Hamilton-Jacobi via les réseaux de neurones

Soutenance de stage ingénieur

Averil PROST

sous la direction de

Olivier BOKANOWSKI

INSTITUT NATIONAL DES SCIENCES APPLIQUÉES ROUEN NORMANDIE

Table of Contents

Problème d'intérêt

Schémas (semi-)lagrangiens

Réseaux de neurones

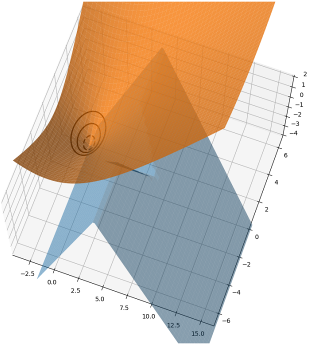
Résultat

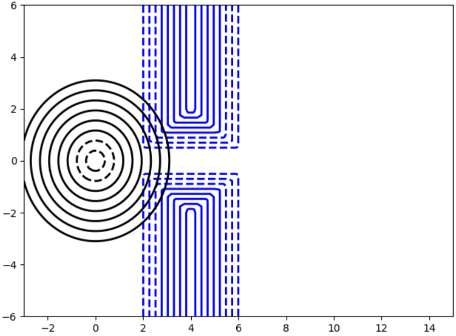
Exploration numérique

Problèmes de contrôle déterministes

Formulation level set

- une région à atteindre : $\varphi \leqslant 0$

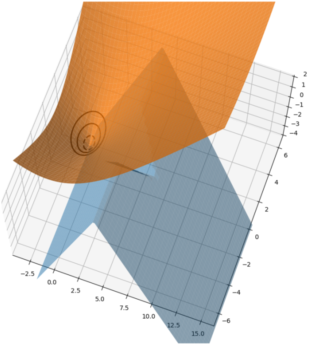


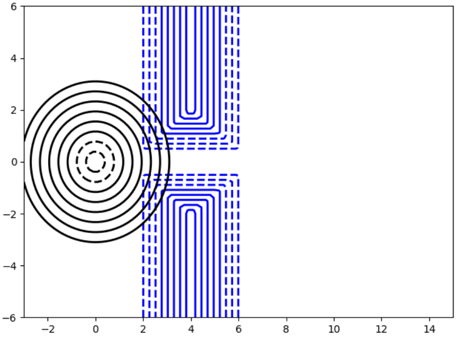


Problèmes de contrôle déterministes

Formulation level set

- une région à atteindre : $\varphi \leqslant 0$
- contraintes sur l'état (obstacle) : $g \leqslant 0$

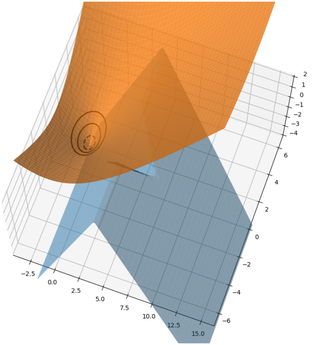


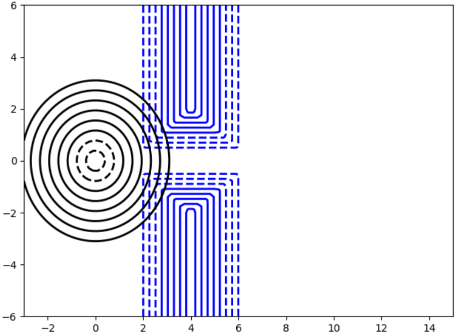


Problèmes de contrôle déterministes

Formulation level set

- une région à atteindre : $\varphi \leqslant 0$
- contraintes sur l'état (obstacle) : $g \leqslant 0$
- ensembles d'atteignabilité en temps t : $u(t,\cdot) \leqslant 0$

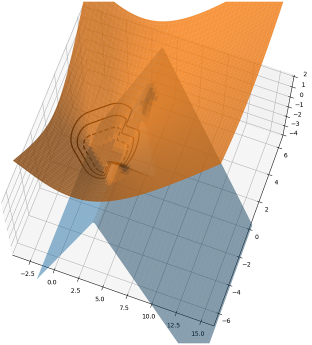


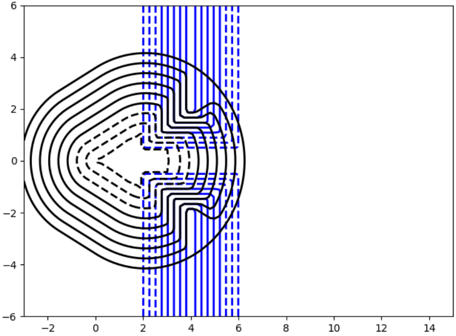


Problèmes de contrôle déterministes

Formulation level set

- une région à atteindre : $\varphi \leqslant 0$
- contraintes sur l'état (obstacle) : $g \leqslant 0$
- ensembles d'atteignabilité en temps t : $u(t, \cdot) \leqslant 0$

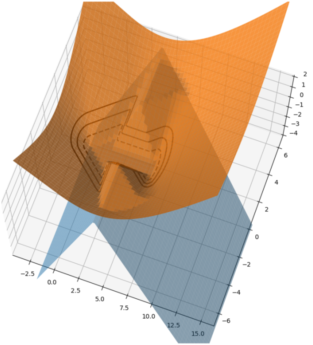


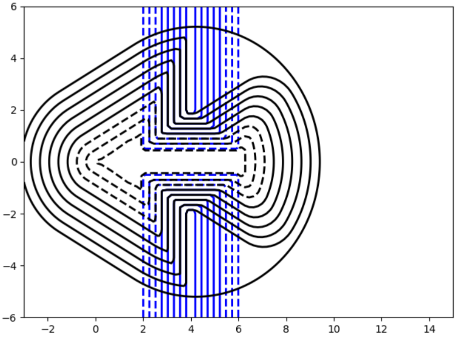


Problèmes de contrôle déterministes

Formulation level set

- une région à atteindre : $\varphi \leqslant 0$
- contraintes sur l'état (obstacle) : $g \leqslant 0$
- ensembles d'atteignabilité en temps t : $u(t, \cdot) \leqslant 0$

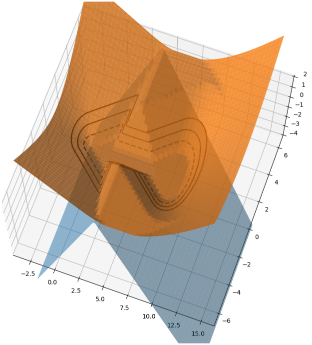


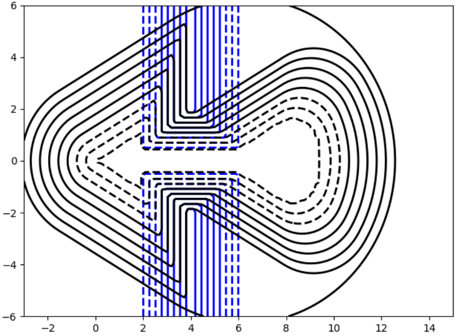


Problèmes de contrôle déterministes

Formulation level set

- une région à atteindre : $\varphi \leqslant 0$
- contraintes sur l'état (obstacle) : $g \leqslant 0$
- ensembles d'atteignabilité en temps t : $u(t,\cdot) \leqslant 0$

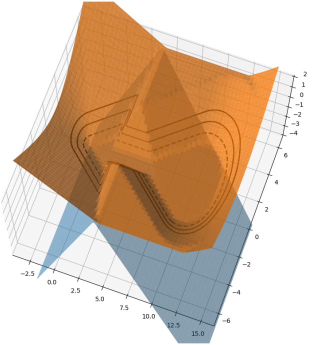


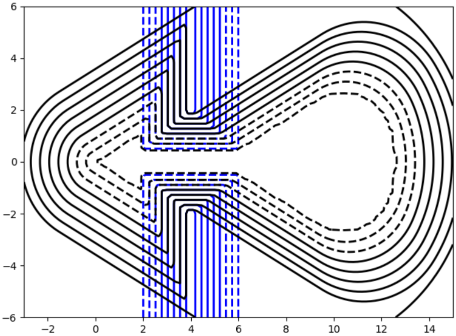


Problèmes de contrôle déterministes

Formulation level set

- une région à atteindre : $\varphi \leqslant 0$
- contraintes sur l'état (obstacle) : $g \leqslant 0$
- ensembles d'atteignabilité en temps t : $u(t, \cdot) \leqslant 0$





Problèmes de contrôle déterministes

Formulation level set

- une région à atteindre : $\varphi \leqslant 0$
- contraintes sur l'état (obstacle) : $g \leqslant 0$
- ensembles d'atteignabilité en temps t : $u(t, \cdot) \leqslant 0$

$$\min\left(\partial_t u + H\left(\nabla u\right), u - g\right) = 0, \quad u(T, \cdot) = \varphi \vee g.$$

Problèmes de contrôle déterministes

Formulation level set

- une région à atteindre : $\varphi \leqslant 0$
- contraintes sur l'état (obstacle) : $g \leqslant 0$
- ensembles d'atteignabilité en temps t : $u(t, \cdot) \leqslant 0$

$$\min\left(\partial_t u + H\left(\nabla u\right), u - g\right) = 0, \quad u(T, \cdot) = \varphi \vee g.$$

Comment s'attaquer au problème numérique :

- Méthodes sur maillages pleins (DF, VF, EF, GD...)
  Problème d'explosion de la dimension.

Problèmes de contrôle déterministes

Formulation level set

- une région à atteindre : $\varphi \leqslant 0$
- contraintes sur l'état (obstacle) : $g \leqslant 0$
- ensembles d'atteignabilité en temps t : $u(t, \cdot) \leqslant 0$

$$\min\left(\partial_t u + H\left(\nabla u\right), u - g\right) = 0, \quad u(T, \cdot) = \varphi \vee g.$$

Comment s'attaquer au problème numérique :

- Méthodes sur maillages pleins (DF, VF, EF, GD...)
  Problème d'explosion de la dimension.
- Méthodes par apprentissage / renforcement.

Cas d'étude

Problème d'obstacle ([ABZ13]) Trouver $u = u(t, x)$ t.q.

$$\begin{cases} \min\left(\partial_t u + \max_{a \in A} \nabla u \cdot f(x, a), u - g(x)\right) = 0, \\ u(T, x) = \varphi(x) \vee g(x). \end{cases}$$

Cas d'étude

Problème d'obstacle ([ABZ13]) Trouver $u = u(t, x)$ t.q.

$$\begin{cases} \min\left(\partial_t u + \max_{a \in A} \nabla u \cdot f(x, a), u - g(x)\right) = 0, \\ u(T, x) = \varphi(x) \vee g(x). \end{cases}$$

Exemples remarquables :

- $f(x, a) = b \in \mathbb{R}^n$ constant. Advection avec obstacle.

Cas d'étude

Problème d'obstacle ([ABZ13]) Trouver $u = u(t, x)$ t.q.

$$\begin{cases} \min\left(\partial_t u + \max_{a \in A} \nabla u \cdot f(x, a), u - g(x)\right) = 0, \\ u(T, x) = \varphi(x) \vee g(x). \end{cases}$$

Exemples remarquables :

- $f(x, a) = b \in \mathbb{R}^n$ constant. Advection avec obstacle.
- $f(x, a) = b(x)$. Notons $\dot{y}_x^a(\theta) = f(y_x^a(\theta), a(\theta))$, $y_x^a(0) := x$. Dès lors,

$$u(t, x) = \varphi(y_x(T - t)) \bigvee \max_{\theta \in [t, T]} g(y_x(\theta)).$$

Cas d'étude

Problème d'obstacle ([ABZ13]) Trouver $u = u(t,x)$ t.q.

$$\begin{cases} \min\left(\partial_t u + \max_{a\in A} \nabla u \cdot f(x,a), u - g(x)\right) = 0, \\ u(T,x) = \varphi(x) \vee g(x). \end{cases}$$

Exemples remarquables :

- $f(x,a) = b \in \mathbb{R}^n$ constant. Advection avec obstacle.
- $f(x,a) = b(x)$. Notons $\dot{y}_x^a(\theta) = f(y_x^a(\theta), a(\theta))$, $y_x^a(0) := x$. Dès lors,

$$u(t,x) = \varphi(y_x(T-t)) \bigvee \max_{\theta\in[t,T]} g(y_x(\theta)).$$

- $f(x,a) = a$, $g \equiv -\infty$: $u(t,x) = \min_{b\in\mathscr{B}(0,t)} \varphi(x - tb)$.

Table of Contents

Problème d'intérêt

Schémas (semi-)lagrangiens

Réseaux de neurones

Résultat

Exploration numérique

PPD

On s'appuie sur la version intégrale de l'équation :

Principe de programmation dynamique $\forall h \in [0, T-t]$,

$$u(t,x) = \inf_{a(\cdot)\in\mathbb{A}_{[t,t+h]}} u(t+h, y_x^a(h)) \bigvee \max_{\theta\in[0,h]} g(y_x^a(\theta))$$

PPD

On s'appuie sur la version intégrale de l'équation :

Principe de programmation dynamique $\forall h \in [0, T-t]$,

$$u(t,x) = \inf_{a(\cdot)\in\mathbb{A}_{[t,t+h]}} u(t+h, y_x^a(h)) \bigvee \max_{\theta\in[0,h]} g(y_x^a(\theta))$$

Premièrement, introduisons des contrôles *feedback* :

$$\alpha(\theta, x) := a(\theta), \quad a \text{ optimal pour le problème issu de } x \text{ en } t = \theta.$$

PPD

On s'appuie sur la version intégrale de l'équation :

Principe de programmation dynamique $\forall h \in [0, T-t]$,

$$u(t,x) = \inf_{a(\cdot) \in \mathbb{A}_{[t,t+h]}} u(t+h, y_x^a(h)) \bigvee \max_{\theta \in [0,h]} g(y_x^a(\theta))$$

Premièrement, introduisons des contrôles *feedback* :

$\alpha(\theta, x) := a(\theta)$, $\quad a$ optimal pour le problème issu de x en $t = \theta$.

- Peu réguliers en général (mesurables bornés si A compact).

PPD

On s'appuie sur la version intégrale de l'équation :

Principe de programmation dynamique $\forall h \in [0, T-t]$,

$$u(t,x) = \inf_{a(\cdot)\in\mathbb{A}_{[t,t+h]}} u(t+h, y_x^a(h)) \bigvee \max_{\theta\in[0,h]} g(y_x^a(\theta))$$

Premièrement, introduisons des contrôles *feedback* :

$$\alpha(\theta, x) \coloneqq a(\theta), \quad a \text{ optimal pour le problème issu de } x \text{ en } t = \theta.$$

- Peu réguliers en général (mesurables bornés si A compact).
- Peuvent ne pas être définis partout (pas d'unicité).

PPD

On s'appuie sur la version intégrale de l'équation :

Principe de programmation dynamique $\forall h \in [0, T-t]$,

$$u(t,x) = \inf_{a(\cdot)\in\mathbb{A}_{[t,t+h]}} u(t+h, y_x^a(h)) \bigvee \max_{\theta\in[0,h]} g(y_x^a(\theta))$$

Premièrement, introduisons des contrôles *feedback* :

$\alpha(\theta, x) \coloneqq a(\theta)$, a optimal pour le problème issu de x en $t = \theta$.

- Peu réguliers en général (mesurables bornés si A compact).
- Peuvent ne pas être définis partout (pas d'unicité).
- Si $\alpha(\cdot,\cdot) \in \mathcal{A}_{[t,t+h]} \coloneqq L^\infty([0,T], A)$, problèmes équivalents.

Minimisation sur un espace

Soit μ une mesure ne s'annulant nulle part telle que $\mu(\Omega) < \infty$, Ω le domaine spatial considéré. Nouveau problème équivalent :

$$\begin{cases} J(t,x,\alpha) := u(t+h, y_x^a(h)) \bigvee \max_{\theta\in[0,h]} g(y_x^a(\theta)) \\ \alpha^* \in \operatorname{argmin}_{\alpha\in\mathcal{A}_{[t,t+h]}} \int_{x\in\Omega} J(t,x,\alpha)\mu(dx), \\ u(t,x) := J(t,x,\alpha^*). \end{cases}$$

Minimisation sur un espace

Soit μ une mesure ne s'annulant nulle part telle que $\mu(\Omega) < \infty$, Ω le domaine spatial considéré. Nouveau problème équivalent :

$$\begin{cases} J(t,x,\alpha) := u(t+h, y_x^a(h)) \bigvee \max_{\theta \in [0,h]} g(y_x^a(\theta)) \\ \alpha^* \in \operatorname{argmin}_{\alpha \in \mathcal{A}_{[t,t+h]}} \int_{x \in \Omega} J(t,x,\alpha)\mu(dx), \\ u(t,x) := J(t,x,\alpha^*). \end{cases}$$

Discrétisation en deux étapes :

- Fonctionnelle : restriction des contrôles à un espace de dimension finie $\hat{A} \subset \mathcal{A}$.

Minimisation sur un espace

Soit μ une mesure ne s'annulant nulle part telle que $\mu(\Omega) < \infty$, Ω le domaine spatial considéré. Nouveau problème équivalent :

$$\begin{cases} J(t,x,\alpha) \coloneqq u(t+h, y_x^a(h)) \bigvee \max_{\theta \in [0,h]} g(y_x^a(\theta)) \\ \alpha^* \in \operatorname{argmin}_{\alpha \in \mathcal{A}_{[t,t+h]}} \int_{x\in\Omega} J(t,x,\alpha)\mu(dx), \\ u(t,x) \coloneqq J(t,x,\alpha^*). \end{cases}$$

Discrétisation en deux étapes :

- Fonctionnelle : restriction des contrôles à un espace de dimension finie $\hat{A} \subset \mathcal{A}$.
- Temporelle : choix de $h = \Delta t$, et calcul de $u^n(\cdot) \simeq u(n\Delta t, \cdot)$ à partir de u^{n+1} par récurrence.

Schéma lagrangien

Soit $\hat{\mathcal{A}}$ un espace de dimension finie (EF, base spectrale, réseaux de neurones...), $N \coloneqq T/\Delta t$, et $M \in \mathbb{N}^*$. On pose $\hat{y}_x^\alpha(\Delta t) \simeq y_x^\alpha(\Delta t)$.

Schéma lagrangien

Soit $\hat{\mathcal{A}}$ un espace de dimension finie (EF, base spectrale, réseaux de neurones...), $N \coloneqq T/\Delta t$, et $M \in \mathbb{N}^*$. On pose $\hat{y}_x^\alpha(\Delta t) \simeq y_x^\alpha(\Delta t)$.

Choisir $\hat{u}^N \coloneqq \varphi \vee g$.
for $n \in [\![N-1, 0]\!]$ **do**
    Tirer $(X_k)_k$ M réalisations iid de $X \sim \mu$.
    Choisir $\hat{\alpha}^n \in \operatorname{argmin}_{\hat{\alpha} \in \hat{\mathcal{A}}} \sum_{i=1}^{M} \hat{u}^{n+1}(\hat{y}_{X_i}^{\hat{\alpha}}(\Delta t)) \vee g(X_i)$.
    Définir $\hat{u}^n(x) \coloneqq \hat{u}^{n+1}(\hat{y}_x^{\hat{\alpha}^n}(\Delta t)) \vee g(x)$.

Schéma lagrangien

Soit $\hat{\mathcal{A}}$ un espace de dimension finie (EF, base spectrale, réseaux de neurones...), $N := T/\Delta t$, et $M \in \mathbb{N}^*$. On pose $\hat{y}^\alpha_x(\Delta t) \simeq y^\alpha_x(\Delta t)$.

Choisir $\hat{u}^N := \varphi \vee g$.
for $n \in [\![N-1, 0]\!]$ **do**
  Tirer $(X_k)_k$ M réalisations iid de $X \sim \mu$.
  Choisir $\hat{\alpha}^n \in \operatorname{argmin}_{\hat{\alpha} \in \hat{\mathcal{A}}} \sum_{i=1}^M \hat{u}^{n+1}(\hat{y}^{\hat{\alpha}}_{X_i}(\Delta t)) \vee g(X_i)$.
  Définir $\hat{u}^n(x) := \hat{u}^{n+1}(\hat{y}^{\hat{\alpha}^n}_x(\Delta t)) \vee g(x)$.

- La variable u^n est abstraite : calculée à partir des $(\hat{\alpha}^k)_{k \geqslant n}$.

Schéma lagrangien

Soit $\hat{\mathcal{A}}$ un espace de dimension finie (EF, base spectrale, réseaux de neurones...), $N := T/\Delta t$, et $M \in \mathbb{N}^*$. On pose $\hat{y}_x^\alpha(\Delta t) \simeq y_x^\alpha(\Delta t)$.

---

1 Choisir $\hat{u}^N := \varphi \vee g$.
2 **for** $n \in [\![N-1, 0]\!]$ **do**
3   Tirer $(X_k)_k$ M réalisations iid de $X \sim \mu$.
4   Choisir $\hat{\alpha}^n \in \operatorname{argmin}_{\hat{\alpha} \in \hat{\mathcal{A}}} \sum_{i=1}^M \hat{u}^{n+1}(\hat{y}_{X_i}^{\hat{\alpha}}(\Delta t)) \vee g(X_i)$.
5   Définir $\hat{u}^n(x) := \hat{u}^{n+1}(\hat{y}_x^{\hat{\alpha}^n}(\Delta t)) \vee g(x)$.

- La variable u^n est abstraite : calculée à partir des $(\hat{\alpha}^k)_{k \geqslant n}$.
- On peut raffiner en considérant $\max\limits_{\theta \in \Theta} g(\hat{y}_{X_i}^{\hat{\alpha}}(\theta))$, $\Theta \subset [0, \Delta t]$.

Schéma lagrangien

Soit $\hat{\mathcal{A}}$ un espace de dimension finie (EF, base spectrale, réseaux de neurones...), $N := T/\Delta t$, et $M \in \mathbb{N}^*$. On pose $\hat{y}_x^\alpha(\Delta t) \simeq y_x^\alpha(\Delta t)$.

---

1 Choisir $\hat{u}^N := \varphi \vee g$.
2 **for** $n \in [\![N-1, 0]\!]$ **do**
3 Tirer $(X_k)_k$ M réalisations iid de $X \sim \mu$.
4 Choisir $\hat{\alpha}^n \in \operatorname{argmin}_{\hat{\alpha}\in\hat{\mathcal{A}}} \sum_{i=1}^M \hat{u}^{n+1}(\hat{y}_{X_i}^{\hat{\alpha}}(\Delta t)) \vee g(X_i)$.
5 Définir $\hat{u}^n(x) := \hat{u}^{n+1}(\hat{y}_x^{\hat{\alpha}^n}(\Delta t)) \vee g(x)$.

- La variable u^n est abstraite : calculée à partir des $(\hat{\alpha}^k)_{k \geqslant n}$.
- On peut raffiner en considérant $\max_{\theta\in\Theta} g(\hat{y}_{X_i}^{\hat{\alpha}}(\theta))$, $\Theta \subset [0, \Delta t]$.
- Complexité quadratique.

Schéma semi-lagrangien

Ajout d'une variable $\hat{U}^n$ en mémoire, utilisée pour l'étape de minimisation. Soit $\hat{\mathcal{U}}$ un espace de dim finie.

Schéma semi-lagrangien

Ajout d'une variable $\hat{U}^n$ en mémoire, utilisée pour l'étape de minimisation. Soit $\hat{\mathcal{U}}$ un espace de dim finie.

1 Choisir $\hat{U}^N := \mathbb{P}_{\hat{\mathcal{U}}}(\varphi \vee g)$.

2 **for** $n \in [\![N-1, 0]\!]$ **do**

3 Tirer $(X_k)_k$ M réalisations iid de $X \sim \mu$.

4 Choisir $\hat{\alpha}^n \in \operatorname{argmin}_{\hat{\alpha} \in \hat{\mathcal{A}}} \sum_{i=1}^{M} \hat{U}^{n+1}(\hat{y}_{X_i}^{\hat{\alpha}}(\Delta t)) \vee g(X_i)$.

5 Définir $\hat{U}^n(x) := \mathbb{P}_{\hat{\mathcal{U}}} \left\{ \hat{U}^{n+1}(\hat{y}_x^{\hat{\alpha}^n}(\Delta t)) \vee g(x) \right\}$.

Schéma semi-lagrangien

Ajout d'une variable $\hat{U}^n$ en mémoire, utilisée pour l'étape de minimisation. Soit $\hat{\mathcal{U}}$ un espace de dim finie.

---

1 Choisir $\hat{U}^N := \mathbb{P}_{\hat{\mathcal{U}}}(\varphi \vee g)$.
2 **for** $n \in [\![N-1, 0]\!]$ **do**
3 Tirer $(X_k)_k$ M réalisations iid de $X \sim \mu$.
4 Choisir $\hat{\alpha}^n \in \operatorname{argmin}_{\hat{\alpha} \in \hat{\mathcal{A}}} \sum_{i=1}^{M} \hat{U}^{n+1}(\hat{y}_{X_i}^{\hat{\alpha}}(\Delta t)) \vee g(X_i)$.
5 Définir $\hat{U}^n(x) := \mathbb{P}_{\hat{\mathcal{U}}} \left\{ \hat{U}^{n+1}(\hat{y}_x^{\hat{\alpha}^n}(\Delta t)) \vee g(x) \right\}$.

- Complexité linéaire (coût mémoire et calcul/étape doublé).

Schéma semi-lagrangien

Ajout d'une variable $\hat{U}^n$ en mémoire, utilisée pour l'étape de minimisation. Soit $\hat{\mathcal{U}}$ un espace de dim finie.

1 Choisir $\hat{U}^N := \mathbb{P}_{\hat{\mathcal{U}}}(\varphi \vee g)$.
2 **for** $n \in [\![N-1, 0]\!]$ **do**
3 Tirer $(X_k)_k$ M réalisations iid de $X \sim \mu$.
4 Choisir $\hat{\alpha}^n \in \operatorname{argmin}_{\hat{\alpha} \in \hat{\mathcal{A}}} \sum_{i=1}^{M} \hat{U}^{n+1}(\hat{y}_{X_i}^{\hat{\alpha}}(\Delta t)) \vee g(X_i)$.
5 Définir $\hat{U}^n(x) := \mathbb{P}_{\hat{\mathcal{U}}} \left\{ \hat{U}^{n+1}(\hat{y}_x^{\hat{\alpha}^n}(\Delta t)) \vee g(x) \right\}$.

- Complexité linéaire (coût mémoire et calcul/étape doublé).
- Diffusion liée à l'étape de projection.

Schéma semi-lagrangien

Ajout d'une variable $\hat{U}^n$ en mémoire, utilisée pour l'étape de minimisation. Soit $\hat{\mathcal{U}}$ un espace de dim finie.

Choisir $\hat{U}^N := \mathbb{P}_{\hat{\mathcal{U}}}(\varphi \vee g)$.
for $n \in [\![N-1, 0]\!]$ **do**
  Tirer $(X_k)_k$ M réalisations iid de $X \sim \mu$.
  Choisir $\hat{\alpha}^n \in \operatorname{argmin}_{\hat{\alpha} \in \hat{\mathcal{A}}} \sum_{i=1}^{M} \hat{U}^{n+1}(\hat{y}_{X_i}^{\hat{\alpha}}(\Delta t)) \vee g(X_i)$.
  Définir $\hat{U}^n(x) := \mathbb{P}_{\hat{\mathcal{U}}}\left\{\hat{U}^{n+1}(\hat{y}_x^{\hat{\alpha}^n}(\Delta t)) \vee g(x)\right\}$.

- Complexité linéaire (coût mémoire et calcul/étape doublé).
- Diffusion liée à l'étape de projection.
- Algorithme hybride en projetant une évaluation lagrangienne.

Table of Contents

Problème d'intérêt

Schémas (semi-)lagrangiens

Réseaux de neurones

Résultat

Exploration numérique

Définition

$$\mathscr{R}(x) = \sigma_1 \circ L_1 \circ \cdots \circ \sigma_p \circ L_p, \quad L_i \text{ affine, } \sigma_i \text{ non linéaire.}$$

- Historiquement, $\sigma \simeq \frac{1}{1+\exp(-x)}$ la fonction logistique : classificateurs.

Définition

$$\mathscr{R}(x) = \sigma_1 \circ L_1 \circ \cdots \circ \sigma_p \circ L_p, \quad L_i \text{ affine, } \sigma_i \text{ non linéaire.}$$

- Historiquement, $\sigma \simeq \frac{1}{1+\exp(-x)}$ la fonction logistique : classificateurs.
- Activation $\sigma(x) = \max(0, x)$ (ReLu) populaire pour l'approximation de fonctions.

Définition

$$\mathscr{R}(x) = \sigma_1 \circ L_1 \circ \cdots \circ \sigma_p \circ L_p, \quad L_i \text{ affine, } \sigma_i \text{ non linéaire.}$$

- Historiquement, $\sigma \simeq \frac{1}{1+\exp(-x)}$ la fonction logistique : classificateurs.
- Activation $\sigma(x) = \max(0, x)$ (ReLu) populaire pour l'approximation de fonctions.
- En général, classe de fonctions de $\mathbb{R}^m \mapsto \mathbb{R}^p$ héritant leur régularité de σ (continuité, voire caractère lipschitz avec GroupSort).

Densité

Soit $\Omega \subset \mathbb{R}$ compact.

Squasher $\lim_{x \to -\infty} \sigma(x) = 0$, σ croissante et $\lim_{x \to \infty} \sigma(x) = 1$.

Densité

Soit $\Omega \subset \mathbb{R}$ compact.

Squasher $\lim_{x \to -\infty} \sigma(x) = 0$, σ croissante et $\lim_{x \to \infty} \sigma(x) = 1$.

Théorème – Théorème d'approximation universelle (Lemma 16.1 de [GKKW02]) L'espace des réseaux de neurones est dense dans $(\mathcal{C}(\Omega, \mathbb{R}^p), |\cdot|_\infty)$.

Densité

Soit $\Omega \subset \mathbb{R}$ compact.

Squasher $\lim_{x \to -\infty} \sigma(x) = 0$, σ croissante et $\lim_{x \to \infty} \sigma(x) = 1$.

Théorème – Théorème d'approximation universelle (Lemma 16.1 de [GKKW02]) L'espace des réseaux de neurones est dense dans $(\mathcal{C}(\Omega, \mathbb{R}^p), |\cdot|_\infty)$.

- Existence d'estimations, en $|\cdot|_\infty$ pour la dim 1, en norme $|\cdot|_{L^2}$ pour le cas général (voir Table 1. de [TSB20]).

Densité

Soit $\Omega \subset \mathbb{R}$ compact.

Squasher $\lim_{x\to-\infty} \sigma(x) = 0$, σ croissante et $\lim_{x\to\infty} \sigma(x) = 1$.

Théorème – Théorème d'approximation universelle (Lemma 16.1 de [GKKW02]) L'espace des réseaux de neurones est dense dans $(\mathcal{C}(\Omega, \mathbb{R}^p), |\cdot|_\infty)$.

- Existence d'estimations, en $|\cdot|_\infty$ pour la dim 1, en norme $|\cdot|_{L^2}$ pour le cas général (voir Table 1. de [TSB20]).
- Fracture entre lesdites estimations (qui explosent) et l'efficacité constatée des réseaux.

Table of Contents

Problème d'intérêt

Schémas (semi-)lagrangiens

Réseaux de neurones

Résultat

Exploration numérique

Hypothèses

Considérons le cas Ω borné. On suppose une certaine régularité des données du problème :

Hypothèses

Considérons le cas Ω borné. On suppose une certaine régularité des données du problème :

- Les contrôles sont à valeurs dans un compact convexe $A \subset \mathbb{R}^p$.

Hypothèses

Considérons le cas Ω borné. On suppose une certaine régularité des données du problème :

- Les contrôles sont à valeurs dans un compact convexe $A \subset \mathbb{R}^p$.
- Les fonctions f, g et φ sont lipschitz-continues.

Hypothèses

Considérons le cas Ω borné. On suppose une certaine régularité des données du problème :

- Les contrôles sont à valeurs dans un compact convexe $A \subset \mathbb{R}^p$.
- Les fonctions f, g et φ sont lipschitz-continues.
- Les densités d'entraînement μ_n sont bornées par $0 < \nu \leqslant \mu_n \leqslant C$.

Hypothèses

Considérons le cas Ω borné. On suppose une certaine régularité des données du problème :

- Les contrôles sont à valeurs dans un compact convexe $A \subset \mathbb{R}^p$.
- Les fonctions f, g et φ sont lipschitz-continues.
- Les densités d'entraînement μ_n sont bornées par $0 < \nu \leqslant \mu_n \leqslant C$.

On suppose de plus que le support des densités μ_n "suit" l'évolution de la dynamique, au sens où $\forall \omega \subset \Omega$ ouvert, $y_\omega^{a^*}(T - t_n) \in \{\varphi \leqslant 0\}$ implique $\mu_n(\omega) > 0$.

Énoncé

Supposons $N \in \mathbb{N}$ fixé. Soit Θ la dimension de l'espace d'approximation des contrôles.

Proposition Soit $(X_n)_n$ une suite de variables aléatoires de lois μ_n.

$$\lim_{\Theta \to \infty} \inf_{\hat{a} \in \hat{\mathcal{A}}_\Theta} \mathbb{E}[|\hat{V}_n(X_n) - V_n(X_n)|] = 0$$

Idée de la preuve

Pas de régularité sur le contrôle : on pose a_n^ε la régularisation de a_n^* par convolution. Décomposition de $\hat{V}_n - V_n$ par récurrence :

$$0 \leqslant \mathbb{E}(\hat{V}_n - V_n) \leqslant C_1 \inf_{\hat{a} \in \hat{\mathcal{A}}} \mathbb{E}\left|a_n^\varepsilon - a_n^*\right| \quad \text{erreur de régularisation}$$

$$\dots$$

Idée de la preuve

Pas de régularité sur le contrôle : on pose a_n^ε la régularisation de a_n^* par convolution. Décomposition de $\hat{V}_n - V_n$ par récurrence :

$$
\begin{aligned}
0 \leqslant \mathbb{E}(\hat{V}_n - V_n) \leqslant\ & C_1 \inf_{\hat{a} \in \hat{\mathcal{A}}} \mathbb{E}\left|a_n^\varepsilon - a_n^*\right| \quad \text{erreur de régularisation} \\
& + C_2 \inf_{\hat{a} \in \hat{\mathcal{A}}} \mathbb{E}\left|\hat{a} - a_n^\varepsilon\right| \quad \text{approximation de } \mathcal{A} \text{ par } \hat{\mathcal{A}} \\
& \ldots
\end{aligned}
$$

Idée de la preuve

Pas de régularité sur le contrôle : on pose a_n^ε la régularisation de a_n^* par convolution. Décomposition de $\hat{V}_n - V_n$ par récurrence :

$$
\begin{aligned}
0 \leqslant \mathbb{E}(\hat{V}_n - V_n) \leqslant {} & C_1 \inf_{\hat{a} \in \hat{\mathcal{A}}} \mathbb{E}\,|a_n^\varepsilon - a_n^*| && \text{erreur de régularisation} \\
& + C_2 \inf_{\hat{a} \in \hat{\mathcal{A}}} \mathbb{E}\,|\hat{a} - a_n^\varepsilon| && \text{approximation de } \mathcal{A} \text{ par } \hat{\mathcal{A}} \\
& + C_3\, \mathbb{E}\left(\hat{V}_{n+1} - V_{n+1}\right) && \text{terme de récurrence}
\end{aligned}
$$

Idée de la preuve

Pas de régularité sur le contrôle : on pose a_n^ε la régularisation de a_n^* par convolution. Décomposition de $\hat{V}_n - V_n$ par récurrence :

$$
\begin{aligned}
0 \leqslant \mathbb{E}(\hat{V}_n - V_n) \leqslant\ & C_1 \inf_{\hat{a} \in \hat{\mathcal{A}}} \mathbb{E}\, |a_n^\varepsilon - a_n^*| && \text{erreur de régularisation} \\
& + C_2 \inf_{\hat{a} \in \hat{\mathcal{A}}} \mathbb{E}\, |\hat{a} - a_n^\varepsilon| && \text{approximation de } \mathcal{A} \text{ par } \hat{\mathcal{A}} \\
& + C_3\, \mathbb{E}\left(\hat{V}_{n+1} - V_{n+1}\right) && \text{terme de récurrence}
\end{aligned}
$$

En laissant $\Theta \to \infty$, contrôle de l'explosion de C_3 quand $\varepsilon \searrow 0$. D'où convergence à N fixé.

Table of Contents

Problème d'intérêt

Schémas (semi-)lagrangiens

Réseaux de neurones

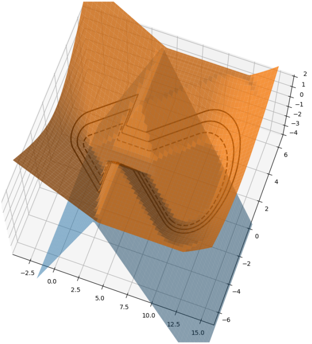
Résultat

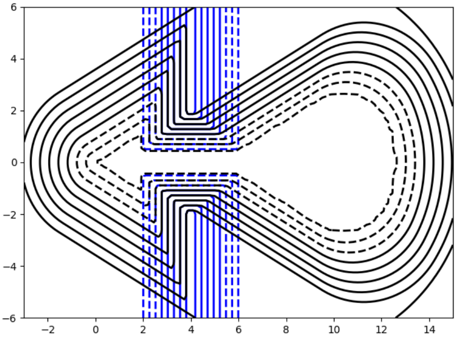
Exploration numérique

Équation eikonale-advection avec obstacle

Soit $A := \mathscr{B}_{\mathbb{R}^n}(0,1)$, $b \in \mathbb{R}^n$, $c \geqslant 0$. Trouver $u = u(t,x)$ t. q.

$$\min(-\partial_t u + \max_{a \in A} \nabla u \cdot [b + ca], u - g) = 0$$





Un exemple de simulation

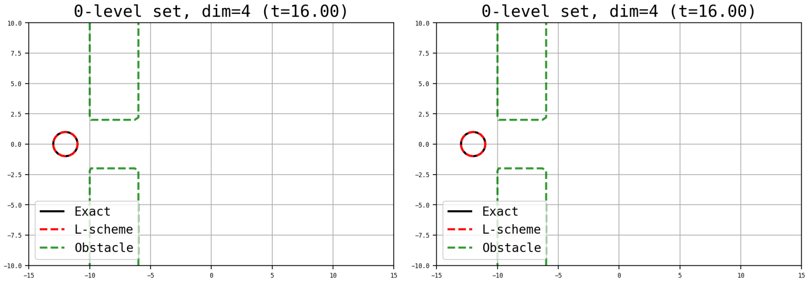


Figure – Gauche : $N_{\text{it}} = 8$, droite : $N_{\text{it}} = 16$, schéma lagrangien.

Paramètres : $b = e_1$, $c = 1/2$, 3 couches de 60 neurones, activation ReLu, 10^5 itérations de G.S. avec 4000 points.

Un exemple de simulation

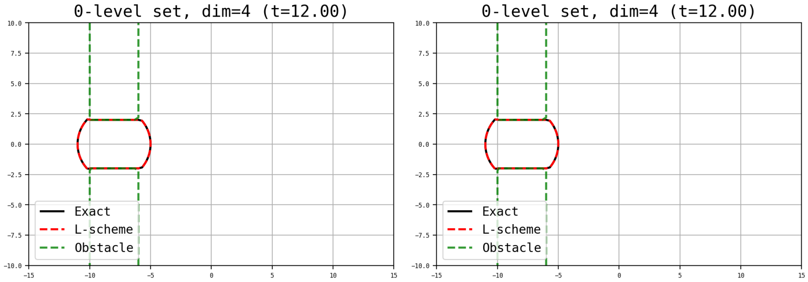


Figure – Gauche : $N_{\text{it}} = 8$, droite : $N_{\text{it}} = 16$, schéma lagrangien.

Paramètres : $b = e_1$, $c = 1/2$, 3 couches de 60 neurones, activation ReLu, 10^5 itérations de G.S. avec 4000 points.

Un exemple de simulation

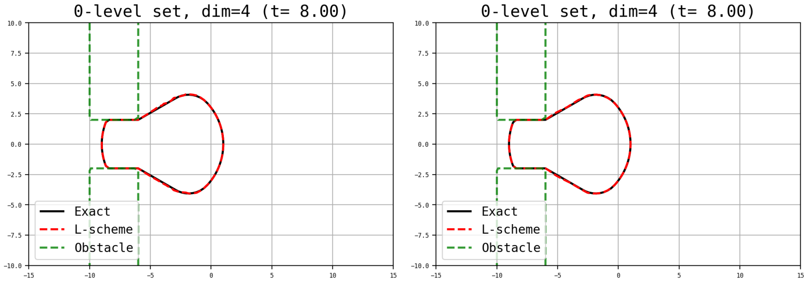


Figure – Gauche : $N_{\text{it}} = 8$, droite : $N_{\text{it}} = 16$, schéma lagrangien.

Paramètres : $b = e_1$, $c = 1/2$, 3 couches de 60 neurones, activation ReLu, 10^5 itérations de G.S. avec 4000 points.

Un exemple de simulation

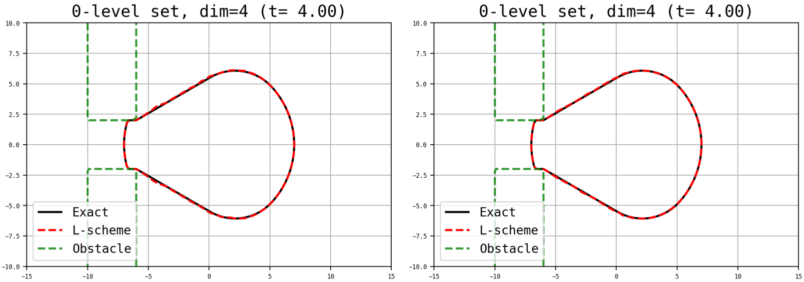


Figure – Gauche : $N_{\text{it}} = 8$, droite : $N_{\text{it}} = 16$, schéma lagrangien.

Paramètres : $b = e_1$, $c = 1/2$, *3 couches de 60 neurones, activation ReLu,* 10^5 *itérations de G.S. avec 4000 points.*

Un exemple de simulation

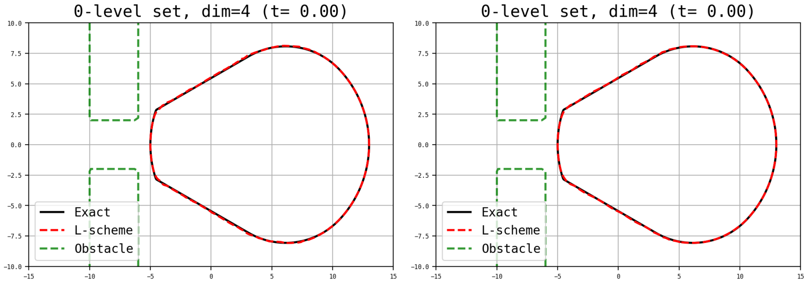


Figure – Gauche : $N_{\text{it}} = 8$, droite : $N_{\text{it}} = 16$, schéma lagrangien.

Paramètres : $b = e_1$, $c = 1/2$, 3 couches de 60 neurones, activation ReLu, 10^5 itérations de G.S. avec 4000 points.

Erreur

d	Erreurs globales		Erreurs locales		Temps
	L_∞	L_1 rel.	L_∞	L_1 rel.	
2	2.66e-01	5.99e-03	1.19e-01	4.61e-02	3h02
4	3.90e-01	6.77e-03	1.16e-01	2.69e-02	8h13
6	9.69e-01	1.09e-02	1.78e-01	2.88e-02	35h20

Table – Erreurs quand d augmente.

Paramètres : $b = e_1$, $c = 1/2$, *3 couches de 60 neurones, activation ReLu,* 10^5 *itérations de G.S. avec 4000 points.*

Conclusion & perspectives

Fait :

- Schéma lagrangien pour des problèmes d'obstacles.

Conclusion & perspectives

Fait :

- Schéma lagrangien pour des problèmes d'obstacles.
- D'une part, résultat de convergence à nombre d'itérations fixé.

Conclusion & perspectives

Fait :

- Schéma lagrangien pour des problèmes d'obstacles.
- D'une part, résultat de convergence à nombre d'itérations fixé.
- D'autre part, simulations numériques encourageantes.

Conclusion & perspectives

Fait :

- Schéma lagrangien pour des problèmes d'obstacles.
- D'une part, résultat de convergence à nombre d'itérations fixé.
- D'autre part, simulations numériques encourageantes.

À faire :

- Influence de la mesure d'entraînement μ.

Conclusion & perspectives

Fait :

- Schéma lagrangien pour des problèmes d'obstacles.
- D'une part, résultat de convergence à nombre d'itérations fixé.
- D'autre part, simulations numériques encourageantes.

À faire :

- Influence de la mesure d'entraînement μ.
- Schéma d'ordre > 1 pour EDO à second membre discontinu !

MERCI

Albert Altarovici, Olivier Bokanowski, and Hasnaa Zidani.
A general Hamilton-Jacobi framework for non-linear state-constrained control problems.
ESAIM : Control, Optimisation and Calculus of Variations, 19(2) :337–357, April 2013.

László Györfi, Michael Kohler, Adam Krzyżak, and Harro Walk.
A Distribution-Free Theory of Nonparametric Regression.
Springer Series in Statistics. Springer New York, New York, NY, 2002.

Ugo Tanielian, Maxime Sangnier, and Gerard Biau.
Approximating Lipschitz continuous functions with GroupSort neural networks.
2020.