IF107 - Machines à états et invariants inductifs

Frédéric Herbreteau
frederic.herbreteau@bordeaux-inp.fr

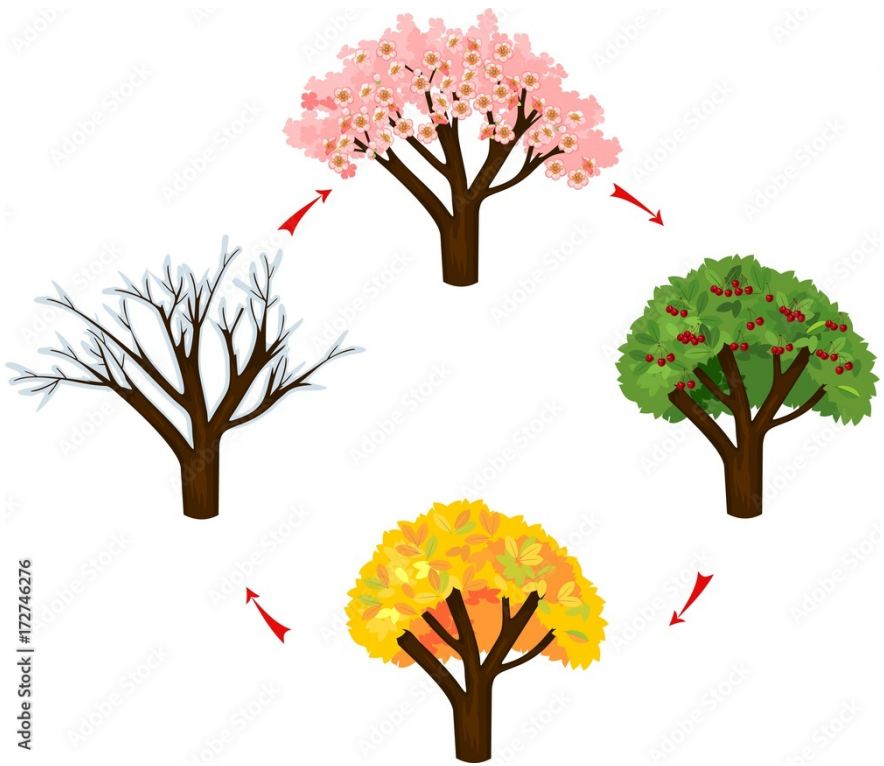
29 novembre 2024

1. Machine à états

Modélisation des systèmes

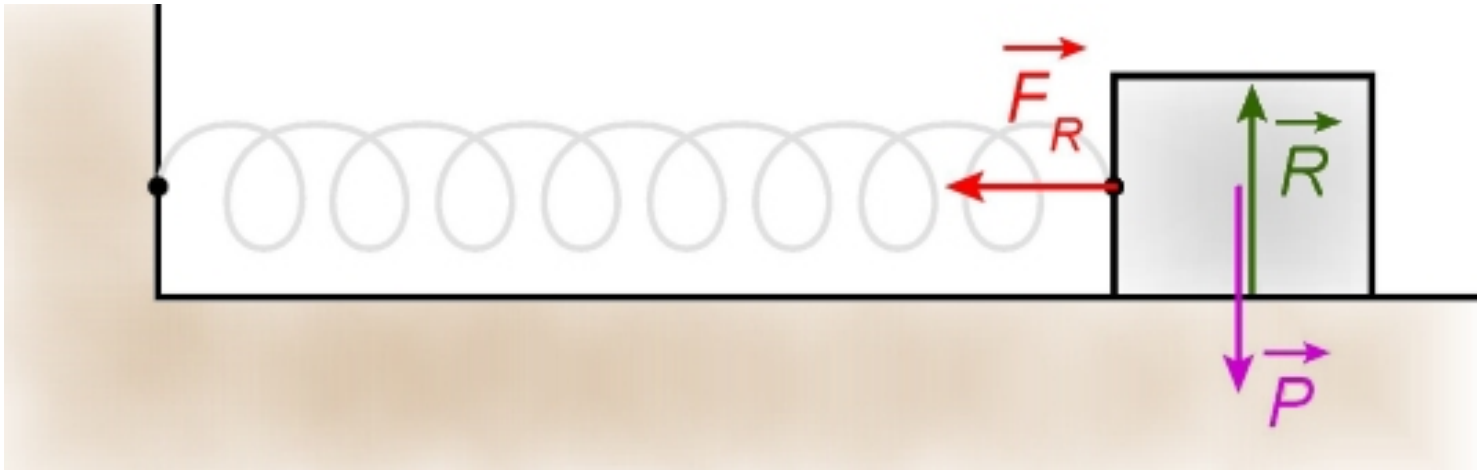
Un système est souvent modélisé par ses **états**, et ses **changements d'états**.

Exemple



credits : Kazakova Maryia

Exemple : système oscillant



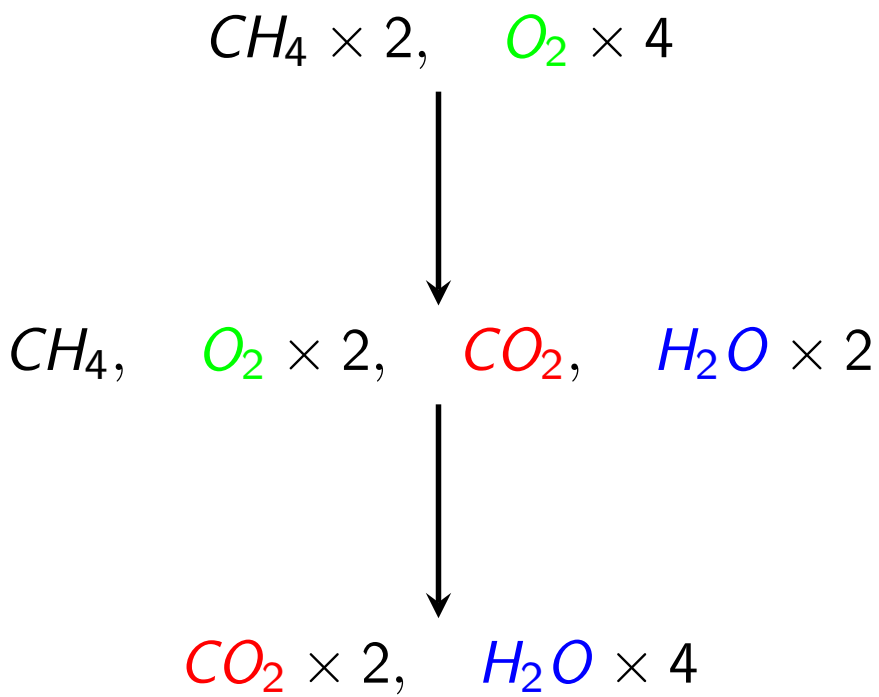
- Solide soumis à 3 forces : son poids $\vec{P}$, la réaction du support $\vec{R}$ et la force de rappel $\vec{F_R}$.
- Allongement du système oscillant :

$$x(t) = x_m \cos\left(\frac{2\pi}{T_0}t + \varphi_0\right)$$

où x_m est l'amplitude (allongement max), T_0 est la période propre de l'oscillateur et φ_0 est le décalage de phase.

Exemple : combustion du méthane

- réaction élémentaire : $CH_4 + 2O_2 \rightarrow CO_2 + 2H_2O$



Machine à états

Un processus qui **évolue étape par étape** peut être modéisé par une machine à états.

Définition

Une **machine à états** est un triplet $(S, \text{Init}, \text{Next})$ où :

- S est un ensemble non vide d'**états** (potentiellement infini)
- $\text{Init} \subseteq S$ est l'ensemble des états initiaux
- $\text{Next} \subseteq S \times S$ est la relation de transition (changement d'état)

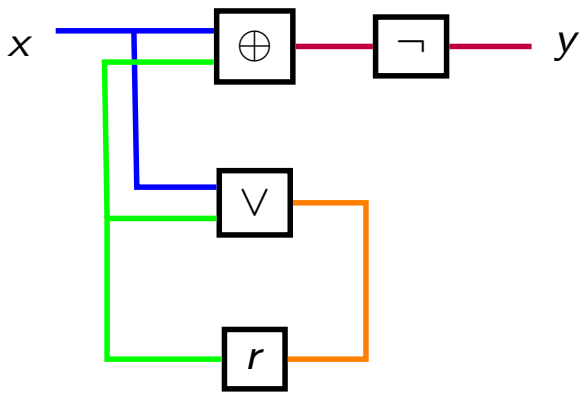
Une transition $(s_1, s_2) \in \text{Next}$ est souvent notée $s_1 \rightarrow s_2$.

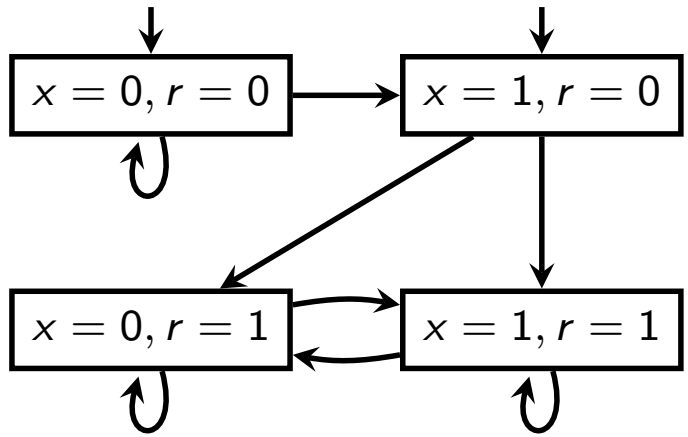
Une machine à états peut être vue comme un **graphe** dirigé, avec des sommets initiaux distingués.

On note $\text{Next}(s)$ l'ensemble des états s' tq $(s, s') \in \text{Next}$.

Exemple : machine à états d'un circuit

- une entrée : x et un registre : r
- fonction de sortie : $y = \neg(x \oplus r)$
- fonction de mise à jour du registre : $\delta_r(x, r) = (x \vee r)$



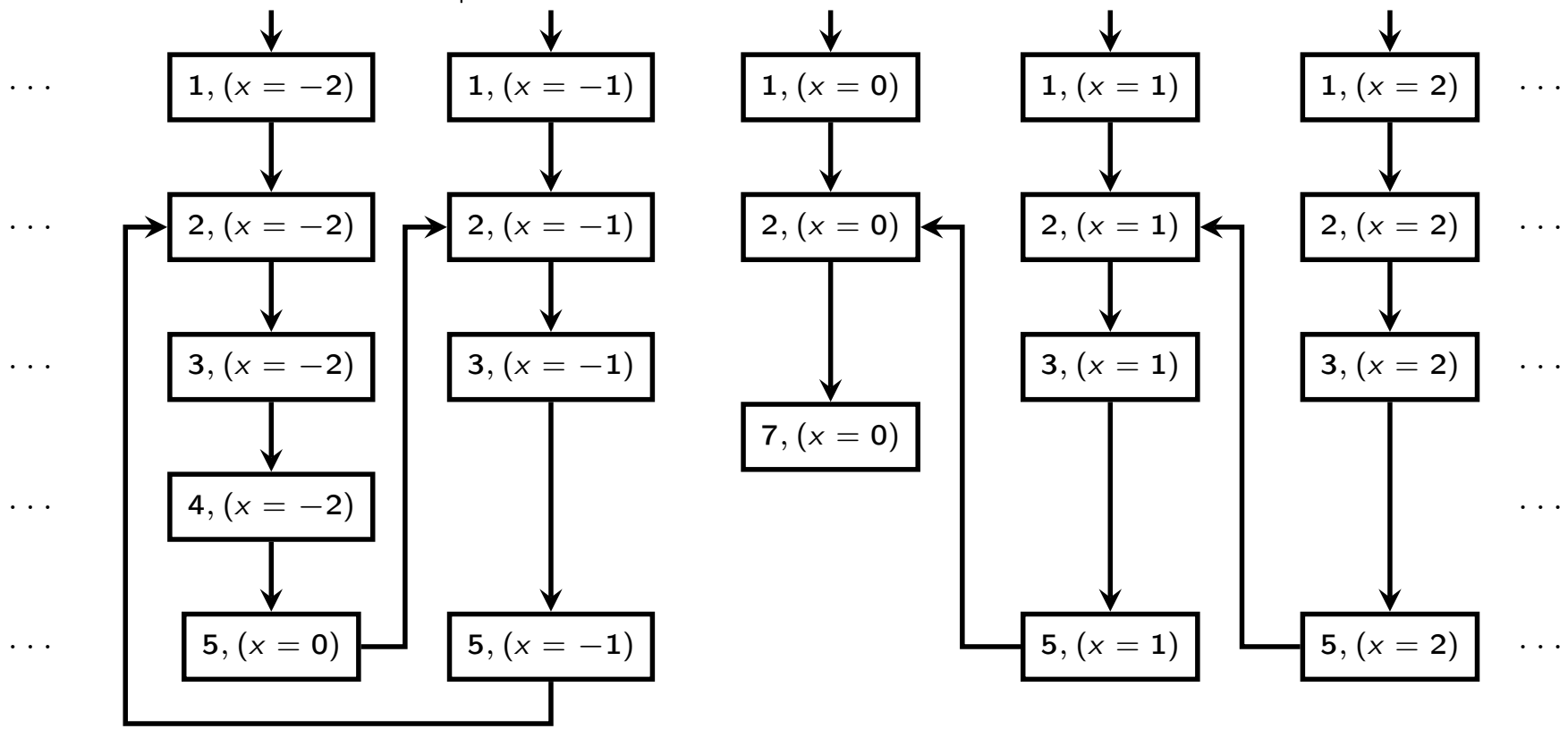


- $S = \{(x, r) \,|\, x, r \in \{0, 1\}\}$
- $\text{Init} = \{(0, 0), (1, 0)\}$
- $\text{Next} = \{(x, r) \to (x', r') \,|\, x' \in \{0, 1\} \text{ et } r' = \delta_r(x, r)\}$

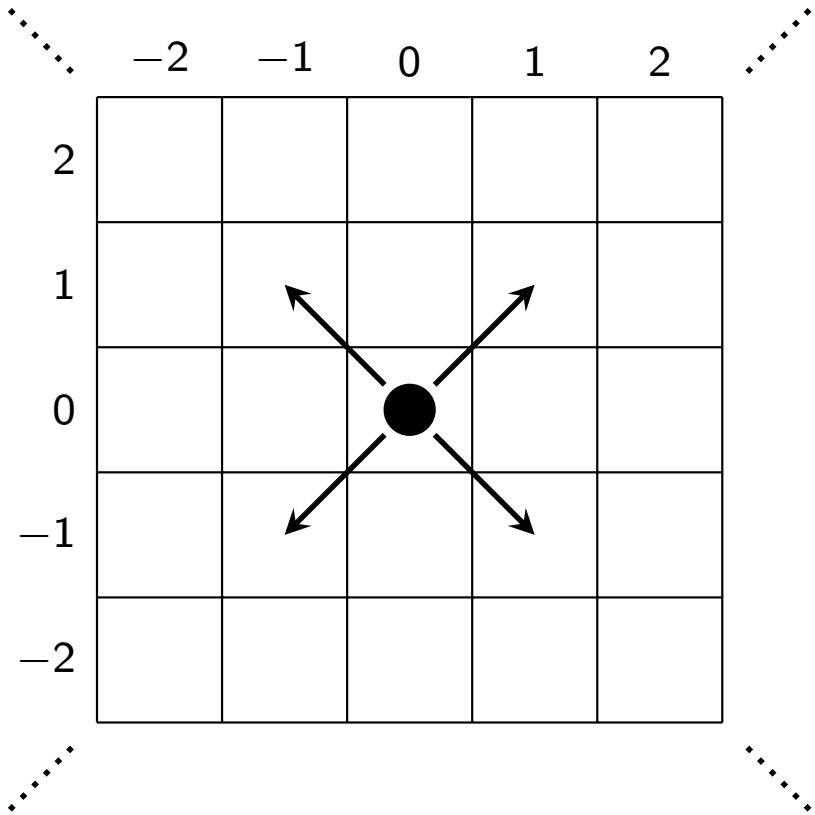
Exemple : machine à états d'un programme

```
1  void f(int x) {
2    while (x != 0) {
3      if (x < -1)
4        x = 0;
5      x = x - 1;
6    }
7  }
```

- $S = \{(pc, x) \mid pc \in [1; 7] \text{ et } x \in \texttt{int}\}$
- $\text{Init} = \{(pc, x) \in S \mid pc = 1\}$
- Next cf. ci-dessous et #15



Exemple : les mouvements d'un robot



- $S = \{(x, y) \mid x, y \in \mathbb{Z}\}$
- $\mathsf{Init} = \{(0, 0)\}$
- $\mathsf{Next} = \{(x, y) \to (x', y') \mid x' = x \pm 1 \text{ et } y' = y \pm 1\}$

Exécution

Définition

Une **exécution (ou trajectoire)** d'une machine à états $(S, \text{Init}, \text{Next})$ est une séquence d'états $s_0, s_1, s_2, \ldots$ possiblement infinie telle que $s_0 \in \text{Init}$ et $(s_i, s_{i+1}) \in \text{Next}$ pour tout i.

Exemple

- Exécution du circuit :
- Exécution du programme :
- Trajectoire du robot :

Accessibilité

Définition

Un état s d'une machine à états $(S, \text{Init}, \text{Next})$ est **accessible** s'il existe une exécution finie $s_0, s_1, \ldots, s_n$ qui aboutit en s (i.e. $s_n = s$)

Exemple

- États accessibles du circuit :

  $(x = 0, r = 0)$? $(x = 0, r = 1)$? $(x = 2, r = 7)$?

- États accessibles du programme :

  $(1, x = -2678)$? $(4, x = 0)$? $(2, x = 1)$?

- États accessibles du robot :

  $(0, 0)$? $(1, 3)$? $(1, 2)$?

2. Invariants inductifs

Invariant

On note Reach(M) l'ensemble des états accessibles d'une machine à états $M = (S, \text{Init}, \text{Next})$.

Définition

Un **invariant** d'une machine à états M est un prédicat $P(s)$ qui vrai pour tout état $s \in \text{Reach}(M)$.

Exemple

- "$x \in [-2; 2]$" est un invariant du programme C si on part d'une valeur initiale entre -2 et 2.
- "$x + y$ est pair" est un invariant du robot

Invariant inductif (Floyd, 1967)

Soit $M = (S, \mathrm{Init}, \mathrm{Next})$ une machine à états.

Définition

Un **invariant inductif** est un prédicat $P(s)$ tel que :

- $P(s)$ est vrai pour tout etat $s \in \mathrm{Init}$
- et, pour tout $(s, s') \in \mathrm{Next}$, $P(s)$ implique $P(s')$.

Théorème

Si $P(s)$ est un invariant inductif, alors $P(s)$ est un invariant

Démonstration.

□

Exemple : domaine d'une variable d'un programme

```
1  void f(int x) {
2    while (x != 0) {
3      if (x < -1)
4        x = 0;
5      x = x - 1;
6    }
7  }
```

- $S = \{(pc, x) \mid pc \in [1; 7] \text{ et } x \in \texttt{int}\}$
- $\text{Init} = \{(pc, x) \in S \mid pc = 1\}$
- Next cf. preuve ci-dessous

Exemple

On suppose que la valeur initiale de x est entre -2 et 2.

$P((pc, x)) \triangleq$ "$x \in [-2, 2]$ si $pc \neq 5$ et $x \in [-1, 2]$ si $pc = 5$"

Montrons que P est un invariant inductif.

(Base)

(Induction) On suppose que $P((pc, x))$ est vrai pour un certain état (pc, x), et considérons ses états successeurs

Alors, P est un invariant inductif, et donc un invariant.

Méthodologie

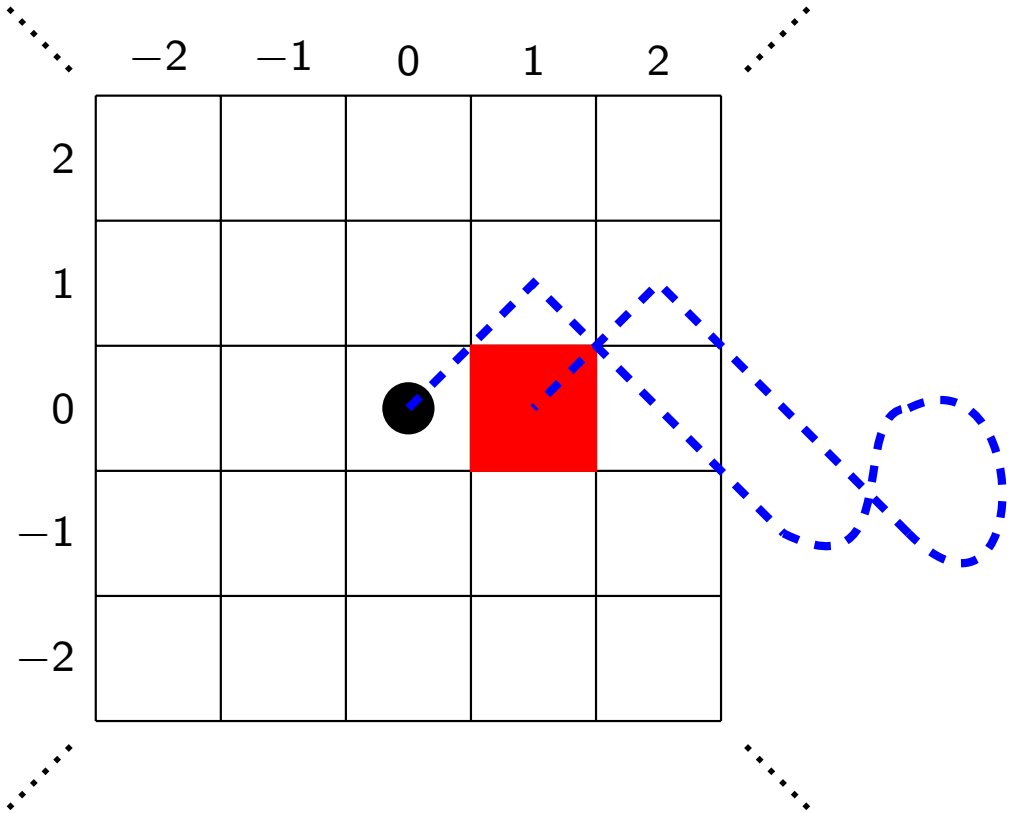
Pour une machine à états $M = (S, \text{Init}, \text{Next})$

- Pour montrer que P est un **invariant inductif** :
  1. Montrer que $P(s)$ est vrai pour tout état $s \in \text{Init}$
  2. Montrer que pour tout $(s, s') \in \text{Next}$, si $P(s)$ est vrai, alors $P(s')$ est vrai

- Pour montrer que P est un **invariant** :
  1. Trouver un prédicat Q sur les états de M
  2. Prouver que Q est un invariant inductif
  3. Montrer que $Q(s) \implies P(s)$ pour tout état accessible s

(Technique introduite par Robert Floyd en 1967)

Exemple : sûreté du robot (1/3)



Propriété de sûreté : "le robot doit éviter la position $(0, 1)$"

Exemple

$P(x, y) \triangleq$ "$(x, y) \neq (0, 1)$" est-il un invariant inductif ?

Exemple : sûreté du robot (2/3)

Machine à états du robot :

- $S = \{(x, y) \mid x, y \in \mathbb{Z}\}$
- $\mathsf{Init} = \{(0, 0)\}$
- $\mathsf{Next} = \{(x, y) \to (x', y') \mid x' = x \pm 1 \text{ et } y' = y \pm 1\}$

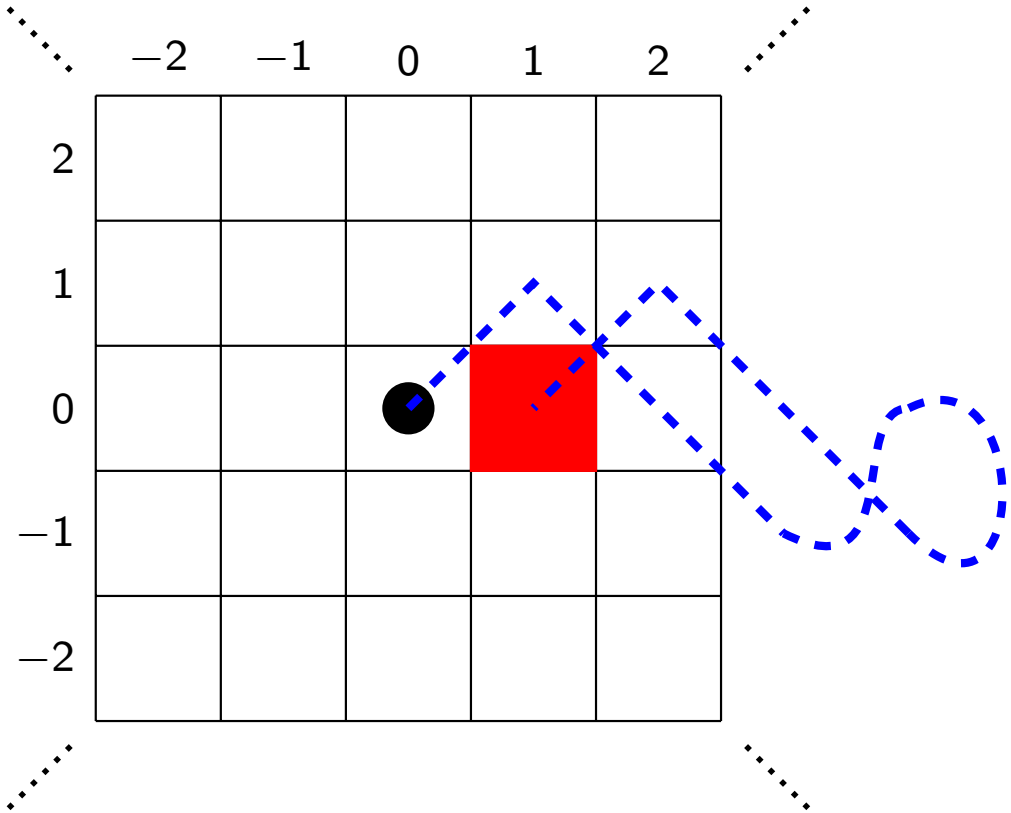
Exemple

$Q(x, y) \triangleq$ "$x + y$ est pair" est un invariant inductif

(Base) $Q(x, y)$ est vrai pour tout $(x, y) \in \mathsf{Init}$

(Induction) pour toute transition $(x, y) \to (x', y')$ si $P(x, y)$ est vrai, alors $P(x', y')$ est vrai

Exemple : sûreté du robot (3/3)



Propriété de sûreté : "le robot doit éviter la position $(0, 1)$"

Exemple

Montrer que $P(x, y) \triangleq$ "$(x, y) \neq (0, 1)$" est un invariant