

Maintenir la viabilité ou la résilience d'un système : les machines à vecteurs de support pour rompre la malédiction de la dimensionnalité ?

Laetitia Chapel

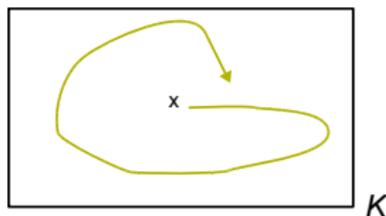
Directeur de thèse : Guillaume Deffuant
Laboratoire d'Ingénierie des Systèmes Complexes

Soutenance de thèse
19 octobre 2007



- Contrôler un système dynamique afin qu'il survive dans un ensemble d'états admissibles
- État $x(t)$, contrôles $u(t)$

$$\begin{cases} x'(t) = \varphi(x(t), u(t)), & \text{pour tout } t \geq 0 \\ u(t) \in U(x(t)) \subset \mathbb{R}^q \end{cases} \quad (1)$$



Exemple : contrôle de la pêche dans un écosystème marin

- Variables d'états : biomasses des espèces de l'écosystème, la pêche
 - biomasse minimale et maximale pour chaque espèce
 - la pêche doit garantir un revenu suffisant aux pêcheurs
- Variable de contrôle : la variation de la pêche
 - la variation ne peut pas être trop brusque
- **Question** : comment contrôler la pêche afin de conserver un écosystème pérenne ?

Cadre spécifique

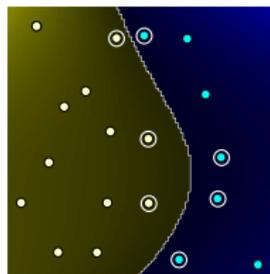
- Théorie de la viabilité (Aubin)
- Outil informatique : algorithme de viabilité (Saint-Pierre)
- Première étape : calcul du noyau de viabilité
- Mais
 - le résultat est un ensemble de points, difficile à manipuler
 - les algorithmes d'approximation souffrent de la malédiction de la dimensionnalité dans l'espace d'état et des contrôles
- Deuxième étape : contrôle du système

Idée : résumer l'ensemble de points par une fonction

Un pari : les machines à vecteurs de support (SVMs)

- Méthode de discrimination
 - **Entrée** : base d'apprentissage, n points étiquetés $+1/-1$
 - **Sortie** : combinaison linéaire $f(x) = \sum_{i=1}^n \alpha_i y_i k(x, x_i) + b$
- Particularité de la solution
 - α_i : influence d'un point sur la solution \rightarrow parcimonie
 - $k(x, x_i)$ projette (virtuellement) les points dans un espace déployé
 - solution non linéaire dans l'espace des données

$$k(x, x_i) = \exp\left(-\frac{\|x-x_i\|^2}{\sigma}\right)$$



Objectifs de la thèse

- Proposer des algorithmes d'approximation de noyau de viabilité et de bassins de capture qui permettent de travailler en plus grande dimension
 - dans l'espace d'état
 - dans l'espace des contrôles
- Définir des contrôleurs compacts et rapides

Double exigence

- Théorique
- Pratique



1. Approcher des noyaux de viabilité avec des SVMs
2. Apprentissage actif de noyau de viabilité
3. Approcher des bassins de capture avec des SVMs
4. Conclusion générale



1. Approcher des noyaux de viabilité avec des SVMs
2. Apprentissage actif de noyau de viabilité
3. Approcher des bassins de capture avec des SVMs
4. Conclusion générale



Approcher des noyaux de viabilité avec des SVMs

Un exemple simple : population

- Exemple simple de croissance d'une population dans un espace limité
- Système dynamique

$$\begin{cases} p(t + dt) = p(t) + p(t)y(t)dt \\ y(t + dt) = y(t) + u(t)dt \end{cases} \quad (2)$$

- Sous contraintes

- $p \in [p_{min}, p_{max}]$
- $y \in [y_{min}, y_{max}]$
- $u \in [-u_{max}, u_{max}]$

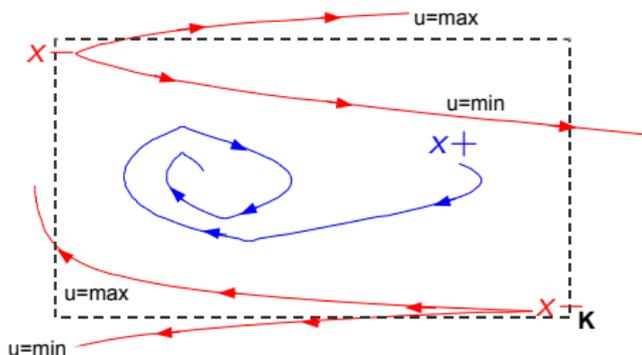


Approcher des noyaux de viabilité avec des SVMs

Un exemple simple : population

Comment contrôler le système afin qu'il reste toujours dans K ?

- Approcher le noyau de viabilité du système

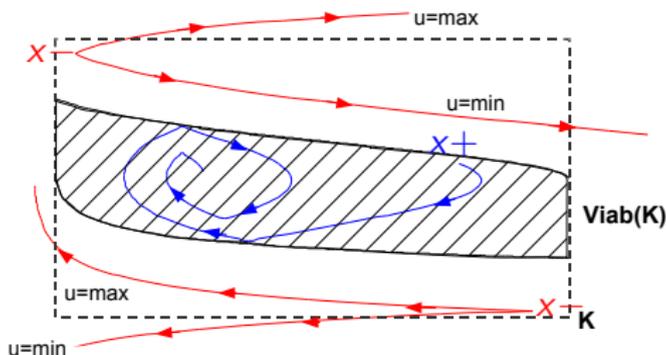


Approcher des noyaux de viabilité avec des SVMs

Un exemple simple : population

Comment contrôler le système afin qu'il reste toujours dans K ?

- Approcher le noyau de viabilité du système

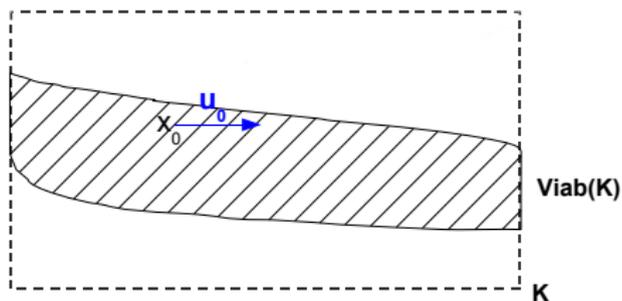


Approcher des noyaux de viabilité avec des SVMs

Un exemple simple : population

Comment contrôler le système afin qu'il reste toujours dans K ?

- Contrôleur lourd de viabilité

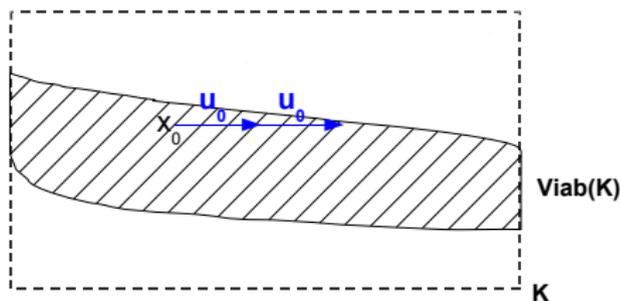


Approcher des noyaux de viabilité avec des SVMs

Un exemple simple : population

Comment contrôler le système afin qu'il reste toujours dans K ?

- Contrôleur lourd de viabilité

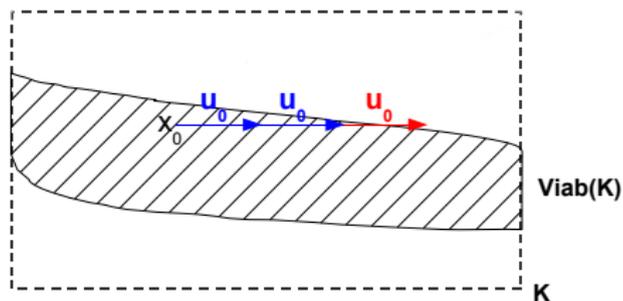


Approcher des noyaux de viabilité avec des SVMs

Un exemple simple : population

Comment contrôler le système afin qu'il reste toujours dans K ?

- Contrôleur lourd de viabilité

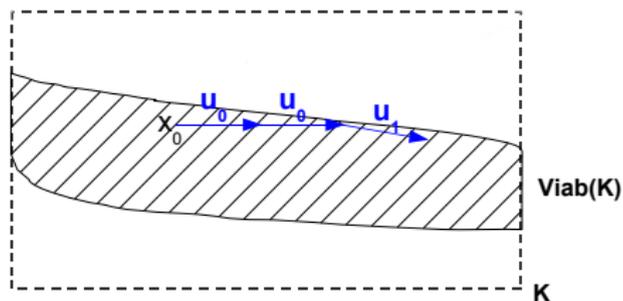


Approcher des noyaux de viabilité avec des SVMs

Un exemple simple : population

Comment contrôler le système afin qu'il reste toujours dans K ?

- Contrôleur lourd de viabilité



Approcher des noyaux de viabilité avec des SVMs

Apprendre un noyau de viabilité avec une méthode de discrimination

Apprendre un noyau de viabilité en utilisant une méthode de discrimination

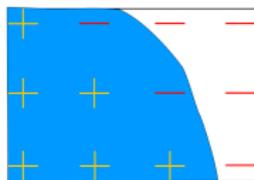
- Algorithme itératif, basé sur l'algorithme de viabilité de Saint-Pierre
- Discrétisation de K
- Base d'apprentissage, composée des points de la grille associés à l'étiquette
 - +1 si le point est viable à l'itération suivante
 - -1 sinon

Approcher des noyaux de viabilité avec des SVMs

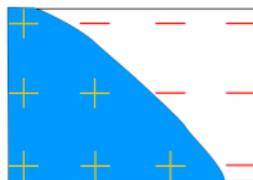
Apprendre un noyau de viabilité avec une méthode de discrimination

Théorème

Sous certaines conditions sur la qualité de l'apprentissage et sur la dynamique, l'algorithme fournit une approximation du noyau de viabilité qui converge vers le vrai noyau lorsque la résolution de la grille tend vers 0



valide



valide



non valide

Approcher des noyaux de viabilité avec des SVMs

Apprendre un noyau de viabilité avec une méthode de discrimination

Les SVMs comme méthode de discrimination : avantages

- Permet l'utilisation d'une méthode d'optimisation (descente de gradient) pour trouver un contrôle viable
- Permet de travailler avec plusieurs pas de temps à la fois
- La fonction SVM peut servir de contrôleur
- La solution est parcimonieuse : moins de points en mémoire ?

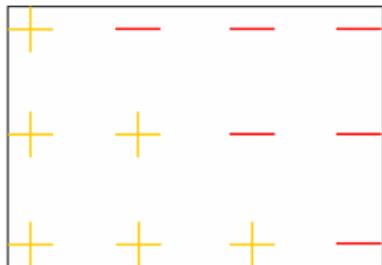
Approcher des noyaux de viabilité avec des SVMs

Apprendre un noyau de viabilité avec une méthode de discrimination

- Mise à jour des étiquettes

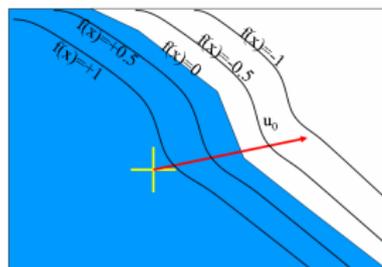
Saint-Pierre

Test exhaustif des contrôles discrétisés



avec SVMs

Optimisation pour trouver un contrôle viable



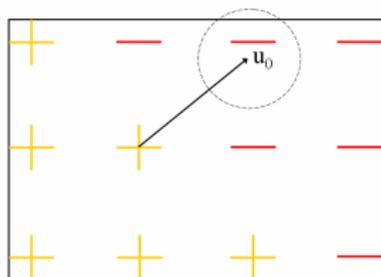
Approcher des noyaux de viabilité avec des SVMs

Apprendre un noyau de viabilité avec une méthode de discrimination

- Mise à jour des étiquettes

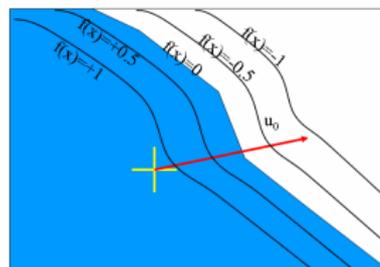
Saint-Pierre

Test exhaustif des contrôles discrétisés



avec SVMs

Optimisation pour trouver un contrôle viable



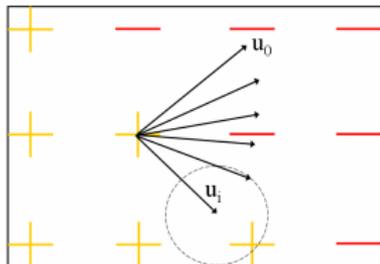
Approcher des noyaux de viabilité avec des SVMs

Apprendre un noyau de viabilité avec une méthode de discrimination

- Mise à jour des étiquettes

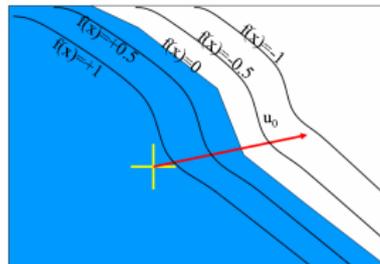
Saint-Pierre

Test exhaustif des contrôles discrétisés



avec SVMs

Optimisation pour trouver un contrôle viable



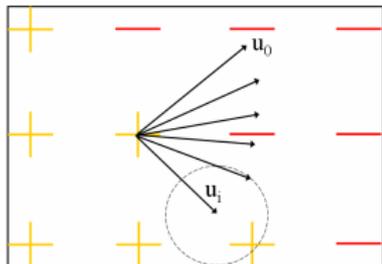
Approcher des noyaux de viabilité avec des SVMs

Apprendre un noyau de viabilité avec une méthode de discrimination

- Mise à jour des étiquettes

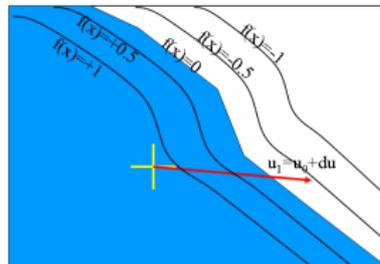
Saint-Pierre

Test exhaustif des contrôles discrétisés



avec SVMs

Optimisation pour trouver un contrôle viable



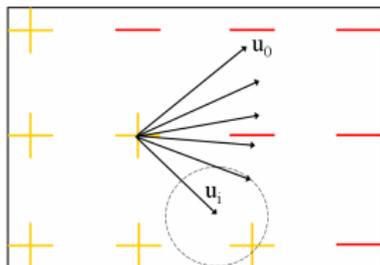
Approcher des noyaux de viabilité avec des SVMs

Apprendre un noyau de viabilité avec une méthode de discrimination

- Mise à jour des étiquettes

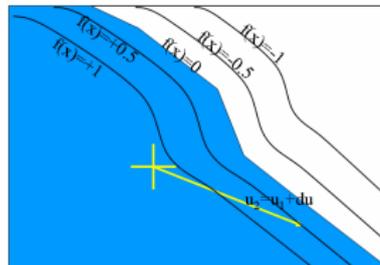
Saint-Pierre

Test exhaustif des contrôles discrétisés



avec SVMs

Optimisation pour trouver un contrôle viable



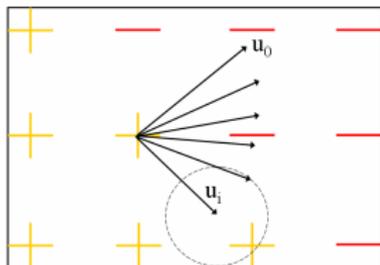
Approcher des noyaux de viabilité avec des SVMs

Apprendre un noyau de viabilité avec une méthode de discrimination

- Mise à jour des étiquettes

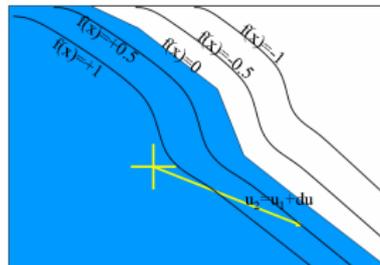
Saint-Pierre

Test exhaustif des contrôles discrétisés



avec SVMs

Optimisation pour trouver un contrôle viable



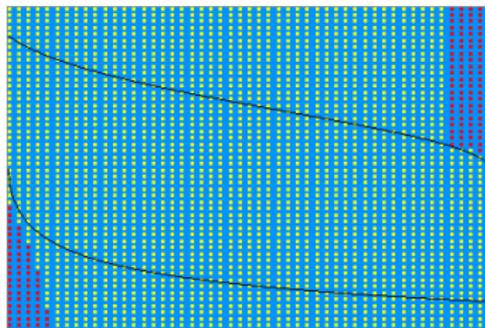
Extension à j pas de temps

Approcher des noyaux de viabilité avec des SVMs

Exemple d'application

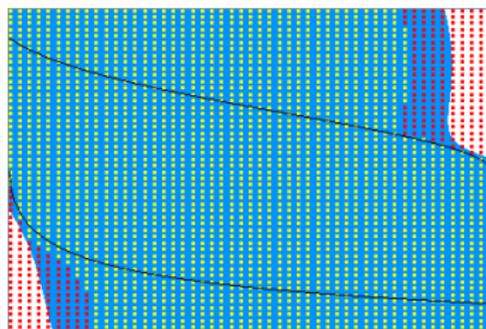
Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps



Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps



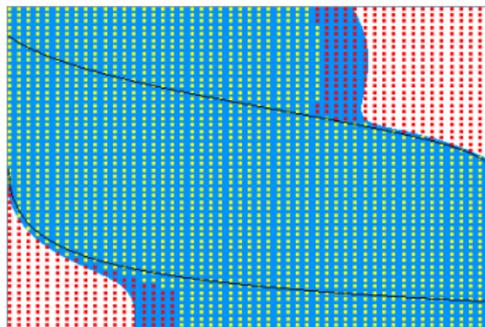
Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps



Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps



Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps

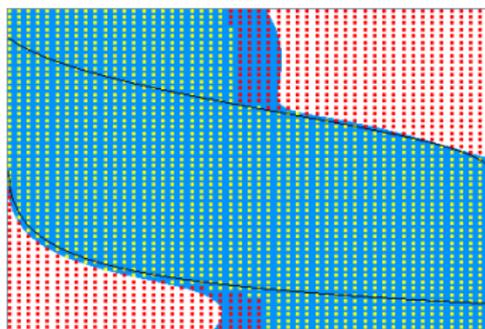


Approcher des noyaux de viabilité avec des SVMs

Exemple d'application

Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps



Approcher des noyaux de viabilité avec des SVMs

Exemple d'application

Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps

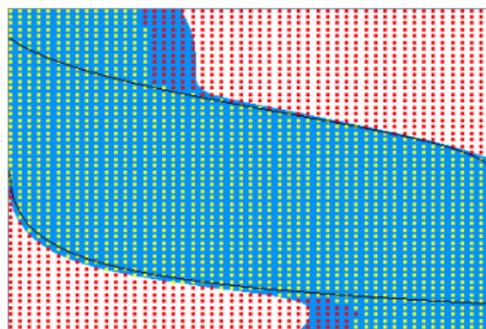


Approcher des noyaux de viabilité avec des SVMs

Exemple d'application

Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps



Approcher des noyaux de viabilité avec des SVMs

Exemple d'application

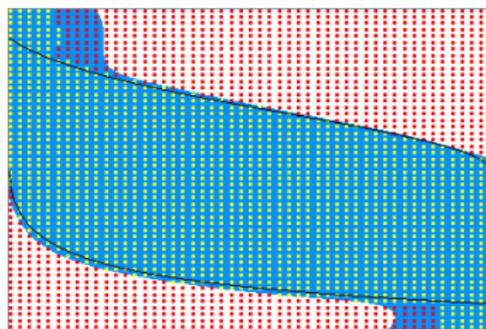
Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps



Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps

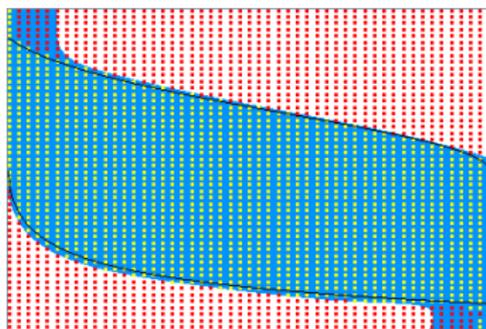


Approcher des noyaux de viabilité avec des SVMs

Exemple d'application

Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps

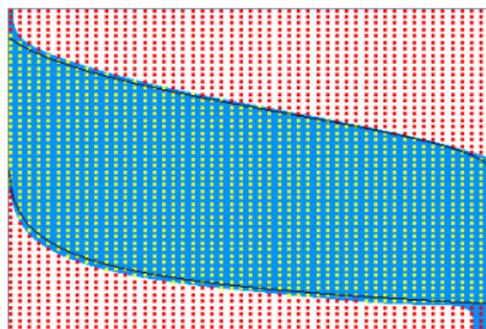


Approcher des noyaux de viabilité avec des SVMs

Exemple d'application

Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps

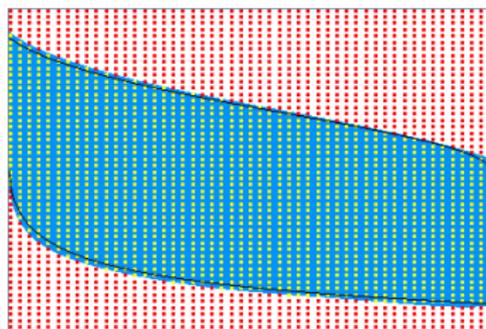


Approcher des noyaux de viabilité avec des SVMs

Exemple d'application

Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps

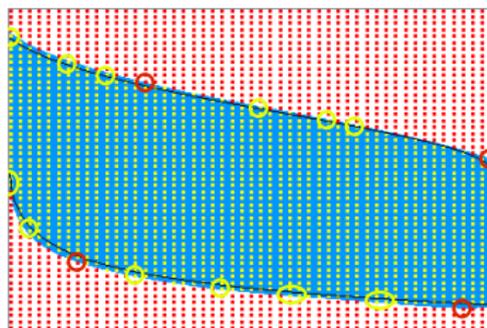


Approcher des noyaux de viabilité avec des SVMs

Exemple d'application

Approximation progressive du noyau de viabilité (population)

- Espace d'état en 2 dimensions, grille de 2601 points, 6 pas de temps



- 12 itérations, 19 SV

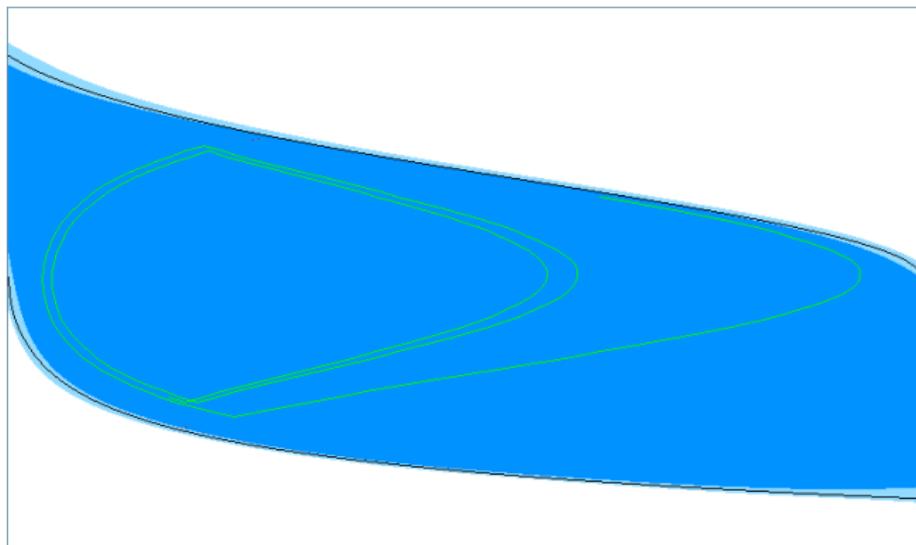
Contrôleur lourd de viabilité avec les SVMs

- Même contrôle u_0 jusqu'à ce que le point suivant atteigne $f(x) < \Delta$
- Trouver un contrôle viable en utilisant une descente de gradient sur f
- Contrôleur plus ou moins prudent, en anticipant sur plusieurs pas de temps

Approcher des noyaux de viabilité avec des SVMs

Contrôleur lourd de viabilité avec les SVMs

Exemple de contrôleur (vision à 5 pas)



Conclusion [Approximation de noyaux de viabilité]

- Utiliser des SVMs pour approcher un noyau de viabilité
 - fournit des **contrôleurs compacts**
 - **permet** de traiter des problèmes avec des espaces des contrôles très importants (grâce à l'optimisation des contrôles)
 - **ne permet pas** de traiter des problèmes avec des espaces d'état importants (toujours une discrétisation de l'espace d'état)



1. Approcher des noyaux de viabilité avec des SVMs
2. Apprentissage actif de noyau de viabilité
3. Approcher des bassins de capture avec des SVMs
4. Conclusion générale

- **Apprentissage actif** : limite le nombre de points à étiqueter / la taille de la base d'apprentissage
 - étiqueter les points est coûteux
 - la taille de la grille est exponentielle avec la dimension
 - apprendre une SVM est quadratique avec la taille de la base d'apprentissage
- On utilise la propriété de parcimonie des SVMs

Apprentissage actif de noyau de viabilité

Algorithme d'apprentissage actif

- 
- **But** : utiliser un nombre de points proche du nombre de SV
 - On ajoute progressivement les points les plus susceptibles d'être SV (par couples)
 - **Question** : quels points choisir ? → on se concentre sur la frontière
 - On utilise une grille multi-résolution virtuelle

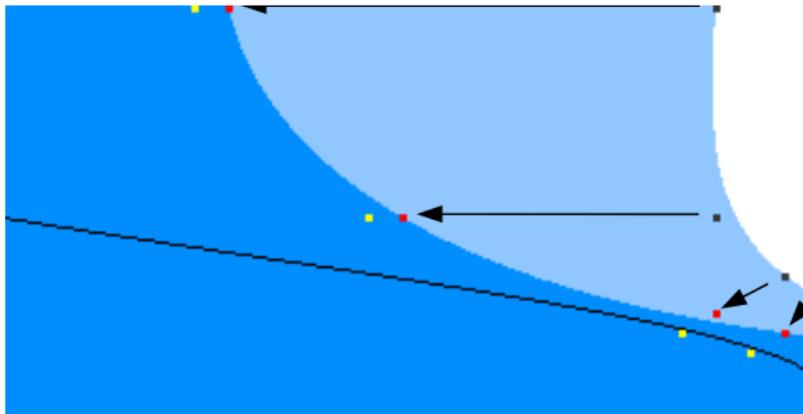
En images

- Grille multi-résolution de profondeur 3



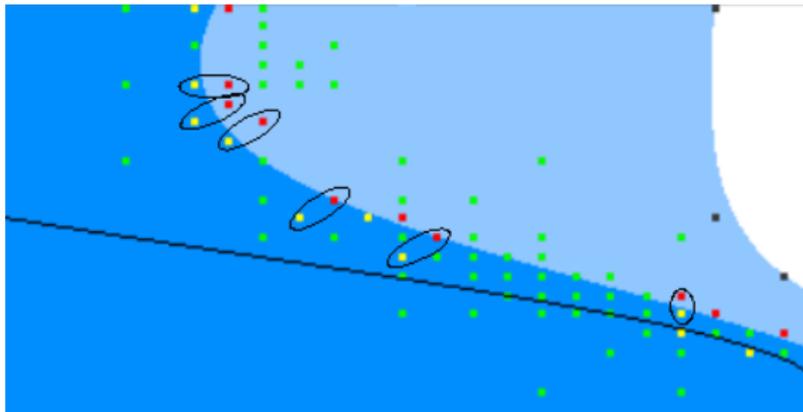
En images

- Grille multi-résolution de profondeur 3



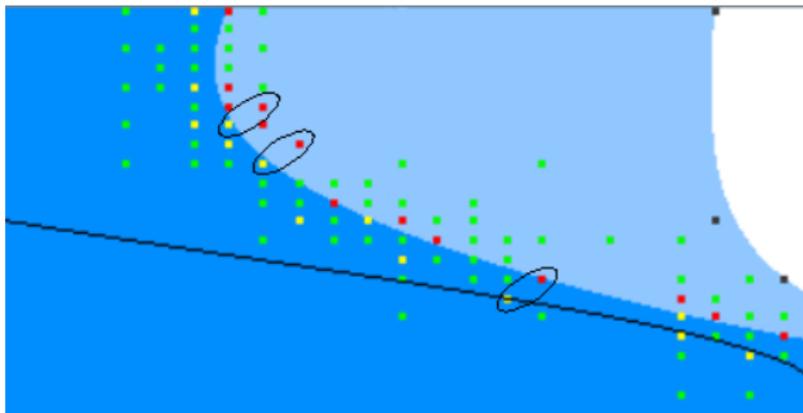
En images

- Grille multi-résolution de profondeur 3



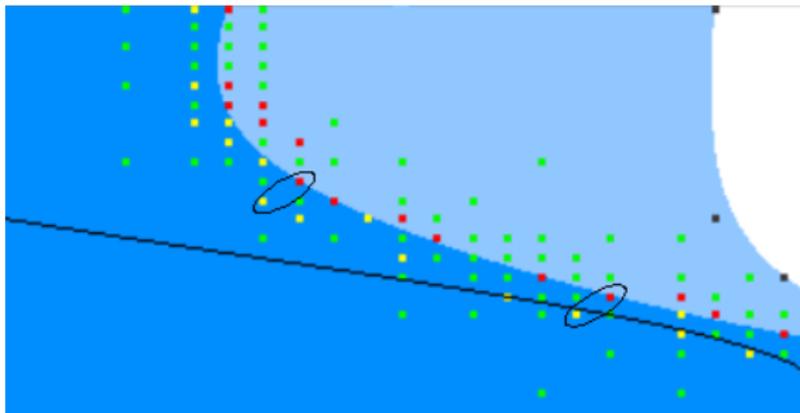
En images

- Grille multi-résolution de profondeur 3



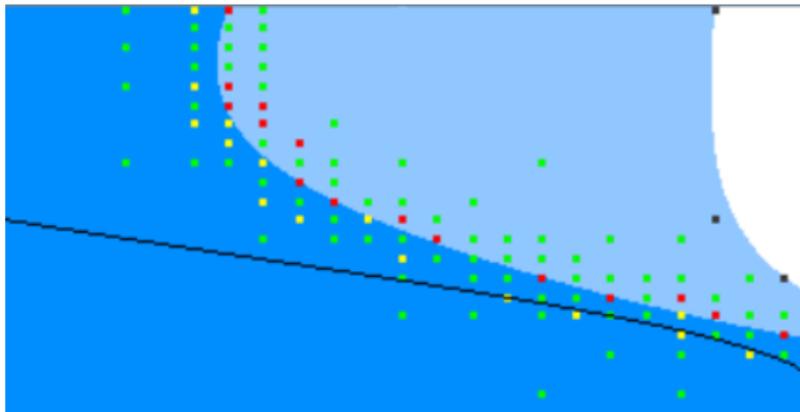
En images

- Grille multi-résolution de profondeur 3



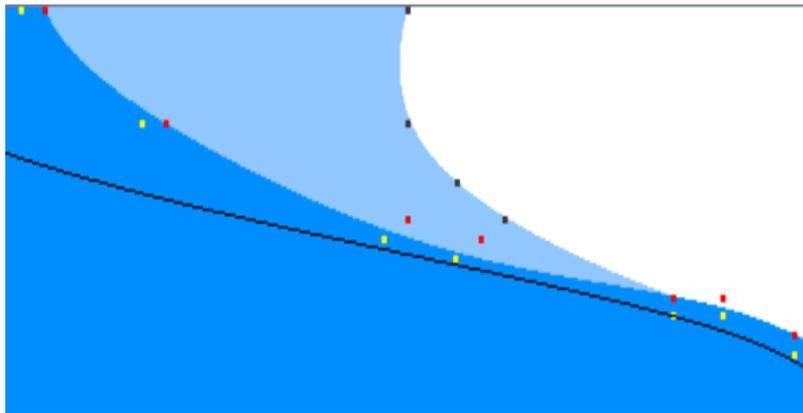
En images

- Grille multi-résolution de profondeur 3



En images

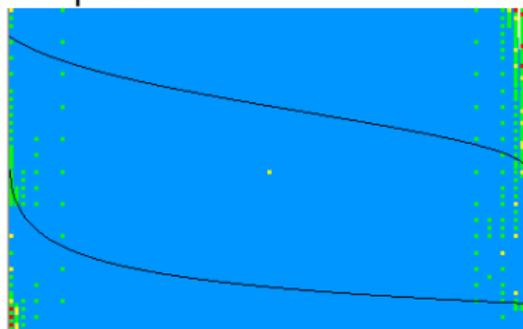
- Grille multi-résolution de profondeur 3



Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

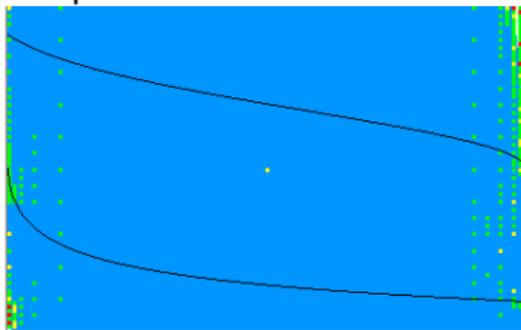


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète



- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

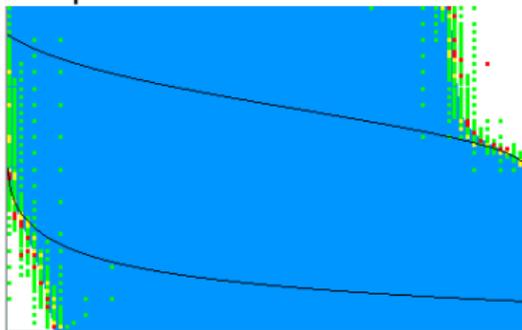


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

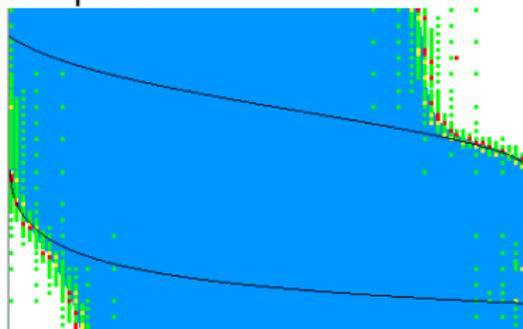


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

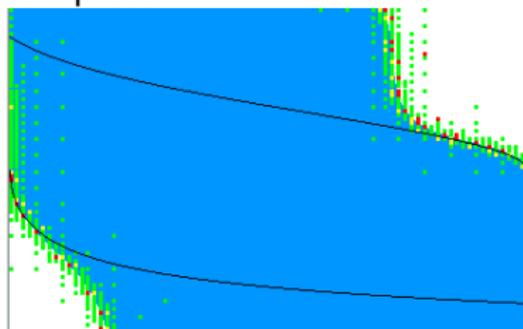


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

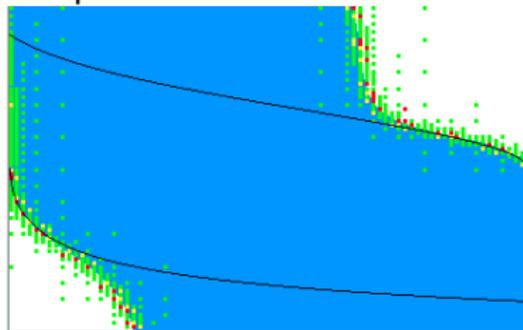


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

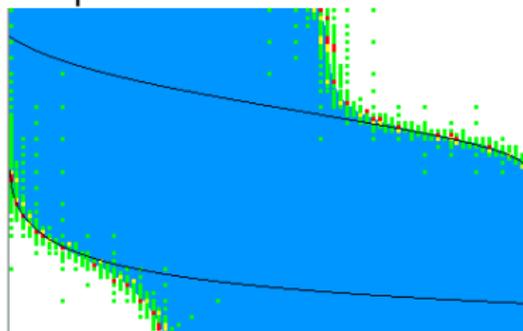


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

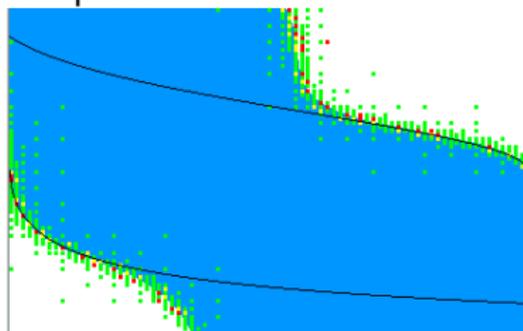


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

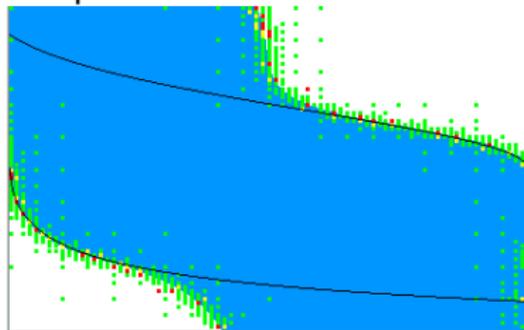


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

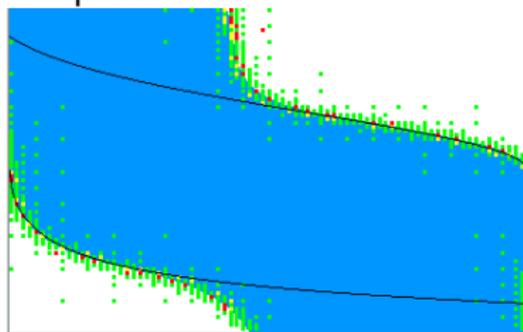


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

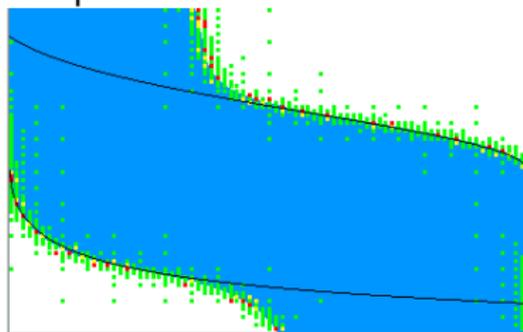


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

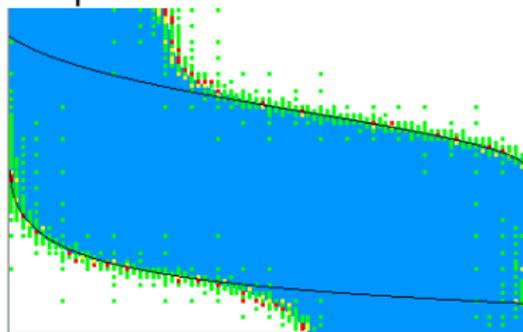


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

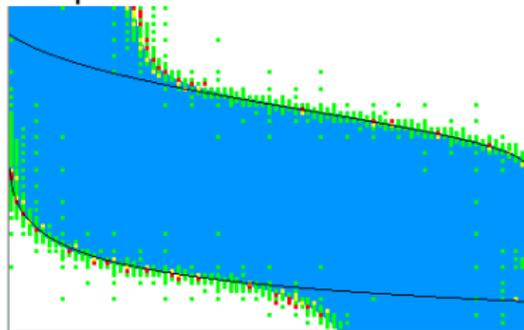


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

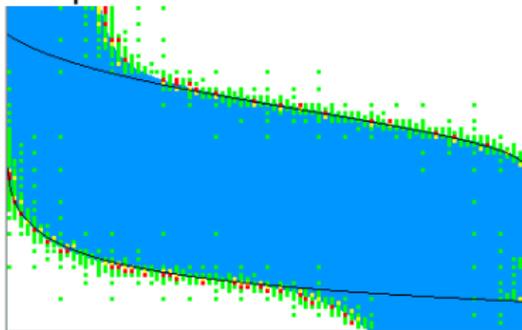


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

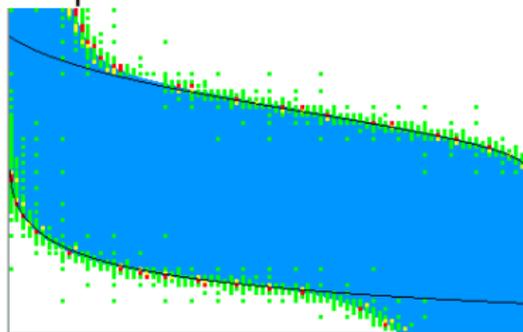


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

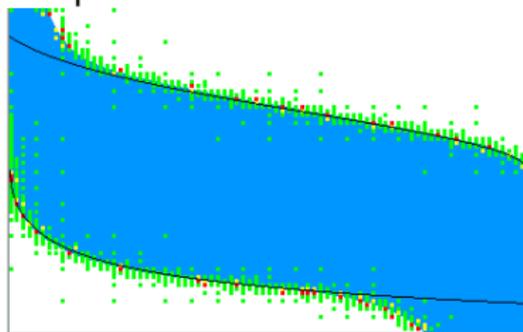


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

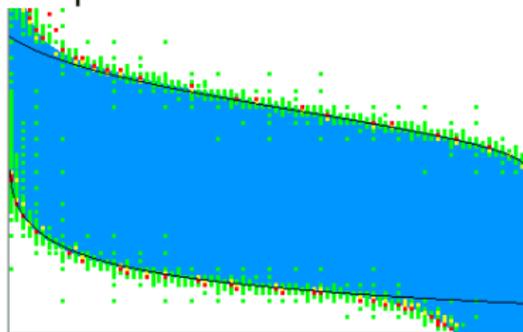


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

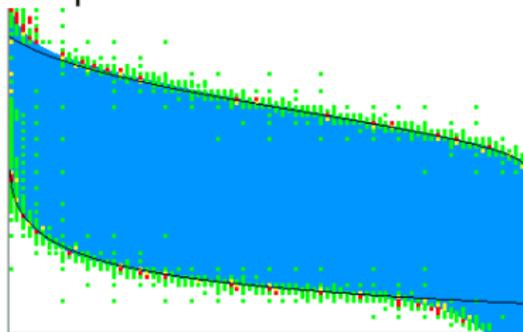


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

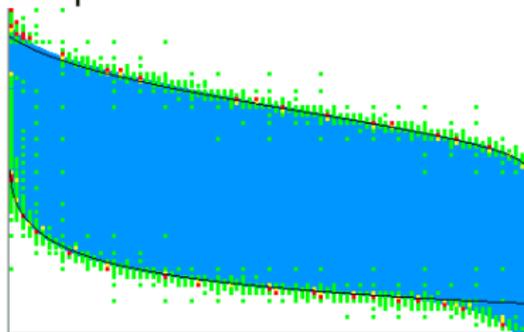


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

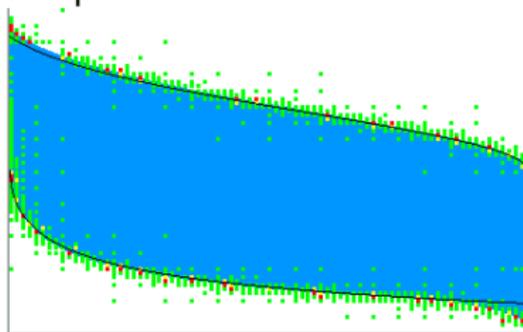


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

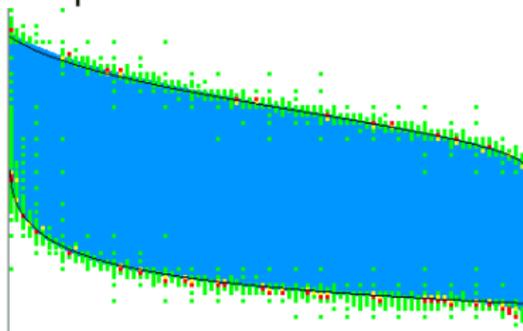


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

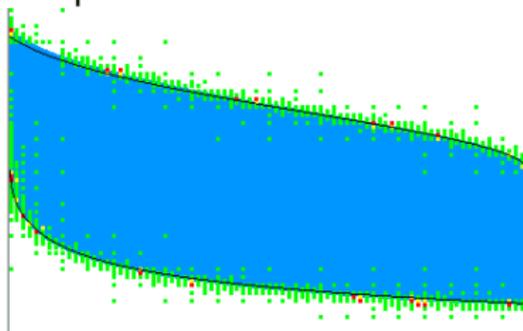


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète

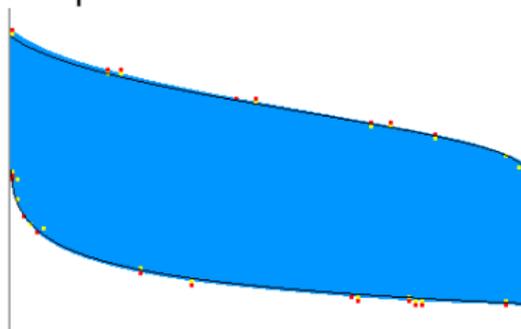


- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Apprentissage actif de noyau de viabilité

Exemple d'application (population)

- 11 points par dimension, grille de profondeur 4 \rightarrow 6561 points sur la grille complète



- 6 apprentissages par itération
- 28 SV, 761 (12%) max étiquetés, 124 (2%) max dans \mathcal{S}

Contrôle d'un vélo dans un circuit

- Dimension 6 pour l'espace d'état
 - angle de déplacement du guidon et vitesse de l'angle
 - angle du vélo par rapport à la verticale et vitesse de l'angle
 - position de la roue avant et angle d'inclinaison du vélo
- Dimension 2 dans l'espace des contrôles
 - poussée sur le guidon
 - déplacement du vélo

Apprentissage actif de noyau de viabilité

Contrôle d'un vélo dans un circuit

- 
- **Première étape:** approximation du noyau de viabilité en dimension 6
 - 531441 points sur la grille complète
 - 3914 SV, 34028 (6,5%) max dans \mathcal{S}
 - **Deuxième étape :** contrôler le système
 - **But :** conduire le vélo dans un circuit, sans le quitter et sans tomber

- Contrôle du vélo dans un circuit en 2 dimensions



Conclusion [Apprentissage actif]

- Permet de limiter la taille de la base d'apprentissage
 - **mais** le temps d'approximation est toujours exponentiel
 - **permet** d'obtenir un contrôleur rapide



1. Approcher des noyaux de viabilité avec des SVMs
2. Apprentissage actif de noyau de viabilité
3. Approcher des bassins de capture avec des SVMs
4. Conclusion générale

Approcher des bassins de capture avec des SVMs

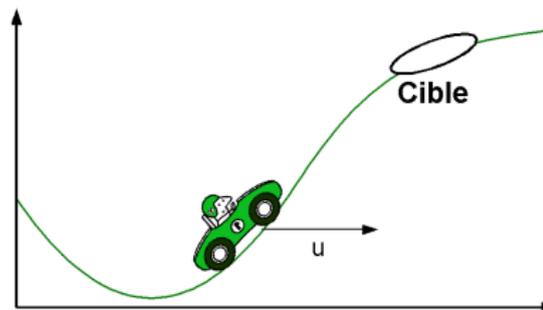
Un exemple simple : voiture sur la colline

- La voiture doit atteindre le plus vite possible le haut de la colline, tout en restant dans un ensemble donné
- Système dynamique

$$\begin{cases} p(t + dt) = p(t) + v(t)dt \\ v(t + dt) = v(t) + f(u(t))dt \end{cases} \quad (3)$$

- Sous contraintes

- $p \in [p_{min}, p_{max}]$
- $v \in [v_{min}, v_{max}]$
- $u \in [-u_{max}, u_{max}]$



Approcher des bassins de capture avec des SVMs

Définition

- Bassin de capture au temps t_{Max} : ensemble des états initiaux qui peuvent atteindre la cible avant le temps $\tau \leq t_{Max}$, tout en restant dans K
- **Algorithme** : ajout d'une dimension, le temps $\tau' = -1$
- On approche le noyau de viabilité du système étendu, avec $K \times [0, t_{Max}]$

$$\begin{aligned} \text{--si } x \notin C \quad F_C(x, \tau) &= \begin{cases} x'(t) = F(x(t), u(t)) \\ \tau'(t) = -1 \end{cases} \\ \text{--si } x \in C \quad F_C(x, \tau) &= 0 \end{aligned} \quad (4)$$

Algorithme d'approximation dans l'espace d'état initial

- Algorithme itératif, basé sur l'algorithme d'approximation de noyau de viabilité
- Initialisation : $+1$ si $x \in C$, -1 sinon
- Base d'apprentissage, composée des points de la grille associés à l'étiquette
 - $+1$ si le point peut atteindre la cible au pas de temps suivant
 - -1 sinon



Théorème [approximation par l'extérieur]

Sous certaines conditions sur la qualité de l'apprentissage et sur la dynamique, l'algorithme fournit une approximation du bassin de capture qui converge par l'extérieur vers le vrai bassin lorsque la résolution de la grille tend vers 0

Théorème [approximation par l'intérieur]

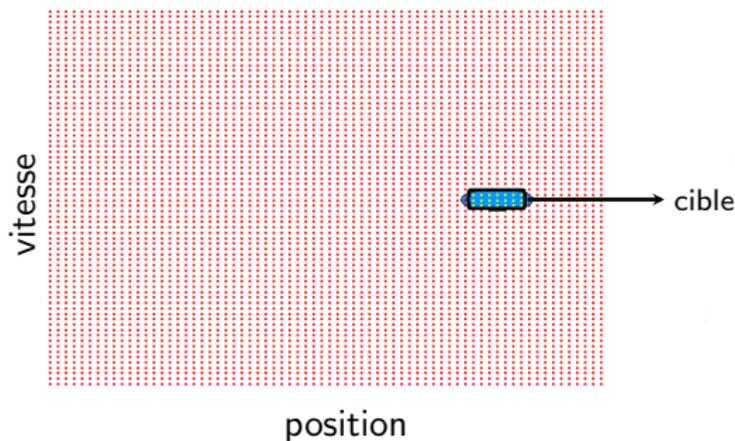
Sous certaines conditions sur la qualité de l'apprentissage et sur la dynamique, l'algorithme fournit une approximation du bassin de capture qui converge par l'intérieur vers le vrai bassin lorsque la résolution de la grille tend vers 0

Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

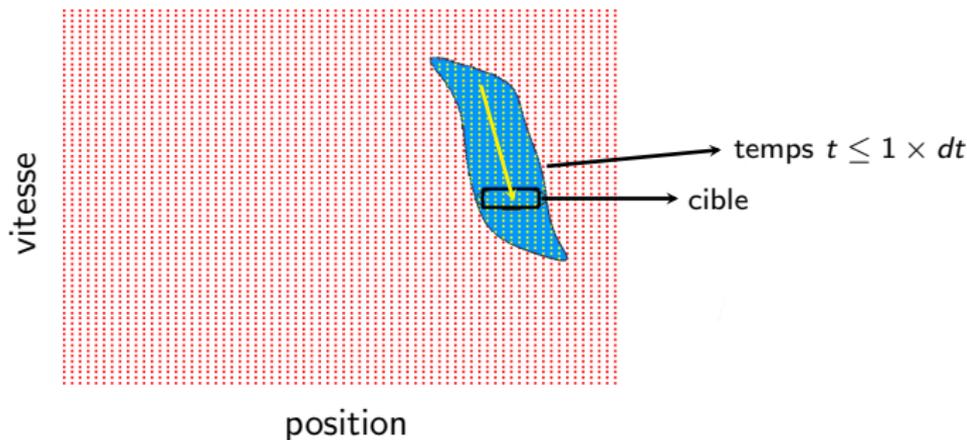


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

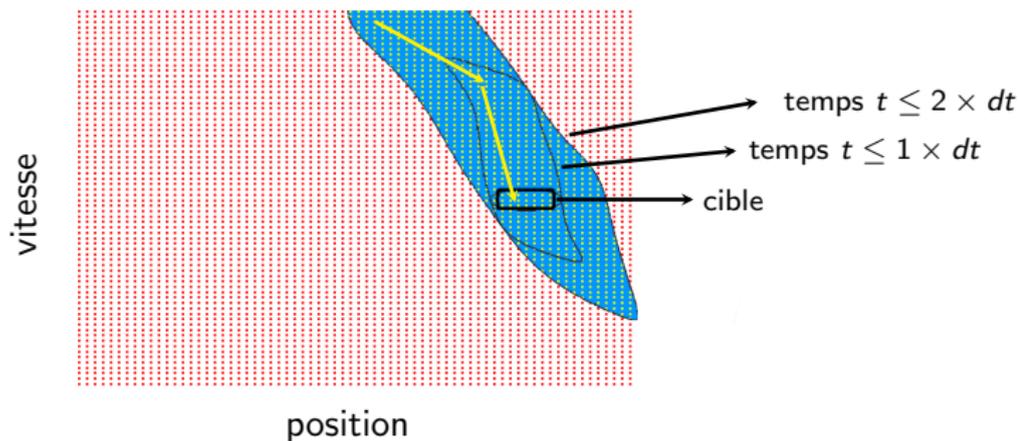


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

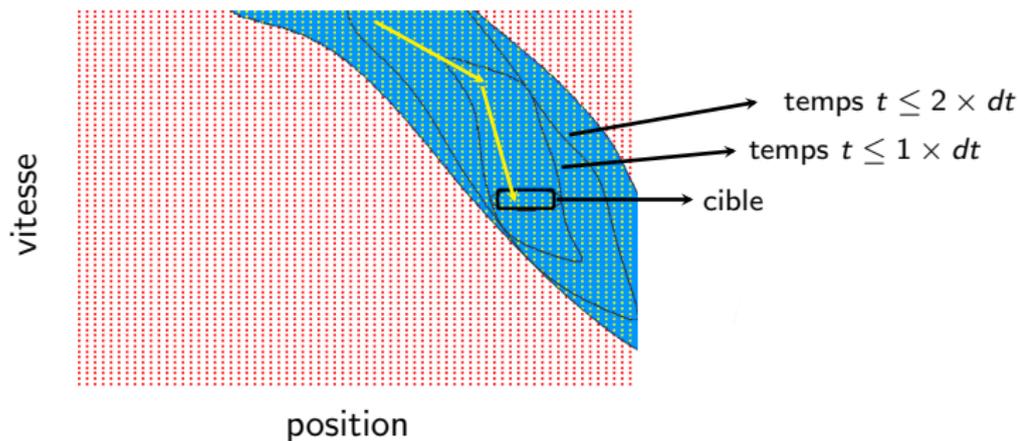


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

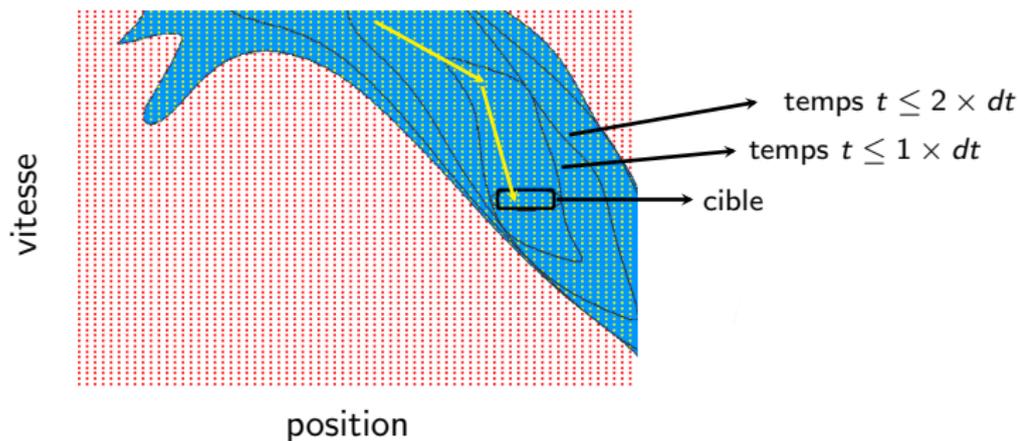


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

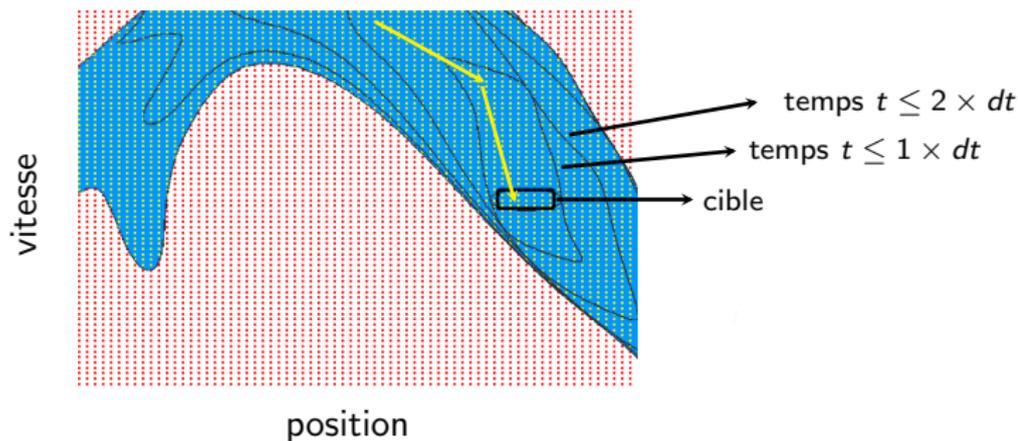


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

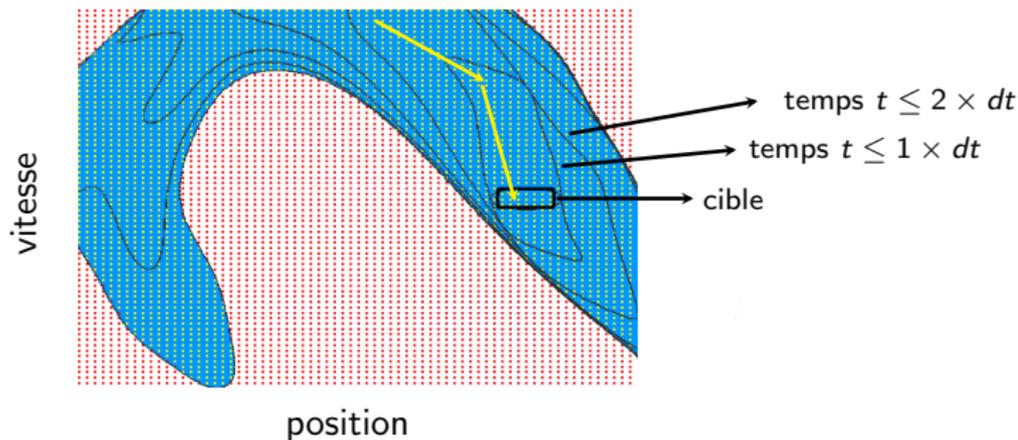


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

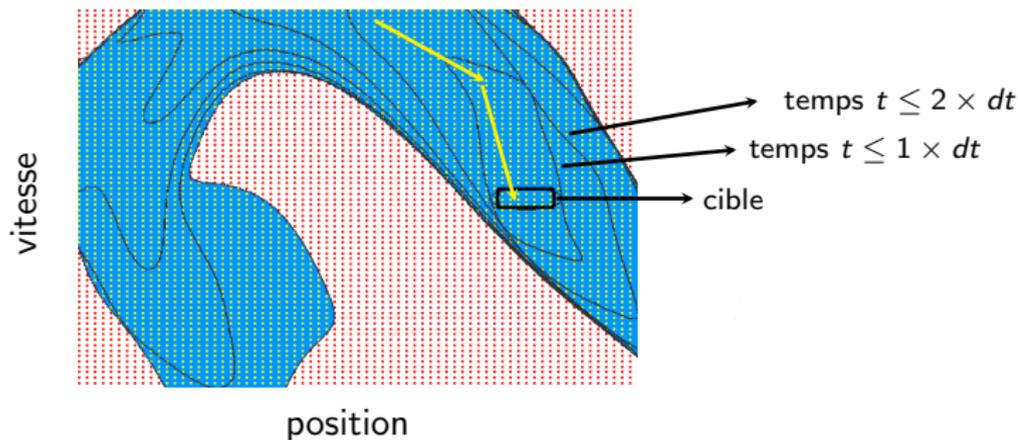


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

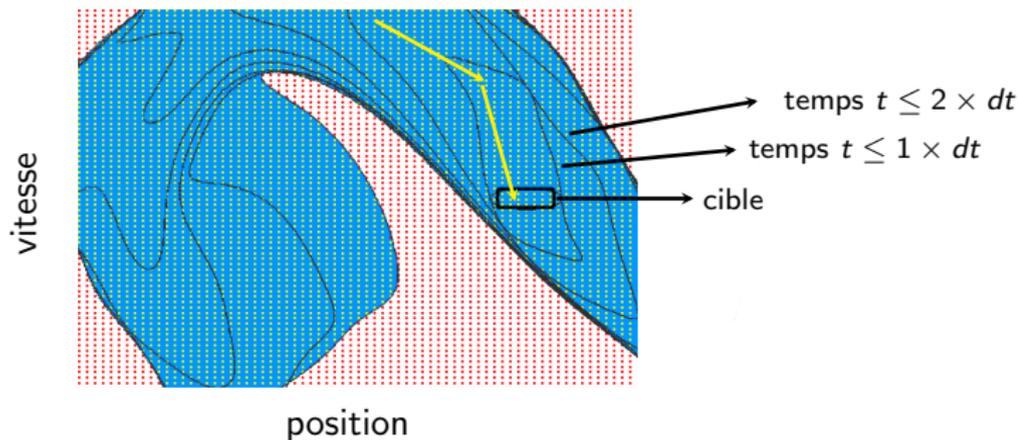


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

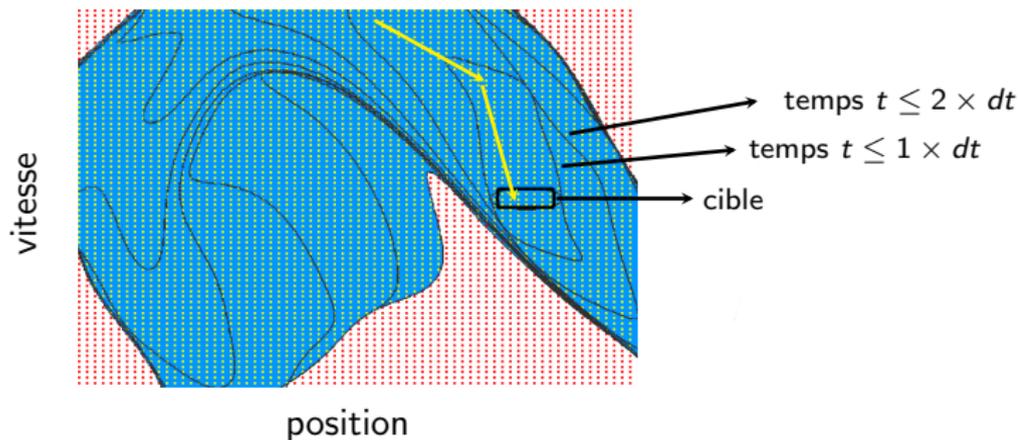


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

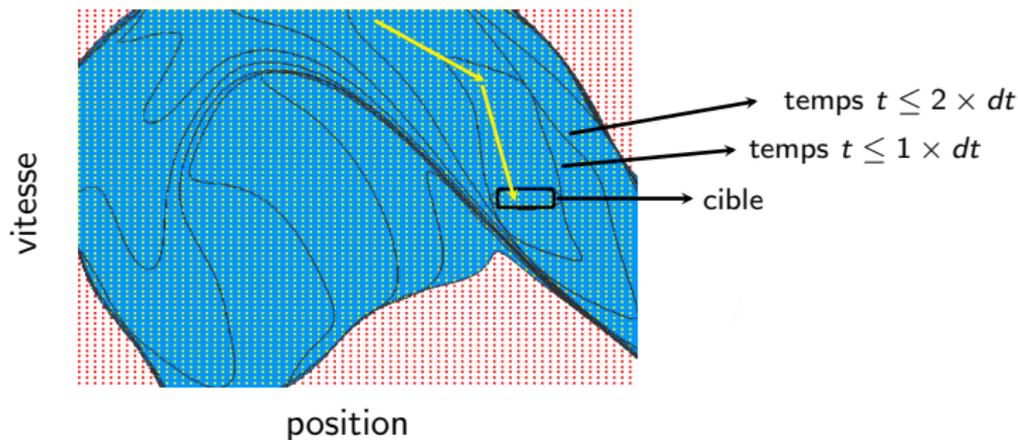


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

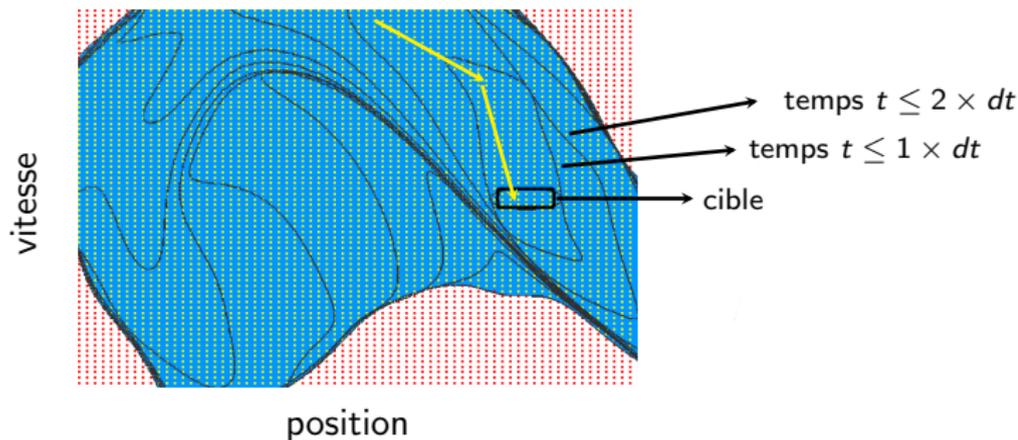


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

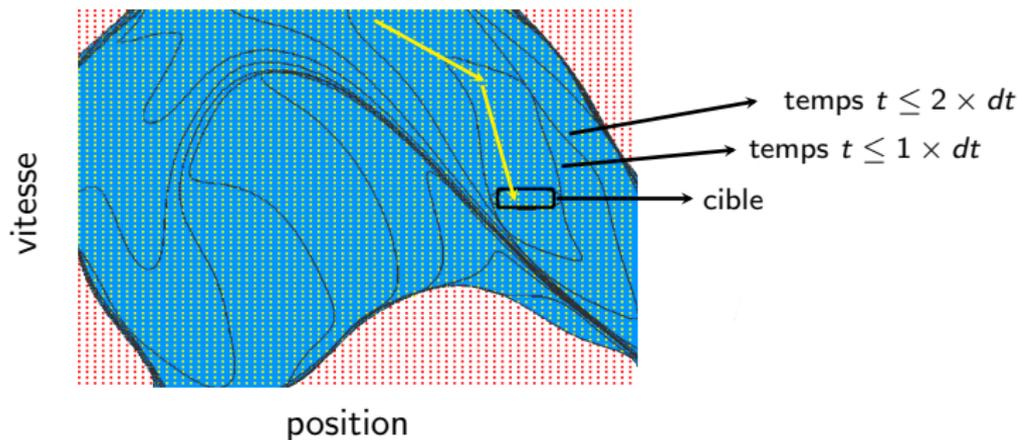


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

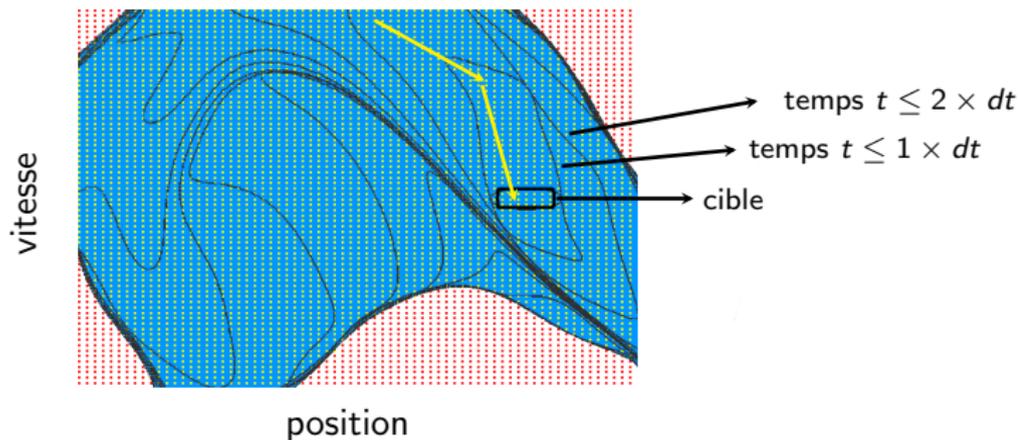


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

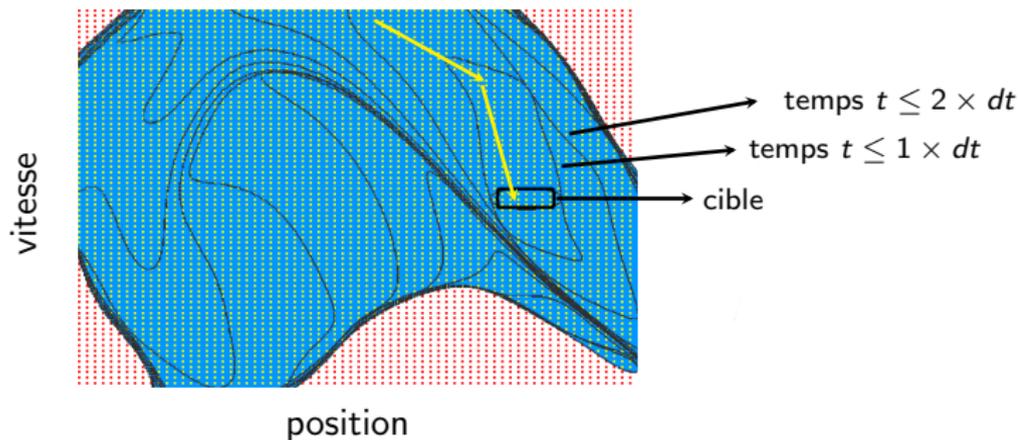


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

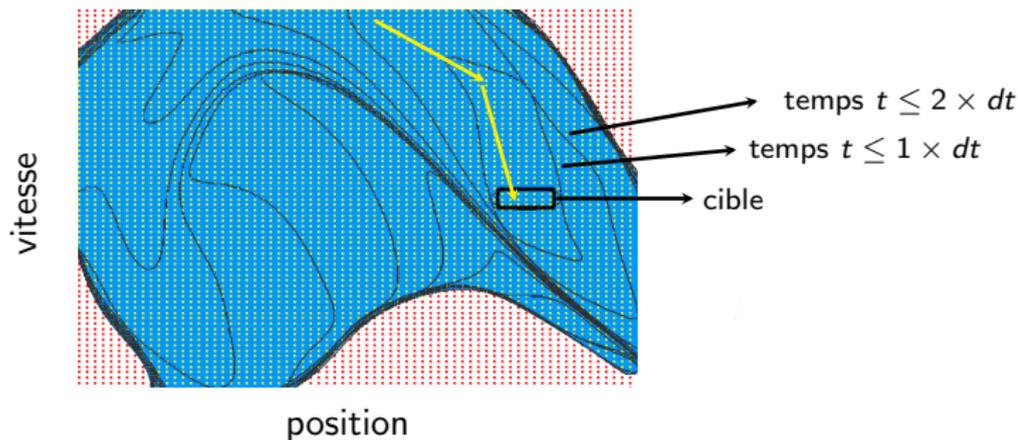


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps

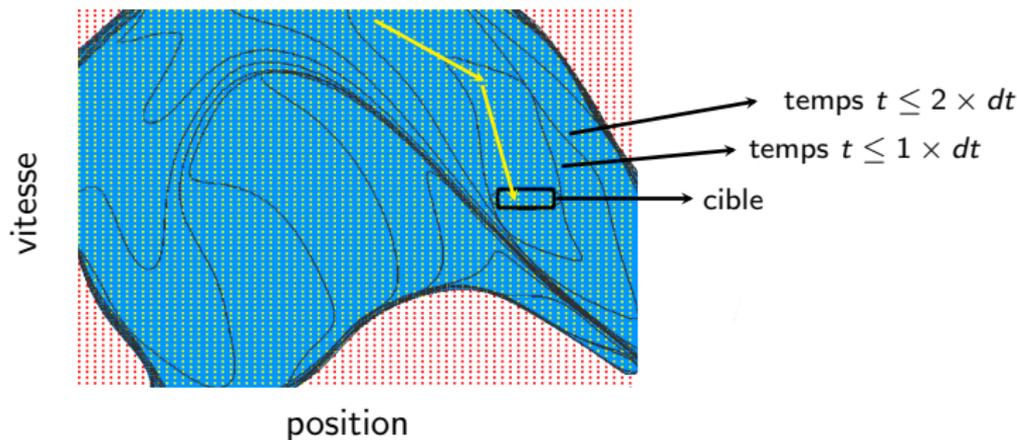


Approcher des bassins de capture avec des SVMs

Exemple d'application

Approximation progressive du bassin de capture par l'extérieur

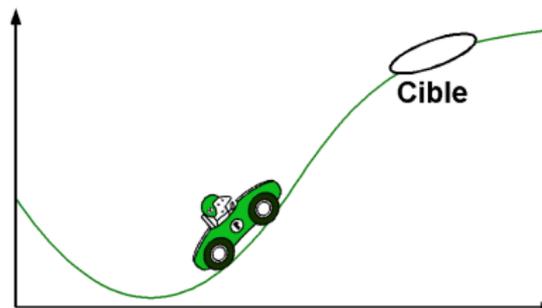
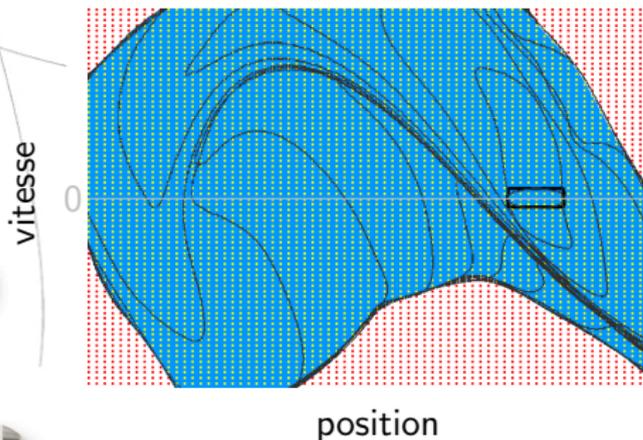
- Espace d'état en 2 dimensions, grille de 5041 points, 8 pas de temps



- Algorithme d'apprentissage actif de bassins de capture

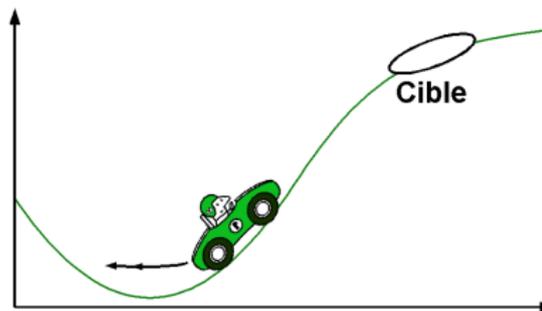
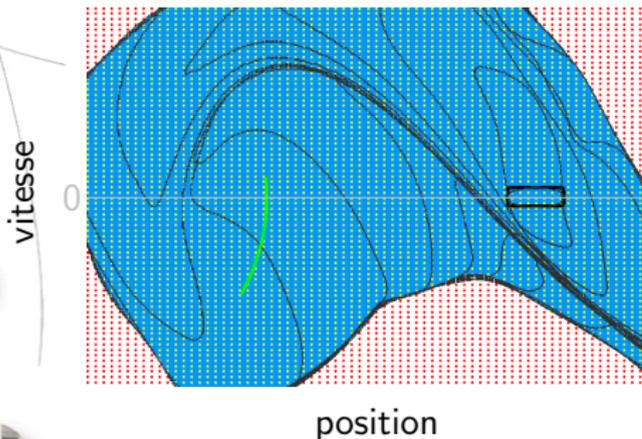
Contrôleur optimal

- Contrôleur optimal à partir de l'approximation par l'intérieur



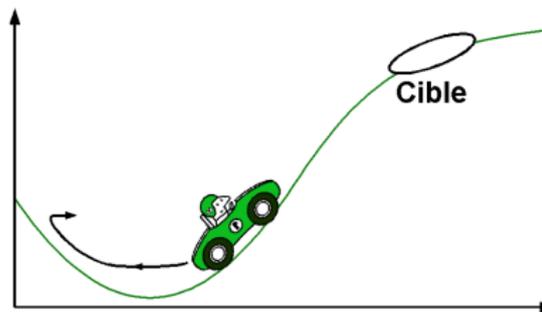
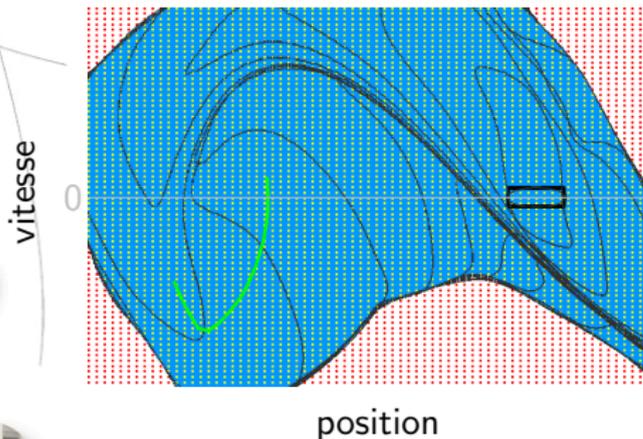
Contrôleur optimal

- Contrôleur optimal à partir de l'approximation par l'intérieur



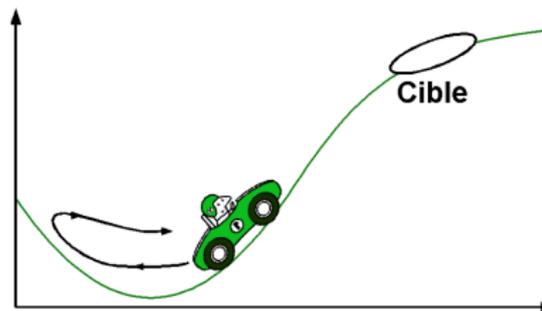
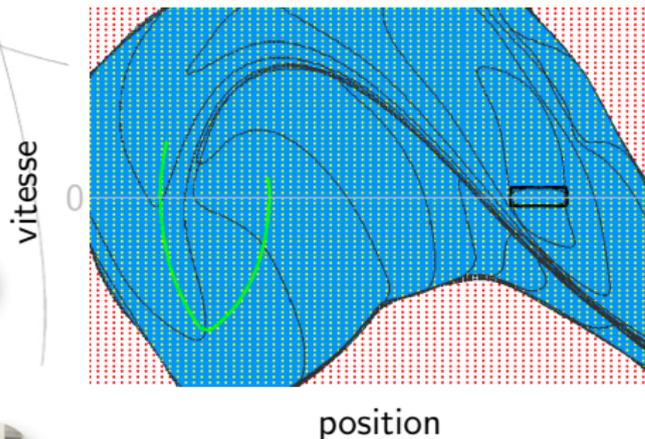
Contrôleur optimal

- Contrôleur optimal à partir de l'approximation par l'intérieur



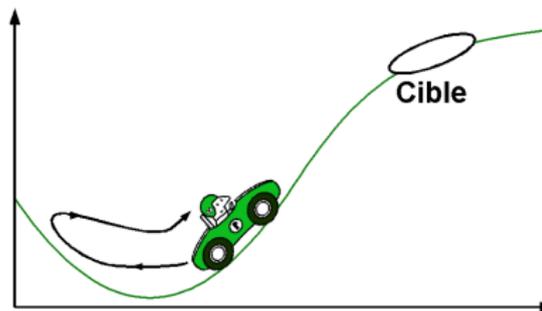
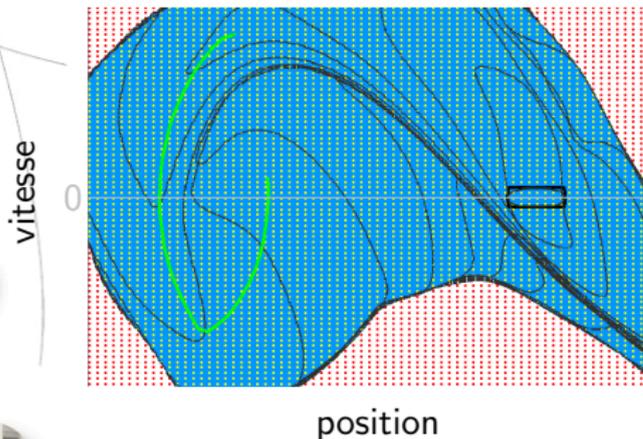
Contrôleur optimal

- Contrôleur optimal à partir de l'approximation par l'intérieur



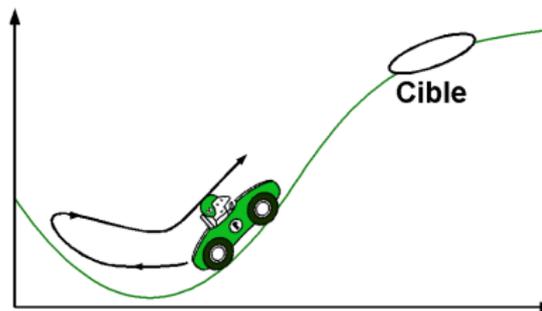
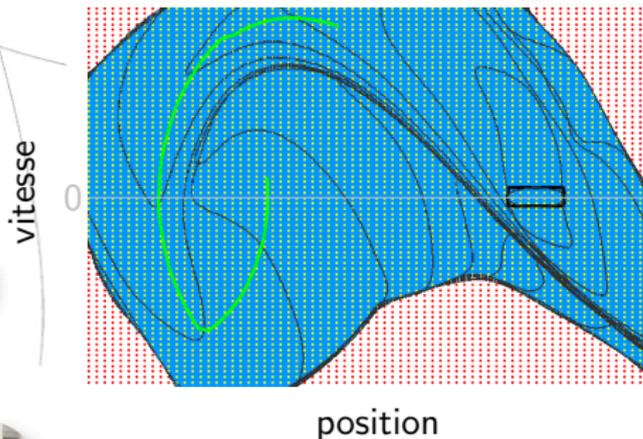
Contrôleur optimal

- Contrôleur optimal à partir de l'approximation par l'intérieur



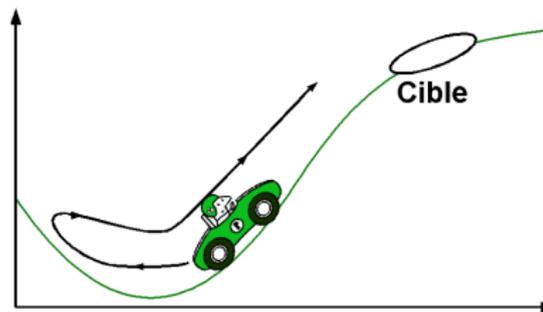
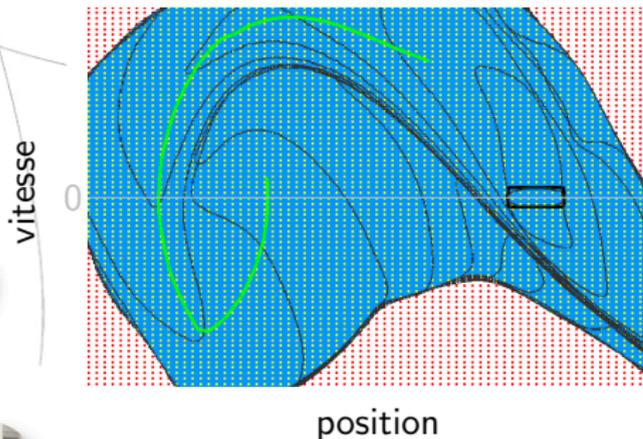
Contrôleur optimal

- Contrôleur optimal à partir de l'approximation par l'intérieur



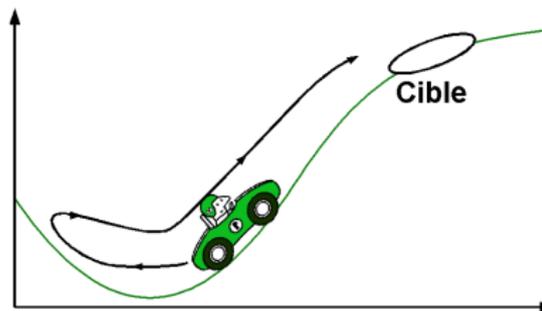
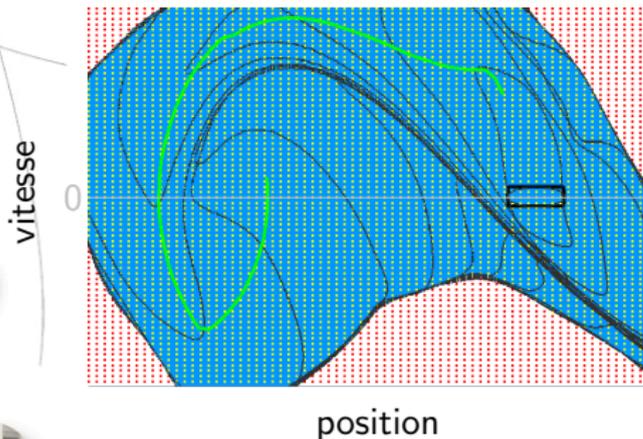
Contrôleur optimal

- Contrôleur optimal à partir de l'approximation par l'intérieur



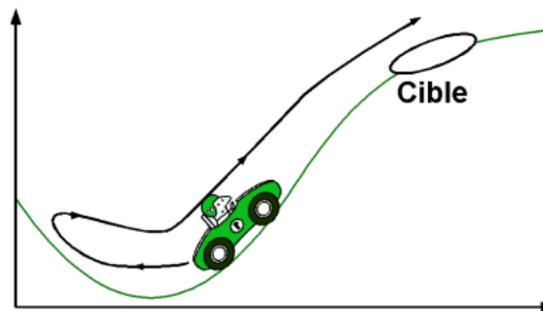
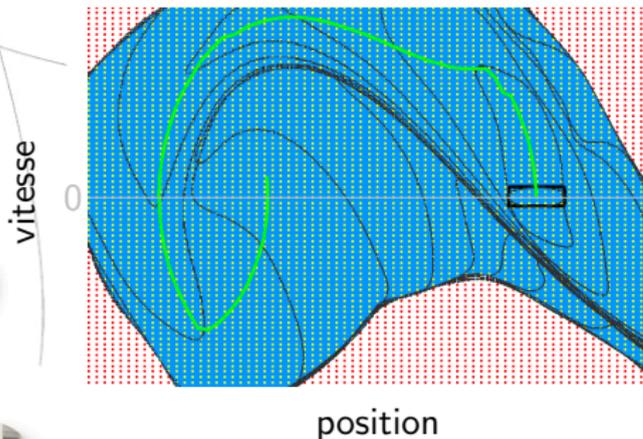
Contrôleur optimal

- Contrôleur optimal à partir de l'approximation par l'intérieur



Contrôleur optimal

- Contrôleur optimal à partir de l'approximation par l'intérieur



Extension au calcul de résilience

- Définition [Martin] : inverse du coût de restauration des propriétés d'intérêt perdues après une perturbation
- Propriétés d'intérêt : espace des contraintes
- Sous certaines conditions, on peut utiliser l'algorithme d'approximation de bassins de capture dans l'espace d'état initial
 - cible : noyau de viabilité du système
 - ensemble des états qui peuvent revenir dans la cible avec un coût $c \leq c_{max}$

Conclusion [Approximation de bassins de capture]

- Deux variantes de l'algorithme dans l'espace d'état initial
 - **permet** de définir un contrôleur qui garantit d'atteindre la cible
 - **permet** de travailler avec des contrôles en grande dimension
 - **ne permet pas** de traiter des problèmes avec des espaces d'état importants



1. Approcher des noyaux de viabilité avec des SVMs
2. Apprentissage actif de noyau de viabilité
3. Approcher des bassins de capture avec des SVMs
4. Conclusion générale



- 
- **Pari** : les SVMs pour rompre la malédiction de la dimensionnalité
 - permettent de représenter des formes complexes en grande dimension avec parcimonie
 - **Dans l'espace des contrôles** : les algorithmes proposés permettent de travailler dans des espaces en grande dimension (18)
 - **Dans l'espace d'état** : les procédures développées ne sont pas suffisamment décisives (6)
 - Définition de contrôleurs compacts et rapides



Perspectives

- Comparaison avec la programmation dynamique
- Étudier un algorithme d'approximation d'un domaine inclus dans le noyau de viabilité afin de garantir le contrôle
- Améliorer l'algorithme d'apprentissage actif

Maintenir la viabilité ou la résilience d'un système : les machines à vecteurs de support pour rompre la malédiction de la dimensionnalité ?

Laetitia Chapel

Directeur de thèse : Guillaume Deffuant
Laboratoire d'Ingénierie des Systèmes Complexes

Soutenance de thèse
19 octobre 2007

