

# Apprentissage par Renforcement Hors Ligne (BATCH)

Alexandre Gerussi, Léo Pérard, Lucas Seguinot

M2 MOCAD - IIR

15 décembre 2015

# Plan de l'exposé

- 1 Introduction
- 2 Principes généraux
- 3 Kernel Based Approximate Dynamic Programming
- 4 Fitted Q-iteration
- 5 Least-Squares Policy Iteration

# Pourquoi batch ?

- en ligne: interactions libres, voir illimitées avec l'environnement
- pas toujours possible
  - sondages
  - "conduite de vélo": nécessite un opérateur humain
  - ?? : casse du matériel en cas d'échec

# Types de batch

- pure batch
- growing batch
- semi-batch

# Principes généraux

- utilisation maximale de l'expérience déjà acquise
- experience replay: faire converger sans explorer
- fitting: accélérer et stabiliser la propagation en globalisant les mises à jour

# Kernel Based Approximate Dynamic Programming

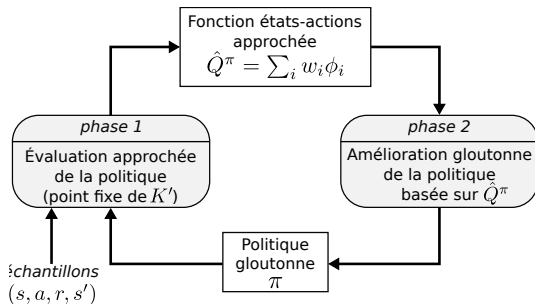
- ??

# Fitted Q-iteration

- ??

# Least-Squares Policy Iteration

$$K'(Q) = \mathcal{P}_Q^\perp(K(Q))$$





# Équilibre et conduite d'un vélo

- rester debout et atteindre un but en vélo
- valeurs sous contrôle:
  - force rotatoire à appliquer au guidon
  - placement du centre de masse par rapport au vélo
- pure hors-ligne
- dizaine de milliers de trajectoires effectuées aléatoirement
- experience replay: quelques passes de l'ensemble des données font converger