

La programmation dynamique *stochastique* pour contrôler un système de stockage d'énergie

Exemple du lissage de la production d'un houlogénérateur

Pierre Haessig

CentraleSupélec — IETR
pierre.haessig@centralesupelec.fr
<http://pierreh.eu>

Réunion du groupe Commande des Systèmes Électriques
(CSE, inter GdR MACS-SEEDS), 9 avril 2015

d'après : P. Haessig, T. Kovaltchouk, B. Multon, H. Ben Ahmed, S. Lascaud
"Computing an Optimal Control Policy for an Energy Storage", EuroSciPy 2013
<http://arxiv.org/abs/1404.6389>

Plan de la présentation

1. Introduction
2. Exemple du lissage de la production d'un houlogénérateur
 - Contexte de la récupération de l'énergie des vagues
 - Un houlogénérateur : le SEAREV
 - Lissage de la puissance produite
3. Mise en œuvre de la programmation dynamique
 - Modélisation : dynamiques déterministes et stochastiques
 - Résolution : équation de la programmation dynamique
4. Conclusion

Plan de la présentation

1. Introduction
2. Exemple du lissage de la production d'un houlogénérateur
3. Mise en œuvre de la programmation dynamique
4. Conclusion

Mon parcours

Pierre Haessig

- Professeur assistant à CentraleSupélec depuis septembre 2014 à Rennes, dans l'équipe d'automatique ASH de l'IETR (Hervé Guéguen).
- Thèse sur les systèmes éolien-stockage (gestion & dimensionnement), laboratoire SATIE (ENS Rennes) et EDF R&D.



StoDynProg : un package pour l'Optim. Dynamique

En travaillant sur la gestion d'un système de **stockage d'énergie pour l'éolien**, je progressivement réalisé que :

- le problème de gestion d'énergie optimale est un problème d'**Optimisation Dynamique** (structure spécifique).
- il existe la méthode de la **Programmation Dynamique** (DP) pour le traiter.
- les algorithmes de DP sont “trop simples pour mériter une implémentation” → pas de “DP toolbox” !!

StoDynProg : un package pour l'Optim. Dynamique

En travaillant sur la gestion d'un système de **stockage d'énergie pour l'éolien**, je progressivement réalisé que :

- le problème de gestion d'énergie optimale est un problème d'**Optimisation Dynamique** (structure spécifique).
- il existe la méthode de la **Programmation Dynamique** (DP) pour le traiter.
- les algorithmes de DP sont “**trop simples pour mériter une implémentation**” → pas de “DP toolbox” !!

→ Écriture d'un code générique pouvant résoudre tous *mes* problèmes et, avec un peu de chance, d'autres problèmes d'Optimisation Dynamique : StoDynProg (en Python).

Pour tester cet objectif de généricité, j'ai choisi un problème différent, en lien avec les travaux du SATIE à Rennes : *le lissage de la production d'un houlogénérateur*.

Plan de la présentation

1. Introduction
2. Exemple du lissage de la production d'un houlogénérateur
 - Contexte de la récupération de l'énergie des vagues
 - Un houlogénérateur : le SEAREV
 - Lissage de la puissance produite
3. Mise en œuvre de la programmation dynamique
4. Conclusion

Récupération de l'Énergie des Vagues



(CC-BY-NC picture by polandeze)

www.flickr.com/photos/polandeze/3151015577

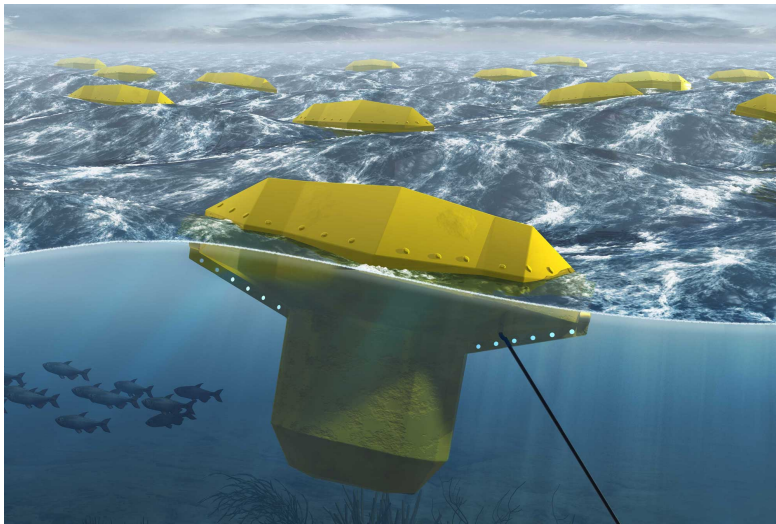
Produire de l'électricité à partir de la houle en électricité (avec de "grosses machines") est un sujet actif de R&D.

Il n'y a pas encore de machines industrialisées (à l'inverse de l'éolien et du PV), mais une *grande variété de prototypes* : les **Houlogénérateurs**.



4 m wide, 180 m long

Un Houlogénérateur : le SEAREV

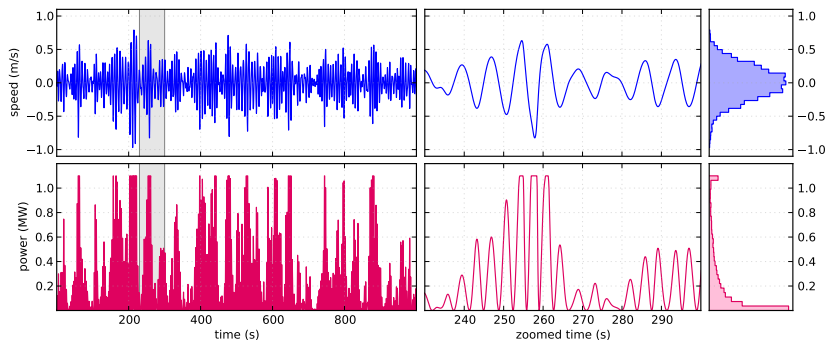


Design hydromécanique de Centrale Nantes.

Équipe du SATIE à Rennes impliquée dans la conception de la génératrice électrique.

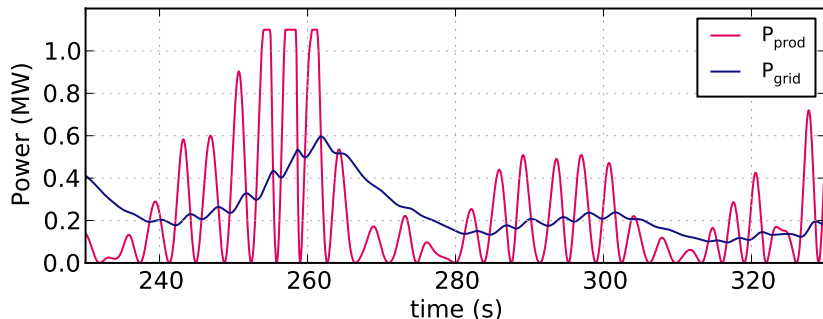
Un Houlogénérateur : le SEAREV

une production très fluctuante



Le SEAREV est un double pendule qui oscille avec les vagues.
Une génératrice électrique freine la roue interne pour générer de la puissance ($P_{prod} = T(\Omega) \times \Omega$).

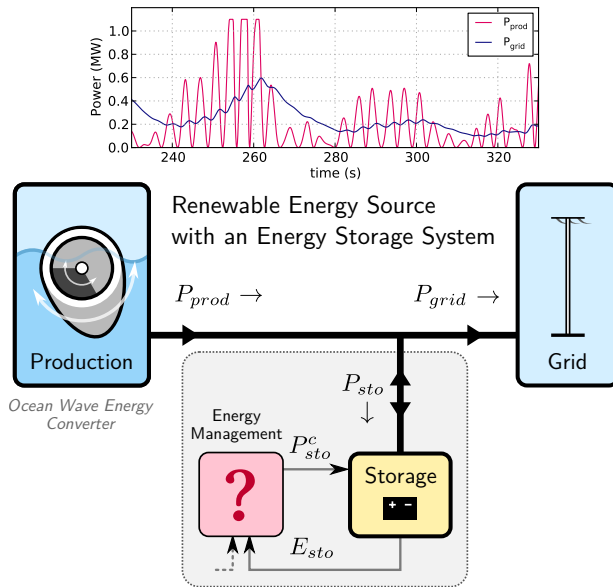
Lissage de puissance



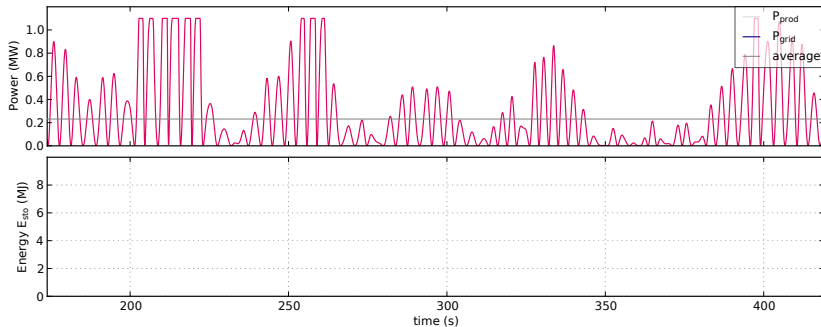
Objectif de cette application

On souhaite *lisser les variations* de la puissance produite.
Cela nécessite un **réservoir tampon d'énergie** pour
compenser la différence $P_{prod} - P_{grid}$.

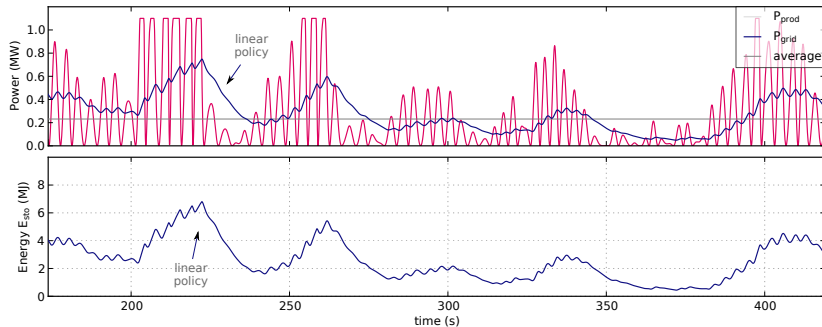
Lissage de puissance par un Système de Stockage d'Énergie



Lissage : Gestion du Stock d'Énergie



Lissage : Gestion du Stock d'Énergie

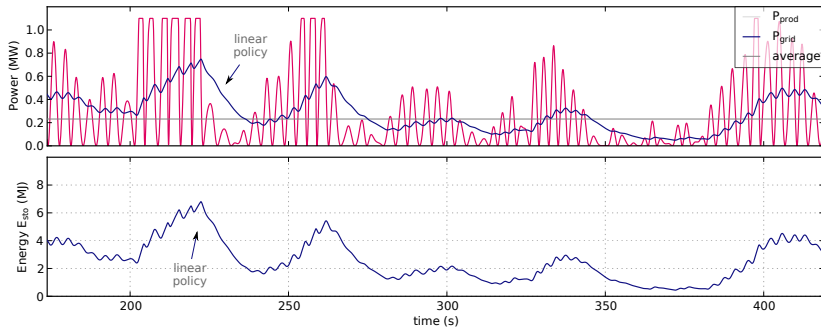


D'abord avec une loi de gestion empirique simple (linéaire) :

$$P_{grid}(t) = \frac{P_{max}}{E_{rated}} E_{sto}(t)$$

...bon résultat, mais stockage sous-utilisé → pourrait **faire mieux**.

Lissage : Gestion du Stock d'Énergie

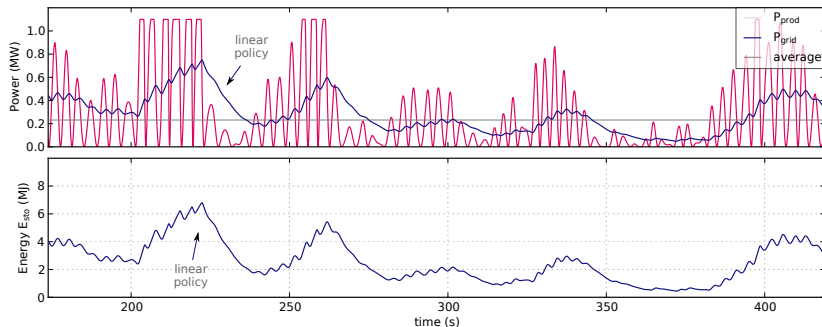


Le “faire mieux” est défini avec une fonction coût additive qui pénalise les variations de P_{grid} :

$$J = \frac{1}{N} \mathbb{E} \left\{ \sum_{k=0}^{N-1} cost(P_{grid}(k) - P_{avg}) \right\} \quad \text{avec } N \rightarrow \infty$$

et ce coût J doit être minimisé.

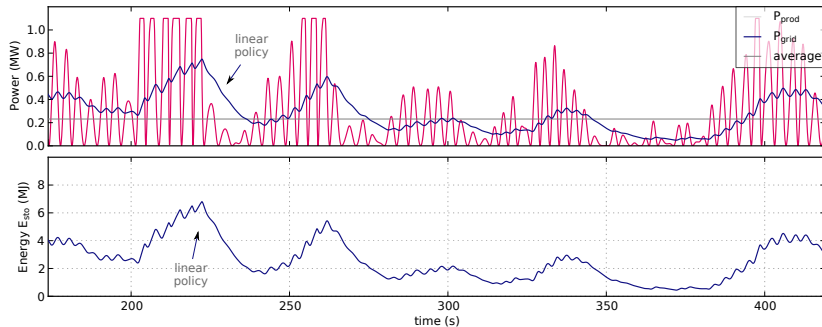
Lissage : Gestion du Stock d'Énergie



Le contrôle du stockage (choix de P_{grid} à chaque pas de temps) pour minimiser une fonction coût est un problème d'**Optimisation Dynamique Stochastique**.

(aussi appelé "Contrôle Optimal Stochastique", selon les points de vue)

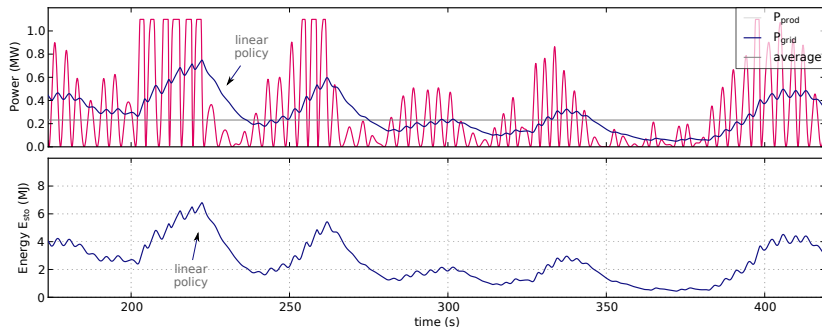
Lissage : Gestion du Stock d'Énergie



La *Programmation Dynamique* (Richard Bellman, ~1950) nous enseigne que la décision optimale est une *stratégie à retour d'état* :

$$P_{grid}(t) = \mu(x(t)) \quad \text{avec } x = (E_{sto}, \text{autres variables ?})$$

Lissage : Gestion du Stock d'Énergie

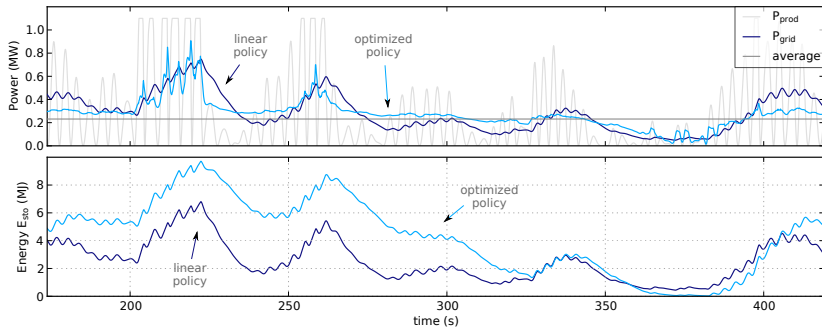


La *Programmation Dynamique* (Richard Bellman, ~1950) nous enseigne que la décision optimale est une *stratégie à retour d'état* :

$$P_{grid}(t) = \mu(x(t)) \quad \text{avec } x = (E_{sto}, \text{autres variables ?})$$

et la DP donne une *méthode pour calculer* cette loi de gestion $\mu \dots$

Lissage : Gestion du Stock d'Énergie



En appliquant la loi de gestion optimale μ^* , l'écart-type de la puissance injectée sur le réseau est réduit de $\sim 20\%$ par rapport à la gestion empirique.

Cette amélioration est "simplement" due à une utilisation plus intelligente de l'énergie stockée.

Plan de la présentation

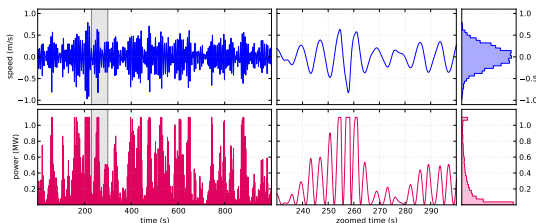
1. Introduction
2. Exemple du lissage de la production d'un houlogénérateur
3. Mise en œuvre de la programmation dynamique
 - Modélisation : dynamiques déterministes et stochastiques
 - Résolution : équation de la programmation dynamique
4. Conclusion

Modélisation des entrées incertaines

La DP accepte les perturbations stochastiques ($\rightarrow$ SDP),
à condition qu'elles soient indépendantes (e.g. décorrélées, bruit blanc).

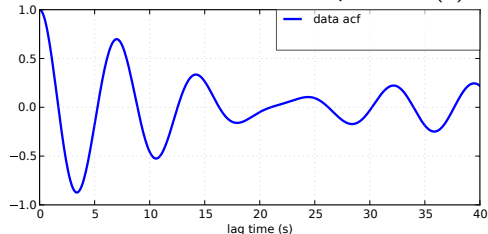
Modélisation des entrées incertaines

La DP accepte les perturbations stochastiques ($\rightarrow$ SDP),
à condition qu'elles soient indépendantes (e.g. décorrelées, bruit blanc).



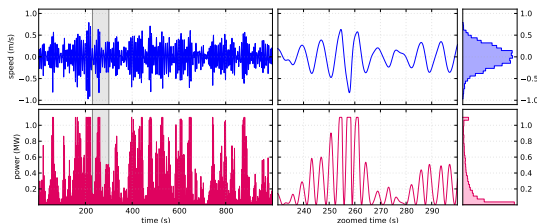
Le SEAREV présente des
pulses de puissance
(bouffées de vagues) :
 $\rightarrow$ clairement pas un
bruit blanc.

Autocorrélation de la vitesse du pendule $\Omega(k)$



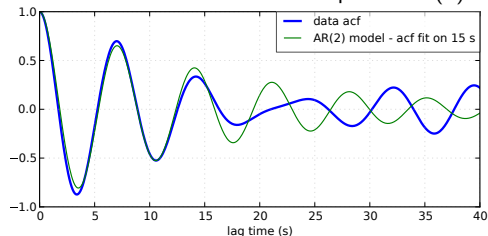
Modélisation des entrées incertaines

La DP accepte les perturbations stochastiques ($\rightarrow$ SDP),
à condition qu'elles soient indépendantes (e.g. décorrélées, bruit blanc).



Le SEAREV présente des
pulses de puissance
(bouffées de vagues) :
 $\rightarrow$ clairement pas un
bruit blanc.

Autocorrélation de la vitesse du pendule $\Omega(k)$



On approxime cette
dynamique oscillante en
ajustant un modèle
autorégressif AR(2) sur
la vitesse $\Omega(k)$.

Modélisation des dynamiques

1. La dynamique du stockage d'énergie est **déterministe** :

$$E_{sto}(k+1) = E_{sto}(k) + P_{sto}(k)\Delta t$$

Modélisation des dynamiques

1. La dynamique du stockage d'énergie est **déterministe** :

$$E_{sto}(k+1) = E_{sto}(k) + P_{sto}(k)\Delta t$$

2. La puissance du SEAREV est une dynamique **stochastique** :

- Modèle autorégressif **stochastique** pour la vitesse Ω :

$$\Omega(k) = \phi_1\Omega(k-1) + \phi_2\Omega(k-2) + w(k)$$

- AR(2) $\rightarrow$ espace d'état, avec vitesse Ω et accélération A
- Une transformation non linéaire statique donne la puissance :

$$P_{prod} = T(\Omega) \times \Omega$$

$\rightarrow$ Modèle **Markovien** de la puissance produite par le SEAREV.

Modélisation des dynamiques

1. La dynamique du stockage d'énergie est **déterministe** :

$$E_{sto}(k+1) = E_{sto}(k) + P_{sto}(k)\Delta t$$

2. La puissance du SEAREV est une dynamique **stochastique** :

- Modèle autorégressif **stochastique** pour la vitesse Ω :

$$\Omega(k) = \phi_1\Omega(k-1) + \phi_2\Omega(k-2) + w(k)$$

- AR(2) $\rightarrow$ espace d'état, avec vitesse Ω et accélération A
- Une transformation non linéaire statique donne la puissance :

$$P_{prod} = T(\Omega) \times \Omega$$

$\rightarrow$ Modèle **Markovien** de la puissance produite par le SEAREV.

Vecteur d'état complet : $x = (E_{sto}, \Omega, A)$

$\rightarrow$ dynamique $x_{k+1} = f(x_k, u_k, w_k)$

Équation de la Programmation Dynamique

Au final, le problème d'optimisation revient à résoudre l'équation de la programmation dynamique (ou “de Bellman”) :

$$J^* + \tilde{J}(x) = \min_{u \in U(x)} \mathbb{E}_w \left\{ \underbrace{\text{cost}(x, u, w)}_{\text{coût instantané}} + \underbrace{\tilde{J}(f(x, u, w))}_{\text{coût du futur}} \right\}$$

x : état, u : commande, w : perturbation stochastique

- C'est une équation *fonctionnelle* : doit être résolue **pour tout x**
- La loi de gestion optimale $\mu : x \mapsto u$ apparaît en argmin .

Équation de la Programmation Dynamique

Au final, le problème d'optimisation revient à résoudre l'équation de la programmation dynamique (ou "de Bellman") :

$$J^* + \tilde{J}(x) = \min_{u \in U(x)} \mathbb{E}_w \left\{ \underbrace{\text{cost}(x, u, w)}_{\text{coût instantané}} + \underbrace{\tilde{J}(f(x, u, w))}_{\text{coût du futur}} \right\}$$

x : état, u : commande, w : perturbation stochastique

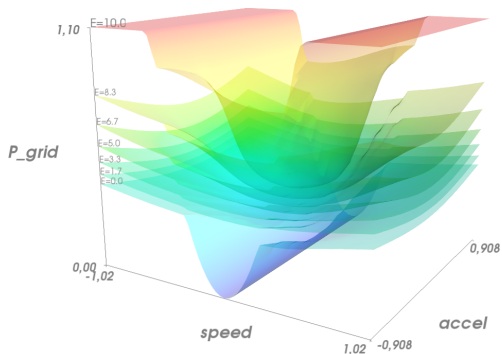
- C'est une équation *fonctionnelle* : doit être résolue **pour tout x**
- La loi de gestion optimale $\mu : x \mapsto u$ apparaît en argmin .

En pratique, numériquement

L'équation est résolue sur une **grille discrète** de l'espace d'état.
Avec $x \in \mathbb{R}^n$, $\tilde{J}$ et μ sont des tableaux de dimension **n** .

Visualisation de la loi de gestion $P_{grid}^*(E_{sto}, \Omega, A)$

La loi de commande optimale est une fonction $\mathbb{R}^3 \mapsto \mathbb{R}$, ou numériquement un tableau 3D.



Surfaces P_{grid}^* (vitesse Ω , accélération A),
pour différents niveaux d'énergie E_{sto}

Plan de la présentation

1. Introduction
2. Exemple du lissage de la production d'un houlogénérateur
3. Mise en œuvre de la programmation dynamique
4. Conclusion

Conclusion

À propos de l'intérêt de la Programmation Dynamique (DP)

Pros :

- La DP permet de calculer une loi de commande **non linéaire**, **non intuitive** (pas d'a priori sur la forme)
- La loi de gestion est calculée **une fois pour toutes** (offline)

Conclusion

À propos de l'intérêt de la Programmation Dynamique (DP)

Pros :

- La DP permet de calculer une loi de commande **non linéaire**, **non intuitive** (pas d'a priori sur la forme)
- La loi de gestion est calculée **une fois pour toutes** (offline)

Cons :

- Le comportement obtenu est **sensible** à des erreurs de modélisation (e.g. non linéarité du pendule)
- Applicabilité de la DP limitée par la **dimension de l'état** ($\dim(X) \leq 4$)

Conclusion

À propos de l'intérêt de la Programmation Dynamique (DP)

Pros :

- La DP permet de calculer une loi de commande **non linéaire**, **non intuitive** (pas d'a priori sur la forme)
- La loi de gestion est calculée **une fois pour toutes** (offline)

Cons :

- Le comportement obtenu est **sensible** à des erreurs de modélisation (e.g. non linéarité du pendule)
- Applicabilité de la DP limitée par la **dimension de l'état** ($\dim(X) \leq 4$)

Code et données disponibles sur GitHub

<https://github.com/pierre-haessig/stodynprog/tree/master/examples/>