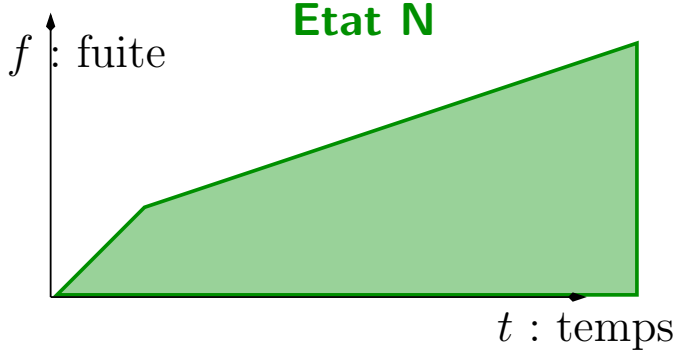
État N'



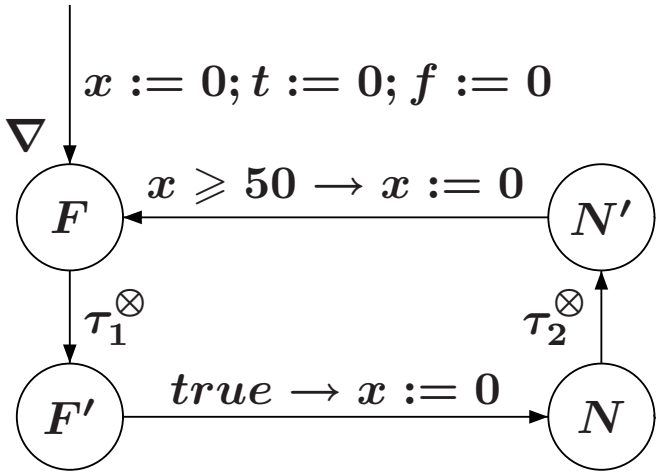
État F'



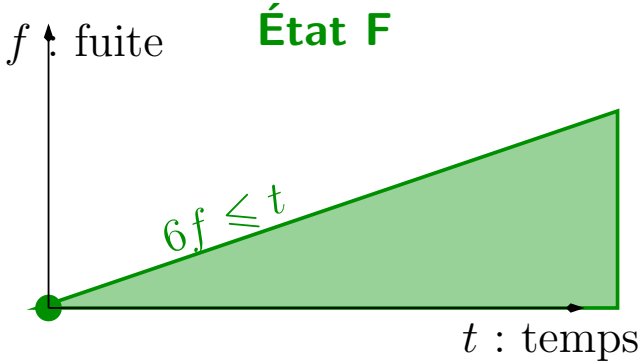
État N

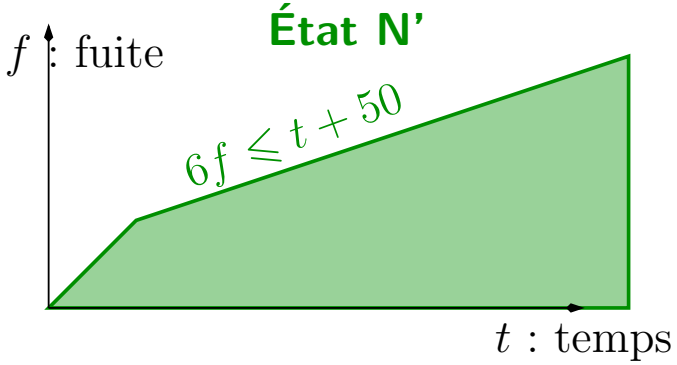


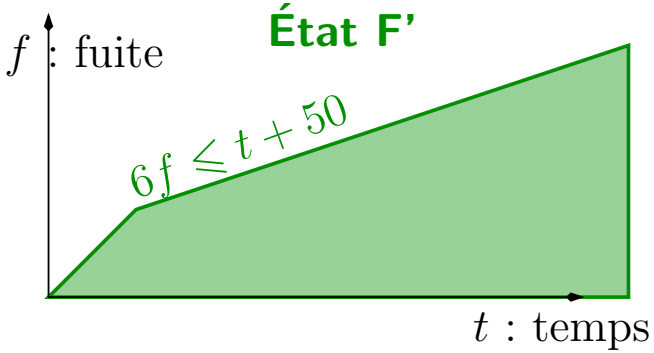
Ex. : application à la chaudière

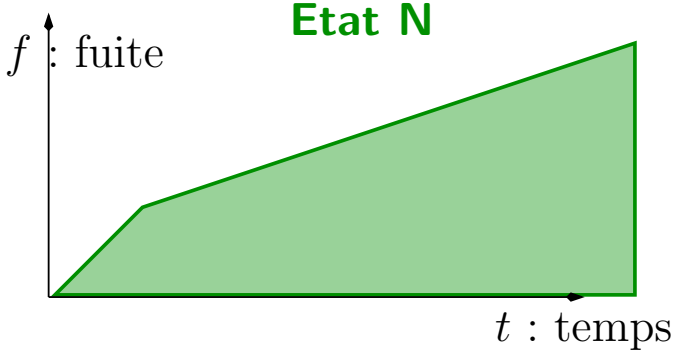


- $\tau_1^{\otimes}$ = "ajoute le rayon $(1, 1, 1)$ tant que $x \leqslant 10$"
- $\tau_2^{\otimes}$ = "ajoute le rayon $(1, 0, 1)$"



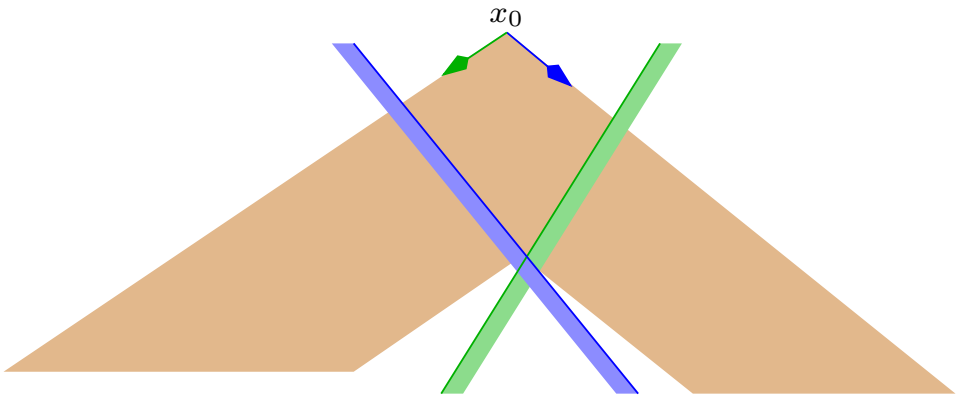






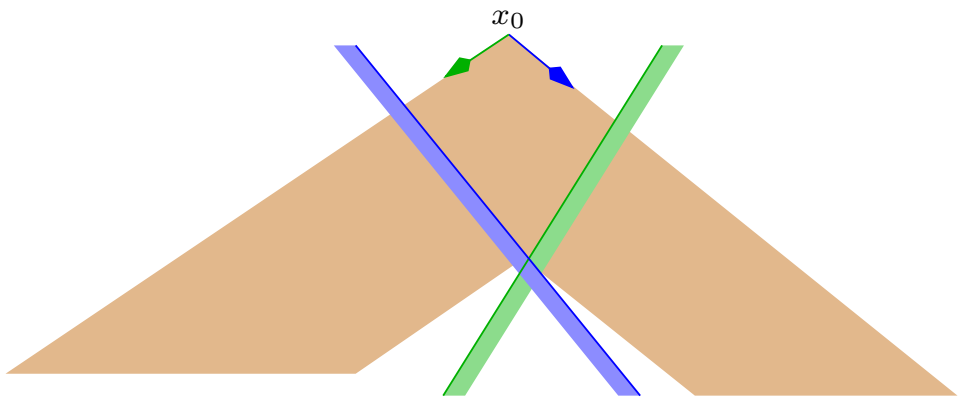
Plusieurs boucles : premières remarques

$(\tau_1$ ou $\tau_2)^*(P_0)$ n'est pas forcément **convexe** :

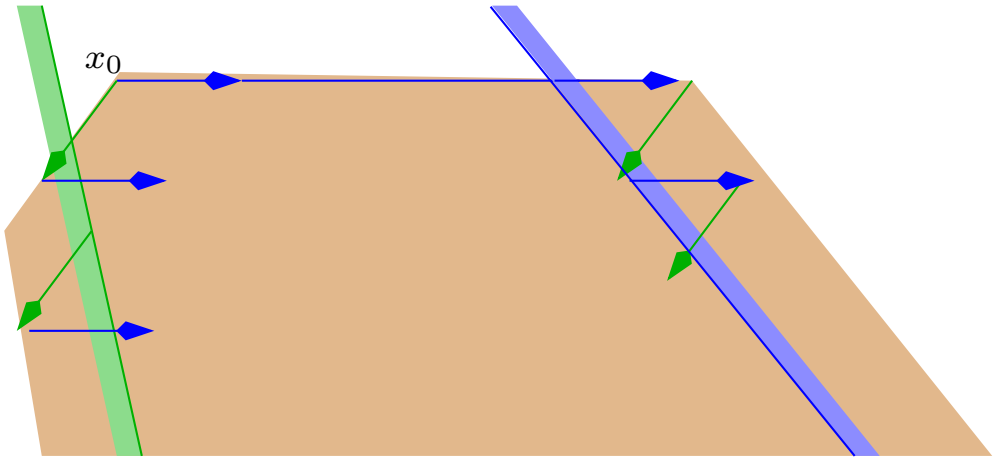


Plusieurs boucles : premières remarques

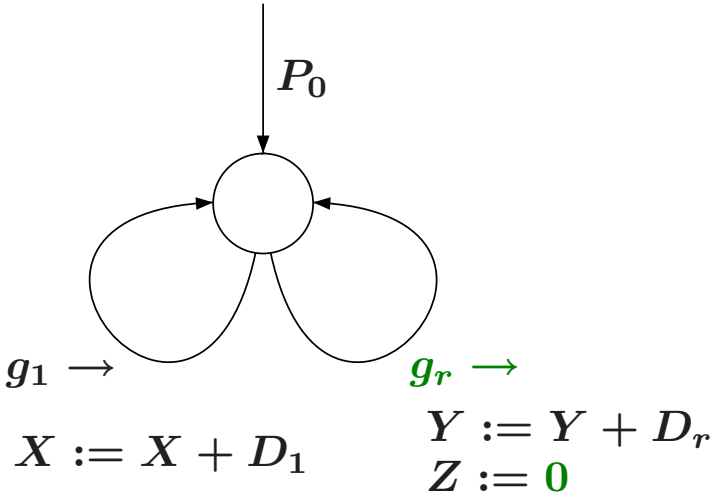
$(\tau_1$ ou $\tau_2)^*(P_0)$ n'est pas forcément **convexe** :



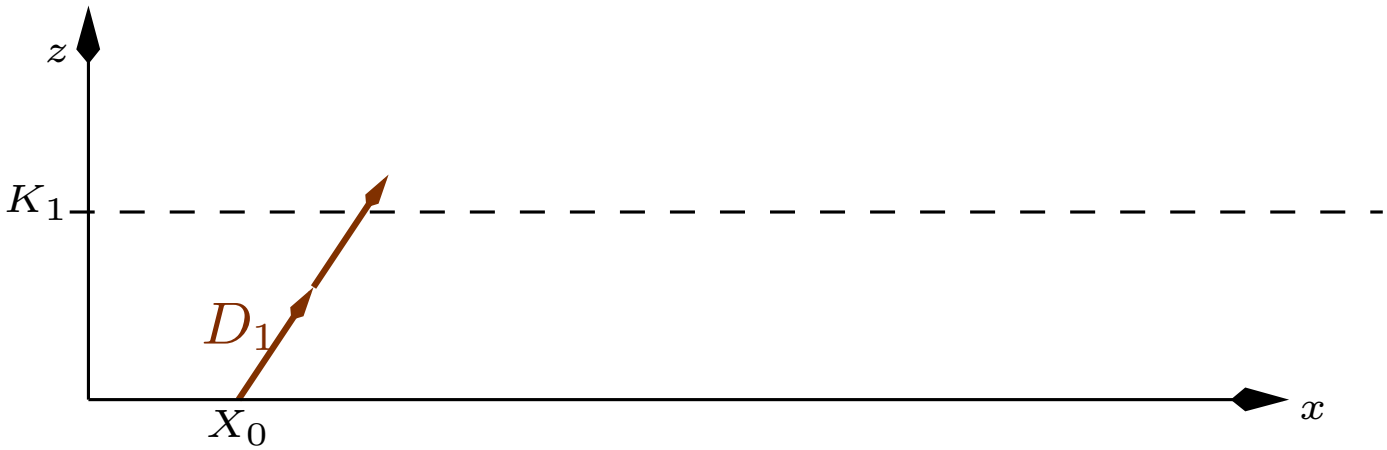
Il peut y avoir des **oscillations** assez complexes :



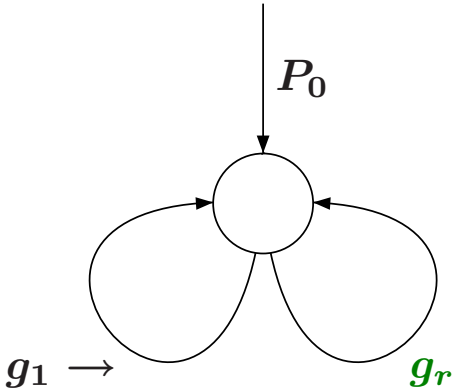
Une combinaison translation et remise à constante (1)



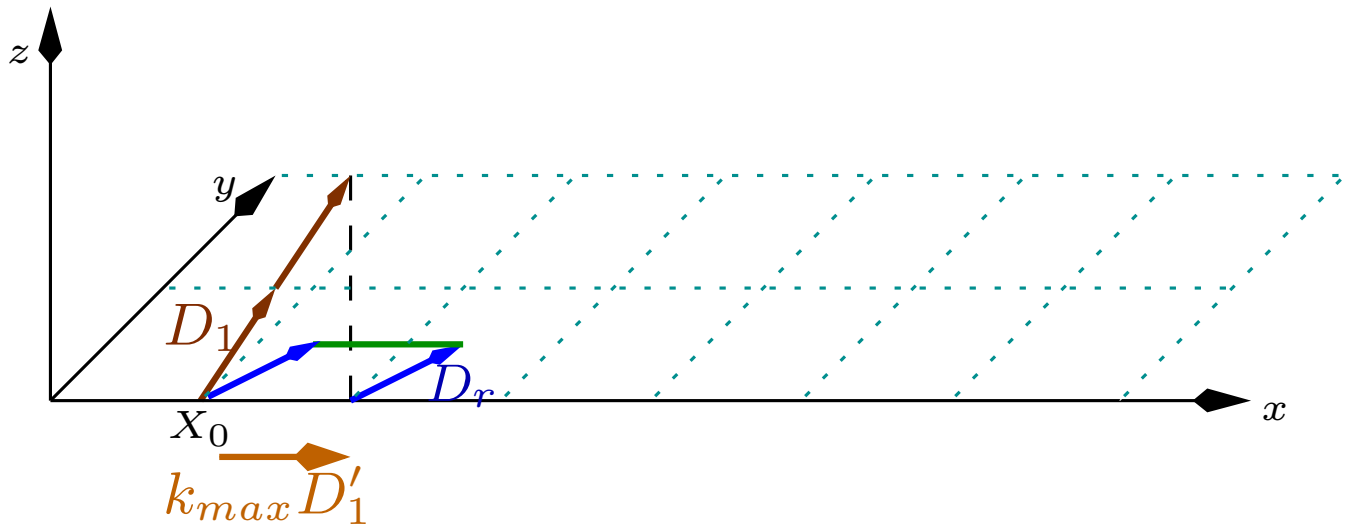
$$g_1 : Z \leqslant K_1,\ g_r = true,$$
$$P_0 \subseteq \{Z = 0\}$$



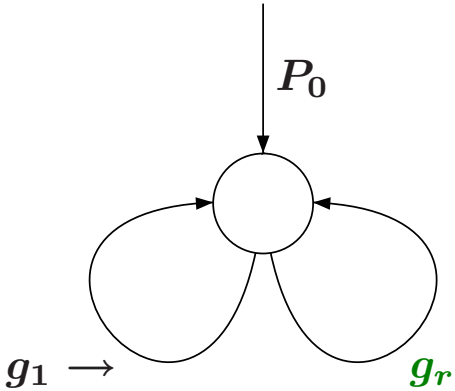
Une combinaison translation et remise à constante (1)



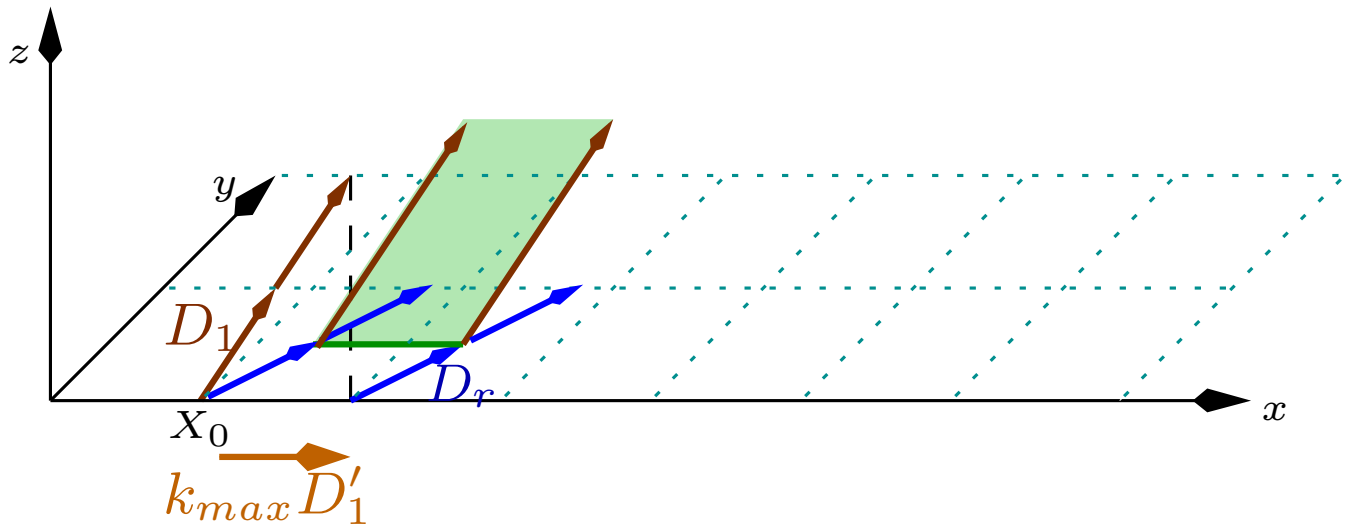
$$g_1 : Z \leqslant K_1,\ g_r = true,$$
$$P_0 \subseteq \{Z = 0\}$$



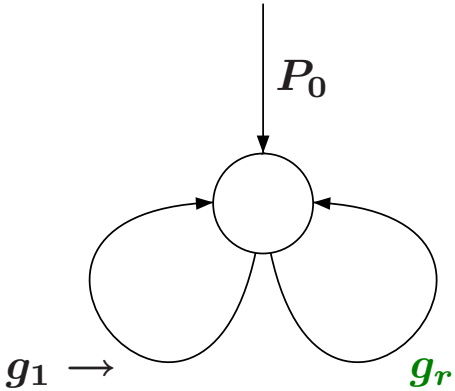
Une combinaison translation et remise à constante (1)



$$g_1 : Z \leqslant K_1,\ g_r = true,$$
$$P_0 \subseteq \{Z = 0\}$$



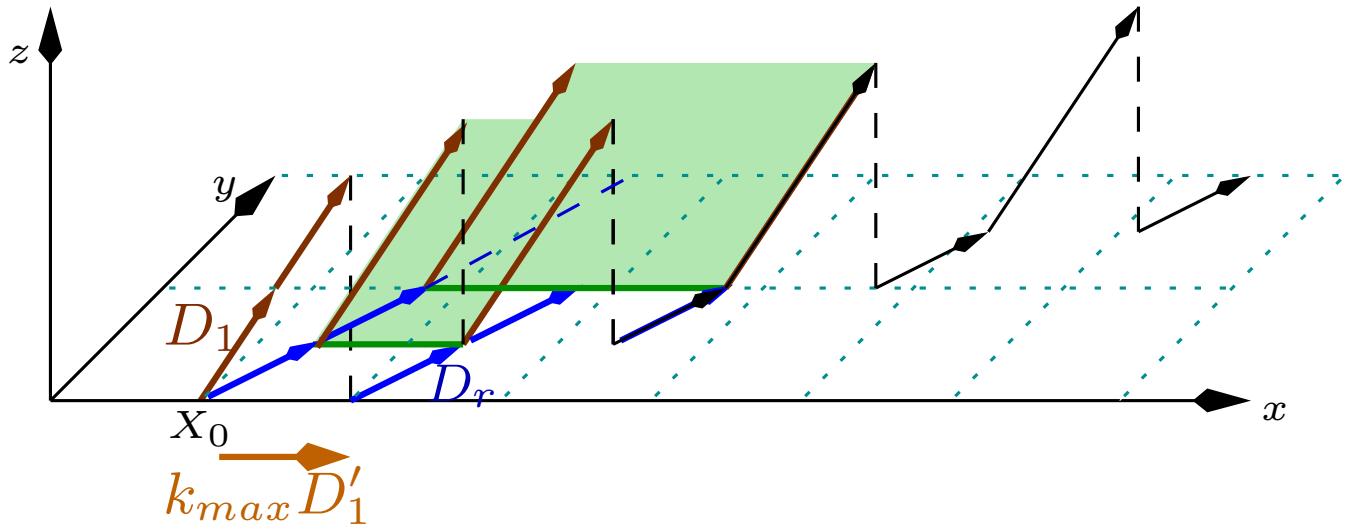
Une combinaison translation et remise à constante (1)



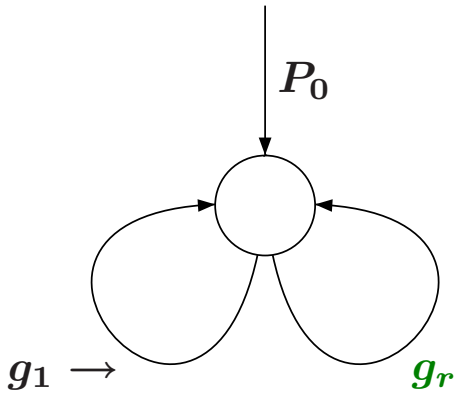
$X := X + D_1$

$Y := Y + D_r$
$Z := 0$

$g_1 : Z \leqslant K_1,\ g_r = true,$
$P_0 \subseteq \{Z = 0\}$



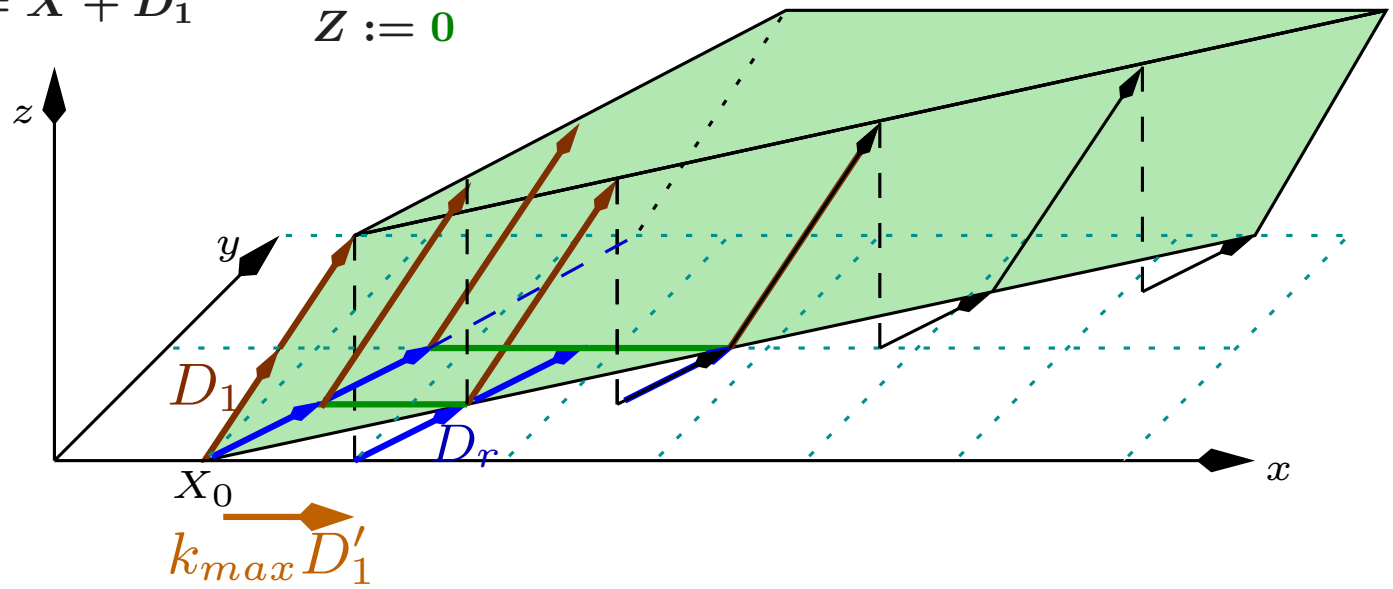
Une combinaison translation et remise à constante (1)



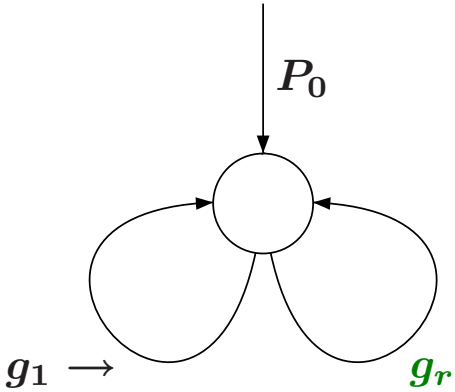
$$g_1 : Z \leqslant K_1,\ g_r = true,$$
$$P_0 \subseteq \{Z = 0\}$$

$X := X + D_1$

$Y := Y + D_r$
$Z := 0$



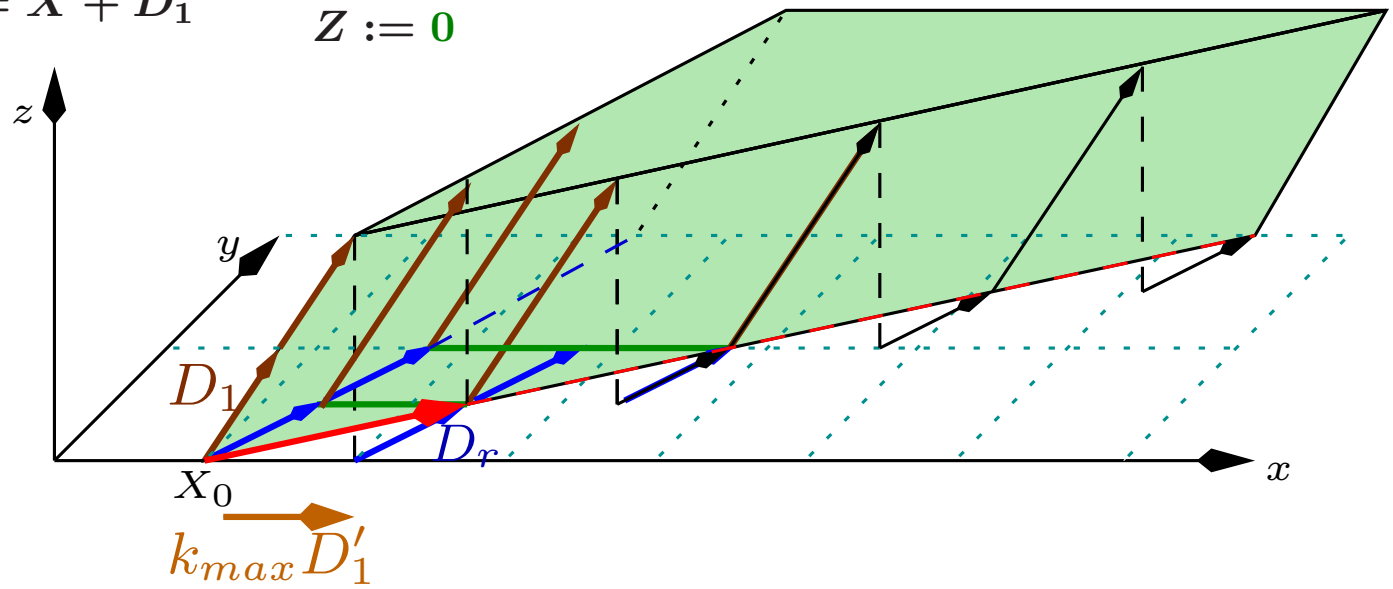
Une combinaison translation et remise à constante (1)



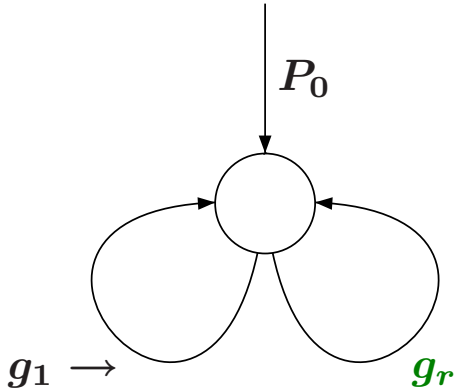
$$g_1 : Z \leqslant K_1,\ g_r = true,$$
$$P_0 \subseteq \{Z = 0\}$$

$X := X + D_1$

$Y := Y + D_r$
$Z := 0$



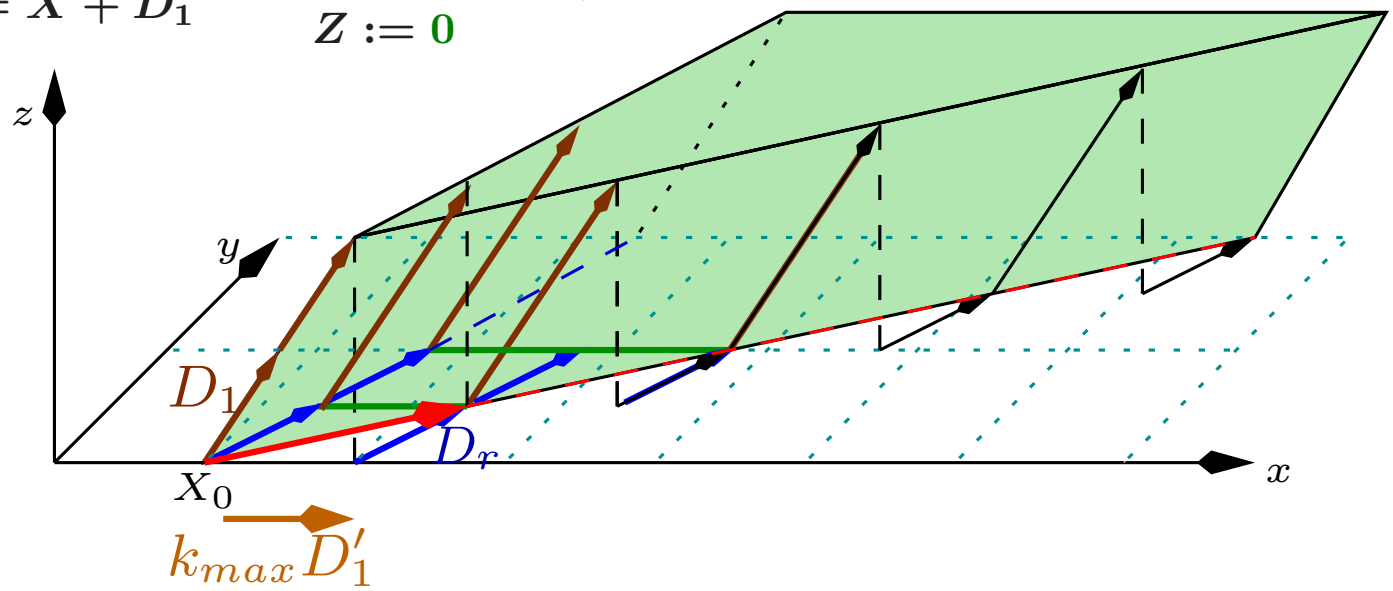
Une combinaison translation et remise à constante (1)



$$g_1 : Z \leqslant K_1,\ g_r = true,$$
$$P_0 \subseteq \{Z = 0\}$$

$X := X + D_1$

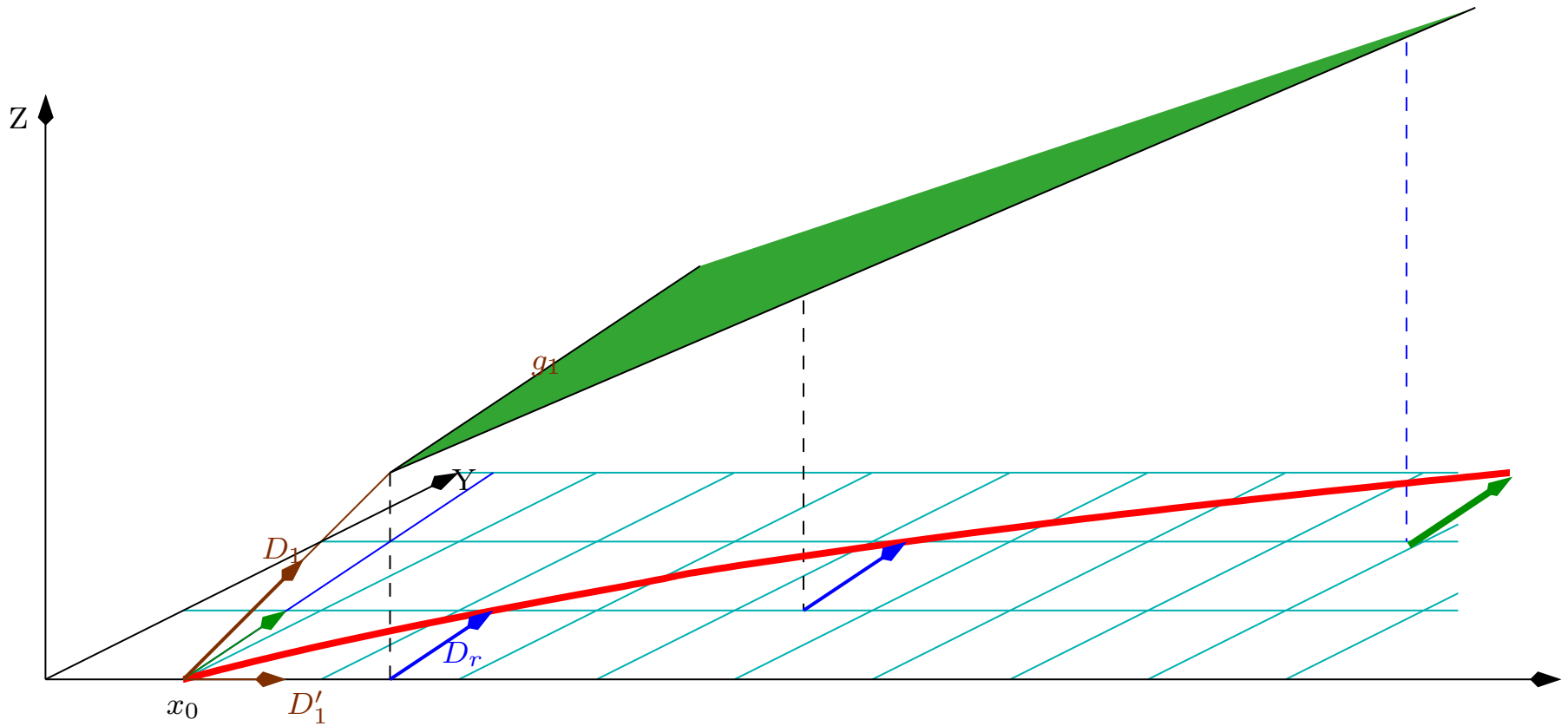
$Y := Y + D_r$
$Z := 0$



$$P_0 \nearrow \{D_1, D_r, k_{max} D'_1 + D_r\} \cap post(g_1)$$

Une combinaison translation et remise à constante (2)

Mais ... Résultat valable que si $g_1 = Z \leqslant K_1$ (à changement de variable près).



Plusieurs boucles - Résultats

Résultats

- Un algorithme pour le cas de deux boucles simultanées de translation. Dans certains cas, meilleure approximation convexe.
- Extension au cas p boucles de translation.

Plusieurs boucles - Résultats

Résultats

- Un algorithme pour le cas de deux boucles simultanées de translation. Dans certains cas, meilleure approximation convexe.
- Extension au cas p boucles de translation.
- Une sous-classe intéressante de combinaison translation/translation remise à constante : relations « **vitesse** ».
- Traitement partiel du cas p boucles translations combinées avec une boucle remise à constante.

Plusieurs boucles - Résultats

Résultats

- Un algorithme pour le cas de deux boucles simultanées de translation. Dans certains cas, meilleure approximation convexe.
- Extension au cas p boucles de translation.
- Une sous-classe intéressante de combinaison translation/translation remise à constante : relations « **vitesse** ».
- Traitement partiel du cas p boucles translations combinées avec une boucle remise à constante.

▶ Dans les autres cas, élargissement.

Caractéristiques d'ASPIC (1)

ASPIC : Accelerated Symbolic Polyhedral Invariant Computation

Caractéristiques de l'outil Aspic :

- Utilisation d'un moteur générique de calcul de point fixe [B. Jeannet] et de Polka (bibliothèque de polyèdres).
- Analyse des relations linéaires en avant, avec élargissement et accélération.
- Calculs d'invariants (+sûreté) à partir d'un langage d'automates ou de Lustre.

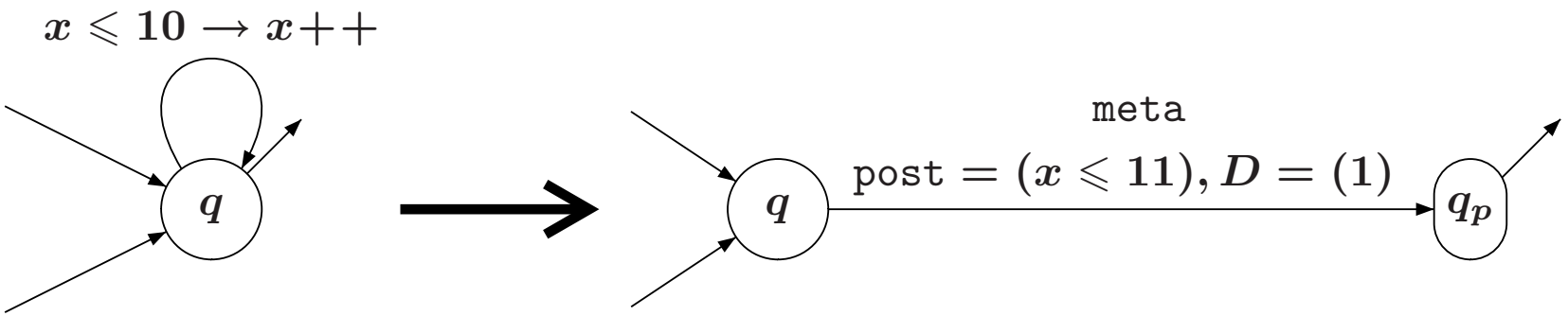
Caractéristiques d'Aspic (2)

Mise en œuvre :

- Un langage textuel d'automates (Fast) avec ou sans but de preuve (formule)
- Structure interne GFC + ensemble éventuel de points de contrôle « mauvais ».
- Précalcul : détection des « configurations » accélérables, composantes fortement connexes, stratégie de calcul, ...
- Transformation de la structure de graphe : **meta-transitions**
- Calcul classique + accélérations.
- Sorties : invariants + diagnostic.

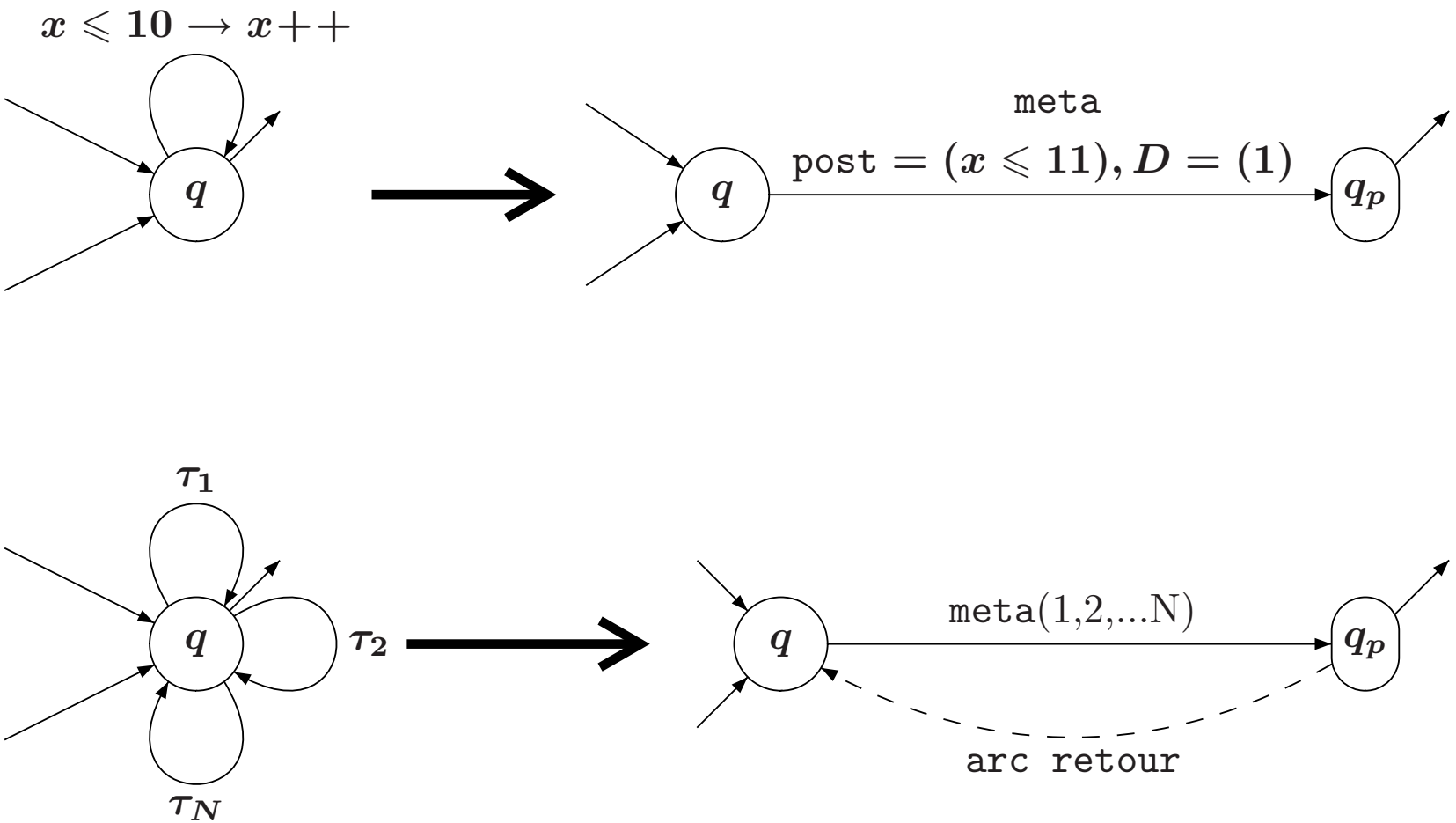
Transformation de la structure du graphe

Les meta-transitions



Transformation de la structure du graphe

Les meta-transitions



Résultats expérimentaux (1)

Nom	ARL classique	**ASPIC**	Gopan/Reps
Exemples sans remise à constante			
Hal79a	$\{0 \leqslant j, 2j \leqslant i \leqslant 104\}$	$\{\boldsymbol{i + 2j \leqslant 204}, i \leqslant 104$ $0 \leqslant j, 2j \leqslant i\}$	idem Aspic
Hal79b	$\{0 \leqslant y \leqslant x \leqslant 102\}$	$\{0 \leqslant y \leqslant x \leqslant 102$ $\boldsymbol{x + y <= 202}\}$	idem Aspic
Chaudière	$\{0 \leqslant x \leqslant f \leqslant t\}$	$\{\boldsymbol{6f \leqslant t + 5x},$ $0 \leqslant x \leqslant 10, x \leqslant f\}$	$\{0 \leqslant x \leqslant f \leqslant t\}$
Exemples avec remises à constante			
VSimple	$\{0 \leqslant s \leqslant d, 0 \leqslant t\}$	$\{0 \leqslant s \leqslant 4,$ $s \leqslant d, \boldsymbol{d \leqslant 4t + s}\}$	$\{0 \leqslant s \leqslant d, \boldsymbol{s \leqslant 4}, 0 \leqslant t\}$
Voiture	$\{0 \leqslant s \leqslant d, 0 \leqslant t\}$	$\{0 \leqslant s \leqslant 2, s \leqslant d,$ $\boldsymbol{d \leqslant 2t + s}, t \leqslant 3\}$	$\{0 \leqslant s \leqslant d, \boldsymbol{s \leqslant 2}, 0 \leqslant t\}$

Gagne en **précision** et en **efficacité**.

Résultats expérimentaux (2)

Quelques applications

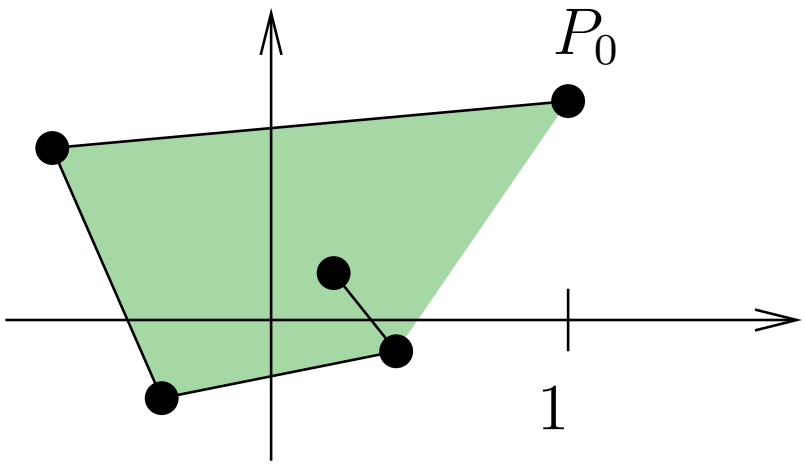
- Accessibilité dans des automates à compteurs (sémantique de SystemC), une centaine de points de contrôle, J. Cornet.
- Propriétés numériques d'automates modélisant une consommation d'énergie de (réseaux de) capteurs, L. Samper et F. Maraninchi.
- Vérification de programmes manipulant des listes, R. Iosif et S. Perarnau.

Conclusion

- Analyse des relations linéaires et amélioration de la précision.
- Étude des méthodes d'accélération et de leur mise en œuvre.
- Approche combinant analyse des relations linéaires et **Accélération Abstraite**.
- Outil complet, résultats expérimentaux montrant un gain de précision.

Perspectives

- Applications contractantes :



$$true \rightarrow$$
$$x := -0.8y$$
$$y := 0.8x$$

- Analyse arrière.
- Combinaison booléens/numériques : pour une intégration dans NBAC (B. Jeannet).
- Améliorations de Aspic : prétraitement et gestion des cycles. . .

Merci.

Accelerated Symbolic Polyhedral Invariant Computation

`http://www-verimag.imag.fr/~gonnord/aspic/aspic.html`