Automates temporisés

Formation Systèmes à Evénements Discrets

1ère édition
Janvier 2024

Plan

1. Motivation d’ajout du temps
2. Modélisation du temps par des automates temporisés
3. Evolution des automates temporisés
4. Analyse des automates temporisés
5. Mise en application

Plan

1. Motivation d’ajout du temps
2. Modélisation du temps par des automates temporisés
3. Evolution des automates temporisés
4. Analyse des automates temporisés
5. Mise en application

1 - Motivation d’ajout du temps

1-1 Besoins de modéliser le temps

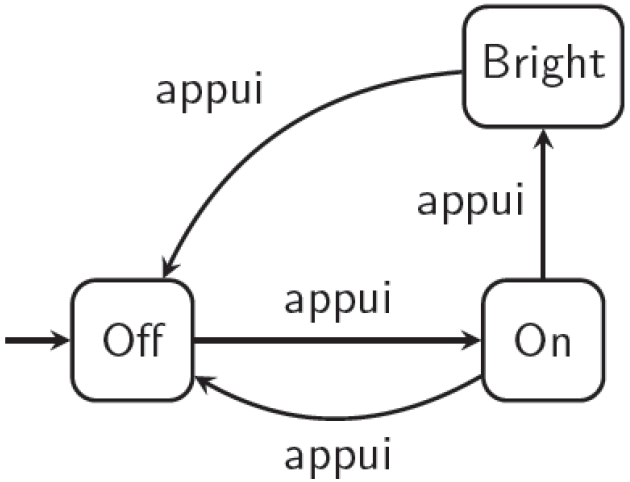
- Analyse des systèmes réactifs à temps continu, nécessité de modéliser le temps qui s’écoule

Exemple. Si j'appuie sur le bouton la lumière s'allume. Si j'appuie deux fois (rapidement) sur le bouton, la lumière s'allume plus fort. Si je rappuie sur le bouton, la lumière s’éteint.

1-1 Besoins de modéliser le temps

- Analyse des systèmes réactifs à temps continu, nécessité de modéliser le temps qui s'écoule

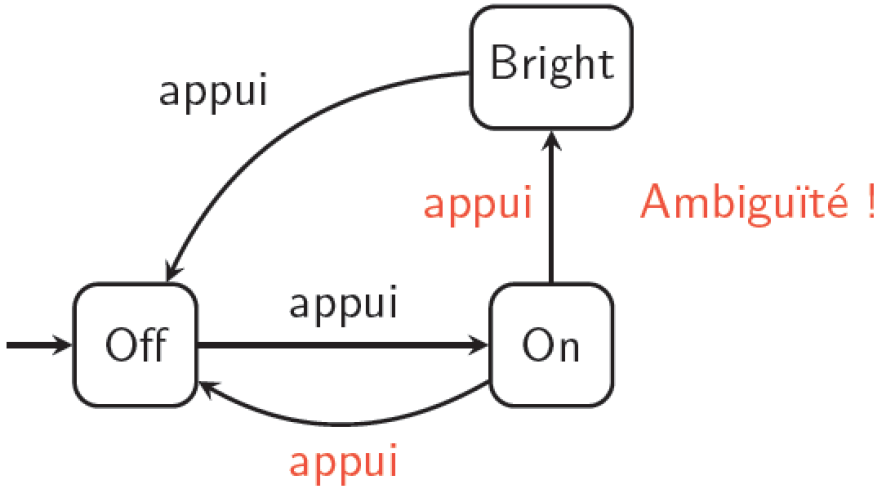
Exemple. Si j'appuie sur le bouton la lumière s'allume. Si j'appuie deux fois (rapidement) sur le bouton, la lumière s'allume plus fort. Si je rappuie sur le bouton, la lumière s'éteint.



1-1 Besoins de modéliser le temps

- Analyse des systèmes réactifs à temps continu, nécessité de modéliser le temps qui s'écoule

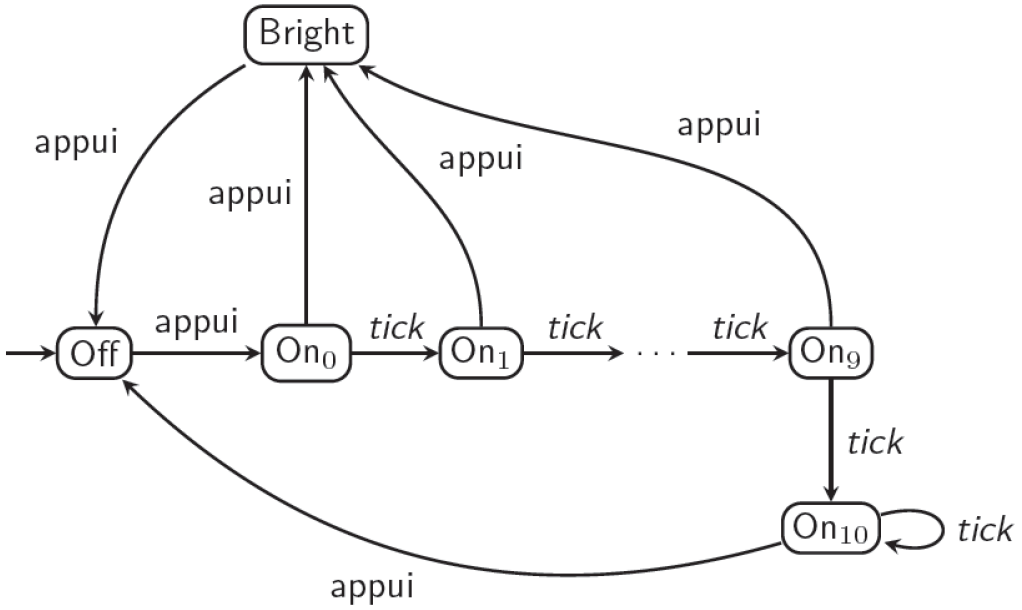
Exemple. Si j'appuie sur le bouton la lumière s'allume. Si j'appuie deux fois (rapidement) sur le bouton, la lumière s'allume plus fort. Si je rappuie sur le bouton, la lumière s'éteint.



1-2 Difficulté de modélisation

- Solution 1 : Discrétiser le temps, en ajoutant un événement « tick » d'horloge

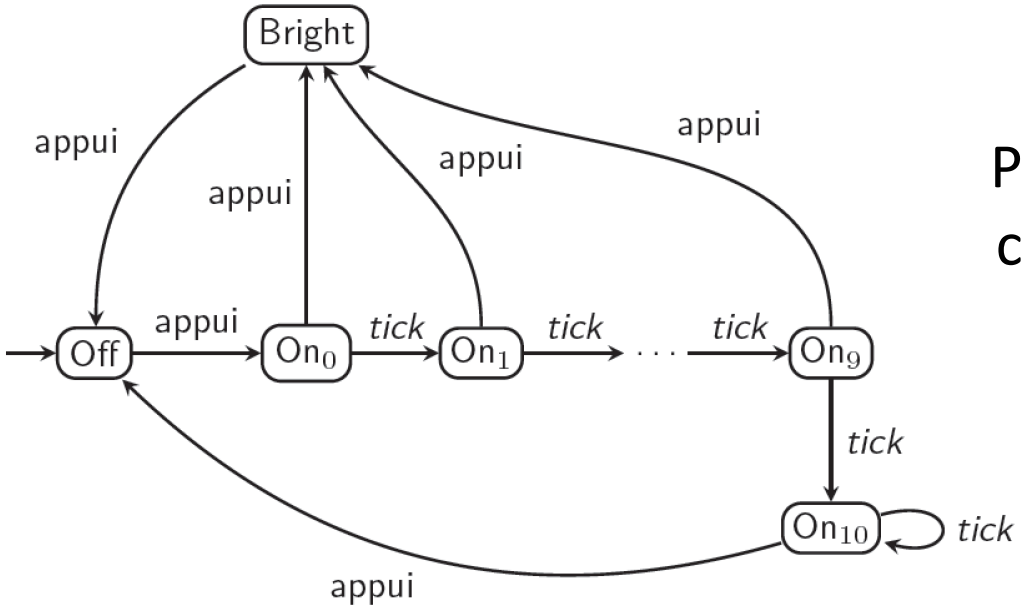
Exemple. Modélisation 1/10 seconde par un événement tick.



1-2 Difficulté de modélisation

- Solution 1 : Discrétiser le temps, en ajoutant un événement « tick » d'horloge

Exemple. Modélisation 1/10 seconde par un événement tick.



Problème : La taille du modèle croît avec la finesse du temps

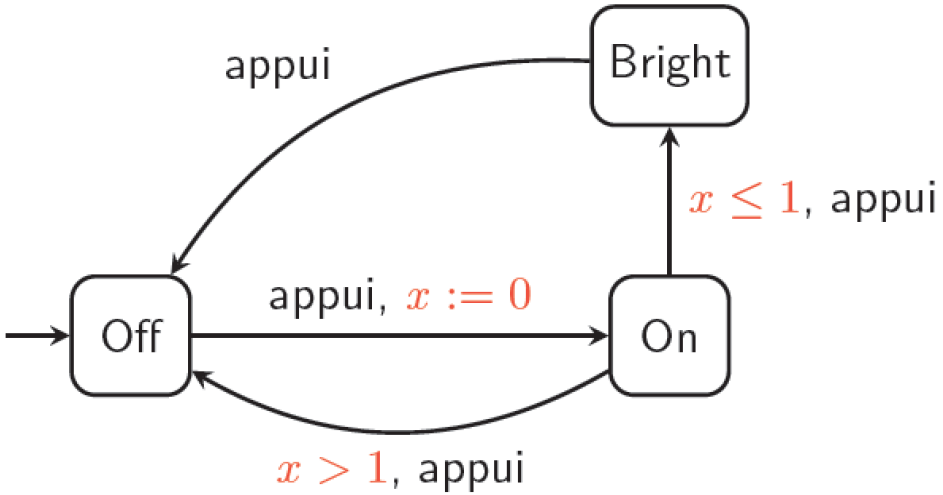
1-2 Difficulté de modélisation

- Solution 1 : Discrétiser le temps, en ajoutant un événement « tick » d'horloge
- Solution 2 : Mesurer le temps de façon continue, en utilisant une variable réelle :
  - Le temps est modélisé dans R par des horloges
  - Il est donc arbitrairement précis

1-2 Difficulté de modélisation

- Solution 1 : Discrétiser le temps, en ajoutant un événement « tick » d'horloge
- Solution 2 : Mesurer le temps de façon continue, en utilisant une variable réelle :
  - Le temps est modélisé dans R par des horloges
  - Il est donc arbitrairement précis

Exemple.



Plan

1. Motivation d’ajout du temps
2. Modélisation du temps par des automates temporisés
3. Evolution des automates temporisés
4. Analyse des automates temporisés
5. Mise en application

2 - Modélisation du temps dans les automates

2- Définition des automates temporisés

- Première définition proposée par Alur et Dill en 1991
  - Automate fini avec des horloges à valeur dans $\mathbb{R}+$
    - qui fonctionnement simultanément
    - qui peuvent être réinitialisées indépendamment
- Chaque horloge mesure le temps écoulé depuis sa dernière initialisation

2- Définition des automates temporisés

Extensions des automates à états finis

- Chaque automate possède un nombre fini de places (localité)
- Le franchissement d'une transition entre deux places est instantané
- Dans chaque place, le temps peut s'écouler
- Les transitions entre places sont conditionnées par des contraintes sur les horloges, appelées gardes
- Les transitions entre places peuvent réinitialiser des horloges de l'automate (action)
- A chaque place est associée une contrainte sur les horloges, appelée invariant

2- Définition des automates temporisés

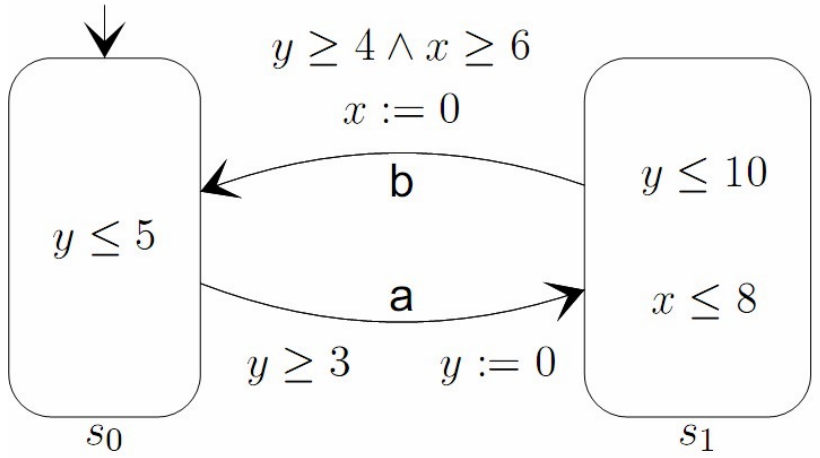
Définition formelle des automates temporisés $A = (\Sigma, X, X_0, H, I, T)$ avec

- Σ est l'ensemble fini d'événements possibles
- X est l'ensemble fini de localités
- $X_0 \in X$ est la localité initiale
- H est l'ensemble fini d'horloge à valeur dans $\mathbb{R}+$
- I : X → C(H) pour définir les invariants de places
- T ⊂ X x Σ x C(H) x 2H x X un ensemble de transitions
- Chaque e = < x, a, Φ, λ, x' > ∈ T correspond à une transition entre la localité x et la localité x', gardée par la contrainte Φ, étiquetée par a ∈ Σ, et qui réinitialise les variables λ ⊂ X

2- Définition des automates temporisés

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

La transition $s_0 \rightarrow s_1$ est étiquetée par
l'événement a, et conditionnée par la garde y ≥ 3.
Au franchissement, l'horloge y est remise à 0.



s_0 possède un invariant y ≤ 5 , c'est-à-dire qu'en s_0 l'horloge y ne peut pas être supérieur à 5

s_1 en possède deux, y ≤ 10 et x ≤ 8, c'est-à-dire qu'en s_0 l'horloge y ne peut pas être supérieur à 10 et l'horloge x ne peut pas être supérieur à 8

Plan

1. Motivation d’ajout du temps
2. Modélisation du temps par des automates temporisés
3. Evolution des automates temporisés
4. Analyse des automates temporisés
5. Mise en application

3 - Evolution des automates temporisés

3- Evolution des automates temporisés

Un automate temporisé peut évoluer de la manière suivante :

- Rester dans la localité courante et laisser le temps s'écouler
- Quitter la localité courante avant que l'invariant ne soit violé
- Franchir une transition
  - Si sa garde est vraie et que l'invariant de la place destination est satisfait
  - Suite à l'occurrence d'un événement
  - Remettre à zéro des horloges

Le franchissement d'une transition entre deux localités est considéré instantané

3- Evolution des automates temporisés

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

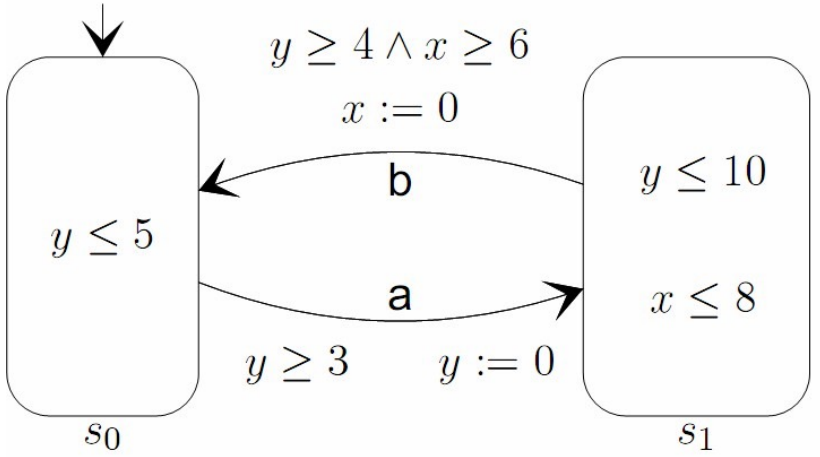
Situation 1 :

A partir de l'état (s_0, y=0, x=0) , si l'événement a se produit à l'instant 4, la transition vers s_1 est franchissable

l'horloge y est remise à 0 et l'automate arrive dans l'état (s_1, y=0, x=4)

A partir de l'état (s_1, y=0, x=4), si l'événement b se produit à l'instant 8, la transition vers s_1 est franchissable

l'horloge x est remise à 0 et l'automate arrive dans l'état (s_1, y=4, x=0)



Situation 2 :

A partir de l'état (s_0, y=0, x=0) , si l'événement a se produit à l'instant 5, la transition vers s_1 est franchissable

l'horloge y est remise à 0 et l'automate arrive dans l'état (s_1, y=0, x=5)

A partir de l'état (s_1, y=0, x=4), si l'événement b se produit à l'instant 8, la transition vers s_1 n'est pas franchissable car y=3

Situation 3 :

A partir de l'état (s_0, y=0, x=0) , si l'événement a se produit à l'instant 9, la transition vers s_1 n'est pas franchissable, car l'invariant x<=8 est violé

3- Evolution des automates temporisés

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

Situation 1 :

A partir de l'état (s_0, y=0, x=0) , si l'événement a se produit à l'instant 4, la transition vers s_1 est franchissable

l'horloge y est remise à 0 et l'automate arrive dans l'état (s_1, y=0, x=4)

A partir de l'état (s_1, y=0, x=4), si l'événement b se produit à l'instant 8, la transition vers s_1 est franchissable

l'horloge x est remise à 0 et l'automate arrive dans l'état (s_1, y=4, x=0)

Situation 2 :

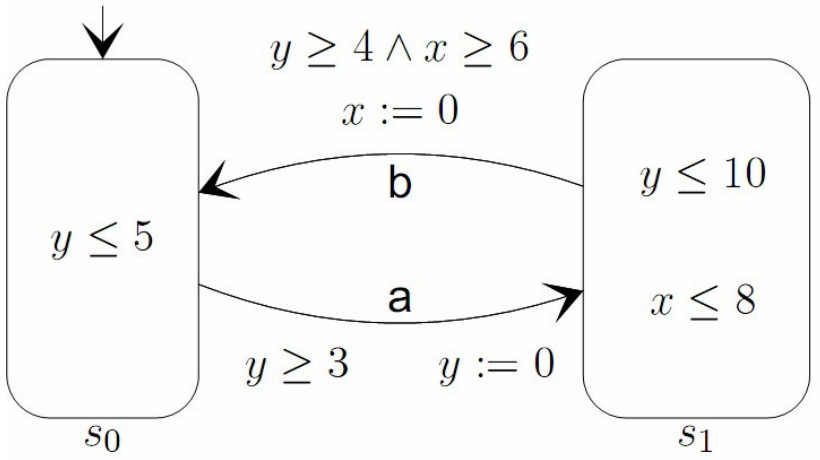
A partir de l'état (s_0, y=0, x=0) , si l'événement a se produit à l'instant 5, la transition vers s_1 est franchissable

l'horloge y est remise à 0 et l'automate arrive dans l'état (s_1, y=0, x=5)

A partir de l'état (s_1, y=0, x=4), si l'événement b se produit à l'instant 8, la transition vers s_1 n'est pas franchissable car y=3

Situation 3 :

A partir de l'état (s_0, y=0, x=0) , si l'événement a se produit à l'instant 9, la transition vers s_1 n'est pas franchissable, car l'invariant x<=8 est violé



3- Evolution des automates temporisés

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

Situation 1 :

A partir de l'état (s_0, y=0, x=0) , si l'événement a se produit à l'instant 4, la transition vers s_1 est franchissable

l'horloge y est remise à 0 et l'automate arrive dans l'état (s_1, y=0, x=4)

A partir de l'état (s_1, y=0, x=4), si l'événement b se produit à l'instant 8, la transition vers s_1 est franchissable

l'horloge x est remise à 0 et l'automate arrive dans l'état (s_1, y=4, x=0)

Situation 2 :

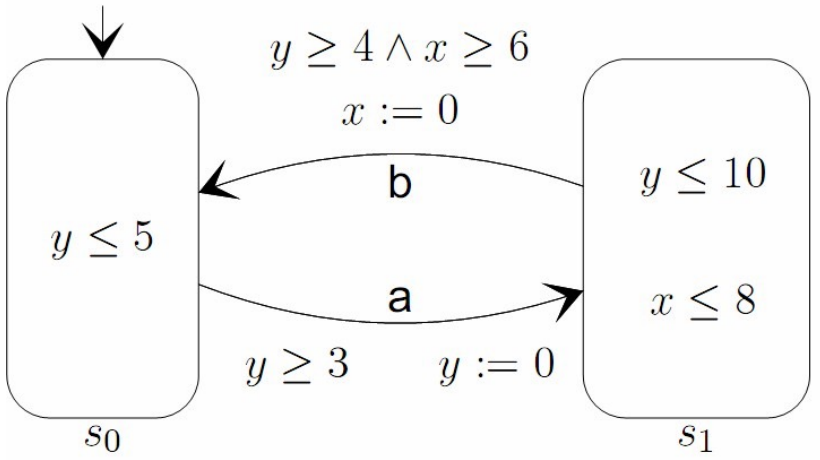
A partir de l'état (s_0, y=0, x=0) , si l'événement a se produit à l'instant 5, la transition vers s_1 est franchissable

l'horloge y est remise à 0 et l'automate arrive dans l'état (s_1, y=0, x=5)

A partir de l'état (s_1, y=0, x=4), si l'événement b se produit à l'instant 8, la transition vers s_1 n'est pas franchissable car y=3

Situation 3 :

A partir de l'état (s_0, y=0, x=0) , si l'événement a se produit à l'instant 9, la transition vers s_1 n'est pas franchissable, car l'invariant x<=8 est violé



Plan

1. Motivation d’ajout du temps
2. Modélisation du temps par des automates temporisés
3. Evolution des automates temporisés
4. Analyse des automates temporisés
5. Mise en application

4 - Analyse des automates temporisés

4-1 Analyse des automates temporisés

L'analyse des automates temporisés est basée sur les mêmes propriétés que les automates à états :

- Un état q de A est accessible à partir d'un état p s'il existe une trajectoire dans A dont l'origine est p et dont q est l'extrémité. L'état q est dit accessible s'il est accessible à partir de l'état initial ;
- Un état p est co-accessible à un état q s'il existe une trajectoire dans A dont l'origine est p et dont q est l'extrémité. L'état p est dit co-accessible s'il est co-accessible à un état marqué.

4-2 Passage vers automate à états

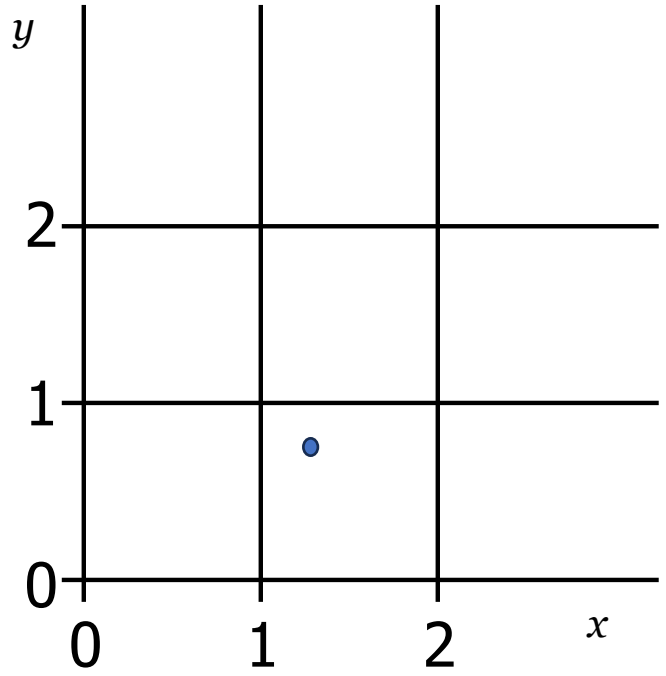
En fonction des valeurs des horloges, le franchissement d'une transition ou l'accès à une localité n'est pas toujours autorisé.

Pour revenir à un automate à états, il faut définir le graphe des régions

4-2 Passage vers automate à états

En fonction des valeurs des horloges, le franchissement d'une transition ou l'accès à une localité n'est pas toujours autorisé.

Pour revenir à un automate à états, il faut définir le graphe des régions

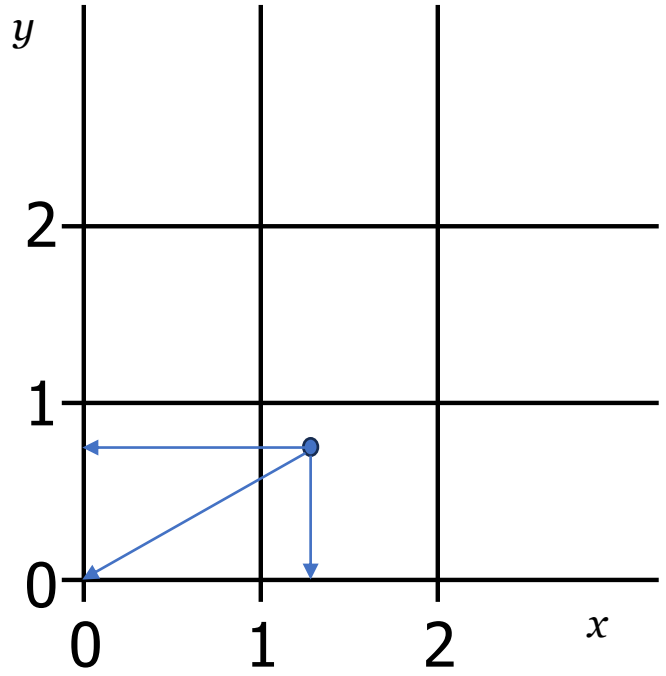


A un instant t, avec des valeurs d'horloge pour x et y,
2 évolutions possibles :

4-2 Passage vers automate à états

En fonction des valeurs des horloges, le franchissement d'une transition ou l'accès à une localité n'est pas toujours autorisé.

Pour revenir à un automate à états, il faut définir le graphe des régions



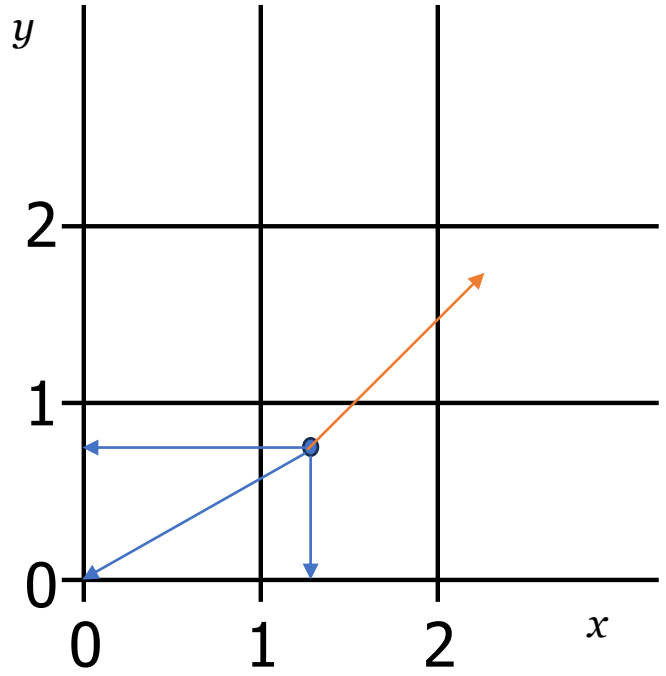
A un instant t, avec des valeurs d'horloge pour x et y,
2 évolutions possibles :

- Remise à zéro

4-2 Passage vers automate à états

En fonction des valeurs des horloges, le franchissement d'une transition ou l'accès à une localité n'est pas toujours autorisé.

Pour revenir à un automate à états, il faut définir le graphe des régions



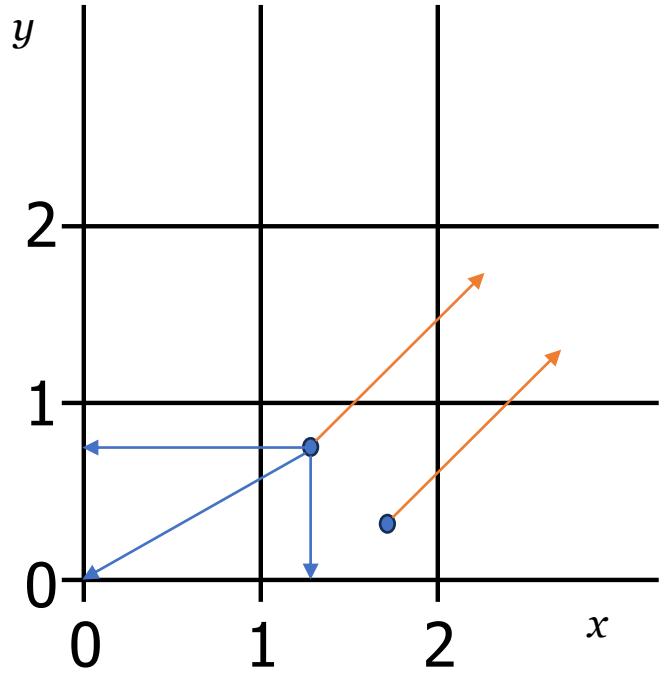
A un instant t, avec des valeurs d'horloge pour x et y,
2 évolutions possibles :

- Remise à zéro
- Ecoulement du temps

4-2 Passage vers automate à états

En fonction des valeurs des horloges, le franchissement d'une transition ou l'accès à une localité n'est pas toujours autorisé.

Pour revenir à un automate à états, il faut définir le graphe des régions



A un instant t, avec des valeurs d'horloge pour x et y,
2 évolutions possibles :

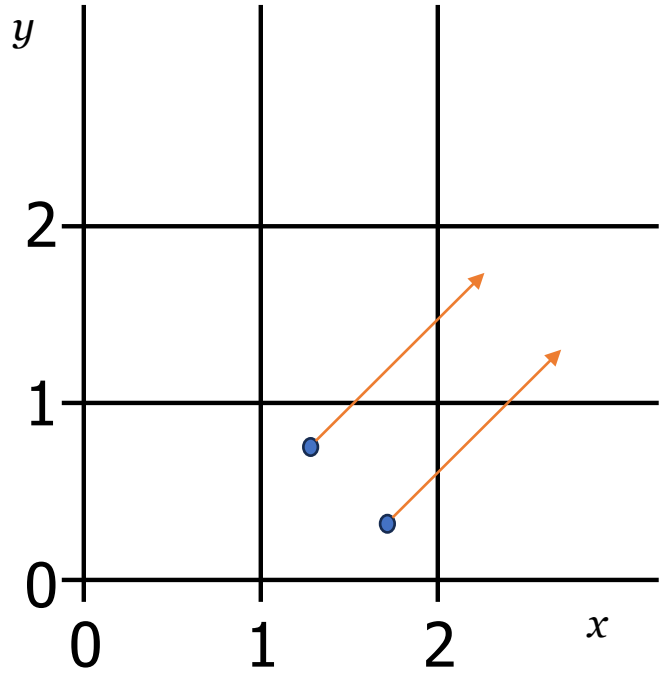
- Remise à zéro
- Ecoulement du temps

En fonction de l'instant t, les évolutions peuvent être différentes

4-2 Passage vers automate à états

En fonction des valeurs des horloges, le franchissement d'une transition ou l'accès à une localité n'est pas toujours autorisé.

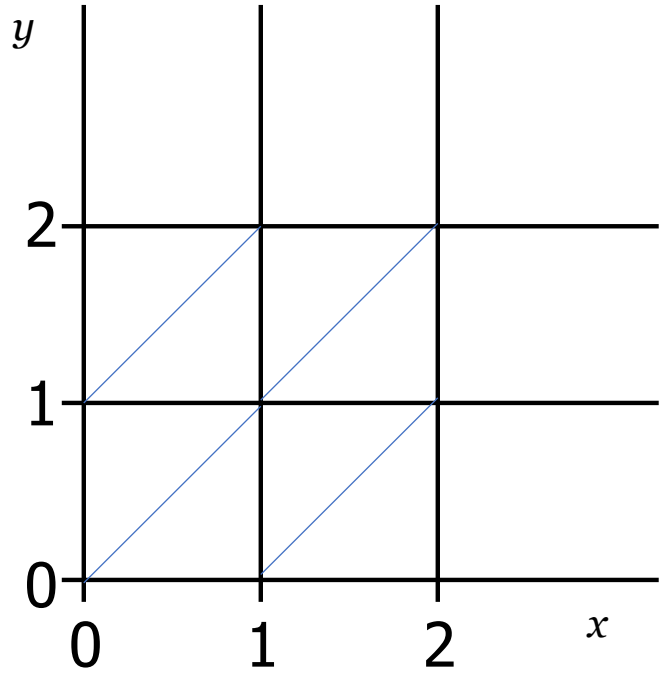
Pour revenir à un automate à états, il faut définir le graphe des régions



Le graphe des régions : discrétiser le domaine des horloges en un nombre fini de classes d'équivalence pour les évaluations des horloges, **à partir des contraintes d'horloges (**constantes entières)

4-2 Passage vers automate à états

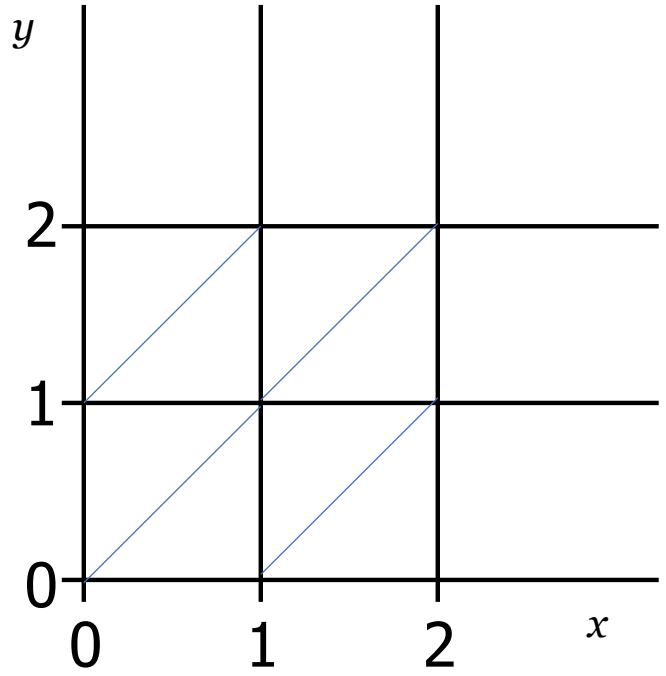
Objectif : Regrouper les évolutions d'horloge dans des classes d'équivalence : les régions dont toutes les évolutions d'horloge satisfont les mêmes contraintes d'horloge sont regroupées ensemble.



4-2 Passage vers automate à états

Méthodologie pour passer d'un automate temporisé vers un automate à états finis

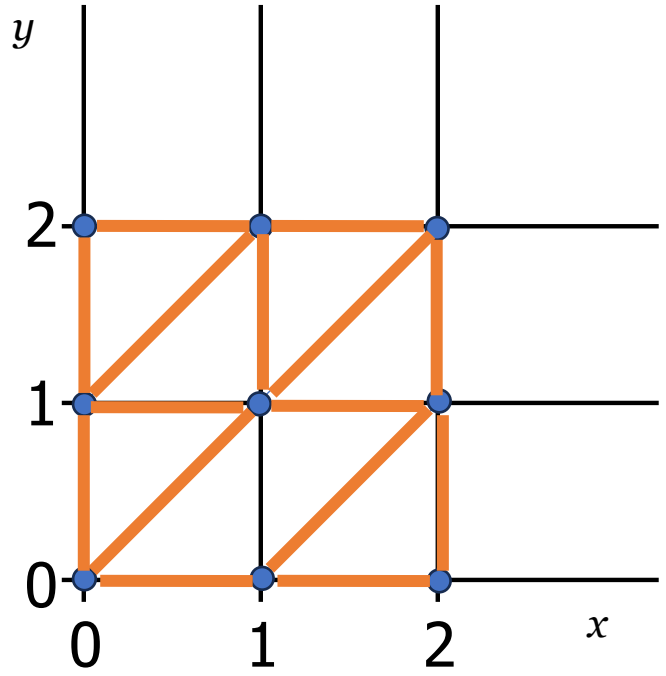
1. Définir les parties fractionnaires : Frac(x) <=, =, >= Frac(x)



4-2 Passage vers automate à états

Méthodologie pour passer d'un automate temporisé vers un automate à états finis

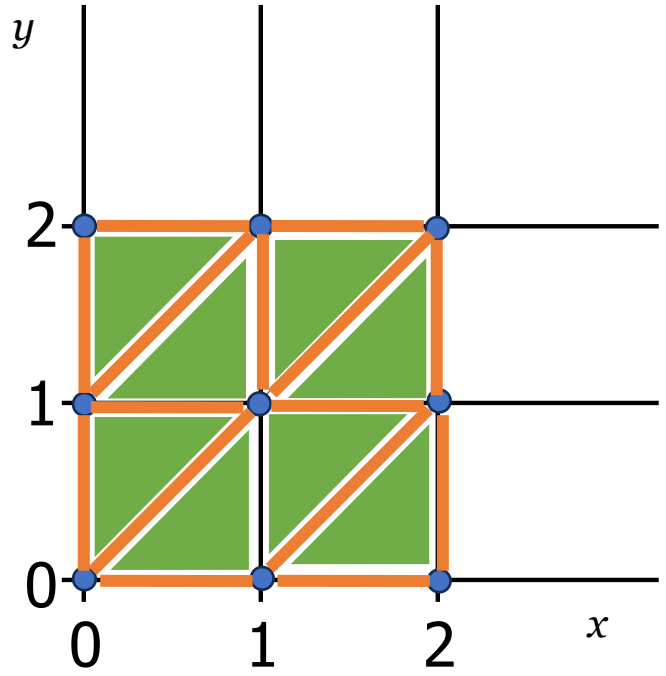
1. Définir les parties fractionnaires : Frac(x) <=, =, >= Frac(x)
2. Définir les régions



4-2 Passage vers automate à états

Méthodologie pour passer d'un automate temporisé vers un automate à états finis

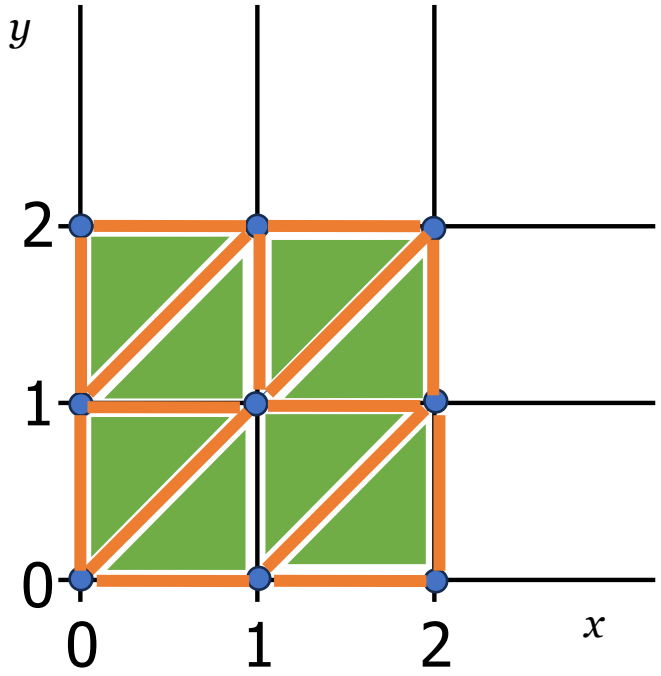
1. Définir les parties fractionnaires : Frac(x) <=, =, >= Frac(x)
2. Définir les régions



4-2 Passage vers automate à états

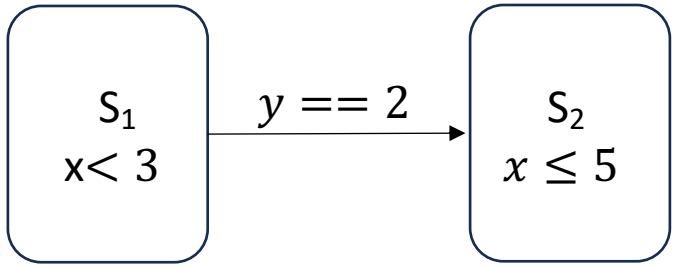
Méthodologie pour passer d'un automate temporisé vers un automate à états finis

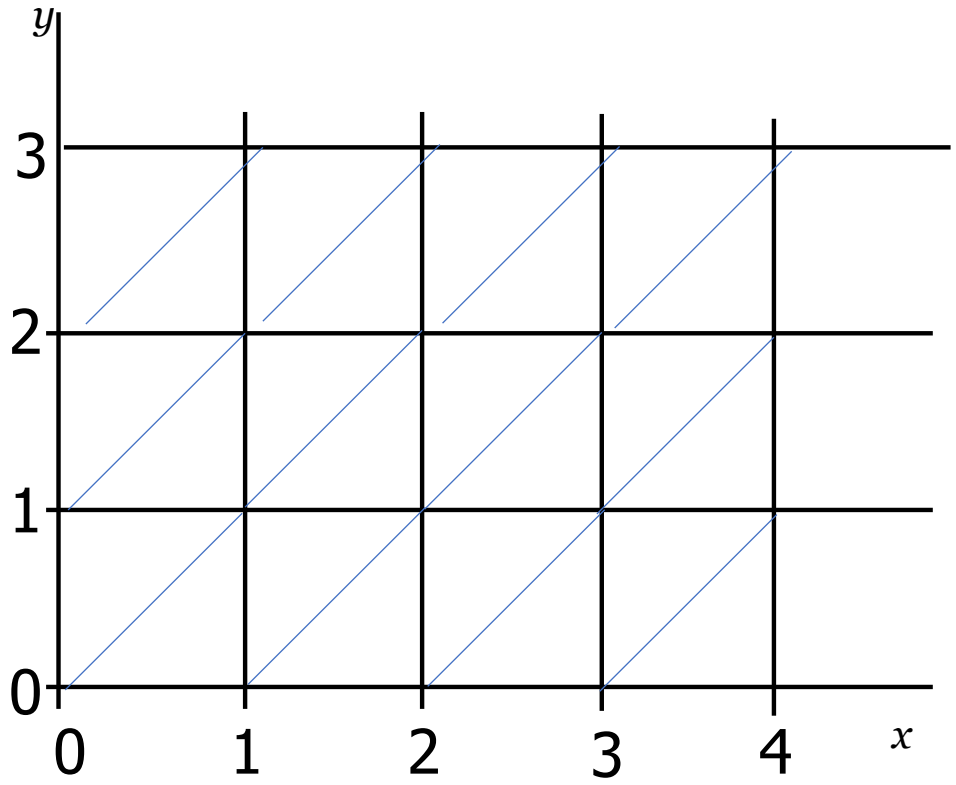
1. Définir les parties fractionnaires : Frac(x) <=, =, >= Frac(x)
2. Définir les régions
3. Dans chaque région, identifier :
   a. Si une transition est franchissable
   b. Si l'invariant de la localité courante est violé
   c. Si l'invariant de la localité suivante est violé
4. Si deux régions ont les mêmes évolutions, et respectent les invariants des localités courante et suivante, elles sont regroupées



4-2 Passage vers automate à états

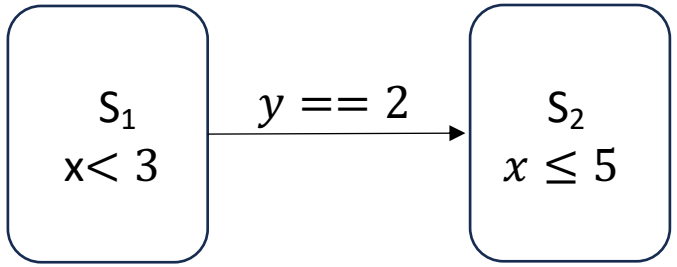
Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)



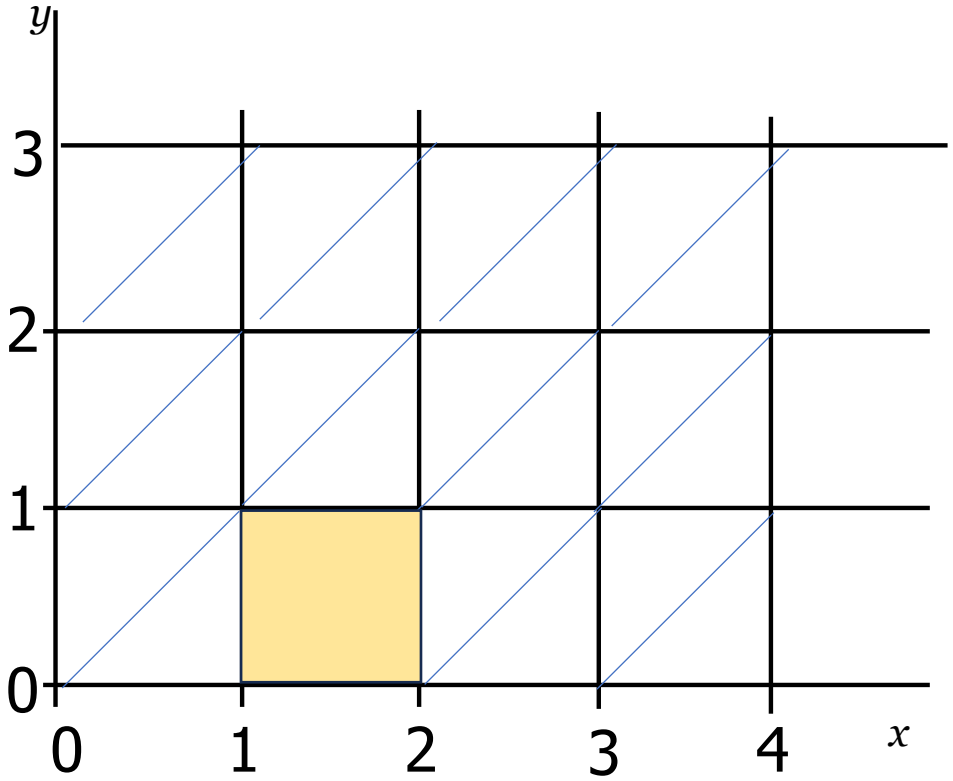


4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

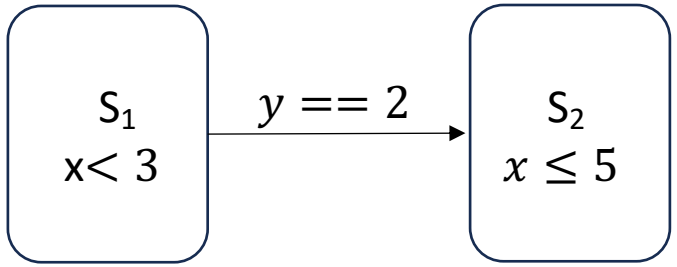


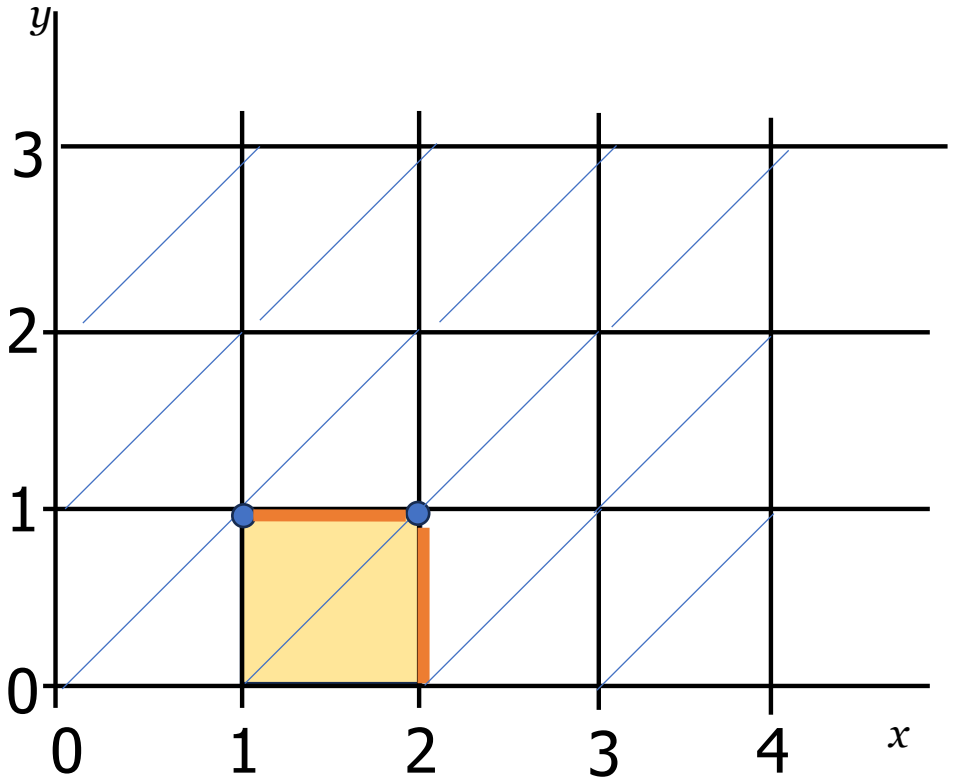
Région de départ : en S_1, $1 < x < 2$ et $0 < y < 1$



4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

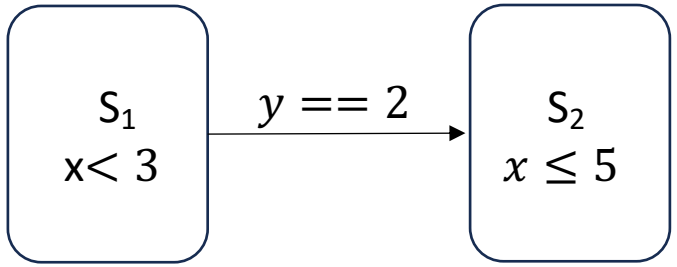




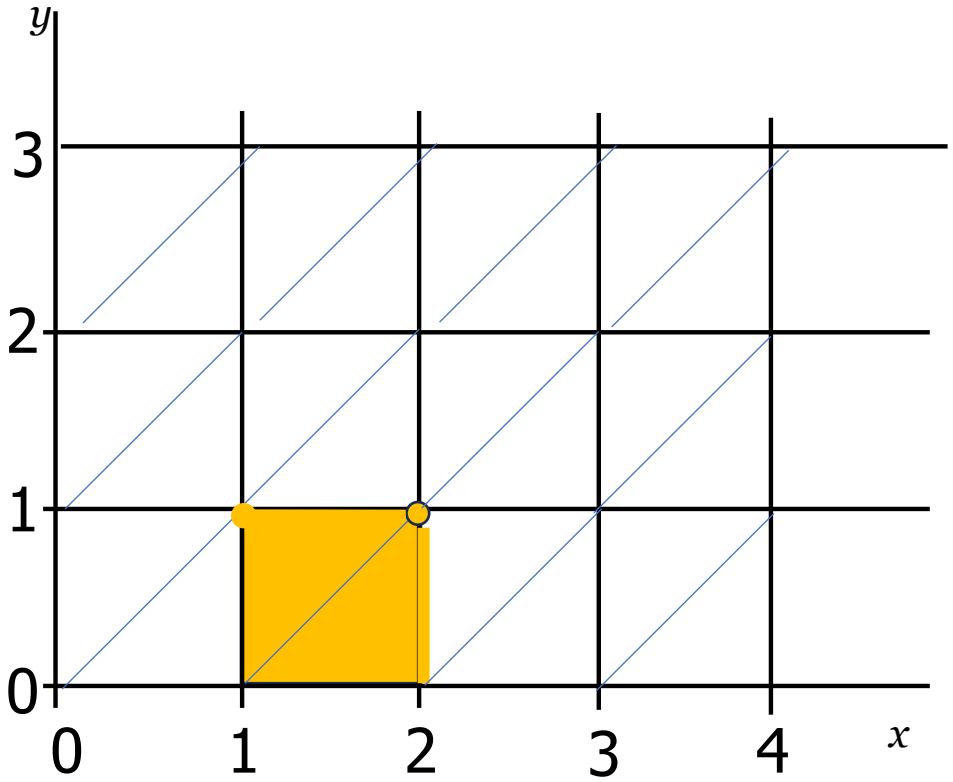
Région jaune : en S_1, $1 < x < 2$ et $0 < y < 1$,
rester en s1 ? Transition franchissable ?

4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

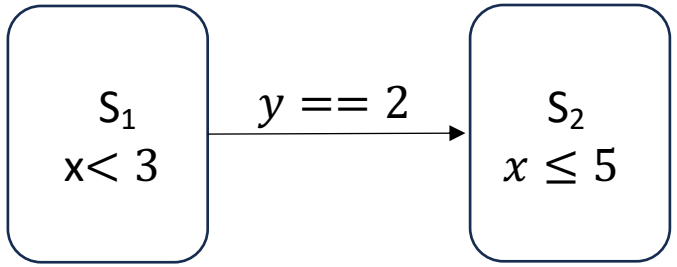


Région jaune : en S_1, $1 < x \leq 2$ et $0 < y \leq 1$,

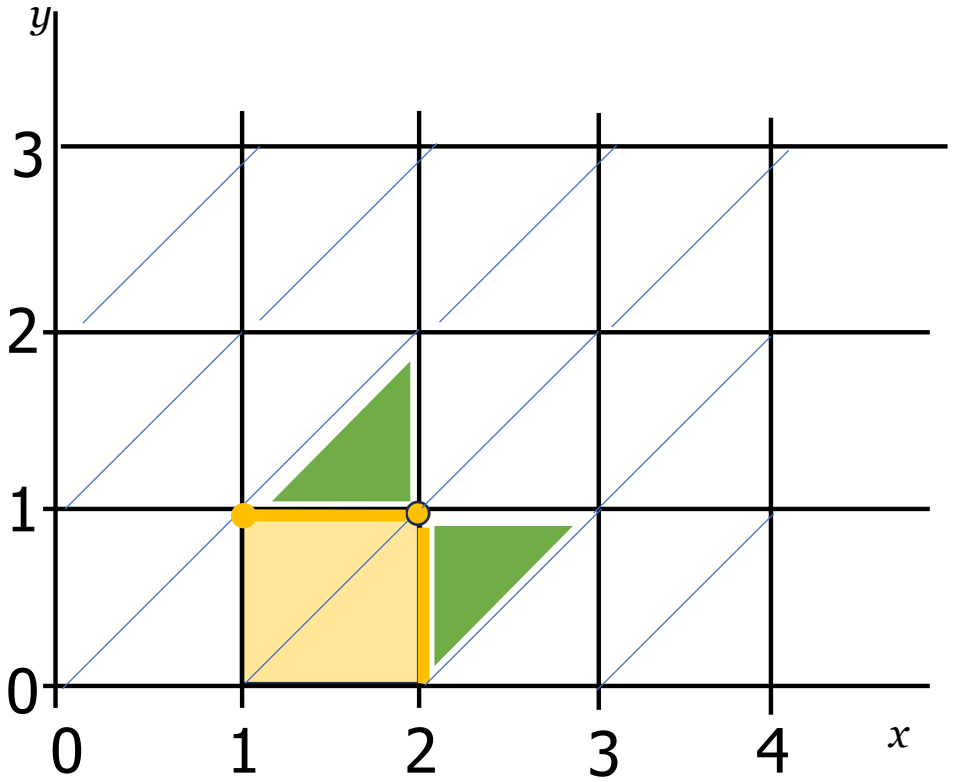


4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

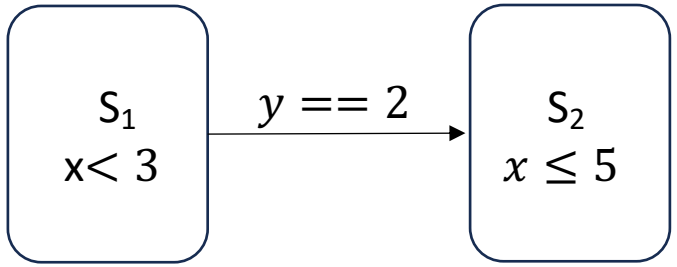


Région jaune : en S_1, $1 < x \leq 2$ et $0 < y \leq 1$,

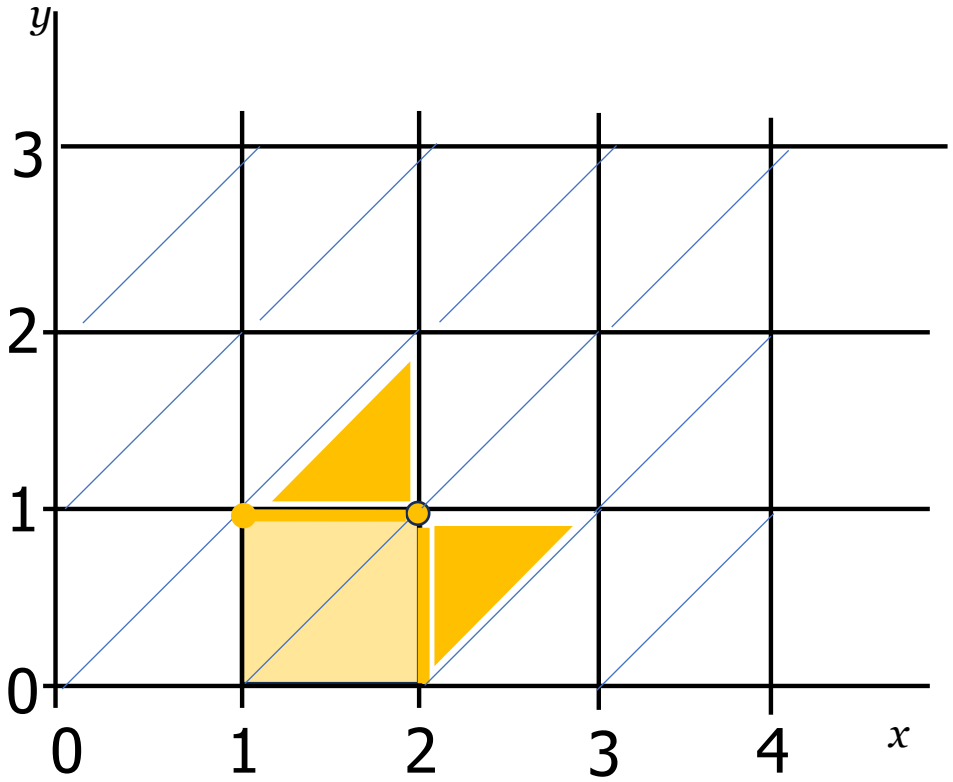


4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

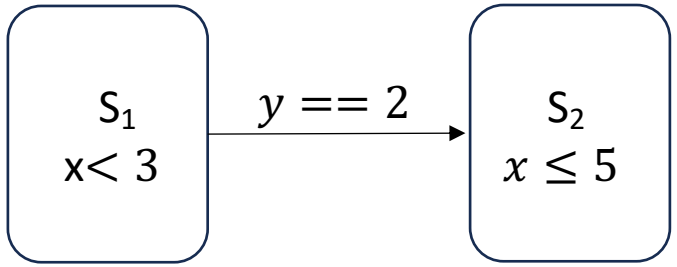


Région jaune : en S_1, $1 < x < 3$ et $0 < y < 2$,

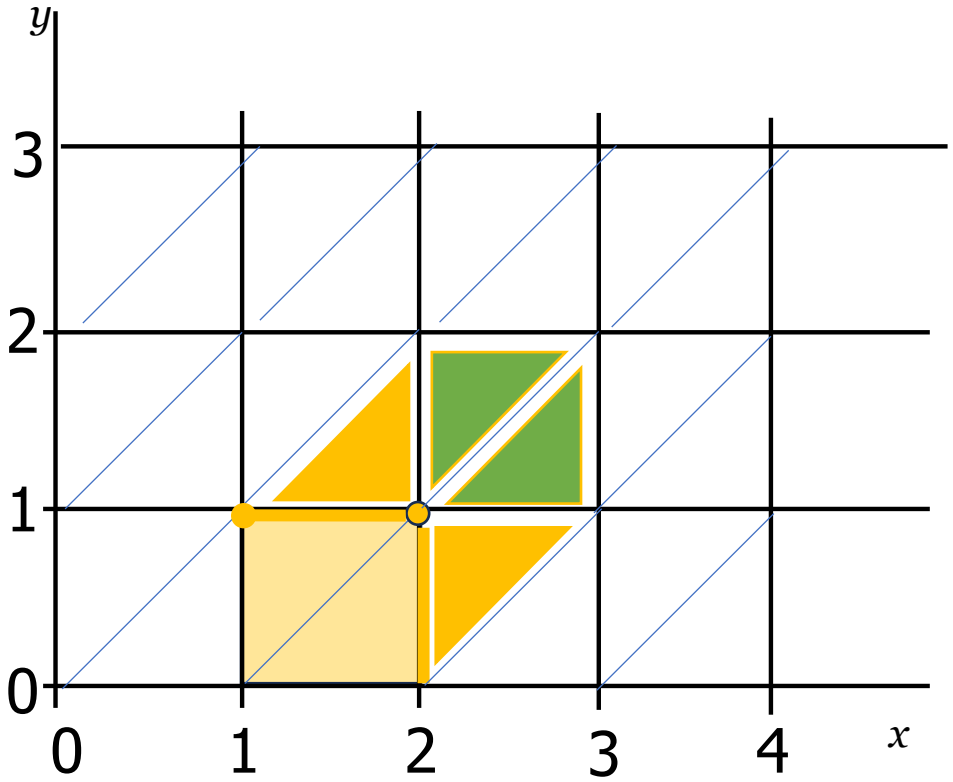


4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

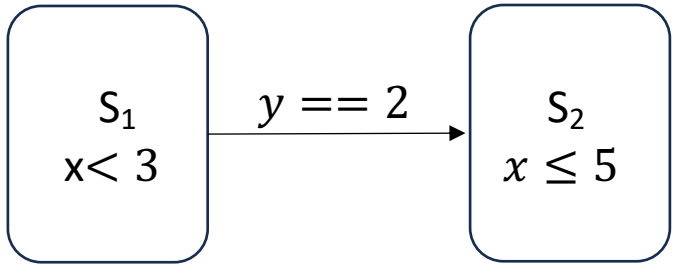


Région jaune : en S_1, $1 < x < 3$ et $0 < y < 2$,

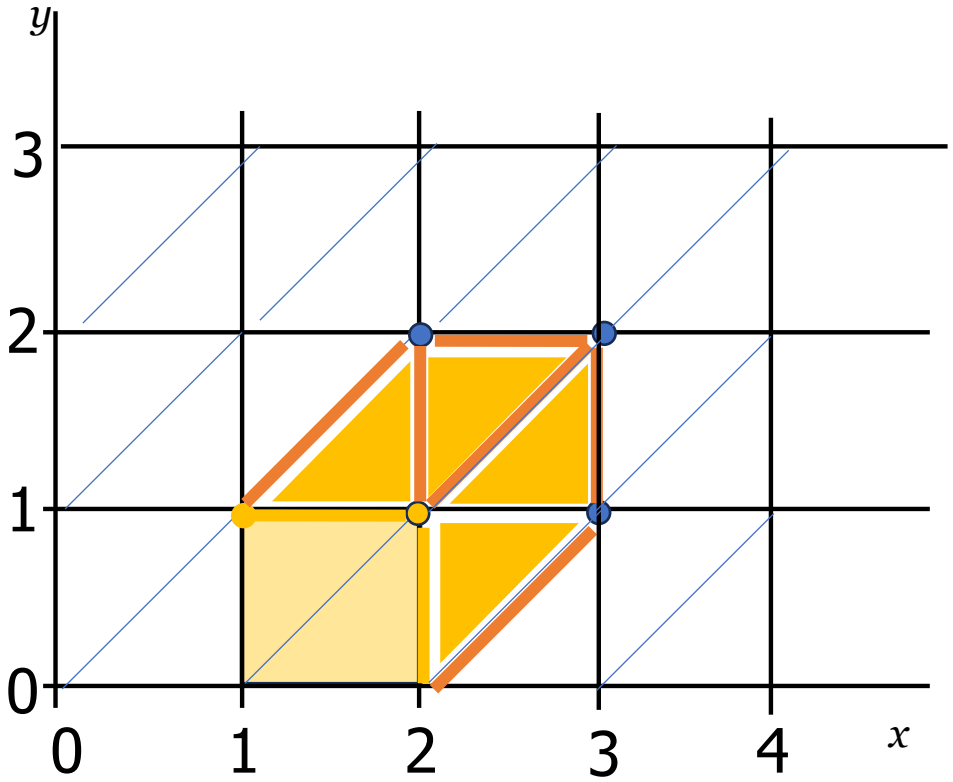


4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

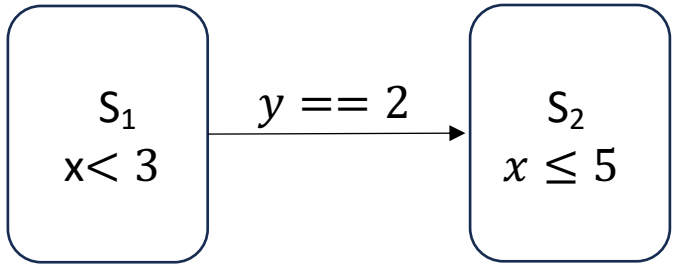


Région jaune : en S_1, $1 < x < 3$ et $0 < y < 2$,

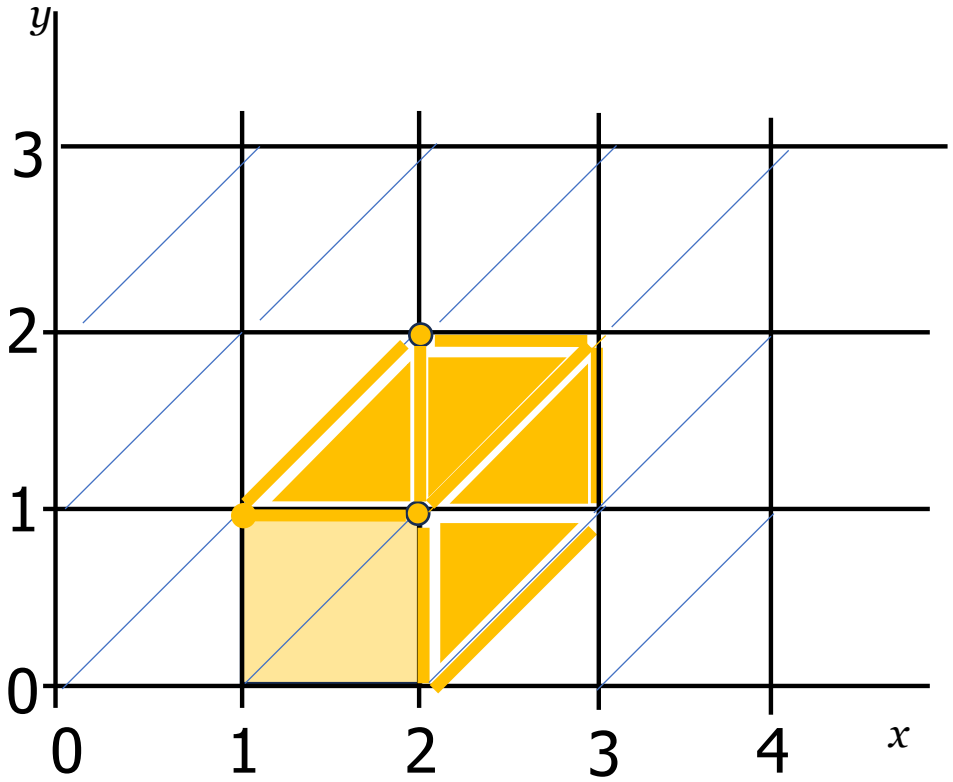


4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

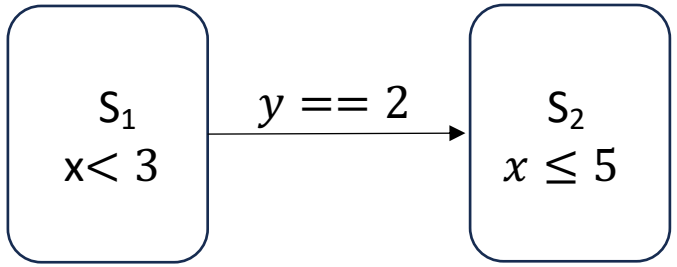


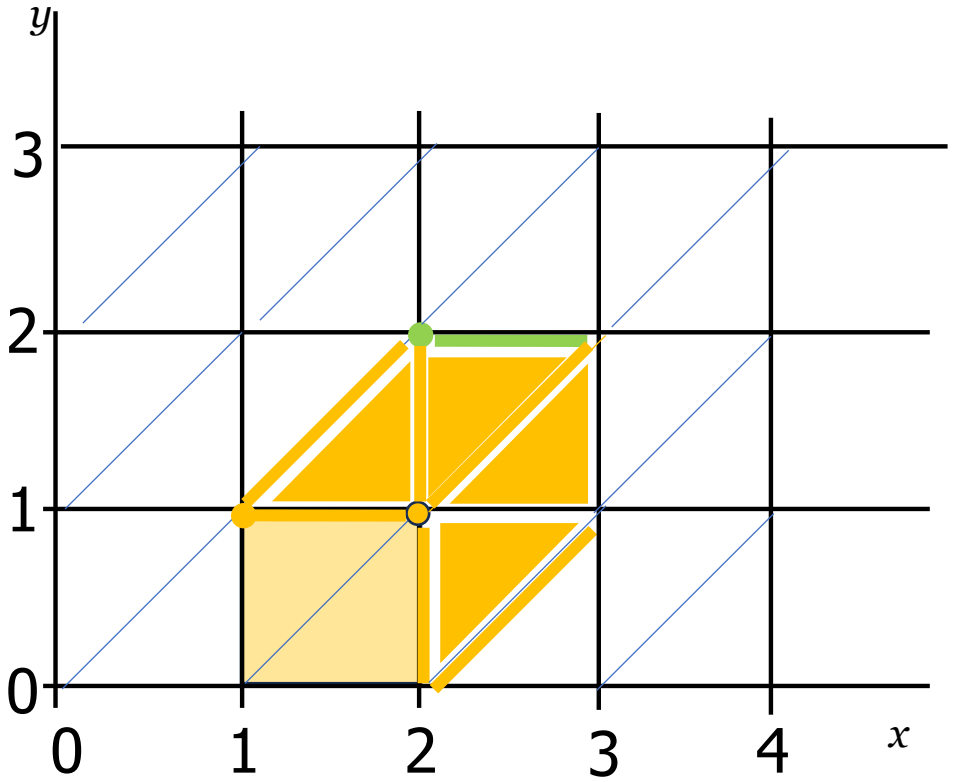
Région jaune : en S_1, $1 < x < 3$ et $0 < y \leq 2$,



4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

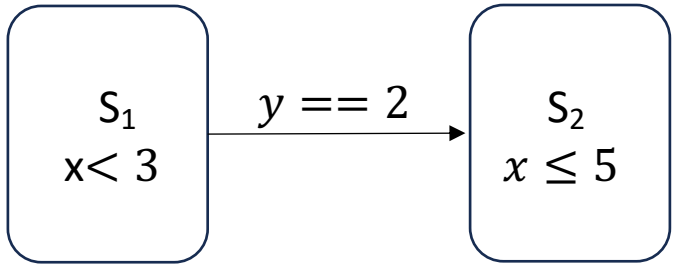




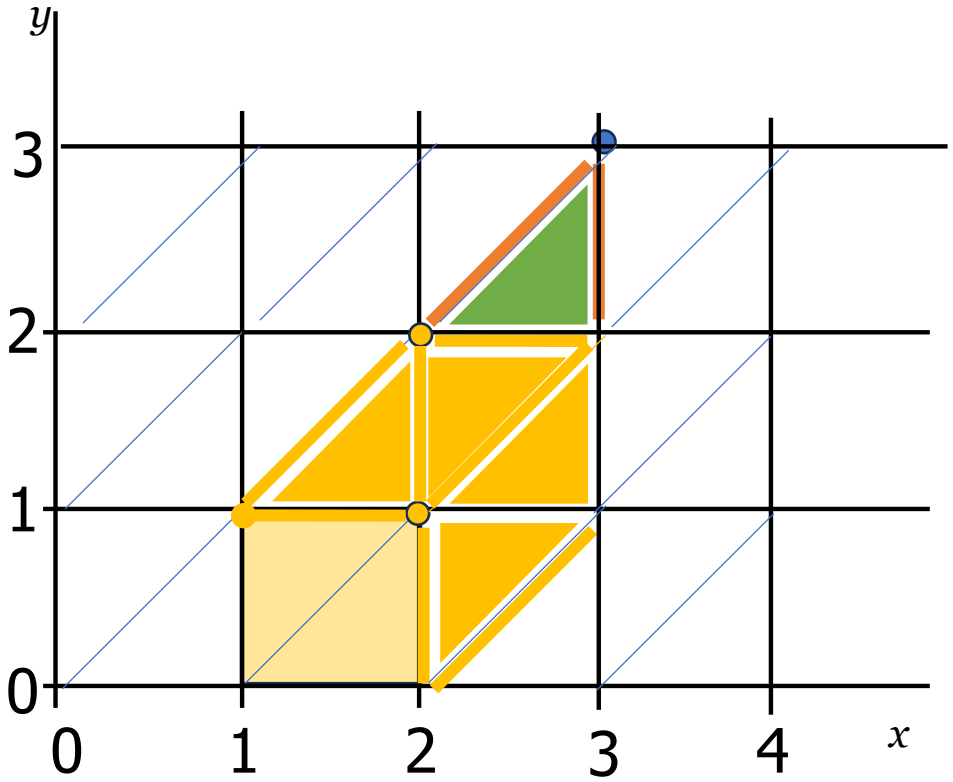
Région jaune : en S_1, $1 < x < 3$ et $0 < y \leq 2$
Région verte : en S_1, $1 < x < 3$ et $y = 2$, transition franchissable

4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

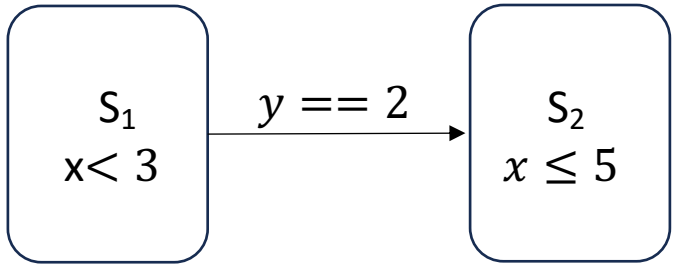


Région jaune : en S_1, $1 < x < 3$ et $0 < y \leq 2$,

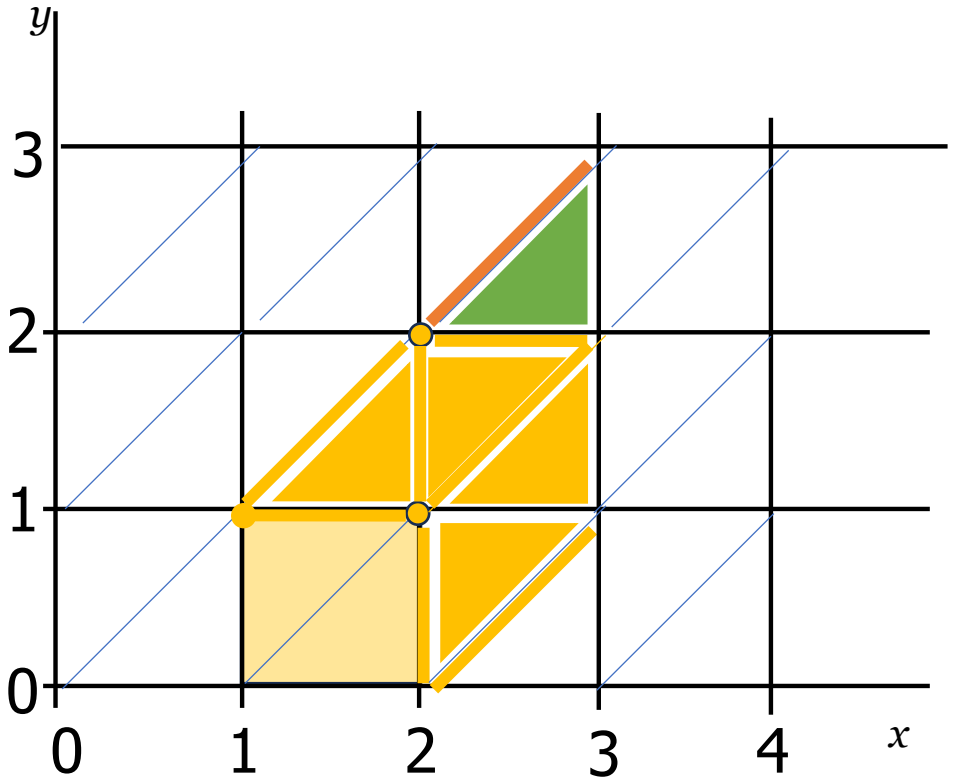


4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)

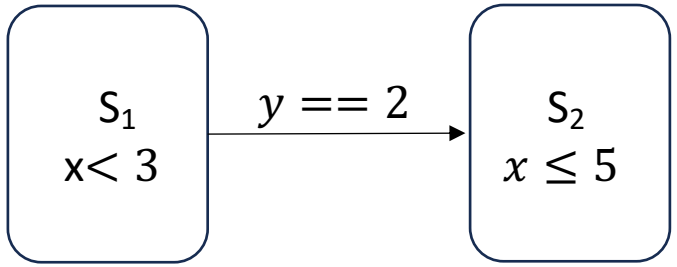


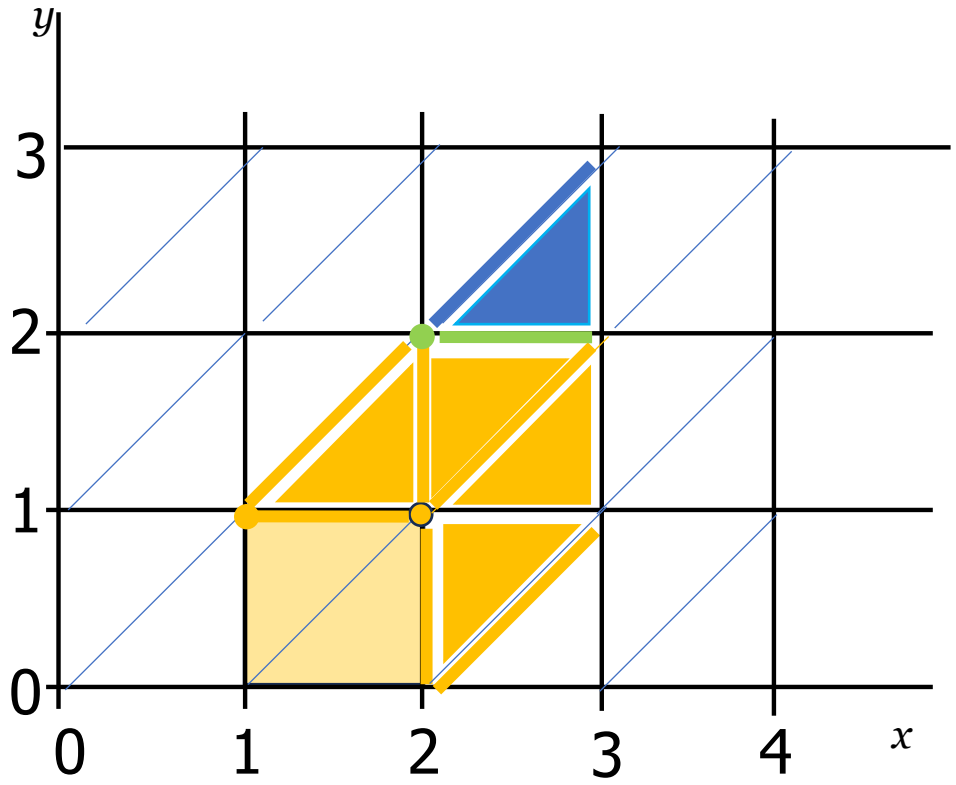
Région jaune : en S_1, $1 < x < 3$ et $0 < y \leq 2$,



4-2 Passage vers automate à états

Exemple. Soit un automate A avec X=(s_0, s_1) et H=(x,y)





Région jaune : en S_1, $1 < x < 3$ et $0 < y \leq 2$
Région verte : en S_1, $1 < x < 3$ et $y = 2$, transition franchissable
Région bleue : en S_1, $1 < x < 3$ et $2 < y < 3$, transition non franchissable, bloquant

Plan

1. Motivation d’ajout du temps
2. Modélisation du temps par des automates temporisés
3. Evolution des automates temporisés
4. Analyse des automates temporisés
5. Mise en application

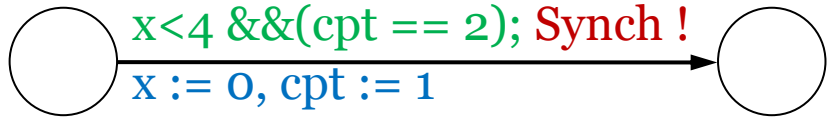
5 - Mise en application

5.1 Outil utilisé : UppAal

Caractéristiques :

- Outil pour modéliser, simuler et vérifier des systèmes en temps réel
- Modélise les systèmes comme une collection d'automates :
  - non déterministes
  - temporisés par des horloges à valeurs réelles
  - Mise à jour d'autres variables que des horloges
  - communiquant par des canaux de messages ou des variables partagées

- Automates finis + variables (horloges + entiers) + synchronisation



- A chaque transition peuvent être associés :
  - des canaux (quand on veut synchroniser des automates)
  - des gardes (= des conditions sur les valeurs des horloges ou des variables)
  - des actions de remise à jour des horloges (attention, elles ne peuvent qu'être remises à zéro) ou des variables

5.2 Application : Passage à Niveau

Soit un système de passage à niveau, afin de vérifier la sécurité de tronçon de route, nous souhaitons modéliser :

- La barrière
  - Après une demande de descente, elle atteint la position en moins de 3 secondes
  - Après une demande de levée, elle atteint la position entre 1 et 2 secondes
  - Sans demande, elle reste dans la dernière position atteinte
- Le train
  - Signale son approche au moins 2 secondes avant la section gardée
  - Parcours la distance entre l'approche et la sortie de zone gardée en pas plus de 5 secondes
- Système de pilotage
  - Quand le train signal son approche, la demande de descente est envoyée à la barrière exactement au bout d'une seconde
  - Quand le train sort, la demande de levée est envoyée en moins d'une seconde

5.2 Application : Passage à Niveau

Modélisons le comportement de la barrière sous le logiciel UppAal

Les modèles de train et de système de pilotage vous sont donnés

Via le simulateur d'UppAal, observer les régions qui sont regroupées

Références bibliographiques

Références bibliographiques

- Cassandras, C. G. and Lafortune, S. (2008). *Introduction to discrete event systems.* Springer.
- Alur, R. and Dill, D.L (1994). *A Theory of timed automata.* Theoretical Computer Science, 126(2):183–235.

Equipe pédagogique

Equipe pédagogique

Auteur.rice.s : Dimitri Lefebvre, Pascale Marangé, Olivier H. Roux

Intervenante : Pascale Marangé