Caractérisation du bassin de capture à l'aide de méthodes ensemblistes.

M. Lhommeau [1] L. Jaulin[2] L. Hardouin[1]

[1]Laboratoire d'Ingénierie des Systèmes Automatisés
ISTIA - Université d'Angers
62, av. Notre Dame du Lac, 49000 Angers, France

[2]E[3]I[2]
ENSIETA
2 rue Françoise Verny, 29806 Brest, France

GT Méthodes Ensemblistes - 19 juillet 2007

Introduction

- Considérons un système dynamique non-linéaire :

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \tag{1}$$

- où $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m$ décrit la dynamique gouvernant l'équation d'état $\mathbf{x}(t) \in \mathbb{R}^n$ en fonction d'un paramètre de contrôle $\mathbf{u}(t)$ choisi dans un ensemble $\mathbb{U} \subset \mathbb{R}^m$.

Introduction

- Considérons un système dynamique non-linéaire :

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \quad (1)$$

- où $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m$ décrit la dynamique gouvernant l'équation d'état $\mathbf{x}(t) \in \mathbb{R}^n$ en fonction d'un paramètre de contrôle $\mathbf{u}(t)$ choisi dans un ensemble $\mathbb{U} \subset \mathbb{R}^m$.

- On introduit $K \subset \mathbb{R}^n$ un ensemble de contraintes sur l'état et $\mathbb{T} \subset K$ une cible.
- On appelle *Bassin de Capture*, noté $\mathbb{C}$, le sous-ensemble des états initiaux $\mathbf{x} \in K$ tels qu'il existe une commande $\mathbf{u} \in \mathbb{U}$, telle qu'il existe une évolution $t \rightarrow \mathbf{x}(t)$ solution du système différentiel (1) et un instant $t^* \geq 0$ pour lesquels

$$\begin{cases} \mathbf{x}(t) & \in & K \quad \forall t \in [0, t^*] \\ \mathbf{x}(t^*) & \in & \mathbb{C} \end{cases}$$

Introduction

- Considérons un système dynamique non-linéaire :

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \tag{1}$$

- où $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m$ décrit la dynamique gouvernant l'équation d'état $\mathbf{x}(t) \in \mathbb{R}^n$ en fonction d'un paramètre de contrôle $\mathbf{u}(t)$ choisi dans un ensemble $\mathbb{U} \subset \mathbb{R}^m$.

- On introduit $K \subset \mathbb{R}^n$ un ensemble de contraintes sur l'état et $\mathbb{T} \subset K$ une cible.
- On appelle *Bassin de Capture*, noté $\mathbb{C}$, le sous-ensemble des états initiaux $\mathbf{x} \in K$ tels qu'il existe une commande $\mathbf{u} \in \mathbb{U}$, telle qu'il existe une évolution $t \to \mathbf{x}(t)$ solution du système différentiel (1) et un instant $t^* \geq 0$ pour lesquels

$$\begin{cases} \mathbf{x}(t) & \in & K & \forall t \in [0, t^*] \\ \mathbf{x}(t^*) & \in & \mathbb{C} \end{cases}$$

Introduction

- Considérons un système dynamique non-linéaire :

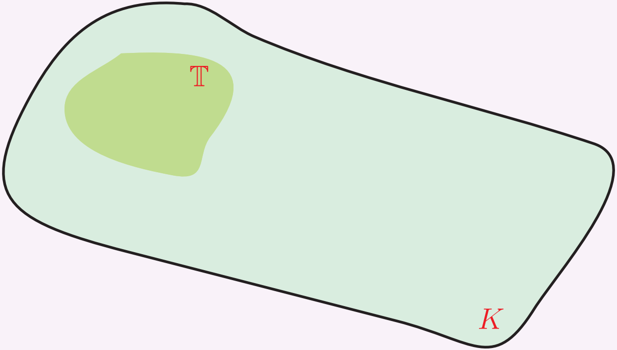
$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \quad (1)$$

- où $\mathbf{f} : \mathbb{R}^n \times \mathbb{R}^m$ décrit la dynamique gouvernant l'équation d'état $\mathbf{x}(t) \in \mathbb{R}^n$ en fonction d'un paramètre de contrôle $\mathbf{u}(t)$ choisi dans un ensemble $\mathbb{U} \subset \mathbb{R}^m$.

- On introduit $K \subset \mathbb{R}^n$ un ensemble de contraintes sur l'état et $\mathbb{T} \subset K$ une cible.
- On appelle *Bassin de Capture*, noté $\mathbb{C}$, le sous-ensemble des états initiaux $\mathbf{x} \in K$ tels qu'il existe une commande $\mathbf{u} \in \mathbb{U}$, telle qu'il existe une évolution $t \rightarrow \mathbf{x}(t)$ solution du système différentiel (1) et un instant $t^* \geq 0$ pour lesquels

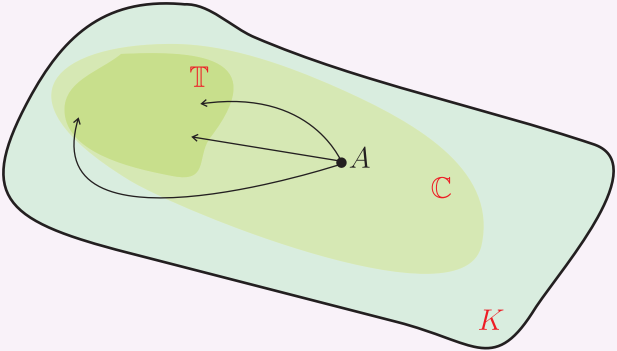
$$\begin{cases} \mathbf{x}(t) \in K & \forall t \in [0, t^*] \\ \mathbf{x}(t^*) \in \mathbb{C} \end{cases}$$

Capture Basin



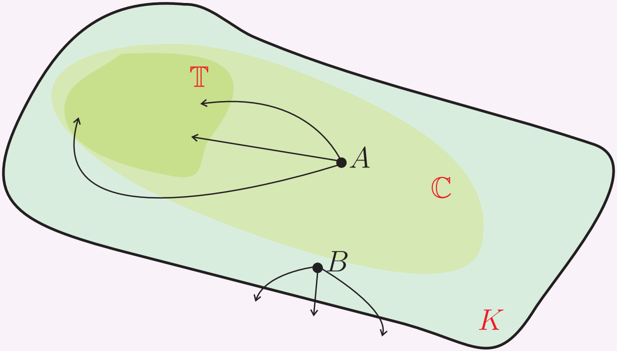
- A partir de A, au moins une évolution atteint la cible $\mathbb{C}$ sans avoir quitté K.
- A partir de B, aucune évolution n'atteint la cible $\mathbb{C}$ sans avoir quitté K.

Capture Basin



- A partir de A, au moins une évolution atteint la cible $\mathbb{C}$ sans avoir quitté K.
- A partir de B, aucune évolution n'atteint la cible $\mathbb{C}$ sans avoir quitté K.

Capture Basin



- A partir de A, au moins une évolution atteint la cible $\mathbb{C}$ sans avoir quitté K.
- A partir de B, aucune évolution n'atteint la cible $\mathbb{C}$ sans avoir quitté K.

Flot

- Pour tout $\mathbf{x}_0 \in K, \mathbf{u} \in \mathbb{U}$ et $t \in \mathbb{R}$, on pose

$$\varphi^t(\mathbf{x}_0, \mathbf{u}) = \mathbf{x}(t) \quad (2)$$

où $\mathbf{x}(t)$ est l'unique solution de $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$ vérifiant $\mathbf{x}(0) = \mathbf{x}_0$ et $\mathbf{u}(0) = \mathbf{u}$.

- La trajectoire de t_1 à t_2

$$\varphi^{[t_1,t_2]}(\mathbf{x}_0, \mathbf{u}) \overset{\text{def}}{=} \{\mathbf{x} \in \mathbb{R}^n, \exists t \in [t_1, t_2], \mathbf{x}(t) = \varphi^t(\mathbf{x}_0, \mathbf{u})\}.$$

Flot

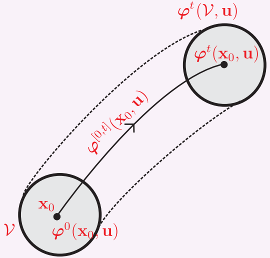
- Pour tout $\mathbf{x}_0 \in K, \mathbf{u} \in \mathbb{U}$ et $t \in \mathbb{R}$, on pose

$$\varphi^t(\mathbf{x}_0, \mathbf{u}) \quad = \quad \mathbf{x}(t) \tag{2}$$

où $\mathbf{x}(t)$ est l'unique solution de $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$ vérifiant $\mathbf{x}(0) = \mathbf{x}_0$ et $\mathbf{u}(0) = \mathbf{u}$.

- La trajectoire de t_1 à t_2

$$\varphi^{[t_1,t_2]}(\mathbf{x}_0, \mathbf{u}) \stackrel{\text{def}}{=} \left\{\mathbf{x} \in \mathbb{R}^n, \exists t \in [t_1, t_2], \mathbf{x}(t) = \varphi^t(\mathbf{x}_0, \mathbf{u})\right\}.$$



Objectif

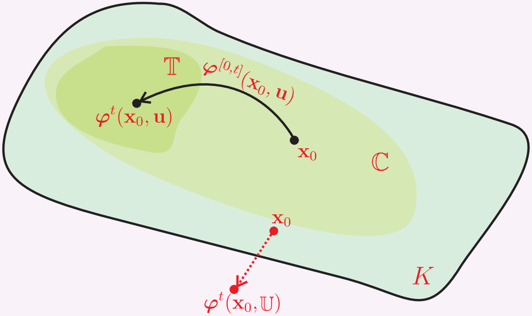
- On cherche deux ensembles $\mathbb{C}^-$ et $\mathbb{C}^+$ tels que

$$\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+,$$

où

$$\mathbb{C} \overset{\text{def}}{=} \{\mathbf{x}_0 \in K, \exists t \geq 0, \exists \mathbf{u} \in \mathcal{F}(\,[0,t] \rightarrow \mathbb{U}), \\ \varphi^t(\mathbf{x}_0, \mathbf{u}) \in \mathbb{T} \text{ et } \varphi^{[0,t]}(\mathbf{x}_0, \mathbf{u}) \subset \mathbb{K}\},$$

avec $\mathcal{F}\,([0,t] \rightarrow \mathbb{U})$ l'ensemble des fonctions continues par morceaux sur $[0,t]$.



Intervalles

- Définition

$$[x] = [\underline{x}, \overline{x}] = \{x \in \mathbb{R} \mid \underline{x} \leq x \leq \overline{x}\}$$

- Les intervalles ont une double nature :
  - *Ensembles* $\Rightarrow$ Opérations ensemblistes
  - *Paires de nombres réels* $\Rightarrow$ On peut construire une arithmétique
- Opérations sur les intervalles

$$\begin{aligned}
[x] + [y] &= [\underline{x} + \underline{y}, \overline{x} + \overline{y}] \\
[x] - [y] &= [\underline{x} - \overline{y}, \overline{x} + \underline{y}] \\
[x] \times [y] &= [\min(\underline{x}, \underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}), \max(\underline{x}, \underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y})]
\end{aligned}$$

- Il existe des équivalents intervalles pour toutes les fonctions mathématiques usuelles :

$$[\exp]([x]) = [\exp(\underline{x}), \exp(\overline{x})]$$

Intervalles

- Définition

$$[x] \quad = \quad [\underline{x}, \overline{x}] = \{x \in \mathbb{R} \mid \underline{x} \leq x \leq \overline{x}\}$$

  - Les intervalles ont une double nature :
    - *Ensembles* $\Rightarrow$ Opérations ensemblistes
    - *Paires de nombres réels* $\Rightarrow$ On peut construire une arithmétique

- Opérations sur les intervalles

$$\begin{aligned} [x] + [y] &= [\underline{x} + \underline{y}, \overline{x} + \overline{y}] \\ [x] - [y] &= [\underline{x} - \overline{y}, \overline{x} + \underline{y}] \\ [x] \times [y] &= \left[\min\left(\underline{x}, \underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}\right), \max\left(\underline{x}, \underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}\right)\right] \end{aligned}$$

- Il existe des équivalents intervalles pour toutes les fonctions mathématiques usuelles :

$$[\exp]([x]) \quad = \quad [\exp(\underline{x}), \exp(\overline{x})]$$

Intervalles

- Définition

$$[x] \quad = \quad [\underline{x}, \overline{x}] = \{x \in \mathbb{R} \mid \underline{x} \leq x \leq \overline{x}\}$$

  - Les intervalles ont une double nature :
    - *Ensembles* $\Rightarrow$ Opérations ensemblistes
    - *Paires de nombres réels* $\Rightarrow$ On peut construire une arithmétique

- Opérations sur les intervalles

$$\begin{aligned} [x] + [y] &= \left[\underline{x} + \underline{y}, \overline{x} + \overline{y}\right] \\ [x] - [y] &= \left[\underline{x} - \overline{y}, \overline{x} + \underline{y}\right] \\ [x] \times [y] &= \left[\min\left(\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}\right), \max\left(\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}\right)\right] \end{aligned}$$

- Il existe des équivalents intervalles pour toutes les fonctions mathématiques usuelles :

$$[\exp]([x]) \quad = \quad [\exp(\underline{x}), \exp(\overline{x})]$$

Intervalles

- Définition

$$[x] = [\underline{x}, \overline{x}] = \{x \in \mathbb{R} \mid \underline{x} \leq x \leq \overline{x}\}$$

- Les intervalles ont une double nature :
  - *Ensembles* ⇒ Opérations ensemblistes
  - *Paires de nombres réels* ⇒ On peut construire une arithmétique

- Opérations sur les intervalles

$$\begin{aligned} [x] + [y] &= [\underline{x} + \underline{y}, \overline{x} + \overline{y}] \\ [x] - [y] &= [\underline{x} - \overline{y}, \overline{x} + \underline{y}] \\ [x] \times [y] &= [\min(\underline{x}, \underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{xy}), \max(\underline{x}, \underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{xy})] \end{aligned}$$

- Il existe des équivalents intervalles pour toutes les fonctions mathématiques usuelles :

$$[\exp]([x]) = [\exp(\underline{x}), \exp(\overline{x})]$$

Intervalles

- Définition

$$[x] \quad = \quad [\underline{x}, \overline{x}] = \{x \in \mathbb{R} \mid \underline{x} \leq x \leq \overline{x}\}$$

  - Les intervalles ont une double nature :
    - *Ensembles* $\Rightarrow$ Opérations ensemblistes
    - *Paires de nombres réels* $\Rightarrow$ On peut construire une arithmétique

- Opérations sur les intervalles

$$\begin{aligned}
[x] + [y] &= \left[\underline{x} + \underline{y}, \overline{x} + \overline{y}\right] \\
[x] - [y] &= \left[\underline{x} - \overline{y}, \overline{x} + \underline{y}\right] \\
[x] \times [y] &= \left[\min\left(\underline{x}, \underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{xy}\right), \max\left(\underline{x}, \underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{xy}\right)\right]
\end{aligned}$$

- Il existe des équivalents intervalles pour toutes les fonctions mathématiques usuelles :

$$[\exp]\left([x]\right) \quad = \quad [\exp(\underline{x}), \exp(\overline{x})]$$

Interval vectors

Définition

- Un pavé $[\mathbf{x}]$ de $\mathbb{R}^n$ est le produit cartésien de n intervalles

$$[\mathbf{x}] = [x_1] \times [x_2] \times \ldots \times [x_n] = ([x_1], [x_2], \ldots, [x_n])^{\mathsf{T}}$$

- Longueur

$$w([\mathbf{x}]) = \max_{1 \leq i \leq n} w([x_i])$$

- L'ensemble de tous les pavés de $\mathbb{R}^n$ est noté $\mathbb{IR}^n$.

Example

- Un pavé $[\mathbf{x}] = [x_1] \times [x_2]$ de $\mathbb{IR}^2$

Interval vectors

Définition

- Un pavé $[\mathbf{x}]$ de $\mathbb{R}^n$ est le produit cartésien de n intervalles

$$[\mathbf{x}] = [x_1] \times [x_2] \times \ldots \times [x_n] = ([x_1], [x_2], \ldots, [x_n])^\mathsf{T}$$

- Longueur

$$w([\mathbf{x}]) = \max_{1 \leq i \leq n} w([x_i])$$

- L'ensemble de tous les pavés de $\mathbb{R}^n$ est noté $\mathbb{IR}^n$.

Example

- Un pavé $[\mathbf{x}] = [x_1] \times [x_2]$ de $\mathbb{IR}^2$

Interval vectors

Définition

- Un pavé $[\mathbf{x}]$ de $\mathbb{R}^n$ est le produit cartésien de n intervalles

$$[\mathbf{x}] = [x_1] \times [x_2] \times \ldots \times [x_n] = ([x_1], [x_2], \ldots, [x_n])^{\mathsf{T}}$$

- Longueur

$$w([\mathbf{x}]) = \max_{1 \leq i \leq n} w([x_i])$$

- L'ensemble de tous les pavés de $\mathbb{R}^n$ est noté $\mathbb{IR}^n$.

Example

- Un pavé $[\mathbf{x}] = [x_1] \times [x_2]$ de $\mathbb{IR}^2$

Interval vectors

Définition

- Un pavé $[\mathbf{x}]$ de $\mathbb{R}^n$ est le produit cartésien de n intervalles

$$[\mathbf{x}] = [x_1] \times [x_2] \times \ldots \times [x_n] = ([x_1], [x_2], \ldots, [x_n])^{\mathsf{T}}$$

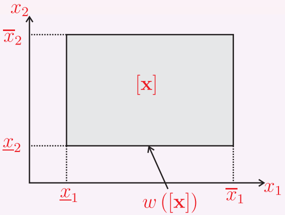
- Longueur

$$w([\mathbf{x}]) = \max_{1 \leq i \leq n} w([x_i])$$

- L'ensemble de tous les pavés de $\mathbb{R}^n$ est noté $\mathbb{IR}^n$.

Example

- Un pavé $[\mathbf{x}] = [x_1] \times [x_2]$ de $\mathbb{IR}^2$



- Les opérations classiques sur les intervalles s'étendent aux pavés

$$\begin{aligned}
\alpha\,[\mathbf{x}] &= (\alpha\,[x_1]) \times \ldots \times (\alpha\,[x_n]) \\
[\mathbf{x}]^{\mathsf{T}} . [\mathbf{y}] &= [x_1] . [y_1] + \ldots + [x_n] . [y_n] \\
[\mathbf{x}] + [\mathbf{y}] &= ([x_1] + [y_1]) \times \ldots \times ([x_n] + [y_n])
\end{aligned}$$

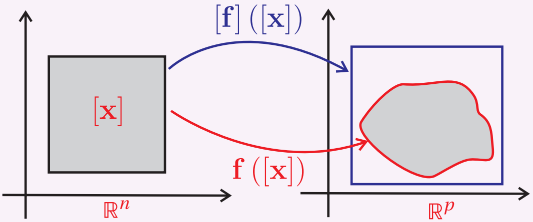
et aux matrices d'intervalles.

Fonction d'inclusion

Definition

- $[\mathbf{f}]$ est une fonction d'inclusion pour $\mathbf{f}$ si

$$\forall [\mathbf{x}] \in \mathbb{IR}^n \quad , \quad \mathbf{f}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}]).$$



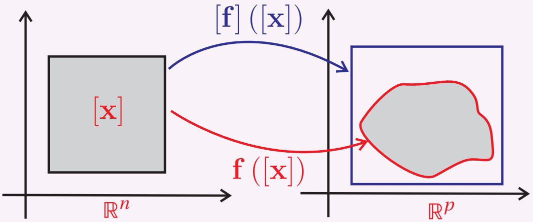
- $\mathbf{f}$ peut être définie par un algorithme ou même une équation différentielle

Fonction d'inclusion

Definition

- $[\mathbf{f}]$ est une fonction d'inclusion pour $\mathbf{f}$ si

$$\forall [\mathbf{x}] \in \mathbb{IR}^n \quad , \quad \mathbf{f}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}]).$$



- $\mathbf{f}$ peut être définie par un algorithme ou même une équation différentielle

Sous-pavages

Les intervalles et les pavés ne sont, en général, pas suffisants pour décrire tous les ensembles qui peuvent nous intéresser

$\Downarrow$

Introduction des sous-pavages

Sous-pavages de $\mathbb{R}^n$ = union de pavés de $\mathbb{IR}^n$ qui ne se recouvrent pas

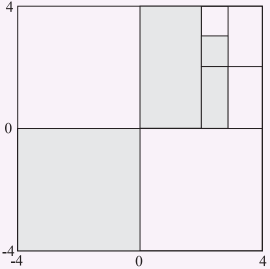
Sous-pavages

Les intervalles et les pavés ne sont, en général, pas suffisants pour décrire tous les ensembles qui peuvent nous intéresser

$\Downarrow$

Introduction des sous-pavages

Sous-pavages de $\mathbb{R}^n$ = union de pavés de $\mathbb{IR}^n$ qui ne se recouvrent pas

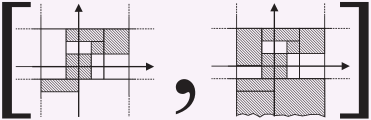


Approximations intérieures et extérieures

- Si les sous-pavages $\underline{\mathbb{S}}$ et $\overline{\mathbb{S}}$ sont tels que

$$\underline{\mathbb{S}} \subset \mathbb{S} \subset \overline{\mathbb{S}}$$

alors l'ensemble $\mathbb{S}$ est encadré par une approximation intérieure et extérieure.



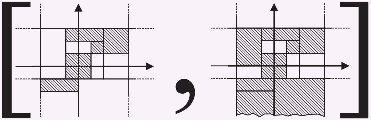
- La distance entre $\underline{\mathbb{S}}$ et $\overline{\mathbb{S}}$ donne une indication sur la qualité de l'approximation de $\mathbb{S}$
- Calcul sur le sous-pavage
  - permet de donner une approximation d'ensembles compacts
  - ingrédients de base de l'algorithme du bassin de capture présenté dans la suite

Approximations intérieures et extérieures

- Si les sous-pavages $\underline{\mathbb{S}}$ et $\overline{\mathbb{S}}$ sont tels que

$$\underline{\mathbb{S}} \subset \mathbb{S} \subset \overline{\mathbb{S}}$$

  alors l'ensemble $\mathbb{S}$ est encadré par une approximation intérieure et extérieure.



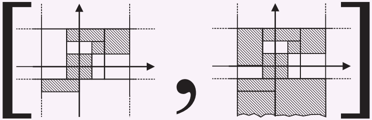
- La distance entre $\underline{\mathbb{S}}$ et $\overline{\mathbb{S}}$ donne une indication sur la qualité de l'approximation de $\mathbb{S}$
- Calcul sur le sous-pavage
  - permet de donner une approximation d'ensembles compacts
  - ingrédients de base de l'algorithme du bassin de capture présenté dans la suite

Approximations intérieures et extérieures

- Si les sous-pavages $\underline{\mathbb{S}}$ et $\overline{\mathbb{S}}$ sont tels que

$$\underline{\mathbb{S}} \subset \mathbb{S} \subset \overline{\mathbb{S}}$$

alors l'ensemble $\mathbb{S}$ est encadré par une approximation intérieure et extérieure.



- La distance entre $\underline{\mathbb{S}}$ et $\overline{\mathbb{S}}$ donne une indication sur la qualité de l'approximation de $\mathbb{S}$
- Calcul sur le sous-pavage
  - permet de donner une approximation d'ensembles compacts
  - ingrédients de base de l'algorithme du bassin de capture présenté dans la suite

Pour l'analyse par intervalle

L'analyse par intervalle permet de d'obtenir des résultats garantis

$$\Downarrow$$

C'est un avantage considérable par rapport aux autres méthodes.

Fonction d'inclusion du Flot

Soit

$$[\varphi] : \begin{cases} \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m & \to & \mathbb{R}^n \\ ([t], [\mathbf{x}], [\mathbf{u}]) & \mapsto & [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]) \end{cases}$$

une fonction d'inclusion pour le flot telle que

$$\forall t \in [t], \forall \mathbf{x} \in [\mathbf{x}], \mathbf{u} \in \mathcal{F}([t] \to [\mathbf{u}]), \varphi^t(\mathbf{x}, \mathbf{u}) \in [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]) .$$

Intégration numérique garantie - Théorème de Picard

- Soient
  - t_1 et t_2 deux nombres réels (avec $t_1 \leq t_2$)
  - $\mathbf{x}(t_1) \in [\mathbf{x}](t_1)$
  - $[\tilde{\mathbf{x}}]$ un pavé de $\mathbb{R}^n$
- Si

$$[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\tilde{\mathbf{x}}]) \subset [\tilde{\mathbf{x}}],$$

alors pour tout $t \in [t_1, t_2]$, on a

Fonction d'inclusion du Flot

Soit

$$[\varphi] : \begin{cases} \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m & \rightarrow & \mathbb{R}^n \\ ([t], [\mathbf{x}], [\mathbf{u}]) & \mapsto & [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]) \end{cases}$$

une fonction d'inclusion pour le flot telle que

$$\forall t \in [t], \forall \mathbf{x} \in [\mathbf{x}], \mathbf{u} \in \mathcal{F}([t] \rightarrow [\mathbf{u}]), \varphi^t(\mathbf{x}, \mathbf{u}) \in [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]) \, .$$

Intégration numérique garantie - Théorème de Picard

- Soient
  - t_1 et t_2 deux nombres réels (avec $t_1 \leq t_2$)
  - $\mathbf{x}(t_1) \in [\mathbf{x}](t_1)$
  - $[\tilde{\mathbf{x}}]$ un pavé de $\mathbb{R}^n$
- Si

$$[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\tilde{\mathbf{x}}]) \subset [\tilde{\mathbf{x}}],$$

alors pour tout $t \in [t_1, t_2]$, on a

$$\Rightarrow \mathbf{x}(t) \in [\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\tilde{\mathbf{x}}]),$$

$$\Rightarrow \mathbf{x}(t_2) \in [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\tilde{\mathbf{x}}]).$$

Fonction d'inclusion du Flot

Soit

$$[\varphi] : \begin{cases} \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m & \rightarrow \quad \mathbb{R}^n \\ ([t], [\mathbf{x}], [\mathbf{u}]) & \mapsto \quad [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]) \end{cases}$$

une fonction d'inclusion pour le flot telle que

$$\forall t \in [t], \forall \mathbf{x} \in [\mathbf{x}], \mathbf{u} \in \mathcal{F}([t] \rightarrow [\mathbf{u}]), \varphi^t(\mathbf{x}, \mathbf{u}) \in [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]).$$

Intégration numérique garantie - Théorème de Picard

- Soient
  - t_1 et t_2 deux nombres réels (avec $t_1 \leq t_2$)
  - $\mathbf{x}(t_1) \in [\mathbf{x}](t_1)$
  - $[\tilde{\mathbf{x}}]$ un pavé de $\mathbb{R}^n$
- Si

$$[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\tilde{\mathbf{x}}]) \subset [\tilde{\mathbf{x}}],$$

alors pour tout $t \in [t_1, t_2]$, on a

$$\Rightarrow \mathbf{x}(t) \in [\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\tilde{\mathbf{x}}]),$$

$$\Rightarrow \mathbf{x}(t_2) \in [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\tilde{\mathbf{x}}]).$$

Fonction d'inclusion du Flot

Soit

$$[\varphi] : \begin{cases} \mathbb{IR} \times \mathbb{IR}^n \times \mathbb{IR}^m & \rightarrow & \mathbb{IR}^n \\ ([t], [\mathbf{x}], [\mathbf{u}]) & \mapsto & [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]) \end{cases}$$

une fonction d'inclusion pour le flot telle que

$$\forall t \in [t], \forall \mathbf{x} \in [\mathbf{x}], \mathbf{u} \in \mathcal{F}([t] \rightarrow [\mathbf{u}]), \varphi^t(\mathbf{x}, \mathbf{u}) \in [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]).$$

Intégration numérique garantie - Théorème de Picard

- Soient
  - t_1 et t_2 deux nombres réels (avec $t_1 \leq t_2$)
  - $\mathbf{x}(t_1) \in [\mathbf{x}](t_1)$
  - $[\tilde{\mathbf{x}}]$ un pavé de $\mathbb{IR}^n$
- Si

$$[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\tilde{\mathbf{x}}]) \subset [\tilde{\mathbf{x}}],$$

alors pour tout $t \in [t_1, t_2]$, on a

$\Rightarrow$ $\mathbf{x}(t) \in [\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\tilde{\mathbf{x}}])$,

$\Rightarrow$ $\mathbf{x}(t_2) \in [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\tilde{\mathbf{x}}])$.

Fonction d'inclusion du Flot

Soit

$$[\varphi] : \begin{cases} \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m & \rightarrow & \mathbb{R}^n \\ ([t], [\mathbf{x}], [\mathbf{u}]) & \mapsto & [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]) \end{cases}$$

une fonction d'inclusion pour le flot telle que

$$\forall t \in [t], \forall \mathbf{x} \in [\mathbf{x}], \mathbf{u} \in \mathcal{F}([t] \rightarrow [\mathbf{u}]), \varphi^t(\mathbf{x}, \mathbf{u}) \in [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]).$$

Intégration numérique garantie - Théorème de Picard

- Soient
  - t_1 et t_2 deux nombres réels (avec $t_1 \leq t_2$)
  - $\mathbf{x}(t_1) \in [\mathbf{x}](t_1)$
  - $[\tilde{\mathbf{x}}]$ un pavé de $\mathbb{R}^n$
- Si

$$[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\tilde{\mathbf{x}}]) \subset [\tilde{\mathbf{x}}],$$

alors pour tout $t \in [t_1, t_2]$, on a

$\Rightarrow$ $\mathbf{x}(t) \in [\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\tilde{\mathbf{x}}])$,

$\Rightarrow$ $\mathbf{x}(t_2) \in [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\tilde{\mathbf{x}}])$.

Fonction d'inclusion du Flot

Soit

$$[\varphi] : \begin{cases} \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m & \rightarrow & \mathbb{R}^n \\ ([t], [\mathbf{x}], [\mathbf{u}]) & \mapsto & [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]) \end{cases}$$

une fonction d'inclusion pour le flot telle que

$$\forall t \in [t], \forall \mathbf{x} \in [\mathbf{x}], \mathbf{u} \in \mathcal{F}([t] \rightarrow [\mathbf{u}]), \varphi^t(\mathbf{x}, \mathbf{u}) \in [\varphi]^{[t]}([\mathbf{x}], [\mathbf{u}]) .$$

Intégration numérique garantie - Théorème de Picard

- Soient
  - t_1 et t_2 deux nombres réels (avec $t_1 \leq t_2$)
  - $\mathbf{x}(t_1) \in [\mathbf{x}](t_1)$
  - $[\tilde{\mathbf{x}}]$ un pavé de $\mathbb{R}^n$
- Si

$$[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\tilde{\mathbf{x}}]) \subset [\tilde{\mathbf{x}}],$$

alors pour tout $t \in [t_1, t_2]$, on a

$\Rightarrow$ $\mathbf{x}(t) \in [\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\tilde{\mathbf{x}}])$,

$\Rightarrow$ $\mathbf{x}(t_2) \in [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\tilde{\mathbf{x}}])$.

Algorithme $[\varphi]$

$[\varphi]$ (in : $[t] = [t_1, t_2], [\mathbf{x}], [\mathbf{u}]$, out :$[\mathbf{w}], [\mathbf{x}]([t_2]))$

1 $[\hat{\mathbf{x}}](t_2) := [\mathbf{x}] + (t_2 - t_1) * [\mathbf{f}]([\mathbf{x}](t_1), [\mathbf{u}]);$

2 $[\mathbf{v}] := [\mathbf{x}](t_1) \sqcup [\hat{\mathbf{x}}](t_2);$

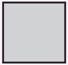
3 $[\mathbf{w}] :=$ inflate$([\mathbf{v}], \alpha.w([\mathbf{v}]) + \beta);$

if $[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\mathbf{w}]), [\mathbf{u}]) \not\subseteq [\mathbf{w}]$

$\{\ [\mathbf{w}] := \mathbb{R}^n;$ return$\}$;

5 $[\mathbf{x}](t_2) := [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\mathbf{w}], [\mathbf{u}]).$

Finalement $[\mathbf{x}](t_2) = [\varphi]^{t_2}([\mathbf{x}](t_1), \mathbf{u})$ et $[\mathbf{w}] = [\varphi]^{[t_1,t_2]}([\mathbf{x}](t_1), \mathbf{u})$



$[\mathbf{x}](t_1)$

Algorithme $[\varphi]$

$[\varphi]$ (in : $[t] = [t_1, t_2], [\mathbf{x}], [\mathbf{u}]$, out :$[\mathbf{w}], [\mathbf{x}]([t_2])$)

1 $[\hat{\mathbf{x}}](t_2) := [\mathbf{x}] + (t_2 - t_1) * [\mathbf{f}]([\mathbf{x}](t_1), [\mathbf{u}]);$

2 $[\mathbf{v}] := [\mathbf{x}](t_1) \sqcup [\hat{\mathbf{x}}](t_2);$

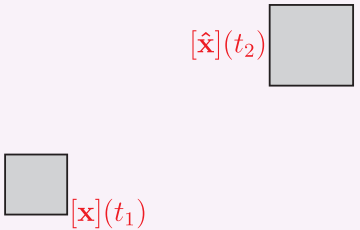
3 $[\mathbf{w}] :=$ inflate$([\mathbf{v}],\alpha.w([\mathbf{v}]) + \beta);$

if $[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\mathbf{w}]), [\mathbf{u}]) \not\subseteq [\mathbf{w}]$

{ $[\mathbf{w}] := \mathbb{R}^n$; return} ;

5 $[\mathbf{x}](t_2) := [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\mathbf{w}], [\mathbf{u}]).$

Finalement $[\mathbf{x}](t_2) = [\varphi]^{t_2}([\mathbf{x}](t_1), \mathbf{u})$ et $[\mathbf{w}] = [\varphi]^{[t_1,t_2]}([\mathbf{x}](t_1), \mathbf{u})$



Algorithme $[\varphi]$

$[\varphi]$ (in : $[t] = [t_1, t_2], [\mathbf{x}], [\mathbf{u}]$, out :$[\mathbf{w}], [\mathbf{x}]([t_2]))$

1 $\quad [\hat{\mathbf{x}}](t_2) := [\mathbf{x}] + (t_2 - t_1) * [\mathbf{f}]([\mathbf{x}](t_1), [\mathbf{u}]);$

2 $\quad [\mathbf{v}] := [\mathbf{x}](t_1) \sqcup [\hat{\mathbf{x}}](t_2);$

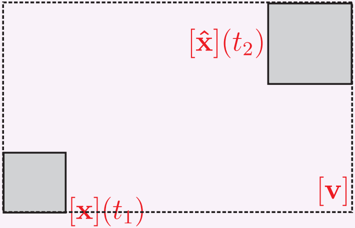
3 $\quad [\mathbf{w}] :=$ inflate$([\mathbf{v}], \alpha.w([\mathbf{v}]) + \beta);$

$\quad$ if $[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\mathbf{w}]), [\mathbf{u}]) \nsubseteq [\mathbf{w}]$

$\quad \{\ [\mathbf{w}] := \mathbb{R}^n;$ return$\}$;

5 $\quad [\mathbf{x}](t_2) := [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\mathbf{w}], [\mathbf{u}]).$

Finalement $[\mathbf{x}](t_2) = [\varphi]^{t_2}([\mathbf{x}](t_1), \mathbf{u})$ et $[\mathbf{w}] = [\varphi]^{[t_1, t_2]}([\mathbf{x}](t_1), \mathbf{u})$



Algorithme $[\varphi]$

$[\varphi]$ (in : $[t] = [t_1, t_2], [\mathbf{x}], [\mathbf{u}]$, out :$[\mathbf{w}], [\mathbf{x}]([t_2]))$

$[\hat{\mathbf{x}}](t_2) := [\mathbf{x}] + (t_2 - t_1) * [\mathbf{f}]([\mathbf{x}](t_1), [\mathbf{u}]);$

$[\mathbf{v}] := [\mathbf{x}](t_1) \sqcup [\hat{\mathbf{x}}](t_2);$

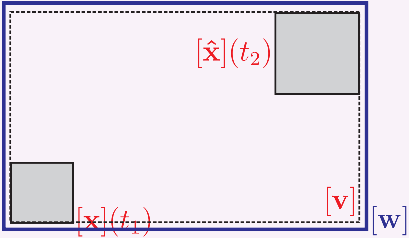
$[\mathbf{w}] :=$ inflate$([\mathbf{v}], \alpha.w([\mathbf{v}]) + \beta);$

if $[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\mathbf{w}]), [\mathbf{u}]) \not\subseteq [\mathbf{w}]$

{ $[\mathbf{w}] := \mathbb{R}^n$; return} ;

$[\mathbf{x}](t_2) := [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\mathbf{w}], [\mathbf{u}]).$

Finalement $[\mathbf{x}](t_2) = [\varphi]^{t_2}([\mathbf{x}](t_1), \mathbf{u})$ et $[\mathbf{w}] = [\varphi]^{[t_1, t_2]}([\mathbf{x}](t_1), \mathbf{u})$



Algorithme $[\varphi]$

$[\varphi]$ (in : $[t] = [t_1, t_2], [\mathbf{x}], [\mathbf{u}]$, out : $[\mathbf{w}], [\mathbf{x}]([t_2])$)

$[\hat{\mathbf{x}}](t_2) := [\mathbf{x}] + (t_2 - t_1) * [\mathbf{f}]([\mathbf{x}](t_1), [\mathbf{u}]);$

$[\mathbf{v}] := [\mathbf{x}](t_1) \sqcup [\hat{\mathbf{x}}](t_2);$

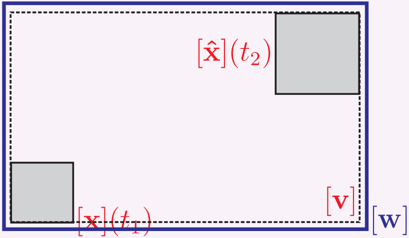
$[\mathbf{w}] :=$ inflate($[\mathbf{v}]$,$\alpha.w([\mathbf{v}]) + \beta$);

if $[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\mathbf{w}]), [\mathbf{u}]) \not\subseteq [\mathbf{w}]$

{ $[\mathbf{w}] := \mathbb{R}^n$; return } ;

$[\mathbf{x}](t_2) := [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\mathbf{w}], [\mathbf{u}])$.

Finalement $[\mathbf{x}](t_2) = [\varphi]^{t_2}([\mathbf{x}](t_1), \mathbf{u})$ et $[\mathbf{w}] = [\varphi]^{[t_1, t_2]}([\mathbf{x}](t_1), \mathbf{u})$



Algorithme $[\varphi]$

$[\varphi]$ (in : $[t] = [t_1, t_2], [\mathbf{x}], [\mathbf{u}]$, out :$[\mathbf{w}], [\mathbf{x}]([t_2])$)

1 $\quad [\hat{\mathbf{x}}](t_2) := [\mathbf{x}] + (t_2 - t_1) * [\mathbf{f}]([\mathbf{x}](t_1), [\mathbf{u}]);$

2 $\quad [\mathbf{v}] := [\mathbf{x}](t_1) \sqcup [\hat{\mathbf{x}}](t_2);$

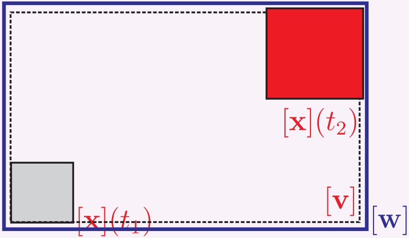
3 $\quad [\mathbf{w}] :=$ inflate$([\mathbf{v}], \alpha.w([\mathbf{v}]) + \beta);$

$\quad$ if $[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\mathbf{w}]), [\mathbf{u}]) \not\subseteq [\mathbf{w}]$

$\quad \{ [\mathbf{w}] := \mathbb{R}^n;$ return$\}$;

5 $\quad [\mathbf{x}](t_2) := [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\mathbf{w}], [\mathbf{u}]).$

Finalement $\quad [\mathbf{x}](t_2) = [\varphi]^{t_2}([\mathbf{x}](t_1), \mathbf{u})$ et $[\mathbf{w}] = [\varphi]^{[t_1, t_2]}([\mathbf{x}](t_1), \mathbf{u})$



Algorithme $[\varphi]$

$[\varphi]$ (in : $[t] = [t_1, t_2], [\mathbf{x}], [\mathbf{u}]$, out :$[\mathbf{w}], [\mathbf{x}]([t_2]))$

$[\hat{\mathbf{x}}](t_2) := [\mathbf{x}] + (t_2 - t_1) * [\mathbf{f}]([\mathbf{x}](t_1), [\mathbf{u}]);$

$[\mathbf{v}] := [\mathbf{x}](t_1) \sqcup [\hat{\mathbf{x}}](t_2);$

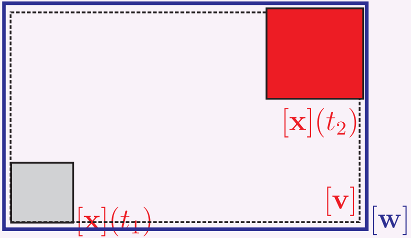
$[\mathbf{w}] :=$ inflate$([\mathbf{v}], \alpha.w([\mathbf{v}]) + \beta);$

if $[\mathbf{x}](t_1) + [0, t_2 - t_1] * [\mathbf{f}]([\mathbf{w}]), [\mathbf{u}]) \not\subseteq [\mathbf{w}]$

{ $[\mathbf{w}] := \mathbb{R}^n$; return} ;

$[\mathbf{x}](t_2) := [\mathbf{x}](t_1) + (t_2 - t_1) * [\mathbf{f}]([\mathbf{w}], [\mathbf{u}]).$

Finalement $[\mathbf{x}](t_2) = [\varphi]^{t_2}([\mathbf{x}](t_1), \mathbf{u})$ et $[\mathbf{w}] = [\varphi]^{[t_1, t_2]}([\mathbf{x}](t_1), \mathbf{u})$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

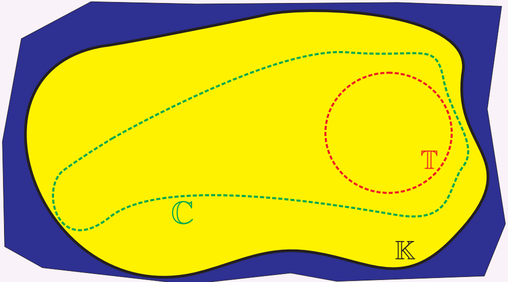
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $\left([\varphi]^{t}\left([\mathbf{x}], \mathbf{u}\right) \subset \mathbb{C}^{-} \wedge [\varphi]^{[0,t]}\left([\mathbf{x}], \mathbf{u}\right) \subset \mathbb{K}\right) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^{t}([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^{-} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^{-} \subset \mathbb{C} \subset \mathbb{C}^{+}$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

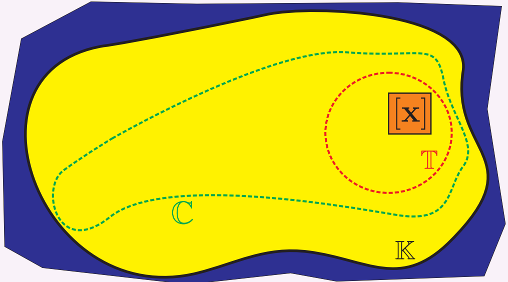
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $([\varphi]^t([\mathbf{x}], \mathbf{u}) \subset \mathbb{C}^- \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbf{u}) \subset \mathbb{K}) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^t([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^- = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

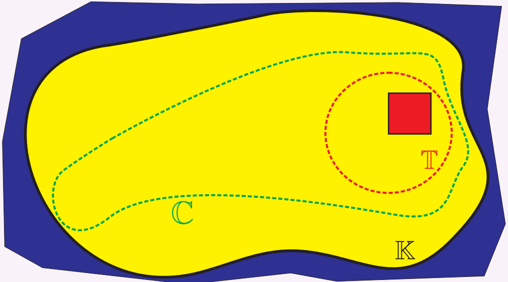
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $([\varphi]^{t}([\mathbf{x}], \mathbf{u}) \subset \mathbb{C}^{-} \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbf{u}) \subset \mathbb{K}) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^{t}([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^{-} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^{-} \subset \mathbb{C} \subset \mathbb{C}^{+}$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

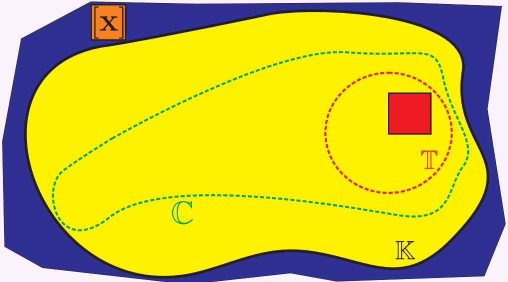
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $\left([\varphi]^{t}\left([\mathbf{x}], \mathbf{u}\right) \subset \mathbb{C}^{-} \wedge [\varphi]^{[0,t]}\left([\mathbf{x}], \mathbf{u}\right) \subset \mathbb{K}\right) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^{t}([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^{-} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^{-} \subset \mathbb{C} \subset \mathbb{C}^{+}$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

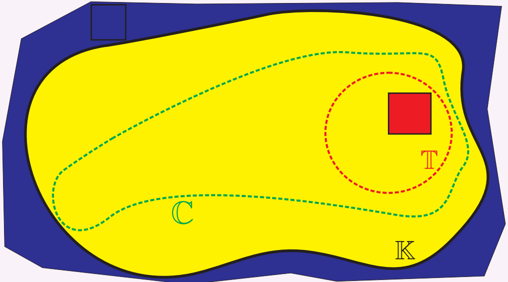
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $\left([\boldsymbol{\varphi}]^t\left([\mathbf{x}], \mathbf{u}\right) \subset \mathbb{C}^- \wedge [\boldsymbol{\varphi}]^{[0,t]}\left([\mathbf{x}], \mathbf{u}\right) \subset \mathbb{K}\right) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^t([\mathrm{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathrm{x}], \mathbb{U}) \cap \mathbb{C}^- = \emptyset \Rightarrow [\mathrm{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathrm{x}]$ est coupé en deux pavés $[\mathrm{x}_1]$ et $[\mathrm{x}_2]$.
- Finalement, $\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

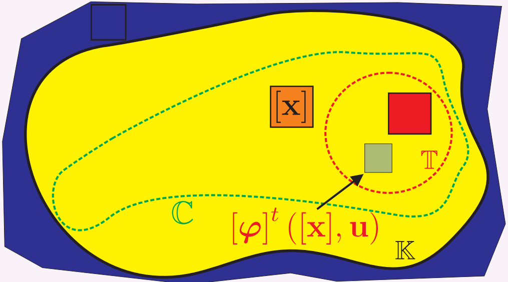
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $([\varphi]^t([\mathbf{x}], \mathbf{u}) \subset \mathbb{C}^- \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbf{u}) \subset \mathbb{K}) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^t([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^- = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

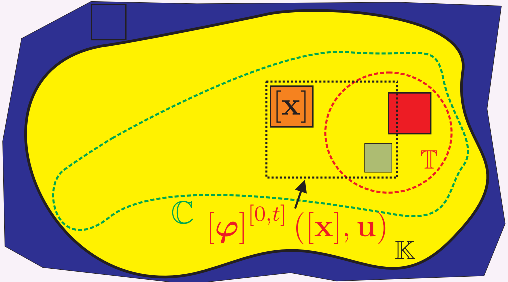
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $([\varphi]^{t}([\mathbf{x}], \mathbf{u}) \subset \mathbb{C}^{-} \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbf{u}) \subset \mathbb{K}) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^{t}([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^{-} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^{-} \subset \mathbb{C} \subset \mathbb{C}^{+}$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

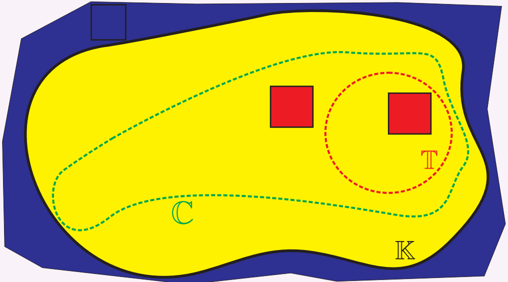
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $([\boldsymbol{\varphi}]^{t}([\mathbf{x}], \mathbf{u}) \subset \mathbb{C}^{-} \wedge [\boldsymbol{\varphi}]^{[0,t]}([\mathbf{x}], \mathbf{u}) \subset \mathbb{K}) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\boldsymbol{\varphi}]^{t}([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\boldsymbol{\varphi}]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^{-} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^{-} \subset \mathbb{C} \subset \mathbb{C}^{+}$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

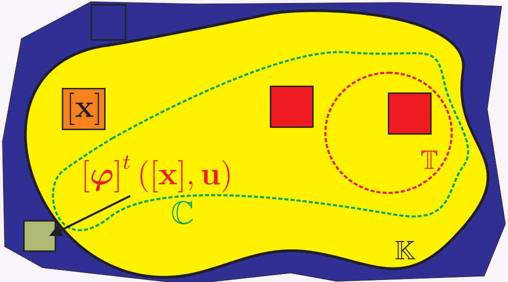
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $([\varphi]^t([\mathbf{x}], \mathbf{u}) \subset \mathbb{C}^- \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbf{u}) \subset \mathbb{K}) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^t([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^- = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

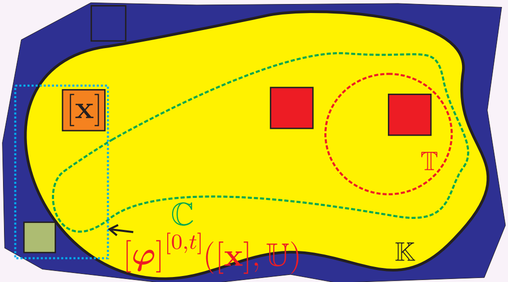
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $([\varphi]^{t}([\mathbf{x}], \mathbf{u}) \subset \mathbb{C}^{-} \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbf{u}) \subset \mathbb{K}) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^{t}([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^{-} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^{-} \subset \mathbb{C} \subset \mathbb{C}^{+}$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

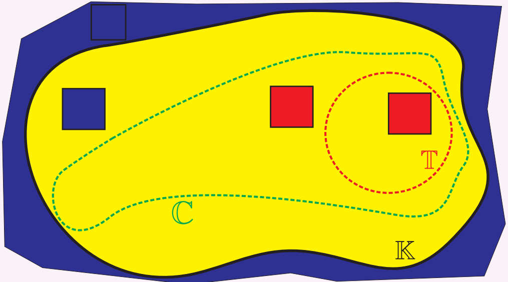
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $([\varphi]^t([\mathbf{x}], \mathbf{u}) \subset \mathbb{C}^- \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbf{u}) \subset \mathbb{K}) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^t([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^- = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

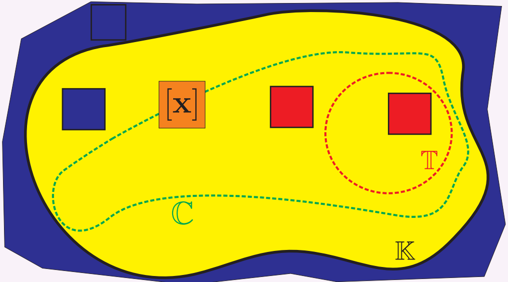
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $([\varphi]^t([\mathbf{x}], \mathbf{u}) \subset \mathbb{C}^- \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbf{u}) \subset \mathbb{K}) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^t([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^- = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

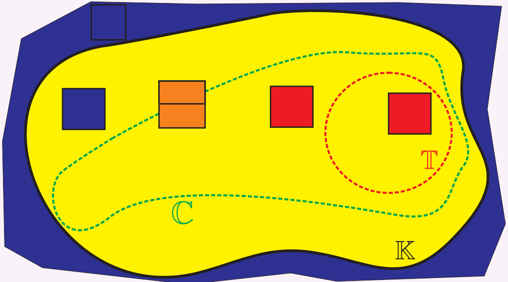
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $([\varphi]^{t}([\mathbf{x}], \mathbf{u}) \subset \mathbb{C}^{-} \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbf{u}) \subset \mathbb{K}) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^{t}([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^{-} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^{-} \subset \mathbb{C} \subset \mathbb{C}^{+}$



Algorithme du Bassin de Capture

L'algorithme du bassin de capture est basé sur les propriétés suivantes :

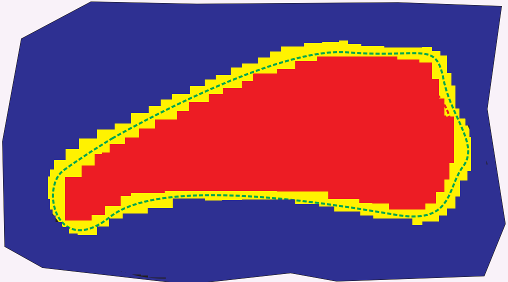
(i) $[\mathbf{x}] \subset \mathbb{T} \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(ii) $[\mathbf{x}] \cap \mathbb{K} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

(iii) $([\varphi]^t([\mathbf{x}], \mathbf{u}) \subset \mathbb{C}^- \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbf{u}) \subset \mathbb{K}) \Rightarrow [\mathbf{x}] \subset \mathbb{C}$

(iv) $[\varphi]^t([\mathbf{x}], \mathbb{U}) \cap \mathbb{K} = \emptyset \wedge [\varphi]^{[0,t]}([\mathbf{x}], \mathbb{U}) \cap \mathbb{C}^- = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{C} = \emptyset$

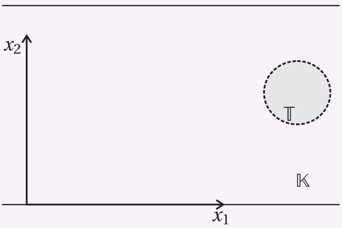
- Sinon, $[\mathbf{x}]$ est coupé en deux pavés $[\mathbf{x}_1]$ et $[\mathbf{x}_2]$.
- Finalement, $\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+$



Illustration

Problème de Zermelo

- Le système dynamique considéré en illustration schématise l'évolution d'un navire sur une rivière ($\mathbb{K} = [-8, 8] \times [-4, 4]$) qui peut contrôler v (puissance) et θ (direction). Le navire doit atteindre une île $\mathbb{T} \triangleq \mathcal{B}(0, 1)$.

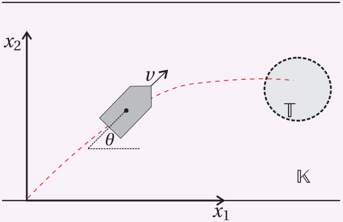


Illustration

Problème de Zermelo

- Dynamique ($\mathbf{x}'(t) = \mathbf{f}(\mathbf{x}(t), \boldsymbol{u})$) du navire

$$\begin{cases} x_1'(t) &= v\cos(\theta) \\ x_2'(t) &= v\sin(\theta) \end{cases},$$



avec les commandes $v \in [0, 0.8]$ and $\theta \in [-\pi, \pi]$.

- La dynamique globale du système (navire + courant) est donnée par

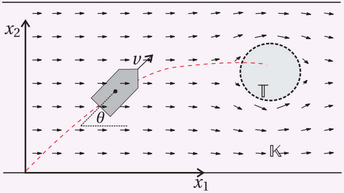
$$\begin{cases} x_1'(t) &= 1 + \dfrac{x_2^2 - x_1^2}{(x_1^2 + x_2^2)^2} + v\cos(\theta) \\ x_2'(t) &= \dfrac{-2x_1x_2}{(x_1^2 + x_2^2)^2} + v\sin(\theta) \end{cases}.$$

Illustration

Problème de Zermelo

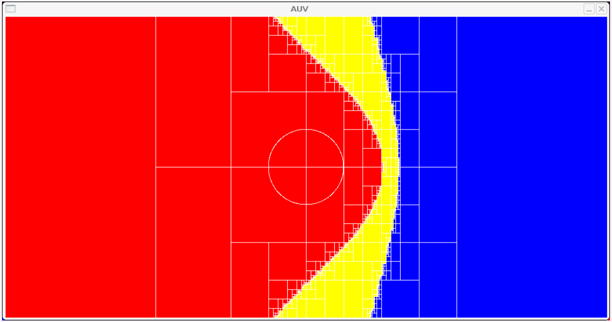
- Dynamique ($\mathbf{x}'(t) = \mathbf{f}(\mathbf{x}(t), \boldsymbol{u})$) du navire avec les commandes $v \in [0, 0.8]$ and $\theta \in [-\pi, \pi]$.
- La dynamique globale du système (navire + courant) est donnée par

$$\left\{ \begin{array}{lcl} x_1'(t) & = & 1 + \dfrac{x_2^2 - x_1^2}{(x_1^2 + x_2^2)^2} + v\cos(\theta) \\ x_2'(t) & = & \dfrac{-2x_1x_2}{(x_1^2 + x_2^2)^2} + v\sin(\theta) \end{array} \right. .$$



Illustration

Résultat



where

- Le circle délimite la frontière de la cible $\mathbb{T}$;
- $\mathbb{C}^-$ = union de toutes les boîtes rouges ;
- $\mathbb{C}^+$ = union de toutes les boîtes rouges et jaunes ;
- Finalement, $\mathbb{C}^- \subset \mathbb{C} \subset \mathbb{C}^+$.

Perspectives

- Traiter des problèmes en grandes dimensions.
- Trouver une suite de commandes garanties permettant d'atteindre la cible en un temps fini.
- Donner une approximation intérieure et extérieure du noyau de viabilité

$$\text{Viab}(K) \quad = \quad \{x_0 \in \mathbb{K}, \exists \mathbf{u}, \forall t \geq 0, x(t) \in \mathbb{K}\}.$$

Perspectives

- Traiter des problèmes en grandes dimensions.
- Trouver une suite de commandes garanties permettant d'atteindre la cible en un temps fini.
- Donner une approximation intérieure et extérieure du noyau de viabilité

$$\mathrm{Viab}(K) = \{x_0 \in \mathbb{K}, \exists \mathbf{u}, \forall t \geq 0, x(t) \in \mathbb{K}\}.$$

Perspectives

- Traiter des problèmes en grandes dimensions.
- Trouver une suite de commandes garanties permettant d'atteindre la cible en un temps fini.
- Donner une approximation intérieure et extérieure du noyau de viabilité

$$\text{Viab}(K) \quad = \quad \{x_0 \in \mathbb{K}, \exists \mathbf{u}, \forall t \geq 0, x(t) \in \mathbb{K}\}\,.$$